Documentation 7.4

ZABBIX

25.05.2026

Contents

Podręcznik użytkownika **5**

Witaj w dokumentacji Zabbix **5**

Rozpocznij pracę z Zabbix . . . 5
Zabbix Cloud . . . 5
Centrum dewelopera . . . 5
Społeczność i inne zasoby . . . 5
1 Wprowadzenie . . . 5
1 Struktura dokumentacji . . . 6
2 Czym jest Zabbix . . . 6
3 Funkcjonalności Zabbix . . . 7
4 Przegląd Zabbixa . . . 8
5 Co nowego w Zabbix 7.4.0 . . . 9
6 Co nowego w Zabbix 7.4.x . . . 18
2 Definicje . . . 20
3 Procesy Zabbix . . . 22
1 Server . . . 22
2 Agent . . . 31
3 Agent 2 . . . 34
4 Proxy . . . 38
5 Java gateway . . . 43
6 Sender . . . 46
7 Get . . . 47
8 JS . . . 48
9 Usługa sieciowa . . . 49
4 Instalacja i pierwsze kroki . . . 50
1 Pobieranie Zabbix . . . 50
2 Wymagania . . . 51
3 Installation from sources . . . 60
4 Installation from packages . . . 76
5 Installation from containers . . . 90
6 Instalacja interfejsu webowego . . . 97
7 Procedura aktualizacji . . . 104
8 Znane problemy . . . 116
9 Zmiany szablonów . . . 127
10 Uwagi dotyczące aktualizacji do wersji 7.4.0 . . . 128
11 Uwagi dotyczące aktualizacji do wersji 7.4.x . . . 131
5 Szybki start . . . 132
1 Logowanie i konfiguracja użytkownika . . . 132
2 Nowy host . . . 137
3 Nowa pozycja . . . 139
4 Nowy wyzwalacz . . . 140
5 Otrzymywanie powiadomienia o problemie . . . 142
6 Nowy szablon . . . 146
6 Urządzenie Zabbix . . . 148
7 Konfiguracja . . . 151
1 Hosty i grupy hostów . . . 161
2 Pozycje . . . 179
3 Wyzwalacze . . . 393
4 Zdarzenia . . . 413
5 Korelacja zdarzeń . . . 416

6 Tagowanie . . . 423
7 Wizualizacja . . . 426
8 Szablony i grupy szablonów . . . 450
9 Szablony dostępne od razu . . . 450
10 Powiadomienia o zdarzeniach . . . 463
11 Makra . . . 513
12 Użytkownicy i grupy użytkowników . . . 527
13 Przechowywanie sekretów . . . 536
14 Zaplanowane raporty . . . 544
15 Eksport danych . . . 547
8 Monitorowanie usług . . . 553
1 Drzewo usług . . . 554
2 SLA . . . 558
3 Przykład konfiguracji . . . 559
9 Monitorowanie WWW . . . 563
1 Pozycje monitorowania WWW . . . 572
2 Scenariusz z życia wzięty . . . 573
10 Monitorowanie maszyn wirtualnych . . . 580
1 Klucze pozycji monitorowania VMware . . . 582
2 Kluczowe pola wykrywania maszyn wirtualnych . . . 600
3 przykłady JSON dla pozycji VMware . . . 604
4 Przykład konfiguracji monitorowania VMware . . . 609
11 Konserwacja . . . 614
12 Wyrażenia regularne . . . 619
13 Potwierdzanie problemu . . . 624
1 Tłumienie problemów . . . 627
14 Eksport/import konfiguracji . . . 628
1 Grupy szablonów . . . 629
2 Grupy hostów . . . 630
4 Hosty . . . 647
5 Mapy sieciowe . . . 663
6 Typy mediów . . . 672
15 Odkrywanie . . . 678
1 Wykrywanie sieci . . . 679
2 Autorejestracja agenta aktywnego . . . 687
3 Odkrywanie niskiego poziomu . . . 691
16 Monitorowanie rozproszone . . . 745
1 Proxy . . . 746
17 Szyfrowanie . . . 754
1 Korzystanie z certyfikatów . . . 761
2 Używanie kluczy współdzielonych z wyprzedzeniem . . . 768
3 Rozwiązywanie problemów . . . 771
18 Interfejs webowy . . . 773
1 Menu . . . 774
2 Sekcje frontend . . . 781
3 Ustawienia użytkownika . . . 1013
4 Wyszukiwanie globalne . . . 1018
5 Tryb konserwacji frontendu . . . 1020
6 Parametry strony . . . 1021
7 Definicje . . . 1024
8 Tworzenie własnego motywu . . . 1024
9 Tryb debugowania . . . 1025
10 Ciasteczka używane przez Zabbix . . . 1026
11 Strefy czasowe . . . 1027
12 Resetowanie hasła . . . 1028
13 Selektor okresu czasu . . . 1028
19 Najlepsze praktyki . . . 1030
1 Najlepsze praktyki bezpieczeństwa . . . 1030
2 Najlepsze praktyki konfiguracji . . . 1038
20 API . . . 1039
Odniesienie do metody . . . 1044
Dodatek 1. Komentarz referencyjny . . . 1852
Dodatek 2. Zmiany z 7.2 do 7.4 . . . 1858

Dodatek 3. Zmiany w wersji 7.4 . . . 1860
21 Rozszerzenia . . . 1860
1 Moduły ładowalne . . . 1862
2 Wtyczki . . . 1871
3 Moduły frontend . . . 1880
22 Aneksy . . . 1881
1 Instalacja i konfiguracja . . . 1881
2 Konfiguracja procesów . . . 1938
3 Protokoły . . . 2021
4 Pozycje . . . 2046
5 Obsługiwane funkcje . . . 2083
6 Makra . . . 2115
7 Symbole jednostek . . . 2150
8 Składnia okresu czasu . . . 2151
9 Wykonywanie poleceń . . . 2152
10 Zgodność wersji . . . 2153
11 Dynamiczna biblioteka łącza Zabbix sender dla Windows . . . 2154
12 Aktualizacja monitorowania usług . . . 2155
13 Inne problemy . . . 2155
14 Porównanie agent vs agent 2 . . . 2156
15 Przykłady ucieczki znaków . . . 2157
23 Szybkie przewodniki referencyjne . . . 2159
Omówienie . . . 2159
1. Monitor Linux with Zabbix agent . . . 2159
2 Monitorowanie systemu Windows za pomocą agent Zabbix . . . 2163
3. Monitor Apache via HTTP . . . 2167
4. Monitor MySQL with Zabbix agent 2 . . . 2171
5 Monitoruj VMware za pomocą Zabbix . . . 2179
6 Monitoruj ruch sieciowy za pomocą Zabbix . . . 2183
7 Monitoruj ruch sieciowy za pomocą aktywnych kontroli . . . 2186
8 Monitorowanie stron internetowych za pomocą pozycji przeglądarki . . . 2189
9 Monitorowanie certyfikatów stron internetowych za pomocą agent Zabbix 2 (pasywnie) . . . 2194
10 Monitorowanie przełącznika sieciowego lub routera za pomocą Zabbix . . . 2198
11 Monitorowanie dziennika zdarzeń Windows przy użyciu aktywnych kontroli . . . 2205

Zabbix Cloud **2209**

Zabbix Cloud **2209**
Konfiguracja i zarządzanie Zabbix Cloud . . . 2209
Dowiedz się więcej o Zabbix Cloud . . . 2209
Wdróż Zabbix w chmurze . . . 2209
Konfiguracja węzła . . . 2213
Przegląd . . . 2213
Karta Przegląd . . . 2214
Karta filtrów dostępu . . . 2215
Karta Szyfrowanie . . . 2216
Karta Kopie zapasowe . . . 2217
Karta Historia . . . 2217
Karta Konserwacja . . . 2218
Karta Upgrade . . . 2219
Dodawanie użytkowników . . . 2221
Wprowadzenie . . . 2221
Dodawanie użytkowników do organizacji . . . 2221
Dodawanie użytkowników do węzłów . . . 2222
Kluczowe różnice w Zabbix Cloud . . . 2223
Wprowadzenie . . . 2223
Dziennik audytu . . . 2225

Centrum dewelopera **2227**

Centrum deweloperskie **2227**
Twórz na Zabbix . . . 2227
Moduły . . . 2227
Struktura plików modułu . . . 2228

Widżety . . . 2235
Poradniki . . . 2243
Przykłady . . . 2270
Biblioteka Python dla Zabbix . . . 2270
Instalacja . . . 2271
Przewodnik szybkiego startu . . . 2271
Korzystanie z API Zabbix . . . 2275
Zbieranie danych z agenta Zabbix . . . 2278
Wysyłanie danych do serwera lub proxy Zabbix . . . 2279
Rejestrowanie debugowania . . . 2283
Wtyczki . . . 2283
Przykłady . . . 2286
Interfejsy wtyczek . . . 2286
Stwórz wtyczkę (samouczek) . . . 2286
Zmiany w tworzeniu rozszerzeń . . . 2291

Strony podręcznika Zabbix **2291**
zabbix_agent2 . . . 2291
zabbix_agentd . . . 2293
zabbix_get . . . 2294
zabbix_js . . . 2297
zabbix_proxy . . . 2298
zabbix_sender . . . 2300
zabbix_server . . . 2305
zabbix_web_service . . . 2307

Informacja o prawach autorskich **2308**

Podręcznik użytkownika

Witaj w dokumentacji Zabbix

Twoje podstawowe źródło informacji o pracy z monitorowaniem Zabbix — od podstawowych konfiguracji po zaawansowane ustawienia. Niniejszy podręcznik obejmuje wszystko, co jest potrzebne do instalacji, konfiguracji i obsługi Zabbix.

Co nowego w Zabbix 7.4

Rozpocznij pracę z Zabbix

Instalacja Instrukcje krok po kroku dotyczące instalacji Zabbixa na wybranej platformie, obejmujące różne systemy operacyjne i konfiguracje baz danych.

Wymagania Lista obsługiwanych platform i wymagań wstępnych dotyczących oprogramowania, która pomoże Ci przygotować środowisko do pomyślnego wdrożenia Zabbix.

Przewodniki szybkiego startu Zwięzłe, nastawione na wykonanie zadania instrukcje krok po kroku, które przeprowadzą Cię przez podstawy — od pierwszych kroków konfiguracji po otrzymanie pierwszego alertu o problemie.

Zabbix Cloud

Rozpocznij korzystanie z Zabbix Cloud Instrukcje krok po kroku, które pomogą szybko uruchomić instancję Zabbix Cloud, w tym przeprowadzić wstępną konfigurację i podłączyć monitorowane urządzenia.

Zabbix Cloud a on-premises Porównaj różnice w zarządzaniu, skalowalności i utrzymaniu między tymi dwoma wdrożeniami i dowiedz się, które z nich może najpłynniej wspierać Twój przepływ pracy.

Poznaj Zabbix Cloud Zapoznaj się z przeglądem Zabbix Cloud, w tym z jego kluczowymi funkcjami, korzyściami oraz tym, co obejmuje subskrypcja. Odpowiedzi na często zadawane pytania dotyczące konfiguracji i użytkowania można znaleźć w FAQ.

Centrum dewelopera

Moduły frontend Przewodniki i materiały referencyjne dotyczące tworzenia niestandardowych modułów, które rozszerzają lub modyfikują frontend Zabbix, aby dostosować go do konkretnych przypadków użycia.

Widgety Omówienie struktury i logiki widgetów wraz z instrukcjami tworzenia niestandardowych elementów pulpitu dostosowanych do Twoich potrzeb.

Wtyczki Przegląd sposobu tworzenia i zarządzania wtyczkami Zabbix w celu rozszerzenia funkcjonalności lub integracji z systemami zewnętrznymi.

Społeczność i inne zasoby

Fora Zabbix Miejsce do wymiany pomysłów, znajdowania rozwiązań i dzielenia się doświadczeniami na wszystkich poziomach zaawansowania w Zabbix.

Blog Zabbix Aktualności, samouczki i studia przypadków przygotowane przez zespół Zabbix oraz współtwórców społeczności.

Filmy instruktażowe Biblioteka filmów zawierająca demonstracje, dyskusje i wskazówki, które pomogą Ci w pełni wykorzystać możliwości Zabbix.

1 Wprowadzenie

Zabbix to platforma monitorowania klasy enterprise o otwartym kodzie źródłowym. Monitoruje parametry w sieciach i infrastrukturze IT (od serwerów i aplikacji po bazy danych, maszyny wirtualne i usługi), a następnie wysyła alerty, gdy wystąpią problemy, jednocześnie przechowując dane historyczne do analizy i wizualizacji trendów.

Możesz tworzyć pulpity, automatyzować reakcje na problemy oraz wizualizować trendy. Zabbix można skalować od małych sieci do infrastruktur obejmujących tysiące hostów.

Niniejszy podręcznik obejmuje instalację, konfigurację i zagadnienia zaawansowane. Aby przejść do konkretnego tematu, skorzystaj z nawigacji w panelu bocznym.

1 Struktura dokumentacji

Struktura

W celu umożliwienia użytkownikowi szybkiego znalezienia interesujących go informacji, treść tego manuala została podzielona na rozdziały i podrozdziały.

Przeglądając określony rozdział, upewnij się, że rozwinąłeś jego menu. Dzięki temu zobaczysz wszystkie informacje, które są zawarte w jego podrozdziałach.

Żeby upewnić się, że dodatkowe informacje powiązane z określonym tematem nie zostaną pominięte przez użytkownika, w treści poszczególnych stron dokumentacji możliwie często rozmieszczone są linki do innych rozdziałów,

Sekcje

Wprowadzenie zawiera ogólne informacje o bieżącym oprogramowaniu Zabbix. Przeczytanie tej sekcji powinno dostarczyć Ci kilku dobrych powodów, aby wybrać Zabbix.

Definicje Zabbix wyjaśniają terminologię używaną w Zabbix, natomiast procesy Zabbix zawierają szczegóły dotyczące komponentów Zabbix.

Sekcje Instalacja i Szybki start powinny pomóc Ci rozpocząć pracę z Zabbix. Appliance Zabbix to alternatywa, jeśli chcesz szybko przekonać się, jak wygląda korzystanie z Zabbix.

Konfiguracja jest jedną z największych i najważniejszych sekcji w tym podręczniku. Zawiera wiele istotnych wskazówek dotyczących tego, jak skonfigurować Zabbix do monitorowania Twojego środowiska — od konfiguracji hostów po zbieranie i przeglądanie kluczowych danych, a także konfigurację powiadomień i zdalnych poleceń, które mają zostać wykonane w przypadku problemów.

Monitorowanie usług opisuje, jak używać Zabbix do uzyskania ogólnego obrazu środowiska monitorowania.

Monitorowanie WWW powinno pomóc Ci nauczyć się monitorować dostępność witryn internetowych.

Monitorowanie maszyn wirtualnych przedstawia instrukcję konfiguracji monitorowania środowiska VMware.

Konserwacja, Wyrażenia regularne, Potwierdzanie problemów oraz Eksport/import konfiguracji to kolejne sekcje, które pokazują, jak korzystać z tych różnych aspektów oprogramowania Zabbix.

Odkrywanie zawiera instrukcje dotyczące konfiguracji automatycznego wykrywania urządzeń sieciowych, aktywnych agentów, systemów plików, interfejsów sieciowych itp.

Monitorowanie rozproszone dotyczy możliwości używania Zabbix w większych i bardziej złożonych środowiskach.

Szyfrowanie pomaga wyjaśnić możliwości szyfrowania komunikacji między komponentami Zabbix.

Interfejs WWW zawiera informacje specyficzne dla korzystania z interfejsu WWW Zabbix.

Dobre praktyki zawierają przydatne wskazówki dotyczące optymalnego i bezpiecznego korzystania.

Sekcja API przedstawia szczegóły pracy z API Zabbix.

Rozszerzenia to zbiór sposobów rozszerzania Zabbix.

Szczegółowe listy informacji technicznych znajdują się w Aneksach.

Przewodniki szybkiego odniesienia zawierają instrukcje krok po kroku dotyczące wykonywania zadań w Zabbix.

2 Czym jest Zabbix

Przegląd

Zabbix został stworzony przez Alexeia Vladisheva i obecnie jest aktywnie rozwijany oraz wspierany przez Zabbix SIA.

Zabbix to korporacyjne, otwartoźródłowe, rozproszone rozwiązanie do monitorowania.

Zabbix to oprogramowanie, które monitoruje liczne parametry sieci oraz stan i integralność serwerów, maszyn wirtualnych, aplikacji, usług, baz danych, witryn internetowych, chmury i nie tylko. Zabbix używa elastycznego mechanizmu powiadomień, który pozwala użytkownikom konfigurować alerty e-mail dla praktycznie każdego zdarzenia. Umożliwia to szybką reakcję na problemy z

serwerem. Zabbix oferuje doskonałe funkcje raportowania i wizualizacji danych oparte na przechowywanych danych. Dzięki temu Zabbix idealnie nadaje się do planowania pojemności.

Zabbix obsługuje zarówno odpytywanie, jak i pułapkowanie. Wszystkie raporty i statystyki Zabbix, a także parametry konfiguracji, są dostępne przez frontend oparty na sieci WWW. Taki frontend oparty na sieci WWW zapewnia, że stan Twojej sieci oraz kondycja Twoich serwerów mogą być oceniane z dowolnej lokalizacji. Odpowiednio skonfigurowany Zabbix może odgrywać ważną rolę w monitorowaniu infrastruktury IT. Dotyczy to zarówno małych organizacji z kilkoma serwerami, jak i dużych firm z wieloma serwerami.

Zabbix jest bezpłatny. Zabbix jest napisany i rozpowszechniany na licencji AGPL-3.0. Oznacza to, że jego kod źródłowy jest swobodnie rozpowszechniany i dostępny dla ogółu społeczeństwa.

Dostępne jest wsparcie komercyjne, które zapewnia firma Zabbix oraz jej partnerzy na całym świecie.

Dowiedz się więcej o funkcjach Zabbix.

Użytkownicy Zabbix

Wiele organizacji różnych rozmiarów korzysta z Zabbix jako głównego narzędzia monitoringu infrastruktury IT.

3 Funkcjonalności Zabbix

Przegląd

Zabbix to wysoce zintegrowane rozwiązanie do monitorowania sieci, oferujące wiele funkcji w jednym pakiecie.

Zbieranie danych

- sprawdzanie dostępności i wydajności
- obsługa SNMP (zarówno trapping, jak i polling), IPMI, JMX, monitorowania VMware
- niestandardowe sprawdzenia
- zbieranie wymaganych danych w niestandardowych interwałach
- wykonywane przez serwer/proxy oraz przez agenty

Elastyczne definicje progów

- można definiować bardzo elastyczne progi problemów, zwane wyzwalaczami, odwołujące się do wartości z bazy danych backendu

Wysoce konfigurowalne alertowanie

- wysyłanie powiadomień można dostosować do harmonogramu eskalacji, odbiorcy, typu medium
- powiadomienia można uczynić znaczącymi i pomocnymi za pomocą zmiennych makro
- automatyczne działania obejmują zdalne polecenia

Wykresy w czasie rzeczywistym

- monitorowane pozycje są natychmiast przedstawiane na wykresach przy użyciu wbudowanej funkcji tworzenia wykresów

Możliwości monitorowania WWW

- Zabbix może śledzić ścieżkę symulowanych kliknięć myszą na stronie WWW i sprawdzać funkcjonalność oraz czas odpowiedzi

Rozbudowane opcje wizualizacji

- możliwość tworzenia niestandardowych wykresów, które mogą łączyć wiele pozycji w jednym widoku
- mapy sieci
- pokazy slajdów w przeglądzie w stylu dashboardu
- raporty
- widok wysokiego poziomu (biznesowy) monitorowanych zasobów

Przechowywanie danych historycznych

- dane przechowywane w bazie danych
- konfigurowalna historia
- wbudowana procedura housekeeping

Łatwa konfiguracja

- dodawanie monitorowanych urządzeń jako hostów
- hosty są automatycznie przejmowane do monitorowania po zapisaniu w bazie danych
- stosowanie szablonów do monitorowanych urządzeń

Użycie szablonów

- grupowanie sprawdzeń w szablonach
- szablony mogą dziedziczyć po innych szablonach

Wykrywanie sieci

- automatyczne wykrywanie urządzeń sieciowych
- automatyczna rejestracja agenta
- wykrywanie systemów plików, interfejsów sieciowych i OID SNMP

Szybki frontend WWW

- frontend WWW oparty na PHP
- dostępny z dowolnego miejsca
- można poruszać się po nim klikając
- dziennik audytu

API Zabbix

- API Zabbix udostępnia programowalny interfejs do Zabbix do masowych operacji, integracji z oprogramowaniem firm trzecich i innych celów.

System uprawnień

- bezpieczne uwierzytelnianie użytkowników
- niektórzy użytkownicy mogą mieć ograniczony dostęp do określonych widoków

W pełni funkcjonalny i łatwo rozszerzalny agent

- wdrażany na monitorowanych celach
- może być wdrażany zarówno w systemach Linux, jak i Windows

Demony binarne

- napisane w C, dla wydajności i niewielkiego zużycia pamięci
- łatwe do przenoszenia

Gotowy na złożone środowiska

- zdalne monitorowanie ułatwione dzięki użyciu proxy Zabbix

4 Przegląd Zabbixa

Architektura

Zabbix składa się z kilku głównych komponentów oprogramowania. Ich zadania opisano poniżej.

Serwer

Zabbix server jest centralnym komponentem, do którego agenty przekazują informacje o dostępności i integralności oraz statystyki. Serwer jest centralnym repozytorium, w którym przechowywane są wszystkie dane konfiguracyjne, statystyczne i operacyjne.

Przechowywanie bazy danych

Wszystkie informacje konfiguracyjne, a także dane zbierane przez Zabbix, są przechowywane w bazie danych.

Interfejs webowy

Aby zapewnić łatwy dostęp do Zabbix z dowolnego miejsca i z dowolnej platformy, udostępniany jest interfejs oparty na przeglądarce internetowej. Interfejs jest częścią serwera Zabbix i zwykle (ale niekoniecznie) działa na tym samym fizycznym komputerze, na którym działa serwer.

Proxy

Proxy Zabbix może zbierać dane o wydajności i dostępności w imieniu serwera Zabbix. Proxy jest opcjonalnym elementem wdrożenia Zabbix; jednak może być bardzo korzystny do rozłożenia obciążenia pojedynczego serwera Zabbix.

Agent

Agenty Zabbix są wdrażane na monitorowanych celach, aby aktywnie monitorować lokalne zasoby i aplikacje oraz przekazywać zebrane dane do serwera Zabbix. Od Zabbix 4.4 dostępne są dwa typy agentów: Zabbix agent (lekki, obsługiwany na wielu platformach, napisany w C) oraz Zabbix agent 2 (bardzo elastyczny, łatwo rozszerzalny za pomocą wtyczek, napisany w Go).

Przepływ danych

Ponadto ważne jest, aby zrobić krok wstecz i przyjrzeć się ogólnemu przepływowi danych w Zabbix.

Aby utworzyć pozycję, która zbiera dane, należy najpierw utworzyć host.

Patrząc na drugi koniec spektrum Zabbix, aby utworzyć wyzwalacz, trzeba najpierw mieć pozycję.

Aby utworzyć akcję, trzeba mieć wyzwalacz.

Jeśli więc chcesz otrzymać alert, że obciążenie CPU na *Server X* jest zbyt wysokie, najpierw musisz utworzyć wpis hosta dla *Server X*, następnie pozycję do monitorowania jego CPU, potem wyzwalacz, który aktywuje się, gdy CPU jest zbyt wysoko obciążone, a na końcu akcję, która wyśle Ci wiadomość e-mail.

Choć może się to wydawać dużą liczbą kroków, przy użyciu szablonów tak naprawdę tak nie jest.

Jednak dzięki takiej konstrukcji możliwe jest stworzenie bardzo elastycznej konfiguracji.

5 Co nowego w Zabbix 7.4.0

Zobacz breaking changes dla tej wersji.

Zagnieżdżone wykrywanie niskiego poziomu

Obecnie możliwe jest tworzenie wielopoziomowego wykrywania obiektów dzięki wprowadzeniu prototypów wykrywania w ramach reguły wykrywania niskiego poziomu. Na przykład możesz chcieć wykryć wszystkie instancje bazy danych na serwerze bazy danych, a następnie wykryć tablespace dla każdej instancji, a potem wykryć tabele dla każdego tablespace.

Prototypy wykrywania to zagnieżdżone reguły wykrywania w ramach ”nadrzędnej” reguły wykrywania. Prototypy wykrywania mają własne pozycja, wyzwalacz, wykres, host i prototypy wykrywania.

Zagnieżdżony prototyp wykrywania może używać tej samej wartości JSON co reguła nadrzędna, ale następnie korzystać z innego ”wycinka” danych z wartości JSON.

Poziomy zagnieżdżenia prototypów wykrywania są nieograniczone.

Prototypy hostów na wykrytych hostach

Prototypy hostów są teraz obsługiwane na wykrytych hostach, co umożliwia Zabbixowi automatyczne wykrywanie i monitorowanie encji wewnątrz innych wykrytych encji (np. hipernadzorców, ich maszyn wirtualnych oraz kontenerów wewnątrz tych maszyn wirtualnych).

Możesz tworzyć prototypy hostów na wykrytych hostach, tworząc reguły niskopoziomowego wykrywania z prototypami hostów lub łącząc szablon z prototypami hostów. Alternatywnie możesz połączyć szablon z prototypem hosta używanym do wykrywania, co spowoduje, że wykryte hosty odziedziczą prototypy hostów z szablonu.

Jeśli Twoja bieżąca konfiguracja zawiera prototypy hostów, które używają szablonów zawierających inne prototypy hostów, zapoznaj się z uwagami dotyczącymi aktualizacji.

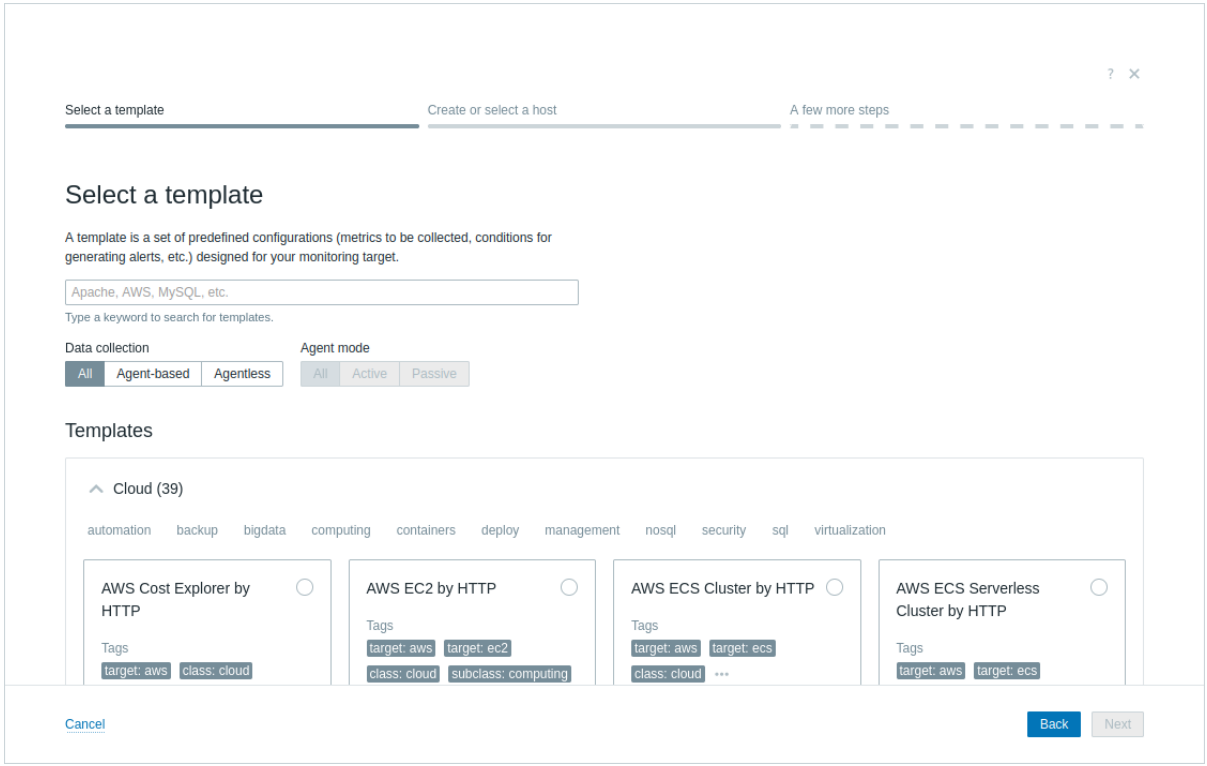
Uwierzytelnianie OAuth 2.0

Uwierzytelnianie OAuth 2.0 jest teraz obsługiwane dla protokołu SMTP. Aby skonfigurować uwierzytelnianie OAuth, wybierz ”OAuth” w parametrze Authentication podczas konfigurowania typu mediów e-mail, a następnie określ parametry dla pobierania tokenów OAuth.

Pobieranie tokenów OAuth ma zautomatyzowane funkcje dla dostawców poczty **Gmail**, **Gmail relay** i **Office365**. Wymagane jest jedynie podanie wartości parametrów *Redirection endpoint*, *Client ID* i *Client secret*. Zabbix automatycznie uzupełni pozostałe wymagane wartości (zobacz domyślne adresy URL OAuth według dostawcy).

Kreator hostów

Nowy Kreator hostów wprowadza prowadzony, krok po kroku interfejs do konfiguracji celu monitorowania (urządzenia, aplikacji, usługi itp.) w Zabbix. Upraszcza konfigurację nowych lub istniejących hostów, prowadząc użytkowników przez kluczowe kroki, takie jak wybór szablonu, instalacja agent Zabbix, dodawanie interfejsów hosta i inne.

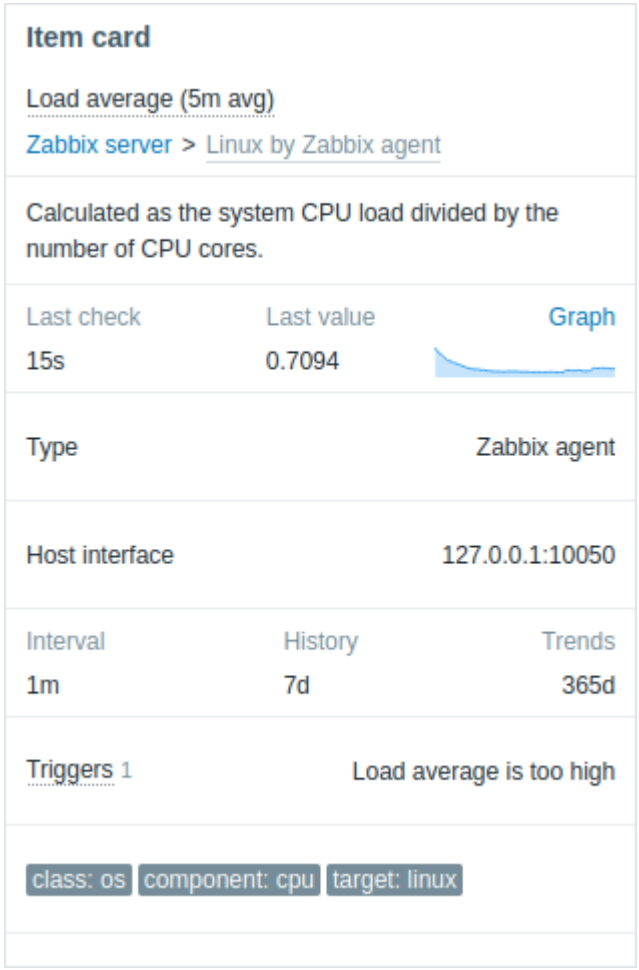


Do Kreatora hostów można uzyskać dostęp z poziomu Zbieranie danych > Hosty lub menu hosta w różnych sekcjach frontend.

Jeśli aktualizujesz z wcześniejszej wersji Zabbix, szablony muszą zostać zaktualizowane, aby działały z Kreatorem hostów. Instrukcje znajdują się w sekcji Aktualizacja szablonu.

Widżety Karta pozycji

Do widżetów pulpitu nawigacyjnego dodano widżet Karta pozycji, który umożliwia wygodne wyświetlanie pełnych informacji o pojedynczej pozycji na pierwszy rzut oka.



Historia pozycji

W widżecie Historia pozycji pozycja przewijania będzie teraz ustawiona na dole, jeśli nowe wartości są skonfigurowane tak, aby były umieszczane na dole. Jest to przydatne do odczytywania najnowszych wartości logów.

Edycja widżetów w czasie rzeczywistym

Podczas edytowania widżetów można teraz podglądać zmiany konfiguracji widżetu w czasie rzeczywistym. Dodatkowo formularze konfiguracji widżetów są teraz przeciągalne, co umożliwia ich dowolne przemieszczanie.

Należy pamiętać, że podgląd wykresu w formularzu widżetu wykresu został usunięty.

Pozycje Pozycja ICMP ping z opcją ponawiania

Dodano nową pozycję `icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]` simple check do monitorowania dostępności hosta za pomocą ping ICMP z możliwością modyfikowania liczby ponowień.

Funkcje Śledzenie znaczników czasu

Dodano nowe funkcje historii do śledzenia znaczników czasu:

- `firstclock` - znacznik czasu najstarszej wartości w zdefiniowanym okresie oceny;
- `lastclock` - znacznik czasu N-tej najnowszej wartości w zdefiniowanym okresie oceny;
- `logtimestamp` - znacznik czasu komunikatu dziennika dla N-tej najnowszej wartości pozycji dziennika.

Makra Śledzenie czasu wartości pozycji

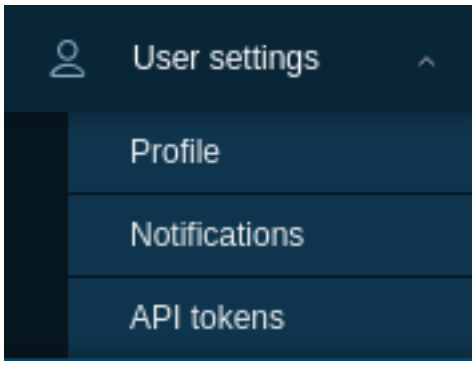
Dodano nowe makra do śledzenia czasu wartości pozycji:

Macro	Description
{ITEM.LASTVALUE.AGE}	Czas, jaki upłynął między pobraniem najnowszej wartości pozycji a oceną makra.
{ITEM.LASTVALUE.DATE}	Data pobrania najnowszej wartości pozycji.
{ITEM.LASTVALUE.TIME}	Godzina pobrania najnowszej wartości pozycji.
{ITEM.LASTVALUE.TIMESTAMP}	Znacznik czasu pobrania najnowszej wartości pozycji.
{ITEM.VALUE.AGE}	Czas, jaki upłynął między pobraniem wartości pozycji a oceną makra.

Macro	Description
{ITEM.VALUE.DATE}	Data pobrania wartości pozycji.
{ITEM.VALUE.TIME}	Godzina pobrania wartości pozycji.
{ITEM.VALUE.TIMESTAMP}	Znacznik czasu pobrania wartości pozycji.

Powiadomienia Osobna sekcja menu dla powiadomień użytkownika

Aby poprawić widoczność mediów użytkownika, powiadomienia użytkownika mają teraz własną sekcję menu w *Ustawieniach użytkownika*.



Nowa sekcja Powiadomienia zawiera dwie karty - *Media* i *Powiadomienia frontend*, które wcześniej były częścią sekcji profilu użytkownika.

Zarządzanie własnymi mediami użytkownika

Domyślnie wszyscy użytkownicy mogą teraz zarządzać własnymi mediami.

Jednak uprawnienia użytkownika do zmiany szczegółów mediów dla siebie mogą być przyznawane lub odbierane na podstawie jego roli użytkownika (zobacz opcję *Tworzenie i edycja własnych mediów*).

Dodatkowo uprawnienia użytkownika Super admin do zmiany szczegółów mediów innych osób również mogą być przyznawane lub odbierane na podstawie jego roli użytkownika (zobacz opcję *Tworzenie i edycja mediów użytkowników*).

Mapy Automatyczne ukrywanie etykiet

Teraz można skonfigurować etykiety elementów mapy/łączy tak, aby były wyświetlane tylko po najechaniu na nie kursorem lub po ich zaznaczeniu. To ustawienie pomaga zmniejszyć wizualny chaos na mapach z wieloma elementami znajdującymi się blisko siebie.

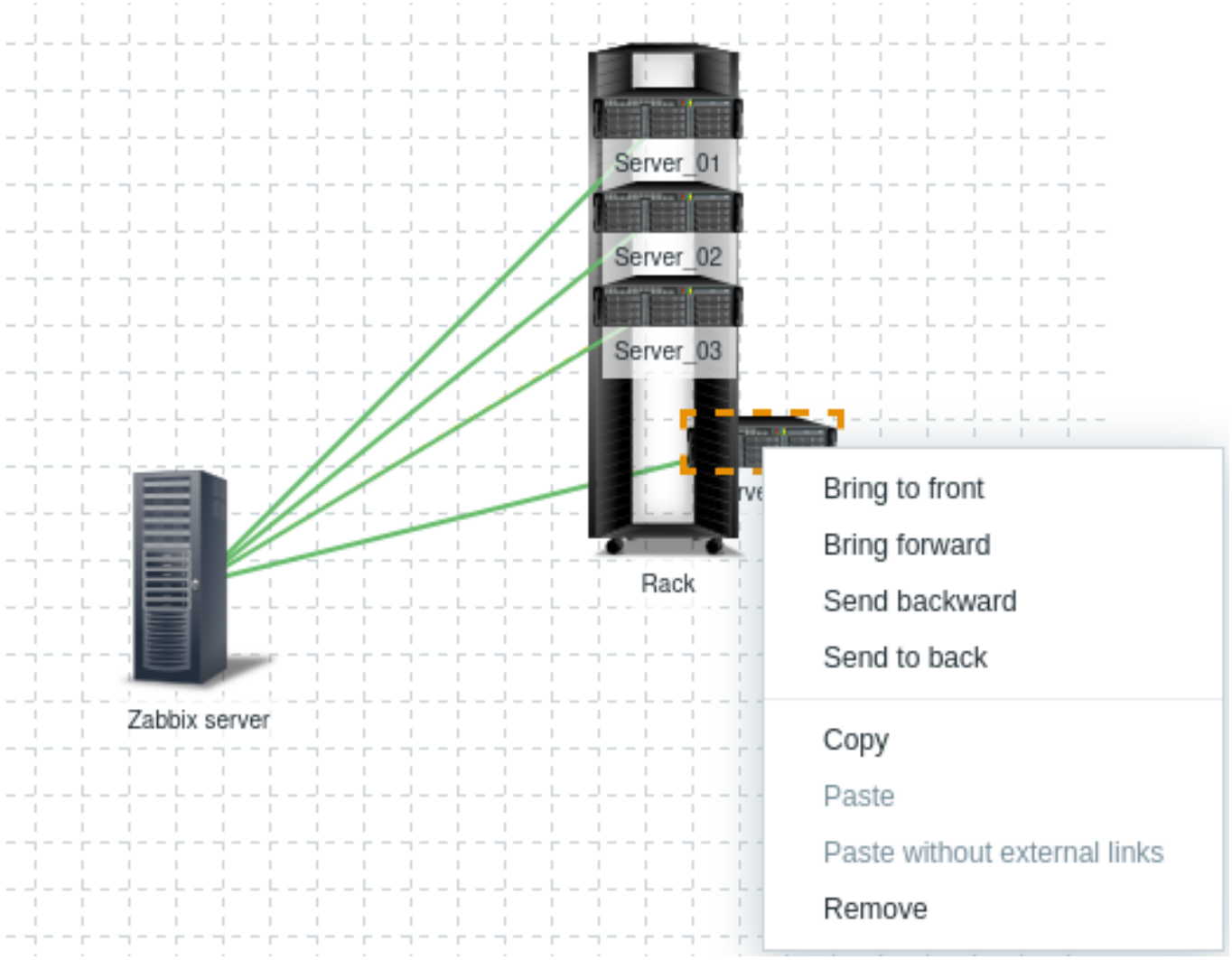
Automatyczne ukrywanie etykiet można skonfigurować globalnie dla wszystkich elementów/łączy mapy lub dla pojedynczego elementu/łącza mapy.

Proporcjonalne skalowanie obrazów tła

Obrazy tła można teraz proporcjonalnie skalować, aby dopasować je do rozmiaru mapy.

Kolejność elementów

Teraz można przenieść jeden element przed drugi (lub odwrotnie), klikając element prawym przyciskiem myszy i wybierając opcje *Przesuń do przodu*/*Przenieś na wierzch* lub *Przesuń do tyłu*/*Przenieś na spód*.



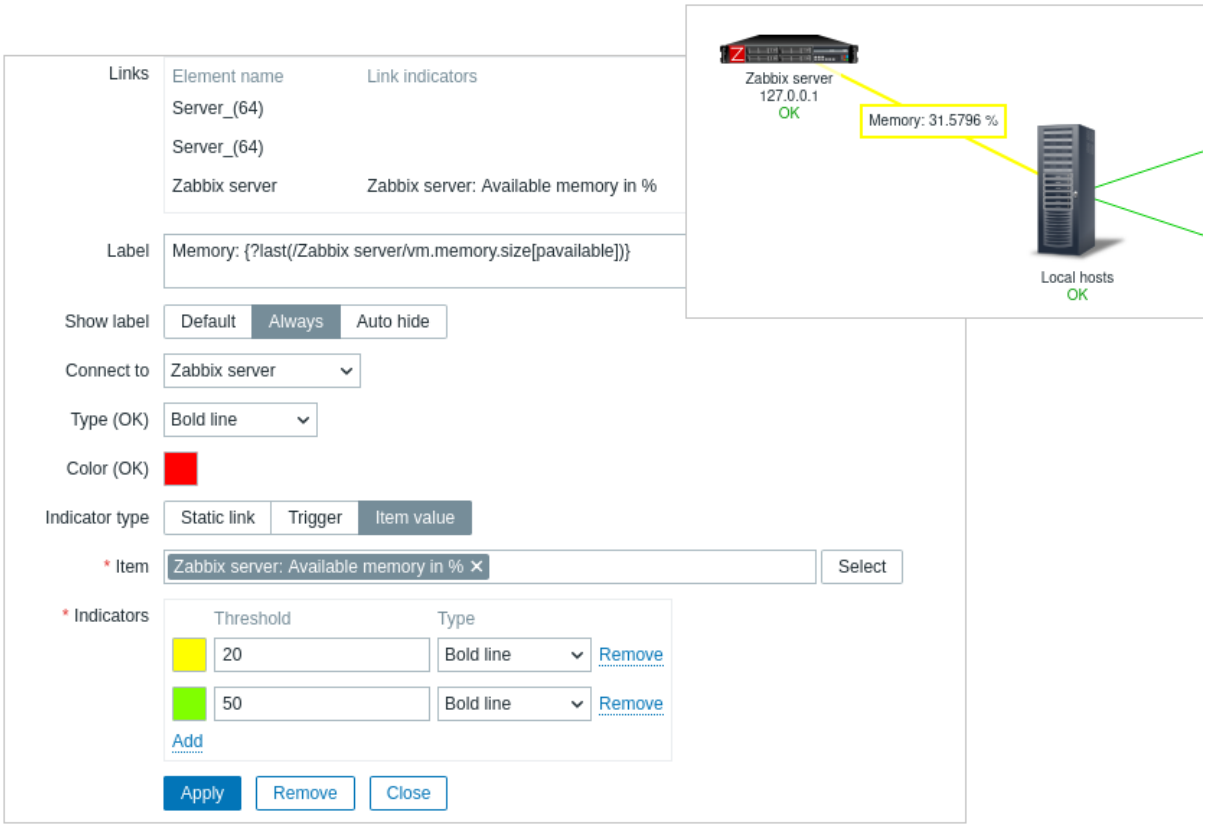
Dodatkowo, jeśli elementy mapy zawierają zarówno łącze między sobą, jak i podświetlenie, łącze będzie teraz wyświetlane pod podświetleniem.

Wskaźniki linków oparte na wartości pozycji

W poprzednich wersjach można było dostosować styl i kolor linku, jeśli jakiś wyzwalacz przechodził w stan problemu. Teraz podobna funkcjonalność jest dostępna na podstawie wartości pozycji.

Możliwe jest dostosowanie stylu i koloru linku, jeśli wartość pozycji:

- osiąga zdefiniowany próg (dla pozycji numerycznych);
- odpowiada wyrażeniu regularnemu (dla pozycji tekstowych).



Procesy Zarządzanie pamięcią podręczną historii

W niektórych scenariuszach zbierania danych określone pozycje mogą tymczasowo blokować pamięć podręczną historii serwera/proxy. Może to opóźnić zapisywanie danych historycznych do bazy danych i spowolnić działanie systemu. Aby ułatwić zarządzanie pamięcią podręczną historii, wprowadzono następujące usprawnienia:

- **Ręczne czyszczenie pamięci podręcznej:** Nowe polecenie czasu wykonywania `history_cache_clear=target` dla Zabbix serwer/proxy umożliwia ręczne wyczyszczenie pamięci podręcznej historii dla określonej pozycji na podstawie jej ID.
- **Automatyczne czyszczenie pamięci podręcznej:** Gdy wyłączysz pozycję, zostaje ona natychmiast usunięta z pamięci podręcznej historii (z wyjątkiem jej ostatniej wartości, która jest zachowywana do logów). Podobnie, gdy wyłączysz host, wszystkie jego pozycje są usuwane z pamięci podręcznej historii (z wyjątkiem ich ostatnich wartości).
- **Logowanie diagnostyki pamięci podręcznej:** Gdy pamięć podręczna historii jest pełna, Zabbix serwer/proxy zapisuje teraz informacje diagnostyczne dotyczące pamięci podręcznej historii od DebugLevel=3 (wcześniej DebugLevel=4). Log zawiera pozycje z największą liczbą wartości w pamięci podręcznej historii.

Aby zapewnić długoterminową stabilność systemu, upewnij się, że zbieranie danych jest zrównoważone względem dostępnych zasobów (wydajność bazy danych, rozmiar pamięci podręcznej, interwały zbierania, parametry pozycji logów itp.). Możesz monitorować pamięć podręczną historii Zabbix za pomocą wewnętrznej pozycji zabbix[wcache]. Możesz również rozważyć zwiększenie rozmiaru pamięci podręcznej historii dla Zabbix serwer/proxy.

Statystyki transakcji synchronizatora historii

Tytuł procesu synchronizatora historii teraz wyświetla szczegółowe statystyki dotyczące transakcji synchronizatora historii dla Zabbix serwer i proxy.

Bezpieczeństwo Szyfrowanie TLS między frontendem a serwerem

Obecnie możliwe jest szyfrowanie komunikacji między frontendem Zabbix a serwerem Zabbix przy użyciu TLS. Ta funkcja jest kontrolowana przez nowe parametry w konfiguracji serwera (TLSListen, TLSFrontendAccept, TLSFrontendCertIssuer, TLSFrontendCertSubject, FrontendAllowedIP).

Rozwiązywanie makr sejfu sekretów niezależnie przez serwer/proxy

Teraz można skonfigurować pobieranie wartości makr sekretów z sejfu przez Zabbix server i Zabbix proxy niezależnie, jeśli opcja *Resolve secret vault macros by* jest ustawiona na "Zabbix server and proxy".

Wtyczki Konfiguracja ścieżki niestandardowych zapytań dla ładowalnych wtyczek

Parametr `Plugins.*.CustomQueriesPath` w plikach konfiguracji wtyczek Zabbix agent 2 dla MySQL, Oracle i PostgreSQL, a także `Plugins.MSSQL.CustomQueriesDir` dla MSSQL, ma teraz wartość domyślną. Ta funkcjonalność jest domyślnie wyłączona

i można ją włączyć za pomocą nowo wprowadzonego parametru `Plugins.*.CustomQueriesEnabled`.

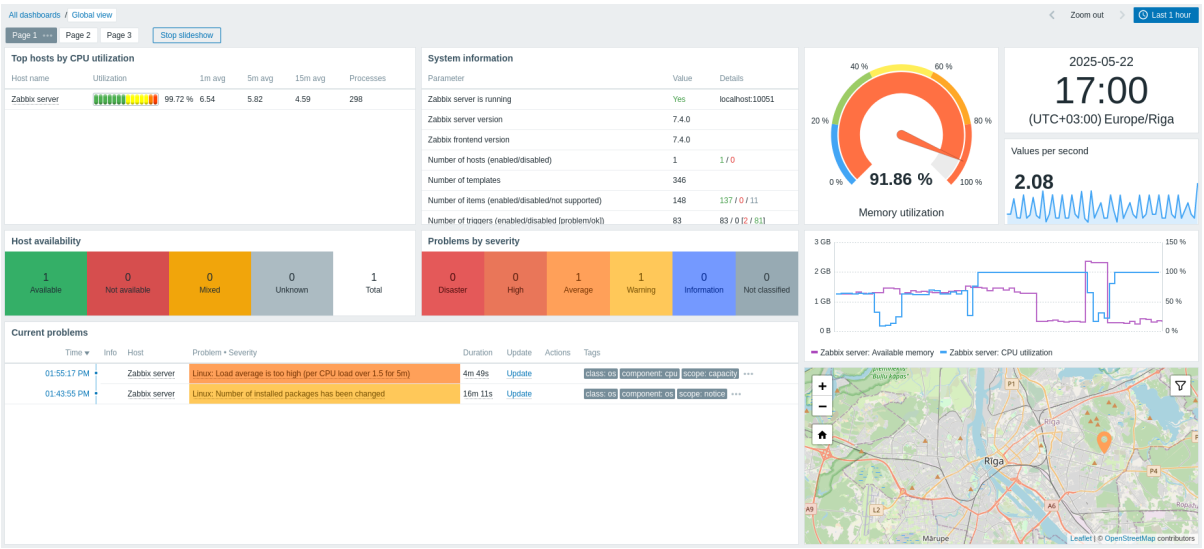
frontend Walidacja wbudowana w formularzach

Następujące formularze frontend obsługują teraz walidację wbudowaną, wyświetlając wszelkie błędy danych wejściowych natychmiast po wypełnieniu pól:

- Szablon konfiguracja
- Host konfiguracja
- Pozycja oraz prototyp pozycji konfiguracja
- Wyzwalacz oraz prototyp wyzwalacza konfiguracja
- Moduł administracja

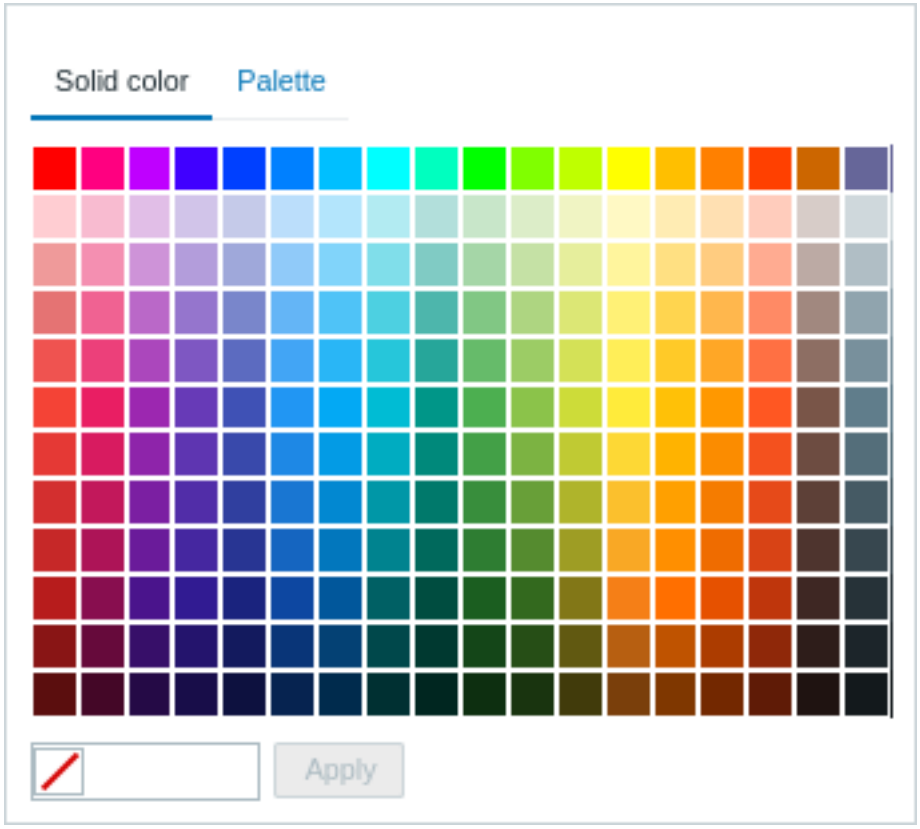
Zaktualizowano domyślny pulpit nawigacyjny

Domyślny pulpit nawigacyjny *Global view* w nowych instalacjach Zabbixa został zaktualizowany i zawiera teraz najnowsze widżety pulpitu nawigacyjnego.

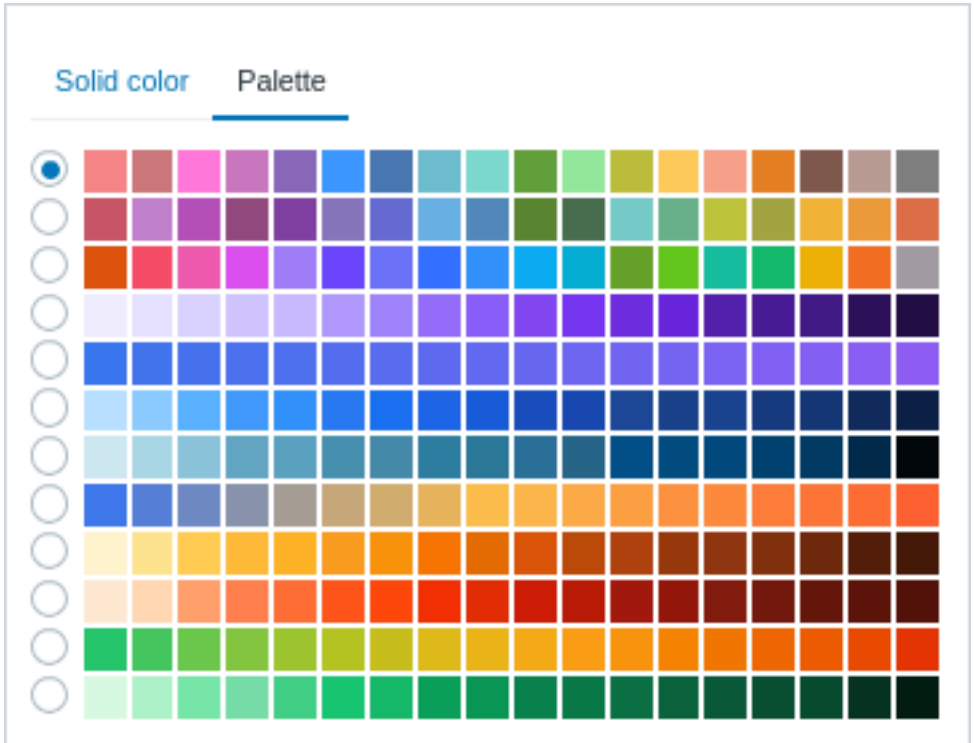


Ulepszony selektor kolorów z obsługą palety

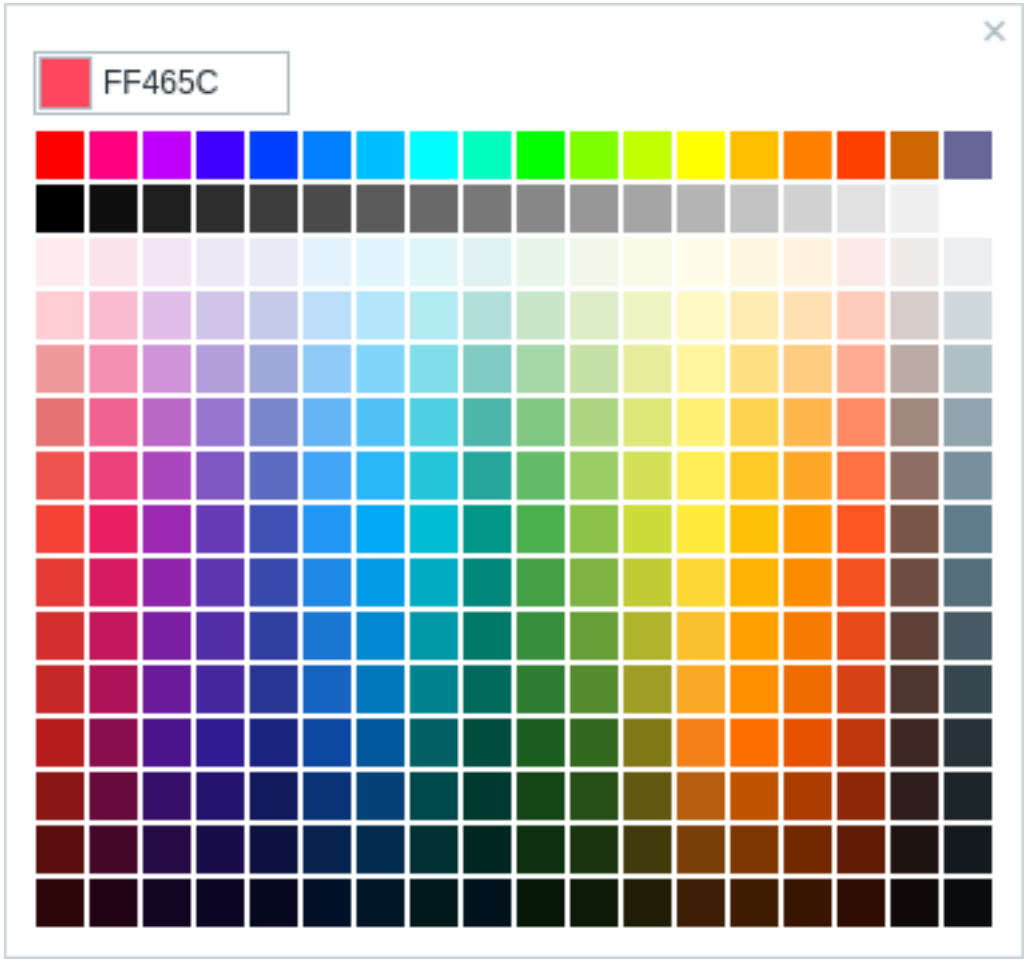
Selektor kolorów w widżetach graph i pie chart został przeprojektowany, aby poprawić wygodę użytkowania. Oprócz kolorów jednolitych i wprowadzania wartości szesnastkowych można teraz przełączyć się na kartę *Palette* i wybierać spośród predefiniowanych wierszy kolorów — każda seria otrzymuje własny, wyraźnie odróżnialny odcień, co ułatwia rozróżnianie. Selektor oferuje także pełną nawigację za pomocą klawiatury oraz walidację na bieżąco.



W Zabbix 7.4 (karta *Solid color*)



W Zabbix 7.4 (karta *Color palette*)



Przed Zabbix 7.4

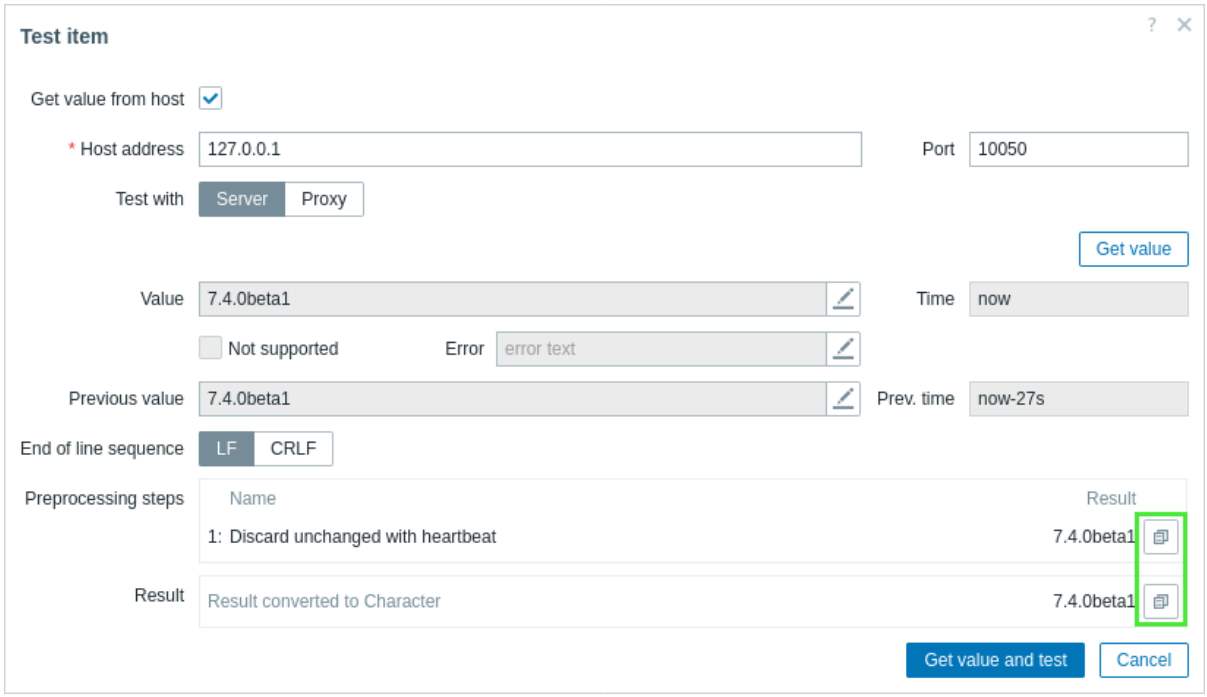
Formularze modalne

Kilka formularzy frontend jest teraz otwieranych w oknach modalnych (wyskakujących):

- konfiguracja wykresu
- konfiguracja prototypu wykresu

Łatwiejsze kopiowanie wyników testów przetwarzania wstępnego

Teraz łatwiej jest kopiować wartości w wynikach testów przetwarzania wstępnego dzięki dodanemu przyciskowi *Copy to clipboard*.



Należy zauważyć, że podobny przycisk zastępuje teraz link *Copy* dla utworzonych tokenów API.

Dokumentacja Skonsolidowane strony dokumentacji dla wydań minor

Dokumentacja wydania dla wersji minor głównego wydania Zabbix będzie teraz gromadzona na pojedynczych stronach dokumentacji odpowiednio dla nowych funkcji i uwag dotyczących aktualizacji.

6 Co nowego w Zabbix 7.4.x

Ta strona zawiera zbiorcze informacje o nowych funkcjach uwzględnionych w wydaniach pomocniczych głównej wersji Zabbix.

Zobacz także Co nowego w głównej wersji.

Informacje o zmianach w istniejących szablonach oraz o nowych szablonach obejmujących różne urządzenia, usługi i rozwiązania partnerów znajdziesz w sekcji Zmiany w szablonach.

Co nowego w Zabbix 7.4.10 Elementy alarmów VMware zwracają szczegóły encji

Zwracany przez pozycje `vmware.*.alarms.get[]` obiekt JSON zawiera teraz obiekt `entity` dla każdego alarmu. Ten obiekt zwraca identyfikator, UUID i typ dotkniętej encji (host, maszyna wirtualna lub datastore).

Jeśli masz skrypty, które analizują dane wyjściowe pozycji alarmów VMware, może być konieczne ich zaktualizowanie, aby obsługiwały dodatkowe pola.

Co nowego w Zabbix 7.4.9 Integracje webhook

- Nowa integracja umożliwia połączenie Zabbix z IBM Maximo Service Request przy użyciu typu mediów Zabbix webhook.
- Typ mediów webhook GLPI został zaktualizowany, aby obsługiwać RESTful API v2 z autoryzacją OAuth2 (domyślnie włączoną), opcjonalną obsługę starszego REST API v1 za pomocą flagi konfiguracyjnej, tworzenie zgłoszeń dla zdarzeń problemów oraz mapowanie pilności konfigurowane przez użytkownika.

Obsługa MySQL 9.6

Maksymalna obsługiwana wersja MySQL to teraz 9.6.X.

Obsługa TimescaleDB 2.26

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.26.X.

Obsługa MariaDB 12.2

Maksymalna obsługiwana wersja MariaDB to teraz 12.2.X.

Obsługa PHP 8.5

Maksymalna obsługiwana wersja PHP to teraz 8.5.

Obsługa SQLite 3.53

Maksymalna obsługiwana wersja SQLite to teraz 3.53.

Co nowego w Zabbix 7.4.8 Obsługa TimescaleDB 2.25

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.25.X.

Co nowego w Zabbix 7.4.7 Obsługa TimescaleDB 2.24

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.24.X.

Co nowego w Zabbix 7.4.6 Zobacz zmiany niekompatybilne związane z ograniczonymi uprawnieniami podczas tworzenia plików dziennika.

Obsługa TimescaleDB 2.23

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.23.X.

Obsługa MySQL 9.5

Maksymalna obsługiwana wersja MySQL to teraz 9.5.X.

Rozszerzone parametry wiersza poleceń dla pakietów instalatora MSI agent

Pakiety instalatora MSI agent Zabbix (zarówno dla Zabbix agent, jak i Zabbix agent 2) obsługują teraz następujące parametry podczas wykonywania instalacji z wiersza poleceń:

- `HOSTNAMEITEM` - określa pozycję Zabbix agent używaną do pobierania nazwy hosta
- `UNSAFEUSERPARAMETERS` - zezwala na przekazywanie w argumentach do parametrów zdefiniowanych przez użytkownika wszystkich znaków (np. ' " ` * ?)
- `STARTAGENTS` - określa liczbę wstępnie rozwidlonych instancji `zabbix_agentd`, które przetwarzają pasywne kontrole
- `DONOTSTART` - zapobiega uruchomieniu usługi Zabbix agent przez instalator MSI
- `SOURCEIP` - określa źródłowy adres IP dla połączeń wychodzących do serwer/proxy Zabbix lub dla połączeń nawiązywanych podczas wykonywania niektórych pozycji (np. `web.page.get`, `net.tcp.port`)

Maksymalny limit czasu dla pozycji ICMP ping

Dla pozycji icmpping* jest teraz stosowany zakodowany na stałe maksymalny możliwy limit czasu wynoszący 600 sekund.

Co nowego w Zabbix 7.4.5 Ta wersja minor nie zawiera żadnych zmian funkcjonalnych.

Co nowego w Zabbix 7.4.4 **Uwaga!** Aktualizacja do tej wersji nie jest zalecana ze względu na błędy związane z wykresami oraz wtyczką MySQL dla agent 2 Zabbix.

Obsługa PostgreSQL 18

Maksymalna obsługiwana wersja PostgreSQL to teraz 18.X.

Obsługa TimescaleDB 2.22

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.22.X.

Wtyczka Redis — obsługa TLS i walidacja podczas uruchamiania

Do wtyczki Redis dla Zabbix agent 2 dodano obsługę TLS.

Zaimplementowano walidację konfiguracji TLS wtyczki podczas uruchamiania oraz ulepszono komunikaty walidacyjne/błędów. Nieprawidłowa logika konfiguracji (na przykład użycie typu połączenia `verify_full` bez określenia `TLSCAFile`) może uniemożliwić uruchomienie Zabbix agent 2.

Co nowego w Zabbix 7.4.3 Obsługa nazwy TNS dla wtyczki Oracle

W natywnej wtyczce Oracle w Zabbix agent 2 dodano obsługę nazwy TNS specyficzną dla Oracle. Klucz TNS lub wartość TNS można określić na poziomie parametrów klucza pozycja item key lub w konfiguracji wtyczki (dla nazwanej sesji lub parametrów domyślnych).

Obsługa nazwy TNS umożliwia opis połączenia w środowisku klastrowej bazy danych przy użyciu jednego hosta Zabbix. Więcej informacji można znaleźć w pliku README wtyczki Oracle README.

Zwiększono maksymalny limit czasu dla zabbix_get i zabbix_js

Maksymalna wartość parametru `timeout` dla narzędzi wiersza poleceń zabbix_get i zabbix_js została zwiększona do 600 sekund.

Obsługa MariaDB 12.0

Maksymalna obsługiwana wersja MariaDB to teraz 12.0.X.

Podświetlanie całego wiersza w widoku Problemy i widżecie Problemy

Opcja *Podświetl cały wiersz* jest teraz dostępna w niezwartym widoku Monitoring > Problemy oraz w widżecie pulpitu Problemy.

Co nowego w Zabbix 7.4.2 Dodatkowa obsługa ról dla wtyczki Oracle

Wtyczka Oracle pozycje dla Zabbix agent 2 obsługuje teraz dodatkowe role, takie jak SysBACKUP, SysDG, SysKM lub SysRAC, w opcjach logowania.

Automatyczne ponowne wczytywanie poświadczeń SNMPv3

Kontrole pozycji walk[] i get[] agenta SNMP teraz automatycznie ponownie wczytują zaktualizowane ustawienia uwierzytelniania i prywatności SNMPv3, gdy odpowiedni interfejs SNMPv3 zostanie zmodyfikowany. Ręczne polecenie ponownego wczytania pamięci podręcznej -R snmp_cache_reload pozostaje dostępne do rozwiązywania problemów oraz dla urządzeń, które nie w pełni spełniają specyfikacje SNMPv3.

smart.disk.discovery — nowy parametr **type**

Pozycja smart.disk.discovery (wtyczka S.M.A.R.T. dla Zabbix agent 2) obsługuje teraz opcjonalny parametr **type**, który służy do określenia wartości, według której mają być skanowane dyski.

Co nowego w Zabbix 7.4.1 Obsługa TimescaleDB 2.21

Maksymalna obsługiwana wersja TimescaleDB to teraz 2.21.X.

Informacje diagnostyczne przetwarzania wstępnego serwera/proxy

Do sekcji przetwarzania wstępnego informacji diagnostycznych czasu działania serwera/proxy dodano sekcje najważniejszych pozycji posortowane według liczby otrzymanych wartości i rozmiaru. Identyfikatory najważniejszych pozycji z najdłuższym czasem przetwarzania są wyróżniane.

Te informacje są dostępne po uruchomieniu serwera Zabbix lub proxy z opcją czasu działania diaginfo, na przykład `zabbix_server -R diaginfo=preprocessing`.

Pomijanie menedżera przetwarzania wstępnego dla pozycji bez przetwarzania wstępnego

Aby zmniejszyć obciążenie przetwarzania, wartości pozycji z pozycji, które nie mają ani przetwarzania wstępnego, ani pozycji zależnych, są dodawane bezpośrednio do pamięci podręcznej historii lub wysyłane do menedżera LLD (zobacz Szczegóły przetwarzania wstępnego). Wcześniej wszystkie wartości pozycji przechodziły przez menedżera przetwarzania wstępnego niezależnie od konfiguracji przetwarzania wstępnego.

Zaktualizowano integrację webhook GLPI

Typ mediów GLPI obsługuje teraz uwierzytelnianie za pomocą tokenów aplikacji.

2 Definicje

Przegląd W tej sekcji możesz poznać znaczenie niektórych terminów powszechnie używanych w Zabbix.

Definicje ***host***

- dowolne fizyczne lub wirtualne urządzenie, aplikacja, usługa albo inny logicznie powiązany zbiór monitorowanych parametrów.

host group

- logiczna grupa hostów. Grupy hostów są używane podczas przypisywania praw dostępu do hostów dla różnych grup użytkowników.

item

- określony fragment danych, który chcesz otrzymywać z hosta, czyli metryka danych.

value preprocessing

- transformacja otrzymanej wartości metryki przed zapisaniem jej do bazy danych.

trigger

- logiczne wyrażenie, które definiuje próg problemu i służy do „oceny” danych otrzymywanych w pozycjach.

Gdy otrzymane dane są powyżej progu, wyzwalacze przechodzą ze stanu „Ok” do stanu „Problem”. Gdy otrzymane dane są poniżej progu, wyzwalacze pozostają w stanie „Ok” lub do niego wracają.

template

- zestaw encji (pozycje, wyzwalacze, wykresy, reguły niskopoziomowego odkrywania, scenariusze web) gotowy do zastosowania na jednym lub kilku hostach.

Zadaniem szablonów jest przyspieszenie wdrażania zadań monitorowania na hoście, a także ułatwienie wprowadzania masowych zmian w zadaniach monitorowania. Szablony są łączone bezpośrednio z poszczególnymi hostami.

template group

- logiczna grupa szablonów. Grupy szablonów są używane podczas przypisywania praw dostępu do szablonów dla różnych grup użytkowników.

event

- pojedyncze wystąpienie czegoś, co zasługuje na uwagę, na przykład zmiana stanu wyzwalacza lub przeprowadzenie odkrywania/autorejestracji agenta.

event tag

- z góry zdefiniowany znacznik dla zdarzenia. Może być używany w korelacji zdarzeń, granulacji uprawnień itp.

event correlation

- metoda elastycznego i precyzyjnego kojarzenia problemów z ich rozwiązaniami.

Na przykład możesz zdefiniować, że problem zgłoszony przez jeden wyzwalacz może zostać rozwiązany przez inny wyzwalacz, który może nawet korzystać z innej metody zbierania danych.

problem

- wyzwalacz, który znajduje się w stanie „Problem”.

problem update

- opcje zarządzania problemami udostępniane przez Zabbix, takie jak dodanie komentarza, potwierdzenie, zmiana ważności lub ręczne zamknięcie.

action

- z góry zdefiniowany sposób reagowania na zdarzenie.

Akcja składa się z operacji (np. wysłania powiadomienia) oraz warunków (*kiedy* operacja jest wykonywana)

escalation

- niestandardowy scenariusz wykonywania operacji w ramach akcji; sekwencja wysyłania powiadomień/wykonywania zdalnych poleceń.

macro

- symbol zastępczy zmiennej, który przyjmuje określoną wartość w zależności od kontekstu.

media

- środek dostarczania powiadomień; kanał dostarczania.

notification

- wiadomość o zdarzeniu wysyłana do użytkownika za pośrednictwem wybranego kanału mediów.

remote command

- z góry zdefiniowane polecenie, które jest automatycznie wykonywane na monitorowanym hoście po spełnieniu określonego warunku.

web scenario

- jedno lub kilka żądań HTTP służących do sprawdzania dostępności witryny internetowej.

frontend

- interfejs webowy dostarczany wraz z Zabbix.

dashboard

- konfigurowalna część interfejsu webowego wyświetlająca podsumowania i wizualizacje ważnych informacji w jednostkach wizualnych zwanych widżetami.

widget

- jednostka wizualna wyświetlająca informacje określonego rodzaju i źródła (podsumowanie, mapa, wykres, zegar itp.), używana na pulpicie.

Zabbix API

- Zabbix API umożliwia używanie protokołu JSON RPC do tworzenia, aktualizowania i pobierania obiektów Zabbix (takich jak hosty, pozycje, wykresy i inne) lub wykonywania innych niestandardowych zadań.

Zabbix server

- centralny proces oprogramowania Zabbix, który wykonuje monitorowanie, współpracuje z proxy Zabbix i agentami, oblicza wyzwalacze, wysyła powiadomienia; centralne repozytorium danych.

Zabbix proxy

- proces, który może zbierać dane w imieniu serwera Zabbix, odciążając serwer z części zadań przetwarzania.

Zabbix agent

- proces wdrażany na monitorowanych celach w celu aktywnego monitorowania lokalnych zasobów i aplikacji.

Zabbix agent 2

- nowa generacja Zabbix agent służąca do aktywnego monitorowania lokalnych zasobów i aplikacji, umożliwiająca używanie niestandardowych wtyczek do monitorowania.

> **Attention:**
> Ponieważ Zabbix agent 2 udostępnia wiele funkcji wspólnych z Zabbix agent, termin "Zabbix agent" w dokumentacji oznacza oba - Zabbix agent i Zabbix agent 2, jeśli zachowanie funkcjonalne jest takie samo. Zabbix agent 2 jest wymieniany z nazwy tylko tam, gdzie jego funkcjonalność różni się.

encryption

- obsługa szyfrowanej komunikacji między komponentami Zabbix (server, proxy, agent, narzędzia zabbix_sender i zabbix_get) z użyciem protokołu Transport Layer Security (TLS).

agent autoregistration

- zautomatyzowany proces, w którym sam Zabbix agent jest rejestrowany jako host i rozpoczyna monitorowanie.

network discovery

- zautomatyzowane wykrywanie urządzeń sieciowych.

low-level discovery

- zautomatyzowane wykrywanie obiektów niskiego poziomu na określonym urządzeniu (np. systemów plików, interfejsów sieciowych itp.).

low-level discovery rule

- zestaw definicji służących do zautomatyzowanego wykrywania obiektów niskiego poziomu na urządzeniu.

item prototype

- metryka z określonymi parametrami jako zmiennymi, gotowa do niskopoziomowego odkrywania. Po niskopoziomowym odkrywaniu zmienne są automatycznie zastępowane rzeczywiście wykrytymi parametrami, a metryka automatycznie zaczyna zbierać dane.

trigger prototype

- wyzwalacz z określonymi parametrami jako zmiennymi, gotowy do niskopoziomowego odkrywania. Po niskopoziomowym odkrywaniu zmienne są automatycznie zastępowane rzeczywiście wykrytymi parametrami, a wyzwalacz automatycznie zaczyna oceniać dane.

Prototypy niektórych innych encji Zabbix są również używane w niskopoziomowym odkrywaniu - prototypy wykresów, prototypy hostów, prototypy grup hostów.

3 Procesy Zabbix

Przegląd Zabbix składa się z kilku procesów, które współpracują ze sobą, aby monitorować systemy i wysyłać powiadomienia. Każdy proces pełni określoną rolę w infrastrukturze monitoringu.

Główne procesy

- Serwer — centralny proces, który wykonuje odpytywanie i przechwytywanie danych, oblicza wyzwalacze oraz wysyła powiadomienia do użytkowników.
- Agent — zbiera dane z monitorowanych hostów.
- Agent 2 — agent nowej generacji z architekturą wtyczek i rozszerzonymi możliwościami.
- Proxy — zbiera dane monitoringu z monitorowanych urządzeń i wysyła informacje do serwera Zabbix.
- Java gateway — umożliwia monitorowanie aplikacji Java.

1 Server

Omówienie

Serwer Zabbix jest centralnym procesem oprogramowania Zabbix.

Serwer wykonuje odpytywanie i przechwytywanie danych, oblicza wyzwalacze, wysyła powiadomienia do użytkowników. Jest centralnym komponentem, do którego agent i proxy Zabbix zgłaszają dane o dostępności i integralności systemów. Serwer może również zdalnie sprawdzać usługi sieciowe (takie jak serwery WWW i serwery poczty) przy użyciu prostych testów usług.

Serwer jest centralnym repozytorium, w którym przechowywane są wszystkie dane konfiguracyjne, statystyczne i operacyjne, a także elementem Zabbix, który aktywnie powiadamia administratorów, gdy w którymkolwiek z monitorowanych systemów wystąpią problemy.

Działanie podstawowego serwera Zabbix jest podzielone na trzy odrębne komponenty; są to: serwer Zabbix, frontend WWW i magazyn bazy danych.

Wszystkie informacje konfiguracyjne dla Zabbix są przechowywane w bazie danych, z którą współpracują zarówno serwer, jak i frontend WWW. Na przykład, gdy tworzysz nową pozycja za pomocą frontend WWW (lub API), jest ona dodawana do tabeli items w bazie danych. Następnie, mniej więcej raz na minutę, serwer Zabbix będzie odpytywał tabelę items o listę aktywnych pozycji, która następnie jest przechowywana w pamięci podręcznej wewnątrz serwera Zabbix. Dlatego wprowadzone w frontend Zabbix zmiany mogą pojawić się w sekcji najnowszych danych dopiero po upływie nawet dwóch minut.

Uruchamianie serwera

Jeśli zainstalowano z pakietu

Serwer Zabbix działa jako proces demona. Serwer można uruchomić, wykonując:

```
systemctl start zabbix-server
```

To zadziała w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

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

Podobnie, aby zatrzymać/uruchomić ponownie/sprawdzić stan, użyj następujących poleceń:

```
systemctl stop zabbix-server
systemctl restart zabbix-server
systemctl status zabbix-server
```

Uruchamianie ręczne

Jeśli powyższe nie działa, musisz uruchomić go ręcznie. Znajdź ścieżkę do pliku binarnego `zabbix_server` i wykonaj:

```
zabbix_server
```

Możesz używać następujących parametrów wiersza poleceń z serwerem Zabbix:

```
-c --config <file>              Ścieżka do pliku konfiguracyjnego (domyślnie /usr/local/etc/zabbix_server.
-f --foreground                 Uruchom serwer Zabbix na pierwszym planie
-R --runtime-control <option>   Wykonaj funkcje administracyjne
-T --test-config                Sprawdź poprawność pliku konfiguracyjnego i zakończ
-h --help                       Wyświetl tę pomoc
-V --version                    Wyświetl numer wersji
```

Przykłady uruchamiania serwera Zabbix z parametrami wiersza poleceń:

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

Sterowanie w czasie działania

Opcje sterowania w czasie działania:

Option	Description	Target
`config_cache_reload`	Ponownie wczytaj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecnie wczytywana.	
`history_cache_clear[,<id>]`	Wyczyść pamięć podręczną historii dla pozycji określonej przez jej ID. Dotyczy wszystkich wartości pozycji, z wyjątkiem pierwszej i ostatniej wartości.	**target** - ID pozycji

Option	Description	Target
`diaginfo[=<section>]`	Zbierz informacje diagnostyczne w pliku dziennika serwera.	`historycache` - statystyki pamięci podręcznej historii; `valuecache` - statystyki pamięci podręcznej wartości; `preprocessing` - statystyki menedżera przetwarzania wstępnego; `alerting` - statystyki menedżera alertów; `lld` - statystyki menedżera LLD; `locks` - lista mutexów (jest pusta w systemach *BSD*); `connector` - statystyki dla konektorów z największą kolejką.
`ha_status`	Zapisz status klastra wysokiej dostępności (HA) do dziennika.	
`ha_remove_node=target`	Usuń węzeł wysokiej dostępności (HA) określony przez nazwę lub ID. Pamiętaj, że nie można usuwać węzłów aktywnych/standby.	**target** - nazwa lub ID węzła (można je uzyskać, uruchamiając `ha_status`).
`ha_set_failover_delay=delay`	Ustaw opóźnienie przełączenia awaryjnego wysokiej dostępności (HA). Obsługiwane są sufiksy czasu, np. 10s, 1m.	
`proxy_config_cache_reload[=target]`	Ponownie wczytaj pamięć podręczną konfiguracji proxy.	**target** - lista nazw proxy rozdzielona przecinkami. Jeśli nie określono target, ponownie wczytana zostanie konfiguracja wszystkich proxy.
`secrets_reload`	Ponownie wczytaj sekrety z Vault.	
`service_cache_reload`	Ponownie wczytaj pamięć podręczną menedżera usług.	
`snmp_cache_reload`	Ponownie wczytaj pamięć podręczną SNMP — wyczyść właściwości silnika SNMP (engine time, engine boots, engine id, credentials) dla wszystkich hostów. Użyj, aby wymusić globalne wyczyszczenie pamięci podręcznej podczas rozwiązywania problemów z SNMP.	
`housekeeper_execute`	Uruchom procedurę housekeeping. Ignorowane, jeśli procedura housekeeping jest obecnie w toku.	
`trigger_housekeeper_execute`	Uruchom procedurę housekeeping dla wyzwalaczy. Ignorowane, jeśli procedura housekeeping dla wyzwalaczy jest obecnie w toku. Do czasu uruchomienia procedury housekeeping dla wyzwalaczy problemy spowodowane przez wyzwalacze, które zostały już usunięte, mogą nadal generować problemy usług i przypisywać je do usług. Jeśli Twoja konfiguracja obejmuje wiele reguł obliczania statusu usług opartych na często wykrywanych/niewykrywanych wyzwalaczach, rozważ zwiększenie częstotliwości procedury housekeeping przez dostosowanie parametru konfiguracji serwera `ProblemHousekeepingFrequency`.	
`log_level_increase[=target]`	Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	**process type** - wszystkie procesy określonego typu (np. `poller`). Zobacz wszystkie typy procesów serwera. **process type,N** - typ procesu i numer (np. `poller,3`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.
`log_level_decrease[=target]`	Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	

Option	Description	Target
`prof_enable[=<target>]`	Włącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony. Włączone profilowanie dostarcza szczegółów wszystkich rwlock/mutex według nazwy funkcji.	**process type** - wszystkie procesy określonego typu (np. `history syncer`) Obsługiwane typy procesów jako targety profilowania: `alerter`, `alert manager`, `availability manager`, `configuration syncer`, `discovery manager`, `escalator`, `history poller`, `history syncer`, `housekeeper`, `http poller`, `icmp pinger`, `ipmi manager`, `ipmi poller`, `java poller`, `lld manager`, `lld worker`, `odbc poller`, `poller`, `preprocessing manager`, `preprocessing worker`, `proxy poller`, `self-monitoring`, `service manager`, `snmp trapper`, `task manager`, `timer`, `trapper`, `unreachable poller`, `vmware collector`. **process type,N** - typ procesu i numer (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'. **scope** - `rwlock`, `mutex`, `processing` można użyć z typem procesu i numerem (np. `history syncer,1,processing`) lub ze wszystkimi procesami danego typu (np. `history syncer,rwlock`).
`prof_disable[=<target>]`	Wyłącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Obsługiwane typy procesów jako targety profilowania: zobacz `prof_enable`. **process type,N** - typ procesu i numer (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.

Przykład użycia kontroli w czasie działania do przeładowania pamięci podręcznej konfiguracji serwera:

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

Przykłady użycia kontroli w czasie działania do przeładowania konfiguracji proxy:

```
### Przeładuj konfigurację wszystkich proxy:
zabbix_server -R proxy_config_cache_reload

### Przeładuj konfigurację Proxy1 i Proxy2:
zabbix_server -R proxy_config_cache_reload=Proxy1,Proxy2
```

Przykład użycia kontroli w czasie działania do wyczyszczenia pamięci podręcznej historii dla pozycji:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R history_cache_clear=42243
```

Przykłady użycia kontroli w czasie działania do zebrania informacji diagnostycznych:

```
### Zbierz wszystkie dostępne informacje diagnostyczne w pliku dziennika serwera:
zabbix_server -R diaginfo

### Zbierz statystyki pamięci podręcznej historii w pliku dziennika serwera:
zabbix_server -R diaginfo=historycache
```

Przykład użycia kontroli w czasie działania do przeładowania pamięci podręcznej SNMP:

```
zabbix_server -R snmp_cache_reload
```

Attention:
Gdy interfejs SNMPv3 jest aktualizowany za pomocą interfejsu użytkownika Zabbix, Zabbix w większości przypadków automatycznie przeładuje nowe poświadczenia SNMPv3 dla tego interfejsu; używaj `-R snmp_cache_reload` tylko wtedy, gdy odpytywanie nadal kończy się niepowodzeniem po zmianie poświadczeń (na przykład z powodu niespójności engineBoots/engineID lub urządzeń niezgodnych z RFC), albo gdy trzeba wymusić globalne wyczyszczenie pamięci podręcznej SNMP na potrzeby rozwiązywania problemów.

Przykład użycia kontroli w czasie działania do wyzwolenia wykonania housekeepera:

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

Przykłady użycia kontroli w czasie działania do zmiany poziomu logowania:

```
### Zwiększ poziom logowania wszystkich procesów:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase

### Zwiększ poziom logowania drugiego procesu pollera:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2

### Zwiększ poziom logowania procesu o PID 1234:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234

### Zmniejsz poziom logowania wszystkich procesów http poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Przykład ustawienia opóźnienia przełączenia awaryjnego HA na minimalne 10 sekund:

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

Użytkownik procesu

Serwer Zabbix jest przeznaczony do uruchamiania jako użytkownik inny niż root. Będzie działać jako dowolny użytkownik niebędący rootem, jako który został uruchomiony. Możesz więc uruchomić serwer jako dowolny użytkownik niebędący rootem bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako 'root', przełączy się na zakodowanego na stałe użytkownika 'zabbix', który musi być obecny w Twoim systemie. Możesz uruchomić serwer jako 'root' tylko wtedy, gdy odpowiednio zmodyfikujesz parametr 'AllowRoot' w pliku konfiguracyjnym serwera.

Jeśli serwer Zabbix i agent są uruchomione na tym samym komputerze, zaleca się używanie innego użytkownika do uruchamiania serwera niż do uruchamiania agenta. W przeciwnym razie, jeśli oba są uruchomione jako ten sam użytkownik, agent może uzyskać dostęp do pliku konfiguracyjnego serwera, a każdy użytkownik na poziomie Admin w Zabbix może dość łatwo pobrać na przykład hasło do bazy danych.

Plik konfiguracyjny

Szczegóły dotyczące konfiguracji `zabbix_server` można znaleźć w opcjach pliku konfiguracyjnego.

Skrypty startowe

Skrypty są używane do automatycznego uruchamiania/zatrzymywania procesów Zabbix podczas uruchamiania/zamykania systemu. Skrypty znajdują się w katalogu misc/init.d.

Typy procesów i wątki serwera

- `agent poller` - asynchroniczny proces pollera dla kontroli pasywnych z wątkiem roboczym;
- `alert manager` - menedżer kolejki alertów;
- `alert syncer` - zapisujący alerty do bazy danych;
- `alerter` - proces wysyłania powiadomień;
- `availability manager` - proces aktualizacji dostępności hostów;
- `browser poller` - poller dla kontroli pozycji przeglądarki;
- `configuration syncer` - proces zarządzania pamięciową pamięcią podręczną danych konfiguracyjnych;
- `configuration syncer worker` - proces rozwiązywania i synchronizowania wartości makr użytkownika w nazwach pozycji;
- `connector manager` - proces menedżera dla konektorów;
- `connector worker` - proces obsługujący żądania od menedżera konektorów;
- `discovery manager` - proces menedżera wykrywania urządzeń;
- `discovery worker` - proces obsługujący zadania wykrywania od menedżera wykrywania;
- `escalator` - proces eskalacji działań;

- `ha manager` - proces zarządzania wysoką dostępnością;
- `history poller` - proces obsługujący kontrole obliczane wymagające połączenia z bazą danych;
- `history syncer` - zapisujący historię do bazy danych;
- `housekeeper` - proces usuwania nieaktualnych danych (historii i trendów pozycji, sesji użytkowników, zdarzeń itp.), a także danych pozostawionych przez usunięte obiekty;
- `http agent poller` - asynchroniczny proces pollera dla kontroli HTTP z wątkiem roboczym;
- `http poller` - poller monitorowania WWW;
- `icmp pinger` - poller dla kontroli icmpping;
- `internal poller` - poller dla kontroli wewnętrznych;
- `ipmi manager` - menedżer pollera IPMI;
- `ipmi poller` - poller dla kontroli IPMI;
- `java poller` - poller dla kontroli Java;
- `lld manager` - proces menedżera zadań niskopoziomowego wykrywania;
- `lld worker` - proces roboczy zadań niskopoziomowego wykrywania;
- `odbc poller` - poller dla kontroli ODBC;
- `poller` - standardowy poller dla kontroli pasywnych;
- `preprocessing manager` - menedżer zadań przetwarzania wstępnego z wątkami roboczymi przetwarzania wstępnego;
- `preprocessing worker` - wątek przetwarzania danych wstępnego;
- `proxy poller` - poller dla pasywnych proxy;
- `proxy group manager` - menedżer równoważenia obciążenia i wysokiej dostępności proxy;
- `report manager`- menedżer zadań generowania raportów zaplanowanych;
- `report writer` - proces generowania raportów zaplanowanych;
- `self-monitoring` - proces zbierania wewnętrznych statystyk serwera;
- `service manager` - proces zarządzania usługami poprzez odbieranie informacji o problemach, tagach problemów i odzyskiwaniu po problemach z history syncer, task manager i alert manager;
- `snmp poller` - asynchroniczny proces pollera dla kontroli SNMP z wątkiem roboczym (tylko pozycje `walk[OID]` i `get[OID]`);
- `snmp trapper` - trapper dla pułapek SNMP;
- `task manager` - proces zdalnego wykonywania zadań żądanych przez inne komponenty (np. zamknięcie problemu, potwierdzenie problemu, sprawdzenie wartości pozycji teraz, funkcja zdalnego polecenia);
- `timer` - timer przetwarzający konserwacje;
- `trapper` - trapper dla kontroli aktywnych, pułapek i komunikacji z proxy;
- `trigger housekeeper` - proces usuwania problemów i zdarzeń wygenerowanych przez wyzwalacze, które zostały później usunięte;
- `unreachable poller` - poller dla niedostępnych urządzeń;
- `vmware collector` - kolektor danych VMware odpowiedzialny za zbieranie danych z usług VMware.

Plik dziennika serwera można wykorzystać do obserwowania tych typów procesów.

Od Zabbix 7.4.6 plik dziennika serwera jest tworzony z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są zabronione.

Różne typy procesów serwera Zabbix można monitorować za pomocą wewnętrznej pozycji `zabbix[process,<type>,<mode>,<state>]`.

Statystyki transakcji synchronizatora historii

Tytuł procesu synchronizatora historii wyświetla szczegółowe statystyki dotyczące transakcji synchronizatora historii:

```
205182 ?        S      0:00  zabbix_server: history syncer #2 [processed 0 values, 0+0 triggers in 0.00002
205183 ?        S      0:00  zabbix_server: history syncer #3 [processed 18 values, 7+0 triggers in 0.0026
205184 ?        S      0:00  zabbix_server: history syncer #4 [processed 0 values, 0+0 triggers in 0.00002
```

W sekcji "A+B triggers":

- A - wyzwalacze przetworzone na podstawie wartości historii;
- B - wyzwalacze przetworzone na podstawie timerów.

Czasy w `processed...in N (<timings>) sec` to:

- Czas spędzony na zapisywaniu wartości pozycji do bazy danych;
- Czas spędzony na aktualizowaniu danych pozycji (stan, błędy, inwentarz hosta itp.);
- Czas spędzony na zapisywaniu trendów do bazy danych;
- Czas spędzony na obliczaniu wyzwalaczy;
- Czas spędzony na przetwarzaniu zdarzeń i działań.

Obsługiwane platformy

Ze względu na wymagania bezpieczeństwa oraz krytyczny charakter działania serwera, UNIX jest jedynym systemem operacyjnym, który może konsekwentnie zapewniać wymaganą wydajność, odporność na awarie i niezawodność. Zabbix działa na wiodących

na rynku wersjach systemu.

Serwer Zabbix jest testowany na następujących platformach:

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

Note:
Zabbix może działać również na innych systemach operacyjnych typu Unix.

Ustawienia regionalne

Należy pamiętać, że serwer wymaga ustawień regionalnych UTF-8, aby niektóre tekstowe pozycje mogły być poprawnie interpretowane. Większość nowoczesnych systemów typu Unix ma domyślnie ustawienia regionalne UTF-8, jednak istnieją systemy, w których może być konieczne ich jawne skonfigurowanie.

1 Wysoka dostępność

Omówienie

Wysoka dostępność (HA) jest zazwyczaj wymagana w infrastrukturach krytycznych, które mogą sobie pozwolić na praktycznie zerowy czas przestoju. Dlatego dla każdej usługi, która może ulec awarii, musi istnieć opcja przełączenia awaryjnego, aby przejąć działanie w przypadku awarii bieżącej usługi.

Zabbix oferuje **natywne** rozwiązanie wysokiej dostępności, które jest łatwe w konfiguracji i nie wymaga wcześniejszego doświadczenia z HA. Natywna HA Zabbixa może być przydatna jako dodatkowa warstwa ochrony przed awariami oprogramowania/sprzętu serwera Zabbix lub w celu skrócenia przestojów związanych z konserwacją.

W trybie wysokiej dostępności Zabbixa wiele serwerów Zabbix działa jako węzły w klastrze. Gdy jeden serwer Zabbix w klastrze jest aktywny, pozostałe pozostają w trybie gotowości, przygotowane do przejęcia działania w razie potrzeby.



Przejście na HA Zabbixa nie jest wiążące. W dowolnym momencie można wrócić do pracy w trybie samodzielnym.

Zobacz także: Szczegóły implementacji

Włączanie wysokiej dostępności

Uruchamianie serwera Zabbix jako węzła klastra

Do uruchomienia serwera Zabbix jako węzła klastra wymagane są dwa parametry w konfiguracji serwera:

- Parametr `HANodeName` musi być określony dla każdego serwera Zabbix, który będzie węzłem klastra HA.

Jest to unikalny identyfikator węzła (np. `zabbix-node-01`), pod którym serwer będzie identyfikowany w konfiguracjach agenta i proxy. Jeśli nie określisz `HANodeName`, serwer zostanie uruchomiony w trybie autonomicznym.

- Parametr `NodeAddress` musi być określony dla każdego węzła.

Parametr `NodeAddress` (`address:port`) będzie używany przez frontend Zabbix do łączenia się z aktywnym węzłem serwera. `NodeAddress` musi odpowiadać adresowi IP lub nazwie FQDN odpowiedniego serwera Zabbix.

Po wprowadzeniu zmian w plikach konfiguracyjnych uruchom ponownie wszystkie serwery Zabbix. Zostaną one teraz uruchomione jako węzły klastra. Nowy status serwerów można zobaczyć w *Raporty* > *Informacje o systemie*, a także po uruchomieniu:

```
zabbix_server -R ha_status
```

To polecenie wykonywane w czasie działania zapisze bieżący status klastra HA do logu serwera Zabbix (oraz na stdout):

```
Failover delay: 60 seconds
Cluster status:
   #  ID                        Name                 Address                Status      Last Access
   1. ckzxxqg7u0001lsropenyzh3m zabbix-node-01       64.227.66.193:10051    standby     0s
   2. ckzxyqo1k00013frpq539e1jp zabbix-node-02       64.227.74.25:10051     active      3s
```

Przygotowanie frontend

Upewnij się, że `address:port` serwera Zabbix nie jest zdefiniowany w konfiguracji frontend (znajdującej się w pliku `conf/zabbix.conf.php` w katalogu plików frontend).

```
// Uncomment and set to desired values to override Zabbix hostname/IP and port.
// $ZBX_SERVER                  = '';
// $ZBX_SERVER_PORT             = '';
```

Frontend Zabbix automatycznie wykryje aktywny węzeł, odczytując ustawienia z tabeli nodes w bazie danych Zabbix. Adres węzła aktywnego będzie używany jako adres serwera Zabbix.

Konfiguracja proxy

Węzły klastra HA (serwery) muszą być wymienione w konfiguracji pasywnego lub aktywnego proxy Zabbix.

W przypadku pasywnego proxy nazwy węzłów muszą być wymienione w parametrze Server parameter proxy, rozdzielone **przecinkiem**.

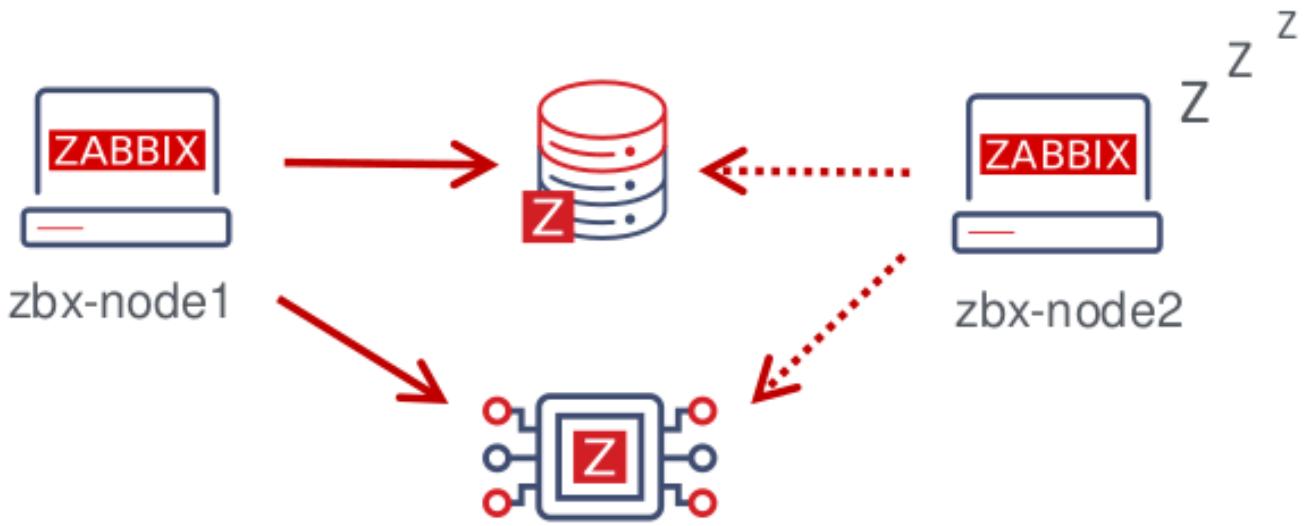
```
Server=zabbix-node-01,zabbix-node-02
```

W przypadku aktywnego proxy nazwy węzłów muszą być wymienione w parametrze Server parameter proxy, rozdzielone **średnikiem**.

```
Server=zabbix-node-01;zabbix-node-02
```

Konfiguracja agent

Węzły klastra HA (serwery) muszą być wymienione w konfiguracji Zabbix agent lub Zabbix agent 2.



Aby włączyć pasywne sprawdzenia, nazwy węzłów muszą być wymienione w parametrze Server parameter, oddzielone **przecinkiem**.

```
Server=zabbix-node-01,zabbix-node-02
```

Aby włączyć aktywne sprawdzenia, nazwy węzłów muszą być wymienione w parametrze ServerActive parameter. Należy pamiętać, że w przypadku aktywnych sprawdzeń węzły muszą być oddzielone przecinkiem od innych serwerów, natomiast same węzły muszą być oddzielone **średnikiem**, np.:

```
ServerActive=zabbix-node-01;zabbix-node-02
```

Przełączenie awaryjne na węzeł rezerwowy

Zabbix automatycznie przełączy się na inny węzeł, jeśli aktywny węzeł przestanie działać. Aby przełączenie awaryjne mogło nastąpić, musi istnieć co najmniej jeden węzeł o statusie rezerwowym.

Jak szybko nastąpi przełączenie awaryjne? Wszystkie węzły aktualizują czas ostatniego dostępu (oraz status, jeśli uległ zmianie) co 5 sekund. Zatem:

- Jeśli aktywny węzeł zostanie zamknięty i zdoła zgłosić swój status jako "stopped", inny węzeł przejmie działanie w ciągu **5 sekund**.
- Jeśli aktywny węzeł zostanie zamknięty / stanie się niedostępny bez możliwości aktualizacji swojego statusu, węzły rezerwowe będą czekać **opóźnienie przełączenia awaryjnego** + 5 sekund przed przejęciem działania.

Opóźnienie przełączenia awaryjnego jest konfigurowalne, a obsługiwany zakres wynosi od 10 sekund do 15 minut (domyślnie jedna minuta). Aby zmienić opóźnienie przełączenia awaryjnego, możesz uruchomić:

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

Zarządzanie klastrem HA

Bieżący stan klastra HA można zarządzać za pomocą dedykowanych opcji runtime control:

- `ha_status` - zapisuje status klastra HA w logu serwera Zabbix (oraz na stdout);
- `ha_remove_node=target` - usuwa węzeł HA zidentyfikowany przez jego `<target>` - nazwę lub ID węzła (nazwę/ID można uzyskać z wyniku działania `ha_status`), np.:

```
zabbix_server -R ha_remove_node=zabbix-node-02
```

Należy pamiętać, że aktywne/w trybie standby węzły nie mogą zostać usunięte.

- `ha_set_failover_delay=delay` - ustawia opóźnienie przełączenia awaryjnego HA (od 10 sekund do 15 minut; obsługiwane są sufiksy czasu, np. 10s, 1m).

Stan węzła można monitorować:

- W *Reports* > *System information*.
- W widżecie pulpitu *System information*.
- Za pomocą opcji runtime control `ha_status` serwera (patrz wyżej).

Do wykrywania węzłów można użyć wewnętrznej pozycji `zabbix[cluster,discovery,nodes]`, ponieważ zwraca ona JSON z informacjami o węzłach wysokiej dostępności.

Wyłączanie klastra HA

Aby wyłączyć klaster wysokiej dostępności:

1. Utwórz kopie zapasowe plików konfiguracyjnych.
2. Zatrzymaj węzły rezerwowe.
3. Usuń parametr HANodeName z aktywnego serwera głównego.
4. Uruchom ponownie serwer główny (zostanie uruchomiony w trybie autonomicznym).

Aktualizacja klastra HA

Aby wykonać aktualizację do głównej wersji dla węzłów HA:

1. Zatrzymaj wszystkie węzły.
2. Utwórz pełną kopię zapasową bazy danych.
3. Jeśli baza danych używa replikacji, upewnij się, że wszystkie węzły są zsynchronizowane i nie występują żadne problemy. Nie wykonuj aktualizacji, jeśli replikacja jest uszkodzona.
4. Wybierz jeden węzeł, który wykona aktualizację bazy danych, zmień jego konfigurację na tryb standalone, komentując `HANodeName`, i zaktualizuj go.
5. Upewnij się, że aktualizacja bazy danych została w pełni zakończona (*System information* powinno wyświetlać, że serwer Zabbix działa).
6. Uruchom ponownie węzeł w trybie HA.
7. Zaktualizuj i uruchom pozostałe węzły (nie ma potrzeby przełączania ich do trybu standalone, ponieważ baza danych jest już w tym momencie zaktualizowana).

W przypadku aktualizacji do wersji minor wystarczy zaktualizować pierwszy węzeł, upewnić się, że został zaktualizowany i działa, a następnie rozpocząć aktualizację następnego węzła.

Szczegóły implementacji

Klaster wysokiej dostępności (HA) jest rozwiązaniem opcjonalnym i jest obsługiwany dla serwera Zabbix. Natywne rozwiązanie HA zostało zaprojektowane tak, aby było proste w użyciu, działało między lokalizacjami i nie miało szczególnych wymagań dotyczących baz danych rozpoznawanych przez Zabbix. Użytkownicy mogą korzystać z natywnego rozwiązania HA Zabbix albo z rozwiązania HA innej firmy, w zależności od tego, co najlepiej odpowiada wymaganiom wysokiej dostępności w ich środowisku.

Rozwiązanie składa się z wielu instancji lub węzłów `zabbix_server`. Każdy węzeł:

- Jest konfigurowany oddzielnie.
- Korzysta z tej samej bazy danych.
- Może mieć kilka trybów: active, standby, unavailable, stopped.

Tylko jeden węzeł może być aktywny (working) w danym momencie. Węzeł standby uruchamia tylko jeden proces - menedżer HA. Węzeł standby nie wykonuje zbierania danych, przetwarzania ani innych standardowych działań serwera; nie nasłuchuje na portach; ma minimalną liczbę połączeń z bazą danych.

Zarówno węzły active, jak i standby aktualizują swój czas ostatniego dostępu co 5 sekund. Każdy węzeł standby monitoruje czas ostatniego dostępu węzła active. Jeśli czas ostatniego dostępu węzła active przekroczy liczbę sekund 'failover delay', węzeł standby sam przełącza się na węzeł active i przypisuje status `unavailable` wcześniej aktywnemu węzłowi.

Węzeł active monitoruje własną łączność z bazą danych - jeśli zostanie utracona na dłużej niż `failover delay`−5 sekund, musi zatrzymać całe przetwarzanie i przełączyć się w tryb standby. Węzeł active monitoruje również status węzłów standby - jeśli czas ostatniego dostępu węzła standby przekroczy `failover delay` sekund, węzłowi standby zostaje przypisany status `unavailable`.

Węzły zostały zaprojektowane tak, aby były zgodne między wersjami pomocniczymi Zabbix.

2 Agent

Przegląd

agent Zabbix jest wdrażany na monitorowanym obiekcie, aby aktywnie monitorować lokalne zasoby i aplikacje (dyski twarde, pamięć, statystyki procesora itp.).

agent zbiera lokalnie informacje operacyjne i przesyła dane do serwer Zabbix w celu dalszego przetwarzania. W przypadku awarii (takich jak zapełnienie dysku twardego lub awaria procesu usługi), serwer Zabbix może aktywnie powiadomić administratorów konkretnej maszyny, która zgłosiła awarię.

agenty Zabbix są bardzo wydajne dzięki użyciu natywnych wywołań systemowych do zbierania informacji statystycznych.

Możliwe jest uruchomienie wielu instancji agenta na host.

Kontrole pasywne i aktywne

Agenty Zabbix mogą wykonywać kontrole pasywne i aktywne:

- Kontrole pasywne - agent Zabbix odpowiada na żądanie z serwer Zabbix (lub proxy). Na przykład serwer żąda danych (np. obciążenia CPU), a agent zwraca wynik.
- Kontrole aktywne - agent Zabbix zbiera i wysyła dane bez oczekiwania na żądanie z serwer Zabbix (lub proxy). Najpierw pobiera listę pozycji monitorowania z serwer (obciążenie CPU, dostępna pamięć itp.), a następnie zbiera wymagane dane i okresowo wysyła do niego nowe wartości.

Typ kontroli agenta jest konfigurowany przez wybranie odpowiedniego typu pozycji monitorowania. Agent Zabbix przetwarza pozycje typu *Zabbix agent* lub *Zabbix agent (active)*.

Obsługiwane platformy

Informacje o obsługiwanych platformach można znaleźć na stronie Wymagania.

Agent w systemach uniksopodobnych

Agent Zabbixa w systemach uniksopodobnych jest uruchamiany na hostcie monitorowanym.

Instalacja

agent Zabbix można zainstalować w systemach opartych na Linuxie, korzystając z jednej z następujących metod:

- Pakiety Zabbix - wybierz komponent Agent (po wybraniu wersji Zabbix, dystrybucji OS i wersji OS), a następnie postępuj zgodnie z podanymi instrukcjami;

- Źródła Zabbix - pobierz pliki źródłowe i skompiluj agent Zabbix, konfigurując źródła z opcją `--enable-agent`.

Attention:
Ogólnie rzecz biorąc, 32-bitowe agenty Zabbix będą działać w systemach 64-bitowych, ale w niektórych przypadkach mogą nie działać poprawnie.

Note:
Wstępnie skompilowane binaria agenta Zabbix są dostępne do pobrania dla systemów macOS, IBM AIX, FreeBSD, OpenBSD i Solaris. Starsze binaria, zgodne z bieżącą wersją serwer/proxy Zabbix, są dostępne dla NetBSD oraz HP-UX.

Jeśli zainstalowano jako pakiet

Agent Zabbix działa jako proces demona. Agent można uruchomić, wykonując:

```
systemctl start zabbix-agent
```

To zadziała w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

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

Aby zatrzymać, uruchomić ponownie lub sprawdzić status agenta Zabbix, użyj następujących poleceń:

```
systemctl stop zabbix-agent
systemctl restart zabbix-agent
systemctl status zabbix-agent
```

Uruchamianie ręczne

Możesz uruchomić agent Zabbix, lokalizując plik binarny `zabbix_agentd` i uruchamiając go bezpośrednio; na przykład:

```
zabbix_agentd
```

Agent w systemach Windows

Agent Zabbix w systemach Windows działa jako usługa systemu Windows.

Instalacja

Agent Zabbix można zainstalować w systemie Windows jedną z następujących metod:

- Wcześniej skompilowane binaria agenta Zabbix - pobierz pakiet instalacyjny MSI agenta Zabbix i postępuj zgodnie z instrukcjami na stronie Instalacja agenta Windows z MSI;
- Źródła Zabbix - pobierz pliki źródłowe i postępuj zgodnie z instrukcjami na stronie Budowanie agenta Zabbix w systemie Windows.

Dodatkowe informacje na temat instalacji agenta Zabbix (z archiwum ZIP) jako usługi systemu Windows znajdują się na stronie Agent Zabbix w Microsoft Windows.

Opcje

Następujące parametry wiersza poleceń można używać z agentem Zabbix:

Parameter	Description
agent UNIX i Windows	
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego. Możesz użyć tej opcji, aby wskazać plik konfiguracyjny inny niż domyślny.
`-f --foreground`	Uruchom agenta Zabbix na pierwszym planie (domyślnie: true).
`-p --print`	Wyświetl znane pozycje i zakończ. Pamiętaj, że aby zwrócić również wyniki parametru użytkownika, musisz podać plik konfiguracyjny (jeśli nie znajduje się w domyślnej lokalizacji).
`-t --test <item key>`	Przetestuj określoną pozycję i zakończ. Pamiętaj, że aby zwrócić również wyniki parametru użytkownika, musisz podać plik konfiguracyjny (jeśli nie znajduje się w domyślnej lokalizacji).
`-T --test-config`	Sprawdź poprawność pliku konfiguracyjnego i zakończ.
`-h --help`	Wyświetl informacje pomocy.
`-V --version`	Wyświetl numer wersji.
Tylko agent UNIX	

Parameter	Description

`-R`
`--runtime-control`
`<option>`
	Wykonaj funkcje administracyjne. Zobacz runtime control.

Tylko agent Windows	
`-m` `--multiple-agents`	Użyj wielu instancji agenta (z opcjami `-i`, `-d`, `-s`, `-x`). Aby rozróżnić nazwy usług poszczególnych instancji, każda nazwa usługi będzie zawierać wartość Hostname z określonego pliku konfiguracyjnego.
`-S --startup-type` `<value>`	Ustaw typ uruchamiania usługi agenta Zabbix dla Windows. Dozwolone wartości: `automatic` - *(domyślnie)* uruchamiaj usługę automatycznie podczas startu Windows; `delayed` - opóźnij uruchomienie usługi do czasu zakończenia startu usług uruchamianych automatycznie (dostępne w Windows Server 2008/Vista i nowszych wersjach); `manual` - uruchamiaj usługę ręcznie (przez użytkownika lub aplikację); `disabled` - wyłącz usługę, aby nie mogła zostać uruchomiona przez użytkownika lub aplikację. Możesz użyć tej opcji razem z opcją `-i` albo osobno, aby zmodyfikować typ uruchamiania już zainstalowanej usługi.
`-i --install`	Zainstaluj agenta Zabbix dla Windows jako usługę.
`-d --uninstall`	Odinstaluj usługę agenta Zabbix dla Windows.
`-s --start`	Uruchom usługę agenta Zabbix dla Windows.
`-x --stop`	Zatrzymaj usługę agenta Zabbix dla Windows.

Konkretne przykłady użycia parametrów wiersza poleceń:

- Wyświetlenie wszystkich wbudowanych pozycji agenta wraz z wartościami.
- Testowanie parametru użytkownika z kluczem `mysql.ping` zdefiniowanym w określonym pliku konfiguracyjnym.
- Instalacja usługi Zabbix Agent dla Windows z użyciem domyślnej ścieżki do pliku konfiguracyjnego `C:\\Program Files\\Zabbix Agent\\zabbix_agentd.conf`.
- Instalacja usługi Zabbix Agent [Hostname] dla Windows z użyciem pliku konfiguracyjnego `zabbix_agentd.conf` znajdującego się w tym samym folderze co plik wykonywalny agenta oraz nadanie nazwie usługi unikalności przez rozszerzenie jej o wartość Hostname z pliku konfiguracyjnego.
- Modyfikacja typu uruchamiania zainstalowanej usługi Zabbix Agent dla Windows z użyciem pliku konfiguracyjnego `zabbix_agentd.conf` znajdującego się w tym samym folderze co plik wykonywalny agenta.

```
zabbix_agentd --print
zabbix_agentd -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
zabbix_agentd.exe -i
zabbix_agentd.exe -i -m -c zabbix_agentd.conf
zabbix_agentd.exe -c zabbix_agentd.conf -S delayed
```

Sterowanie w czasie działania

Za pomocą opcji sterowania w czasie działania można zmieniać poziom logowania procesów agent.

Option	Description	Target
`log_level_increase[<target>]`	Zwiększ poziom logowania. Jeśli target nie zostanie określony, dotyczy to wszystkich procesów.	Target można określić jako: **typ procesu** - wszystkie procesy określonego typu (np. `listener`) Zobacz wszystkie typy procesów agent. **typ procesu,N** - typ procesu i numer (np. `listener,3`) **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'typ procesu,N'.
`log_level_decrease[<target>]`	Zmniejsz poziom logowania. Jeśli target nie zostanie określony, dotyczy to wszystkich procesów.	
`userparameter_reload`	Wczytuje ponownie wartości opcji `UserParameter` i `Include` z bieżącego pliku konfiguracyjnego.	

Przykłady:

- Zwiększenie poziomu logowania wszystkich procesów

- Zwiększenie poziomu logowania trzeciego procesu listener
- Zwiększenie poziomu logowania procesu o PID 1234
- Zmniejszenie poziomu logowania wszystkich procesów aktywnych kontroli

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

Note:
Sterowanie w czasie działania nie jest obsługiwane w OpenBSD, NetBSD i Windows.

Typy procesów agenta

Typy procesów agenta są następujące:

- `active checks` - proces do wykonywania aktywnych kontroli;
- `collector` - proces do zbierania danych;
- `listener` - proces do nasłuchiwania pasywnych kontroli.

Plik dziennika agenta może służyć do obserwowania tych typów procesów.

Od Zabbix 7.4.6 plik dziennika agenta jest tworzony z uprawnieniami do odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są zabronione.

Użytkownik procesu

agent Zabbix w systemie UNIX jest zaprojektowany do działania jako użytkownik inny niż root. Będzie działał jako ten użytkownik inny niż root, jako który został uruchomiony. Możesz więc uruchomić agent jako dowolnego użytkownika innego niż root bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako `root`, przełączy się na zakodowanego na stałe użytkownika `zabbix`, który musi być obecny w twoim systemie. Możesz uruchomić agent jako `root` tylko wtedy, gdy odpowiednio zmodyfikujesz parametr `AllowRoot` w pliku konfiguracyjnym agenta.

Plik konfiguracyjny

Domyślna ścieżka do pliku konfiguracyjnego to:

- W systemach UNIX domyślnie jest to `/usr/local/etc/zabbix_agentd.conf` lub wartość ustawiona przez zmienne compile-time `--sysconfdir` albo `--prefix`.
- W systemie Windows domyślnie jest to `C:\\Program Files\\Zabbix Agent\\zabbix_agentd.conf`.

Jeśli na host działa pojedyncza instancja agenta, może ona używać domyślnego pliku konfiguracyjnego lub pliku konfiguracyjnego określonego w wierszu poleceń. W przypadku wielu instancji każda instancja agenta musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Szczegółowe informacje o konfiguracji agenta Zabbix znajdują się w opcjach pliku konfiguracyjnego dla agenta Zabbix dla Unix lub agenta Zabbix dla Windows.

Ustawienia regionalne

Należy pamiętać, że agent wymaga ustawień regionalnych UTF-8, aby niektóre tekstowe pozycje agenta mogły zwracać oczekiwaną zawartość. Większość nowoczesnych systemów typu Unix domyślnie używa ustawień regionalnych UTF-8, jednak istnieją systemy, w których może być konieczne ich jawne skonfigurowanie.

Kod wyjścia

Zabbix agent zwraca 0 w przypadku pomyślnego zakończenia i 1 w przypadku niepowodzenia.

3 Agent 2

Przegląd

Zabbix agent 2 to nowa generacja Zabbix agent, napisana w Go (z wykorzystaniem części kodu C odziedziczonego po Zabbix agent). Został zaprojektowany, aby:

- Zmniejszyć liczbę połączeń TCP;
- Zapewnić lepszą równoległość sprawdzeń;

- Być łatwo rozszerzalny za pomocą wtyczek, które zapewniają proste sprawdzenia przy minimalnej ilości kodu oraz obsługują złożone sprawdzenia składające się z długo działających skryptów i samodzielnego zbierania danych z okresowym raportowaniem;
- Działać jako zamiennik dla Zabbix agent, obsługując wszystkie wcześniejsze funkcje.

Wszystkie metryki obsługiwane przez Zabbix agent 2 są zbierane przez wtyczki.

Pasywne i aktywne sprawdzenia

Zabbix agent 2 obsługuje pasywne i aktywne sprawdzenia, podobnie jak Zabbix agent. Dodatkowo aktywne sprawdzenia Zabbix agent 2 obsługują elastyczne/interwały harmonogramu oraz współbieżność sprawdzeń w ramach jednego aktywnego serwera.

Note:
Domyślnie po ponownym uruchomieniu Zabbix agent 2 zaplanuje pierwsze pobranie danych dla aktywnych sprawdzeń w warunkowo losowym czasie w obrębie interwału aktualizacji pozycji, aby zapobiec skokom użycia zasobów. Aby wykonać aktywne sprawdzenia, które nie mają *Harmonogramu* interwału aktualizacji bezpośrednio po ponownym uruchomieniu agenta, ustaw parametr `ForceActiveChecksOnStart` (na poziomie globalnym) lub `Plugins.<Plugin name>.System.ForceActiveChecksOnStart` (dotyczy tylko określonych sprawdzeń wtyczki) w pliku konfiguracyjnym. Parametr na poziomie wtyczki, jeśli jest ustawiony, zastąpi parametr globalny.

Sprawdzanie współbieżności

Sprawdzenia z różnych wtyczek mogą być wykonywane równolegle. Liczba równoczesnych sprawdzeń w ramach jednej wtyczki jest ograniczona przez ustawienie pojemności wtyczki. Każda wtyczka może mieć zakodowane na stałe ustawienie pojemności (1000 jako wartość domyślna), które można obniżyć za pomocą ustawienia `Plugins.<PluginName>.System.Capacity=N` w parametrze konfiguracji `Plugins`.

Obsługiwane platformy

Informacje o obsługiwanych platformach można znaleźć na stronie Wymagania.

Agent 2 w systemach uniksopodobnych

Zabbix agent 2 w systemach uniksopodobnych jest uruchamiany na host monitorowanym.

Instalacja

Zabbix agent 2 można zainstalować w systemach opartych na Linuxie, korzystając z jednej z następujących metod:

- Pakiety Zabbix - wybierz komponent Agent 2 (po wybraniu wersji Zabbix, dystrybucji systemu operacyjnego i wersji systemu operacyjnego) i postępuj zgodnie z instrukcjami;
- Źródła Zabbix - pobierz pliki źródłowe i skompiluj agent, konfigurując go z opcją `--enable-agent2`.

Note:
Możliwości monitorowania Zabbix agent 2 można rozszerzyć za pomocą ładowalnych wtyczek, które są dostępne osobno. Szczegóły można znaleźć w sekcji Ładowalne wtyczki.

Jeśli zainstalowano jako pakiet

Zabbix agent 2 działa jako proces pierwszoplanowy i do działania w tle korzysta z zewnętrznego menedżera usług (np. systemd). Zabbix agent 2 nie ma wbudowanej obsługi demonizacji w systemie Linux.

Agent można uruchomić, wykonując:

```
systemctl start zabbix-agent2
```

Aby zatrzymać, uruchomić ponownie lub sprawdzić stan Zabbix agent 2, użyj następujących poleceń:

```
systemctl stop zabbix-agent2
systemctl restart zabbix-agent2
systemctl status zabbix-agent2
```

Uruchomienie ręczne

Możesz uruchomić agent Zabbix, lokalizując plik binarny `zabbix_agent2` i uruchamiając go bezpośrednio; na przykład:

```
zabbix_agent2
```

Agent 2 w systemach Windows

Zabbix agent 2 działa jako samodzielny proces; może jednak również działać jako usługa systemu Windows.

Instalacja

Zabbix agent 2 można zainstalować w systemie Windows jedną z następujących metod:

- Wstępnie skompilowane binaria Zabbix agent - pobierz pakiet instalacyjny MSI agent i postępuj zgodnie z instrukcjami na stronie Instalacja agenta Windows z MSI;
- Źródła Zabbix - pobierz pliki źródłowe i postępuj zgodnie z instrukcjami na stronie Kompilowanie Zabbix agent 2 w systemie Windows.

Note:
Możliwości monitorowania Zabbix agent 2 można rozszerzyć za pomocą ładowalnych wtyczek, które są dostępne osobno. Szczegóły znajdują się na stronie Ładowalne wtyczki.

Dodatkowe informacje na temat instalacji Zabbix agent 2 (z archiwum ZIP) jako usługi systemu Windows znajdują się na stronie Zabbix agent w Microsoft Windows.

Opcje

Następujące parametry wiersza poleceń można używać z agentem Zabbix 2:

Parameter	Description
agent UNIX i Windows	
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego. Możesz użyć tej opcji, aby wskazać plik konfiguracyjny inny niż domyślny.
`-f --foreground`	Uruchom agenta Zabbix na pierwszym planie (domyślnie: true).
`-p --print`	Wyświetl znane pozycje i zakończ. Pamiętaj, że aby zwrócić również wyniki user parameter, musisz podać plik konfiguracyjny (jeśli nie znajduje się w domyślnej lokalizacji).
`-t --test <item key>`	Przetestuj określoną pozycję i zakończ. Pamiętaj, że aby zwrócić również wyniki user parameter, musisz podać plik konfiguracyjny (jeśli nie znajduje się w domyślnej lokalizacji).
`-T --test-config`	Sprawdź poprawność pliku konfiguracyjnego i zakończ.
`-h --help`	Wyświetl informacje pomocy i zakończ.
`-v --verbose`	Wyświetl informacje debugowania. Użyj tej opcji razem z opcjami `-p` i `-t`.
`-V --version`	Wyświetl wersję agenta i informacje o licencji.
`-R --runtime-control <option>`	Wykonaj funkcje administracyjne. Zobacz runtime control.
Tylko agent Windows	
`-m --multiple-agents`	Użyj wielu instancji agenta (z opcjami `-i`, `-d`, `-s`, `-x`). Aby rozróżnić nazwy usług poszczególnych instancji, każda nazwa usługi będzie zawierać wartość Hostname z określonego pliku konfiguracyjnego.
`-S --startup-type <value>`	Ustaw typ uruchamiania usługi agenta Zabbix dla Windows. Dozwolone wartości: `automatic` - *(domyślnie)* uruchamiaj usługę automatycznie podczas startu systemu Windows; `delayed` - opóźnij uruchomienie usługi do czasu zakończenia startu usług uruchamianych automatycznie; `manual` - uruchamiaj usługę ręcznie (przez użytkownika lub aplikację); `disabled` - wyłącz usługę, aby nie mogła zostać uruchomiona przez użytkownika lub aplikację. Możesz użyć tej opcji razem z opcją `-i` albo osobno, aby zmodyfikować typ uruchamiania już zainstalowanej usługi.
`-i --install`	Zainstaluj agenta Zabbix dla Windows jako usługę.
`-d --uninstall`	Odinstaluj usługę agenta Zabbix dla Windows.
`-s --start`	Uruchom usługę agenta Zabbix dla Windows.
`-x --stop`	Zatrzymaj usługę agenta Zabbix dla Windows.

Przykłady użycia parametrów wiersza poleceń:

- Wyświetl wszystkie wbudowane pozycje agenta wraz z wartościami.
- Przetestuj user parameter z kluczem `mysql.ping` zdefiniowanym w określonym pliku konfiguracyjnym.
- Instalowanie usługi Zabbix Agent dla Windows z użyciem domyślnej ścieżki do pliku konfiguracyjnego `C:\\Program Files\\Zabbix Agent 2\\zabbix_agent2.conf`.
- Modyfikowanie typu uruchamiania zainstalowanej usługi Zabbix Agent dla Windows z użyciem pliku konfiguracyjnego `zabbix_agent2.conf` znajdującego się w tym samym folderze co plik wykonywalny agenta.

```
zabbix_agent2 --print
zabbix_agent2 -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
zabbix_agent2.exe -i
zabbix_agent2.exe -c zabbix_agent2.conf -S delayed
```

Sterowanie w czasie działania

Sterowanie w czasie działania udostępnia kilka opcji zdalnego sterowania.

Option	Description
`log_level_increase`	Zwiększa poziom logowania.
`log_level_decrease`	Zmniejsza poziom logowania.
`metrics`	Wyświetla dostępne metryki.
`version`	Wyświetla wersję agenta.
`userparameter_reload`	Przeładowuje wartości opcji `UserParameter` i `Include` z bieżącego pliku konfiguracyjnego.
`help`	Wyświetla informacje pomocy dotyczące sterowania w czasie działania.

Przykłady:

- Zwiększenie poziomu logowania dla agenta 2.
- Wyświetlenie opcji sterowania w czasie działania.

```
zabbix_agent2 -R log_level_increase
zabbix_agent2 -R help
```

Od Zabbix 7.4.1 polecenia sterowania w czasie działania agenta 2 zapisują wynik do `stdout` (standardowe wyjście) zamiast do `stderr` (standardowe wyjście błędów).

Plik konfiguracyjny

Domyślna ścieżka do pliku konfiguracyjnego to:

- W systemach UNIX domyślnie jest to `/usr/local/etc/zabbix_agent2.conf` lub wartość ustawiona przez zmienne czas kompilacji `--sysconfdir` albo `--prefix`.
- W systemie Windows domyślnie jest to `C:\\Program Files\\Zabbix Agent 2\\zabbix_agent2.conf`.

Jeśli na hoście działa pojedyncza instancja agenta, może ona używać domyślnego pliku konfiguracyjnego lub pliku konfiguracyjnego określonego w wierszu poleceń. W przypadku wielu instancji każda instancja agenta musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Parametry konfiguracyjne Zabbix agent 2 w większości odpowiadają parametrom Zabbix agent, z pewnymi wyjątkami:

Parameters	Description
Parametry konfiguracyjne specyficzne dla Zabbix agent 2	
`ControlSocket`	Ścieżka do gniazda sterowania w czasie działania. Zabbix agent 2 używa gniazda sterowania do poleceń w czasie działania.
`EnablePersistentBuffer`, `PersistentBufferFile`, `PersistentBufferPeriod`	Parametry te służą do konfiguracji trwałego przechowywania w Zabbix agent 2 dla aktywnych pozycji.
`ForceActiveChecksOnStart`	Określa, czy agent ma wykonywać aktywne sprawdzenia natychmiast po restarcie, czy rozkładać je równomiernie w czasie.

Parameters		Description
	Plugins	Wtyczki mogą mieć własne parametry w formacie Plugins.<Plugin name>.<Parameter>=<value>. Częstym parametrem wtyczki jest System.Capacity, który ustawia limit sprawdzeń możliwych do wykonania jednocześnie.
	StatusPort	Port, na którym Zabbix agent 2 będzie nasłuchiwał żądań HTTP dotyczących stanu oraz wyświetlania listy skonfigurowanych wtyczek i niektórych parametrów wewnętrznych.
Parametry konfiguracyjne specyficzne dla Zabbix agent, nieobsługiwane w Zabbix agent 2		
	AllowRoot, User	Nieobsługiwane w Zabbix agent 2, ponieważ nie obsługuje on demonizacji.
	EnableRemoteCommands	Przestarzałe; zamiast tego w obu agentach używane są parametry AllowKey i DenyKey.
	ListenBacklog	Używane w Zabbix agent do definiowania maksymalnej liczby oczekujących połączeń w kolejce TCP. Nieobsługiwane w Zabbix agent 2.
	LoadModule, LoadModulePath	Moduły ładowane dynamicznie nie są obsługiwane w Zabbix agent 2.
	MaxLinesPerSecond, LogRemoteCommands	W Zabbix agent 2 zamiast tego używane są Plugins.Log.MaxLinesPerSecond i Plugins.SystemRun.LogRemoteCommands.
	StartAgents	Ten parametr jest używany w Zabbix agent do zwiększania współbieżności pasywnych sprawdzeń lub ich wyłączania. W Zabbix agent 2 współbieżność jest konfigurowana na poziomie wtyczki i może być ograniczona ustawieniem pojemności. Pasywne sprawdzenia są wyłączone w Zabbix Agent 2, jeśli nie określono parametru Server.

Więcej informacji można znaleźć w opcjach pliku konfiguracyjnego dla Zabbix agent2.

Kody wyjścia

Zabbix agent 2 może być również skompilowany ze starszymi wersjami OpenSSL (1.0.1, 1.0.2).

W takim przypadku Zabbix udostępnia mutexy do blokowania w OpenSSL. Jeśli blokada lub odblokowanie mutexa zakończy się niepowodzeniem, komunikat o błędzie zostanie wypisany do standardowego strumienia błędów (STDERR), a Agent 2 zakończy działanie z kodem zwrotnym 2 lub 3, odpowiednio.

4 Proxy

Przegląd

Zabbix proxy to proces, który może zbierać dane monitoringu z jednego lub większej liczby monitorowanych urządzeń i wysyłać

te informacje do serwera Zabbix, działając zasadniczo w imieniu serwera. Wszystkie zebrane dane są buforowane lokalnie, a następnie przesyłane do serwera Zabbix, do którego należy dany proxy.

Wdrożenie proxy jest opcjonalne, ale może być bardzo korzystne dla rozłożenia obciążenia pojedynczego serwera Zabbix. Jeśli dane zbierają wyłącznie proxy, przetwarzanie na serwerze wymaga mniej zasobów CPU i operacji wejścia/wyjścia dysku.

Zabbix proxy jest idealnym rozwiązaniem do scentralizowanego monitorowania zdalnych lokalizacji, oddziałów i sieci bez lokalnych administratorów.

Zabbix proxy wymaga oddzielnej bazy danych.

Attention:
Należy pamiętać, że bazami danych obsługiwanymi przez Zabbix proxy są SQLite, MySQL i PostgreSQL.

Zobacz także: Używanie proxy w środowisku rozproszonym

Uruchamianie proxy

Jeśli zainstalowano jako pakiet

Zabbix proxy działa jako proces demona. Proxy można uruchomić, wykonując:

```
systemctl start zabbix-proxy
```

To zadziała w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

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

Podobnie, aby zatrzymać/uruchomić ponownie/wyświetlić status Zabbix proxy, użyj następujących poleceń:

```
systemctl stop zabbix-proxy
systemctl restart zabbix-proxy
systemctl status zabbix-proxy
```

Uruchamianie ręczne

Jeśli powyższe nie działa, należy uruchomić go ręcznie. Znajdź ścieżkę do pliku binarnego `zabbix_proxy` i wykonaj:

```
zabbix_proxy
```

Z proxy Zabbix można używać następujących parametrów wiersza poleceń:

```
-c --config <file>              Ścieżka do pliku konfiguracyjnego
-f --foreground                 Uruchom proxy Zabbix na pierwszym planie
-R --runtime-control <option>   Wykonaj funkcje administracyjne
-T --test-config                Sprawdź poprawność pliku konfiguracyjnego i zakończ
-h --help                       Wyświetl tę pomoc
-V --version                    Wyświetl numer wersji
```

Przykłady uruchamiania proxy Zabbix z parametrami wiersza poleceń:

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

Sterowanie w czasie działania

Opcje sterowania w czasie działania:

Option	Description	Target
`config_cache_reload`	Przeładuj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecnie ładowana. Aktywny proxy Zabbix połączy się z serwerem Zabbix i zażąda danych konfiguracji. Pasywny proxy Zabbix zażąda danych konfiguracji z serwera Zabbix przy następnym połączeniu serwera z proxy.	

Option	Description	Target
`history_cache_clear=<id>`	Wyczyść pamięć podręczną historii dla pozycji określonej przez jej ID. Dotyczy wszystkich wartości pozycji, z wyjątkiem pierwszej i ostatniej wartości.	**target** - ID pozycji.
`diaginfo[=<section>]`	Zbierz informacje diagnostyczne w pliku dziennika proxy.	`historycache` - statystyki pamięci podręcznej historii; `preprocessing` - statystyki menedżera preprocessingu; `locks` - lista mutexów (pusta w systemach *BSD*).
`snmp_cache_reload`	Przeładuj pamięć podręczną SNMP - wyczyść właściwości silnika SNMP (engine time, engine boots, engine id, credentials) dla wszystkich hostów. Użyj, aby wymusić globalne wyczyszczenie pamięci podręcznej podczas rozwiązywania problemów z SNMP.	
`housekeeper_execute`	Uruchom procedurę housekeeping. Ignorowane, jeśli procedura housekeeping jest obecnie w toku.	
`log_level_increase[=<target>]`	Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	**process type** - wszystkie procesy określonego typu (np. `poller`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `poller,3`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.
`log_level_decrease[=<target>]`	Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	
`prof_enable[=<target>]`	Włącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony. Włączone profilowanie dostarcza szczegółów wszystkich rwlocks/mutexes według nazwy funkcji.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `history syncer,1`) **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'. **scope** - `rwlock`, `mutex`, `processing` mogą być używane z typem procesu i numerem (np. `history syncer,1,processing`) lub ze wszystkimi procesami danego typu (np. `history syncer,rwlock`).
`prof_disable[=<target>]`	Wyłącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej konfiguracji proxy:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R config_cache_reload
```

Przykład użycia sterowania w czasie działania do wyczyszczenia pamięci podręcznej historii dla pozycji:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R history_cache_clear=42243
```

Przykłady użycia sterowania w czasie działania do zbierania informacji diagnostycznych:

```
### Zbierz wszystkie dostępne informacje diagnostyczne w pliku dziennika proxy:
zabbix_proxy -R diaginfo
```

```
### Zbierz statystyki pamięci podręcznej historii w pliku dziennika proxy:
zabbix_proxy -R diaginfo=historycache
```

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej SNMP:

```
zabbix_proxy -R snmp_cache_reload
```

Attention:
Gdy interfejs SNMPv3 jest aktualizowany za pośrednictwem interfejsu Zabbix UI, Zabbix w większości przypadków automatycznie przeładuje nowe poświadczenia SNMPv3 dla tego interfejsu; używaj `-R snmp_cache_reload` tylko wtedy, gdy odpytywanie nadal kończy się niepowodzeniem po zmianie poświadczeń (na przykład z powodu niespójności engineBoots/engineID lub urządzeń niezgodnych z RFC) albo gdy chcesz wymusić globalne wyczyszczenie pamięci podręcznej SNMP podczas rozwiązywania problemów.

Przykład użycia sterowania w czasie działania do uruchomienia housekeeper:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R housekeeper_execute
```

Przykłady użycia sterowania w czasie działania do zmiany poziomu logowania:

```
### Zwiększ poziom logowania wszystkich procesów:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase

### Zwiększ poziom logowania drugiego procesu poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=poller,2

### Zwiększ poziom logowania procesu o PID 1234:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=1234

### Zmniejsz poziom logowania wszystkich procesów http poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_decrease="http poller"
```

Użytkownik procesu

Zabbix proxy jest zaprojektowany do działania jako użytkownik inny niż root. Będzie działać jako dowolny użytkownik inny niż root, jako który zostanie uruchomiony. Możesz więc uruchomić proxy jako dowolnego użytkownika innego niż root bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako `root`, przełączy się na zakodowanego na stałe użytkownika `zabbix`, który musi być obecny w twoim systemie. Możesz uruchomić proxy jako `root` tylko wtedy, gdy odpowiednio zmodyfikujesz parametr `AllowRoot` w pliku konfiguracji proxy.

Plik konfiguracyjny

Szczegółowe informacje dotyczące konfiguracji `zabbix_proxy` można znaleźć w opcjach pliku konfiguracyjnego.

Typy procesów i wątki proxy

- `agent poller` - asynchroniczny proces pollera dla kontroli pasywnych z wątkiem roboczym;
- `availability manager` - proces aktualizujący dostępność hostów;
- `browser poller` - poller do kontroli pozycji przeglądarkowych;
- `configuration syncer` - proces zarządzający pamięcią podręczną danych konfiguracyjnych w pamięci;
- `data sender` - wysyłacz danych proxy;
- `discovery manager` - proces zarządzający wykrywaniem urządzeń;
- `discovery worker` - proces obsługujący zadania wykrywania z procesu discovery manager;
- `history syncer` - zapisujący dane historyczne do bazy danych;
- `housekeeper` - proces usuwający nieaktualną historię pozycji;
- `http agent poller` - asynchroniczny proces pollera dla kontroli HTTP z wątkiem roboczym;
- `http poller` - poller monitorowania WWW;
- `icmp pinger` - poller dla kontroli icmpping;
- `internal poller` - poller dla kontroli wewnętrznych;
- `ipmi manager` - menedżer pollera IPMI;
- `ipmi poller` - poller dla kontroli IPMI;
- `java poller` - poller dla kontroli Java;
- `odbc poller` - poller dla kontroli ODBC;
- `poller` - zwykły poller dla kontroli pasywnych;
- `preprocessing manager` - menedżer zadań przetwarzania wstępnego z wątkami roboczymi przetwarzania wstępnego;

- `preprocessing worker` - wątek do przetwarzania wstępnego danych;
- `self-monitoring` - proces zbierający wewnętrzne statystyki serwera;
- `snmp poller` - asynchroniczny proces pollera dla kontroli SNMP z wątkiem roboczym (tylko pozycje `walk[OID]` i `get[OID]`);
- `snmp trapper` - trapper dla pułapek SNMP;
- `task manager` - proces do zdalnego wykonywania zadań żądanych przez inne komponenty (np. zamknięcie problemu, potwierdzenie problemu, sprawdzenie wartości pozycji teraz, funkcjonalność zdalnych poleceń);
- `trapper` - trapper dla kontroli aktywnych, pułapek i komunikacji z proxy;
- `unreachable poller` - poller dla niedostępnych urządzeń;
- `vmware collector` - kolektor danych VMware odpowiedzialny za zbieranie danych z usług VMware.

Plik dziennika proxy może służyć do obserwowania tych typów procesów.

Od Zabbix 7.4.6 plik dziennika proxy jest tworzony z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są odrzucone.

Różne typy procesów proxy Zabbix można monitorować za pomocą wewnętrznej pozycji `zabbix[process,<type>,<mode>,<state>]`.

Statystyki transakcji synchronizatora historii

Tytuł procesu synchronizatora historii wyświetla szczegółowe statystyki dotyczące transakcji synchronizatora historii.

```
205276 ?        S      0:00  zabbix_proxy: history syncer #1 [processed 1 values in 0.001179 (0.001167,0.0
205277 ?        S      0:00  zabbix_proxy: history syncer #2 [processed 0 values in 0.000022 (0.000000,0.0
```

Czasy w formacie `processed...in N (<timings>) sec` oznaczają:

- Czas spędzony na zapisywaniu wartości pozycji do bazy danych.
- Czas spędzony na aktualizowaniu danych pozycji (stan, błędy).

Obsługiwane platformy

Zabbix proxy działa na tej samej liście obsługiwanych platform co serwer Zabbix.

Bufor pamięci

Bufor pamięci umożliwia przechowywanie nowych danych (wartości pozycji, wykrywanie sieci, automatyczna rejestracja hosta) w buforze i wysyłanie ich do serwera Zabbix bez dostępu do bazy danych. Bufor pamięci został wprowadzony dla proxy w Zabbix 7.0.

W instalacjach sprzed Zabbix 7.0 zebrane dane były przechowywane w bazie danych przed wysłaniem do serwera Zabbix. W takich instalacjach pozostaje to domyślnym zachowaniem po uaktualnieniu do Zabbix 7.0.

Aby uzyskać zoptymalizowaną wydajność, zaleca się skonfigurowanie użycia bufora pamięci na proxy. Można to zrobić, zmieniając wartość `ProxyBufferMode` z `disk` (wbudowana wartość domyślna dla istniejących instalacji) na `hybrid` (zalecane) lub `memory`. Wymagane jest również ustawienie rozmiaru bufora pamięci (parametr `ProxyMemoryBufferSize`).

W trybie hybrydowym bufor jest chroniony przed utratą danych przez zapisywanie niewysłanych danych do bazy danych, jeśli proxy zostanie zatrzymany, bufor się zapełni lub dane staną się zbyt stare. Gdy wszystkie wartości zostaną zapisane do bazy danych, proxy wraca do korzystania z bufora pamięci.

W trybie pamięciowym będzie używany bufor pamięci, jednak nie ma ochrony przed utratą danych. Jeśli proxy zostanie zatrzymany lub pamięć zostanie przepełniona, niewysłane dane zostaną odrzucone.

Tryb hybrydowy (`ProxyBufferMode=hybrid`) jest stosowany do wszystkich nowych instalacji od Zabbix 7.0.

Dodatkowe parametry, takie jak `ProxyMemoryBufferSize` i `ProxyMemoryBufferAge`, określają odpowiednio rozmiar bufora pamięci oraz maksymalny wiek danych w buforze.

Należy pamiętać, że w przypadku sprzecznej konfiguracji proxy wyświetli błąd i nie uruchomi się, na przykład jeśli:

- `ProxyBufferMode` jest ustawiony na `hybrid` lub `memory`, a `ProxyMemoryBufferSize` ma wartość 0.
- `ProxyBufferMode` jest ustawiony na `hybrid` lub `memory`, a `ProxyLocalBuffer` nie ma wartości 0.

Ustawienia regionalne

Zwróć uwagę, że proxy wymaga ustawień regionalnych UTF-8, aby niektóre tekstowe pozycje mogły być poprawnie interpretowane. Większość nowoczesnych systemów typu Unix ma domyślnie ustawienia regionalne UTF-8, jednak istnieją systemy, w których trzeba je skonfigurować ręcznie.

Obliczanie kolejek podczas konserwacji

Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły znajdują się w sekcji Obliczanie kolejek podczas konserwacji.

5 Java gateway

Przegląd

Zabbix Java gateway można zainstalować z kodu źródłowego lub pakietów.

Natywna obsługa monitorowania aplikacji JMX jest dostępna w postaci demona Zabbix o nazwie "Zabbix Java gateway". Zabbix Java gateway to demon napisany w języku Java. Aby odczytać wartość określonego licznika JMX na host, serwer Zabbix wysyła zapytanie do Zabbix Java gateway, który używa JMX management API do zdalnego odpytywania interesującej aplikacji. Aplikacja nie wymaga instalacji dodatkowego oprogramowania, musi jedynie zostać uruchomiona z opcją `-Dcom.sun.management.jmxremote` w wierszu poleceń.

Java gateway akceptuje połączenia przychodzące z serwer Zabbix lub proxy i może być używany wyłącznie jako "passive proxy". W przeciwieństwie do Zabbix proxy, może być również używany z poziomu Zabbix proxy (proxy Zabbix nie mogą być łączone w łańcuch). Dostęp do każdego Java gateway jest konfigurowany bezpośrednio w pliku konfiguracyjnym serwer Zabbix lub proxy, dlatego dla serwer Zabbix lub proxy Zabbix można skonfigurować tylko jeden Java gateway. Jeśli host będzie miał pozycje typu *JMX agent* oraz pozycje innego typu, tylko pozycje *JMX agent* zostaną przekazane do Java gateway w celu pobrania.

Gdy pozycja musi zostać zaktualizowana za pośrednictwem Java gateway, serwer Zabbix lub proxy połączy się z Java gateway i zażąda wartości, którą Java gateway następnie pobiera i przekazuje z powrotem do serwera lub proxy. W związku z tym Java gateway nie buforuje żadnych wartości.

Serwer Zabbix lub proxy ma określony typ procesów, które łączą się z Java gateway, kontrolowany przez opcję `StartJavaPollers`. Wewnętrznie Java gateway uruchamia wiele wątków, kontrolowanych przez opcję `START_POLLERS` option. Po stronie serwera, jeśli połączenie trwa dłużej niż "Timeout" sekund, zostanie zakończone, ale Java gateway może nadal być zajęty pobieraniem wartości z licznika JMX. Aby to rozwiązać, w Java gateway dostępna jest opcja `TIMEOUT`, która umożliwia ustawienie limitu czasu dla operacji sieciowych JMX.

Serwer Zabbix lub proxy będzie próbował możliwie najbardziej grupować żądania do pojedynczego celu JMX (zależnie od interwałów pozycji) i wysyłać je do Java gateway w jednym połączeniu, aby uzyskać lepszą wydajność.

Zaleca się, aby `StartJavaPollers` było mniejsze lub równe `START_POLLERS`; w przeciwnym razie mogą wystąpić sytuacje, w których w Java gateway nie będzie dostępnych żadnych wątków do obsługi przychodzących żądań. W takich przypadkach Java gateway używa `ThreadPoolExecutor.CallerRunsPolicy`, co oznacza, że główny wątek obsłuży przychodzące żądanie i tymczasowo nie będzie przyjmował żadnych nowych żądań.

Jeśli próbujesz monitorować aplikacje Java oparte na Wildfly za pomocą Zabbix Java gateway, zainstaluj najnowszy plik `jboss-client.jar` dostępny na stronie pobierania Wildfly.

1 Instalacja z pakietów RHEL

Przegląd

Jeśli zainstalowano z pakietów, poniższe informacje pomogą w konfiguracji Zabbix Java gateway.

Konfigurowanie i uruchamianie Java gateway

Parametry konfiguracji Zabbix Java gateway można dostosować w pliku:

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

Więcej informacji można znaleźć w sekcji konfiguracji Zabbix Java gateway parameters.

Aby uruchomić Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Aby automatycznie uruchamiać Zabbix Java gateway podczas startu systemu:

RHEL 7 i nowsze:

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

RHEL starsze niż 7:

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

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa, należy wskazać serwerowi Zabbix, gdzie ma znaleźć Zabbix Java gateway. Robi się to przez określenie parametrów JavaGateway i JavaGatewayPort w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, parametry połączenia należy określić zamiast tego w pliku konfiguracyjnym proxy.

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

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wstępnie rozwidlonych instancji Java pollers. Robi się to w taki sam sposób, w jaki określa się zwykłe pollers i trappers.

```
StartJavaPollers=5
```

Nie zapomnij ponownie uruchomić serwera lub proxy po zakończeniu konfiguracji.

Debugowanie Java gateway

Plik dziennika Java gateway Zabbix to:

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

Jeśli chcesz zwiększyć poziom rejestrowania, edytuj plik:

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

i zmień info w level="info" na debug lub nawet trace (w przypadku zaawansowanego rozwiązywania problemów):

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

</configuration>
```

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

2 Instalacja z pakietów Debian/Ubuntu

Przegląd

Jeśli zainstalowano z pakietów, poniższe informacje pomogą w konfiguracji Java gateway Zabbix.

Konfigurowanie i uruchamianie Java gateway

Konfigurację Java gateway można dostosować w pliku:

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

Więcej informacji można znaleźć w sekcji dotyczącej parametrów konfiguracji Zabbix Java gateway.

Aby uruchomić Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Aby automatycznie uruchamiać Zabbix Java gateway podczas startu systemu:

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

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa, należy wskazać serwerowi Zabbix, gdzie ma znaleźć Zabbix Java gateway. Robi się to przez określenie parametrów JavaGateway i JavaGatewayPort w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, parametry połączenia należy określić zamiast tego w pliku konfiguracyjnym proxy.

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

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wstępnie rozwidlonych instancji Java pollers. Robi się to w taki sam sposób, w jaki określa się zwykłe pollers i trappers.

```
StartJavaPollers=5
```

Nie zapomnij ponownie uruchomić serwera lub proxy po zakończeniu konfiguracji.

Debugowanie Java gateway

Plik dziennika Zabbix Java gateway to:

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

Jeśli chcesz zwiększyć poziom logowania, edytuj plik:

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

i zmień `info` w `level="info"` na debug lub nawet `trace` (do zaawansowanego rozwiązywania problemów):

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

</configuration>
```

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

3 Instalacja z kodu źródłowego

Przegląd

Jeśli zainstalowano ze źródeł, poniższe informacje pomogą w konfiguracji Zabbix Java gateway.

Przegląd plików

Jeśli pobrano Java gateway ze źródeł, powinien zostać utworzony zestaw skryptów powłoki, plików JAR i plików konfiguracyjnych w katalogu `$PREFIX/sbin/zabbix_java`. Rola tych plików została podsumowana poniżej.

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

Sam plik JAR Java gateway.

```
lib/logback-core-1.5.16.jar
lib/logback-classic-1.5.16.jar
lib/slf4j-api-2.0.16.jar
lib/android-json-4.3_r3.1.jar
```

Zależności Java gateway: biblioteka Logback, SLF4J oraz Android JSON.

```
lib/logback.xml
lib/logback-console.xml
```

Pliki konfiguracyjne dla Logback.

```
shutdown.sh
startup.sh
```

Pomocnicze skrypty do uruchamiania i zatrzymywania Java gateway.

```
settings.sh
```

Plik konfiguracyjny, który jest wczytywany przez powyższe skrypty startup i shutdown.

Konfigurowanie i uruchamianie Java gateway

Domyślnie Java gateway nasłuchuje na porcie 10052. Jeśli planujesz uruchamiać Java gateway na innym porcie, możesz to określić w skrypcie `settings.sh`. Zobacz opis pliku konfiguracyjnego Java gateway, aby dowiedzieć się, jak określić tę i inne opcje.

Warning:
Port 10052 nie jest zarejestrowany w IANA.

Gdy ustawienia będą już odpowiednie, możesz uruchomić Java gateway, wykonując skrypt `startup.sh`:

```
./startup.sh
```

Analogicznie, gdy Java gateway nie będzie już potrzebny, uruchom skrypt `shutdown.sh`, aby go zatrzymać:

```
./shutdown.sh
```

Należy pamiętać, że w przeciwieństwie do serwer lub proxy, Java gateway jest lekki i nie wymaga bazy danych.

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa, należy wskazać serwerowi Zabbix, gdzie ma znaleźć Zabbix Java gateway. W tym celu należy określić parametry `JavaGateway` i `JavaGatewayPort` w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, parametry połączenia należy określić zamiast tego w pliku konfiguracyjnym proxy.

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

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wstępnie rozwidlonych instancji pollerów Java. Robi się to w taki sam sposób, jak określa się zwykłe pollery i trappers.

```
StartJavaPollers=5
```

Nie zapomnij ponownie uruchomić serwera lub proxy po zakończeniu konfiguracji.

Debugowanie Java gateway

W przypadku problemów z Java gateway lub gdy komunikat o błędzie dotyczący pozycja w frontendzie nie jest wystarczająco opisowy, warto zajrzeć do pliku dziennika Java gateway.

Domyślnie Java gateway zapisuje swoje działania do pliku `/tmp/zabbix_java.log` z poziomem logowania "info". Czasami te informacje nie są wystarczające i potrzebne są dane na poziomie logowania "debug". Aby zwiększyć poziom logowania, zmodyfikuj plik `lib/logback.xml` i zmień atrybut level znacznika `<root>` na debug:

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

Należy pamiętać, że w przeciwieństwie do serwer lub proxy Zabbix, po zmianie pliku `logback.xml` nie ma potrzeby ponownego uruchamiania Java gateway Zabbix - zmiany w `logback.xml` zostaną automatycznie uwzględnione. Po zakończeniu debugowania możesz przywrócić poziom logowania do "info".

Jeśli chcesz zapisywać logi do innego pliku lub całkowicie innego medium, na przykład bazy danych, dostosuj plik `logback.xml` do swoich potrzeb. Więcej informacji znajdziesz w Logback Manual.

Czasami w celach debugowania przydatne jest uruchomienie Java gateway jako aplikacji konsolowej zamiast demona. Aby to zrobić, zakomentuj zmienną `PID_FILE` w `settings.sh`. Jeśli `PID_FILE` zostanie pominięta, skrypt `startup.sh` uruchomi Java gateway jako aplikację konsolową i sprawi, że Logback będzie używać pliku `lib/logback-console.xml`, który nie tylko zapisuje logi do konsoli, ale ma również włączony poziom logowania "debug".

Na koniec należy pamiętać, że ponieważ Java gateway używa SLF4J do logowania, możesz zastąpić Logback wybranym przez siebie frameworkiem, umieszczając odpowiedni plik JAR w katalogu lib. Więcej informacji znajdziesz w SLF4J Manual.

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

6 Sender

Przegląd

Zabbix sender to narzędzie wiersza poleceń, które może być używane do wysyłania danych wydajnościowych do serwer Zabbix w celu ich przetworzenia.

Narzędzie jest zwykle używane w długo działających skryptach użytkownika do okresowego wysyłania danych o dostępności i wydajności.

Aby wysyłać wyniki bezpośrednio do serwer Zabbix lub proxy, należy skonfigurować pozycja typu *Zabbix trapper*.

Zobacz także bibliotekę Python dla Zabbix, która ma wbudowaną funkcjonalność działającą podobnie jak Zabbix sender.

Uruchamianie Zabbix sender

Przykład uruchomienia Zabbix UNIX sender:

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

gdzie:

- `z` - host serwera Zabbix (można też użyć adresu IP);
- `s` - techniczna nazwa monitorowanego hosta (zarejestrowana w frontend Zabbix);
- `k` - klucz pozycji;
- `o` - wartość do wysłania.

Attention:
Opcje zawierające spacje muszą być ujęte w podwójne cudzysłowy.

Zabbix sender może być używany do wysyłania wielu wartości z pliku wejściowego. Więcej informacji można znaleźć w stronie podręcznika Zabbix sender.

Jeśli określono plik konfiguracyjny, Zabbix sender używa wszystkich adresów zdefiniowanych w parametrze konfiguracyjnym agenta `ServerActive` do wysyłania danych. Jeśli wysyłanie do jednego adresu się nie powiedzie, sender próbuje wysłać dane na pozostałe adresy. Jeśli wysyłanie danych wsadowych do jednego adresu się nie powiedzie, kolejne partie nie są wysyłane na ten adres.

Zabbix sender akceptuje ciągi znaków w kodowaniu UTF-8 (zarówno w systemach uniksowych, jak i Windows) bez znaku kolejności bajtów (BOM) na początku pliku.

Zabbix sender w systemie Windows można uruchomić w podobny sposób:

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

Scenariusze wysyłania w czasie rzeczywistym `zabbix_sender` będą zbierać wiele wartości przekazanych w krótkich odstępach czasu i wysyłać je do serwera w jednym połączeniu. Wartość, która nie jest oddzielona od poprzedniej o więcej niż 0.2 sekundy, może zostać umieszczona w tym samym stosie, ale maksymalny czas odpytywania nadal wynosi 1 sekundę.

Note:
Zabbix sender zakończy działanie, jeśli w podanym pliku konfiguracyjnym znajduje się nieprawidłowy wpis parametru (niezgodny z notacją `parameter=value`).

Uruchamianie Zabbix sender z wykrywaniem niskiego poziomu

Przykład uruchomienia Zabbix sender do wysłania wartości w formacie JSON dla wykrywania niskiego poziomu:

```
./zabbix_sender -z 192.168.1.113 -s "Zabbix server" -k trapper.discovery.item -o '[{"{#FSNAME}":"/","{#FST
```

Aby to działało, reguła wykrywania niskiego poziomu musi mieć typ pozycji *Zabbix trapper* (w tym przykładzie z kluczem `trapper.discovery.item`).

7 Get

Przegląd

Zabbix get to narzędzie wiersza poleceń, które może być używane do komunikacji z agent i pobierania wymaganych informacji z agent.

Narzędzie to jest zwykle używane do rozwiązywania problemów z agentami Zabbix.

Zobacz także bibliotekę Python dla Zabbix, która ma wbudowaną funkcjonalność działania jak Zabbix get.

Uruchamianie Zabbix get

Przykład uruchomienia Zabbix get w systemie UNIX w celu pobrania wartości obciążenia procesora z agent:

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

Inny przykład uruchomienia Zabbix get do przechwycenia ciągu znaków ze strony internetowej:

```
cd bin
./zabbix_get -s 192.168.1.1 -p 10050 -k "web.page.regexp[www.example.com,,,\"USA: ([a-zA-Z0-9.-]+)\",,\1]"
```

Zwróć uwagę, że klucz pozycji zawiera tutaj spację, dlatego do oznaczenia klucza pozycji dla powłoki użyto cudzysłowów. Cudzysłowy nie są częścią klucza pozycji; zostaną usunięte przez powłokę i nie zostaną przekazane do Zabbix agent.

Jeśli klucz pozycji nie jest obsługiwany, Zabbix get zwróci kod wyjścia 1.

Zabbix get akceptuje następujące parametry wiersza poleceń:

```
-s --host <host name or IP>             Określ nazwę hosta lub adres IP hosta
-p --port <port number>                 Określ numer portu agent uruchomionego na hoście (domyślnie: 10050
-I --source-address <IP address>        Określ źródłowy adres IP
-t --timeout <seconds>                  Określ limit czasu. Prawidłowy zakres: 1-600 sekund (domyślnie: 30
-k --key <item key>                     Określ klucz pozycji, dla której ma zostać pobrana wartość
-P --protocol <value>                   Protokół używany do komunikacji z agent. Wartości:
                                            auto - połącz przy użyciu protokołu JSON, w razie niepowodzeni
                                            json - połącz przy użyciu protokołu JSON
                                            plaintext - połącz przy użyciu protokołu tekstowego, w którym
-h --help                               Wyświetl ten komunikat pomocy
-V --version                            Wyświetl numer wersji

--tls-connect <value>                   Sposób połączenia z agent. Wartości:
                                            unencrypted - połącz bez szyfrowania (domyślnie)
                                            psk - połącz przy użyciu TLS i klucza współdzielonego
                                            cert - połącz przy użyciu TLS i certyfikatu
--tls-ca-file <CA file>                 Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów
--tls-crl-file <CRL file>               Pełna ścieżka do pliku zawierającego unieważnione certyfikaty
--tls-agent-cert-issuer <cert issuer>   Dozwolony wystawca certyfikatu agent
--tls-agent-cert-subject <cert subject> Dozwolony temat certyfikatu agent
--tls-cert-file <cert file>             Pełna ścieżka do pliku zawierającego certyfikat lub łańcuch certyf
--tls-key-file <key file>               Pełna ścieżka do pliku zawierającego klucz prywatny
--tls-psk-identity <PSK-identity>       Unikalny ciąg znaków rozróżniający wielkość liter, używany do iden
--tls-psk-file <PSK-file>               Pełna ścieżka do pliku zawierającego klucz współdzielony
--tls-cipher13 <cipher-string>          Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3. Zastępuje
--tls-cipher <cipher-string>            Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfrów
```

Zobacz także stronę podręcznika Zabbix get, aby uzyskać więcej informacji.

Zabbix get w systemie Windows można uruchomić podobnie:

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

8 JS

Przegląd

`zabbix_js` to narzędzie wiersza poleceń, które można używać do testowania osadzonych skryptów.

To narzędzie wykona skrypt użytkownika z parametrem typu string i wyświetli wynik. Skrypty są wykonywane przy użyciu osadzonego silnika skryptowego Zabbix.

W przypadku błędów kompilacji lub wykonania `zabbix_js` wyświetli błąd na stderr i zakończy działanie z kodem 1.

Użycie

```
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` akceptuje następujące parametry wiersza poleceń:

```
-s, --script script-file          Określa nazwę pliku skryptu do wykonania
                                  Jeśli jako nazwę pliku zostanie podane "-", skrypt zostanie odczytany ze
-i, --input input-file            Określa nazwę pliku z informacjami wejściowymi
                                  Zawartość zostanie przekazana do skryptu jako wartość
                                  Wzajemnie wykluczające się z -p
                                  Jeśli jako nazwę pliku zostanie podane "-", skrypt zostanie odczytany ze
-p, --param input-param           Określa parametr wejściowy: zmienną, która zostanie przekazana do skrypt
                                  Wzajemnie wykluczające się z -i
-l, --loglevel log-level          Określa poziom logowania
                                  Zakres: 0-5
-t, --timeout timeout             Określa limit czasu w sekundach
                                  Prawidłowy zakres: 1-600 sekund (domyślnie: 10 sekund)
-h, --help                        Wyświetla informacje pomocy
-V, --version                     Wyświetla numer wersji
-w <webdriver url>                Włącza monitorowanie przeglądarki
```

Przykłady:

Przykład 1: Wykonanie skryptu ze zmienną zawierającą wartość

Skrypt (`script-file.js`):

```
return value;
```

Użycie:

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

Wyjście: `example_value`

Przykład 2: Wykonanie skryptu z plikiem zawierającym informacje wejściowe

Plik ze skryptem (`script-file.js`):

```
return value;
```

Plik z informacjami wejściowymi (`example.txt`):

```
Przykład informacji wejściowych z pliku
```

Użycie:

```
zabbix_js -s script-file.js -i example.txt
```

Wyjście: `Przykład informacji wejściowych z pliku`

Przykład 3: Wykonanie skryptu z danymi wejściowymi odczytanymi ze stdin

Plik ze skryptem (`script-file.js`):

```
return value;
```

Użycie:

```
zabbix_js -s script-file.js -i -
```

Dane wejściowe (stdin):

```
Przykład danych wejściowych ze stdin
```

Wyjście: `Przykład danych wejściowych ze stdin`

9 Usługa sieciowa

Przegląd

Zabbix web service to proces używany do komunikacji z zewnętrznymi usługami webowymi. Obecnie Zabbix web service służy do generowania i wysyłania zaplanowanych raportów, a w przyszłości planowane jest dodanie dodatkowej funkcjonalności.

Serwer Zabbix łączy się z web service za pośrednictwem HTTP(S). Zabbix web service wymaga zainstalowania Google Chrome na tym samym hoście; w niektórych dystrybucjach usługa może również działać z Chromium (zobacz znane problemy).

Instalacja

Oficjalny pakiet zabbix-web-service jest dostępny w repozytorium Zabbix.

Aby skompilować Zabbix web service ze źródeł, należy podać opcję konfiguracji `--enable-webservice`.

Aby skonfigurować Zabbix web service, zaktualizuj parametry pliku konfiguracyjnego `zabbix_web_service.conf`.

Attention:
Zdecydowanie zaleca się skonfigurowanie szyfrowania między serwerem Zabbix a Zabbix web service z użyciem certyfikatów. Domyślnie dane przesyłane między serwerem Zabbix a Zabbix web service nie są szyfrowane, co może prowadzić do nieautoryzowanego dostępu.

4 Instalacja i pierwsze kroki

Aby skonfigurować Zabbix od podstaw, wykonaj następujące kroki:

1. Upewnij się, że wymagane oprogramowanie jest zainstalowane i uruchomione.
2. Zainstaluj Zabbix.
3. Zainstaluj interfejs webowy Zabbix.
4. Rozpocznij monitorowanie.

Poniżej znajdują się bardziej szczegółowe informacje dotyczące każdego kroku.

Wymagane oprogramowanie Zabbix działa dobrze z następującym oprogramowaniem (często nazywanym stosem LAMP):

- **System operacyjny Linux:** Ubuntu, Debian, CentOS lub inne
- **Serwer WWW:** Apache lub NGINX
- **Serwer bazy danych:** MySQL/MariaDB lub PostgreSQL
- **PHP**

Na większości nowoczesnych systemów Linux PHP może być już zainstalowany. W takim przypadku może być konieczne jedynie skonfigurowanie serwera bazy danych i serwera WWW.

Bardziej szczegółowe informacje, takie jak obsługiwane wersje oprogramowania, znajdziesz w sekcji Wymagane oprogramowanie.

Instalacja Zabbix **Najprostszym** sposobem instalacji Zabbix jest użycie **pakietów dystrybucyjnych** (alternatywnie, z kontenerów). Pakiety instalują drobne zależności oprogramowania.

Dokładne instrukcje instalacji, dostosowane do Twojego systemu operacyjnego, są dostępne na **stronie Zabbix**.

Wybierz swoją platformę i postępuj zgodnie z podanymi instrukcjami. Na początek zaleca się zainstalowanie serwera Zabbix, agenta Zabbix 2 oraz frontend (interfejsu webowego). Później możesz dodać proxy Zabbix i kolejnych agentów.

Częścią procesu instalacji jest utworzenie bazy danych (do przechowywania konfiguracji i danych historycznych) oraz zaimportowanie do niej danych początkowych.

Instalacja web interface Zabbix Wykonaj wszystkie kroki **kreatora instalacji** web interface.

Zostaniesz poproszony o hasło do bazy danych (utworzone w poprzednim kroku), aby frontend mógł komunikować się z bazą danych.

Rozpoczęcie monitorowania Zaloguj się do web interface Zabbix.

Skorzystaj z **krótkich przewodników**, aby rozpocząć monitorowanie popularnych zasobów (Linux, Apache, MySQL lub innych). Te krótkie przewodniki wyjaśniają, z którymi szablonami należy pracować, aby rozpocząć monitorowanie popularnych zasobów.

Możesz zacząć od przewodnika dotyczącego monitorowania serwera Linux.

1 Pobieranie Zabbix

Przegląd

Istnieją cztery sposoby uzyskania Zabbix:

- Zainstaluj go z pakietów dystrybucyjnych
- Pobierz najnowsze archiwum źródłowe i skompiluj je samodzielnie
- Zainstaluj go z kontenerów
- Pobierz wirtualne urządzenie

Aby pobrać najnowsze pakiety dystrybucyjne, wstępnie skompilowane źródła lub wirtualne urządzenie, przejdź do strony pobierania Zabbix, gdzie dostępne są bezpośrednie linki do najnowszych wersji.

Pobieranie kodu źródłowego Zabbix

Istnieje kilka sposobów pobrania kodu źródłowego Zabbix:

- Możesz pobrać stabilne wersje wydane z oficjalnej strony Zabbix
- Możesz pobrać nocne kompilacje z oficjalnej strony dla deweloperów Zabbix
- Możesz pobrać najnowszą wersję rozwojową z systemu repozytorium kodu źródłowego Git:
  - Główna lokalizacja pełnego repozytorium znajduje się pod adresem https://git.zabbix.com/scm/zbx/zabbix.git
  - Główna gałąź oraz obsługiwane wydania są również mirrorowane na Githubie pod adresem https://github.com/zabbix/zabbix

Aby sklonować repozytorium, musi być zainstalowany klient Git. Oficjalny narzędziowy klient Git wiersza poleceń jest w dystrybucjach zwykle nazywany **git**. Aby zainstalować go na przykład w Debianie/Ubuntu, uruchom:

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

Aby pobrać cały kod źródłowy Zabbix, przejdź do katalogu, w którym chcesz umieścić kod, i wykonaj:

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

2 Wymagania

Sprzęt

Pamięć

Zabbix wymaga zarówno pamięci fizycznej, jak i pamięci dyskowej. Wymagana ilość pamięci dyskowej zależy oczywiście od liczby hostów i parametrów, które są monitorowane. Jeśli planujesz przechowywać długą historię monitorowanych parametrów, powinieneś przewidzieć co najmniej kilka gigabajtów, aby mieć wystarczająco dużo miejsca na zapisanie historii w bazie danych. Każdy proces demona Zabbix wymaga kilku połączeń z serwerem bazy danych. Ilość pamięci przydzielonej dla połączenia zależy od konfiguracji silnika bazy danych.

> **Note:**
> Im więcej masz pamięci fizycznej, tym szybciej działa baza danych (a tym samym Zabbix).

CPU

Zabbix, a w szczególności baza danych Zabbix, może wymagać znacznych zasobów CPU w zależności od liczby monitorowanych parametrów i wybranego silnika bazy danych.

Inny sprzęt

Port komunikacji szeregowej oraz szeregowy modem GSM są wymagane do korzystania z obsługi powiadomień SMS w Zabbix. Konwerter USB-do-portu-szeregowego również będzie działać.

Przykłady konfiguracji sprzętowej

Tabela przedstawia przykłady konfiguracji sprzętowej przy założeniu platformy **Linux/BSD/Unix**.

Są to przykłady rozmiaru i konfiguracji sprzętowej na początek. Każda instalacja Zabbix jest unikalna. Przed wdrożeniem instalacji Zabbix do środowiska produkcyjnego należy koniecznie przeprowadzić testy wydajności systemu Zabbix w środowisku testowym lub deweloperskim, aby w pełni zrozumieć swoje wymagania.

Rozmiar instalacji	Monitorowane metryki[1]	Rdzenie CPU/vCPU	Pamięć (GiB)	Baza danych	Amazon EC2[2]
Mała	1 000	2	8	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.large/m6g.large

Rozmiar instalacji	Monitorowane metryki[1]	Rdzenie CPU/vCPU	Pamięć (GiB)	Baza danych	Amazon EC2[2]
Średnia	10 000	4	16	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.xlarge/m6g.xlarge
Duża	100 000	16	64	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.4xlarge/m6g.4xlarge
Bardzo duża	1 000 000	32	96	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.8xlarge/m6g.8xlarge

[1] 1 metryka = 1 pozycja + 1 wyzwalacz + 1 wykres
 [2] Przykład z użyciem instancji Amazon EC2 ogólnego przeznaczenia, z architekturą ARM64 lub x86_64; podczas oceny i testowania instalacji Zabbix, przed wdrożeniem do środowiska produkcyjnego, należy wybrać odpowiedni typ instancji, taki jak zoptymalizowany pod kątem mocy obliczeniowej/pamięci/pamięci masowej.

Note:
Rzeczywista konfiguracja w dużym stopniu zależy od liczby aktywnych pozycji i częstotliwości odświeżania (szczegóły znajdują się w sekcji rozmiar bazy danych na tej stronie). W przypadku dużych instalacji zdecydowanie zaleca się uruchamianie bazy danych na oddzielnym serwerze.

Obsługiwane platformy

Ze względu na wymagania bezpieczeństwa oraz krytyczny charakter serwera monitoringu, UNIX jest jedynym systemem operacyjnym, który może stale zapewniać wymaganą wydajność, odporność na awarie i niezawodność. Zabbix działa na wiodących na rynku wersjach systemów.

Komponenty Zabbix są dostępne i przetestowane dla następujących platform:

Platforma	Serwer	Agent	Agent 2	Uwagi
Linux	x	x	x	
Windows	-	x	x	Agent Zabbix jest obsługiwany we wszystkich desktopowych i serwerowych wersjach systemu od Windows XP (64-bit)/Server 2003. Agent Zabbix 2 jest obsługiwany we wszystkich desktopowych i serwerowych wersjach systemu od Windows 10 (32-bit)/Server 2016, ponieważ jest kompilowany wyłącznie przy użyciu obsługiwanej wersji Go, aby zapobiec krytycznym lukom bezpieczeństwa. Od Go 1.21 minimalne wymagane wersje Windows zostały podniesione, co sprawia, że Windows 10/Server 2016 jest minimalną wersją dla agenta Zabbix 2.
macOS	x	x	-	
IBM AIX	x	x	-	Agent Zabbix nie działa na platformach AIX w wersjach niższych niż 6.1 TL07 / 7.1 TL01.
FreeBSD	x	x	-	
OpenBSD	x	x	-	
Solaris	x	x	-	
NetBSD	x	x	-	
HP-UX	x	x	-	

Note:
Serwer/agent Zabbix może działać na innych systemach operacyjnych typu Unix.

Attention:
Zabbix wyłącza zrzuty core, jeśli został skompilowany z obsługą szyfrowania, i nie uruchamia się, jeśli system nie pozwala na wyłączenie zrzutów core.

Wymagane oprogramowanie

Zabbix jest oparty na nowoczesnych serwerach WWW, wiodących silnikach baz danych oraz języku skryptowym PHP.

Oprogramowanie zewnętrzne firm trzecich

Jeśli określono jako wymagane, wymagane oprogramowanie/biblioteka jest bezwzględnie konieczne. Opcjonalne elementy są potrzebne do obsługi niektórych konkretnych funkcji.

Software	Mandatory status	Supported versions	Comments
MySQL/Percona	One of	8.0.30-9.6.X	Wymagane, jeśli MySQL (lub Percona) jest używany jako baza danych backendowa Zabbix. Wymagany jest silnik InnoDB. Dodano obsługę wersji MySQL: - 9.5.X od Zabbix 7.4.6; - 9.6.X od Zabbix 7.4.9. Zalecamy użycie biblioteki C API (libmysqlclient) do budowania serwer/proxy.
MariaDB		10.5.00-12.2.X	Wymagany jest silnik InnoDB. Zalecana wersja to 11.4. Zalecamy użycie biblioteki MariaDB Connector/C do budowania serwer/proxy. Dodano obsługę wersji MariaDB: - 12.0.X od Zabbix 7.4.3; - 12.2.X od Zabbix 7.4.9. Zobacz także: Possible deadlocks with MariaDB oraz Access to UI elements with MariaDB 10.5.1-10.5.9.
PostgreSQL		13.0-18.X	Wymagane, jeśli PostgreSQL jest używany jako baza danych backendowa Zabbix. W zależności od rozmiaru instalacji może być konieczne zwiększenie właściwości konfiguracyjnej PostgreSQL *work_mem* (domyślna wartość to 4 MB), aby ilość pamięci używanej przez bazę danych dla określonej operacji była wystarczająca, a wykonanie zapytań nie trwało zbyt długo. Dodano obsługę wersji PostgreSQL: - 18.X od Zabbix 7.4.4.
TimescaleDB for PostgreSQL		2.13.0-2.26.X	Wymagane, jeśli TimescaleDB jest używany jako rozszerzenie bazy danych PostgreSQL. Upewnij się, że instalujesz TimescaleDB Community Edition, która obsługuje kompresję. Uwaga: PostgreSQL 15 jest obsługiwany od TimescaleDB 2.10. Więcej informacji na temat zgodności wersji PostgreSQL i TimescaleDB można znaleźć w dokumentacji TimescaleDB. Dodano obsługę wersji TimescaleDB: - 2.20.X, 2.21.X od Zabbix 7.4.1; - 2.22.X od Zabbix 7.4.4; - 2.23.X od Zabbix 7.4.6; - 2.24.X od Zabbix 7.4.7; - 2.25.X od Zabbix 7.4.8; - 2.26.X od Zabbix 7.4.9.
SQLite	Optional	3.3.5-3.53.X	SQLite jest obsługiwany tylko z proxy Zabbix. Wymagany, jeśli SQLite jest używany jako baza danych proxy Zabbix. Obsługa wersji SQLite: - 3.53.X od Zabbix 7.4.9.

Software	Mandatory status	Supported versions	Comments
Elasticsearch		7.X	Elasticsearch jest obsługiwany tylko z serwerami Zabbix, wyłącznie do przechowywania danych historycznych. Obsługa Elasticsearch jest obecnie eksperymentalna. Zobacz także wymagane oprogramowanie dla serwer/proxy.
smartmontools		7.1 or later	Wymagane dla Zabbix agent 2.
who			Wymagane dla pluginu zliczania użytkowników.
dpkg			Wymagane dla pluginu system.sw.packages.
pkgtool			Wymagane dla pluginu system.sw.packages.
rpm			Wymagane dla pluginu system.sw.packages.
pacman			Wymagane dla pluginu system.sw.packages.
q applets			`qlist` i `qsize`, jako część q applets, są wymagane dla pluginu system.sw.packages w Gentoo Linux.

Note:
Chociaż Zabbix może działać z bazami danych dostępnymi w systemach operacyjnych, dla najlepszych rezultatów zalecamy korzystanie z baz danych zainstalowanych z oficjalnych repozytoriów dostawcy bazy danych.

Frontend

Minimalna obsługiwana szerokość ekranu dla frontend Zabbix wynosi 1200px.

Jeśli określono jako obowiązkowe, wymagane oprogramowanie/biblioteka jest bezwzględnie konieczne. Opcjonalne są potrzebne do obsługi niektórych konkretnych funkcji.

Software	Mandatory status	Supported versions	Comments
PHP	Yes	8.0.0 - 8.5.X	Dodano obsługę wersji PHP: - 8.5.X od Zabbix 7.4.9.
Apache	One of	2.4 or later	
Nginx		1.20 or later	
MySQL	One of	See Third-party external surrounding software	
PostgreSQL			
PHP extensions			
mysqli	Yes		Wymagane, jeśli MySQL jest używany jako baza danych backendu Zabbix.
pgsql			Wymagane, jeśli PostgreSQL jest używany jako baza danych backendu Zabbix.
bcmath			php-bcmath (*--enable-bcmath*)
mbstring			php-mbstring (*--enable-mbstring*)
sockets			php-net-socket (*--enable-sockets*); wymagane do obsługi skryptów użytkownika.
gd		2.0.28 or later	php-gd (jeśli dostarczane jako osobny pakiet przez dystrybutora); rozszerzenie PHP GD musi obsługiwać obrazy PNG (*--with-png-dir*), obrazy JPEG (*--with-jpeg-dir*) oraz FreeType 2 (*--with-freetype-dir*). Wersja 2.3.0 lub nowsza może być wymagana, aby uniknąć możliwego nakładania się tekstu na wykresach dla niektórych języków frontend.
libxml		2.6.15 or later	php-xml (jeśli dostarczane jako osobny pakiet przez dystrybutora)
xmlwriter			php-xmlwriter (jeśli dostarczane jako osobny pakiet przez dystrybutora)
xmlreader			php-xmlreader (jeśli dostarczane jako osobny pakiet przez dystrybutora)
ctype			php-ctype (*--enable-ctype*)

Software	Mandatory status	Supported versions	Comments
session			php-session (jeśli dostarczane jako osobny pakiet przez dystrybutora)
ldap	No		php-ldap; wymagane do uwierzytelniania LDAP.
openssl			php-openssl; wymagane do uwierzytelniania SAML.
gettext			php-gettext (*--with-gettext*); wymagane do tłumaczeń.
cURL		7.19.4 or later	php-curl; wymagane dla Duo Universal Prompt MFA oraz uwierzytelniania SMTP.

Biblioteki frontend firm trzecich dostarczane z Zabbix:

Library	Mandatory status	Supplied version	Comments
jQuery JavaScript Library	Yes	3.6.0	Biblioteka JavaScript, która upraszcza tworzenie aplikacji działających w różnych przeglądarkach.
jQuery UI		1.12.1	Zestaw interakcji interfejsu użytkownika, efektów, widżetów i motywów oparty na jQuery.
SAML PHP Toolkit		4.3.1	Zestaw narzędzi PHP, który dodaje obsługę uwierzytelniania SAML 2.0, aby można było zalogować się do Zabbix. Historia dostarczanych wersji: - 4.0.0 od Zabbix 7.4.0; - 4.3.1 od Zabbix 7.4.8 (zalecane).
Duo Universal PHP library		1.1.2	Biblioteka PHP, która dodaje obsługę uwierzytelniania wieloskładnikowego Duo Universal Prompt dla Zabbix. Historia dostarczanych wersji: - 1.0.2 od Zabbix 7.4.0; - 1.1.0 od Zabbix 7.4.7; - 1.1.2 od Zabbix 7.4.10 (zalecane).
Symfony Yaml Component		5.1.0	Dodaje obsługę eksportu i importu elementów konfiguracji Zabbix w formacie YAML.

Note:
Zabbix może również działać na wcześniejszych wersjach Apache, MySQL i PostgreSQL.

Attention:
W przypadku innych czcionek niż domyślna DejaVu może być wymagana funkcja PHP imagerotate. Jeśli jej brakuje, czcionki te mogą być renderowane nieprawidłowo podczas wyświetlania wykresu. Ta funkcja jest dostępna tylko wtedy, gdy PHP jest skompilowany z dołączonym GD, co nie ma miejsca w Debianie i innych dystrybucjach.

Biblioteki firm trzecich używane do pisania i debugowania kodu frontend Zabbix:

Library	Mandatory status	Minimum version	Description
Composer	No	2.4.1	Menedżer pakietów na poziomie aplikacji dla PHP, który zapewnia standardowy format zarządzania zależnościami oprogramowania PHP i wymaganych bibliotek.
PHPUnit		8.5.29	Framework do testów jednostkowych PHP służący do testowania frontend Zabbix.
SASS		3.4.22	Język skryptowy preprocesora, który jest interpretowany i kompilowany do kaskadowych arkuszy stylów (CSS).

Przeglądarka internetowa po stronie klienta

Pliki cookie i JavaScript muszą być włączone.

Obsługiwane są najnowsze stabilne wersje Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari i Opera.

Warning:
Stosowana jest polityka same-origin dla IFrame, co oznacza, że Zabbix nie może być osadzony w ramkach w innej domenie.

 Jednak strony osadzone w ramce Zabbix będą miały dostęp do frontend Zabbix (przez JavaScript), jeśli strona osadzona w ramce i frontend Zabbix znajdują się w tej samej domenie. Strona taka jak `http://secure-zabbix.com/cms/page.html`, jeśli zostanie umieszczona na pulpitach w `http://secure-zabbix.com/zabbix/`, będzie miała pełny dostęp JS do Zabbix.

Server/proxy

Jeśli jest to określone jako obowiązkowe, wymagane oprogramowanie/biblioteka jest bezwzględnie konieczna. Opcjonalne są potrzebne do obsługi niektórych konkretnych funkcji.

Requirement	Mandatory status	Description
libpcre2	Yes	Biblioteka PCRE2 jest wymagana do obsługi Perl Compatible Regular Expression (PCRE). Obsługiwana jest wersja PCRE2 v10.x.
libevent		Wymagana do komunikacji międzyprocesowej. Wersja 2.0.10 lub nowsza.
libevent-pthreads		Wymagana do komunikacji międzyprocesowej.
libpthread		Wymagana do obsługi mutexów i blokad odczytu-zapisu (może być częścią libc).
libresolv		Wymagana do rozwiązywania nazw DNS (może być częścią libc).
libiconv		Wymagana do kodowania tekstu/konwersji formatów (może być częścią libc). Obowiązkowa dla serwer Zabbix w systemie Linux.
libz		Wymagana do obsługi kompresji.
libm		Biblioteka matematyczna. Wymagana tylko przez serwer Zabbix.
libmysqlclient	One of	Wymagana, jeśli używany jest MySQL.
libmariadb		Wymagana, jeśli używany jest MariaDB.
libpq5		Wymagana, jeśli używany jest PostgreSQL; wersja *libpq5* musi być zgodna z wersją używanej bazy danych PostgreSQL lub wyższa.
libsqlite3		Wymagana, jeśli używany jest SQLite. Wymagana tylko przez proxy Zabbix.
libOpenIPMI	No	Wymagana do obsługi IPMI. Wymagana tylko przez serwer Zabbix.
libssh2 or *libssh*		Wymagana dla sprawdzeń SSH. Wersja 1.8.0 lub nowsza (libssh2); 0.9.0 lub nowsza (libssh).
libcurl		Wymagana dla następujących funkcji: - Monitorowanie WWW, monitorowanie VMware oraz pozycje HTTP agent (dla wszystkich: wersja 7.19.1 lub nowsza); - pozycje Zabbix agent web.page.* (wersja 7.19.1 lub nowsza; zobacz także wymagania dla agent); - uwierzytelnianie SMTP (Basic: wersja 7.20.0 lub nowsza; OAuth: wersja 7.33 lub nowsza; zobacz także wymagania dla frontend); - Elasticsearch (wersja 7.28.0 lub nowsza). Zalecana jest wersja 7.28.0 lub nowsza dla wszystkich funkcji. Aby używać zaktualizowanych funkcji cURL dla pozycji `web.page.*`, uruchom ponownie serwer/proxy Zabbix. Do uwierzytelniania SMTP w czasie działania użyj pakietu `libcurl-full`.
libxml2		Wymagana do monitorowania VMware oraz wstępnego przetwarzania XML XPath.
net-snmp		Wymagana do obsługi SNMP. Wersja 5.3.0 lub nowsza. Obsługa silnych protokołów szyfrowania (AES192/AES192C, AES256/AES256C) jest dostępna od biblioteki net-snmp 5.8; w systemach opartych na RHEL 8+ zaleca się używanie net-snmp 5.8.15 lub nowszej.
libunixodbc		Wymagana do monitorowania baz danych.
libgnutls or *libopenssl*		Wymagana przy używaniu szyfrowania. Minimalne wersje: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1
libldap		Wymagana do obsługi LDAP.
fping		Wymagana dla pozycji ICMP ping.
c-ares		Wymagana do asynchronicznego rozwiązywania DNS, jeśli Zabbix jest skonfigurowany z opcją `--with-ares`. W przeciwnym razie użyta zostanie biblioteka *libevent*. Minimalna wersja: 1.16.0

Agent

Wymaganie	Status obow-iązkowości	Opis
libpcre2	Tak	Biblioteka PCRE2 jest wymagana do obsługi Perl Compatible Regular Expression (PCRE). Obsługiwana jest wersja PCRE2 v10.x. Wymagana do monitorowania logów. Wymagana również w systemie Windows.
libpthread		Wymagana do obsługi mutexów oraz blokad odczytu-zapisu (może być częścią libc). Niewymagana w systemie Windows.
libresolv		Wymagana do rozwiązywania nazw DNS (może być częścią libc). Niewymagana w systemie Windows.
libiconv		Wymagana do kodowania/konwersji formatu tekstu do UTF-8 w pozycjach logów, zawartości plików, pozycjach file regex oraz regmatch (może być częścią libc). Niewymagana w systemie Windows.
libgnutls lub *libopenssl*	Nie	Wymagane, jeśli używane jest szyfrowanie. Minimalne wersje: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1 W systemie Microsoft Windows wymagany jest OpenSSL 1.1.1 lub nowszy.
libldap		Wymagana, jeśli używany jest LDAP. Nieobsługiwane w systemie Windows.
libcurl		Wymagana do rozszerzonej obsługi pozycji web.page.* agenta Zabbix. Bez libcurl dostępna jest podstawowa funkcjonalność (np. web.page.get[http://example.com]). Z libcurl agent obsługuje dodatkowe funkcje, takie jak adresy URL HTTP z poświadczeniami (np. http://user:password@example.com) oraz adresy URL HTTPS. Wymagana jest wersja 7.19.1 lub nowsza (zalecana jest wersja 7.28.0 lub nowsza). Aby używać rozszerzonych funkcji cURL, uruchom ponownie agenta Zabbix.
libmodbus		Wymagana tylko, jeśli używane jest monitorowanie Modbus. Wersja 3.0 lub nowsza.

Agent 2

Requirement	Mandatory status	Description
Go	Yes	Wymagane do zbudowania Zabbix agent 2 oraz jego wtyczek ze źródeł. Obsługiwana jest wersja Go 1.24.10 lub nowsza. Instrukcje instalacji znajdują się na go.dev. Biblioteki Go używane przez Zabbix agent 2 i jego wtyczki są wymienione w repozytorium Git Zabbix (biblioteki oznaczone jako `indirect` w repozytorium są zależnościami innych wymaganych bibliotek): - Zabbix agent 2 - Plugin support - PostgreSQL - MongoDB - MSSQL - Ember+ - NVIDIA GPU - Example plugin
libpcre2	Yes	Biblioteka PCRE2 jest wymagana do obsługi Perl Compatible Regular Expression (PCRE). Obsługiwana jest wersja PCRE2 v10.x. Wymagana do monitorowania logów. Wymagana również w systemie Windows.
libopenssl	No	Wymagana podczas używania szyfrowania. Na platformach UNIX wymagana jest biblioteka OpenSSL 1.0.1 lub nowsza. Biblioteka OpenSSL musi mieć włączoną obsługę PSK. LibreSSL nie jest obsługiwany. W systemach Microsoft Windows wymagana jest biblioteka OpenSSL 1.1.1 lub nowsza.

Usługa web

Najnowsza stabilna wersja Google Chrome jest obsługiwana do generowania zaplanowanych raportów przy użyciu usługi web Zabbix.

Wymagana wersja Go do budowania usługi web jest taka sama jak ta używana dla Zabbix agent 2.

Java gateway

Jeśli Zabbix został pozyskany z repozytorium kodu źródłowego lub z archiwum, wszystkie niezbędne zależności są już zawarte w drzewie źródeł.

Jeśli Zabbix został pozyskany z pakietu dostarczanego przez dystrybucję, wszystkie niezbędne zależności są już zapewniane przez system pakietów.

W obu powyższych przypadkach oprogramowanie jest gotowe do użycia i nie są wymagane żadne dodatkowe pliki do pobrania.

Jeśli jednak chcesz dostarczyć własne wersje tych zależności (na przykład podczas przygotowywania pakietu dla jakiejś dystrybucji Linuksa), poniżej znajduje się lista wersji bibliotek, z którymi Java gateway działa poprawnie. Zabbix może również działać z innymi wersjami tych bibliotek.

Poniższa tabela zawiera listę plików JAR, które są obecnie dołączane do Java gateway w oryginalnym kodzie:

Library	Mandatory status	Bundled version	Comments
android-json	Tak	4.3r1	JSON (JavaScript Object Notation) to lekki format wymiany danych. Jest to zgodna z org.json implementacja Androida wyodrębniona z Android SDK.
logback-classic		1.5.16	
logback-core		1.5.16	
slf4j-api		2.0.16	

Java gateway można zbudować przy użyciu zarówno Oracle Java, jak i otwartego OpenJDK (wersja 1.6 lub nowsza). Pakiety dostarczane przez Zabbix są kompilowane przy użyciu OpenJDK. Poniższa tabela zawiera listę pakietów OpenJDK używanych do budowania pakietów Zabbix dla poszczególnych dystrybucji:

Distribution	OpenJDK package
AlmaLinux 9	`java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3)
AlmaLinux 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.332.b09-2; arm64: 1.8.0.382.b05-2)
Amazon Linux 2023	`java-22-amazon-corretto-devel` (amd64, arm64: 22.0.2+9-1)
CentOS Stream 9	`java-11-openjdk-devel` (amd64, arm64: 11.0.18.0.10-3)
CentOS Stream 8	`java-1.8.0-openjdk-devel` (amd64, arm64: 1.8.0.362.b08-3)
CentOS 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Debian 12	`default-jdk-headless` (amd64, arm64: 2:1.17-74)
Debian 11	`default-jdk-headless` (amd64: 2:1.11-72)
OpenSUSE Leap 15	`java-17-openjdk-devel` (amd64: 17.0.5.0-150400.3.9.3; arm64: 17.0.8.0-150400.3.27.1)
Oracle Linux 9	`java-11-openjdk-devel` (amd64: 11.0.19.0.7-4.0.1; arm64: 11.0.20.0.8-2.0.1)
Oracle Linux 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4.0.1); `java-11-openjdk-devel` (arm64: 11.0.20.0.8-3.0.1)
Oracle Linux 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Raspberry Pi OS 12	`default-jdk-headless` (arm64, armhf: 2:1.17-74)
Raspberry Pi OS 11	`default-jdk-headless` (arm64: 2:1.11-72; armhf: 2:1.11-72+b4)
RHEL 9	`java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3)
RHEL 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4; arm64: 1.8.0.382.b05-2)
RHEL 7	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.282.b08-1)
Rocky Linux 9	`java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3)
Rocky Linux 8	`java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4; arm64: 1.8.0.382.b05-2)
SLES 15	`java-17-openjdk-devel` (amd64: 17.0.5.0-150400.3.9.3; arm64: 17.0.8.0-150400.3.27.1)
Ubuntu 24.04	`default-jdk-headless` (amd64, arm64: 2:1.21-75+exp1)
Ubuntu 22.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72build2)
Ubuntu 20.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72)

Domyślne numery portów

Poniższa lista otwartych portów dla poszczególnych komponentów ma zastosowanie do konfiguracji domyślnej:

Komponent Zabbix	Numer portu	Protokół	Typ połączenia
Zabbix agent	10050	TCP	na żądanie
Zabbix agent 2	10050	TCP	na żądanie
Zabbix serwer	10051	TCP	na żądanie
Zabbix proxy	10051	TCP	na żądanie
Zabbix Java gateway	10052	TCP	na żądanie
Usługa internetowa Zabbix	10053	TCP	na żądanie
Zabbix frontend	80	HTTP	na żądanie
	443	HTTPS	na żądanie
Zabbix trapper	10051	TCP	na żądanie

Note:
Numery portów powinny być otwarte w zaporze sieciowej, aby umożliwić komunikację Zabbix. Wychodzące połączenia TCP zwykle nie wymagają jawnych ustawień zapory sieciowej.

Rozmiar bazy danych

Dane konfiguracyjne Zabbix wymagają stałej ilości miejsca na dysku i nie zwiększają się znacząco.

Rozmiar bazy danych Zabbix zależy głównie od następujących zmiennych, które określają ilość przechowywanych danych historycznych:

- Liczba przetwarzanych wartości na sekundę

Jest to średnia liczba nowych wartości, które serwer Zabbix otrzymuje co sekundę. Na przykład, jeśli mamy 3000 pozycji do monitorowania z interwałem odświeżania 60 sekund, liczba wartości na sekundę jest obliczana jako 3000/60 = **50**.

Oznacza to, że co sekundę do bazy danych Zabbix dodawanych jest 50 nowych wartości.

- Ustawienia housekeepera dla historii

Zabbix przechowuje wartości przez stały okres czasu, zwykle kilka tygodni lub miesięcy. Każda nowa wartość wymaga określonej ilości miejsca na dysku na dane i indeks.

Jeśli więc chcemy przechowywać historię przez 30 dni i otrzymujemy 50 wartości na sekundę, całkowita liczba wartości wyniesie około (**30***24*3600)* **50** = 129.600.000, czyli około 130 mln wartości.

W zależności od używanego silnika bazy danych oraz typu otrzymywanych wartości (liczby zmiennoprzecinkowe, liczby całkowite, ciągi znaków, pliki logów itp.), ilość miejsca na dysku potrzebna do przechowywania pojedynczej wartości może wynosić od 40 bajtów do setek bajtów. Zwykle dla numerycznych pozycji jest to około 90 bajtów na wartość[2]. W naszym przypadku oznacza to, że 130 mln wartości będzie wymagać 130 mln * 90 bajtów = **10.9GB** miejsca na dysku.

Note:
Rozmiaru wartości tekstowych/logów dla pozycji nie da się dokładnie przewidzieć, ale można przyjąć około 500 bajtów na wartość.

- Ustawienie housekeepera dla trendów

Zabbix przechowuje 1-godzinny zestaw wartości maks./min./śr./liczba dla każdej pozycji w tabeli **trends**. Dane te są używane do analizy trendów i wykresów długookresowych. Okres jednej godziny nie może być dostosowany.

Baza danych Zabbix, w zależności od typu bazy danych, wymaga około 90 bajtów dla każdego takiego zestawu. Załóżmy, że chcemy przechowywać dane trendów przez 5 lat. Wartości dla 3000 pozycji będą wymagać 3000*24*365* **90** = **2.2GB** rocznie, czyli **11GB** przez 5 lat.

- Ustawienia housekeepera dla zdarzeń

Każde zdarzenie Zabbix wymaga około 250 bajtów miejsca na dysku[1]. Trudno oszacować liczbę zdarzeń generowanych codziennie przez Zabbix. W najgorszym przypadku można założyć, że Zabbix generuje jedno zdarzenie na sekundę.

Dla każdego odzyskanego zdarzenia tworzony jest rekord event_recovery. Zwykle większość zdarzeń zostanie odzyskana, więc można założyć jeden rekord event_recovery na zdarzenie. Oznacza to dodatkowe 80 bajtów na zdarzenie.

Opcjonalnie zdarzenia mogą mieć tagi, przy czym każdy rekord tagu wymaga około 100 bajtów miejsca na dysku[1]. Liczba tagów na zdarzenie (#tags) zależy od konfiguracji. Każde zdarzenie będzie więc wymagać dodatkowo #tags * 100 bajtów miejsca na dysku.

Oznacza to, że jeśli chcemy przechowywać zdarzenia przez 3 lata, będzie to wymagać 3*365*24*3600* (250+80+#tags*100) = **~30GB**+#tags*100B miejsca na dysku[2].

Note:
[1] Więcej w przypadku nazw zdarzeń, tagów i wartości zawierających znaki spoza ASCII.
 [2] Przybliżenia rozmiaru opierają się na MySQL i mogą się różnić w przypadku innych baz danych.

Tabela zawiera wzory, które można wykorzystać do obliczenia ilości miejsca na dysku wymaganego przez system Zabbix:

Parametr	Wzór na wymagane miejsce na dysku (w bajtach)
Konfiguracja Zabbix	Stały rozmiar. Zwykle 10MB lub mniej.
Historia	days*(items/refresh rate)*24*3600*bytes items : liczba pozycji days : liczba dni przechowywania historii refresh rate : średni interwał odświeżania pozycji bytes : liczba bajtów wymagana do przechowywania pojedynczej wartości, zależy od silnika bazy danych, zwykle ~90 bajtów.
Trendy	days*(items/3600)*24*3600*bytes items : liczba pozycji days : liczba dni przechowywania historii bytes : liczba bajtów wymagana do przechowywania pojedynczego trendu, zależy od silnika bazy danych, zwykle ~90 bajtów.
Zdarzenia	days*events*24*3600*bytes events : liczba zdarzeń na sekundę. Jedno (1) zdarzenie na sekundę w najgorszym przypadku. days : liczba dni przechowywania historii bytes : liczba bajtów wymagana do przechowywania pojedynczego zdarzenia, zależy od silnika bazy danych, zwykle ~330 + średnia liczba tagów na zdarzenie * 100 bajtów.

Całkowite wymagane miejsce na dysku można więc obliczyć jako:

Konfiguracja + Historia + Trendy + Zdarzenia

Miejsce na dysku NIE zostanie wykorzystane natychmiast po instalacji Zabbix. Rozmiar bazy danych będzie rósł, a następnie w pewnym momencie przestanie się zwiększać, w zależności od ustawień housekeepera.

Synchronizacja czasu

Bardzo ważne jest, aby na serwerze, na którym działa Zabbix, czas systemowy był precyzyjnie ustawiony. ntpd to najpopularniejszy demon synchronizujący czas hosta z czasem innych maszyn. Zdecydowanie zaleca się utrzymywanie zsynchronizowanego czasu systemowego na wszystkich systemach, na których działają komponenty Zabbix.

Wymagania sieciowe

Poniższa lista otwartych portów dla poszczególnych komponentów ma zastosowanie do konfiguracji domyślnej.

Components	Port
Frontend	http na 80, https na 443
Server	10051 (do użycia z aktywnym proxy/agentami)
Active Proxy	10051
Passive Proxy	10051
Agent2	10050
Trapper	
JavaGateway	10052
WebService	10053

Note:
Numery portów powinny być otwarte w zaporze sieciowej, aby umożliwić zewnętrzną komunikację z Zabbix. Wychodzące połączenia TCP zwykle nie wymagają jawnych ustawień zapory sieciowej.

3 Installation from sources

You can get the very latest version of Zabbix by compiling it from the sources.

A step-by-step tutorial for installing Zabbix from the sources is provided here.

Instalowanie demonów Zabbix

1 Pobierz archiwum źródłowe

Przejdź do strony pobierania Zabbix i pobierz archiwum źródłowe. Po pobraniu rozpakuj źródła, uruchamiając:

```
tar -zxvf zabbix-7.4.0.tar.gz
```

> **Note:**
> Wprowadź w poleceniu prawidłową wersję Zabbix. Musi ona odpowiadać nazwie pobranego archiwum.

2 Utwórz konto użytkownika

Wszystkie procesy demona Zabbix działają w ramach nieuprzywilejowanych użytkowników systemowych.
Jeśli demon Zabbix zostanie uruchomiony z nieuprzywilejowanego konta użytkownika, będzie nadal działał jako ten użytkownik.

W domyślnej konfiguracji, jeśli demon zostanie uruchomiony jako `root`, przełączy się na konto użytkownika `zabbix`, które musi istnieć.
Aby utworzyć systemowego użytkownika i grupę `zabbix`, uruchom poniższe polecenia.

System oparty na RedHat:

```
groupadd --system zabbix
useradd --system -g zabbix -d /usr/lib/zabbix -s /sbin/nologin -c "Zabbix Monitoring System" zabbix
```

System oparty na Debianie:

```
addgroup --system --quiet zabbix
adduser --quiet --system --disabled-login --ingroup zabbix --home /var/lib/zabbix --no-create-home zabbix
```

Nie ma potrzeby tworzenia oddzielnego konta użytkownika dla frontend Zabbix.

Zalecenie dotyczące bezpieczeństwa

Jeśli serwer Zabbix i agent działają na tej samej maszynie, zaleca się uruchamianie ich na **oddzielnych kontach użytkowników**. Uruchamianie obu procesów jako ten sam użytkownik umożliwia agentowi dostęp do pliku konfiguracyjnego serwera, co może ujawnić poufne informacje — takie jak hasło do bazy danych — każdemu użytkownikowi z uprawnieniami Admin w Zabbix.

> **Attention:**
> Uruchamianie Zabbix jako `root`, `bin` lub jakiekolwiek inne konto ze specjalnymi uprawnieniami stanowi zagrożenie bezpieczeństwa.

Katalog domowy (opcjonalnie)

Procesy Zabbix nie wymagają katalogu domowego, więc jego tworzenie zazwyczaj nie jest zalecane.
Jeśli jednak potrzebujesz funkcjonalności wymagającej katalogu domowego (na przykład przechowywania poświadczeń MySQL w $HOME/.my.cnf), możesz go utworzyć za pomocą poniższych poleceń.

W systemach opartych na RedHat uruchom:

```
mkdir -m u=rwx,g=rwx,o= -p /usr/lib/zabbix
chown zabbix:zabbix /usr/lib/zabbix
```

W systemach opartych na Debianie uruchom:

```
mkdir -m u=rwx,g=rwx,o= -p /var/lib/zabbix
chown zabbix:zabbix /var/lib/zabbix
```

3 Utwórz bazę danych Zabbix

Dla demonów Zabbix serwer i proxy, a także dla frontend Zabbix, wymagana jest baza danych. Nie jest ona potrzebna do uruchomienia Zabbix agent.

Dostępne są skrypty SQL do tworzenia schematu bazy danych i wstawiania zestawu danych. Baza danych Zabbix proxy wymaga tylko schematu, natomiast baza danych Zabbix serwer wymaga dodatkowo zestawu danych oprócz schematu.

Po utworzeniu bazy danych Zabbix przejdź do kolejnych kroków kompilacji Zabbix.

4 Konfigurowanie źródeł

Do budowy Zabbix server, Zabbix proxy lub Zabbix agent wymagany jest C99 z rozszerzeniami GNU. Tę wersję można jawnie określić, ustawiając CFLAGS="-std=gnu99":

```
export CFLAGS="-std=gnu99"
```

Note:
Jeśli instalacja odbywa się z repozytorium Git Zabbix, najpierw należy uruchomić:
`./bootstrap.sh`

Podczas konfigurowania źródeł dla Zabbix server lub proxy należy określić typ bazy danych, który ma być używany. Tylko jeden typ bazy danych może być skompilowany jednocześnie z procesem server lub proxy.

Aby wyświetlić wszystkie obsługiwane opcje konfiguracji, w rozpakowanym katalogu źródeł Zabbix uruchom:

```
./configure --help
```

Aby skonfigurować źródła dla Zabbix server i agent, możesz uruchomić na przykład:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Aby skonfigurować źródła dla Zabbix server (z PostgreSQL itp.), możesz uruchomić:

```
./configure --enable-server --with-postgresql --with-net-snmp
```

Aby skonfigurować źródła dla Zabbix proxy (z SQLite itp.), możesz uruchomić:

```
./configure --prefix=/usr --enable-proxy --with-net-snmp --with-sqlite3 --with-ssh2
```

Aby skonfigurować źródła dla Zabbix agent, możesz uruchomić:

```
./configure --enable-agent
```

lub dla Zabbix agent 2:

```
./configure --enable-agent2
```

Note:
Do zbudowania Zabbix agent 2 musi być zainstalowana obsługiwana wersja Go.

Uwagi dotyczące opcji kompilacji:

- `--enable-agent` - kompiluje Zabbix agent, a także narzędzia wiersza poleceń Zabbix get i Zabbix sender.
- `--with-libcurl` - wymagane do monitorowania maszyn wirtualnych, uwierzytelniania SMTP oraz pozycji `web.page.*` Zabbix agent items. Zobacz także: Wymagania (libcurl).
- `--with-libxml2` - wymagane do monitorowania maszyn wirtualnych.
- `--with-libpcre2[=DIR]` - Zabbix zawsze kompiluje się z biblioteką PCRE2; ta opcja pozwala jedynie określić niestandardową ścieżkę instalacji PCRE2.
- `--with-mysql=/path/to/mysql_config` - określa ścieżkę do konkretnej konfiguracji biblioteki klienta MySQL. Przydatne, gdy zainstalowanych jest wiele wersji MySQL lub MariaDB.
- `--enable-static` - statycznie linkuje biblioteki (nieobsługiwane na Solaris). Użyj tej opcji, jeśli planujesz rozpowszechniać skompilowane binaria na systemach bez wymaganych bibliotek. Nie jest zalecana podczas budowania Zabbix server. Aby statycznie zbudować server, wymagana jest statyczna wersja każdej biblioteki zewnętrznej. Skrypt configure nie sprawdza tego automatycznie.
- `--with-stacksize=<value>` - ustawia rozmiar stosu na wątek w kilobajtach (np. `--with-stacksize=512`). Możesz zwiększyć tę wartość, jeśli Zabbix ulega awarii lub zawiesza się z powodu przepełnienia stosu (np. podczas preprocessing w systemach z niskimi domyślnymi limitami rozmiaru stosu wątków).

Attention:
Jeśli `./configure` zakończy się niepowodzeniem z powodu brakujących bibliotek lub innych problemów, sprawdź plik `config.log`, aby uzyskać szczegółowe informacje o błędzie.
Na przykład, jeśli brakuje `libssl`, bezpośredni komunikat o błędzie może być mylący:
```
checking for main in -lmysqlclient... no
configure: error: Not found mysqlclient library
```
W takim przypadku `config.log` ujawnia rzeczywistą przyczynę:
```
/usr/bin/ld: cannot find -lssl
/usr/bin/ld: cannot find -lcrypto
```

Zobacz także:

- Kompilowanie Zabbix z obsługą szyfrowania
- Znane problemy z kompilacją

5 Zbuduj i zainstaluj wszystko

> **Note:**
> Jeśli instalujesz z repozytorium Git Zabbix, najpierw należy uruchomić:
> `$ make dbschema`

`make install`

Ten krok należy uruchomić jako użytkownik z wystarczającymi uprawnieniami (zwykle `root` lub przy użyciu `sudo`).

Uruchomienie `make install` domyślnie zainstaluje pliki binarne demona (zabbix_server, zabbix_agentd, zabbix_proxy) w /usr/local/sbin oraz pliki binarne klienta (zabbix_get, zabbix_sender) w /usr/local/bin.

> **Note:**
> Aby określić inną lokalizację niż /usr/local, użyj klucza --prefix w poprzednim kroku konfiguracji źródeł, na przykład --prefix=/home/zabbix. W takim przypadku pliki binarne demona zostaną zainstalowane w <prefix>/sbin, natomiast narzędzia w <prefix>/bin. Strony podręcznika zostaną zainstalowane w <prefix>/share.

6 Przejrzyj i edytuj pliki konfiguracyjne

- edytuj plik konfiguracyjny agenta Zabbixa **/usr/local/etc/zabbix_agentd.conf**

Musisz skonfigurować ten plik dla każdego hosta z zabbix_agentd zainstalowanym.

Musisz podać w pliku adres IP serwera Zabbixa. Połączenia z innych hostów będą odrzucane.

- edytuj plik konfiguracyjny serwera Zabbixa **/usr/local/etc/zabbix_server.conf**

Musisz podać nazwę bazy danych, użytkownika i hasło (jeśli jest używane).

Pozostałe parametry będą odpowiednie z wartościami domyślnymi, jeśli masz małą instalację (do dziesięciu monitorowanych hostów). Powinieneś jednak zmienić domyślne parametry, jeśli chcesz zmaksymalizować wydajność serwera Zabbixa (lub proxy).

- jeśli zainstalowałeś proxy Zabbixa, edytuj plik konfiguracyjny proxy **/usr/local/etc/zabbix_proxy.conf**

Musisz podać adres IP serwera i nazwę hosta proxy (musi być znana serwerowi), a także nazwę bazy danych, użytkownika i hasło (jeśli jest używane).

> **Note:**
> W przypadku SQLite należy podać pełną ścieżkę do pliku bazy danych; użytkownik DB i hasło nie są wymagane.

7 Uruchamianie demonów

Uruchom zabbix_server po stronie serwera.

`zabbix_server`

> **Note:**
> Upewnij się, że system umożliwia przydzielenie 36 MB (lub nieco więcej) pamięci współdzielonej, w przeciwnym razie serwer może się nie uruchomić i w pliku dziennika serwera pojawi się komunikat "Cannot allocate shared memory for <type of cache>.". Może się to zdarzyć w systemach FreeBSD, Solaris 8.

Uruchom zabbix_agentd na wszystkich monitorowanych maszynach.

`zabbix_agentd`

> **Note:**
> Upewnij się, że system umożliwia przydzielenie 2 MB pamięci współdzielonej, w przeciwnym razie agent może się nie uruchomić i w pliku dziennika agenta pojawi się komunikat "Cannot allocate shared memory for collector.". Może się to zdarzyć w systemie Solaris 8.

Jeśli zainstalowano proxy Zabbix, uruchom zabbix_proxy.

`zabbix_proxy`

Instalowanie interfejsu webowego Zabbix

Kopiowanie plików PHP

frontend Zabbixa jest napisany w PHP, więc do jego uruchomienia potrzebny jest serwer WWW obsługujący PHP. Instalacja polega po prostu na skopiowaniu plików PHP z katalogu ui do katalogu dokumentów HTML serwera WWW.

Typowe lokalizacje katalogów dokumentów HTML dla serwerów WWW Apache obejmują:

- /usr/local/apache2/htdocs (domyślny katalog podczas instalacji Apache ze źródeł)
- /srv/www/htdocs (OpenSUSE, SLES)
- /var/www/html (Debian, Ubuntu, Fedora, RHEL)

Zaleca się użycie podkatalogu zamiast katalogu głównego HTML. Aby utworzyć podkatalog i skopiować do niego pliki frontend Zabbixa, wykonaj następujące polecenia, zastępując <htdocs> rzeczywistym katalogiem:

```
mkdir <htdocs>/zabbix
cd ui
cp -a . <htdocs>/zabbix
```

Jeśli planujesz używać innego języka niż angielski, zobacz Instalacja dodatkowych języków frontend, aby uzyskać instrukcje.

Instalacja frontend

Informacje o kreatorze instalacji frontend Zabbix znajdziesz na stronie Instalacja interfejsu WWW.

Instalowanie ładowalnych wtyczek Zabbix agent 2

Instalacja ładowalnych wtyczek Zabbix agent 2 jest wymagana tylko wtedy, gdy chcesz monitorować cele nieobsługiwane przez wbudowane wtyczki (np. serwery lub klastry MongoDB, PostgreSQL i jego forki itp.). Zobacz pełną listę ładowalnych wtyczek oraz wbudowanych wtyczek.

Attention:
Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

Aby zainstalować ze źródeł, najpierw pobierz i rozpakuj archiwum źródłowe ładowalnej wtyczki.

Aby skompilować wtyczkę, przejdź do rozpakowanego katalogu wtyczki i uruchom `make`:

```
make
```

Note:
Do zbudowania ładowalnych wtyczek Zabbix agent 2 musi być zainstalowana obsługiwana wersja Go.

Plik wykonywalny wtyczki może znajdować się w dowolnym miejscu, o ile jest możliwy do załadowania przez Zabbix agent 2. Określ ścieżkę do binarnego pliku wtyczki w pliku konfiguracyjnym wtyczki, np. w postgresql.conf dla wtyczki PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

Ścieżka do pliku konfiguracyjnego wtyczki musi być określona w parametrze Include pliku konfiguracyjnego Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Przejdź do sekcji konfiguracja, aby uzyskać więcej informacji na temat konfigurowania wtyczek.

Ładowalne wtyczki dostarczane przez Zabbix używają prostych plików makefile z następującymi celami kompilacji:

- `make` - zbuduj wtyczkę
- `make clean` - usuń wszystkie pliki utworzone podczas budowania wtyczki
- `make check` - uruchom testy własne (wymaga rzeczywistego celu monitorowania, np. bazy danych PostgreSQL)
- `make style` - sprawdź styl kodu Go za pomocą `golangci-lint`
- `make format` - sformatuj kod Go za pomocą `go fmt`
- `make dist` - utwórz archiwum źródłowe zawierające wszystkie zależności

Instalowanie Java gateway

Instalacja Java gateway jest wymagana tylko wtedy, gdy chcesz monitorować aplikacje JMX. Java gateway jest lekki i nie wymaga bazy danych.

Aby zainstalować ze źródeł, najpierw pobierz i rozpakuj archiwum źródłowe.

Aby skompilować Java gateway, uruchom skrypt `./configure` z opcją `--enable-java`. Zaleca się określenie opcji `--prefix`, aby wskazać ścieżkę instalacji inną niż domyślna /usr/local, ponieważ instalacja Java gateway utworzy całe drzewo katalogów, a nie tylko pojedynczy plik wykonywalny.

```
./configure --enable-java --prefix=$PREFIX
```

Aby skompilować i spakować Java gateway do pliku JAR, uruchom `make`. Pamiętaj, że na tym etapie będziesz potrzebować plików wykonywalnych `javac` i `jar` w swojej ścieżce.

```
make
```

Teraz masz plik zabbix-java-gateway-$VERSION.jar w src/zabbix_java/bin. Jeśli chcesz uruchamiać Java gateway z katalogu src/zabbix_java w katalogu dystrybucyjnym, możesz przejść do instrukcji konfiguracji i uruchamiania Java gateway. W przeciwnym razie upewnij się, że masz wystarczające uprawnienia i uruchom `make install`.

```
make install
```

Przejdź do konfiguracji, aby uzyskać więcej informacji na temat konfigurowania i uruchamiania Java gateway.

Instalacja usługi webowej Zabbix

Instalacja usługi webowej Zabbix jest wymagana tylko wtedy, gdy chcesz używać raportów zaplanowanych.

Aby zainstalować z kodu źródłowego, najpierw pobierz i rozpakuj archiwum źródłowe.

Aby skompilować usługę webową Zabbix, uruchom skrypt `./configure` z opcją `--enable-webservice`.

> **Note:**
> Do zbudowania usługi webowej Zabbix musi być zainstalowana obsługiwana wersja Go.

Uruchom zabbix_web_service na maszynie, na której usługa webowa jest zainstalowana:

```
zabbix_web_service
```

Przejdź do konfiguracji, aby uzyskać więcej informacji na temat konfigurowania generowania raportów zaplanowanych.

1 Budowanie Zabbix agent na Windows

Przegląd

Ta strona pokazuje, jak zbudować Zabbix agent ze źródeł w systemie Windows 10 (64-bit).

Te instrukcje dotyczą wersji Windows, które obsługują Visual Studio 2022.

Budowanie Zabbix agent wymaga:

- kompilatora C (dołączonego do Build Tools for Visual Studio 2022)
- OpenSSL (dla funkcji szyfrowania w Zabbix)
- PCRE2 (Perl Compatible Regular Expressions; dla funkcji dopasowywania wzorców wyrażeń regularnych w Zabbix)

Możesz zbudować Zabbix agent, korzystając z jednej z następujących metod:

- Korzystanie z vcpkg — zautomatyzowane podejście, które upraszcza zarządzanie zależnościami przy użyciu menedżera pakietów C++.
- Budowanie ręczne — ręczne podejście, które wymaga zainstalowania wszystkich zależności przed skompilowaniem agent.
- W zależności od potrzeb monitorowania mogą być wymagane dodatkowe biblioteki. Więcej informacji można znaleźć w sekcji Wymagania.

> **Attention:**
> Przed rozpoczęciem procesu budowania pamiętaj o następujących kwestiach:

> - Do wykonywania poleceń używaj x64 Native Tools Command Prompt (dołączonego do Build Tools for Visual Studio 2022), uruchomionego przez użytkownika z wystarczającymi uprawnieniami do zapisu w chronionych folderach.
> - Zaleca się utworzenie katalogu roboczego `C:\Zabbix` dla wszystkich plików źródłowych i folderów budowania. Jednak skompilowane komponenty powinny być instalowane w `C:\Program Files\Zabbix\x64`.

Budowanie agent Zabbix za pomocą vcpkg

Ta sekcja zawiera instrukcje dotyczące budowania agent Zabbix za pomocą vcpkg, menedżera pakietów, który upraszcza zarządzanie zależnościami i integrację z projektami C++.

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji upewnij się, że wybierzesz obciążenie *Desktop development with C++*, które zawiera narzędzia wymagane do budowania agent za pomocą vcpkg:

- kompilator C (Microsoft Visual C++)
- narzędzie wiersza poleceń NMake
- menedżer pakietów vcpkg
- x64 Native Tools Command Prompt

2. Zainicjuj vcpkg i zainstaluj zależności wymagane do budowania agent Zabbix (zwróć uwagę, że może to potrwać pewien czas):

```
cd C:\Zabbix
vcpkg new --application
vcpkg add port pcre2
vcpkg add port openssl
vcpkg install --triplet x64-windows-static --x-install-root="C:\Program Files\Zabbix\x64"
```

3. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-7.4.0`.

4. Przejdź do katalogu kompilacji Zabbix (`C:\Zabbix\zabbix-7.4.0\build\win32\project`) i utwórz następujący skrypt `build.bat`; upewnij się, że poprawnie określisz katalogi, w których zainstalowano OpenSSL i PCRE2:

```
:: Set vcpkg installation path:
set vcpkg=C:\Program Files\Zabbix\x64\x64-windows-static

:: Run the build process:
nmake -f Makefile CPU=AMD64 ^
    PCRE2INCDIR="%vcpkg%\include" ^
    PCRE2LIBDIR="%vcpkg%\lib" ^
    TLS=openssl ^
    TLSINCDIR="%vcpkg%\include" ^
    TLSLIBDIR="%vcpkg%\lib" ^
    LIBS="$(LIBS) Crypt32.lib" ^
    all
```

5. Skompiluj agent Zabbix, uruchamiając skrypt:

```
build.bat
```

Po kompilacji pliki binarne komponentów Zabbix będą znajdować się w `C:\Zabbix\zabbix-7.4.0\bin\win64`. Plik konfiguracyjny agent Zabbix znajduje się w `C:\Zabbix\zabbix-7.4.0\conf`.

Aby uruchomić agent, skopiuj `zabbix_agent.exe` oraz jego plik konfiguracyjny do dedykowanego folderu (np. `C:\Zabbix\agent`), a następnie uruchom agent:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent.exe C:\Zabbix\agent\
copy C:\Zabbix\zabbix-7.4.0\conf\zabbix_agent.win.conf C:\Zabbix\agent\

C:\Zabbix\agent\zabbix_agent.exe -c C:\Zabbix\agent\zabbix_agent.win.conf
```

Ręczne budowanie Zabbix agent

Attention:
Ta metoda budowania Zabbix agent jest odpowiednia dla użytkowników, którzy wymagają pełnej kontroli nad środowiskiem budowania lub znajdują się w środowisku z ograniczeniami, gdzie użycie vcpkg nie jest możliwe.

Ta sekcja zawiera instrukcje dotyczące ręcznego budowania Zabbix agent, co obejmuje instalację wymaganych narzędzi do budowania i zależności (Perl, OpenSSL, PCRE2), a następnie kompilację agent.

Instalacja narzędzi kompilacyjnych

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji upewnij się, że wybrano obciążenie *Desktop development with C++*, które zawiera narzędzia wymagane do ręcznego zbudowania agenta:

- kompilator C (Microsoft Visual C++)
- narzędzie wiersza poleceń NMake
- x64 Native Tools Command Prompt

Instalacja OpenSSL

Note:
Aby skompilować Zabbix agent bez obsługi TLS, przejdź do sekcji Installing PCRE2.

1. Pobierz i zainstaluj Strawberry Perl (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\Strawberry`.

2. Zainstaluj moduł Perl `Text::Template`:

```
cpanm Text::Template
```

3. Sprawdź, czy Netwide Assembler (NASM; wymagany do kompilacji OpenSSL) został skompilowany podczas instalacji Strawberry Perl:

```
nasm -v
#### NASM version 2.16.01 compiled on May  3 2024
```

Jeśli NASM nie został skompilowany, zainstaluj go ręcznie. Więcej informacji znajdziesz w dokumentacji NASM.

4. Pobierz archiwum źródłowe OpenSSL i rozpakuj je do `C:\Zabbix\openssl-3.5.0`.

5. Przejdź do rozpakowanego katalogu i skonfiguruj OpenSSL, na przykład:

```
cd C:\Zabbix\openssl-3.5.0
perl Configure VC-WIN64A no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1
```

Attention:
Jeśli podczas kompilacji Zabbix agent w systemie Windows wybierzesz niestandardowy katalog dla OpenSSL (na przykład `C:\zabbix` lub `C:\openssl-64bit`), upewnij się, że odebrano uprawnienia do zapisu w tym katalogu użytkownikom niebędącym administratorami. W przeciwnym razie agent będzie ładować ustawienia SSL ze ścieżki, którą mogą modyfikować użytkownicy bez uprawnień, co stwarza potencjalną lukę bezpieczeństwa.

- Opcja `no-shared` powoduje, że biblioteki statyczne OpenSSL libcrypto.lib i libssl.lib są samowystarczalne, dzięki czemu pliki binarne Zabbix zawierają OpenSSL bez potrzeby używania zewnętrznych bibliotek DLL. Oznacza to, że pliki binarne Zabbix można kopiować na inne komputery z systemem Windows bez bibliotek OpenSSL; jednak po wydaniu nowej wersji OpenSSL zawierającej poprawki błędów konieczna będzie ponowna kompilacja Zabbix agent.
- Bez opcji `no-shared` Zabbix korzysta z bibliotek DLL OpenSSL w czasie działania. Oznacza to, że aktualizacje OpenSSL mogą nie wymagać ponownej kompilacji Zabbix agent; jednak podczas kopiowania go na inne komputery należy również skopiować biblioteki DLL OpenSSL.

Więcej informacji o innych opcjach konfiguracji OpenSSL znajdziesz w dokumentacji OpenSSL.

6. Skompiluj OpenSSL i uruchom testy (pamiętaj, że może to zająć trochę czasu):

Attention:
Uruchamiaj testy bez uprawnień administracyjnych; w przeciwnym razie może to prowadzić do nieoczekiwanych wyników lub zagrożeń bezpieczeństwa. Jeśli niektóre testy zakończą się niepowodzeniem, zapoznaj się z dokumentacją OpenSSL, aby rozwiązać problem.

```
nmake
nmake test
...
All tests successful.
Files=325, Tests=3101, 822 wallclock secs ( 4.81 usr +  0.81 sys =  5.62 CPU)
Result: PASS
```

7. Zainstaluj OpenSSL:

```
nmake install
```

Aby zainstalować tylko komponenty programowe (biblioteki, pliki nagłówkowe, ale bez dokumentacji), możesz użyć `nmake install_sw`.

Instalacja PCRE2

1. Pobierz i zainstaluj CMake (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\CMake`, oraz wybierz opcję *Add CMake to the PATH environment variable*.

2. Pobierz archiwum kodu źródłowego PCRE2 i rozpakuj je do `C:\Zabbix\pcre2-10.45`.

3. Utwórz katalog `build` w rozpakowanym katalogu PCRE2 i przejdź do niego:

```
mkdir C:\Zabbix\pcre2-10.45\build
cd C:\Zabbix\pcre2-10.45\build
```

4. Skonfiguruj PCRE2:

```
cmake -G "NMake Makefiles" -DPCRE_SUPPORT_UNICODE_PROPERTIES=ON -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL
```

Note:
Jeśli wystąpią jakiekolwiek błędy, zaleca się usunięcie pamięci podręcznej CMake przed ponowną próbą przeprowadzenia procesu budowania CMake. Pamięć podręczna (`CMakeCachecache.txt`) znajduje się w katalogu build rozpakowanego katalogu PCRE2.

5. Zbuduj PCRE2 za pomocą NMake:

```
nmake
```

6. Zainstaluj PCRE2:

```
cmake --install .
```

Kompilowanie agent Zabbix

1. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-7.4.0`.

Jeśli musisz wygenerować archiwum źródłowe z surowego repozytorium kodu źródłowego (np. aby zastosować własne poprawki lub zbudować z najnowszego kodu źródłowego), uruchom następujące polecenia na komputerze z **Linux**:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure --enable-agent --enable-ipv6 --prefix=`pwd`
make dist
```

Spowoduje to utworzenie archiwum źródłowego, które można następnie skopiować na komputer z systemem Windows.

2. Przejdź do katalogu kompilacji Zabbix i skompiluj agent Zabbix (lub inne komponenty); upewnij się, że poprawnie określono katalogi, w których są zainstalowane OpenSSL i PCRE2:

```
cd C:\Zabbix\zabbix-7.4.0\build\win32\project

#### Z obsługą TLS:
nmake /K -f Makefile_agent PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program
nmake /K -f Makefile_get PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program F
nmake /K -f Makefile_sender PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Progra

#### Bez obsługi TLS:
nmake /K -f Makefile_agent PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program
nmake /K -f Makefile_get PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Program F
nmake /K -f Makefile_sender PCRE2INCDIR="C:\Program Files\Zabbix\x64\PCRE2\include" PCRE2LIBDIR="C:\Progra
```

Po kompilacji pliki binarne komponentów Zabbix będą znajdować się w `C:\Zabbix\zabbix-7.4.0\bin\win64`. Plik konfiguracyjny agent Zabbix znajduje się w `C:\Zabbix\zabbix-7.4.0\conf`.

Aby uruchomić agent, skopiuj `zabbix_agent.exe` oraz jego plik konfiguracyjny do osobnego folderu (np. `C:\Zabbix\agent`), a następnie uruchom agent:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agentd.exe C:\Zabbix\agent\
copy C:\Zabbix\zabbix-7.4.0\conf\zabbix_agentd.win.conf C:\Zabbix\agent\

C:\Zabbix\agent\zabbix_agentd.exe -c C:\Zabbix\agent\zabbix_agentd.win.conf -f
```

2 Budowanie Zabbix agent 2 w systemie Windows

Przegląd

Ta strona pokazuje, jak zbudować Zabbix agent 2 ze źródeł w systemie Windows 10 (64-bitowym lub 32-bitowym).

Zarówno wersję 32-bitową, jak i 64-bitową można zbudować na platformie 64-bitowej, ale na platformie 32-bitowej można zbudować tylko wersję 32-bitową.

Do zbudowania Zabbix agent 2 wymagane są:

- narzędzia kompilacyjne MinGW
- język programowania Go
- OpenSSL (dla funkcji szyfrowania w Zabbix)

- PCRE2 (Perl Compatible Regular Expressions; dla funkcji dopasowywania wzorców wyrażeń regularnych w Zabbix)

Zabbix agent 2 można zbudować jedną z następujących metod:

- Użycie vcpkg — zautomatyzowane podejście, które upraszcza zarządzanie zależnościami za pomocą menedżera pakietów C++.
- Ręczna kompilacja — ręczne podejście, które wymaga zainstalowania wszystkich zależności przed skompilowaniem agenta.

Attention:
Przed rozpoczęciem procesu kompilacji należy pamiętać o następujących kwestiach:

- Aby wykonywać polecenia, użyj Wiersza polecenia uruchomionego przez użytkownika z wystarczającymi uprawnieniami do zapisu w chronionych folderach. Jednak podczas instalowania OpenSSL i PCRE2 użyj terminala MSYS2.
- Zaleca się utworzenie katalogu roboczego `C:\Zabbix` dla wszystkich plików źródłowych i folderów kompilacji. Jednak skompilowane komponenty powinny być instalowane w `C:\Zabbix\x64` (lub `C:\Zabbix\x86` w przypadku kompilacji 32-bitowych).

Budowanie Zabbix agent 2 z vcpkg

Ta sekcja zawiera instrukcje dotyczące budowania Zabbix agent z użyciem vcpkg, menedżera pakietów, który upraszcza zarządzanie zależnościami i integrację z projektami C++.

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji upewnij się, że wybrano obciążenie *Desktop development with C++*, które zawiera menedżer pakietów vcpkg.

2. Pobierz i zainstaluj Go (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\Go`.

3. Pobierz dystrybucję MinGW, która korzysta z biblioteki uruchomieniowej Microsoft Visual C; na przykład:

- Dla kompilacji 64-bitowych: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Dla kompilacji 32-bitowych: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

Następnie rozpakuj ją do `C:\Zabbix\mingw64` (lub `C:\Zabbix\mingw32` dla kompilacji 32-bitowych).

4. Zainicjuj vcpkg i zainstaluj zależności wymagane do budowania Zabbix agent 2 (zwróć uwagę, że może to potrwać):

```
cd C:\Zabbix

set PATH=%PATH%;"C:\Program Files (x86)\Microsoft Visual Studio\2022\BuildTools\VC\vcpkg"
vcpkg new --application
vcpkg add port pcre2
vcpkg add port libiconv
vcpkg add port openssl

#### For 64-bit builds:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
vcpkg install --triplet x64-mingw-static --x-install-root=x64

#### For 32-bit builds:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
vcpkg install --triplet x86-mingw-static --x-install-root=x86
```

5. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-7.4.0`.

6. Przejdź do katalogu kompilacji Zabbix (`C:\Zabbix\zabbix-7.4.0\build\mingw`) i utwórz następujący skrypt `build.bat`:

- Dla kompilacji 64-bitowych:

```
:: Add MinGW and Go to the system `PATH` variable for the current session:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Set vcpkg installation path:
set vcpkg="C:\Zabbix\x64\x64-mingw-static"

:: Set linker flags for Crypt32 library:
SET CGO_LDFLAGS="-lCrypt32"

:: Run the build process:
```

```
mingw32-make GOFLAGS="-buildvcs=false" ARCH=AMD64 ^
    PCRE2="%vcpkg%" ^
    OPENSSL="%vcpkg%" ^
    all
```

- Dla kompilacji 32-bitowych:

```
:: Add MinGW and Go to the system `PATH` variable for the current session:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Set vcpkg installation path:
set vcpkg="C:\Zabbix\x86\x86-mingw-static"

:: Set linker flags for Crypt32 library:
SET CGO_LDFLAGS="-lCrypt32"

:: Run the build process:
mingw32-make GOFLAGS="-buildvcs=false" ARCH=x86 ^
    PCRE2="%vcpkg%" ^
    OPENSSL="%vcpkg%" ^
    all
```

7. Skompiluj Zabbix agent 2, uruchamiając skrypt:

```
build.bat
```

Po kompilacji plik binarny Zabbix agent 2 będzie znajdować się w `C:\Zabbix\zabbix-7.4.0\bin\win64` (dla kompilacji 64-bitowych) lub `C:\Zabbix\zabbix-7.4.0\bin\win32` (dla kompilacji 32-bitowych). Pliki konfiguracyjne Zabbix agent 2 znajdują się w `C:\Zabbix\zabbix-7.4.0\src\go\conf`.

Aby uruchomić agent, skopiuj `zabbix_agent2.exe` oraz jego pliki konfiguracyjne do dedykowanego folderu (np. `C:\Zabbix\agent2`), a następnie uruchom agent:

```
mkdir C:\Zabbix\agent2

#### For 64-bit builds:
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\

#### For 32-bit builds:
copy C:\Zabbix\zabbix-7.4.0\bin\win32\zabbix_agent2.exe C:\Zabbix\agent2\

copy C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.win.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.win.conf
```

W razie potrzeby kontynuuj, kompilując ładowalne wtyczki dla Zabbix agent 2.

Kompilowanie ładowalnych wtyczek Zabbix agent 2

1. Pobierz źródła wtyczki Zabbix, które odpowiadają wersji Twojego Zabbix agent 2 (np. `zabbix-agent2-plugin-postgresql-7.4.0.tar`) i rozpakuj je do `C:\Zabbix`.

> **Attention:**
> Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

2. Przejdź do rozpakowanego katalogu wtyczki i skompiluj wtyczkę:

```
cd C:\Zabbix\zabbix-agent2-plugin-ember-plus-7.4.0

#### Dla kompilacji 64-bitowej:
mingw32-make ARCH=AMD64

#### Dla kompilacji 32-bitowej:
mingw32-make ARCH=x86
```

Po kompilacji plik binarny wtyczki Zabbix agent 2 oraz jej plik konfiguracyjny będą znajdować się w tym samym katalogu wtyczki.

Plik wykonywalny wtyczki może znajdować się w dowolnym miejscu, o ile może być załadowany przez Zabbix agent 2. Określ ścieżkę do pliku binarnego wtyczki w pliku konfiguracyjnym wtyczki, np. w pliku postgresql.conf dla wtyczki PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

Ścieżka do pliku konfiguracyjnego wtyczki musi być podana w parametrze Include pliku konfiguracyjnego Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Przejdź do sekcji konfiguracja, aby uzyskać więcej informacji na temat konfigurowania wtyczek.

Ręczne budowanie Zabbix agent 2

Attention:
Ta metoda budowania Zabbix agent 2 jest odpowiednia dla użytkowników, którzy wymagają pełnej kontroli nad środowiskiem kompilacji lub znajdują się w środowisku o ograniczeniach, w którym użycie vcpkg nie jest możliwe.

Ta sekcja zawiera instrukcje dotyczące ręcznego budowania Zabbix agent 2, obejmujące instalację wymaganych narzędzi i zależności kompilacyjnych, a następnie skompilowanie agenta.

Konfigurowanie narzędzi do budowania

1. Pobierz i zainstaluj MSYS2 (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny zostanie wskazany `C:\Zabbix\msys64`.

2. Pobierz i zainstaluj Go (dostępny jako instalator MSI; zobacz aktualnie obsługiwane wersje Go). Podczas instalacji upewnij się, że jako folder instalacyjny zostanie wskazany `C:\Zabbix\Go`.

3. Pobierz dystrybucję MinGW, która korzysta z biblioteki uruchomieniowej Microsoft Visual C; na przykład:

- Dla kompilacji 64-bitowych: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Dla kompilacji 32-bitowych: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

Następnie rozpakuj ją do `C:\Zabbix\mingw64` (lub `C:\Zabbix\mingw32` w przypadku kompilacji 32-bitowych).

Instalowanie OpenSSL

Note:
Aby skompilować agent Zabbix bez obsługi TLS, przejdź do sekcji Instalowanie PCRE2.

1. Otwórz terminal MSYS2 MSYS z uprawnieniami administratora i uruchom następujące polecenia:

```
pacman -S perl-Locale-Maketext-Simple
pacman -S nasm
pacman -S make
pacman -S cmake
```

2. Pobierz archiwum źródłowe OpenSSL i rozpakuj je do `C:\Zabbix\openssl-3.5.0`.

3. Przejdź do rozpakowanego katalogu OpenSSL i utwórz następujący skrypt `build.sh`:

- Dla kompilacji 64-bitowych:

```
####!/usr/bin/env bash

export PATH="/c/Zabbix/mingw64/bin:$PATH"
export d="/c/Zabbix/x64/OpenSSL-Win64-350-static"

perl Configure mingw64 no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1_2

make
make install
```

- Dla kompilacji 32-bitowych:

```
####!/usr/bin/env bash

export PATH="/c/Zabbix/mingw32/bin:$PATH"
export d="/c/Zabbix/x86/OpenSSL-Win64-350-static"

perl Configure mingw no-shared no-capieng no-winstore no-srp no-gost no-dgram no-dtls1-method no-dtls1_2-m
```

```
make
make install
```

Attention:
Upewnij się, że odebrano uprawnienia do zapisu użytkownikom innym niż administratorzy w katalogu `C:\Zabbix\x64\OpenSSL-Win64-350-static`. W przeciwnym razie agent będzie ładować ustawienia SSL ze ścieżki, którą mogą modyfikować użytkownicy bez uprawnień, co może prowadzić do potencjalnej luki bezpieczeństwa.

- Opcja `no-shared` sprawia, że biblioteki statyczne OpenSSL libcrypto.lib i libssl.lib są samodzielne, dzięki czemu pliki binarne Zabbix zawierają OpenSSL bez potrzeby używania zewnętrznych plików DLL. Oznacza to, że pliki binarne Zabbix można skopiować na inne komputery z systemem Windows bez bibliotek OpenSSL; jednak po wydaniu nowej wersji poprawkowej OpenSSL agent Zabbix będzie trzeba ponownie skompilować.
- Bez opcji `no-shared` Zabbix w czasie działania korzysta z bibliotek DLL OpenSSL. Oznacza to, że aktualizacje OpenSSL mogą nie wymagać ponownej kompilacji agenta Zabbix; jednak podczas kopiowania go na inne komputery należy również skopiować biblioteki DLL OpenSSL.

Więcej informacji o innych opcjach konfiguracji OpenSSL można znaleźć w dokumentacji OpenSSL.

4. Skonfiguruj i zainstaluj OpenSSL, wykonując skrypt (zauważ, że może to potrwać pewien czas):

```
cd /c/Zabbix/openssl-3.5.0
./build.sh
```

Instalowanie PCRE2

1. Pobierz archiwum źródłowe PCRE2 i rozpakuj je do `C:\Zabbix\pcre2-10.45`.

2. Otwórz terminal MSYS2 MSYS z uprawnieniami administratora. Następnie utwórz katalog `build` w rozpakowanym katalogu PCRE2 i przejdź do niego:

```
mkdir /c/Zabbix/pcre2-10.45/build
cd /c/Zabbix/pcre2-10.45/build
```

3. Skonfiguruj PCRE2:

```
#### Dla kompilacji 64-bitowej:
export PATH="/c/Zabbix/mingw64/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-O2 -g" -DCMAKE_INSTALL_PREFIX="/c/Zabbix/x64/PCRE2" ..

#### Dla kompilacji 32-bitowej:
export PATH="/c/Zabbix/mingw32/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-m32 -O2 -g" -DCMAKE_EXE_LINKER_FLAGS="-Wl,-mi386pe" -DCMAKE
```

Note:
Jeśli wystąpią jakiekolwiek błędy, zaleca się usunięcie pamięci podręcznej CMake przed ponowną próbą powtórzenia procesu kompilacji CMake. Pamięć podręczna (`CMakeCachecache.txt`) może znajdować się w katalogu `build` rozpakowanego katalogu PCRE2.

4. Zainstaluj PCRE2:

```
make install
```

Kompilacja Zabbix agent 2

1. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-7.4.0`.

Jeśli musisz wygenerować archiwum źródłowe z surowego repozytorium źródeł (np. aby zastosować własne poprawki lub zbudować z najnowszego kodu źródłowego), uruchom następujące polecenia na komputerze z systemem **Linux** i zainstalowanym Go (wymaganym do konfiguracji Zabbix agent 2):

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure --enable-agent2 --enable-ipv6 --prefix=`pwd`
make dist
```

Spowoduje to utworzenie archiwum źródłowego, które następnie można skopiować na komputer z systemem Windows.

2. Otwórz Wiersz polecenia z uprawnieniami administratora. Następnie przejdź do katalogu kompilacji Zabbix i skompiluj Zabbix agent; upewnij się, że poprawnie wskazujesz katalogi, w których zainstalowano OpenSSL i PCRE2:

- Dla kompilacji 64-bitowej:

```
cd C:\Zabbix\zabbix-7.4.0\build\mingw
set PATH=C:\Zabbix\mingw64\bin;%PATH%
mklink /D C:\Zabbix\x64\OpenSSL-Win64-350-static\lib C:\Zabbix\x64\OpenSSL-Win64-350-static\lib64

#### Z obsługą TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2" OPENSSL="C:\Zabbix\x64\OpenSSL-Win64-350-static"

#### Bez obsługi TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2"
```

- Dla kompilacji 32-bitowej:

```
cd C:\Zabbix\zabbix-7.4.0\build\mingw
set PATH=C:\Zabbix\mingw32\bin;%PATH%

#### Z obsługą TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2" OPENSSL="C:\Zabbix\x86\OpenSSL-Win64-350-static"

#### Bez obsługi TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2"
```

Po kompilacji plik binarny Zabbix agent 2 będzie znajdować się w `C:\Zabbix\zabbix-7.4.0\bin\win64` (lub w `C:\Zabbix\zabbix-7.4.` w przypadku kompilacji 32-bitowej). Pliki konfiguracyjne Zabbix agent 2 znajdują się w `C:\Zabbix\zabbix-7.4.0\src\go\conf`.

Aby uruchomić agent, skopiuj plik binarny `zabbix_agent2.exe` oraz jego pliki konfiguracyjne do dedykowanego folderu (np. `C:\Zabbix\agent2`), a następnie uruchom agenta:

```
mkdir C:\Zabbix\agent2
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\
copy C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.win.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.win.conf
```

W razie potrzeby przejdź do kompilacji ładowalnych wtyczek Zabbix agent 2.

3 Budowanie Zabbix agent na macOS

Przegląd

Ta sekcja pokazuje, jak zbudować binaria agenta Zabbix dla systemu macOS ze źródeł, z obsługą TLS lub bez niej.

Wymagania wstępne

Będziesz potrzebować narzędzi deweloperskich wiersza poleceń (Xcode nie jest wymagany), Automake, pkg-config oraz PCRE (v8.x) lub PCRE2 (v10.x). Jeśli chcesz zbudować pliki binarne agent z obsługą TLS, będziesz także potrzebować OpenSSL lub GnuTLS.

Aby zainstalować Automake i pkg-config, będziesz potrzebować menedżera pakietów Homebrew ze strony https://brew.sh/. Aby go zainstalować, otwórz terminal i uruchom następujące polecenie:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Następnie zainstaluj Automake i pkg-config:

```
brew install automake
brew install pkg-config
```

Przygotowanie bibliotek PCRE, OpenSSL i GnuTLS zależy od sposobu, w jaki będą one linkowane z agent.

Jeśli zamierzasz uruchamiać pliki binarne agent na komputerze z macOS, który ma już te biblioteki, możesz użyć wstępnie skompilowanych bibliotek dostarczanych przez Homebrew. Zwykle są to komputery z macOS, które używają Homebrew do budowania plików binarnych Zabbix agent lub do innych celów.

Jeśli pliki binarne agent będą używane na komputerach z macOS, które nie mają współdzielonej wersji bibliotek, powinieneś skompilować biblioteki statyczne ze źródeł i połączyć z nimi Zabbix agent.

Budowanie plików binarnych agent z bibliotekami współdzielonymi

Zainstaluj PCRE2 (w razie potrzeby zastąp *pcre2* przez *pcre* w poniższych poleceniach):

```
brew install pcre2
```

Podczas budowania z TLS zainstaluj OpenSSL i/lub GnuTLS:

```
brew install openssl
brew install gnutls
```

Pobierz kod źródłowy Zabbix:

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

Zbuduj agent bez TLS:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6
make
make install
```

Zbuduj agent z OpenSSL:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-openssl=/usr/local/opt/
make
make install
```

Zbuduj agent z GnuTLS:

```
cd zabbix-source/
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-gnutls=/usr/local/opt/g
make
make install
```

Budowanie plików binarnych agent z bibliotekami statycznymi bez TLS

Załóżmy, że statyczne biblioteki PCRE zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
```

Pobierz i zbuduj PCRE z obsługą właściwości 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
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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
```

Budowanie plików binarnych agent z bibliotekami statycznymi przy użyciu OpenSSL

Podczas budowania OpenSSL zaleca się uruchomienie `make test` po pomyślnym zakończeniu kompilacji. Nawet jeśli kompilacja zakończyła się sukcesem, testy czasami kończą się niepowodzeniem. W takim przypadku należy zbadać problemy i je rozwiązać przed kontynuowaniem.

Załóżmy, że statyczne biblioteki PCRE i OpenSSL zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39 oraz OpenSSL 1.1.1a.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
OPENSSL_PREFIX="$HOME/static-libs/openssl-1.1.1a"
```

Zbudujmy biblioteki statyczne w `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Pobierz i zbuduj PCRE z obsługą właściwości 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 ..
```

Pobierz i zbuduj 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 no-
make
make test
make install_sw
cd ..
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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
```

Budowanie plików binarnych agent z bibliotekami statycznymi przy użyciu GnuTLS

GnuTLS zależy od backendu kryptograficznego Nettle oraz biblioteki arytmetycznej GMP. Zamiast używać pełnej biblioteki GMP, ten przewodnik wykorzystuje mini-gmp, które jest dołączone do Nettle.

Podczas budowania GnuTLS i Nettle zaleca się uruchomienie `make check` po pomyślnym zakończeniu kompilacji. Nawet jeśli kompilacja zakończyła się powodzeniem, testy czasami kończą się niepowodzeniem. W takim przypadku należy zbadać problemy i rozwiązać je przed kontynuowaniem.

Załóżmy, że statyczne biblioteki PCRE, Nettle i GnuTLS zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39, Nettle 3.4.1 oraz 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"
```

Zbudujmy biblioteki statyczne w `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Pobierz i zbuduj 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 --disable-a
make
make check
make install
cd ..
```

Pobierz i zbuduj 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 --dis
make
make check
make install
cd ..
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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 Installation from packages

Przegląd

Oficjalne pakiety Zabbix 7.4 są dostępne na stronie Zabbix, gdzie można wybrać system operacyjny i komponent Zabbix, aby wygenerować wymagane instrukcje instalacji oraz polecenia. Upewnij się, że zapoznasz się z poniższymi uwagami dotyczącymi instalacji pakietów oraz stroną Instalacja i konfiguracja.

Pakiety są dostępne dla następujących dystrybucji Linux:

- Red Hat Enterprise Linux i jego pochodnych: AlmaLinux, Amazon Linux 2023, CentOS Stream, CentOS 7, Oracle Linux, Rocky Linux
- Debian, Ubuntu, Raspberry Pi OS, Raspbian
- SUSE Linux Enterprise Server, openSUSE Leap

Attention:
Niektóre dystrybucje systemu operacyjnego (w szczególności dystrybucje oparte na Debianie) udostępniają własne pakiety Zabbix. Pakiety te **nie** są obsługiwane przez Zabbix i mogą być nieaktualne lub nie zawierać najnowszych funkcji oraz poprawek błędów. Zaleca się korzystanie wyłącznie z oficjalnych pakietów z Oficjalnego repozytorium Zabbix. Jeśli wcześniej zainstalowano Zabbix z repozytorium systemu operacyjnego, zobacz kroki dotyczące aktualizacji pakietów Zabbix z repozytoriów systemu operacyjnego.

Pakiety obsługują bazę danych MySQL/PostgreSQL oraz frontend WWW Apache/Nginx. Należy pamiętać, że serwer Zabbix i proxy nie mogą współdzielić tej samej bazy danych; użyj różnych nazw baz danych, jeśli oba są zainstalowane na tym samym host.

W razie potrzeby osobne pakiety dla Zabbix agent/agent 2, Zabbix get i Zabbix sender są dostępne w Oficjalnym repozytorium Zabbix.

Zabbix udostępnia również wstępnie skompilowane binaria Zabbix agent dla systemów innych niż Linux; zobacz:

- Instalacja agent dla Windows z MSI
- Instalacja agent dla macOS z PKG
- Starsze binaria (dla starszych lub mniej popularnych systemów, takich jak HP-UX, NetBSD, Tru64 oraz starsze wersje SLES)

Uwagi dotyczące instalacji pakietów

Poniższe uwagi dotyczą wszystkich systemów:

- Jeśli używasz PostgreSQL, ustawienie `DBHost=localhost` (lub adresu IP) w konfiguracji Zabbix serwer/proxy powoduje, że PostgreSQL używa gniazda sieciowego zamiast lokalnego gniazda UNIX; zobacz Konfiguracja SELinux, aby uzyskać powiązane instrukcje konfiguracji.
- Jeśli używasz TimescaleDB, zobacz dodatkowe informacje dotyczące konfiguracji TimescaleDB.
- Jeśli instalujesz Zabbix Java gateway (do monitorowania aplikacji JMX), zobacz dodatkowe instrukcje konfiguracji dla systemów opartych na RHEL oraz opartych na Debianie.
- Aby uruchamiać agent Zabbix jako root, zobacz Uruchamianie agenta jako root.

Poniższe uwagi dotyczą RHEL i jego pochodnych:

- Jeśli włączyłeś repozytorium EPEL dla EL9, które również dostarcza pakiety Zabbix, należy je wykluczyć z rozwiązywania zależności pakietów przed instalacją oficjalnych pakietów Zabbix; zobacz Przypadkowa instalacja pakietów Zabbix z EPEL.
- Aby instalować pakiety Zabbix w środowiskach Red Hat UBI, zobacz Pakiety Zabbix dla RHEL w środowiskach Red Hat UBI.
- Do używania pozycji ICMP ping pakiety dla `fping` są również dostępne w Oficjalnym repozytorium Zabbix.

Konfiguracja SELinux

Zabbix używa komunikacji międzyprocesowej opartej na gniazdach. W systemach, w których włączony jest Security-Enhanced Linux (SELinux), może być konieczne dodanie reguł SELinux, aby umożliwić Zabbix tworzenie/używanie gniazd domeny UNIX w katalogu SocketDir. Pliki gniazd są używane przez serwer Zabbix (alerter, preprocessing, IPMI) oraz proxy Zabbix (IPMI) i są obecne, gdy proces jest uruchomiony.

Przy włączonym SELinux w trybie enforcing wykonaj następujące polecenia, aby umożliwić komunikację między frontendem Zabbix a serwerem:

Dla RHEL 7 (i nowszych), AlmaLinux, CentOS Stream, Oracle Linux, Rocky Linux 8 (i nowszych):

```
setsebool -P httpd_can_connect_zabbix on
```

Jeśli baza danych jest dostępna przez sieć (w tym `localhost` dla PostgreSQL), zezwól również frontendowi Zabbix na połączenie z bazą danych:

```
setsebool -P httpd_can_network_connect_db on
```

Dla RHEL wcześniejszych niż 7:

```
setsebool -P httpd_can_network_connect on
setsebool -P zabbix_can_network on
```

Po zastosowaniu ustawień SELinux uruchom ponownie Apache:

```
systemctl restart httpd
```

Opcjonalnie możesz zainstalować gotowy pakiet `zabbix-selinux-policy` z Oficjalnego repozytorium Zabbix. Pakiet ten jest dostarczany dla wszystkich obsługiwanych wersji systemów operacyjnych, aby uprościć wdrożenie Zabbix i zapobiec wyłączaniu SELinux przez użytkowników z powodu złożoności konfiguracji.

> **Attention:**
> Dla maksymalnego bezpieczeństwa zaleca się ustawienie niestandardowych ustawień SELinux.

Pakiet `zabbix-selinux-policy` zawiera podstawową politykę SELinux, która umożliwia Zabbix tworzenie i używanie gniazd oraz zezwala HTTPd na połączenie z PostgreSQL (używanym przez frontend).

Źródłowy plik `zabbix_policy.te` zawiera następujące reguły:

```
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;
```

Pakiety debuginfo

Pakiety Debuginfo zawierają symbole debugowania dla binariów Zabbix. Nie są wymagane do standardowej instalacji ani działania, ale są przydatne podczas zaawansowanego rozwiązywania problemów.

Należy pamiętać, że pakiety Debuginfo są dostępne dla następujących wersji: Red Hat Enterprise Linux 7 i wcześniejszych (w tym pochodnych) oraz SUSE Linux Enterprise Server 15 i wcześniejszych (w tym pochodnych).

Aby włączyć repozytorium zabbix-debuginfo:

- W RHEL 7 edytuj /etc/yum.repos.d/zabbix.repo i ustaw enabled=1 dla sekcji zabbix-debuginfo:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo - $basearch
baseurl=http://repo.zabbix.com/zabbix/7.4/stable/rhel/7/$basearch/debuginfo/
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
gpgcheck=1
```

- W SUSE edytuj /etc/zypp/repos.d/zabbix.repo i ustaw enabled=1 dla sekcji zabbix-debuginfo:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/7.4/stable/sles/15/$basearch/debuginfo/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/7.4/stable/sles/15/$basearch/debuginfo/repodata/repomd.xml.key
enabled=0
update=1
```

Po włączeniu zainstaluj pakiety:

- W RHEL zainstaluj pojedynczy pakiet z informacjami debugowania dla wszystkich komponentów Zabbix:

```
dnf install zabbix-debuginfo
```

- W SUSE zainstaluj pakiety debuginfo specyficzne dla komponentów:

```
zypper install zabbix-<component>-debuginfo
```

1 Instalacja agenta Windows z pakietu MSI

Przegląd

Zabbix agent można zainstalować w systemie Windows przy użyciu 32-bitowych lub 64-bitowych pakietów instalatora MSI, dostępnych do pobrania.

Minimalne wymagania systemu operacyjnego dla instalacji MSI to:

- **Dla Zabbix agent:** Windows XP (64-bit) lub Windows Server 2003
- **Dla Zabbix agent 2:** Windows 10 (32-bit) lub Windows Server 2016

Pakietów 32-bitowych nie można instalować w systemach 64-bitowych.

Pakiety zawierają:

- obsługę TLS (konfiguracja TLS jest opcjonalna)
- narzędzia Zabbix get i Zabbix sender (mogą być instalowane razem z Zabbix agent/agent 2 lub osobno)

> **Attention:**
> Pakiety Zabbix agent 2 nie zawierają ładowalnych wtyczek (MongoDB, PostgreSQL, MSSQL), które należy pobrać i zainstalować osobno.

Instalację można przeprowadzić przy użyciu kreatora instalacji lub wiersza poleceń.

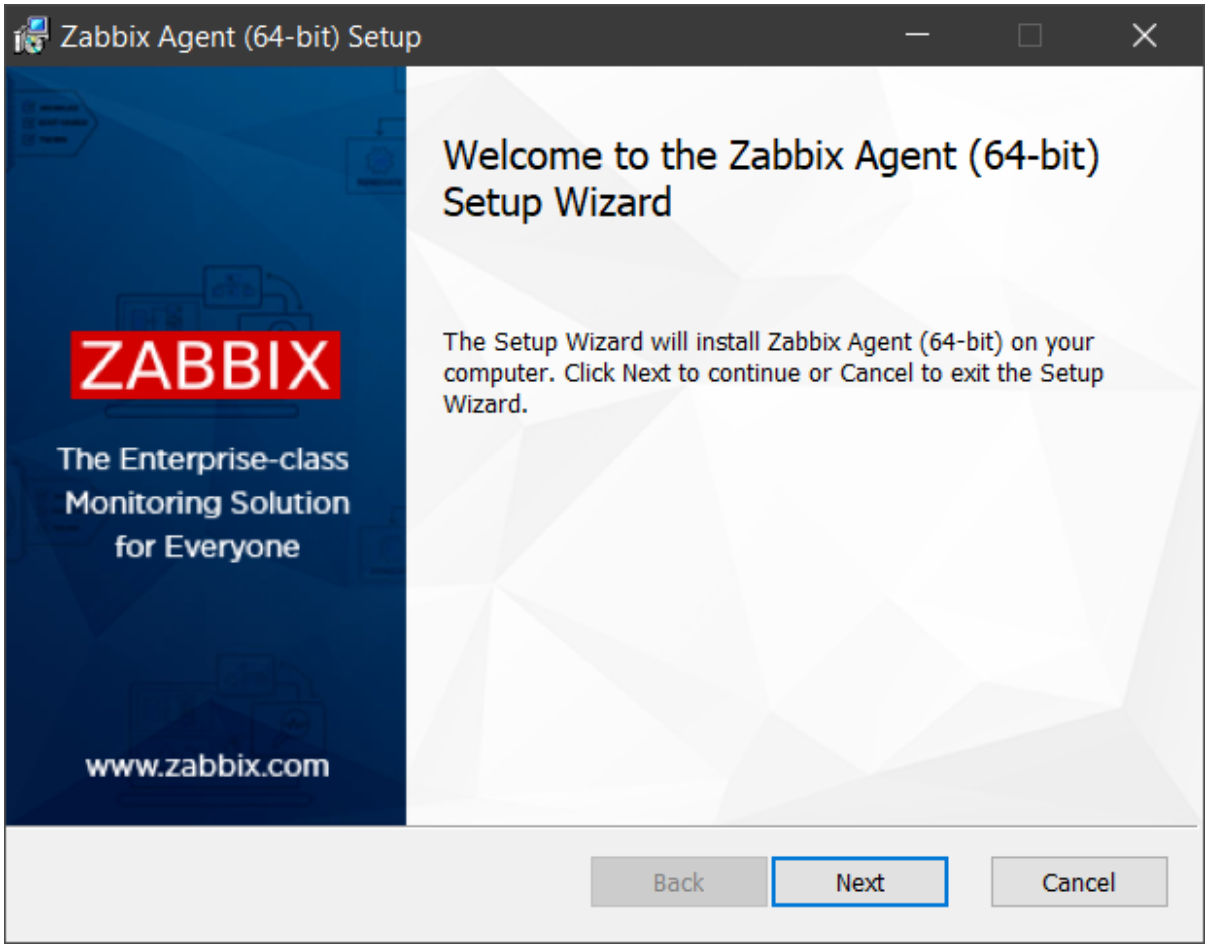
Chociaż instalacja przy użyciu pakietów MSI jest w pełni obsługiwana, zaleca się zainstalowanie co najmniej Microsoft .NET Framework 2, aby zapewnić prawidłową obsługę błędów.

Attention:
Zaleca się użycie domyślnej lokalizacji instalacji oferowanej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może obniżyć bezpieczeństwo instalacji.

Instalacja z kreatora instalacji

Poniższe kroki instalacji dotyczą zarówno Zabbix agent, jak i Zabbix agent 2.

1. Kliknij dwukrotnie pobrany plik MSI, aby rozpocząć instalację:

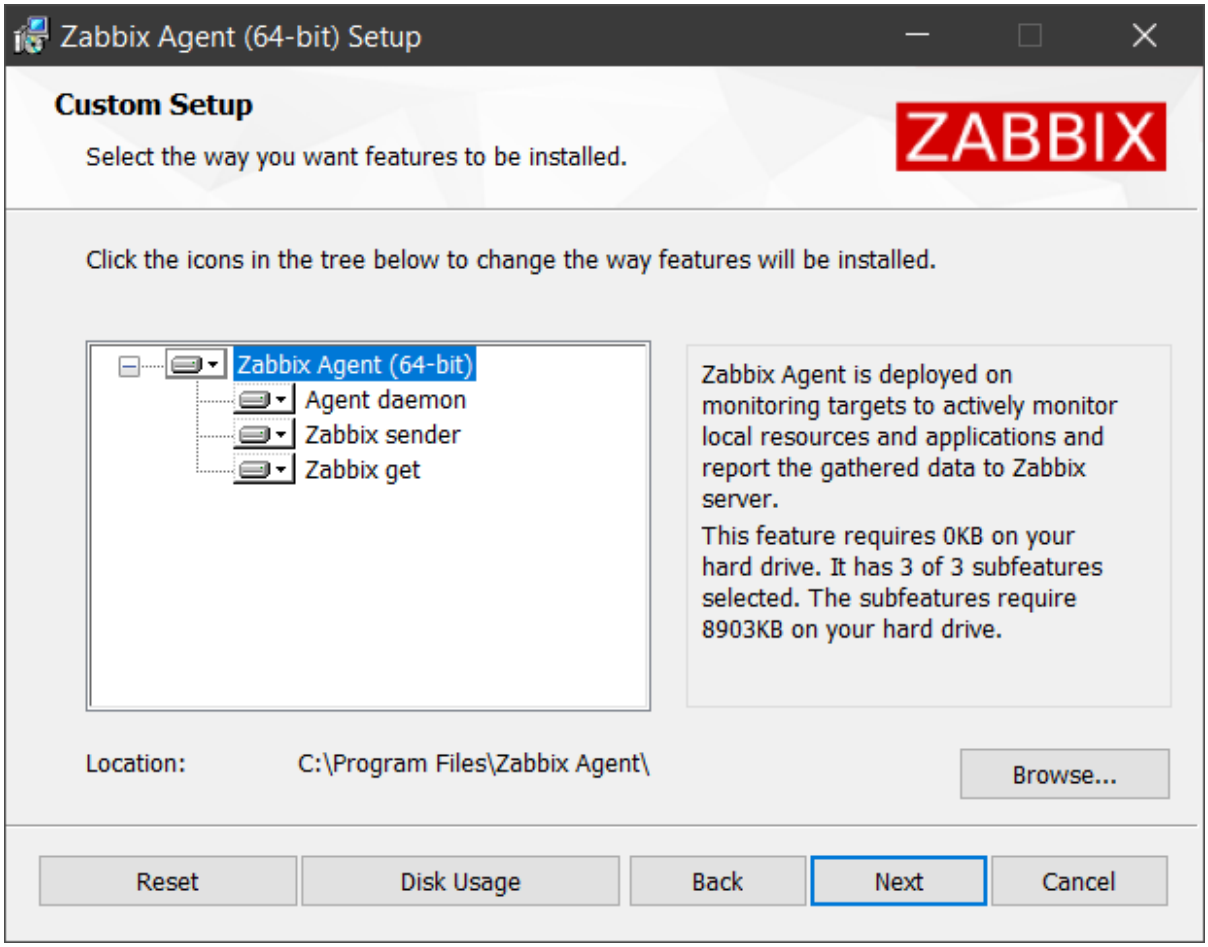


2. Zaakceptuj Umowę licencyjną użytkownika końcowego:



3. Wybierz komponenty Zabbix (Agent daemon, Zabbix sender, Zabbix get) do zainstalowania:

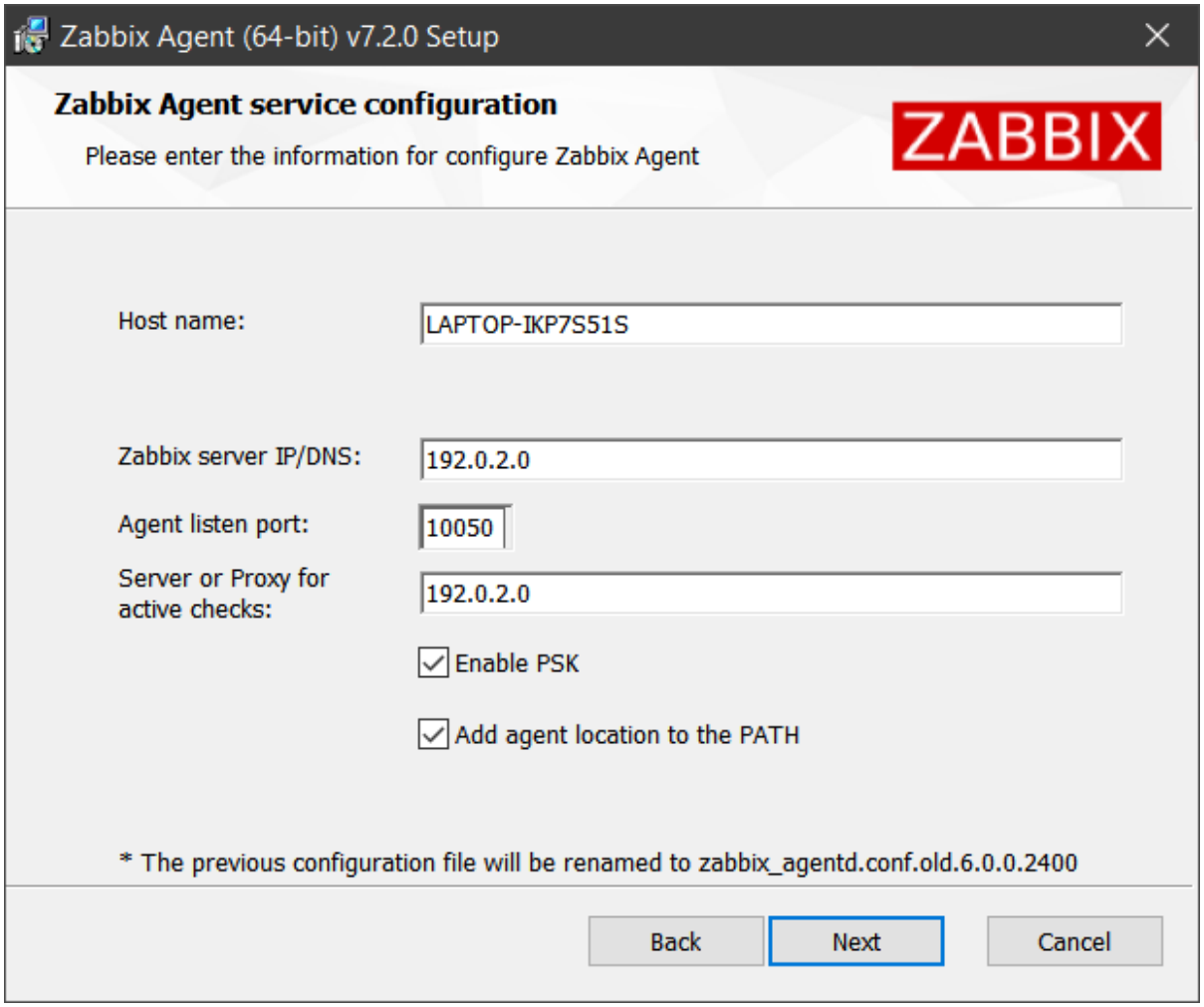
Attention:
Zaleca się użycie domyślnej lokalizacji instalacji udostępnionej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może zagrozić bezpieczeństwu instalacji.



4. Skonfiguruj następujące parametry. Ich wartości zostaną zapisane w pliku konfiguracyjnym Zabbix agent:

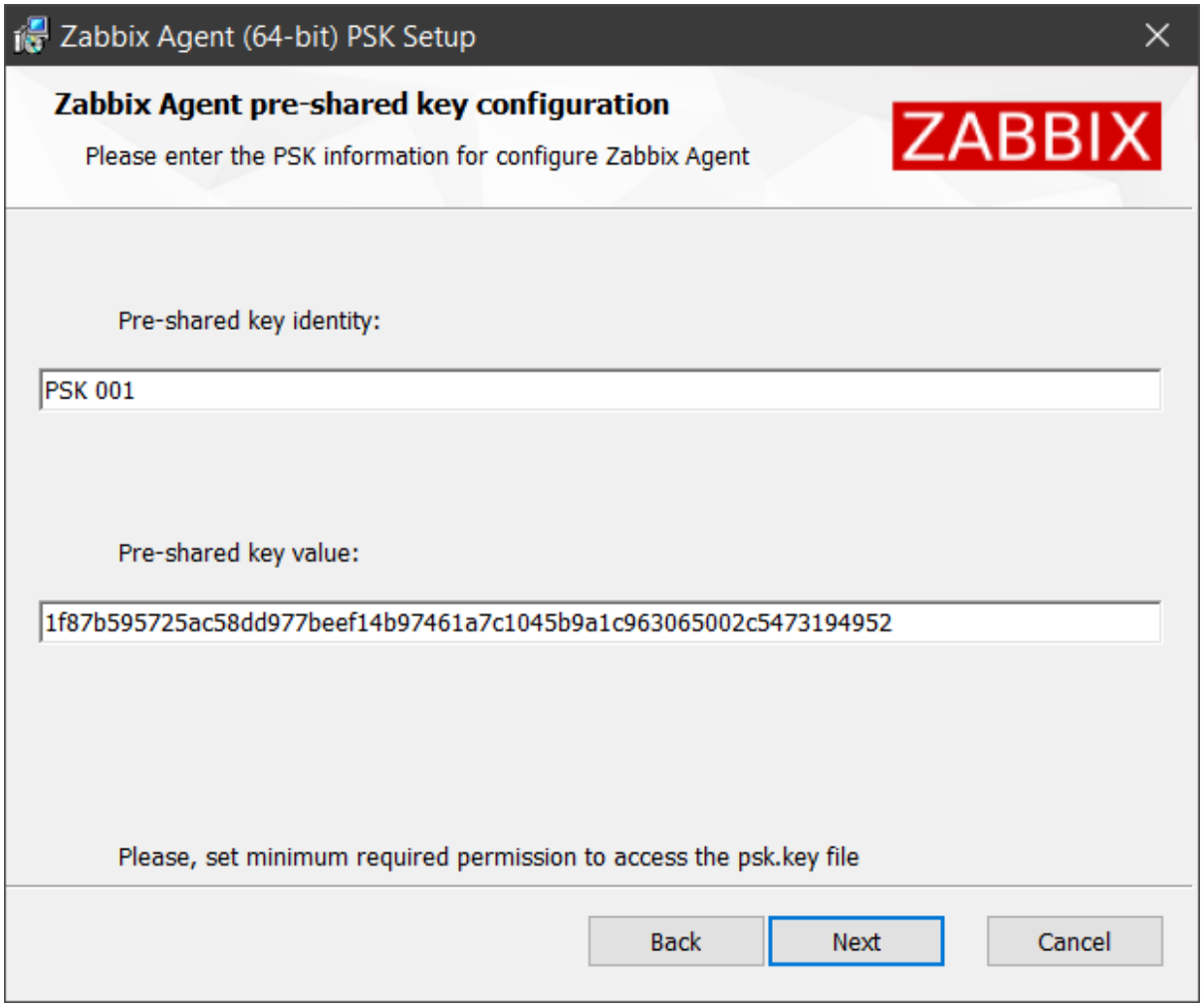
Parameter	Description
Host name	Nazwa hosta komputera, na którym jest instalowany Zabbix agent. Ustawia parametr Hostname.
Zabbix server IP/DNS	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Ten parametr jest **obowiązkowy**. Ustawia parametr Server.
Agent listen port	Agent będzie nasłuchiwał na tym porcie połączeń z serwera. Ustawia parametr ListenPort.
Server or Proxy for active checks	Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego mają być pobierane active checks. Adres serwera/proxy jest adresem IP lub nazwą DNS, z opcjonalnym portem oddzielonym dwukropkiem. Ustawia parametr ServerActive.
Enable PSK	Zaznacz pole wyboru, aby włączyć obsługę TLS z użyciem kluczy współdzielonych. Ustawia parametry TLSConnect i TLSAccept na psk.
Add agent location to the PATH	Zaznacz pole wyboru, aby dodać lokalizację Zabbix agent do systemowej zmiennej PATH.

Note:
Jeśli zostanie wykryty istniejący Zabbix agent, zostaną wyświetlone wartości parametrów z jego pliku konfiguracyjnego. Dodatkowo istniejący plik konfiguracyjny zostanie podczas instalacji zmieniony nazwę, a nowy plik konfiguracyjny zostanie utworzony.



5. Skonfiguruj parametry PSK, jeśli w poprzednim kroku zaznaczono pole *Enable PSK*. Te parametry również zostaną zapisane w pliku konfiguracyjnym Zabbix agent:

Parameter	Description
Pre-shared key identity	Identyfikator klucza współdzielonego. Ustawia parametr TLSPSKIdentity.
Pre-shared key value	Wartość klucza współdzielonego. Tworzy plik psk.key zawierający klucz i ustawia parametr TLSPSKFile na lokalizację klucza (domyślnie: `C:\Program Files\Zabbix Agent\psk.key`). **Zaleca się** ograniczenie dostępu do pliku klucza współdzielonego przez dostosowanie ustawień zabezpieczeń pliku tak, aby mógł go odczytać tylko Zabbix agent (lub użytkownik uruchamiający agenta).

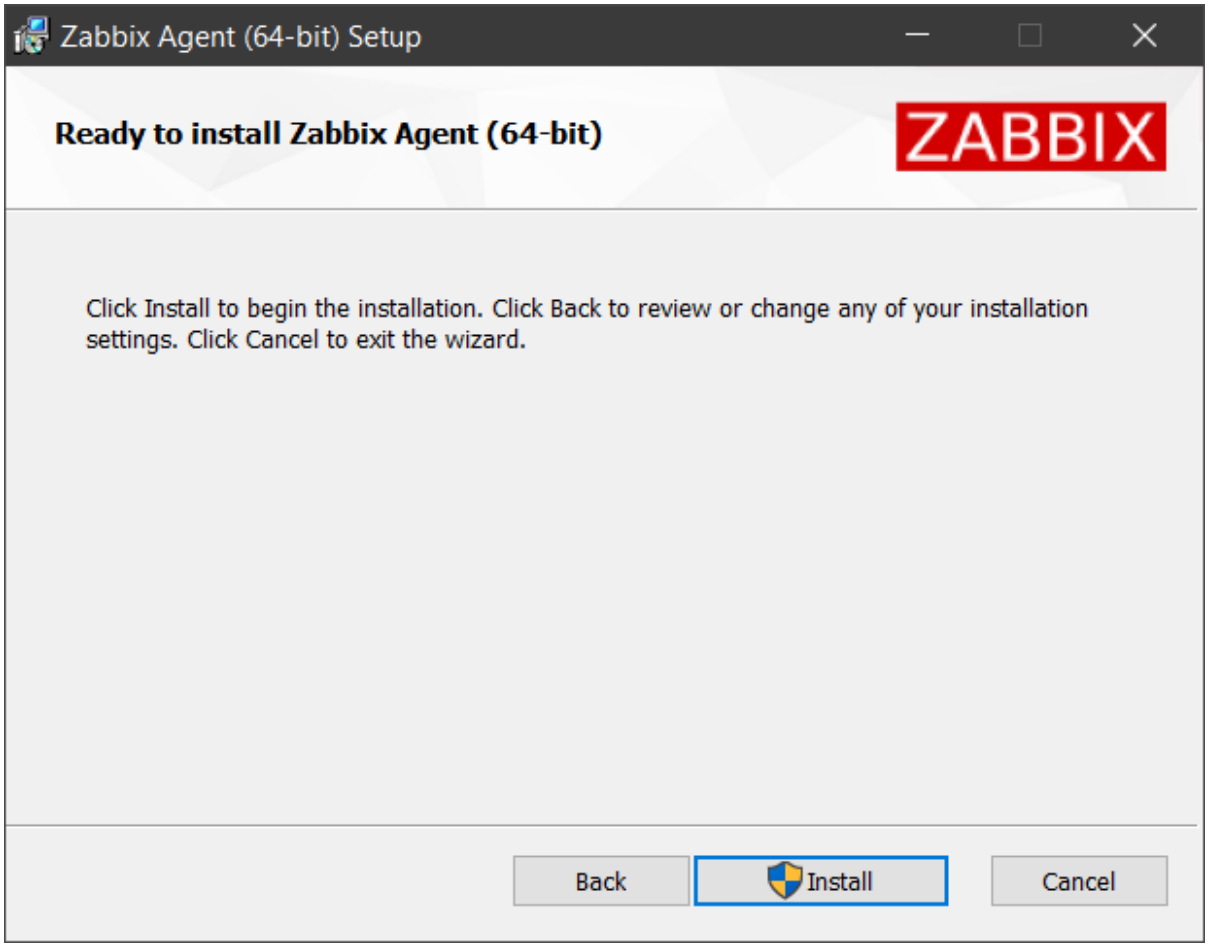


6. Kliknij *Install*, aby rozpocząć instalację.

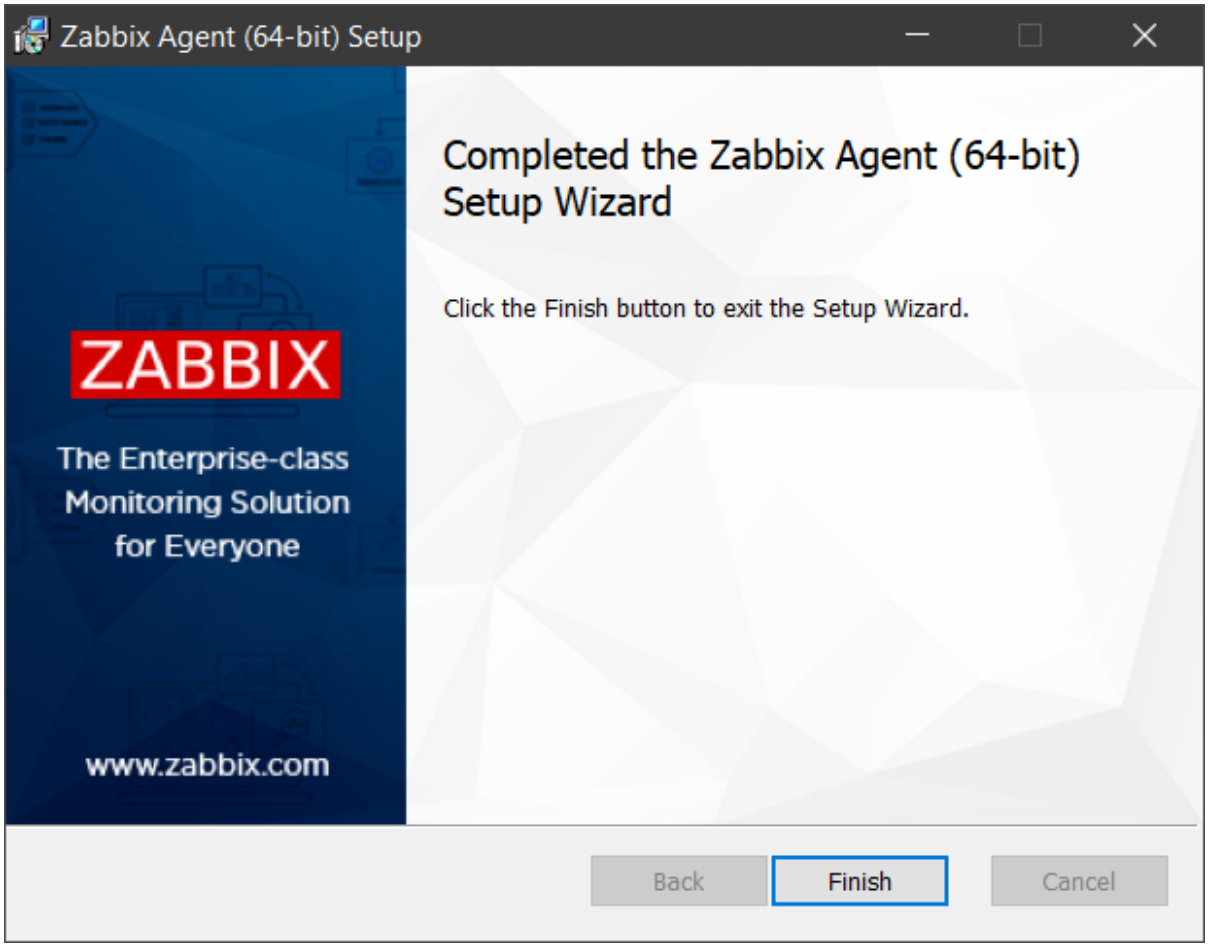
Wszystkie wybrane komponenty Zabbix oraz plik konfiguracyjny Zabbix agent zostaną zainstalowane w określonej lokalizacji (domyślnie: `C:\Program Files\Zabbix Agent`). To samo dotyczy Zabbix agent 2, z tą różnicą, że dodatkowe pliki konfiguracyjne dla jego wbudowanych wtyczek zostaną zainstalowane w podfolderze `zabbix_agent2.d\plugins.d`.

Dodatkowo zabbix_agentd.exe (lub zabbix_agent2.exe) zostanie skonfigurowany jako usługa systemu Windows z opóźnionym automatycznym uruchamianiem (lub automatycznym uruchamianiem w wersjach systemu Windows wcześniejszych niż Windows Vista/Server 2008).

Jeśli podczas instalacji działa inna wersja Zabbix agent, zostanie wyświetlony monit o wybranie jednej z opcji: zamknięcie aplikacji i ponowną próbę jej uruchomienia albo pozostawienie jej otwartej, co będzie wymagało ponownego uruchomienia systemu.



7. Kliknij przycisk *Finish*, aby zakończyć kreator instalacji.



Instalacja z wiersza poleceń

agent Zabbix można zainstalować z wiersza poleceń, uruchamiając instalator MSI za pomocą msiexec. Na przykład:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.0-windows-amd64-openssl.msi" /qn+ SERVER=192.0.2
```

Ta metoda umożliwia instalacje bez nadzoru oraz niestandardowe konfiguracje z użyciem parametrów.

Obsługiwane parametry

Pakiety instalatora MSI Zabbix agent obsługują następujące parametry zarówno dla Zabbix agent, jak i Zabbix agent 2.

Note:
Parametry Zabbix agent/agent2 są ustawiane w pliku konfiguracyjnym podczas instalacji. Kliknij nazwę parametru, aby wyświetlić jego szczegółowy opis i przykłady konfiguracji na stronie Zabbix agent (Windows). W przypadku Zabbix agent 2 zobacz stronę Zabbix agent 2 (Windows).

Parameter	Description
ADDDEFAULT	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane w domyślnej konfiguracji. Więcej informacji: właściwość ADDDEFAULT. Możliwe wartości: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Przykład: `ADDDEFAULT=AgentProgram,GetProgram`
ADDLOCAL	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane lokalnie. Więcej informacji: właściwość ADDLOCAL. Możliwe wartości: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Przykład: `ADDLOCAL=AgentProgram,SenderProgram`
ALLOWDENYKEY	Lista parametrów AllowKey lub DenyKey rozdzielonych średnikami, służąca do ograniczenia sprawdzeń Zabbix agent. W razie potrzeby użyj ukośnika odwrotnego do poprzedzenia separatora (`\;`). Ustawia parametry AllowKey i DenyKey w pliku konfiguracyjnym agenta. Przykład: `ALLOWDENYKEY="AllowKey=system.run[more C:\Windows\System32\drivers\etc\hosts\\; echo 'File read complete'];DenyKey=system.run[*]"`
CONF	Pełna ścieżka do pliku konfiguracyjnego szablonu dla Zabbix agent. Podczas instalacji ten plik stanie się plikiem konfiguracyjnym agenta. Plik musi zawierać co najmniej parametry Server i LogFile. Przykład: `CONF="C:\full\path\to\example.conf"`
DONOTSTART	Użyj `DONOTSTART=1`, aby uniemożliwić instalatorowi MSI uruchomienie usługi Zabbix agent. Obsługiwane w pakietach instalatora MSI od Zabbix 7.4.6.
ENABLEPATH	Użyj `ENABLEPATH=1`, aby dodać lokalizację Zabbix agent do systemowej zmiennej PATH.
ENABLEPERSISTENTBUFFER	Tylko Zabbix agent 2. Włącza użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji.
HOSTINTERFACE	Opcjonalny parametr definiujący interfejs hosta.
HOSTMETADATA	Opcjonalny parametr definiujący metadane hosta.
HOSTMETADATAITEM	Opcjonalny parametr definiujący pozycję używaną do pobierania metadanych hosta.
HOSTNAME	Opcjonalny parametr definiujący nazwę hosta.
HOSTNAMEITEM	Opcjonalny parametr definiujący pozycję używaną do pobierania nazwy hosta. Obsługiwane w pakietach instalatora MSI od Zabbix 7.4.6.
INCLUDE	Lista pojedynczych plików lub wszystkich plików w katalogu, rozdzielona średnikami, które mają zostać dołączone do pliku konfiguracyjnego Zabbix agent.
INSTALLFOLDER	Pełna ścieżka do folderu, w którym zostaną zainstalowane komponenty Zabbix oraz plik konfiguracyjny Zabbix agent. W przypadku Zabbix agent 2 dodatkowe pliki konfiguracyjne dla wbudowanych wtyczek zostaną zainstalowane w podfolderze `zabbix_agent2.d\plugins.d`. Przykład: `INSTALLFOLDER="C:\Program Files\Zabbix Agent"`
LISTENIP	Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać.
LISTENPORT	Agent będzie nasłuchiwał na tym porcie na połączenia z serwera.
LOGFILE	Nazwa pliku dziennika Zabbix agent.
LOGTYPE	Typ wyjścia dziennika.
NONMSICONFNAME	Pełna ścieżka do niestandardowego pliku konfiguracyjnego dla Zabbix agent. Podczas instalacji wszystkie prawidłowe parametry konfiguracji agenta obecne w tym pliku (ograniczone do tych wymienionych w tej tabeli) zostaną zapisane w nowo utworzonym pliku konfiguracyjnym agenta. Plik musi zawierać co najmniej parametr Server. Przykład: `NONMSICONFNAME="C:\full\path\to\example.conf"`
PERSISTENTBUFFERFILE	Tylko Zabbix agent 2. Plik, w którym Zabbix agent 2 ma przechowywać bazę danych SQLite.
PERSISTENTBUFFERPERIOD	Tylko Zabbix agent 2. Okres, przez który dane mają być przechowywane, gdy nie ma połączenia z serwerem lub proxy.

Parameter	Description
SERVER	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Ten parametr jest **obowiązkowy**, z wyjątkiem sytuacji, gdy STARTAGENTS ma wartość 0.
SERVERACTIVE	Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego mają być pobierane aktywne sprawdzenia.
SKIP	Użyj `SKIP=fw`, aby uniemożliwić instalatorowi MSI dodanie reguły wyjątku w Zaporze systemu Windows dla Zabbix agent.
SOURCEIP	Adres IP źródłowy dla połączeń wychodzących do serwera Zabbix lub proxy Zabbix albo do nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.). Obsługiwane w pakietach instalatora MSI od Zabbix 7.4.6.
STARTAGENTS	Liczba wstępnie uruchomionych instancji zabbix_agentd, które przetwarzają pasywne sprawdzenia. Jeśli ustawione na 0, pasywne sprawdzenia są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP. Obsługiwane w pakietach instalatora MSI od Zabbix 7.4.6.
STARTUPTYPE	Typ uruchamiania usługi Zabbix agent. Możliwe wartości: **automatic** - uruchamiaj usługę automatycznie podczas startu systemu Windows; **delayed** - *(domyślnie)* opóźnij uruchomienie usługi po zakończeniu startu usług uruchamianych automatycznie (dostępne w Windows Vista/Server 2008 i nowszych wersjach); **manual** - uruchamiaj usługę ręcznie (przez użytkownika lub aplikację); **disabled** - wyłącz usługę, aby nie mogła zostać uruchomiona przez użytkownika lub aplikację. Przykład: `STARTUPTYPE=disabled`
STATUSPORT	Tylko Zabbix agent 2. Jeśli ustawione, agent będzie nasłuchiwał na tym porcie na żądania stanu HTTP (http://localhost:<port>/status).
TIMEOUT	Określa, jak długo czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy Zabbix lub serwerem.
TLSACCEPT	Przychodzące połączenia, które mają być akceptowane (używane dla pasywnych sprawdzeń). Jeśli ustawione na `psk`, to TLSCONNECT również zostanie ustawione na `psk` (o ile nie określono inaczej).
TLSCAFILE	Pełna ścieżka do pliku zawierającego certyfikaty urzędów CA najwyższego poziomu do weryfikacji certyfikatu partnera.
TLSCERTFILE	Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów.
TLSCONNECT	Sposób, w jaki agent powinien łączyć się z serwerem Zabbix lub proxy (używane dla aktywnych sprawdzeń). Jeśli ustawione na `psk`, to TLSACCEPT również zostanie ustawione na `psk` (o ile nie określono inaczej).
TLSCRLFILE	Pełna ścieżka do pliku zawierającego certyfikaty unieważnione.
TLSKEYFILE	Pełna ścieżka do pliku zawierającego klucz prywatny Zabbix agent.
TLSPSKFILE	Pełna ścieżka do pliku zawierającego klucz współdzielony Zabbix agent. Jeśli ustawione są zarówno TLSPSKFILE, jak i TLSPSKVALUE, wartość TLSPSKVALUE zostanie zapisana do pliku określonego w TLSPSKFILE. **Zaleca się** ograniczenie dostępu do pliku z kluczem współdzielonym poprzez odpowiednie ustawienie zabezpieczeń pliku, tak aby tylko Zabbix agent (lub użytkownik uruchamiający agenta) mógł go odczytać.
TLSPSKIDENTITY	Łańcuch identyfikatora klucza współdzielonego.
TLSPSKVALUE	Wartość klucza współdzielonego. Jeśli ustawione są zarówno TLSPSKFILE, jak i TLSPSKVALUE, wartość TLSPSKVALUE zostanie zapisana do pliku określonego w TLSPSKFILE. Przykład: `TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952`
TLSSERVERCERTISSUER	Dozwolony wystawca certyfikatu serwera (proxy).
TLSSERVERCERTSUBJECT	Dozwolony podmiot certyfikatu serwera (proxy).
UNSAFEUSERPARAMETERS	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Obsługiwane w pakietach instalatora MSI od Zabbix 7.4.6.

Przykłady

Poniższy przykład instaluje agent Zabbix z niestandardową konfiguracją. Włącza również obsługę TLS z użyciem kluczy współdzielonych.

```
mkdir "C:\Program Files\Zabbix Agent" 2>nul
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.0-windows-amd64-openssl.msi" /qn+^
 SERVER=192.0.2.1^
```

```
INSTALLFOLDER="C:\Program Files\Zabbix Agent"^
HOSTNAME=LAPTOP-IKP7S51S^
TLSACCEPT=psk^
TLSCONNECT=psk^
TLSPSKIDENTITY="PSK 001"^
TLSPSKFILE="C:\Program Files\Zabbix Agent\psk.key"^
TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952^
ENABLEPATH=1^
ALLOWDENYKEY="AllowKey=system.run[type C:\Windows\System32\drivers\etc\hosts];DenyKey=system.run[*]"
```

Następny przykład instaluje nowszą wersję agenta Zabbix i używa **pliku konfiguracyjnego szablonu** (`CONF="C:\agent-template.conf"`). Podczas instalacji ten plik stanie się plikiem konfiguracyjnym agenta. Aby odziedziczyć parametry ze starego pliku konfiguracyjnego, użyj symboli zastępczych parametrów (np. `[AllowDenyKey]`).

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### agent-template.conf example:
LogFile=[LogFile]
[AllowDenyKey]
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
[Include]
[TLSConnect]
[TLSAccept]
[TLSPSKIdentity]
[TLSPSKFile]
```

Alternatywnie można użyć **niestandardowego pliku konfiguracyjnego** (`NONMSICONFNAME="C:\agent-custom.conf"`). Podczas instalacji wszystkie prawidłowe parametry konfiguracji agenta obecne w tym pliku (ograniczone do tych wymienionych w powyższej tabeli) zostaną zapisane w nowo utworzonym pliku konfiguracyjnym agenta. Aby zachować istniejącą konfigurację agenta, zdefiniuj parametry, które mają zostać zachowane.

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### agent-custom.conf example:
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
```

Ładowalne wtyczki Zabbix agent 2

Ładowalne wtyczki Zabbix agent 2 loadable plugins można zainstalować w systemie Windows przy użyciu 64-bitowych pakietów instalatora MSI, dostępnych do pobrania.

Minimalne wymagania systemowe dla instalacji MSI to Windows 10 (64-bit) lub Windows Server 2016.

Podobnie jak w przypadku Zabbix agent/agent2, ładowalne wtyczki można instalować za pomocą Kreatora instalacji lub wiersza poleceń.

Attention:
Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

Instalacja za pomocą kreatora konfiguracji

1. Kliknij dwukrotnie pobrany plik MSI, aby rozpocząć instalację.
2. Zaakceptuj umowę licencyjną użytkownika końcowego.
3. Wybierz ładowalne wtyczki Zabbix agent 2 do zainstalowania.

Attention:
Zaleca się użycie domyślnej lokalizacji instalacji udostępnionej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może zagrozić bezpieczeństwu instalacji.

4. Kliknij *Install*, aby rozpocząć instalację.

Wszystkie wybrane ładowalne wtyczki Zabbix agent 2 zostaną zainstalowane w określonej lokalizacji (domyślnie: `C:\Program Files\Zabbix Agent 2`), a ich pliki konfiguracyjne zostaną zainstalowane w podfolderze `zabbix_agent2.d`.

5. Kliknij przycisk *Finish*, aby zamknąć kreatora konfiguracji.

Instalacja z wiersza poleceń

Ładowalne wtyczki Zabbix agent 2 można zainstalować z wiersza poleceń, uruchamiając instalator MSI za pomocą msiexec. Na przykład:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent2_plugins-7.4.0-windows-amd64.msi" /qn+
```

Pakiety instalatora MSI dla ładowalnych wtyczek Zabbix agent 2 obsługują następujące parametry.

Parameter	Description
ADDDEFAULT	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane w domyślnej konfiguracji. Więcej informacji można znaleźć w ADDDEFAULT property. Possible values: `ALL`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Example: `ADDDEFAULT=MongodbPlugin,PostgresqlPlugin`
ADDLOCAL	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane lokalnie. Więcej informacji można znaleźć w ADDLOCAL property. Possible values: `ALL`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Example: `ADDLOCAL=MongodbPlugin,MssqlPlugin`
INSTALLFOLDER	Pełna ścieżka do folderu, w którym zostaną zainstalowane komponenty Zabbix, a ich pliki konfiguracyjne zostaną umieszczone w podfolderze `zabbix_agent2.d`. Example: `INSTALLFOLDER="C:\Program Files\Zabbix Agent 2"`

2 Instalacja agenta macOS z pakietu PKG

Przegląd

Agent Zabbix można zainstalować w systemie macOS za pomocą pakietów instalatora PKG, dostępnych do pobrania.

Pakiety agenta Zabbix są dostępne z szyfrowaniem lub bez niego.

Instalowanie agent

Agent można zainstalować za pomocą graficznego interfejsu użytkownika lub z wiersza poleceń, na przykład:

```
sudo installer -pkg zabbix_agent-7.4.0-macos-arm64-openssl.pkg -target /
```

Upewnij się, że w poleceniu używasz właściwej wersji pakietu Zabbix. Musi ona odpowiadać nazwie pobranego pakietu.

Uruchamianie agent

Agent uruchomi się automatycznie po instalacji lub restarcie.

W razie potrzeby możesz edytować plik konfiguracyjny `/usr/local/etc/zabbix/zabbix_agentd.conf`.

Aby uruchomić agent ręcznie, możesz wykonać:

```
sudo launchctl start com.zabbix.zabbix_agentd
```

Aby zatrzymać agent ręcznie:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

Podczas aktualizacji istniejący plik konfiguracyjny nie jest nadpisywany. Zamiast tego tworzony jest nowy plik `zabbix_agentd.conf.NEW`, który służy do przeglądu i aktualizacji istniejącego pliku konfiguracyjnego, jeśli to konieczne. Pamiętaj, aby po ręcznych zmianach w pliku konfiguracyjnym ponownie uruchomić agent.

Rozwiązywanie problemów i usuwanie agent

W tej sekcji wymieniono kilka przydatnych poleceń, których można użyć do rozwiązywania problemów i usuwania instalacji Zabbix agent.

Sprawdź, czy Zabbix agent jest uruchomiony:

```
ps aux | grep zabbix_agentd
```

Sprawdź, czy Zabbix agent został zainstalowany z pakietów:

```
pkgutil --pkgs | grep zabbix
com.zabbix.pkg.ZabbixAgent
```

Sprawdź pliki, które zostały zainstalowane z pakietu instalacyjnego (zwróć uwagę, że początkowy znak / nie jest wyświetlany w tym widoku):

```
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
```

Zatrzymaj Zabbix agent, jeśli został uruchomiony za pomocą `launchctl`:

```
sudo launchctl unload /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
```

Usuń pliki (w tym konfigurację i logi), które zostały zainstalowane przez pakiet instalacyjny:

```
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
```

Usuń informację o tym, że Zabbix agent został zainstalowany:

```
sudo pkgutil --forget com.zabbix.pkg.ZabbixAgent
```

3 Niestabilne wydania

Przegląd

Poniższe instrukcje dotyczą włączania niestabilnych repozytoriów wydań Zabbix (domyślnie wyłączonych), używanych dla kandydatów do wydania drobnych wersji Zabbix.

Najpierw zainstaluj lub zaktualizuj pakiet zabbix-release do najnowszej wersji. Aby włączyć pakiety rc w swoim systemie, wykonaj następujące czynności:

Red Hat Enterprise Linux

Otwórz plik `/etc/yum.repos.d/zabbix.repo` i ustaw `enabled=1` dla repozytorium `zabbix-unstable`.

```
[zabbix-unstable]
name=Zabbix Official Repository (unstable) - $basearch
baseurl=https://repo.zabbix.com/zabbix/7.4/unstable/rhel/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
```

Debian/Ubuntu

Otwórz plik `/etc/apt/sources.list.d/zabbix.list` i usuń komentarz z wpisu "Zabbix unstable repository".

```
#### Zabbix unstable repository
deb https://repo.zabbix.com/zabbix/7.4/unstable/debian bullseye main
deb-src https://repo.zabbix.com/zabbix/7.4/unstable/debian bullseye main
```

SUSE

Otwórz plik `/etc/zypp/repos.d/zabbix.repo` i ustaw `enable=1` dla repozytorium `zabbix-unstable`.

```
[zabbix-unstable]
name=Zabbix Official Repository
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/7.4/unstable/sles/15/x86_64/
gpgcheck=1
```

```
gpgkey=https://repo.zabbix.com/zabbix/7.4/unstable/sles/15/x86_64/repodata/repomd.xml.key
enabled=1
update=1
```

5 Installation from containers

Przegląd Ta strona opisuje, jak wdrożyć Zabbix przy użyciu Docker lub Docker Compose.

Docker Compose to najszybszy sposób wdrożenia Zabbix. Odczytuje plik konfiguracyjny i automatycznie uruchamia wszystkie kontenery składające się na kompletne środowisko Zabbix, we właściwej kolejności.

Docker (ręczne wdrożenie) osiąga ten sam rezultat, wdrażając każdy komponent krok po kroku.

Attention:
Od Zabbix 6.0 deterministyczne wyzwalacze muszą być tworzone podczas instalacji. Jeśli włączone jest binarne logowanie dla MySQL/MariaDB, wymaga to uprawnień superużytkownika lub ustawienia zmiennej/parametru konfiguracyjnego `log_bin_trust_function_creators = 1`. Instrukcje dotyczące ustawiania tej zmiennej znajdują się w sekcji Skrypty tworzenia bazy danych.

 Należy pamiętać, że jeśli wykonujesz to z konsoli, zmienna zostanie ustawiona tylko tymczasowo i zostanie usunięta po ponownym uruchomieniu Docker. W takim przypadku pozostaw usługę SQL uruchomioną, zatrzymaj tylko usługę `zabbix-server`, uruchamiając `docker compose down zabbix-server`, a następnie `docker compose up -d zabbix-server`.

 Alternatywnie można ustawić tę zmienną w pliku konfiguracyjnym.

Wymagania wstępne Przed rozpoczęciem upewnij się, że Docker (1.12.0 lub nowszy) jest zainstalowany w systemie. Jeśli nie, skorzystaj z przewodnika instalacji Dockera. W przypadku wdrożenia z użyciem Docker Compose wymagany jest również Docker Compose (2.24.0 lub nowszy).

Niektóre komponenty Zabbix wymagają otwarcia określonych portów na hoście, na którym działa Docker (np. 10051/TCP dla serwera Zabbix, 162/UDP dla pułapek SNMP). Pełną □□□□ portów używanych przez komponenty Zabbix znajdziesz w sekcji Wymagania. Dla serwera Zabbix i agent domyślny port można zmienić, ustawiając zmienną środowiskową `ZBX_LISTENPORT` w odpowiednim kontenerze.

Dostępne obrazy Docker Zabbix udostępnia obraz Docker dla każdego komponentu Zabbix, a wszystkie są publikowane w serwisie Docker Hub. Każdy obraz służy do utworzenia kontenera uruchamiającego dany komponent.

Komponent Zabbix	Obraz Docker
Agent	zabbix/zabbix-agent
Agent 2	zabbix/zabbix-agent2
Serwer (MySQL)	zabbix/zabbix-server-mysql
Serwer (PostgreSQL)	zabbix/zabbix-server-pgsql
Interfejs webowy (Apache + MySQL)	zabbix/zabbix-web-apache-mysql
Interfejs webowy (Apache + PostgreSQL)	zabbix/zabbix-web-apache-pgsql
Interfejs webowy (Nginx + MySQL)	zabbix/zabbix-web-nginx-mysql
Interfejs webowy (Nginx + PostgreSQL)	zabbix/zabbix-web-nginx-pgsql
Proxy (SQLite3)	zabbix/zabbix-proxy-sqlite3
Proxy (MySQL)	zabbix/zabbix-proxy-mysql
Java gateway	zabbix/zabbix-java-gateway
Usługa webowa	zabbix/zabbix-web-service
Trapy SNMP	zabbix/zabbix-snmptraps

Note:
Aby używać trapów SNMP, kontener trapów SNMP musi współdzielić wolumin z kontenerem serwera Zabbix lub proxy (zobacz jak używać tego obrazu w Docker Hub oraz poniższy przykład).

Wszystkie obrazy komponentów Zabbix w Docker Hub są oparte na najnowszych głównych wersjach obsługiwanych systemów operacyjnych. Obrazy te są automatycznie przebudowywane po aktualizacji bazowych obrazów systemu operacyjnego.

Tagi obrazów

Każdy obraz obsługuje tagi służące do wyboru bazowego systemu operacyjnego i wersji Zabbix:

```
zabbix/<image>:<os>-<version>
```

Obsługiwane wartości <os>:

- `alpine` - Alpine Linux
- `ubuntu` - Ubuntu
- `centos` - CentOS Stream
- `ol` - Oracle Linux
- `ltsc2022` - Windows 11 LTSC 2022 (tylko agent Zabbix)

Obsługiwane wartości <version>:

- `latest` - Najnowsza stabilna wersja Zabbix na Alpine Linux
- `<os>-latest` - Najnowsza stabilna wersja Zabbix na wybranym systemie operacyjnym
- `<os>-trunk` - Najnowsza kompilacja rozwojowa (nightly) na wybranym systemie operacyjnym
- `<os>-X.X-latest` - Najnowsze wydanie pomocnicze Zabbix dla określonej głównej wersji Zabbix na wybranym systemie operacyjnym
- `<os>-X.X.*` - Określone wydanie pomocnicze Zabbix na wybranym systemie operacyjnym

Przykłady:

```
### Najnowszy stabilny proxy Zabbix (SQLite3) na Alpine Linux:
docker pull zabbix/zabbix-proxy-sqlite3:latest

### Najnowszy stabilny proxy Zabbix (SQLite3) na Ubuntu:
docker pull zabbix/zabbix-proxy-sqlite3:ubuntu-latest

### Najnowsza rozwojowa (nightly) kompilacja serwer Zabbix (MySQL) na Ubuntu:
docker pull zabbix/zabbix-server-mysql:ubuntu-trunk

### Najnowsze wydanie pomocnicze 7.4 serwer Zabbix (MySQL) na Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-7.4-latest

### Wersja 7.4.1 serwer Zabbix (MySQL) na Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-7.4.1
```

Docker Compose Docker Compose to najszybszy sposób wdrożenia Zabbix. Odczytuje plik konfiguracyjny (plik Compose) i obsługuje całą konfigurację — pobieranie obrazów Docker, tworzenie wewnętrznej sieci między kontenerami, konfigurację magazynu, inicjalizację bazy danych oraz uruchamianie wszystkiego we właściwej kolejności.

Repozytorium oficjalnych plików Docker Zabbix udostępnia gotowe do użycia pliki Docker Compose oraz system konfiguracji oparty na `.env`, obsługujący różne systemy operacyjne, backendy baz danych i konfigurację komponentów Zabbix.

1. Sklonuj repozytorium, przejdź do niego i przełącz się na wersję 7.4:

```
git clone https://github.com/zabbix/zabbix-docker.git
cd zabbix-docker
git checkout 7.4
```

2. (Opcjonalnie) Dostosuj wdrożenie za pomocą zmiennych środowiskowych. Ten krok można pominąć w przypadku domyślnej konfiguracji.

3. Wdróż domyślną konfigurację — serwer Zabbix, interfejs webowy (działający na Nginx) oraz backend bazy danych (MySQL lub PostgreSQL), każdy uruchomiony w osobnym kontenerze na Alpine Linux.

```
### Z MySQL jako bazą danych:
docker compose -f ./compose.yaml up -d

### Z PostgreSQL jako bazą danych:
docker compose -f ./compose_pgsql.yaml up -d
```

Po uruchomieniu kontenerów (zwykle w ciągu 1–3 minut) serwer Zabbix zostanie uruchomiony, a interfejs webowy będzie dostępny pod adresem `http://localhost`.

Note:
Użyj `docker compose ps`, aby sprawdzić stan kontenerów. Wszystkie kontenery (z wyjątkiem kontenera `zabbix-docker-server-db-init-1`) powinny mieć status Up. Jeśli którykolwiek ma status `Exited`, sprawdź jego logi za pomocą `docker logs <container-name>`.

Zmienne środowiskowe

Zachowanie Docker Compose oraz konfigurację komponentów Zabbix można dostosować za pomocą zmiennych środowiskowych.

Zmienne na poziomie Compose (zdefiniowane w pliku `.env`) kontrolują, które obrazy Docker, porty i zakresy adresów IP sieci są używane. Możesz używać tych zmiennych bezpośrednio przed poleceniem `docker compose` albo edytować plik `.env`.

Na przykład poniższe polecenie wdraża pełną, wielokontenerową konfigurację Zabbix z użyciem obrazu opartego na Ubuntu (zamiast domyślnego Alpine Linux) oraz udostępnia frontend przez Nginx na niestandardowych portach HTTP (8282) i HTTPS (8443):

```
OS=ubuntu \
ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
ZABBIX_WEB_NGINX_HTTPS_PORT=8443 \
docker compose -f ./compose.yaml up -d
```

Zmienne na poziomie komponentu (zdefiniowane w plikach `env_vars/.env_<component>`) kontrolują konfigurację komponentów Zabbix. Przed uruchomieniem polecenia `docker compose` edytuj odpowiedni plik `.env_<component>`.

Na przykład możesz dostosować liczbę pasywnych pollerów serwera Zabbix i ustawienia pamięci podręcznej, a także skonfigurować strefę czasową frontendu, edytując następujące zmienne:

```
### env_vars/.env_srv
ZBX_STARTPOLLERS=20
ZBX_CACHESIZE=64M
ZBX_HISTORYCACHESIZE=32M

### env_vars/.env_web
PHP_TZ=Europe/Paris
```

Note:
Zmienne środowiskowe na poziomie komponentu odpowiadają parametrom konfiguracji komponentów Zabbix, ale używają innego stylu nazewnictwa (np. `ZBX_STARTPOLLERS` odpowiada `StartPollers`). Niektóre zmienne są specyficzne dla Dockera, a niektórych parametrów konfiguracji nie można zmienić (np. `PIDFile` i `LogType`). Korzystając ze zmiennych środowiskowych, zapoznaj się z sekcją *Environment variables* w opisie każdego komponentu obrazu Docker na Docker Hub.

Wolumeny

Docker Compose przechowuje trwałe dane w katalogu `zbx_env/` utworzonym obok pliku Compose. Katalog ten zachowuje dane komponentów między ponownymi uruchomieniami kontenerów i aktualizacjami.

Zawartość `zbx_env/` jest zdefiniowana z góry dla obrazu każdego komponentu. Na przykład:

- `zbx_env/etc/` - używany przez kontener interfejsu web Zabbix
- `zbx_env/usr/` i `zbx_env/var/` - używane przez kontener serwera Zabbix

Więcej informacji o każdym wolumenie można znaleźć w sekcji *Allowed volumes* w opisie obrazu Docker każdego komponentu na Docker Hub.

Makefile

Repozytorium Zabbix Dockerfiles udostępnia również plik Makefile jako skrót do typowych zadań Docker Compose. Zamiast używać pełnego polecenia `docker compose`, można korzystać z krótszych poleceń `make` (uruchom `make help`, aby zobaczyć dostępne opcje):

```
### Domyślne wdrożenie (serwer Zabbix, frontend, MySQL, wszystko na Alpine Linux)
make up

### Własne wdrożenie (serwer Zabbix, frontend z niestandardowymi portami Nginx, PostgreSQL, wszystko na Ub
make up \
  OS=ubuntu \
  DB=pgsql \
  ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
  ZABBIX_WEB_NGINX_HTTPS_PORT=8443
```

Attention:
Podczas zatrzymywania lub usuwania kontenerów zawsze podawaj ten sam typ bazy danych, który został wdrożony (np. `make down DB=pgsql`).

Domyślnie `make up` uruchamia tylko minimalny zestaw usług (serwer Zabbix, frontend i baza danych). Dzięki temu konfiguracja pozostaje lekka i nie uruchamia komponentów, które mogą nie być potrzebne. Aby uwzględnić dodatkowe komponenty, można użyć następujących profili Compose:

```
### Domyślne wdrożenie + agent Zabbix, Java gateway, usługa web i pułapki SNMP:
make up COMPOSE_PROFILES=full

### COMPOSE_PROFILES=full + proxy Zabbix (MySQL i SQLite3):
make up COMPOSE_PROFILES=all
```

Docker (ręczne wdrożenie) Użyj ręcznego wdrożenia, gdy chcesz wdrażać Zabbix krok po kroku, uruchamiać poszczególne komponenty, integrować je z istniejącym środowiskiem lub korzystać z alternatywnego środowiska uruchomieniowego kontenerów, takiego jak Podman.

Na przykład, aby wdrożyć proxy Zabbix, uruchom następujące polecenie:

```
docker run --name zabbix-proxy-sqlite3 \
  -e ZBX_SERVER_HOST=192.0.2.1 \
  -e ZBX_PROXYMODE=0 \
  -e ZBX_HOSTNAME=zabbix-proxy-sqlite3 \
  -v zabbix-proxy-data:/var/lib/zabbix/db_data \
  --init \
  -d \
  zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest
```

To polecenie:

- Pobiera obraz Docker `zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest` Docker image.
- Tworzy i uruchamia kontener `zabbix-proxy-sqlite3` oparty na obrazie Docker (z flagami `--init` i `-d`).
- Konfiguruje parametr proxy Zabbix `Server` za pomocą zmiennej środowiskowej `ZBX_SERVER_HOST`. Ten parametr określa adres IP serwera Zabbix, z którego proxy będzie pobierać dane konfiguracyjne i do którego będzie wysyłać zebrane dane. Zwróć uwagę, że inne zmienne wymagane do działania aktywnego proxy używają wartości domyślnych i można je pominąć.

Note:
Zmienne środowiskowe odpowiadają parametrom konfiguracji komponentów Zabbix, ale używają innego stylu nazewnictwa (np. `ZBX_SERVER_HOST` odpowiada `Server`). Niektóre zmienne są specyficzne dla Dockera, a niektórych parametrów konfiguracji nie można zmienić (np. `PIDFile` i `LogType`). Podczas korzystania ze zmiennych środowiskowych zapoznaj się z sekcją *Environment variables* w opisie każdego Docker image komponentu na Docker Hub.

- Łączy lokalizację magazynu zarządzaną przez Docker (np. `/var/lib/docker/volumes/zabbix-proxy-data/`) z katalogiem `/var/lib/zabbix` kontenera, aby dane proxy Zabbix były przechowywane trwale, nawet jeśli kontener zostanie usunięty.

Note:
Więcej informacji o każdym woluminie znajdziesz w sekcji *Allowed volumes* w opisie każdego Docker image komponentu na Docker Hub.

Po wdrożeniu kontenera proxy Zabbix możesz przejść do dodania proxy w frontend Zabbix oraz skonfigurowania hostów, które mają być monitorowane przez to proxy.

Poniższe przykłady obejmują trzy dodatkowe scenariusze wdrożenia:

- serwer Zabbix (MySQL) z Java gateway
- serwer Zabbix (PostgreSQL) z pułapkami SNMP
- serwer Zabbix (MySQL) z Java gateway na RHEL 8

Więcej przykładów znajdziesz w opisie każdego Docker image komponentu na Docker Hub.

Wdrożenie serwera Zabbix (MySQL) z Java gateway

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z backendem MySQL, Java gateway do monitorowania JMX oraz interfejsem webowym opartym na Nginx.

1. Utwórz dedykowaną sieć Docker, aby wszystkie kontenery komponentów Zabbix mogły się wzajemnie komunikować po nazwie kontenera:

```
docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Uruchom pusty kontener serwera MySQL (zamień `zabbix_pwd` i `root_pwd` na silne hasła):

```
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 \
  --restart unless-stopped \
  -d mysql:8.0-oracle \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_bin \
  --default-authentication-plugin=mysql_native_password
```

Attention:
Jeśli używasz obrazu Docker dla MySQL 8.4 lub nowszego, zastąp `--default-authentication-plugin=mysql_native_password` opcją `--authentication-policy=caching_sha2_password`.

 MySQL musi zostać w pełni zainicjalizowany przed uruchomieniem kontenera serwera Zabbix w kroku 4, w przeciwnym razie schemat Zabbix może nie zostać załadowany poprawnie. Aby potwierdzić, że MySQL jest gotowy, uruchom `docker logs mysql-server` i przejdź dalej dopiero wtedy, gdy zobaczysz `/usr/sbin/mysqld: ready for connections`.

3. Uruchom kontener Zabbix Java gateway:

```
docker run --name zabbix-java-gateway -t \
  --network=zabbix-net \
  --restart unless-stopped \
  -d zabbix/zabbix-java-gateway:alpine-7.4-latest
```

4. Uruchom kontener serwera Zabbix i połącz go z kontenerem serwera MySQL (zamień `zabbix_pwd` i `root_pwd` na te same hasła, których użyto w kroku 2):

```
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-7.4-latest
```

Attention:
Po uruchomieniu kontenera serwera Zabbix poczekaj, aż serwer Zabbix zakończy inicjalizację schematu bazy danych. Aby potwierdzić, że jest gotowy, uruchom `docker logs zabbix-server-mysql` i przejdź dalej dopiero wtedy, gdy nie będzie już widoczny komunikat `Creating 'zabbix' schema in MySQL`.

5. Uruchom kontener interfejsu webowego Zabbix i połącz go z kontenerami serwera Zabbix oraz serwera MySQL (zamień `zabbix_pwd` i `root_pwd` na te same hasła, których użyto w kroku 2):

```
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-7.4-latest
```

Po uruchomieniu kontenerów (zwykle w ciągu 1–3 minut) serwer Zabbix zostanie uruchomiony, a interfejs webowy będzie dostępny pod adresem `http://localhost`.

Wdrożenie serwera Zabbix (PostgreSQL) z SNMP traps

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z backendem PostgreSQL, SNMP traps oraz interfejsem webowym opartym na Nginx.

1. Utwórz dedykowaną sieć Docker, aby wszystkie kontenery komponentów Zabbix mogły się wzajemnie komunikować po nazwie kontenera:

```
docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Uruchom pusty kontener serwera PostgreSQL (zamień `zabbix_pwd` na silne hasło):

```
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. Uruchom kontener Zabbix SNMP traps:

```
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-7.4-latest
```

4. Uruchom kontener serwera Zabbix, połącz go z kontenerem serwera PostgreSQL oraz kontenerem SNMP traps (zamień `zabbix_pwd` na to samo hasło, którego użyto w kroku 2):

```
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-7.4-latest
```

> **Attention:**
> Po uruchomieniu kontenera serwera Zabbix poczekaj, aż serwer Zabbix zakończy inicjalizację schematu bazy danych. Aby potwierdzić, że jest gotowy, uruchom `docker logs zabbix-server-pgsql` i przejdź dalej dopiero wtedy, gdy nie będzie już widoczny komunikat `Creating 'zabbix' schema in PostgreSQL`.

5. Uruchom kontener interfejsu webowego Zabbix i połącz go z kontenerami serwera Zabbix oraz serwera PostgreSQL (zamień `zabbix_pwd` na to samo hasło, którego użyto w kroku 2):

```
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-7.4-latest
```

Po uruchomieniu kontenerów (zwykle w ciągu 1–3 minut) serwer Zabbix zostanie uruchomiony, a interfejs webowy będzie dostępny pod adresem `http://localhost`.

Wdrożenie serwera Zabbix (MySQL) z Java gateway na RHEL 8–10

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z backendem MySQL, Java gateway do monitorowania JMX oraz interfejsem webowym opartym na Nginx, działające na Red Hat Enterprise Linux 8, 9 lub 10.

W Red Hat Enterprise Linux zalecanym środowiskiem uruchomieniowym kontenerów jest Podman zamiast Docker. Podman działa podobnie do Dockera, ale nie wymaga działającej w tle usługi uruchamianej jako root, co lepiej pasuje do środowisk Red Hat.

1. Utwórz nowy pod o nazwie `zabbix` z wystawionymi portami dla interfejsu webowego Zabbix i trappera serwera Zabbix:

```
podman pod create --name zabbix -p 80:8080 -p 10051:10051
```

2. (Opcjonalnie) Uruchom kontener agenta Zabbix w podzie `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-74:latest
```

3. Utwórz katalog `./mysql/` na hoście RHEL:

```
mkdir -p ./mysql
```

4. Uruchom pusty kontener serwera MySQL (zamień `zabbix_pwd` i `root_pwd` na silne hasła):

```
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=utf8mb4 --collation-server=utf8mb4_bin \
  --default-authentication-plugin=mysql_native_password
```

Attention:
Jeśli używasz obrazu Docker dla MySQL 8.4 lub nowszego, zastąp `--default-authentication-plugin=mysql_native_password` opcją `--authentication-policy=caching_sha2_password`.

 MySQL musi zostać w pełni zainicjalizowany przed uruchomieniem kontenera serwera Zabbix w kroku 6, w przeciwnym razie schemat Zabbix może nie zostać załadowany poprawnie. Aby potwierdzić, że MySQL jest gotowy, uruchom `podman logs mysql-server` i przejdź dalej dopiero wtedy, gdy zobaczysz `/usr/sbin/mysqld: ready for connections`.

5. Uruchom kontener Zabbix Java gateway:

```
podman run --name zabbix-java-gateway -t \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-java-gateway-74
```

6. Uruchom kontener serwera Zabbix (zamień `zabbix_pwd` i `root_pwd` na te same hasła, których użyto w kroku 4):

```
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-74
```

7. Uruchom kontener interfejsu webowego Zabbix (zamień `zabbix_pwd` i `root_pwd` na te same hasła, których użyto w kroku 4):

```
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-74
```

Note:
Pod `zabbix` udostępnia port 80/TCP (HTTP) na hoście z portu 8080/TCP kontenera `zabbix-web-mysql`.

6 Instalacja interfejsu webowego

Ta sekcja zawiera instrukcje krok po kroku dotyczące instalacji frontend Zabbix.

Frontend Zabbix jest napisany w PHP, więc do jego uruchomienia potrzebny jest serwer WWW obsługujący PHP.

Note:
Więcej informacji na temat konfiguracji SSL dla frontend Zabbix można znaleźć, zapoznając się z tymi najlepszymi praktykami.

Ekran powitalny

Otwórz adres URL frontend Zabbix w przeglądarce. Jeśli Zabbix został zainstalowany z pakietów, adres URL to:

- dla Apache: *http://<server_ip_or_name>/zabbix*
- dla Nginx: *http://<server_ip_or_name>*

Powinien zostać wyświetlony pierwszy ekran kreatora instalacji frontend.

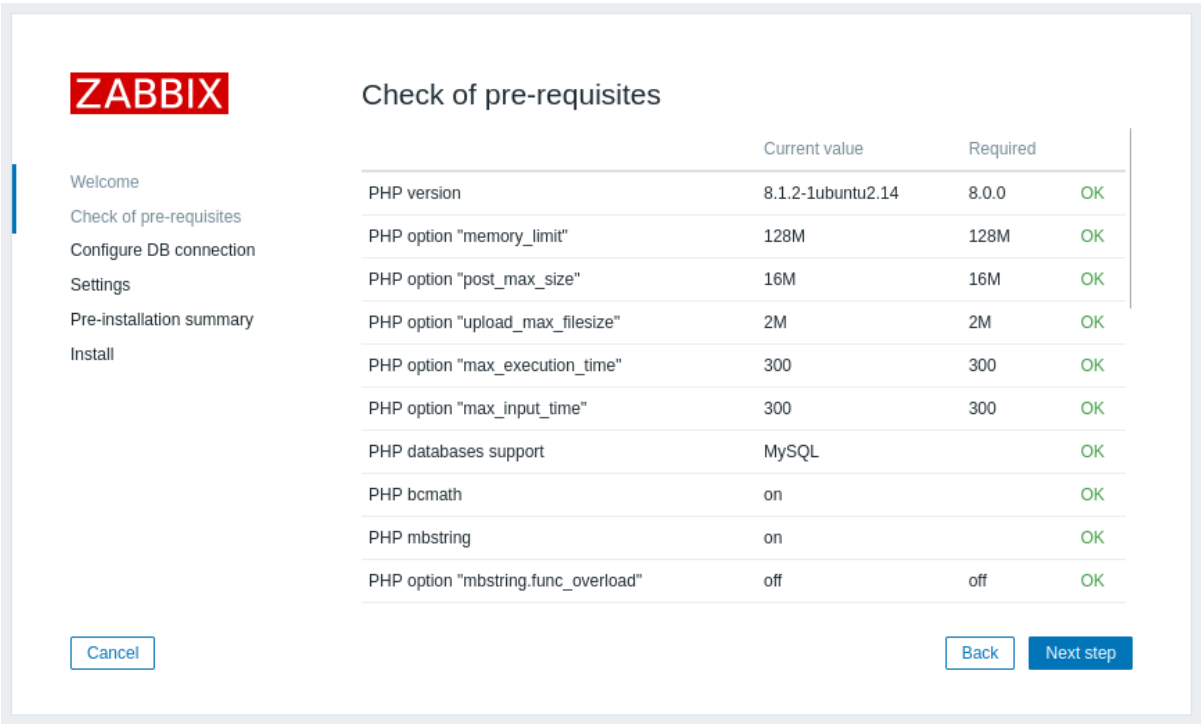
Użyj listy rozwijanej *Default language*, aby zmienić domyślny język systemu i kontynuować proces instalacji w wybranym języku (opcjonalnie). Więcej informacji można znaleźć w sekcji Instalacja dodatkowych języków frontend.

Należy pamiętać, że ustawienie języka na *English (en_US)* spowoduje również włączenie amerykańskiego formatu daty i czasu w frontend.



Sprawdzenie wymagań wstępnych

Upewnij się, że wszystkie obowiązkowe wymagania wstępne frontend Zabbix są spełnione.



Wymaganie wstępne	Wartość minimalna	Opis
Wersja PHP	8.0.0	
Opcja PHP "memory_limit"	128MB	W pliku php.ini: memory_limit = 128M
Opcja PHP "post_max_size"	16MB	W pliku php.ini: post_max_size = 16M
Opcja PHP "upload_max_filesize"	2MB	W pliku php.ini: upload_max_filesize = 2M
Opcja PHP "max_execution_time"	300 sekund	W pliku php.ini: max_execution_time = 300 (dozwolone są również wartości 0 i -1)
Opcja PHP "max_input_time"	300 sekund	W pliku php.ini: max_input_time = 300 (dozwolone są również wartości 0 i -1)

Wymaganie wstępne	Wartość minimalna	Opis
Obsługa baz danych PHP	Jedna z: MySQL, PostgreSQL	Listę wszystkich obowiązkowych i opcjonalnych rozszerzeń PHP znajdziesz w sekcji Wymagania. Pamiętaj, że niespełnione opcjonalne wymagania wstępne są wyświetlane ze statusem czerwonego *Ostrzeżenia*, a proces instalacji może być kontynuowany nawet wtedy, gdy nie są spełnione.
PHP bcmath	musi być włączone	
PHP mbstring	musi być włączone	
Opcja PHP "mbstring.func_overload"	musi być wyłączona	W pliku php.ini: mbstring.func_overload = 0
Opcja PHP "session.auto_start"	musi być wyłączona	W pliku php.ini: session.auto_start = 0
Opcja PHP "arg_separator.output"	&	W pliku php.ini: arg_separator.output = "&" (wartość "&" jest również dozwolona)

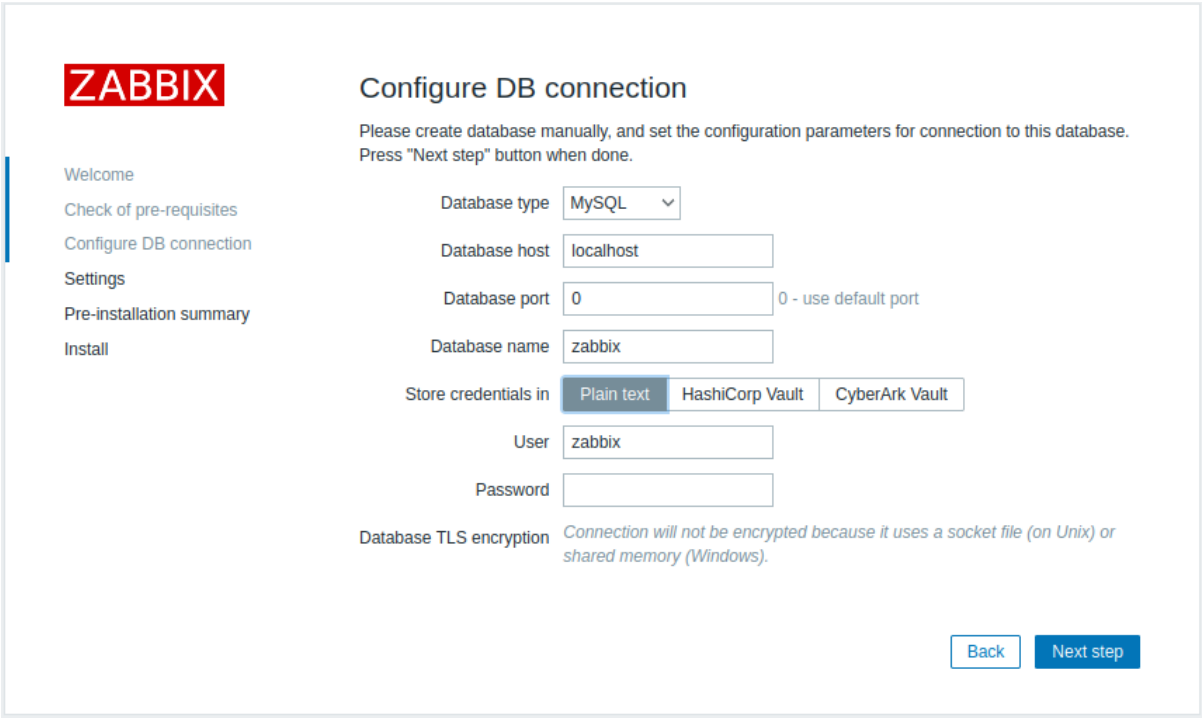
Attention:
Jeśli trzeba zmienić użytkownika lub grupę użytkownika Apache, sprawdź uprawnienia do folderu sesji; w przeciwnym razie konfiguracja Zabbix może nie być w stanie kontynuować.

Konfiguracja połączenia z bazą danych

Wprowadź szczegóły połączenia z bazą danych. Baza danych Zabbix musi już być utworzona.

W przypadku MySQL wpisanie `localhost` lub pozostawienie pola *Database host* pustego powoduje połączenie za pomocą domyślnego gniazda Unix. Formularz instalacyjny nie udostępnia osobnego pola *Database socket*, więc aby użyć niestandardowego gniazda, skonfiguruj je w ustawieniach serwera Zabbix (na przykład za pomocą `DBSocket=` w pliku zabbix_server.conf). Dzięki temu frontend pozostaje zgodny z ustawieniami połączenia serwera z bazą danych.

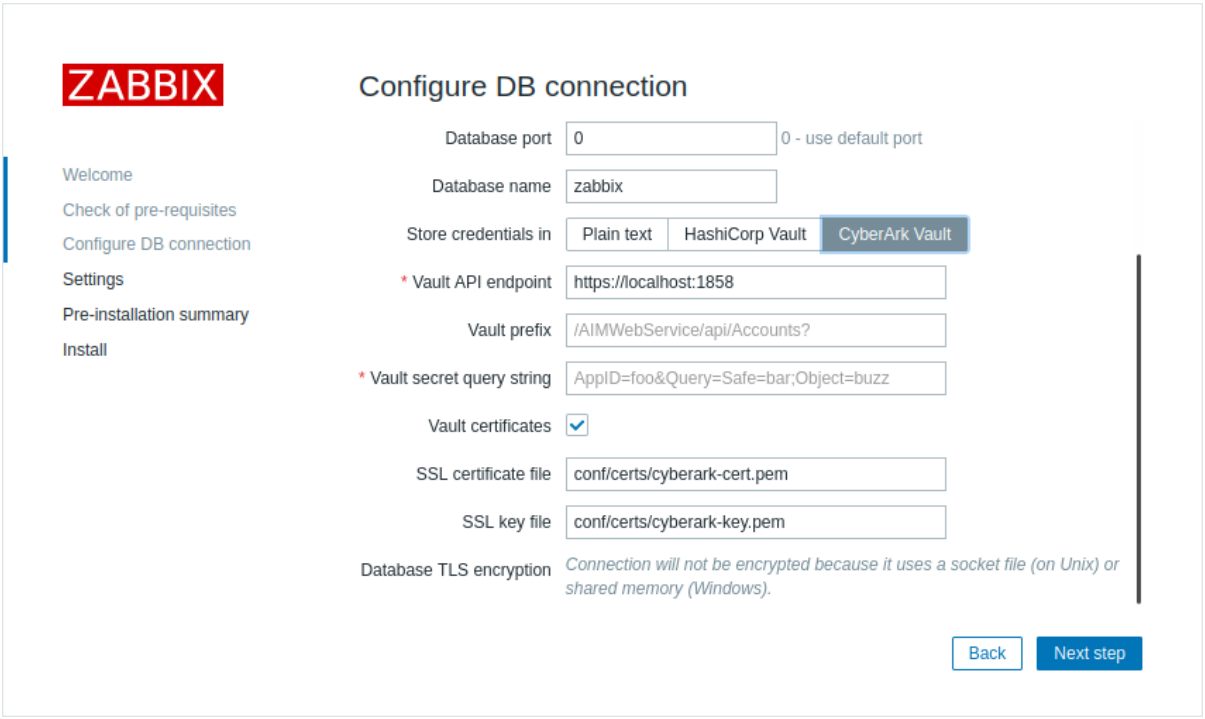
W przypadku PostgreSQL, jeśli pole *Database host* pozostanie puste, zostanie użyte domyślne gniazdo domeny Unix. Jeśli zostanie podana ścieżka do gniazda (na przykład `/var/run/pgbouncer`), zostanie użyte to gniazdo domeny Unix.



Jeśli zaznaczona jest opcja *Database TLS encryption*, w formularzu pojawią się dodatkowe pola do konfiguracji połączenia TLS z bazą danych (tylko MySQL lub PostgreSQL).

Jeśli opcja *Store credentials in* jest ustawiona na HashiCorp Vault lub CyberArk Vault, staną się dostępne dodatkowe parametry:

- dla HashiCorp Vault: punkt końcowy API Vault, prefiks vault, ścieżka sekretu oraz token uwierzytelniający;
- dla CyberArk Vault: punkt końcowy API Vault, prefiks vault, ciąg zapytania sekretu oraz certyfikaty. Po zaznaczeniu pola wyboru *Vault certificates* pojawią się dwa nowe pola do określenia ścieżek do pliku certyfikatu SSL i pliku klucza SSL.

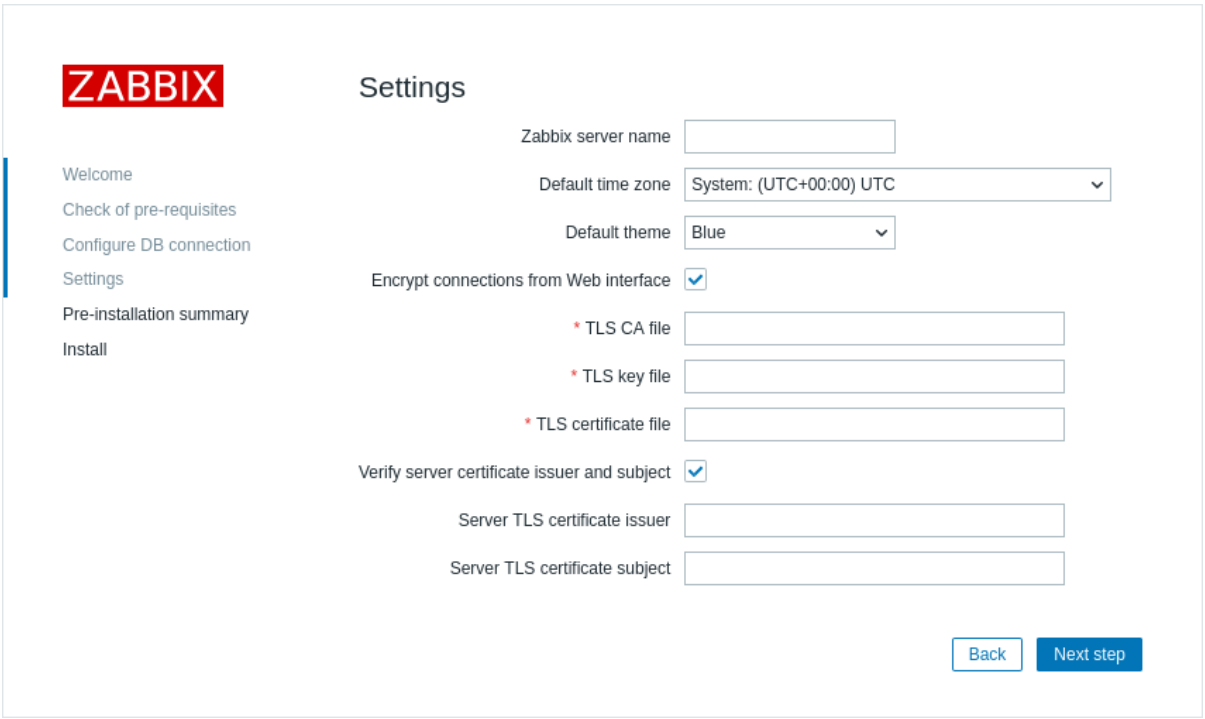


Ustawienia

Wprowadzenie nazwy serwera Zabbix jest opcjonalne, jednak jeśli zostanie podana, będzie wyświetlana na pasku menu i w tytułach stron.

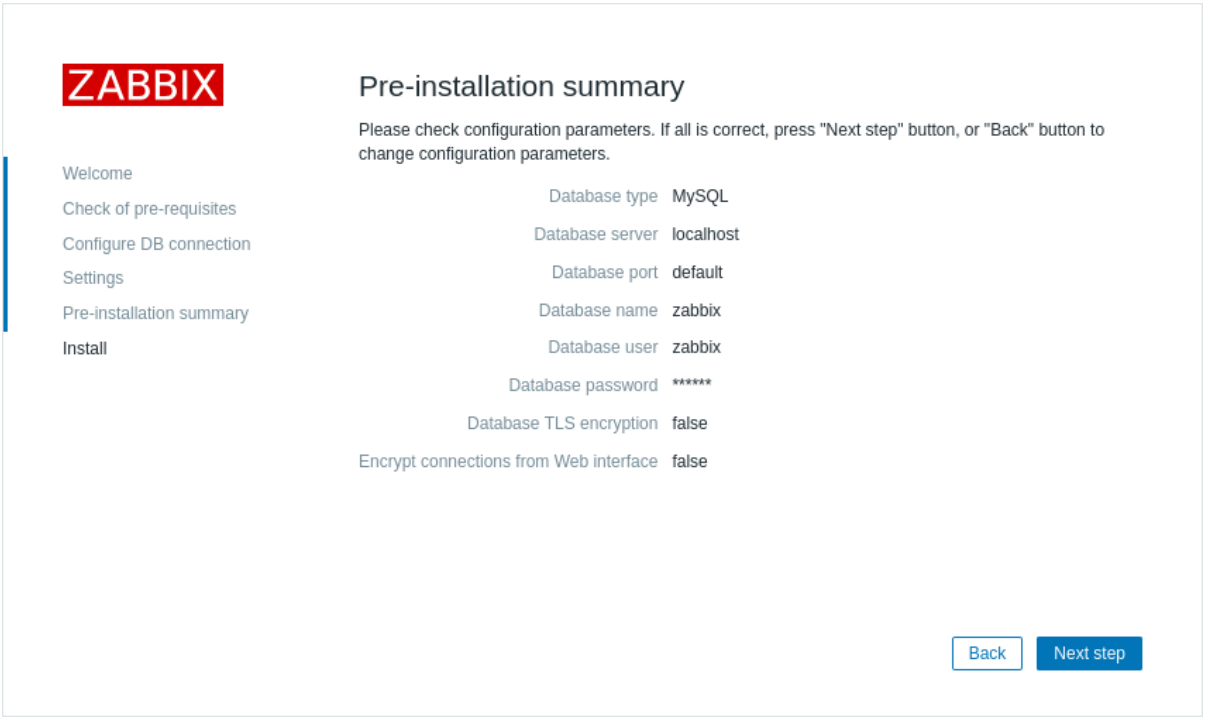
Ustaw domyślną strefę czasową i motyw dla frontend.

Jeśli zaznaczona jest opcja *Encrypt connections from Web interface*, w formularzu pojawią się dodatkowe pola do konfigurowania połączenia TLS między serwerem Zabbix a frontend.

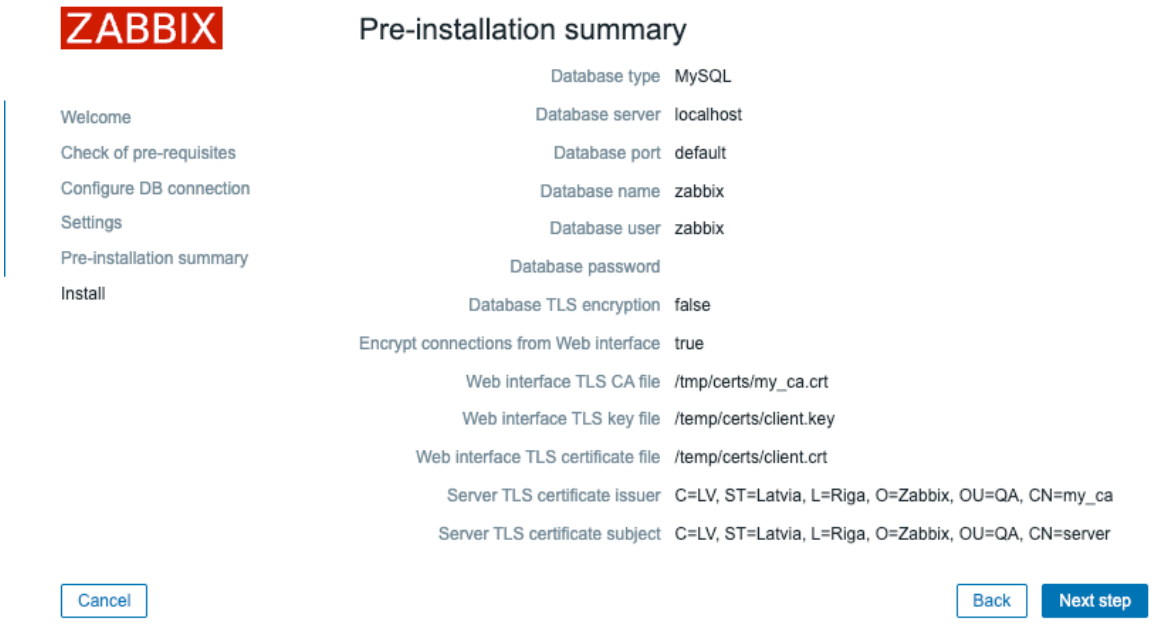


Podsumowanie przed instalacją

Przejrzyj podsumowanie ustawień.

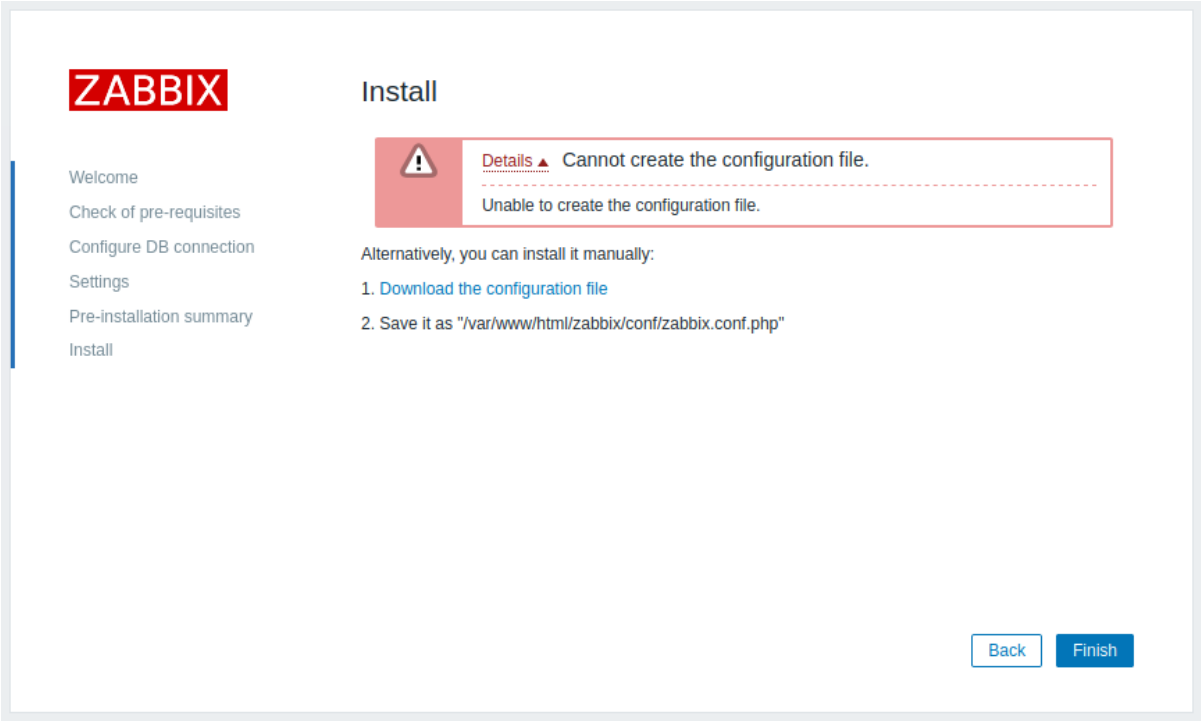


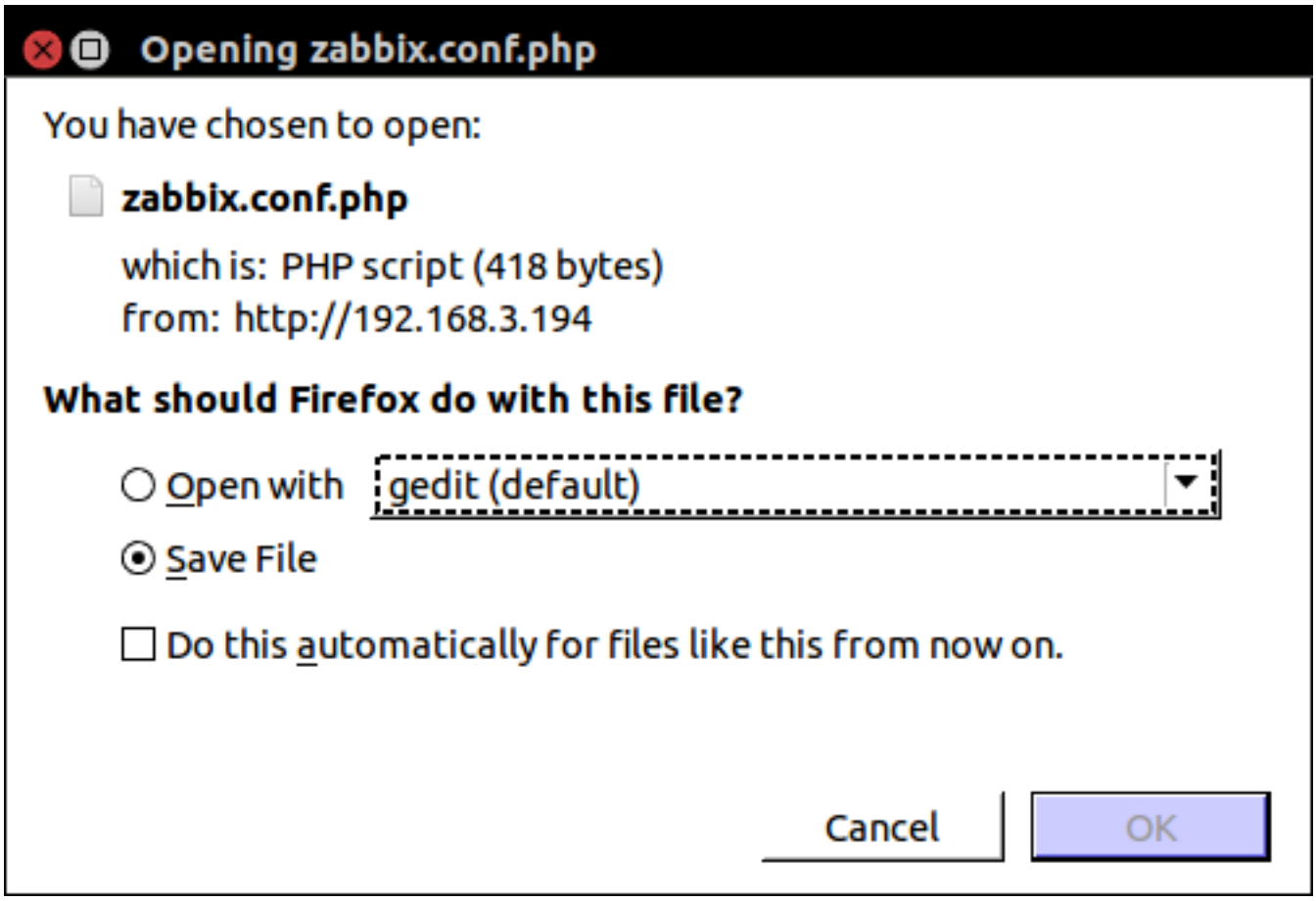
Podstrona wyświetli dane, jeśli konfiguracja TLS została dodana.



Instalacja

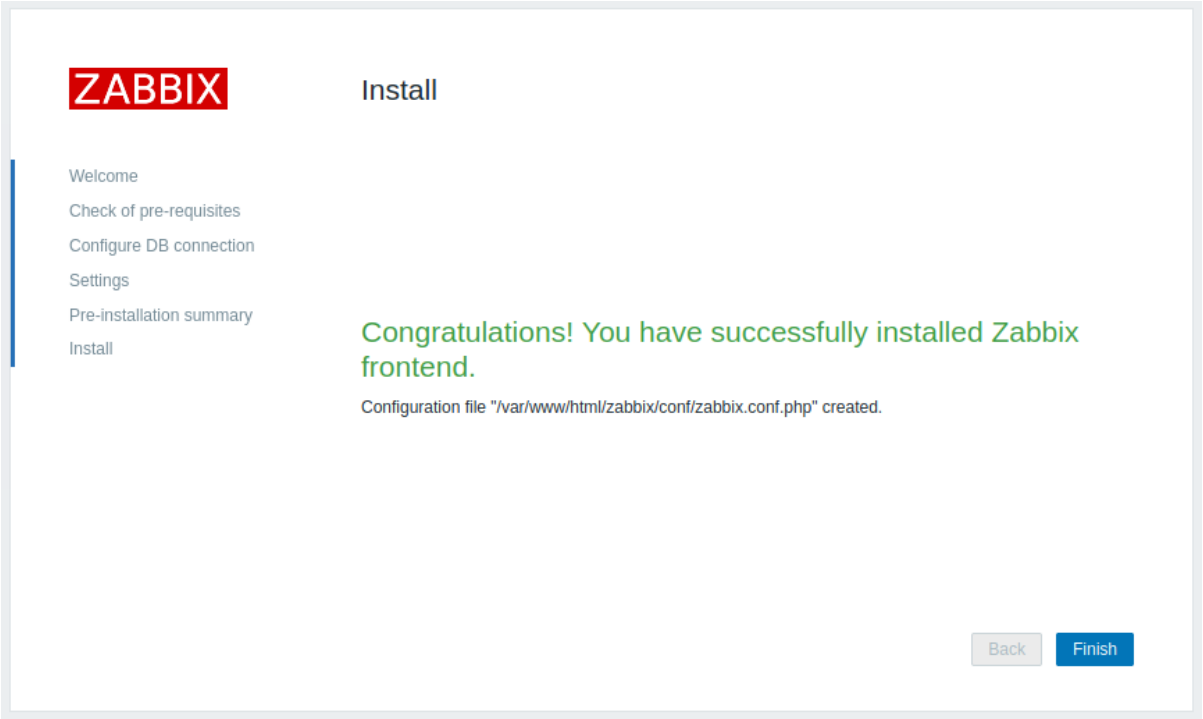
Jeśli instalujesz Zabbix ze źródeł, pobierz plik konfiguracyjny i umieść go w katalogu conf/ w podkatalogu dokumentów HTML serwera WWW, do którego skopiowano pliki PHP Zabbix.





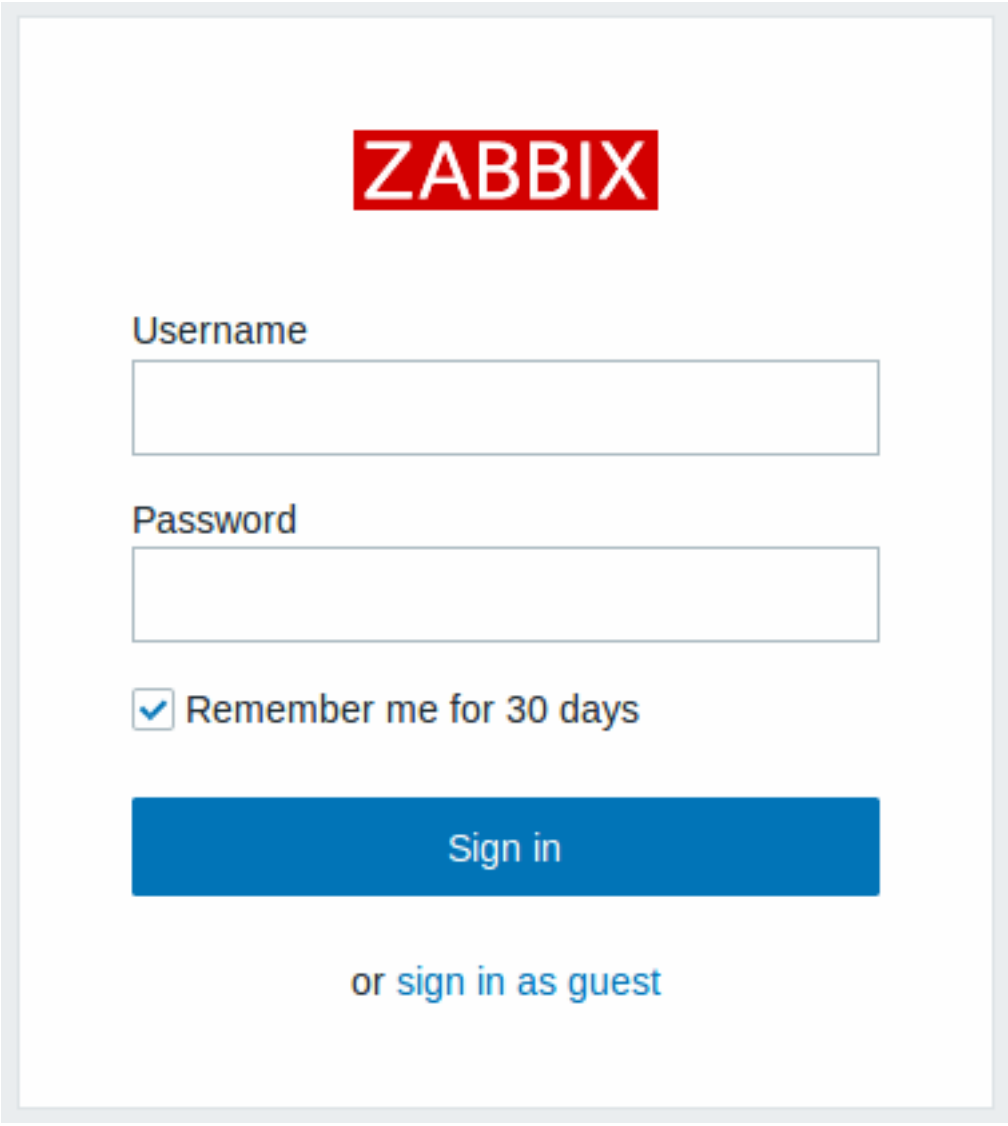
Note:
Jeśli użytkownik serwera WWW ma uprawnienia do zapisu w katalogu conf/, plik konfiguracyjny zostanie zapisany automatycznie i będzie można od razu przejść do następnego kroku.

Zakończ instalację.



Logowanie

Frontend Zabbixa jest gotowy! Domyślna nazwa użytkownika to **Admin**, hasło **zabbix**.



Przejdź do pierwszych kroków z Zabbixem.

7 Procedura aktualizacji

Omówienie

Ta sekcja zawiera informacje o aktualizacji Zabbix **7.4**:

- Przy użyciu pakietów dla Red Hat Enterprise Linux lub Debian/Ubuntu
- Przy użyciu źródeł
- Przy użyciu kontenerów

Powiązane instrukcje:

- Dla serwerów w klastrze wysokiej dostępności (HA) zobacz Aktualizacja klastra HA
- Dla bazy danych TimescaleDB zobacz Aktualizacja schematu TimescaleDB

Aktualizacja agentów Zabbix jest zalecana, ale nie jest obowiązkowa.

Aktualizacja proxy Zabbix jest wysoce zalecana. Serwer Zabbix w pełni obsługuje proxy w tej samej głównej wersji co serwer. Serwer Zabbix obsługuje również proxy, które nie są **starsze** niż wersja poprzedniego wydania LTS serwera Zabbix, ale z ograniczoną funkcjonalnością (zbieranie danych, wykonywanie zdalnych poleceń, natychmiastowe sprawdzanie wartości pozycji). Aktualizacja konfiguracji jest również wyłączona, a nieaktualne proxy będą działać tylko ze starą konfiguracją.

Attention:
Proxy starsze niż poprzednie wydanie LTS serwera Zabbix lub nowsze niż główna wersja serwera Zabbix nie są obsługiwane. Serwer Zabbix będzie ignorował dane z nieobsługiwanych proxy, a cała komunikacja z serwerem Zabbix zakończy się niepowodzeniem z ostrzeżeniem. Więcej informacji można znaleźć w sekcji Version compatibility.

Aby zminimalizować przestoje i utratę danych podczas aktualizacji, zaleca się zatrzymanie, zaktualizowanie i uruchomienie serwera Zabbix, a następnie zatrzymywanie, aktualizowanie i uruchamianie proxy kolejno, jeden po drugim. Podczas przestoju serwera działające proxy będą kontynuować zbieranie danych. Gdy serwer zostanie ponownie uruchomiony i będzie działał, nieaktualne proxy wyślą dane do nowszego serwera (konfiguracja proxy nie zostanie jednak zaktualizowana) i pozostaną częściowo funkcjonalne. Wszelkie powiadomienia o problemach podczas przestoju serwera Zabbix zostaną wygenerowane dopiero po uruchomieniu zaktualizowanego serwera.

Jeśli proxy Zabbix jest uruchamiany po raz pierwszy, a plik bazy danych SQLite jest nieobecny, proxy utworzy go automatycznie.

Uwaga: jeśli proxy Zabbix używa SQLite3 i podczas uruchamiania wykryje, że wersja istniejącego pliku bazy danych jest starsza niż wymagana, **usunie plik bazy danych automatycznie** i utworzy nowy. W związku z tym dane historyczne zapisane w pliku bazy danych SQLite zostaną utracone. Jeśli wersja proxy Zabbix jest starsza niż wersja pliku bazy danych, Zabbix zapisze błąd w dzienniku i zakończy działanie.

W zależności od rozmiaru bazy danych aktualizacja bazy danych do wersji 7.4 może zająć dużo czasu.

Bezpośrednia aktualizacja do Zabbix 7.4.x jest możliwa z Zabbix **7.2**.x, **7.0**.x, **6.4**.x, **6.2**.x, **6.0**.x, **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 oraz **2.0**.x. W przypadku aktualizacji z wcześniejszych wersji należy zapoznać się z dokumentacją Zabbix dla wersji 2.0 i wcześniejszych.

Note:
Należy pamiętać, że po aktualizacji niektóre integracje Zabbix z oprogramowaniem firm trzecich mogą zostać zakłócone, jeśli zewnętrzne oprogramowanie nie jest zgodne z zaktualizowaną wersją Zabbix.

Dostępne są następujące uwagi dotyczące aktualizacji:

Upgrade from	Read full upgrade notes	Most important changes between versions
7.2.x	For: Zabbix 7.4	Obsługa biblioteki PCRE została usunięta.
7.0.x LTS	For: Zabbix 7.2 Zabbix 7.4	Obsługa bazy danych Oracle została usunięta.

Upgrade from	Read full upgrade notes	Most important changes between versions
6.4.x	For: Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja PHP została podniesiona z 7.4.0 do 8.0.0. Asynchroniczne pollery dla sprawdzeń agent, HTTP agent, SNMP walk[OID] i SNMP get[OID]. Oddzielna tabela bazy danych dla proxy. Zmieniono domyślną lokalizację pliku konfiguracji agenta dla Windows. Baza danych Oracle została oznaczona jako przestarzała. Stary numeryczny (float) typ wartości został usunięty.
6.2.x	For: Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja MySQL została podniesiona z 8.0.0 do 8.0.30. Biblioteka 'libevent_pthreads' jest wymagana dla serwer/proxy Zabbix. Przy pierwszym uruchomieniu po aktualizacji proxy Zabbix z SQLite3 automatycznie usuwa starą wersję bazy danych (wraz z całą historią) i tworzy nową.
6.0.x LTS	For: Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja PHP została podniesiona z 7.2.5 do 7.4.0. Monitoring usług został znacząco przebudowany. Deterministyczne wyzwalacze muszą zostać utworzone podczas aktualizacji. Jeśli w MySQL/MariaDB włączone jest binarne logowanie, wymaga to uprawnień superużytkownika lub ustawienia parametru zmiennej/konfiguracji *log_bin_trust_function_creators = 1*. Instrukcje ustawiania tej zmiennej znajdują się w sekcji Skrypty tworzenia bazy danych.
5.4.x	For: Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Podniesiono minimalne wymagane wersje baz danych. Serwer/proxy nie uruchomi się w przypadku nieaktualnej bazy danych. Rekordy dziennika audytu zostały utracone z powodu zmiany struktury bazy danych.
5.2.x	For: Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Podniesiono minimalne wymagane wersje baz danych. Agregowane pozycje zostały usunięte jako osobny typ.
5.0.x LTS	For: Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja PHP została podniesiona z 7.2.0 do 7.2.5. Algorytm haszowania haseł został zmieniony z MD5 na bcrypt.
4.4.x	For: Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Obsługa IBM DB2 została usunięta. Minimalna wymagana wersja PHP została podniesiona z 5.4.0 do 7.2.0. Podniesiono minimalne wymagane wersje baz danych. Zmieniono katalog plików PHP Zabbix.

Upgrade from	Read full upgrade notes	Most important changes between versions
4.2.x	For: Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Usunięto typy mediów Jabber i Ez Texting.
4.0.x LTS	For: Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Starsze proxy nie mogą już raportować danych do zaktualizowanego serwer. Nowsze agenty nie będą już mogły współpracować ze starszym serwerem Zabbix.
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 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Biblioteki ’libpthread’ i ’zlib’ są teraz obowiązkowe. Obsługa protokołu tekstowego została usunięta, a nagłówek jest obowiązkowy. Agenty Zabbix w wersji sprzed 1.4 nie są już obsługiwane. Parametr Server w konfiguracji pasywnego proxy jest teraz obowiązkowy.
3.2.x	For: Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Obsługa SQLite jako bazy backendowej została usunięta dla serwer/frontend Zabbix. Obsługiwane są wyrażenia regularne zgodne z Perl Compatible Regular Expressions (PCRE) zamiast rozszerzonych POSIX. Biblioteki ’libpcre’ i ’libevent’ są obowiązkowe dla serwer Zabbix. Dodano sprawdzanie kodu wyjścia dla parametrów użytkownika, zdalnych poleceń i pozycji system.run[] bez flagi ’nowait’, a także dla skryptów wykonywanych przez serwer Zabbix. Java gateway Zabbix musi zostać zaktualizowany, aby obsługiwać nową funkcjonalność.

Upgrade from	Read full upgrade notes	Most important changes between versions
3.0.x LTS	For: 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 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Aktualizacja bazy danych może być powolna, w zależności od rozmiaru tabeli historii.
2.4.x	For: 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 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja PHP została podniesiona z 5.3.0 do 5.4.0. Parametr agenta LogFile musi być określony.
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 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Usunięto rozproszone monitorowanie oparte na węzłach.

Upgrade from	Read full upgrade notes	Most important changes between versions
2.0.x	For: 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 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4	Minimalna wymagana wersja PHP została podniesiona z 5.1.6 do 5.3.0. Do poprawnego działania serwer wymagany jest MySQL z uwzględnieniem wielkości liter w nazwach bazy danych; do prawidłowego działania serwer Zabbix z bazą danych MySQL wymagany jest zestaw znaków utf8 oraz sortowanie utf8_bin. Zobacz skrypty tworzenia bazy danych. Wymagane jest rozszerzenie PHP 'mysqli' zamiast 'mysql'.

1 Aktualizacja z kodu źródłowego

Omówienie

Ta sekcja zawiera kroki wymagane do pomyślnego upgrade z Zabbix **7.2**.x do Zabbix **7.4**.x przy użyciu oficjalnych źródeł Zabbix.

Warning:
Przed upgrade'em upewnij się, że zapoznałeś się z odpowiednimi **upgrade notes**!

Warto również sprawdzić requirements dla wersji 7.4.

Note:
Pomocne może być uruchomienie dwóch równoległych sesji SSH podczas upgrade'u: w jednej wykonywać kroki upgrade'u, a w drugiej monitorować logi serwera/proxy. Na przykład w drugiej sesji SSH uruchom `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log`, aby na bieżąco widzieć najnowsze wpisy w pliku logu oraz ewentualne błędy. Może to mieć kluczowe znaczenie w przypadku środowisk produkcyjnych.

Proces aktualizacji serwera

1 Zatrzymaj serwer

Zatrzymaj serwer Zabbix, aby upewnić się, że do bazy danych nie są wstawiane żadne nowe dane.

2 Utwórz kopię zapasową istniejącej bazy danych Zabbix

To bardzo ważny krok. Upewnij się, że masz kopię zapasową swojej bazy danych. Pomoże ona, jeśli procedura aktualizacji zakończy się niepowodzeniem (brak miejsca na dysku, zanik zasilania, jakikolwiek nieoczekiwany problem).

3 Utwórz kopię zapasową plików konfiguracyjnych, plików PHP i plików binarnych Zabbix

Utwórz kopię zapasową plików binarnych Zabbix, plików konfiguracyjnych oraz katalogu plików PHP.

4 Zainstaluj nowe binaria serwera

Użyj tych instructions , aby skompilować serwer Zabbix ze źródeł.

5 Przejrzyj parametry konfiguracji serwera

Upewnij się, że przejrzysz Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakieś zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

6 Uruchom nowe pliki binarne Zabbix

Uruchom nowe pliki binarne. Sprawdź pliki dziennika, aby zobaczyć, czy pliki binarne zostały uruchomione pomyślnie.

serwer Zabbix automatycznie zaktualizuje bazę danych. Podczas uruchamiania serwer Zabbix zgłasza bieżące (obowiązkową i opcjonalną) oraz wymaganą wersję bazy danych. Jeśli bieżąca wersja obowiązkowa jest starsza niż wymagana wersja, serwer Zabbix automatycznie wykona wymagane poprawki aktualizacji bazy danych. Rozpoczęcie i poziom postępu (procentowy) aktualizacji bazy danych są zapisywane w pliku dziennika serwera Zabbix. Po zakończeniu aktualizacji do pliku dziennika zostaje zapisany komunikat „database upgrade fully completed". Jeśli którakolwiek z poprawek aktualizacji zakończy się niepowodzeniem, serwer Zabbix nie uruchomi się. serwer Zabbix nie uruchomi się również, jeśli bieżąca obowiązkowa wersja bazy danych jest nowsza niż wymagana. serwer Zabbix uruchomi się tylko wtedy, gdy bieżąca obowiązkowa wersja bazy danych odpowiada wymaganej obowiązkowej wersji.

```
8673:20161117:104750.259 current database version (mandatory/optional): 03040000/03040000
8673:20161117:104750.259 required mandatory version: 03040000
```

Przed uruchomieniem serwera:

- Upewnij się, że użytkownik bazy danych ma wystarczające uprawnienia (create table, drop table, create index, drop index)
- Upewnij się, że masz wystarczającą ilość wolnego miejsca na dysku.

7 Zainstaluj nowy frontend Zabbix

Minimalna wymagana wersja PHP to 8.0.0. W razie potrzeby zaktualizuj ją i postępuj zgodnie z instrukcjami instalacji.

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby interfejs webowy Zabbix działał prawidłowo.

Proces aktualizacji proxy

1 Zatrzymaj proxy

Zatrzymaj Zabbix proxy.

2 Utwórz kopię zapasową plików konfiguracyjnych i plików binarnych Zabbix proxy

Utwórz kopię zapasową pliku binarnego Zabbix proxy oraz pliku konfiguracyjnego.

3 Zainstaluj nowe binaria proxy

Skorzystaj z tych instructions , aby skompilować proxy Zabbix ze źródeł.

4 Przejrzyj parametry konfiguracji proxy

Upewnij się, że zapoznałeś się z Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

5 Uruchom nowy Zabbix proxy

Uruchom nowy Zabbix proxy. Sprawdź pliki dziennika, aby zobaczyć, czy proxy uruchomił się pomyślnie.

Zabbix proxy automatycznie zaktualizuje bazę danych. Aktualizacja bazy danych przebiega podobnie jak podczas uruchamiania Zabbix server.

Proces aktualizacji agenta

> **Attention:**
> Aktualizacja agentów nie jest obowiązkowa. Należy zaktualizować agenty tylko wtedy, gdy jest to wymagane do uzyskania dostępu do nowych funkcji.

Procedura aktualizacji opisana w tej sekcji może być używana do aktualizacji zarówno agenta Zabbix, jak i agenta Zabbix 2.

1 Zatrzymaj agent

Zatrzymaj agent Zabbix.

2 Utwórz kopię zapasową plików konfiguracyjnych i plików binarnych agenta Zabbix

Utwórz kopię zapasową pliku binarnego agenta Zabbix oraz pliku konfiguracyjnego.

3 Zainstaluj nowe binaria agenta

Skorzystaj z tych instrukcji , aby skompilować agenta Zabbix ze źródeł.

Alternatywnie możesz pobrać wstępnie skompilowane agenty Zabbix ze strony pobierania Zabbix.

4 Przejrzyj parametry konfiguracji agenta

Upewnij się, że zapoznałeś się z Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

5 Uruchom nowy agent Zabbix

Uruchom nowy agent Zabbix. Sprawdź pliki dziennika, aby zobaczyć, czy agent został uruchomiony pomyślnie.

Aktualizacja między wersjami minor

Podczas aktualizacji między wersjami minor 7.4.x (na przykład z 7.4.1 do 7.4.3) należy wykonać te same działania dla serwer/proxy/agent, co podczas aktualizacji między wersjami major. Jedyna różnica polega na tym, że podczas aktualizacji między wersjami minor nie są wprowadzane żadne zmiany w bazie danych.

2 Aktualizacja z pakietów

Przegląd

Ta sekcja zawiera kroki wymagane do pomyślnego uaktualnienia przy użyciu oficjalnych pakietów RPM i DEB dostarczanych przez Zabbix dla:

- Red Hat Enterprise Linux
- Debian/Ubuntu

Pakiety Zabbix z repozytoriów systemu operacyjnego

Niektóre dystrybucje systemów operacyjnych (w szczególności dystrybucje oparte na Debianie) udostępniają własne pakiety Zabbix. Pakiety te **nie są wspierane przez Zabbix** i mogą być nieaktualne lub nie zawierać najnowszych funkcji oraz poprawek błędów. Zaleca się korzystanie wyłącznie z oficjalnych pakietów z Oficjalnego Repozytorium Zabbix.

Jeśli przeprowadzasz aktualizację z pakietów dostarczanych przez dystrybucje systemu operacyjnego (lub były one zainstalowane w pewnym momencie), wykonaj poniższą procedurę, aby przejść na oficjalne pakiety Zabbix:

1. Odinstaluj stare pakiety.
2. Sprawdź, czy po starej instalacji nie pozostały żadne pliki, i usuń je.
3. Zainstaluj oficjalne pakiety Zabbix zgodnie z instrukcjami instalacji udostępnionymi przez Zabbix.

Nie wykonuj bezpośredniej aktualizacji, ponieważ może to spowodować uszkodzenie instalacji.

1 Red Hat Enterprise Linux

Omówienie

Ta sekcja zawiera instrukcje dotyczące aktualizacji z Zabbix **7.2.x** do najnowszej wersji Zabbix **7.4.x** przy użyciu oficjalnych pakietów Zabbix dla Red Hat Enterprise Linux lub jego pochodnych - AlmaLinux, CentOS Stream, Oracle Linux i Rocky Linux.

Warning:
Przed rozpoczęciem aktualizacji zapoznaj się z odpowiednimi uwagami dotyczącymi aktualizacji i upewnij się, że Twój system spełnia wymagania dla Zabbix 7.4.

Note:
Rozważ uruchomienie dwóch równoległych sesji SSH podczas aktualizacji: jednej do wykonywania kroków aktualizacji i drugiej do monitorowania logów serwera/proxy. Na przykład w drugiej sesji uruchom `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log`, aby na bieżąco wyświetlać najnowsze wpisy w logach i ewentualne błędy. Może to mieć kluczowe znaczenie w przypadku środowisk produkcyjnych.

Instrukcje dotyczące aktualizacji między wersjami podrzędnymi Zabbix 7.4.x (na przykład z 7.4.1 do 7.4.3) znajdziesz w sekcji Aktualizacja między wersjami podrzędnymi.

Procedura aktualizacji

1 Zatrzymaj procesy Zabbix

Zatrzymaj serwer Zabbix, aby upewnić się, że żadne nowe dane nie są wstawiane do bazy danych:

```
systemctl stop zabbix-server
```

Jeśli aktualizujesz Zabbix proxy, agent lub agent 2, zatrzymaj również te komponenty:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Utwórz kopię zapasową bazy danych Zabbix

Utwórz kopię zapasową istniejącej bazy danych Zabbix, aby zabezpieczyć się przed niepowodzeniem aktualizacji (na przykład z powodu problemów z miejscem na dysku, utraty zasilania lub nieoczekiwanych problemów).

3 Utwórz kopię zapasową plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix

Utwórz kopię zapasową istniejących plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix.

Dla plików konfiguracyjnych uruchom:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/httpd/conf.d/zabbix.conf  /opt/zabbix-backup/
```

Dla plików PHP i plików binarnych Zabbix uruchom:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Zaktualizuj pakiet konfiguracji repozytorium

Przed kontynuowaniem aktualizacji zaktualizuj bieżący pakiet repozytorium do najnowszej wersji, aby zapewnić zgodność z najnowszymi pakietami oraz uwzględnić wszelkie ostatnie poprawki zabezpieczeń i błędy.

W przypadku **RHEL 10** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

W przypadku **RHEL 9** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

> **Note:**
> W przypadku starszych wersji RHEL lub ich pochodnych zastąp powyższy link właściwym linkiem z repozytorium Zabbix. Należy jednak pamiętać, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix, a aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę zawartych komponentów znajdziesz w sekcji Pakiety Zabbix.

Następnie wyczyść pamięć podręczną menedżera pakietów `dnf` (w tym nagłówki, metadane i pliki pakietów pobrane podczas poprzednich instalacji lub aktualizacji):

```
dnf clean all
```

Przy następnej operacji `dnf` pobierze świeże metadane z repozytoriów, ponieważ stare metadane zostały usunięte.

Zobacz także: Znane problemy dotyczące aktualizacji pakietu konfiguracji repozytorium w RHEL.

5 Aktualizacja komponentów Zabbix

Aby zaktualizować komponenty Zabbix, uruchom:

```
dnf install zabbix-server-mysql zabbix-web-mysql zabbix-agent
```

- Jeśli używasz PostgreSQL, w poleceniu zastąp `mysql` przez `pgsql`.
- Jeśli aktualizujesz proxy, w poleceniu zastąp `server` przez `proxy`.
- Jeśli aktualizujesz agent 2, zastąp `zabbix-agent` przez `zabbix-agent2 zabbix-agent2-plugin-*` w poleceniu.

> **Attention:**
> Aktualizacja Zabbix agent 2 za pomocą polecenia `dnf install zabbix-agent2` może prowadzić do błędu. Więcej informacji znajdziesz w *Znane problemy*.

Następnie, aby zaktualizować frontend Zabbix z Apache i ponownie uruchomić Apache, wykonaj:

```
dnf install zabbix-apache-conf
systemctl restart httpd
```

6 Przejrzyj parametry konfiguracji komponentu

Przejrzyj odpowiednie uwagi dotyczące aktualizacji, aby sprawdzić, czy konieczne są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

7 Uruchom procesy Zabbix

Uruchom zaktualizowane komponenty Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby interfejs webowy Zabbix działał prawidłowo.

Aktualizacja między wersjami minor

Możliwe jest przeprowadzenie aktualizacji między wersjami minor Zabbix 7.4.x (na przykład z 7.4.1 do 7.4.3).

Aby zaktualizować wszystkie komponenty Zabbix, uruchom:

```
dnf upgrade 'zabbix-*'
```

- Aby zaktualizować tylko Zabbix serwer, zastąp `'zabbix-*'` w poleceniu ciągiem `'zabbix-server-*'`.
- Aby zaktualizować tylko Zabbix proxy, zastąp `'zabbix-*'` w poleceniu ciągiem `'zabbix-proxy-*'`.
- Aby zaktualizować tylko Zabbix agent, zastąp `'zabbix-*'` w poleceniu ciągiem `'zabbix-agent-*'`.
- Aby zaktualizować tylko Zabbix agent 2, zastąp `'zabbix-*'` w poleceniu ciągiem `'zabbix-agent2-*'`.

2 Debian/Ubuntu

Przegląd

Ta sekcja zawiera instrukcje dotyczące aktualizacji z Zabbix **7.2.x** do najnowszej wersji Zabbix **7.4.x** przy użyciu oficjalnych pakietów Zabbix dla Debian/Ubuntu.

Warning:
Przed aktualizacją zapoznaj się z odpowiednimi uwagami dotyczącymi aktualizacji i upewnij się, że Twój system spełnia wymagania dla Zabbix 7.4.

Note:
Rozważ uruchomienie dwóch równoległych sesji SSH podczas aktualizacji: jednej do wykonywania kroków aktualizacji, a drugiej do monitorowania logów serwera/proxy. Na przykład uruchom `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log` w drugiej sesji, aby na bieżąco wyświetlać najnowsze wpisy w logach i możliwe błędy. Może to mieć kluczowe znaczenie w przypadku środowisk produkcyjnych.

Instrukcje dotyczące aktualizacji między wersjami podrzędnymi Zabbix 7.4.x (na przykład z 7.4.1 do 7.4.3) znajdziesz w sekcji Aktualizacja między wersjami podrzędnymi.

Procedura aktualizacji

1 Zatrzymaj procesy Zabbix

Zatrzymaj serwer Zabbix, aby upewnić się, że do bazy danych nie są wstawiane żadne nowe dane:

```
systemctl stop zabbix-server
```

Jeśli aktualizujesz Zabbix proxy, agent lub agent 2, zatrzymaj również te komponenty:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Utwórz kopię zapasową bazy danych Zabbix

Utwórz kopię zapasową istniejącej bazy danych Zabbix, aby zabezpieczyć się przed niepowodzeniem aktualizacji (na przykład z powodu braku miejsca na dysku, utraty zasilania lub nieoczekiwanych problemów).

3 Utwórz kopię zapasową plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix

Utwórz kopię zapasową istniejących plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix.

W przypadku plików konfiguracyjnych uruchom:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
```

W przypadku plików PHP i plików binarnych Zabbix uruchom:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Zaktualizuj pakiet konfiguracji repozytorium

Przed kontynuowaniem aktualizacji odinstaluj bieżący pakiet repozytorium Zabbix:

```
rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Może być również konieczne ręczne usunięcie starych pakietów Zabbix z katalogu roboczego (np. `rm zabbix-release_latest+debian12_` przed pobraniem nowego, aby zapobiec ponownemu użyciu przez menedżera pakietów nieaktualnej wersji podczas procesu aktualizacji.

Następnie zainstaluj najnowszy pakiet konfiguracji repozytorium, aby zapewnić zgodność z najnowszymi pakietami oraz uwzględnić wszelkie ostatnie poprawki bezpieczeństwa i błędy.

W przypadku **Debian 12** uruchom:

```
wget https://repo.zabbix.com/zabbix/7.4/release/debian/pool/main/z/zabbix-release/zabbix-release_latest+de
dpkg -i zabbix-release_latest+debian12_all.deb
```

> **Note:**
> W przypadku starszych wersji Debian zastąp powyższy link prawidłowym linkiem z repozytorium Zabbix. Należy jednak pamiętać, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix, a aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę zawartych komponentów znajdziesz w sekcji Pakiety Zabbix.

W przypadku **Ubuntu 24.04** uruchom:

```
wget https://repo.zabbix.com/zabbix/7.4/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu24.04_all.deb
```

W przypadku **Ubuntu 22.04** uruchom:

```
wget https://repo.zabbix.com/zabbix/7.4/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu22.04_all.deb
```

> **Note:**
> W przypadku starszych wersji Ubuntu zastąp powyższy link prawidłowym linkiem z repozytorium Zabbix. Należy jednak pamiętać, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix, a aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę zawartych komponentów znajdziesz w sekcji Pakiety Zabbix.

Może pojawić się monit dotyczący konfiguracji repozytorium Zabbix:

```
Configuration file '/etc/apt/sources.list.d/zabbix.list'
==> Deleted (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ?  Your options are:
Y or I  : install the package maintainer's version
N or O  : keep your currently-installed version
D       : show the differences between the versions
Z       : start a shell to examine the situation
The default action is to keep your current version.
*** zabbix.list (Y/I/N/O/D/Z) [default=N] ?
```

Wpisz `Y` (lub `I`), aby zainstalować wersję pliku konfiguracyjnego repozytorium Zabbix dostarczoną przez opiekuna pakietu.

Następnie zaktualizuj informacje o repozytorium:

```
apt update
```

5 Aktualizacja komponentów Zabbix

Aby zaktualizować komponenty Zabbix, uruchom:

```
apt install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

- Jeśli używasz PostgreSQL, zastąp `mysql` przez `pgsql` w poleceniu.
- Jeśli aktualizujesz proxy, zastąp `server` przez `proxy` w poleceniu.
- Jeśli aktualizujesz Zabbix agent 2, zastąp `zabbix-agent` przez `zabbix-agent2 zabbix-agent2-plugin-*` w poleceniu.

Attention:
Aktualizacja Zabbix agent 2 za pomocą polecenia `apt install zabbix-agent2` może prowadzić do błędu. Aby uzyskać więcej informacji, zobacz *Znane problemy*.

Może zostać wyświetlony monit dotyczący konfiguracji serwera Zabbix (lub proxy):

```
Configuration file '/etc/zabbix/zabbix_server.conf'
==> Modified (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ?  Your options are:
Y or I  : install the package maintainer's version
N or O  : keep your currently-installed version
D       : show the differences between the versions
Z       : start a shell to examine the situation
The default action is to keep your current version.
*** zabbix_server.conf (Y/I/N/O/D/Z) [default=N] ?
```

Wybierz opcję, która najlepiej pasuje do Twojej sytuacji. Na przykład wpisz `D`, aby porównać bieżącą i nową konfigurację, a następnie zdecyduj, czy chcesz zainstalować wersję dostarczoną przez opiekuna pakietu (`Y` lub `I`).

Następnie, aby zaktualizować frontend Zabbix z Apache i ponownie uruchomić Apache, wykonaj:

```
apt install zabbix-apache-conf
systemctl restart apache2
```

6 Przejrzyj parametry konfiguracji komponentu

Przejrzyj odpowiednie uwagi dotyczące aktualizacji, aby sprawdzić, czy konieczne są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

7 Uruchom procesy Zabbix

Uruchom zaktualizowane komponenty Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby interfejs webowy Zabbix działał prawidłowo.

Aktualizacja między wersjami minor

Możliwe jest przeprowadzenie aktualizacji między wersjami minor Zabbix 7.4.x (na przykład z 7.4.1 do 7.4.3).

Najpierw zaktualizuj informacje o repozytorium:

```
apt update
```

Następnie, aby zaktualizować wszystkie komponenty Zabbix, uruchom:

```
apt install --only-upgrade 'zabbix*'
```

- Aby zaktualizować tylko serwer Zabbix, zastąp `'zabbix*'` w poleceniu przez `'zabbix-server*'`.
- Aby zaktualizować tylko proxy Zabbix, zastąp `'zabbix*'` w poleceniu przez `'zabbix-proxy*'`.
- Aby zaktualizować tylko agent Zabbix, zastąp `'zabbix*'` w poleceniu przez `'zabbix-agent*'`.
- Aby zaktualizować tylko agent Zabbix 2, zastąp `'zabbix*'` w poleceniu przez `'zabbix-agent2*'`.

3 Aktualizacja z kontenerów

Omówienie

Ta sekcja opisuje kroki wymagane do pomyślnego uaktualnienia kontenerów Zabbix **7.4**.x.

Dostępne są oddzielne zestawy instrukcji dla:

- uaktualnienia obrazu komponentu Zabbix
- uaktualnienia pliku Docker Compose

Warning:
Przed uaktualnieniem upewnij się, że zapoznałeś się z odpowiednimi **uwagami dotyczącymi uaktualnienia**!

Attention:
Przed rozpoczęciem uaktualnienia sprawdź, czy użytkownicy mają odpowiednie uprawnienia do bazy danych na potrzeby uaktualnienia.

 W przypadku uaktualnień z Zabbix 6.0 lub starszego podczas uaktualnienia będzie trzeba utworzyć deterministyczne wyzwalacze. Jeśli dla MySQL/MariaDB włączone jest binarne logowanie, wymaga to uprawnień superużytkownika lub ustawienia parametru zmiennej/konfiguracji `log_bin_trust_function_creators = 1`. Instrukcje dotyczące ustawiania tej zmiennej znajdują się w sekcji Skrypty tworzenia bazy danych.

 Należy pamiętać, że jeśli polecenie jest wykonywane z konsoli, zmienna zostanie ustawiona tylko tymczasowo i zostanie usunięta po ponownym uruchomieniu Dockera. W takim przypadku pozostaw uruchomioną usługę SQL, a zatrzymaj tylko usługę `zabbix-server`, wykonując `docker compose down zabbix-server`, a następnie `docker compose up -d zabbix-server`.

 Alternatywnie można ustawić tę zmienną w pliku konfiguracyjnym.

W zależności od rozmiaru bazy danych uaktualnienie bazy danych do wersji 7.4 może zająć dużo czasu.

Aktualizacja obrazu komponentu Zabbix

Te instrukcje służą do aktualizacji obrazów Docker używanych jako baza dla kontenerów komponentów Zabbix.

Poniższe kroki wykorzystują proxy Zabbix jako przykład. Zastąp `zabbix-proxy-sqlite3` nazwą kontenera swojego komponentu.

1. Sprawdź wszystkie kontenery, aby zidentyfikować ich nazwy i bieżące wersje obrazów:

```
docker ps -a

#### CONTAINER ID   IMAGE                                          (...) NAMES
#### 5a0c2e3d02b4   zabbix/zabbix-proxy-sqlite3:alpine-7.2-latest   (...) zabbix-proxy-sqlite3
```

2. Pobierz nową wersję obrazu:

```
docker pull zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest
```

3. Zatrzymaj kontener komponentu Zabbix:

```
docker stop zabbix-proxy-sqlite3
```

4. Usuń kontener komponentu Zabbix:

Attention:
Upewnij się, że przed kontynuowaniem zanotowano wszystkie flagi konfiguracji oraz montowania woluminów użyte w oryginalnym poleceniu `docker run`, ponieważ będą one potrzebne do odtworzenia kontenera.

```
docker rm zabbix-proxy-sqlite3
```

5. Uruchom nowy kontener, używając zaktualizowanego obrazu:

```
docker run --name zabbix-proxy-sqlite3 \
  -e ZBX_SERVER_HOST=192.0.2.1 \
```

```
  -e ZBX_PROXYMODE=0 \
  -e ZBX_HOSTNAME=zabbix-proxy-sqlite3 \
  -v zabbix-proxy-data:/var/lib/zabbix/db_data \
  --init \
  -d \
  zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest
```

Note:
Więcej informacji na temat uruchamiania kontenerów Zabbix można znaleźć w sekcji Docker (manualne wdrożenie).

6. Sprawdź, czy aktualizacja zakończyła się powodzeniem, analizując logi kontenera:

```
docker logs -f zabbix-proxy-sqlite3
```

Aktualizacja pliku Docker Compose

Te instrukcje służą do aktualizacji plików Docker Compose używanych do wdrażania komponentów Zabbix. Postępuj zgodnie z tymi instrukcjami, jeśli zainstalowałeś Zabbix przy użyciu Docker Compose.

1. Sprawdź wszystkie kontenery, aby zidentyfikować ich nazwy i bieżące wersje obrazów:

```
docker ps -a

#### CONTAINER ID   IMAGE                                          (...) NAMES
#### d095f486deef   zabbix/zabbix-server-mysql:alpine-7.2-latest    (...) zabbix-docker-zabbix-server-1
#### 70627d833da0   zabbix/zabbix-web-nginx-mysql:alpine-7.2-latest (...) zabbix-docker-zabbix-web-ngina
#### aaaa664cc185   zabbix/zabbix-server-mysql:alpine-7.2-latest    (...) zabbix-docker-server-db-init-1
#### 9ab3e77492ff   mysql:8.0-oracle                               (...) zabbix-docker-mysql-server-1
```

2. Przejdź do repozytorium Zabbix Docker, które wcześniej sklonowałeś, pobierz najnowsze aktualizacje i przełącz się na gałąź 7.4:

Attention:
Jeśli zmodyfikowałeś plik `.env`, `compose.yaml` lub inne pliki w lokalnym repozytorium, polecenie `git pull` może je nadpisać. Przed wykonaniem tego kroku wykonaj kopię zapasową wszystkich lokalnych zmian.

```
cd zabbix-docker
git pull
git checkout 7.4
```

3. Wdróż komponenty Zabbix, używając zaktualizowanego pliku Docker Compose:

```
#### Z MySQL jako bazą danych:
docker compose -f ./compose.yaml up -d

#### Z PostgreSQL jako bazą danych:
docker compose -f ./compose_pgsql.yaml up -d
```

Note:
Po ponownym utworzeniu i uruchomieniu wszystkich kontenerów wznowienie zbierania danych może potrwać 1—3 minuty.
Więcej informacji o uruchamianiu kontenerów Zabbix znajdziesz w sekcji Docker Compose.

4. Sprawdź, czy aktualizacja zakończyła się pomyślnie, analizując logi kontenera (zastąp `zabbix-docker-zabbix-server-1` nazwą kontenera swojego komponentu):

```
docker logs -f zabbix-docker-zabbix-server-1
```

8 Znane problemy

Zobacz także: Problemy z kompilacją.

Znane problemy w 7.4.4

Nie zaleca się aktualizacji do tej wersji z powodu:

- nagłego skoku użycia CPU, jeśli korzystasz z wtyczki MySQL w Zabbix agent 2 (zobacz ZBX-27156)
- wykresu dla pozycji Zabbix active agent wyświetlającego ostrzeżenie "Undefined array key" z powodu błędu niezdefiniowanego indeksu (zobacz ZBX-27153)

Aktualizacja

Ustawienie trybu SQL dla pomyślnej aktualizacji

Ustawienie `sql_mode` w MySQL/MariaDB musi mieć ustawiony tryb "STRICT_TRANS_TABLES". Jeśli go brakuje, aktualizacja bazy danych Zabbix zakończy się niepowodzeniem (zobacz także ZBX-19435).

Aktualizacja z MariaDB 10.2.1 i wcześniejszą

Aktualizacja Zabbix może się nie powieść, jeśli tabele bazy danych zostały utworzone przy użyciu MariaDB 10.2.1 lub wcześniejszej, ponieważ w tych wersjach domyślny format wiersza to compact. Można to naprawić, zmieniając format wiersza na dynamic (zobacz także ZBX-17690).

Szablony

Zgodność szablonów w środowiskach dual-stack (IPv4/IPv6)

W środowiskach dual-stack (systemach skonfigurowanych do obsługi zarówno IPv4, jak i IPv6) nazwa hosta `localhost` zazwyczaj rozwiązuje się do adresów IPv4 i IPv6. Ze względu na częste nadawanie priorytetu IPv6 przed IPv4 przez wiele systemów operacyjnych i resolverów DNS, szablony Zabbix mogą nie działać poprawnie, jeśli monitorowana usługa jest skonfigurowana do nasłuchiwania wyłącznie na IPv4.

Usługi, które nie są skonfigurowane do nasłuchiwania na adresach IPv6, mogą stać się niedostępne, co prowadzi do niepowodzeń monitorowania. Użytkownicy mogą skonfigurować dostęp poprawnie dla IPv4, a mimo to nadal napotykać problemy z łącznością z powodu domyślnego zachowania polegającego na priorytetyzacji IPv6.

Obejściem tego problemu jest upewnienie się, że usługi (Nginx, Apache, PostgreSQL itp.) są skonfigurowane do nasłuchiwania zarówno na adresach IPv4, jak i IPv6, oraz że serwer/agent Zabbix ma przyznany dostęp przez IPv6. Dodatkowo w szablonach i konfiguracjach Zabbix należy używać jawnie `localhost` zamiast `127.0.0.1`, aby zapewnić zgodność zarówno z IPv4, jak i IPv6.

Na przykład, podczas monitorowania PostgreSQL za pomocą szablonu PostgreSQL by Zabbix agent 2 może być konieczna edycja pliku `pg_hba.conf`, aby zezwolić na połączenia dla użytkownika `zbx_monitor`. Jeśli w środowisku dual-stack priorytet ma IPv6 (system rozwiązuje `localhost` do `::1`) i skonfigurujesz `localhost`, ale dodasz tylko wpis IPv4 (`127.0.0.1/32`), połączenie nie powiedzie się, ponieważ nie ma pasującego wpisu IPv6.

Poniższy przykład pliku `pg_hba.conf` zapewnia, że użytkownik `zbx_monitor` może łączyć się z dowolną bazą danych z lokalnej maszyny przy użyciu zarówno adresów IPv4, jak i IPv6, z różnymi metodami uwierzytelniania:

```
### TYPE     DATABASE     USER            ADDRESS          METHOD
  host     all          zbx_monitor     localhost        trust
  host     all          zbx_monitor     127.0.0.1/32     md5
  host     all          zbx_monitor     ::1/128          scram-sha-256
```

W razie potrzeby można również użyć bezpośrednio adresu IPv4 (`127.0.0.1`) podczas konfigurowania makra szablonu PostgreSQL by Zabbix agent 2 dla ciągu połączenia.

Przypadkowa instalacja pakietów Zabbix z EPEL

Jeśli repozytorium EPEL jest zainstalowane i włączone, instalacja pakietów Zabbix może spowodować pobranie wersji z EPEL zamiast oficjalnych pakietów Zabbix. Aby rozwiązać ten problem:

1. Usuń wszystkie pakiety Zabbix zainstalowane z EPEL:

```
dnf remove zabbix-server-mysql
```

2. Wyklucz pakiety Zabbix z EPEL, dodając następujący wiersz do pliku `/etc/yum.repos.d/epel.repo`:

```
[epel]
...
excludepkgs=zabbix*
```

3. Zainstaluj ponownie oficjalny pakiet serwera Zabbix:

```
dnf install zabbix-server-mysql
```

Podczas instalacji oficjalne pakiety Zabbix zawierają słowo `release` w ciągu wersji (na przykład `7.0.0-release1.el8`), co odróżnia je od pakietów EPEL.

Pakiety Zabbix dla RHEL w środowiskach Red Hat UBI

Podczas instalacji Zabbix z pakietów Red Hat Enterprise Linux w środowiskach Red Hat Universal Base Image należy zapewnić dostęp do wymaganych repozytoriów i zależności. Pakiety Zabbix zależą od bibliotek `libOpenIPMI.so` i `libOpenIPMIposix.so`, które nie są dostarczane przez żaden pakiet w domyślnych repozytoriach menedżera pakietów włączonych w systemach UBI, co spowoduje niepowodzenie instalacji.

Biblioteki `libOpenIPMI.so` i `libOpenIPMIposix.so` są dostępne w pakiecie `OpenIPMI-libs`, który jest udostępniany przez repozytorium `redhat-#-for-<arch>-appstream-rpms`. Dostęp do tego repozytorium jest regulowany przez subskrypcje, które w przypadku środowisk UBI są propagowane przez montowanie katalogów konfiguracji repozytoriów i sekretów hosta RHEL do przestrzeni nazw systemu plików kontenera.

Aby uzyskać więcej informacji, zobacz ZBX-24291.

Wygasły klucz podpisu dla pakietów RHEL

Podczas aktualizacji Zabbixa na Red Hat Enterprise Linux lub jego pochodnych może wystąpić problem z wygasłym kluczem podpisu dla pakietów w repozytorium Zabbixa. Gdy klucz podpisu wygasa, próby weryfikacji podpisów pakietów zakończą się błędem wskazującym, że certyfikat lub klucz nie są już ważne. Na przykład:

```
error: Verifying a signature using certificate D9AA84C2B617479C6E4FCF4D19F2475308EFA7DD (Zabbix LLC (Jul 2
  1. Certificate 19F2475308EFA7DD invalid: certificate is not alive
      because: The primary key is not live
      because: Expired on 2024-07-04T11:41:23Z
  2. Key 19F2475308EFA7DD invalid: key is not alive
      because: The primary key is not live
      because: Expired on 2024-07-04T11:41:23Z
```

Aby rozwiązać ten problem, ręcznie zainstaluj ponownie najnowszy pakiet `zabbix-release` dla swojej konkretnej wersji RHEL (zamień poniższy link na właściwy z repozytorium Zabbixa).

Na przykład w przypadku **RHEL 9** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

Następnie zaktualizuj informacje o repozytorium:

```
dnf update
```

Więcej informacji można znaleźć w ZBX-24761.

Timescale DB: wysokie użycie pamięci przy dużej liczbie partycji

Wersje PostgreSQL 9.6-12 zużywają zbyt dużo pamięci podczas aktualizacji tabel z dużą liczbą partycji. Problem ten ujawnia się, gdy Zabbix aktualizuje trendy w systemach z TimescaleDB, jeśli trendy są podzielone na relatywnie małe fragmenty (np. 1 dzień). Prowadzi to do obecności setek fragmentów w tabelach trendów przy domyślnych ustawieniach housekeeping — jest to sytuacja, w której PostgreSQL może najprawdopodobniej wyczerpać pamięć.

Problem został rozwiązany od Zabbix 5.0.1 dla nowych instalacji z TimescaleDB, ale jeśli TimescaleDB skonfigurowano z Zabbix wcześniej, zapoznaj się z ZBX-16347, aby sprawdzić uwagi dotyczące migracji.

Timescale DB 2.5.0: polityka kompresji może zakończyć się niepowodzeniem w tabelach zawierających liczby całkowite

Problem ten występuje podczas używania TimescaleDB 2.5.0/2.5.1. Został on rozwiązany od TimescaleDB 2.5.2.

Aby uzyskać więcej informacji, zobacz TimescaleDB Issue #3773.

Połączenie TLS z bazą danych przy użyciu MariaDB

Połączenie TLS z bazą danych nie jest obsługiwane z opcją 'verify_ca' dla parametru DBTLSConnect, jeśli używana jest MariaDB.

Możliwe zakleszczenia z MySQL/MariaDB

Podczas pracy pod dużym obciążeniem, gdy zaangażowany jest więcej niż jeden worker LLD, może dojść do zakleszczenia spowodowanego błędem InnoDB związanym ze strategią blokowania wierszy (zobacz błąd upstream). Błąd został naprawiony w MySQL od wersji 8.0.29, ale nie w MariaDB. Więcej informacji można znaleźć w ZBX-21506.

Globalna korelacja zdarzeń

Zdarzenia mogą nie zostać skorelowane poprawnie, jeśli odstęp czasu między pierwszym a drugim zdarzeniem jest bardzo mały, tj. pół sekundy lub mniej.

Zakres typu danych Numeric (float) w PostgreSQL 11 i wcześniejszych

PostgreSQL 11 i wcześniejsze wersje obsługują tylko zakres wartości zmiennoprzecinkowych w przybliżeniu od -1.34E-154 do 1.34E+154.

NetBSD 8.0 i nowsze

Różne procesy Zabbix mogą losowo ulegać awarii podczas uruchamiania w wersjach NetBSD 8.X i 9.X. Jest to spowodowane zbyt małym domyślnym rozmiarem stosu (4 MB), który należy zwiększyć, uruchamiając:

```
ulimit -s 10240
```

Aby uzyskać więcej informacji, zobacz powiązane zgłoszenie problemu: ZBX-18275.

Ograniczenia wyrażeń regularnych w Zabbix agent 2

Zabbix agent 2 nie obsługuje asercji lookahead i lookbehind w wyrażeniach regularnych ze względu na ograniczenia standardowej biblioteki Go regexp.

Kontrole IPMI

Kontrole IPMI nie będą działać ze standardowym pakietem biblioteki OpenIPMI w Debianie wcześniejszym niż 9 (stretch) oraz Ubuntu wcześniejszym niż 16.04 (xenial). Aby to naprawić, należy ponownie skompilować bibliotekę OpenIPMI z włączoną obsługą OpenSSL, jak omówiono w ZBX-6139.

IPMI — niezaufane hosty mogą spowodować awarię OpenIPMI

W bibliotece OpenIPMI używanej przez Zabbix do odpytywania danych IPMI występuje błąd, który może zostać wywołany przez specjalnie spreparowane odpowiedzi z niezaufanego urządzenia.
Niezaufane urządzenie IPMI może wysłać spreparowane dane, które spowodują awarię biblioteki OpenIPMI, co z kolei może doprowadzić do zakończenia procesu serwera Zabbix wykonującego odpytywanie IPMI.

Kontrole SSH

- Niektóre dystrybucje Linuksa, takie jak Debian i Ubuntu, nie obsługują zaszyfrowanych kluczy prywatnych (z hasłem) w przypadku, gdy biblioteka libssh2 jest zainstalowana z pakietów. Więcej informacji znajduje się w ZBX-4850.
- W przypadku używania libssh 0.9.x w niektórych dystrybucjach Linuksa z OpenSSH 8, kontrole SSH mogą sporadycznie zgłaszać błąd „Cannot read data from SSH server”. Jest to spowodowane problemem w libssh (bardziej szczegółowy raport). Oczekuje się, że błąd został naprawiony w stabilnym wydaniu libssh 0.9.5. Szczegóły znajdują się także w ZBX-17756.
- Użycie znaku potoku „|” w skrypcie SSH może prowadzić do błędu „Cannot read data from SSH server”. W takim przypadku zaleca się aktualizację wersji biblioteki libssh. Szczegóły znajdują się także w ZBX-21337.

Kontrole ODBC

- Sterownik MySQL unixODBC nie powinien być używany z serwer Zabbix lub proxy Zabbix skompilowanymi z biblioteką łącznika MariaDB i odwrotnie; jeśli to możliwe, lepiej również unikać używania tego samego łącznika co sterownika ze względu na błąd po stronie projektu upstream. Zalecana konfiguracja:

  Łącznik PostgreSQL, SQLite lub Oracle → sterownik MariaDB lub MySQL unixODBC
Łącznik MariaDB → sterownik MariaDB unixODBC
Łącznik MySQL → sterownik MySQL unixODBC

  Więcej informacji i dostępne obejścia można znaleźć w ZBX-7665.

- Dane XML pobierane z Microsoft SQL Server mogą być obcinane na różne sposoby w systemach Linux i UNIX.
- Zaobserwowano, że używanie kontroli ODBC do monitorowania baz danych Oracle przy użyciu różnych wersji Oracle Instant Client dla systemu Linux powoduje awarie serwer Zabbix.
 Zobacz także: ZBX-18402, ZBX-20803.
- W przypadku używania sterownika FreeTDS UnixODBC należy dodać na początku zapytania SQL instrukcję ’SET NOCOUNT ON’ (na przykład `SET NOCOUNT ON DECLARE @strsql NVARCHAR(max) SET @strsql = ....`). W przeciwnym razie pozycja monitorowania bazy danych w Zabbix nie będzie w stanie pobrać informacji i zwróci błąd ”SQL query returned empty result”.
 Więcej informacji można znaleźć w ZBX-19917.

Nieprawidłowy parametr metody żądania w pozycjach

Parametr metody żądania, używany wyłącznie w kontrolach HTTP, może być nieprawidłowo ustawiony na „1”, czyli wartość inną niż domyślna dla wszystkich pozycji, w wyniku aktualizacji z wersji Zabbix starszej niż 4.0. Szczegółowe informacje o tym, jak naprawić tę sytuację, można znaleźć w ZBX-19308.

Monitorowanie WWW i agent HTTP

Serwer Zabbix powoduje wyciek pamięci w niektórych dystrybucjach Linuksa z powodu błędu upstream, gdy opcja „SSL verify peer” jest włączona w scenariuszach WWW lub w agencie HTTP. Więcej informacji oraz dostępne obejścia problemu można znaleźć w ZBX-10486.

Proste testy

W wersjach **fping** wcześniejszych niż v3.10 występuje błąd powodujący nieprawidłową obsługę zduplikowanych pakietów odpowiedzi echo. Może to powodować nieoczekiwane wyniki dla pozycji `icmpping`, `icmppingloss`, `icmppingsec`. Zaleca się używanie najnowszej wersji **fping**. Więcej szczegółów można znaleźć w ZBX-11726.

Błędy wykonywania fping w kontenerach rootless

Gdy kontenery działają w trybie rootless lub w środowisku z określonymi ograniczeniami, podczas wykonywania kontroli ICMP mogą wystąpić błędy związane z uruchamianiem fping, takie jak `fping: Operation not permitted` lub utrata wszystkich pakietów do wszystkich zasobów.

Aby rozwiązać ten problem, dodaj `--cap-add=net_raw` do poleceń "docker run" lub "podman run".

Dodatkowo uruchamianie fping w środowiskach bez uprawnień roota może wymagać modyfikacji sysctl, tj.:

```
sudo sysctl -w "net.ipv4.ping_group_range=0 1995"
```

gdzie "1995" to GID użytkownika zabbix. Więcej informacji można znaleźć w ZBX-22833.

Kontrole SNMP

Jeśli używany jest system operacyjny OpenBSD, błąd use-after-free w bibliotece Net-SNMP do wersji 5.7.3 może spowodować awarię serwera Zabbix, jeśli parametr SourceIP jest ustawiony w pliku konfiguracyjnym serwera Zabbix. Jako obejście problemu nie należy ustawiać parametru SourceIP. Ten sam problem dotyczy również systemu Linux, ale nie powoduje zatrzymania pracy serwera Zabbix. Lokalna poprawka dla pakietu net-snmp w OpenBSD została zastosowana i zostanie wydana wraz z OpenBSD 6.3.

Skoki danych SNMP

Zaobserwowano skoki danych SNMP, które mogą być związane z pewnymi czynnikami fizycznymi, takimi jak skoki napięcia w sieci elektrycznej. Więcej szczegółów znajduje się w ZBX-14318.

Pułapki SNMP

Pakiet "net-snmp-perl", wymagany do obsługi pułapek SNMP, został usunięty w RHEL 8.0-8.2; ponownie dodany w RHEL 8.3.

Jeśli więc używasz RHEL 8.0-8.2, najlepszym rozwiązaniem jest aktualizacja do RHEL 8.3.

Aby uzyskać więcej informacji, zobacz także ZBX-17192.

Awaria procesu alerter w RHEL 7

W RHEL 7 odnotowano przypadki awarii procesu alerter serwera Zabbix. Szczegółowe informacje znajdują się w ZBX-10461.

Aktualizacja Zabbix agent 2 (6.0.5 lub starszy)

Podczas aktualizacji Zabbix agent 2 (wersja 6.0.5 lub starsza) z pakietów może wystąpić błąd konfliktu plików związany z wtyczkami. Aby naprawić ten błąd, wykonaj kopię zapasową konfiguracji agent 2 (jeśli to konieczne), odinstaluj agent 2 i zainstaluj go ponownie.

W systemach opartych na RHEL uruchom:

```
dnf remove zabbix-agent2
dnf install zabbix-agent2
```

W systemach opartych na Debian uruchom:

```
apt remove zabbix-agent2
apt install zabbix-agent2
```

Aby uzyskać więcej informacji, zobacz ZBX-23250.

Przełączanie ustawień regionalnych frontend

Zaobserwowano, że ustawienia regionalne frontend mogą zmieniać się bez wyraźnej logiki, tzn. niektóre strony (lub części stron) są wyświetlane w jednym języku, podczas gdy inne strony (lub części stron) w innym języku. Zwykle problem może wystąpić, gdy jest kilku użytkowników, z których część używa jednych ustawień regionalnych, a pozostali innych.

Znanym obejściem tego problemu jest wyłączenie wielowątkowości w PHP i Apache.

Problem jest związany ze sposobem działania ustawiania locale w PHP: informacje o locale są utrzymywane na poziomie procesu, a nie wątku. Dlatego w środowisku wielowątkowym, gdy kilka projektów jest uruchamianych przez ten sam proces Apache, możliwe jest, że locale zostanie zmienione w innym wątku, co wpływa na sposób przetwarzania danych w wątku Zabbix.

Więcej informacji można znaleźć w powiązanych zgłoszeniach problemów:

- ZBX-10911 (Problem z przełączaniem ustawień regionalnych frontend)
- ZBX-16297 (Problem z przetwarzaniem liczb na wykresach przy użyciu funkcji `bcdiv` z zestawu funkcji BC Math)

Wykresy

Problemy z wykresami (klasycznymi)

Jeśli napotkasz problemy z klasycznymi wykresami, zaleca się aktualizację biblioteki GD (libgd) do wersji 2.3.3-13 lub nowszej oraz PHP do wersji 8.0.19, 8.1.33, 8.2.29, 8.3.25, 8.4.12 lub nowszej.

Czas letni

Zmiany związane z czasem letnim (DST) powodują nieprawidłowości podczas wyświetlania etykiet osi X (duplikacja dat, brak dat itp.).

Agregacja sumy

W przypadku użycia agregacji sumy na wykresie dla okresu krótszego niż jedna godzina, wykresy wyświetlają nieprawidłowe (pomnożone) wartości, gdy dane pochodzą z trendów.

Nakładanie się tekstu

W przypadku niektórych języków frontendu (np. japońskiego) lokalne czcionki mogą powodować nakładanie się tekstu w legendzie wykresu. Aby tego uniknąć, użyj PHP GD extension w wersji 2.3.0 (lub nowszej).

Monitorowanie plików dziennika

Pozycje `log[]` i `logrt[]` wielokrotnie odczytują ponownie plik dziennika od początku, jeśli system plików jest zapełniony w 100% i do pliku dziennika są dopisywane nowe dane (więcej informacji można znaleźć w ZBX-10884).

Wolne zapytania MySQL

Serwer Zabbix generuje wolne zapytania `SELECT` w przypadku nieistniejących wartości dla pozycji. Ten problem występuje w wersjach MySQL 5.6/5.7 (szersze omówienie znajduje się w ZBX-10652) i w określonych przypadkach może również występować w nowszych wersjach MySQL. Obejściem tego problemu jest wyłączenie optymalizatora `index_condition_pushdown` lub `prefer_ordering_index` w MySQL. Należy jednak pamiętać, że to obejście może nie rozwiązać wszystkich problemów związanych z wolnymi zapytaniami.

Trwałe ustawienia filtra z linków

Po otwarciu linku do strony Zabbix frontend, który zawiera ustawienia filtra, w tym selektor czasu, filtr jest automatycznie zapisywany w bazie danych dla użytkownika, zastępując wcześniej zapisany filtr i/lub ustawienia selektora czasu dla tej strony. Ustawienia te pozostają aktywne, dopóki użytkownik nie zaktualizuje ich ręcznie lub nie zresetuje.

Problem z adresem IPv6 w pułapkach SNMPv3

Z powodu błędu net-snmp adres IPv6 może nie być poprawnie wyświetlany podczas używania SNMPv3 w pułapkach SNMP. Więcej szczegółów oraz możliwe obejście problemu można znaleźć w ZBX-14541.

Przycięty długi adres IPv6 w informacji o nieudanym logowaniu

Komunikat o nieudanej próbie logowania wyświetli tylko pierwsze 39 znaków zapisanego adresu IP, ponieważ jest to limit znaków w polu bazy danych. Oznacza to, że adresy IPv6 dłuższe niż 39 znaków będą wyświetlane niekompletnie.

Kontrole Zabbix agent w systemie Windows

Nieistniejące wpisy DNS w parametrze `Server` pliku konfiguracyjnego Zabbix agent (zabbix_agentd.conf) mogą wydłużyć czas odpowiedzi Zabbix agent w systemie Windows. Dzieje się tak, ponieważ demon buforowania DNS systemu Windows nie buforuje odpowiedzi negatywnych dla adresów IPv4. Jednak w przypadku adresów IPv6 odpowiedzi negatywne są buforowane, więc możliwym obejściem tego problemu jest wyłączenie IPv4 na hoście.

Eksport/import YAML

Istnieją pewne znane problemy z eksportem/importem:

- Komunikaty o błędach nie podlegają tłumaczeniu;
- Prawidłowy JSON z rozszerzeniem pliku .yaml czasami nie może zostać zaimportowany;
- Daty czytelne dla człowieka bez cudzysłowów są automatycznie konwertowane na znaczniki czasu Unix.

Kreator konfiguracji na SUSE z NGINX i php-fpm

Kreator konfiguracji frontendu nie może zapisać pliku konfiguracyjnego na SUSE z NGINX + php-fpm. Jest to spowodowane ustawieniem w jednostce /usr/lib/systemd/system/php-fpm.service, które uniemożliwia Zabbixowi zapis do /etc. (wprowadzone w PHP 7.4).

Dostępne są dwie opcje obejścia problemu:

- Ustaw opcję ProtectSystem na 'true' zamiast 'full' w jednostce systemd php-fpm.
- Zapisz ręcznie plik /etc/zabbix/web/zabbix.conf.php.

Przekazywanie nagłówka Authorization

W niektórych przypadkach Apache lub NGINX mogą uniemożliwiać dotarcie nagłówka Authorization w żądaniach API do Zabbix. Może to powodować problemy z uwierzytelnianiem podczas korzystania z Zabbix API lub usług jednokrotnego logowania (SSO), takich jak SAML z Okta.

Aby rozwiązać ten problem, zaktualizuj konfigurację swojego serwera WWW.

W przypadku **Apache**, jeśli używasz go jako reverse proxy (konfiguracja bez CGI), dodaj następującą dyrektywę do `/etc/httpd/conf/httpd` (w systemach opartych na RHEL) lub `/etc/apache2/apache2.conf` (w Debian/Ubuntu):

```
SetEnvIfNoCase ^Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
```

Jeśli Apache bezpośrednio wykonuje skrypty obsługujące żądania (np. przy użyciu mod_cgi), zamiast tego dodaj następującą dyrektywę:

```
CGIPassAuth On
```

Natomiast **NGINX** obsługuje nagłówek Authorization automatycznie. Jeśli jednak NGINX działa jako reverse proxy, możesz jawnie przekazywać nagłówek Authorization, dodając następujące dyrektywy do `/etc/nginx/nginx.conf` (dla lokalizacji twojego frontend Zabbix):

```
...
location / {
...
    proxy_set_header Authorization $http_authorization;
    proxy_pass http://backend_server;
...
}
```

Po zaktualizowaniu konfiguracji uruchom ponownie swój serwer WWW.

Więcej informacji znajdziesz tutaj:

- ZBX-22952
- Apache 2.4 + PHP-FPM and Authorization headers
- Dyrektywy SetEnvIfNoCase i CGIPassAuth
- NGINX Reverse Proxy

Chromium dla usługi web Zabbix w Ubuntu 20

Chociaż w większości przypadków usługa web Zabbix może działać z Chromium, w Ubuntu 20.04 użycie Chromium powoduje następujący błąd:

```
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
```

Ten błąd występuje, ponieważ `/var/lib/zabbix` jest używany jako katalog domowy użytkownika „zabbix”.

Niestandardowe kody błędów MySQL

Gdy Zabbix wykryje, że baza danych backend jest niedostępna, wysyła powiadomienie i nadal próbuje się połączyć. Dla niektórych silników baz danych rozpoznawane są określone kody błędów. W MySQL do rozpoznawanych kodów błędów należą:

- 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

Dodatkowo podczas używania Zabbix z instalacją MySQL na platformie Azure w logach Zabbix może pojawić się ogólny komunikat o błędzie *[9002] Some errors occurred*. Ten komunikat jest wysyłany przez bazę danych do serwera lub proxy Zabbix. Aby ustalić przyczynę błędu, sprawdź logi Azure.

Nieprawidłowe wyrażenia regularne po przełączeniu na PCRE2

W Zabbix 6.0 dodano obsługę PCRE2. Mimo że PCRE jest nadal obsługiwane, pakiety instalacyjne Zabbix dla RHEL 7 i nowszych, SLES (wszystkie wersje), Debian 9 i nowszych, Ubuntu 16.04 i nowszych zostały zaktualizowane tak, aby używać PCRE2. Chociaż zmiana na PCRE2 przynosi wiele korzyści, może spowodować, że niektóre istniejące wzorce regexp PCRE staną się nieprawidłowe lub będą działać inaczej. W szczególności dotyczy to wzorca *^[\w-\.]*. Aby ponownie uczynić to wyrażenie regularne prawidłowym

bez wpływu na jego semantykę, zmień wyrażenie na `^[-\w\.]` . Dzieje się tak dlatego, że PCRE2 traktuje znak myślnika jako separator, tworząc zakres wewnątrz klasy znaków.

Błąd widżetu Geomap

Mapy w widżecie Geomap mogą nie ładować się poprawnie, jeśli przeprowadzono aktualizację ze starszej wersji Zabbix z NGINX i podczas aktualizacji nie przełączono się na nowy plik konfiguracyjny NGINX.

Aby rozwiązać ten problem, można odrzucić stary plik konfiguracyjny, użyć pliku konfiguracyjnego z pakietu bieżącej wersji i ponownie go skonfigurować zgodnie z opisem w instrukcjach pobierania w sekcji *e. Configure PHP for Zabbix frontend*.

Alternatywnie można ręcznie edytować istniejący plik konfiguracyjny NGINX (zwykle */etc/zabbix/nginx.conf*). Aby to zrobić, otwórz plik i znajdź następujący blok:

```
location ~ /(api\/|conf[^\.]|include|locale|vendor) {
        deny            all;
        return          404;
}
```

Następnie zastąp ten blok następującym:

```
location ~ /(api\/|conf[^\.]|include|locale) {
        deny            all;
        return          404;
}

location /vendor {
        deny            all;
        return          404;
}
```

Wstępne przetwarzanie — zmienne globalne są niebezpieczne

JavaScript we wstępnym przetwarzaniu jest uruchamiany dla każdego żądania, ale przypisania do niezadeklarowanych identyfikatorów (na przykład `secret = value`) tworzą niejawne zmienne globalne, które mogą utrzymywać się poza bieżącym wykonaniem. Przechowywanie danych wrażliwych (tokenów, haseł itp.) w niejawnych zmiennych globalnych zwiększa ryzyko przypadkowego ujawnienia lub ponownego użycia przez kolejne uruchomienia wstępnego przetwarzania albo inne integracje wykonywane w tym samym środowisku.

Nie polegaj na niejawnych zmiennych globalnych. Zawsze deklaruj zmienne za pomocą `var` lub `const` i unikaj dołączania sekretów do obiektów globalnych (na przykład `globalThis` lub `window`). Nie ma obsługiwanego sposobu zastępowania wbudowanych obiektów globalnych z poziomu wstępnego przetwarzania.

Bezpieczny przykład:

```
var apiToken = payload.token;
var count = 1;
return JSON.stringify({ token: apiToken, calls: count });
```

Grupy procesorów w systemie Windows

Dokumentacja firmy Microsoft stwierdza, że systemy z mniej niż 64 procesorami logicznymi zawsze mają jedną grupę procesorów, Grupę 0. Jednak użytkownicy Zabbix zgłosili rzadki błąd ZBX-20260, gdy w systemach z 64 lub mniejszą liczbą procesorów logicznych występowały dwie grupy procesorów. Skutkowało to dostępnością liczników wydajności "\Processor(n)" tylko dla jednej z dwóch grup procesorów. Rzeczywista przyczyna źródłowa tego błędu nie jest znana. Jednak podobny przypadek opisano na stackoverflow.com, a jego przyczyną była współpraca między BIOS-em a systemem Windows.

Ograniczenia filtrowania przy użyciu porównań utf8mb4

Filtry (np. w *Data collection* > *Maintenance*) mogą nie działać poprawnie, gdy są stosowane do encji zawierających określone znaki Unicode (np. 𝓀, 𝔞). Problem ten wynika ze sposobu, w jaki domyślne porównanie utf8mb4_bin w bazach danych MySQL lub MariaDB obsługuje sortowanie i porównywanie znaków Unicode.

Aby obejść to ograniczenie, użytkownicy mogą zmienić porównanie kolumn bazy danych na alternatywne, takie jak utf8mb4_0900_bin, utf8mb4_0900_ai_ci lub utf8mb4_unicode_520_ci. Należy jednak pamiętać, że zmiana porównania może powodować nieoczekiwane zachowanie przy obsłudze pustych znaków, a także podczas sortowania i filtrowania innych znaków.

Więcej informacji o zmianie porównań można znaleźć w dokumentacji MySQL lub dokumentacji MariaDB. Szczegółowe informacje o różnicach między porównaniami znajdują się w sekcji Unicode Character Sets w dokumentacji MySQL.

Dostęp do elementów interfejsu użytkownika przy użyciu MariaDB 10.5.1-10.5.9

Uzyskiwanie dostępu do frontend Zabbix z rolą inną niż Super Admin może skutkować wyświetleniem komunikatu: „Wystąpił błąd systemowy. Skontaktuj się z administratorem Zabbix.”. Ten problem dotyczy instalacji korzystających z wersji MariaDB od 10.5.1 do 10.5.9.

Aby uniknąć tego problemu, zaktualizuj MariaDB do wersji nowszej niż 10.5.9. Więcej informacji znajduje się w ZBX-25746.

Profilowanie nadmiernego użycia pamięci za pomocą tcmalloc

Jeśli podejrzewasz, że Twoja instalacja Zabbixa zużywa zbyt dużo pamięci, możesz użyć funkcji profilowania pamięci tcmalloc, aby zbadać zużycie pamięci przez serwer/proxy Zabbixa.

1. Podczas instalacji Zabbixa ze źródeł skonfiguruj dodatkowe flagi:

```
export CFLAGS="-std=gnu99 -g -O0"
```

Flaga `-std=gnu99` jest wymagana do kompilacji serwera Zabbixa, proxy Zabbixa lub agenta Zabbixa. Flaga `-g` dodaje dodatkowe informacje debugowania, natomiast `-O0` wyłącza optymalizacje, które mogą zakłócać profilowanie tcmalloc.

2. Ustaw następujące zmienne środowiskowe przed uruchomieniem serwera Zabbixa. Zmienne te informują tcmalloc, jak śledzić i raportować użycie pamięci:

```
LD_PRELOAD="/usr/lib/aarch64-linux-gnu/libtcmalloc.so" \
HEAPPROFILE=./heap_profile \
HEAP_PROFILE_ALLOCATION_INTERVAL=0 \
HEAP_PROFILE_INUSE_INTERVAL=4294967296 \
HEAPPROFILESIGNAL=5 \
MALLOCSTATS=1 \
./sbin/zabbix_server -f -c /etc/zabbix/zabbix_server.conf
```

3. Wyzwól zrzut profilu, wysyłając sygnał 5 do procesu docelowego. Zastąp 1234 rzeczywistym identyfikatorem procesu (PID):

```
kill -5 1234
```

4. Wyświetl wygenerowany profil:

```
pprof-symbolize -text ./sbin/zabbix_server ./heap_profile.0001.heap

Using local file ./sbin/zabbix_server.
Using local file ./heap_profile.0001.heap.
Total: 1078.1 MB
  1076.8  99.9%  99.9%   1076.8  99.9% zbx_malloc2
     1.0   0.1% 100.0%      1.0   0.1% __GI___strdup
     0.2   0.0% 100.0%      0.2   0.0% CRYPTO_zalloc@@OPENSSL_3.0.0
     0.1   0.0% 100.0%      0.1   0.0% OPENSSL_LH_insert@@OPENSSL_3.0.0
     0.0   0.0% 100.0%      0.0   0.0% zbx_realloc2
     0.0   0.0% 100.0%      0.1   0.0% PKCS7_decrypt@@OPENSSL_3.0.0
     0.0   0.0% 100.0%      0.0   0.0% find_best_tree_node
     0.0   0.0% 100.0%      0.0   0.0% CRYPTO_strndup@@OPENSSL_3.0.0
     ...
     0.0   0.0% 100.0%      0.0   0.0% preprocessing_flush_value
     0.0   0.0% 100.0%   1074.0  99.6% preprocessor_add_request
```

W tym przykładzie zbx_malloc2 odpowiada za niemal wszystkie alokacje pamięci.

Zobacz także:

- ZBX-25050 oraz ZBX-25584 — powiązane zgłoszenia problemów.
- GCC Option Summary — opcje kompilacji (`-std=gnu99`, `-g`, `-O0` itd.).
- Dokumentacja Gperftools Heap Profiler dotycząca zmiennych środowiskowych dla profilowania tcmalloc.

Replikacja grupowa MySQL 8.0 w trybie multi-primary

Podczas korzystania z replikacji grupowej MySQL 8.0 w trybie multi-primary możesz napotkać błąd podczas zatwierdzania transakcji podobny do poniższego:

```
1531697:20250128:064734.697 query [txnlev:1] [update alerts set status=1,retries=0,error='' where alertid=
1531697:20250128:064734.713 query [txnlev:1] [commit;]
1531697:20250128:064734.753 [Z3005] query failed: [3101] Plugin instructed the server to rollback the curr
```

Wygląda na to, że ten błąd jest wywoływany przez problemy z operacjami wycofywania transakcji obejmującymi ograniczenia kluczy obcych.

Zobacz także:

- ZBX-26060 — powiązane zgłoszenie problemu.
- MySQL Bug #96758 "Rollbacks with Foreign Keys on single node" — problem zgłoszony po stronie upstream.

1 Problemy z kompilacją

To są znane problemy dotyczące kompilacji Zabbix ze źródeł. We wszystkich pozostałych przypadkach zobacz stronę Znane problemy.

Biblioteka w niestandardowej lokalizacji

Zabbix umożliwia określenie biblioteki znajdującej się w niestandardowej lokalizacji. W poniższym przykładzie Zabbix uruchomi `curl-config` z podanej niestandardowej lokalizacji i użyje jego wyniku do określenia właściwej biblioteki libcurl do użycia.

```
$ ./configure --enable-server --with-mysql --with-libcurl=/usr/local/bin/curl-config
```

To zadziała, jeśli jest to jedyna zainstalowana w systemie biblioteka libcurl, ale może nie zadziałać, jeśli inna biblioteka libcurl jest zainstalowana w standardowej lokalizacji (na przykład przez menedżer pakietów). Taka sytuacja ma miejsce, gdy potrzebujesz nowszej wersji biblioteki dla Zabbix, a starszej dla innych aplikacji.

Dlatego wskazanie komponentu w niestandardowej lokalizacji nie zawsze zadziała, gdy ten sam komponent istnieje również w standardowej lokalizacji.

Na przykład, jeśli używasz nowszej biblioteki libcurl zainstalowanej w `/usr/local`, a pakiet libcurl nadal jest zainstalowany, Zabbix może wybrać niewłaściwą wersję i kompilacja zakończy się niepowodzeniem:

```
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
```

W tym przypadku funkcja `curl_easy_header()` nie jest dostępna w starszej bibliotece `/usr/lib/x86_64-linux-gnu/libcurl.so`, ale jest dostępna w nowszej `/usr/local/lib/libcurl.so`.

Problem leży w kolejności flag linkera, a jednym z rozwiązań jest podanie pełnej ścieżki do biblioteki w zmiennej LDFLAGS:

```
$ LDFLAGS="-Wl,--no-as-needed /usr/local/lib/libcurl.so" ./configure --enable-server --with-mysql --with-l
```

Zwróć uwagę na opcję `-Wl,--no-as-needed`, która może być wymagana w niektórych systemach (zobacz też: domyślne opcje linkowania w systemach opartych na Debianie).

Zbyt mały rozmiar stosu w niektórych systemach

Jeśli Zabbix ulega awarii lub zawiesza się z powodu przepełnienia stosu, możesz zwiększyć rozmiar stosu dla pojedynczego wątku, używając opcji `--with-stacksize` podczas konfigurowania źródeł. Problem ten może wystąpić w systemach z niskimi domyślnymi limitami stosu wątków, zwłaszcza podczas preprocessing, gdy tworzonych jest wiele wątków.

Poniższy przykład ustawia rozmiar stosu na 512 KB na wątek:

```
./configure --enable-server --with-mysql --with-stacksize=512
```

Możesz sprawdzić limity stosu wątków systemu w czasie działania za pomocą polecenia `ulimit -s` w systemach opartych na Linuxie.

2 Problemy z aktualizacją związane z ucieczką znaków

Przekroczony limit znaków dla parametrów funkcji po aktualizacji

W Zabbix 7.0 dodano prawidłowe stosowanie znaków ucieczki dla ukośników odwrotnych w parametrach tekstowych funkcji historii.

Ponieważ podczas aktualizacji z wersji Zabbix starszych niż 7.0 dodawane są dodatkowe ukośniki odwrotne, prowadzi to do wydłużenia parametrów, co może skutkować uszkodzeniem funkcji wyzwalaczy, jeśli długość parametru przekroczy maksymalny rozmiar danych wynoszący 255 znaków.

Aby zilustrować, jakie problemy mogą wystąpić podczas aktualizacji bazy danych, jaki mają wpływ oraz jak je rozpoznać i naprawić, rozważmy przykładową konfigurację z 3 pozycjami i 2 wyzwalaczami.

Konfiguracja

1. Wersja Zabbix 6.0.
2. Plik `/tmp/ONE`, który zawiera 114 ukośników odwrotnych, z literą „a" na końcu:

```
\\\
```

4. Wyzwalacze zostały zaktualizowane w następujący sposób:

Trig_228:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\
```

Zmiany w 7.4.8 Ta wersja minor nie zawiera żadnych zmian w szablonach.

Zmiany w 7.4.7 Zaktualizowane szablony

- Ciena 3906 by SNMP został zaktualizowany o pozycje dotyczące systemu plików i obciążenia CPU.
- Vyatta Virtual Router by SNMP został zaktualizowany o nowe pozycje OID oraz ulepszenia panelu.

Zmiany w 7.4.6 Nowe szablony

- Ribbon SBC Edge by HTTP, szablon do monitorowania urządzeń Ribbon SBC Edge (wcześniej SWe Lite) przez HTTP.
- Ribbon SBC SWe Core by HTTP, szablon do monitorowania urządzeń SBC SWe Core przez HTTP oraz Ribbon SBC SWe CE by HTTP, szablon do monitorowania instancji Ribbon SBC SWe Call Engine (CE) przez HTTP.
- VeloCloud SD-WAN Edge by HTTP, szablon do monitorowania urządzeń VeloCloud SD-WAN Edge przez HTTP.

Zaktualizowane szablony

- MySQL by Zabbix agent, MySQL by Zabbix agent 2, MySQL by Zabbix agent active, MySQL by Zabbix agent 2 active oraz MySQL by ODBC zostały zaktualizowane, aby obsługiwać zarówno `SHOW SLAVE STATUS` (stara składnia), jak i `SHOW REPLICA STATUS` (nowa składnia).
- VeloCloud SD-WAN by HTTP, wcześniej *VMWare SD-WAN VeloCloud by HTTP*, został przemianowany i zaktualizowany, aby dostosować się do najnowszej platformy VeloCloud SD-WAN.

Zmiany w 7.4.4 Nowe szablony

- Stormshield SNS by SNMP, szablon do monitorowania urządzeń Stormshield Network Security (SNS) za pomocą SNMP.

Zmiany w 7.4.3 Nowe szablony

- Aruba CX 8300s by SNMP, szablon zapewniający monitorowanie serii przełączników Aruba CX 8300 na podstawie SNMP.
- Ciena 3906 by SNMP, szablon do monitorowania urządzeń Ciena 3906.
- Vyatta Virtual Router by SNMP, szablon do monitorowania wirtualnego routera Vyatta 1908e.

Zaktualizowane szablony

- Huawei OceanStor Dorado by SNMP został zaktualizowany o szczegóły dotyczące pokrycia urządzeń.
- Szablony Zabbix server health i Zabbix proxy health zostały zaktualizowane o rozszerzone możliwości monitorowania przepustowości preprocesora.

Zmiany w 7.4.2 Nowe szablony

- Zestaw szablonów AWS by HTTP został uzupełniony o szablon *AWS Backup Vault by HTTP*.

Zmiany w 7.4.1 Nowe szablony

- Cisco Secure Firewall Threat Defense by HTTP, szablon zapewniający możliwości monitorowania urządzeń Cisco Secure Firewall Threat Defense przy użyciu REST API.

Zaktualizowane szablony

- Szablon *VMware Guest* w zestawach szablonów VMware oraz VMware FQDN został zaktualizowany o nową pozycję *Boot time*. Ta pozycja zastępuje teraz *Uptime of guest OS* w wyzwalaczu *VM has been restarted*, aby uniknąć fałszywych alarmów po migracjach maszyn wirtualnych, takich jak vMotion.

10 Uwagi dotyczące aktualizacji do wersji 7.4.0

Te uwagi dotyczą aktualizacji z Zabbix 7.2.x do Zabbix 7.4.0.

Wszystkie uwagi są pogrupowane w:

- **Zmiany powodujące niezgodność** - zmiany, które mogą spowodować awarię istniejących instalacji, oraz inne istotne informacje związane z procesem aktualizacji
- **Inne** - wszystkie pozostałe informacje opisujące zmiany w funkcjonalności Zabbix

Zobacz także:

- Procedura aktualizacji zawierająca wszystkie istotne informacje dotyczące aktualizacji z wersji wcześniejszych niż Zabbix 7.4.0;
- Aktualizacja klastra HA z instrukcjami dotyczącymi aktualizacji serwerów w klastrze **wysokiej dostępności** (HA).

Proces aktualizacji

Aby pomyślnie zakończyć aktualizację serwera Zabbix na MySQL/MariaDB, może być konieczne ustawienie `GLOBAL log_bin_trust_functi` = 1 w MySQL, jeśli włączone jest binarne logowanie, nie ma uprawnień superużytkownika, a `log_bin_trust_function_creators` = 1 nie jest ustawione w pliku konfiguracyjnym MySQL.

Aby ustawić tę zmienną za pomocą konsoli MySQL, uruchom:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
```

Po pomyślnym zakończeniu aktualizacji tę opcję można wyłączyć:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 0;
```

Przełomowe zmiany Biblioteka PCRE została wycofana

Biblioteka PCRE (lub PCRE1) została wycofana. Zabbix jest teraz kompilowany z użyciem PCRE2.

Zarządzanie własnymi mediami użytkownika

Domyślnie wszyscy użytkownicy mogą teraz zarządzać własnymi mediami.

Uprawnienia użytkowników do zmiany szczegółów mediów dla siebie mogą być przyznawane (lub odbierane) na podstawie ich roli użytkownika (zobacz opcję *Tworzenie i edycja własnych mediów*).

Dodatkowo uprawnienia użytkowników Super admin do zmiany szczegółów mediów innych osób mogą być również przyznawane/odbierane na podstawie ich roli użytkownika (zobacz opcję *Tworzenie i edycja mediów użytkownika*).

W przypadku aktualizacji ze starszych wersji obie opcje będą włączone dla wszystkich ról, dla których włączono *Domyślny dostęp do nowych akcji*. Należy pamiętać, że jeśli *Domyślny dostęp do nowych akcji* nie jest włączony, administratorzy i superadministratorzy **mogą utracić** możliwość edycji mediów po aktualizacji.

Escapowanie ukośników odwrotnych w parametrach funkcji historii

Prawidłowe escapowanie ukośników odwrotnych zostało dodane w parametrach tekstowych funkcji historii od Zabbix 7.0.0.

Ponieważ podczas aktualizacji z wersji Zabbix wcześniejszych niż 7.0 dodawane są dodatkowe ukośniki odwrotne, prowadzi to do dłuższych parametrów, co może skutkować nieprawidłowym działaniem funkcji wyzwalacza, jeśli długość parametru przekroczy maksymalny rozmiar danych wynoszący 255 znaków.

Aby uniknąć tego problemu, zaleca się ręczne przeniesienie długich parametrów do makr użytkownika przed aktualizacją.

W przypadku aktualizacji z wersji Zabbix wcześniejszych niż 7.0 dodano dodatkowe kontrole długości wynikowego parametru. Jeśli wynikowa długość przekracza maksymalny rozmiar, takie parametry **nie są aktualizowane**, a w logach jest wyświetlane ostrzeżenie z prośbą o ręczne poprawienie wymienionych parametrów.

Zobacz też problemy z aktualizacją związane z escapowaniem.

Aktualizacja wtyczki MSSQL Zabbix agent 2

Szablon MSSQL by Zabbix agent 2 został zaktualizowany o filtry umożliwiające uwzględnianie lub wykluczanie wykrytych członków quorum według nazwy, a także o filtr usługi, który pozwala filtrować według nazwy klastra, aby wykluczyć puste klastry. Aby szablon działał bez błędów, wtyczka MSSQL Zabbix agent 2 musi zostać zaktualizowana do wersji równej lub wyższej niż 7.4.0.

Minimalna wymagana wersja libssh2

Minimalna wymagana wersja libssh2 została podniesiona z 1.0.0 do 1.8.0.

Prototypy hostów na wykrytych hostach

Jeśli bieżąca konfiguracja zawiera prototypy hostów, które używają szablonów zawierających inne prototypy hostów, uaktualnienie do Zabbix 7.4 spowoduje, że wszystkie wykryte hosty będą miały prototypy hostów zdefiniowane w tych szablonach. Jeśli takie zachowanie nie jest zamierzone, przed uaktualnieniem należy ręcznie usunąć prototypy hostów z odpowiednich szablonów.

Aby zidentyfikować wykryte hosty, które otrzymają prototypy hostów podczas uaktualnienia, można uruchomić następujące zapytanie SQL w bazie danych Zabbix:

```
SELECT h.hostid,ht.templateid
FROM hosts_templates ht
JOIN hosts h ON ht.hostid=h.hostid
WHERE h.flags=4
  AND EXISTS (
    SELECT NULL
    FROM items i,host_discovery hd
    WHERE i.hostid=ht.templateid
```

```
    AND hd.parent_itemid=i.itemid
    )
ORDER BY hostid;
```

Parametry połączenia z bazą danych

DBPort i DBSocket są teraz wzajemnie wykluczające się. Podczas określania parametrów połączenia z bazą danych można zdefiniować albo DBPort, albo DBSocket, albo pozostawić oba niezdefiniowane, aby użyć wartości domyślnych.

Inne Nieobsługiwane pozycje SNMP walk/discovery bez prawidłowych OID

Pozycje SNMP walk, w przypadku gdy nie istnieją żadne OID ani instancje OID, teraz zwracają błąd, a pozycja staje się nieobsługiwana. Szczegóły nieistniejącego OID/instancji są rejestrowane przy DebugLevel=5. Wcześniej w takich przypadkach zwracany był pusty ciąg.

Podobnie pozycje SNMP discovery, w przypadku gdy nie istnieją żadne OID ani instancje OID, teraz zwracają błąd, a pozycja staje się nieobsługiwana. Wcześniej w takich przypadkach zwracana była pusta tablica.

Należy pamiętać, że pusty ciąg/tablica jest teraz zwracany tylko wtedy, gdy OID/instancja istnieje, ale nie ma dla niej danych.

Zarządzanie pamięcią podręczną historii

W niektórych scenariuszach zbierania danych określone pozycje mogą tymczasowo blokować pamięć podręczną historii serwera/proxy. Może to opóźnić zapisywanie danych historycznych do bazy danych i spowolnić działanie systemu. Aby ułatwić zarządzanie pamięcią podręczną historii, wprowadzono następujące ulepszenia:

- **Ręczne czyszczenie pamięci podręcznej:** Nowe polecenie czasu działania history_cache_clear=target dla Zabbix serwer/proxy umożliwia ręczne wyczyszczenie pamięci podręcznej historii dla określonej pozycji na podstawie jej ID.
- **Automatyczne czyszczenie pamięci podręcznej:** Gdy wyłączysz pozycję, jest ona natychmiast usuwana z pamięci podręcznej historii (z wyjątkiem jej ostatniej wartości, która jest zachowywana do logów). Podobnie, gdy wyłączysz host, wszystkie jego pozycje są usuwane z pamięci podręcznej historii (z wyjątkiem ich ostatnich wartości).
- **Logowanie diagnostyki pamięci podręcznej:** Gdy pamięć podręczna historii jest pełna, Zabbix serwer/proxy zapisuje teraz informacje diagnostyczne dotyczące pamięci podręcznej historii od DebugLevel=3 (wcześniej DebugLevel=4). Log zawiera pozycje z największą liczbą wartości w pamięci podręcznej historii.

Aby zapewnić długoterminową stabilność systemu, upewnij się, że zbieranie danych jest zrównoważone względem dostępnych zasobów (wydajność bazy danych, rozmiar pamięci podręcznej, interwały zbierania, parametry pozycji logów itp.). Możesz monitorować pamięć podręczną historii Zabbix za pomocą wewnętrznej pozycji zabbix[wcache]. Możesz również rozważyć zwiększenie rozmiaru pamięci podręcznej historii dla Zabbix serwer/proxy.

Osobna sekcja menu dla powiadomień użytkownika

Aby poprawić widoczność mediów użytkownika, powiadomienia użytkownika mają teraz własną sekcję menu w obszarze *Ustawienia użytkownika*.

Nowa sekcja Powiadomienia zawiera dwie karty - *Media* i *Powiadomienia frontend*, które wcześniej były częścią sekcji profilu użytkownika.

Prawidłowe informacje z zagnieżdżonych grup hostów na mapach

Informacje z zagnieżdżonych grup hostów są teraz poprawnie wyświetlane na mapach, na przykład:

- Etykieta grupy hostów teraz wyświetla podsumowanie problemów ze wszystkich hostów w zagnieżdżonych grupach hostów;
- Widok „Elementy grupy hostów" teraz wyświetla osobny element mapy dla każdego hosta w zagnieżdżonych grupach hostów;
- Etykieta mapy teraz wyświetla podsumowanie wszystkich problemów zawartych w zagnieżdżonych grupach hostów.

Tabela bazy danych dla przekonwertowanych ustawień

Tabela `settings` zastępuje teraz tabelę `config` do przechowywania parametrów globalnej konfiguracji. Nowa tabela używa formatu klucz-wartość zamiast przechowywania parametrów w jednym wierszu z jedną kolumną na każdy parametr. Przykład tego, jak wpływa to na wymianę danych, można znaleźć w sekcji Protokół wymiany danych serwer-proxy (obiekt `"settings"`).

Zwiększone maksymalne rozmiary pamięci podręcznej dla serwera i proxy

Maksymalne rozmiary pamięci podręcznej zostały zwiększone z 2 GB do 16 GB, aby opóźnić wyczerpanie cache i utrzymać działanie podczas tymczasowych problemów (np. z konfiguracją, bazą danych lub siecią):

- serwer: zwiększono maksymalną wartość dla HistoryCacheSize, HistoryIndexCacheSize oraz TrendCacheSize;
- proxy: zwiększono maksymalne wartości dla HistoryCacheSize oraz HistoryIndexCacheSize.

Skrócony domyślny okres przechowywania sesji użytkowników

Domyślny okres przechowywania danych sesji użytkowników został skrócony z 365 dni do 31 dni. Zmiana ta dotyczy parametru hk_sessions, który teraz domyślnie ma wartość 31d zamiast 365d.

Minimalna obsługiwana wersja Go

Minimalna obsługiwana wersja Go została podniesiona z 1.21 do 1.23.

Jeśli wcześniej kompilowano Zabbix agent 2, ładowalne wtyczki agent 2 lub usługę webową ze źródeł przy użyciu wersji Go starszej niż 1.23, zaleca się ponowną kompilację tych komponentów przy użyciu obsługiwanej wersji Go, aby otrzymywać najnowsze aktualizacje zabezpieczeń i poprawki błędów. Komponenty skompilowane przy użyciu starszej wersji Go będą nadal działać, ale ich aktualizacja będzie wymagała nowszego środowiska Go.

Aktualizacja szablonu dla Host Wizard

Nowy Host Wizard wprowadza prowadzony, krok po kroku interfejs do konfigurowania monitorowanego obiektu docelowego (urządzenia, aplikacji, usługi itp.) w Zabbix. Upraszcza konfigurację nowych lub istniejących hostów, prowadząc użytkowników przez kluczowe kroki, takie jak wybór szablonu, instalacja Zabbix agent, dodawanie interfejsów hosta i inne.

Po uaktualnieniu z wcześniejszej wersji Zabbix szablony muszą zostać zaktualizowane, aby działały z Host Wizard. Instrukcje znajdują się w sekcji Aktualizacja szablonu.

11 Uwagi dotyczące aktualizacji do wersji 7.4.x

Ta strona zawiera zbiorcze uwagi dotyczące aktualizacji dla wydań minorowych głównej wersji Zabbix.

Zobacz także uwagi dotyczące aktualizacji dla wersji głównej.

Upgrade notes for 7.4.11 This minor version does not have any upgrade notes.

Uwagi dotyczące aktualizacji do wersji 7.4.10 Elementy alarmów VMware zwracają szczegóły encji

Zwracany przez pozycje `vmware.*.alarms.get[]` JSON zawiera teraz obiekt `entity` dla każdego alarmu. Ten obiekt zwraca identyfikator, UUID oraz typ dotkniętej encji (host, maszyna wirtualna lub datastore).

Jeśli masz skrypty, które analizują dane wyjściowe pozycji alarmów VMware, może być konieczne ich zaktualizowanie, aby obsługiwały dodatkowe pola.

Zaktualizowana biblioteka Duo Universal PHP

Biblioteka Duo Universal PHP (dostarczana wraz z Zabbix) została zaktualizowana z wersji 1.1.1 do 1.1.2.

Uwagi dotyczące aktualizacji do wersji 7.4.9 Ta wersja minor nie zawiera żadnych uwag dotyczących aktualizacji.

Uwagi dotyczące aktualizacji do wersji 7.4.8 Wartości pozycji wyświetlane jako HTML przy użyciu iframe'ów

Dane pozycji w widżecie Item history, gdy są wyświetlane jako tekst sformatowany w HTML, są teraz izolowane w iframe'ach.

Zaktualizowana biblioteka SAML PHP Toolkit

Biblioteka SAML PHP Toolkit (dostarczana z Zabbix) została zaktualizowana z wersji 4.0.0 do 4.3.1.

Uwagi dotyczące aktualizacji do 7.4.7 Zaktualizowana biblioteka Duo Universal PHP

Biblioteka Duo Universal PHP (dostarczana z Zabbix) została zaktualizowana z wersji 1.0.2 do 1.1.1.

Uwagi dotyczące aktualizacji do 7.4.6 Ograniczone uprawnienia dla plików logów

Pliki logów komponentów Zabbix są teraz tworzone z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo pliki są czytelne dla grupy właściciela. Wszystkie pozostałe uprawnienia są odrzucone.

W systemie Linux ta zmiana dotyczy uprawnień z 0666 na 0640. Zmiana wpływa na pliki logów serwera Zabbix, proxy i agent. Uprawnienia dla plików logów Zabbix agent 2 zostały zaktualizowane z 0644 na 0640 (przez usunięcie uprawnienia odczytu dla innych).

Ponadto kilka **innych plików** jest teraz tworzonych z uprawnieniami 0640:

- pliki UNIX pid dla procesów Zabbix
- pliki eksportu danych

- pliki tymczasowe dla kontroli `icmpping*`
- pliki trwałego logu

Zapytania o status replikacji MySQL obsługiwane przez agent Zabbix agent 2

agent Zabbix agent 2 wtyczka MySQL obsługuje teraz zarówno `SHOW SLAVE STATUS`, jak i `SHOW REPLICA STATUS`. Wtyczka automatycznie wybiera i wykonuje odpowiednie polecenie w zależności od wersji serwera oraz zwraca wynik w standardowym formacie JSON.

Pozycje agent 2, takie jak `mysql.replication.discovery` i `mysql.replication.get_slave_status`, działają z serwerami używającymi zarówno `SHOW SLAVE STATUS`, jak i `SHOW REPLICA STATUS`. Nie są wymagane żadne zmiany w istniejących pozycjach agent 2 ani szablonach agent 2.

Minimalna obsługiwana wersja Go

Minimalna obsługiwana wersja Go została podniesiona z 1.23 do 1.24.10.

Jeśli wcześniej kompilowano Zabbix agent 2, ładowalne wtyczki agent 2 lub usługę web ze źródeł przy użyciu wersji Go starszej niż 1.24.10, zaleca się ponowne skompilowanie tych komponentów przy użyciu obsługiwanej wersji Go, aby otrzymywać najnowsze aktualizacje zabezpieczeń i poprawki błędów. Komponenty zbudowane przy użyciu starszej wersji Go będą nadal działać, ale ich aktualizacja będzie wymagała nowszego środowiska Go.

Dołączanie plików lub folderów do konfiguracji usługi webowej

Do pliku konfiguracyjnego usługi webowej Zabbix dodano parametr Include. Parametr ten umożliwia dołączenie pojedynczych plików lub wszystkich plików w katalogu do konfiguracji.

Uwagi dotyczące aktualizacji do wersji 7.4.5 Ta wersja minor nie zawiera żadnych uwag dotyczących aktualizacji.

Uwagi dotyczące aktualizacji do wersji 7.4.4 **Ostrzeżenie!** Aktualizacja do tej wersji nie jest zalecana z powodu błędów związanych z wykresami oraz wtyczką MySQL dla Zabbix agent 2.

Uwagi dotyczące aktualizacji do wersji 7.4.3 Minimalna wersja PHP cURL

Minimalna obsługiwana wersja rozszerzenia cURL PHP to teraz 7.19.4.

Uwagi dotyczące aktualizacji do wersji 7.4.1 Strumień wyjściowy sterowania uruchomieniowego agent 2

Polecenia Zabbix agent 2 runtime control (`zabbix_agent2 -R <option>`) teraz zapisują dane wyjściowe do `stdout` (standardowe wyjście) zamiast do `stderr` (standardowe wyjście błędów). Zaktualizuj wszystkie skrypty, które opierają się na poprzednim zachowaniu.

5 Szybki start

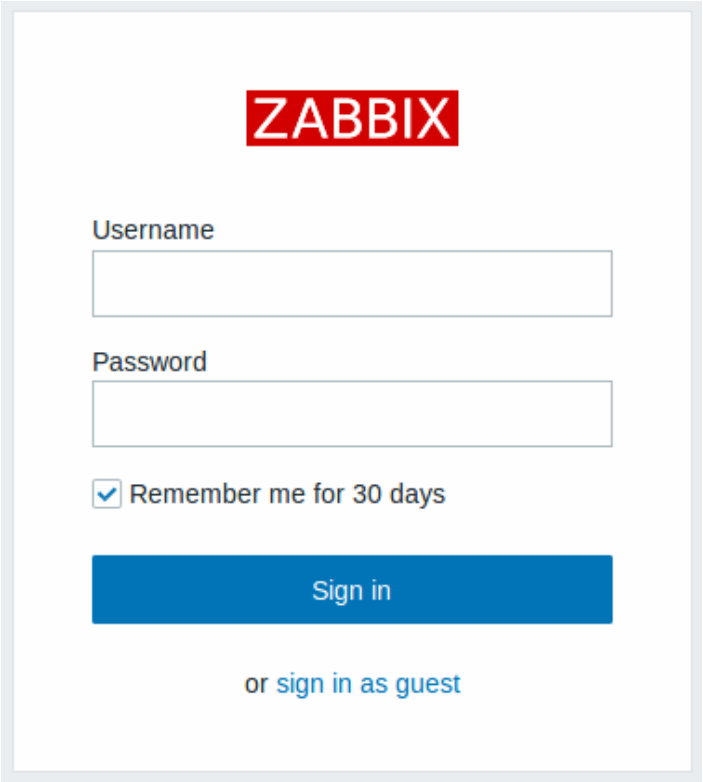
Ta sekcja zawiera instrukcję krok po kroku dotyczącą konfiguracji podstawowego monitorowania: jak zalogować się do Zabbixa, dodać host, zdefiniować pozycję, utworzyć wyzwalacz, otrzymywać powiadomienia oraz zapisać konfigurację jako szablon.

1 Logowanie i konfiguracja użytkownika

Przegląd

W tej sekcji dowiesz się, jak zalogować się i skonfigurować użytkownika systemowego w Zabbix.

Logowanie



To jest ekran powitalny Zabbixa. Wprowadź nazwę użytkownika **Admin** oraz hasło **zabbix**, aby zalogować się jako superużytkownik Zabbixa. Zostanie przyznany dostęp do wszystkich sekcji menu.

Ze względów bezpieczeństwa zdecydowanie zaleca się natychmiastową zmianę domyślnego hasła dla konta Admin po pierwszym logowaniu.

Trwałe logowanie

Aby pozostać zalogowanym przez maksymalnie 30 dni, zaznacz opcję *Zapamiętaj na 30 dni* przed kliknięciem *Zaloguj się*.

Opcja *Zapamiętaj na 30 dni* włączona:

- Twoja sesja pozostaje aktywna przez 30 dni.
- Ustawienie *Auto-logout* jest nadpisywane, dzięki czemu pozostajesz zalogowany do końca tego okresu.
- Przy kolejnych wizytach w ciągu 30 dni zostaniesz zalogowany automatycznie bez ponownego wprowadzania danych logowania.

Opcja *Zapamiętaj na 30 dni* wyłączona:

- Wszelkie wcześniej włączone automatyczne logowanie zostaje wyczyszczone.
- Sesja wygaśnie zgodnie ze skonfigurowanym interwałem *Auto-logout*.

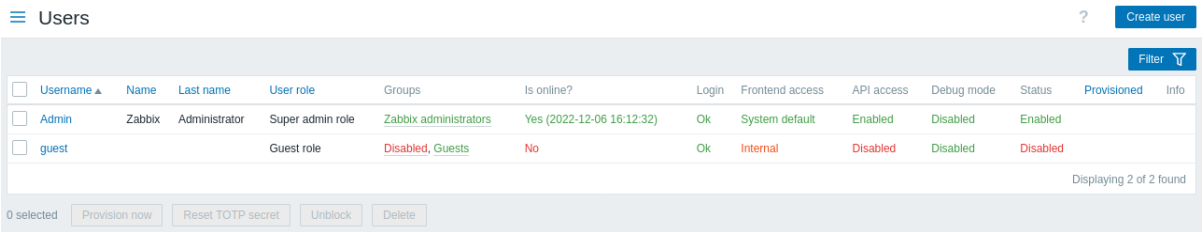
Ochrona przed atakami brute force

W przypadku pięciu kolejnych nieudanych prób logowania interfejs Zabbix zostanie wstrzymany na 30 sekund, aby zapobiec atakom brute force i słownikowym.

Adres IP nieudanej próby logowania zostanie wyświetlony po pomyślnym zalogowaniu.

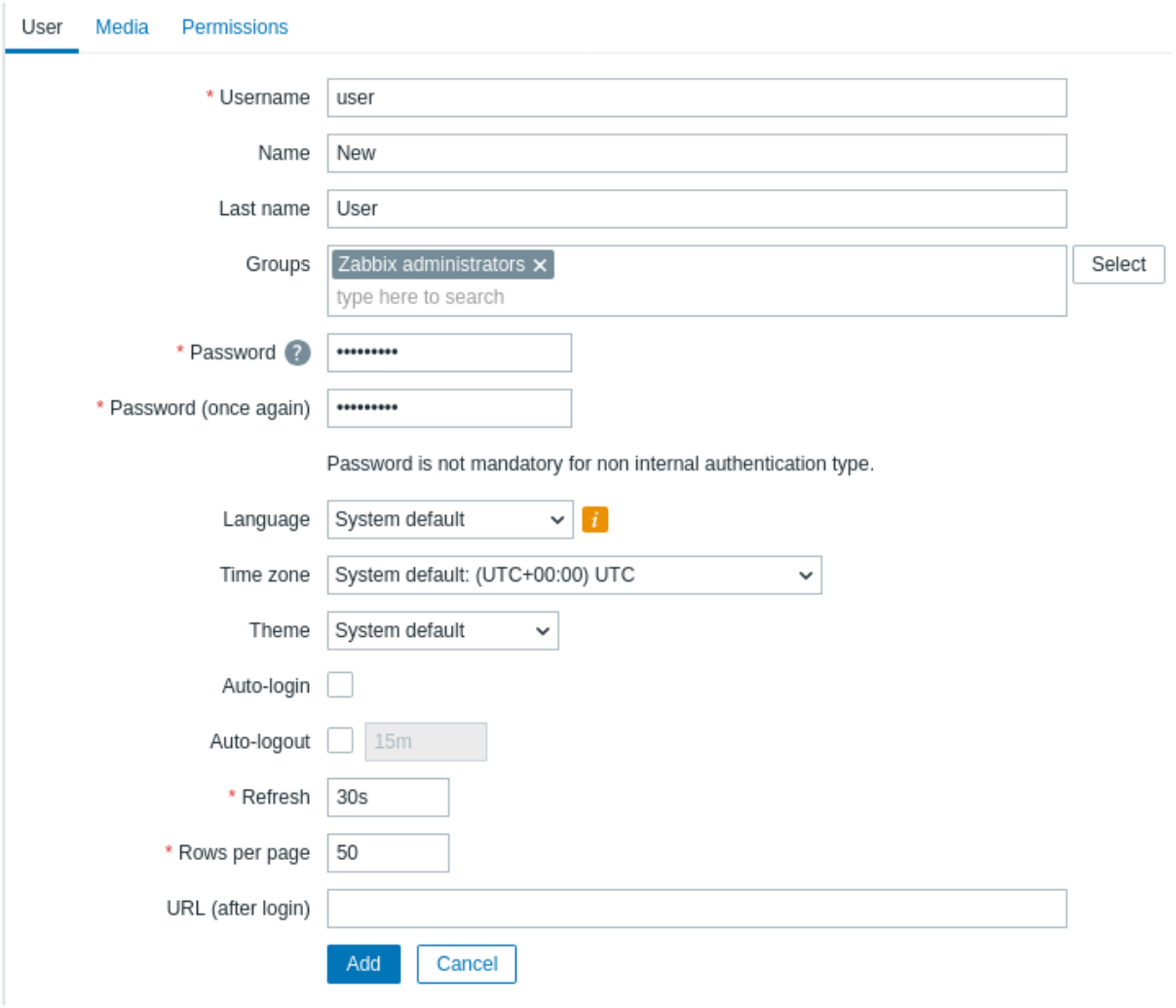
Dodawanie użytkownika

Aby wyświetlić informacje o użytkownikach, przejdź do *Users* > *Users* w pionowym menu bocznym.



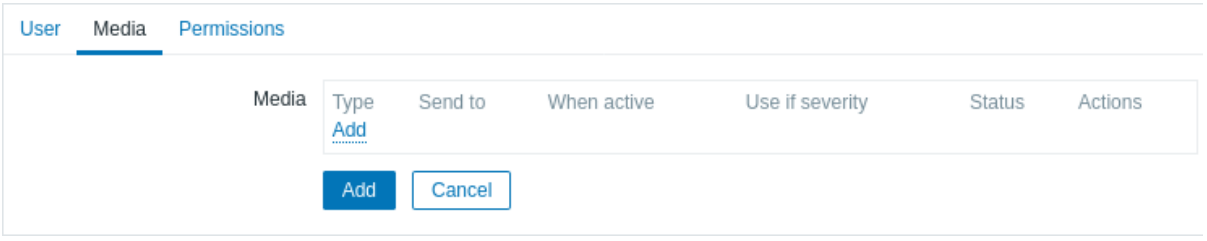
Aby dodać nowego użytkownika, wybierz *Create user* w prawym górnym rogu.

W formularzu nowego użytkownika upewnij się, że dodajesz użytkownika do jednej z istniejących grup użytkowników, na przykład 'Zabbix administrators'.



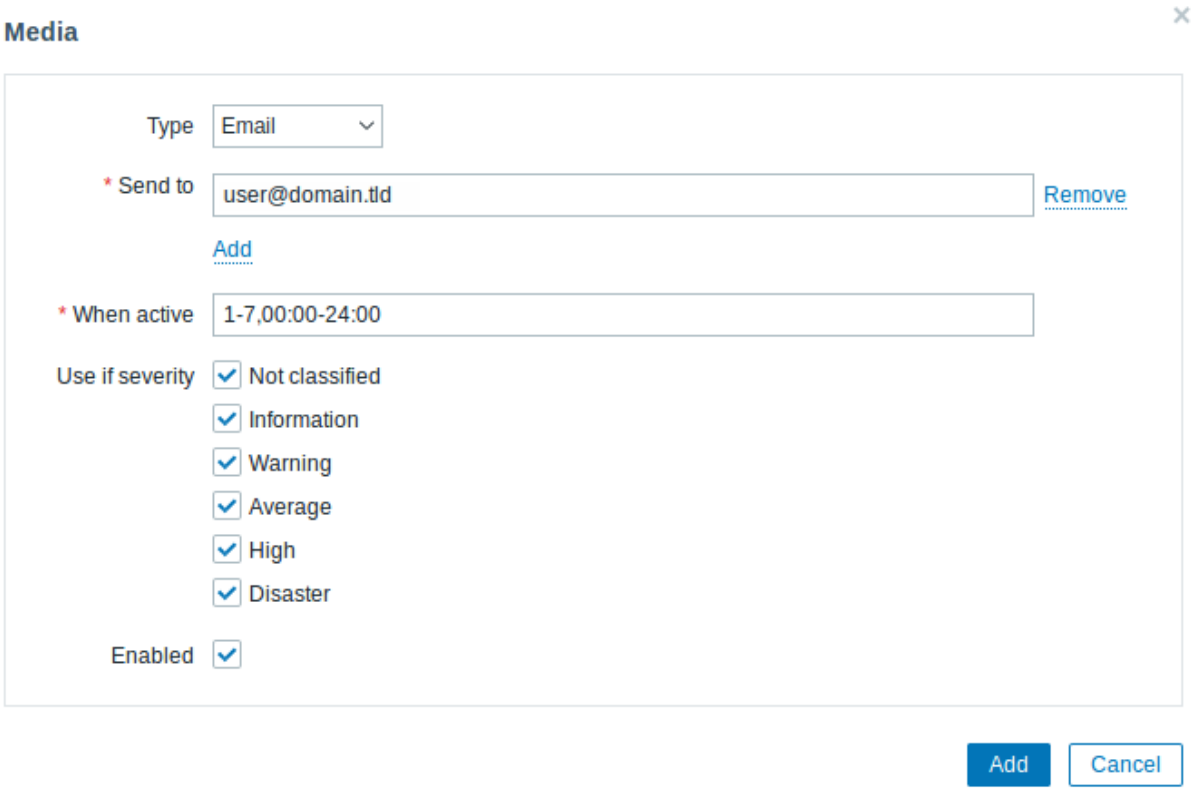
Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką. Szczegółowe informacje o polach wejściowych w tym formularzu konfiguracyjnym znajdziesz na stronie User settings.

Domyślnie nowi użytkownicy nie mają zdefiniowanych żadnych mediów (metod dostarczania powiadomień). Aby utworzyć medium, przejdź do karty 'Media' i kliknij *Add*.



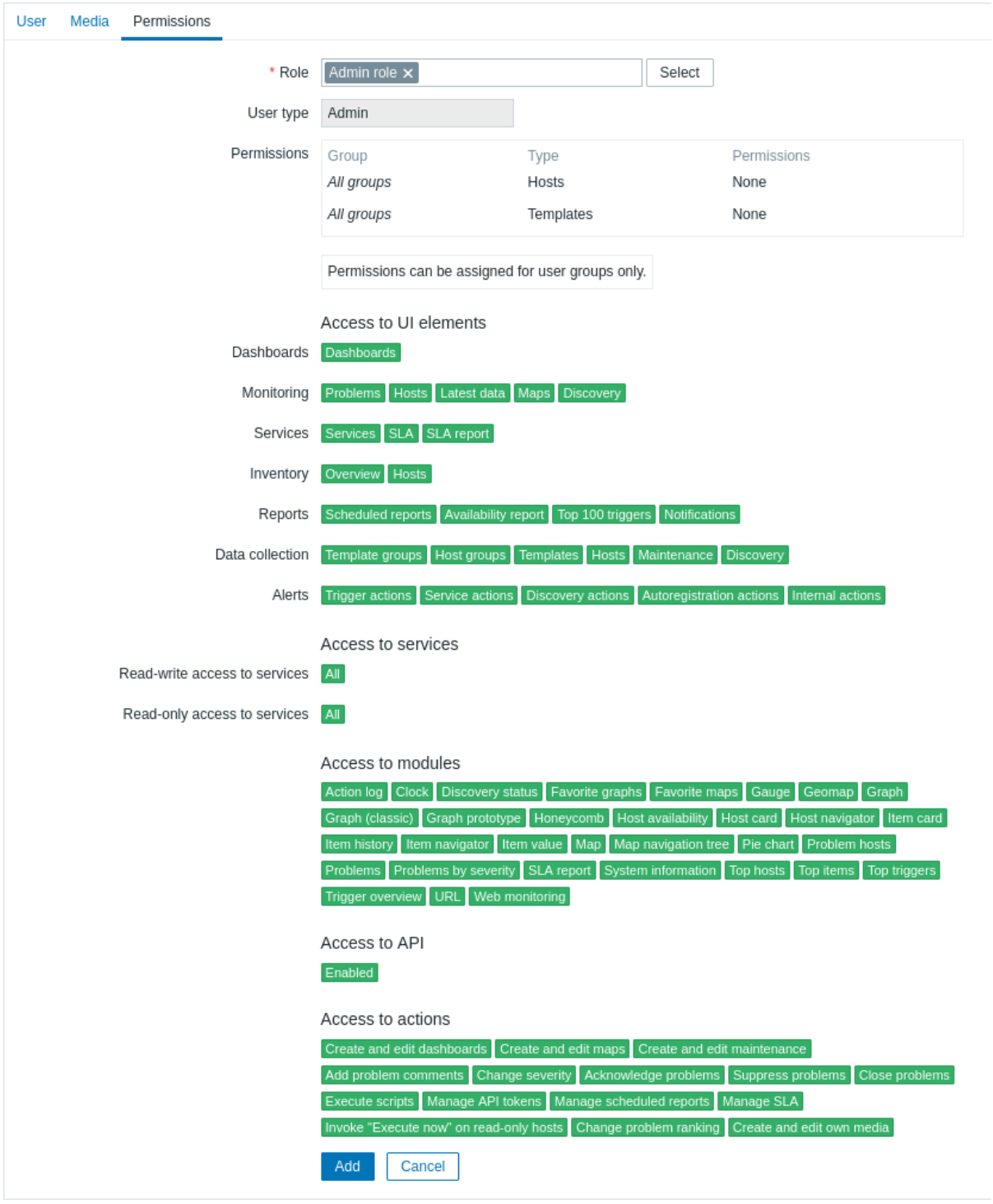
W oknie podręcznym wpisz adres e-mail użytkownika.

Możesz określić przedział czasu, w którym medium będzie aktywne (zobacz stronę Time period specification, aby poznać opis formatu). Domyślnie medium jest zawsze aktywne. Możesz też dostosować poziomy trigger severity, dla których medium będzie aktywne, ale na razie pozostaw wszystkie włączone.



Kliknij *Add*, aby zapisać medium, a następnie przejdź do karty Permissions.

Karta Permissions ma obowiązkowe pole *Role*. Rola określa, które elementy frontend użytkownik może wyświetlać i jakie działania może wykonywać. Naciśnij Select i wybierz jedną z ról z listy. Na przykład wybierz *Admin role*, aby umożliwić dostęp do wszystkich sekcji frontend Zabbixa, z wyjątkiem Administration. Później możesz zmodyfikować uprawnienia lub utworzyć więcej ról użytkowników. Po wybraniu roli uprawnienia pojawią się na tej samej karcie:

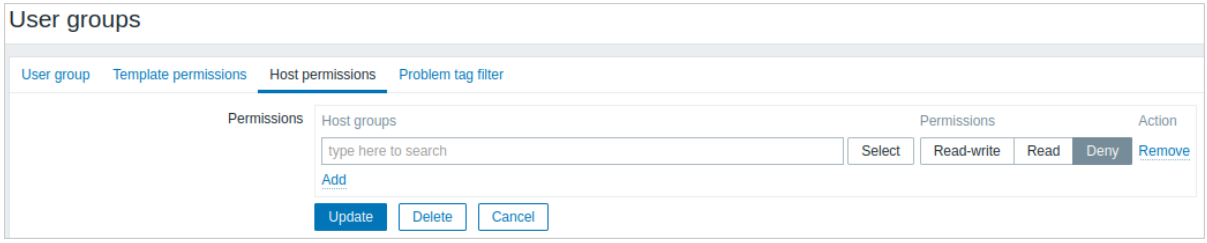


Kliknij *Add* w formularzu właściwości użytkownika, aby zapisać użytkownika. Nowy użytkownik pojawi się na liście użytkowników.



Dodawanie uprawnień

Domyślnie nowy użytkownik nie ma uprawnień dostępu do hostów i szablonów. Aby nadać użytkownikowi prawa, kliknij grupę użytkownika w kolumnie *Groups* (w tym przypadku - 'Zabbix administrators'). W formularzu właściwości *User groups* przejdź do karty *Host permissions*, aby przypisać uprawnienia do grup hostów. Kliknij Add w polu wyboru grupy hostów, aby wyświetlić:



Następnie kliknij *Select* obok pola, aby zobaczyć listę grup hostów. Ten użytkownik ma mieć dostęp tylko do odczytu do grupy *Linux servers*, więc zaznacz odpowiednie pole wyboru na liście i kliknij *Select*, aby potwierdzić wybór.



Kliknij przycisk *Read*, aby ustawić poziom uprawnień, a następnie *Update*, aby zapisać zmiany wprowadzone w konfiguracji grupy użytkowników.

Aby nadać uprawnienia do szablonów, należy przejść do karty *Template permissions* i określić grupy szablonów. Kroki są identyczne jak przy przypisywaniu uprawnień do grup hostów. Omówienie szablonów jest dostępne w sekcji Nowy szablon tego Quickstart.

Attention:
W Zabbix prawa dostępu do hostów i szablonów są przypisywane do grup użytkowników, a nie do pojedynczych użytkowników.

Gotowe! Możesz spróbować zalogować się, używając danych uwierzytelniających nowego użytkownika.

2 Nowy host

Przegląd

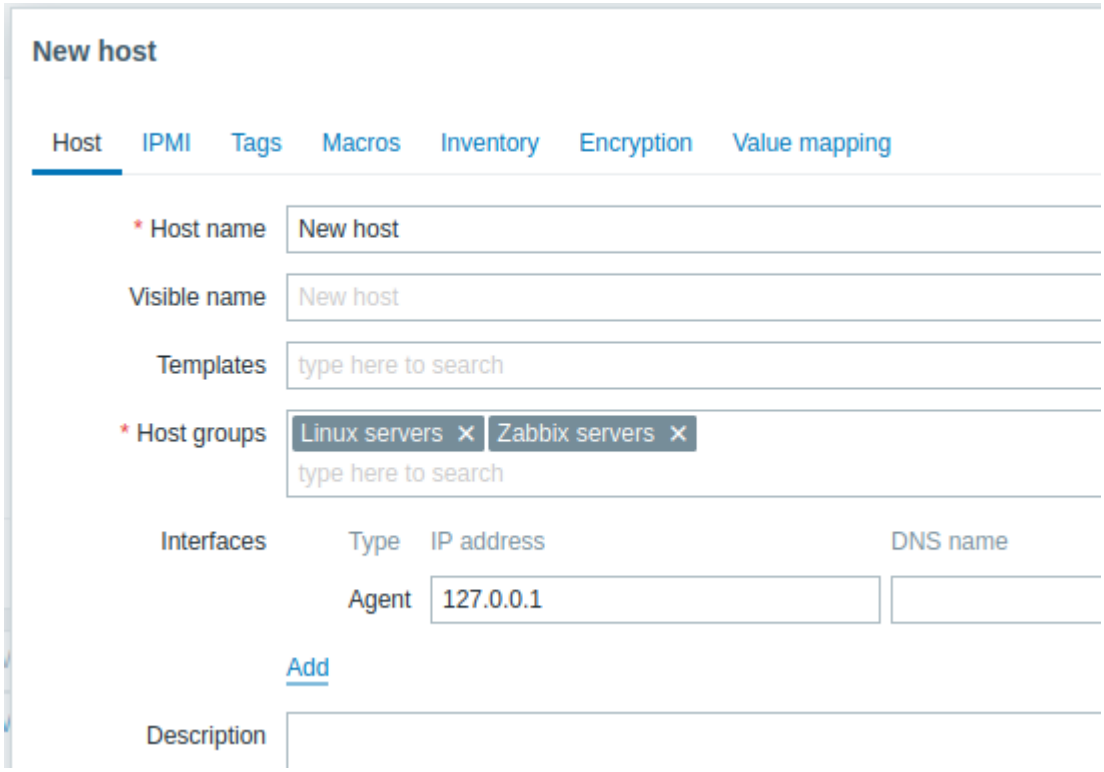
W tej sekcji dowiesz się, jak skonfigurować nowy host.

Host w Zabbix to obiekt sieciowy (fizyczny, wirtualny), który chcesz monitorować. Definicja tego, co może być ”hostem” w Zabbix, jest dość elastyczna. Może to być fizyczny serwer, przełącznik sieciowy, maszyna wirtualna lub jakaś aplikacja.

Dodawanie hosta

Informacje o skonfigurowanych hostach w Zabbix są dostępne w sekcjach menu *Data collection* > *Hosts* oraz *Monitoring* > *Hosts*. Istnieje już jeden predefiniowany host o nazwie ”Zabbix server”, ale chcemy nauczyć się dodawania kolejnego.

Aby dodać nowy host, kliknij *Create host*. Zostanie otwarty formularz konfiguracji hosta.



Wszystkie wymagane pola są oznaczone czerwoną gwiazdką.

Minimalny zakres danych, które należy tutaj wprowadzić, to:

Host name

- Wprowadź nazwę hosta. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia.

Host groups

- Wybierz jedną lub kilka istniejących grup, klikając przycisk *Select*, albo wprowadź nazwę nieistniejącej grupy, aby utworzyć nową grupę.

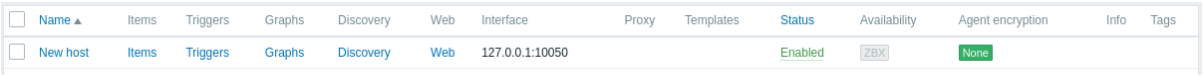
Note:
Wszystkie uprawnienia dostępu są przypisywane do grup hostów, a nie do pojedynczych hostów. Dlatego host musi należeć do co najmniej jednej grupy.

Interfaces: IP address

- Chociaż technicznie nie jest to pole wymagane, interfejs hosta jest niezbędny do zbierania niektórych metryk. Aby używać pasywnych kontroli Zabbix agent, określ w tym polu adres IP lub DNS agenta. Pamiętaj, że w dyrektywie 'Server' w pliku konfiguracyjnym Zabbix agent należy również podać adres IP lub DNS serwera Zabbix. Jeśli Zabbix agent i serwer Zabbix są zainstalowane na tej samej maszynie, należy podać ten sam adres IP/DNS w obu miejscach.

Pozostałe opcje można pozostawić bez zmian, używając ich wartości domyślnych.

Po zakończeniu kliknij *Add*. Nowy host powinien być widoczny na liście hostów.



Kolumna Availability zawiera wskaźniki dostępności hosta dla każdego interfejsu. Zdefiniowaliśmy interfejs Zabbix agent, więc możemy użyć ikony dostępności agenta (z napisem 'ZBX'), aby sprawdzić dostępność hosta:

-  - status hosta nie został jeszcze ustalony; nie wykonano jeszcze żadnej kontroli metryki
-  - host jest dostępny, kontrola metryki zakończyła się powodzeniem
-  - host jest niedostępny, kontrola metryki zakończyła się niepowodzeniem (przesuń kursor myszy nad ikonę, aby zobaczyć komunikat o błędzie). Może występować błąd komunikacji, prawdopodobnie spowodowany nieprawidłowymi danymi uwierzytelniającymi interfejsu. Sprawdź, czy serwer Zabbix działa, i spróbuj odświeżyć stronę później.

3 Nowa pozycja

Przegląd

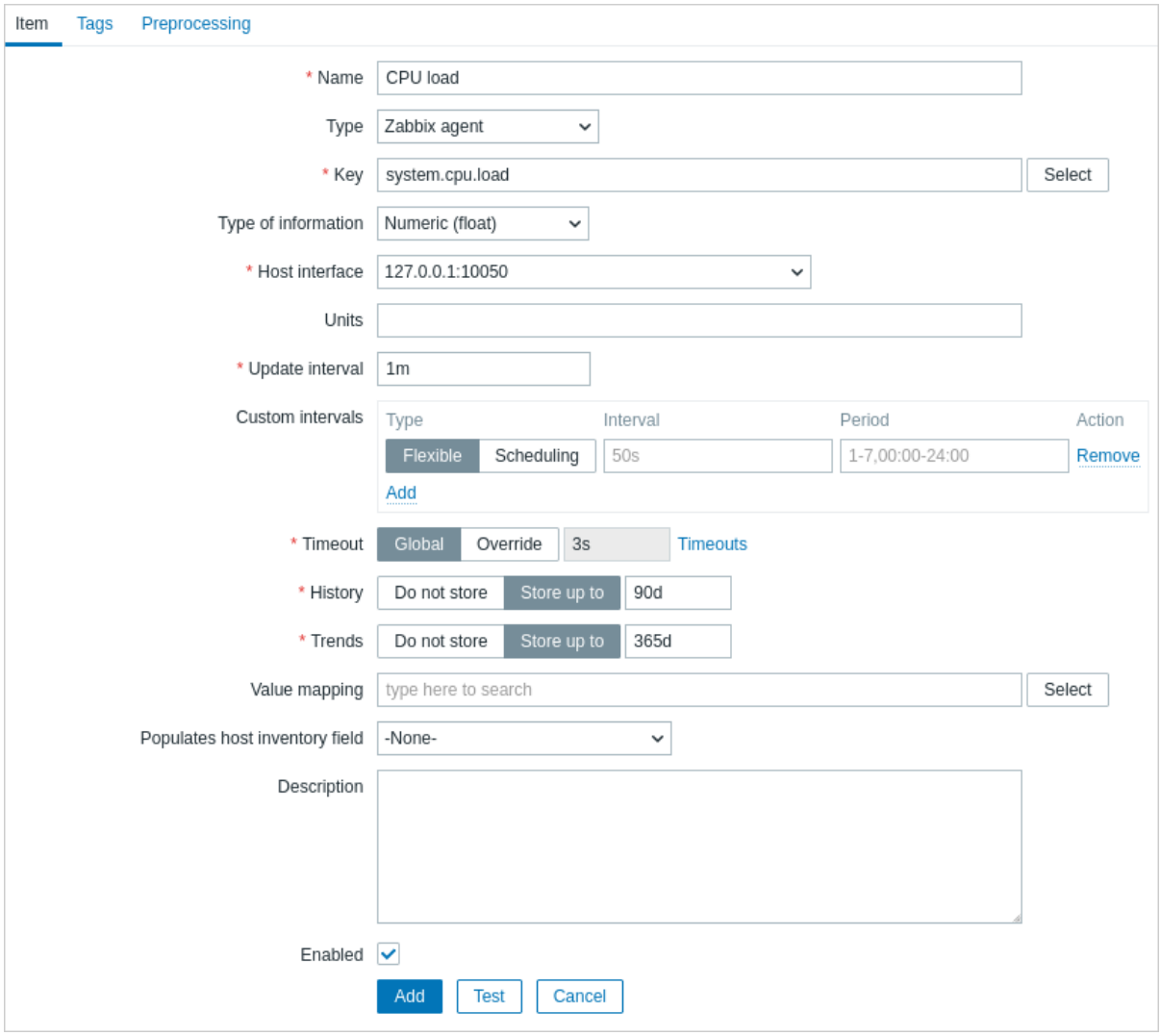
W tej sekcji dowiesz się, jak skonfigurować pozycję.

Pozycje są podstawą zbierania danych w Zabbix. Bez pozycji nie ma danych - ponieważ tylko pozycja definiuje pojedynczą metrykę lub rodzaj danych do zbierania z hosta.

Dodawanie pozycji

Wszystkie pozycje są grupowane wokół hostów. Dlatego, aby skonfigurować przykładową pozycję, przechodzimy do *Data collection* > *Hosts* i odnajdujemy utworzony przez nas "New host".

Kliknij link *Items* w wierszu "New host", a następnie kliknij *Create item*. Zostanie otwarty formularz konfiguracji pozycji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W przypadku naszej przykładowej pozycji podstawowe informacje do wprowadzenia to:

Name

- Wpisz wartość *CPU load*. Będzie to nazwa pozycji wyświetlana na listach i w innych miejscach.

Key

- Ręcznie wpisz wartość *system.cpu.load*. Jest to techniczna nazwa pozycji, która identyfikuje typ informacji, jakie będą zbierane. Ten konkretny klucz jest tylko jednym z predefined keys, które są dostarczane z agentem Zabbix.

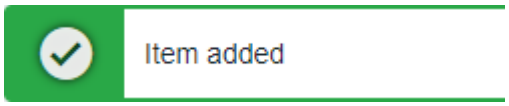
Type of information

- Ten atrybut definiuje format oczekiwanych danych. Dla klucza *system.cpu.load* to pole zostanie automatycznie ustawione na *Numeric (float)*.

Note:
Możesz również zmniejszyć liczbę dni przechowywania item history do 7 lub 14. To dobra praktyka, ponieważ odciąża bazę danych od przechowywania dużej liczby wartości historycznych.

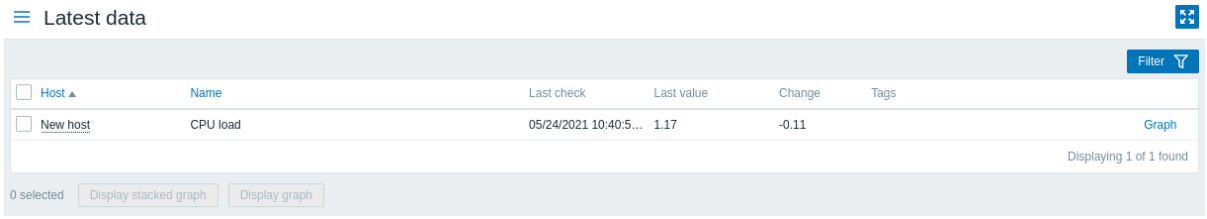
Other options można pozostawić bez zmian, używając wartości domyślnych.

Po zakończeniu kliknij *Add*. Nowa pozycja powinna pojawić się na liście pozycji, a Ty powinieneś zobaczyć komunikat o powodzeniu.



Wyświetlanie danych

Po zdefiniowaniu pozycji możesz chcieć sprawdzić, czy rzeczywiście zbiera ona dane. W tym celu przejdź do *Monitoring* > *Latest data*, wybierz 'New host' w filtrze i kliknij *Apply*.



Należy jednak pamiętać, że pojawienie się pierwszych danych może potrwać do 60 sekund. Domyślnie z taką częstotliwością serwer odczytuje zmiany konfiguracji i wykrywa nowe pozycje do wykonania.

Jeśli w kolumnie 'Change' nie widzisz żadnej wartości, być może do tej pory odebrano tylko jedną wartość. Poczekaj 30 sekund na pojawienie się kolejnej wartości.

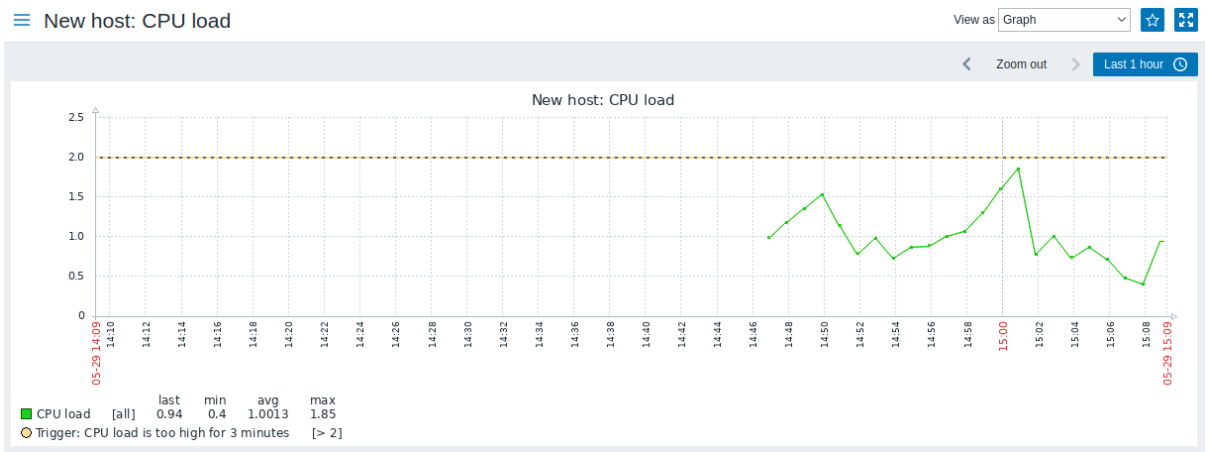
Jeśli nie widzisz informacji o pozycji jak na zrzucie ekranu, upewnij się, że:

- pola pozycji 'Key' i 'Type of information' zostały wypełnione dokładnie tak jak na zrzucie ekranu;
- zarówno agent, jak i serwer są uruchomione;
- status hosta to 'Monitored', a jego ikona dostępności jest zielona;
- wybrany host w filtrze hosta jest poprawny;
- pozycja jest włączona.

Wykresy

Gdy pozycja działa już od pewnego czasu, może nadejść moment, aby zobaczyć coś wizualnego. Proste wykresy są dostępne dla każdej monitorowanej numerycznej pozycji bez dodatkowej konfiguracji. Te wykresy są generowane w czasie działania.

Aby wyświetlić wykres, przejdź do *Monitoring* > *Latest data* i kliknij łącze 'Graph' obok pozycji.



4 Nowy wyzwalacz

Przegląd

W tej sekcji dowiesz się, jak skonfigurować wyzwalacz.

Pozycje tylko zbierają dane. Aby automatycznie oceniać przychodzące dane, musimy zdefiniować wyzwalacze. Wyzwalacz zawiera wyrażenie, które definiuje próg określający, jaki poziom danych jest akceptowalny.

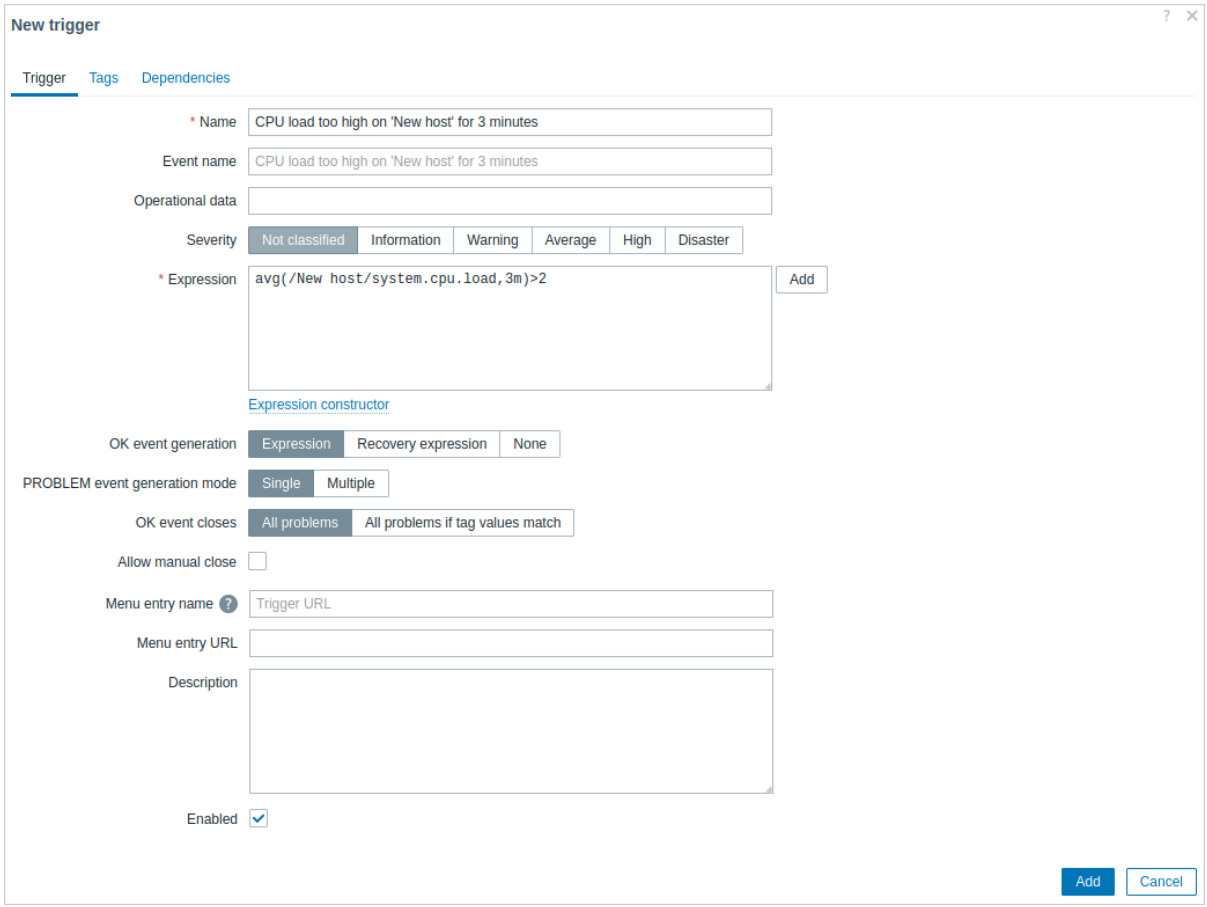
Jeśli ten poziom zostanie przekroczony przez przychodzące dane, wyzwalacz zostanie „uruchomiony" lub przejdzie w stan „Problem" — co oznacza, że wydarzyło się coś, co może wymagać uwagi. Jeśli poziom ponownie stanie się akceptowalny, wyzwalacz wraca do stanu „Ok".

Dodawanie wyzwalacza

Aby skonfigurować wyzwalacz dla naszej pozycji:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij link *Triggers* w wierszu „New host".
3. Kliknij *Create trigger*.

Zostanie otwarty formularz konfiguracji wyzwalacza.



W przypadku naszego wyzwalacza najważniejsze informacje do wprowadzenia tutaj to:

Name

- Wprowadź wartość *CPU load too high on 'New host' for 3 minutes*. Będzie to nazwa wyzwalacza wyświetlana na listach i w innych miejscach.

Expression

- Wprowadź: `avg(/New host/system.cpu.load,3m)>2`

To jest wyrażenie wyzwalacza. Upewnij się, że wyrażenie zostało wpisane poprawnie, aż do ostatniego znaku. Klucz pozycji tutaj (`system.cpu.load`) służy do odwołania się do pozycji. To konkretne wyrażenie zasadniczo oznacza, że próg problemu zostaje przekroczony, gdy średnia wartość obciążenia CPU z 3 minut jest większa niż 2. Więcej informacji znajdziesz w składni wyrażeń wyzwalaczy.

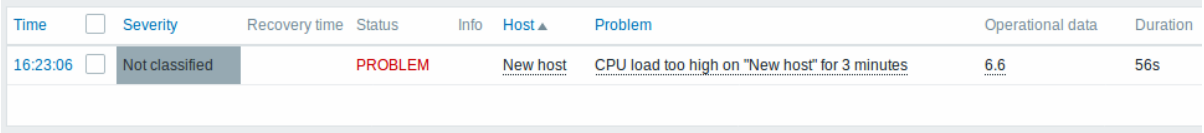
Pozostałe opcje można pozostawić bez zmian, używając ich wartości domyślnych.

Po zakończeniu kliknij *Add*. Nowy wyzwalacz powinien pojawić się na liście wyzwalaczy.

Wyświetlanie statusu wyzwalacza

Po zdefiniowaniu wyzwalacza możesz chcieć sprawdzić jego status.

Jeśli obciążenie CPU przekroczyło poziom progowy zdefiniowany w wyzwalaczu, problem zostanie wyświetlony w *Monitoring* > *Problems*.



Miganie w kolumnie statusu wskazuje na niedawną zmianę statusu wyzwalacza, która nastąpiła w ciągu ostatnich 30 minut.

5 Otrzymywanie powiadomienia o problemie

Przegląd

W tej sekcji dowiesz się, jak skonfigurować alerty w formie powiadomień w Zabbix.

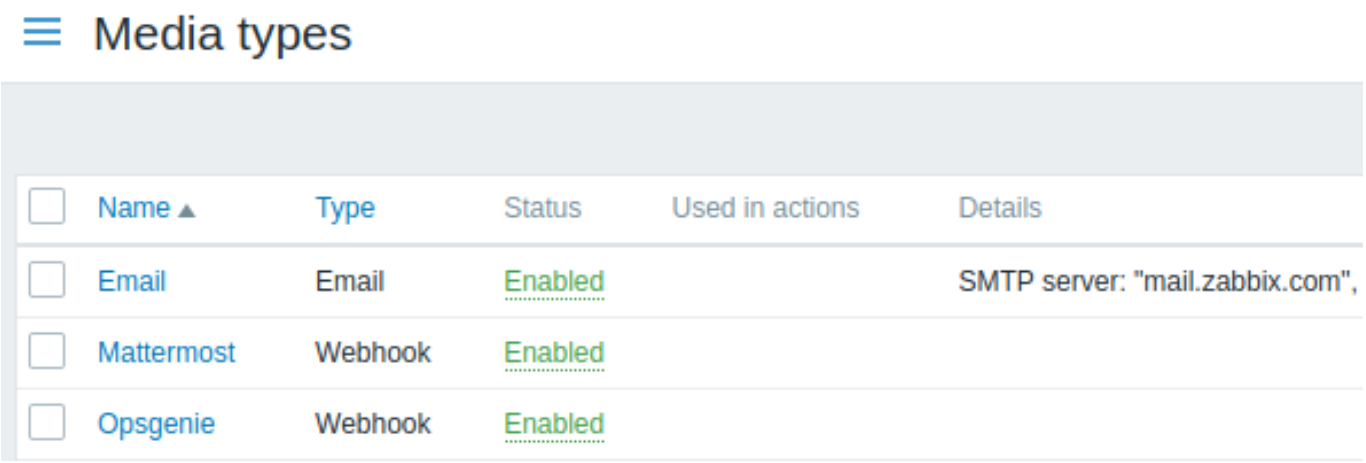
Gdy pozycje zbierają dane, a wyzwalacze są zaprojektowane tak, aby „uruchamiać się" w sytuacjach problemowych, przydatne jest również posiadanie mechanizmu alertowania, który będzie informował o ważnych zdarzeniach, nawet gdy frontend Zabbix nie jest aktywnie sprawdzany.

Temu właśnie służą powiadomienia. Ponieważ e-mail jest najpopularniejszą metodą dostarczania powiadomień o problemach, nauczymy się, jak skonfigurować powiadomienie e-mail.

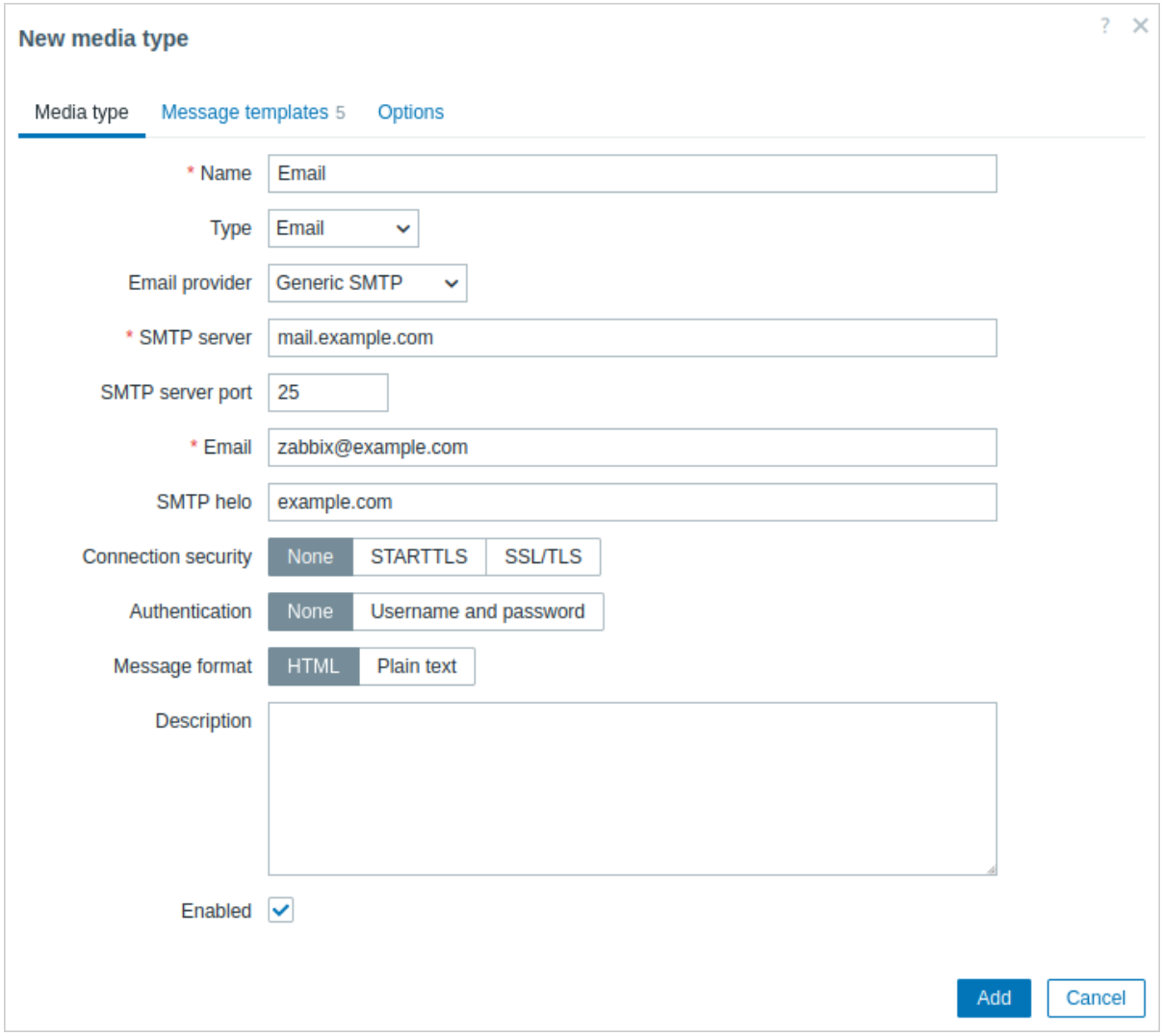
Ustawienia e-mail

Początkowo w Zabbix dostępnych jest kilka predefiniowanych metod dostarczania powiadomień. Jedną z nich jest Email.

Aby skonfigurować ustawienia e-mail, przejdź do *Alerts* > *Media types* i kliknij *Email* na liście predefiniowanych typów mediów.



Zostanie otwarty formularz konfiguracji ustawień e-mail.



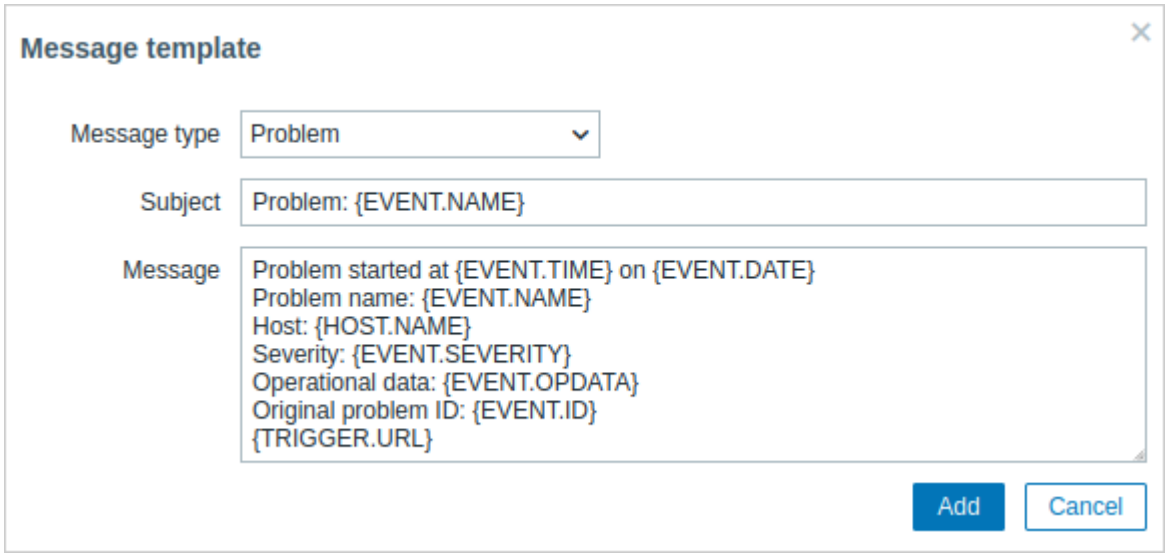
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W zakładce *Media type* ustaw wartości *SMTP server*, *SMTP helo* oraz *Email* odpowiednio do swojego środowiska.

Note:
Wartość w polu *Email* będzie używana jako adres "From" dla powiadomień wysyłanych z Zabbix.

Następnie należy zdefiniować treść komunikatu o problemie. Treść jest definiowana za pomocą szablonu wiadomości, skonfigurowanego w zakładce *Message templates*.

Kliknij *Add*, aby utworzyć szablon wiadomości, i wybierz *Problem* jako typ wiadomości.

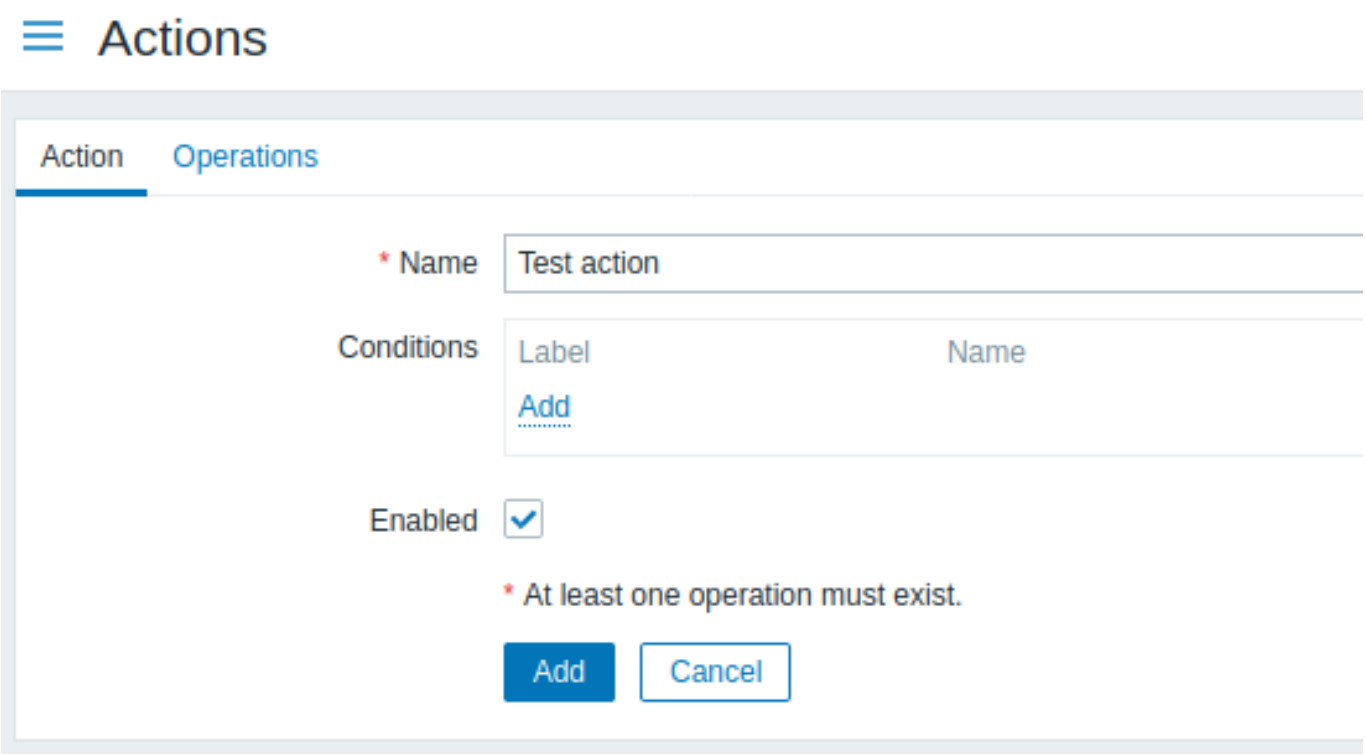


Gdy wszystko będzie gotowe, kliknij *Add* i zapisz formularz.

Teraz skonfigurowałeś *Email* jako działający typ mediów. Typ mediów musi być również powiązany z użytkownikami poprzez zdefiniowanie konkretnych adresów dostarczania (tak jak zrobiliśmy to podczas konfigurowania nowego użytkownika); w przeciwnym razie nie będzie używany.

Nowa akcja

Dostarczanie powiadomień to jedna z rzeczy, które w Zabbiksie wykonują akcje. Dlatego, aby skonfigurować powiadomienie, przejdź do *Alerts* > *Actions* > *Trigger actions* i kliknij *Create action*.

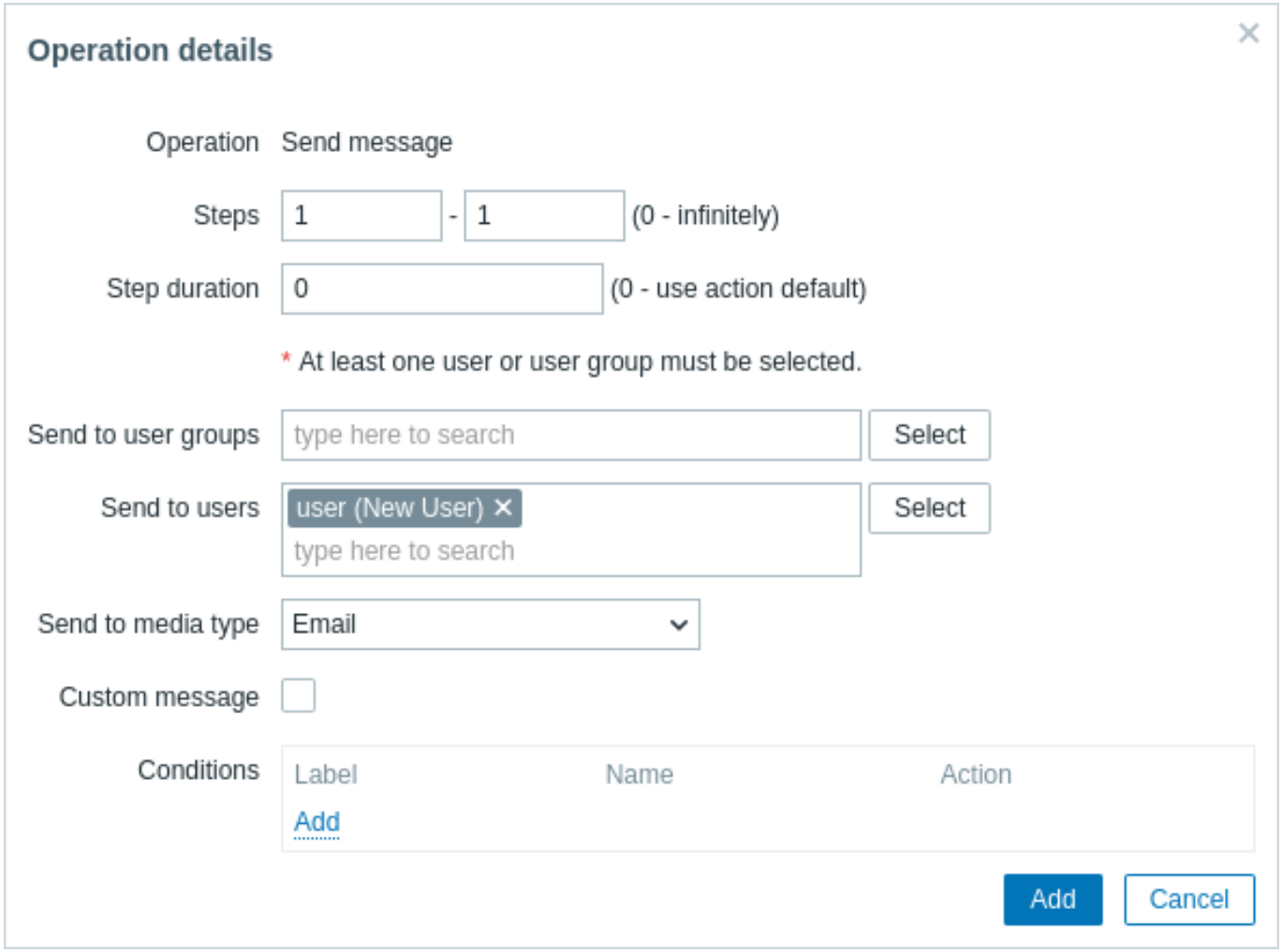


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W tym formularzu wprowadź nazwę akcji.

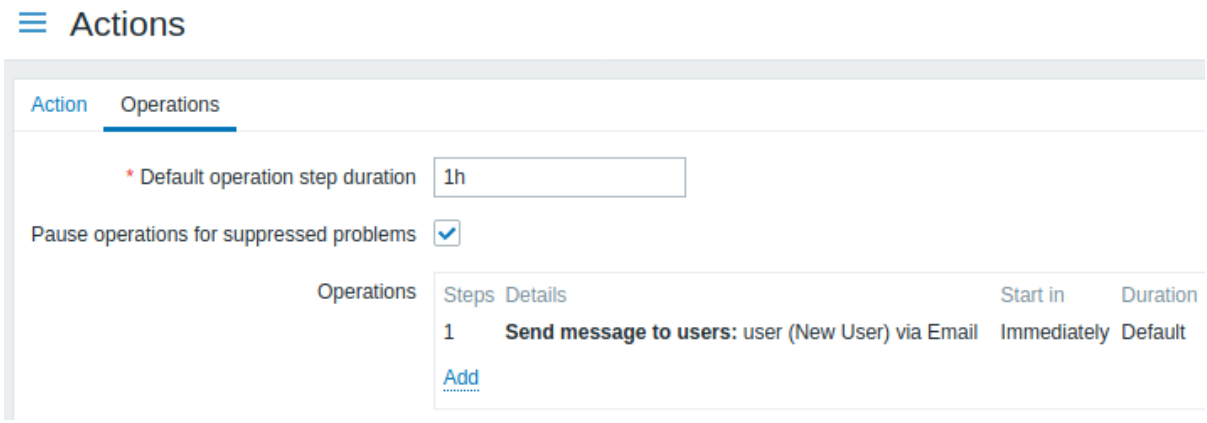
W najprostszym przypadku, jeśli nie dodamy żadnych bardziej szczegółowych warunków, akcja zostanie wykonana przy każdej zmianie wyzwalacza z 'Ok' na 'Problem'.

Musimy jeszcze określić, co akcja ma zrobić - i służy do tego karta *Operations*. Kliknij *Add* w sekcji *Operations*, aby otworzyć nowy formularz operacji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Tutaj kliknij *Select* w sekcji *Send to Users* i wybierz użytkownika ('user'), którego zdefiniowaliśmy. Jako wartość *Send to media type* wybierz "Email". Gdy to zrobisz, kliknij *Add*, a operacja powinna zostać dodana:



To wszystko w przypadku prostej konfiguracji akcji, więc kliknij *Add* w formularzu akcji.

Otrzymywanie powiadomienia

Teraz, gdy skonfigurowano dostarczanie powiadomień, warto faktycznie otrzymać jedno z nich. Aby to ułatwić, możemy celowo zwiększyć obciążenie na naszym host - tak, aby nasz wyzwalacz "zadziałał" i abyśmy otrzymali powiadomienie o problemie.

Otwórz konsolę na swoim host i uruchom:

```
cat /dev/urandom | md5sum
```

Możesz uruchomić jeden lub kilka z tych procesów.

Następnie przejdź do *Monitoring* > *Latest data* i sprawdź, czy wartości 'CPU Load' wzrosły.Pamiętaj, że aby nasz wyzwalacz "zadziałał", wartość 'CPU Load' musi przekraczać '2' przez 3 minuty z rzędu. Gdy tak się stanie:

- w *Monitoring* > *Problems* zobaczysz wyzwalacz z migającym statusem 'Problem';
- otrzymasz powiadomienie o problemie w swojej poczcie e-mail.

Attention:
Jeśli powiadomienia nie działają:
- sprawdź ponownie, czy zarówno ustawienia poczty e-mail, jak i akcja zostały poprawnie skonfigurowane
- upewnij się, że utworzony użytkownik ma co najmniej uprawnienia do odczytu na host, który wygenerował zdarzenie, jak opisano w kroku *Dodawanie użytkownika*. Użytkownik, będąc częścią grupy użytkowników 'Zabbix administrators', musi mieć co najmniej dostęp do odczytu do grupy hostów 'Linux servers', do której należy nasz host.
- Dodatkowo możesz sprawdzić dziennik akcji, przechodząc do *Reports* > *Action log*.

6 Nowy szablon

Przegląd

W tej sekcji dowiesz się, jak skonfigurować szablon.

Wcześniej nauczyliśmy się, jak skonfigurować pozycję, wyzwalacz oraz jak otrzymać powiadomienie o problemie dla hosta.

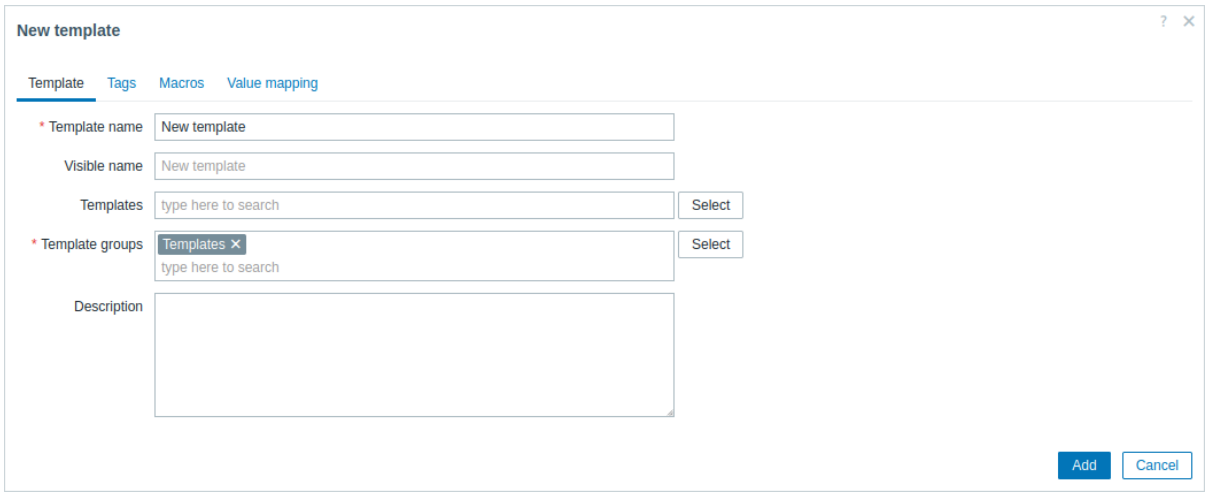
Chociaż wszystkie te kroki same w sobie oferują duży stopień elastyczności, może się wydawać, że wykonanie ich wszystkich to sporo pracy, jeśli trzeba to zrobić na przykład dla tysiąca hostów. Przydałaby się pewna automatyzacja.

Właśnie tutaj z pomocą przychodzą szablony. Szablony umożliwiają grupowanie przydatnych pozycji, wyzwalaczy i innych obiektów, tak aby można było używać ich wielokrotnie, przypisując je do hostów w jednym kroku.

Gdy szablon jest powiązany z hostem, host dziedziczy wszystkie obiekty szablonu. W praktyce oznacza to, że wcześniej przygotowany zestaw kontroli można bardzo szybko zastosować.

Dodawanie szablonu

Aby rozpocząć pracę z szablonami, musimy najpierw utworzyć jeden z nich. W tym celu w *Data collection* > *Templates* kliknij *Create template*. Spowoduje to wyświetlenie formularza konfiguracji szablonu.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Wymagane parametry do wprowadzenia tutaj to:

Template name

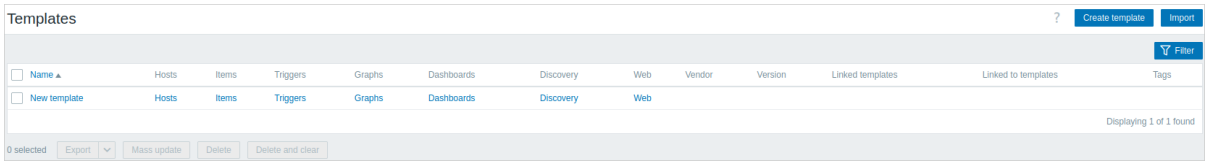
- Wprowadź nazwę szablonu. Dozwolone są znaki alfanumeryczne, spacje i podkreślenia.

Template groups

- Wybierz jedną lub kilka grup, klikając przycisk *Select*. Szablon musi należeć do grupy.

Note:
Uprawnienia dostępu do grup szablonów są przypisywane w konfiguracji user group na karcie **Template permissions** w taki sam sposób jak uprawnienia hosta. Wszystkie uprawnienia dostępu są przypisywane do grup, a nie do poszczególnych szablonów, dlatego dodanie szablonu do co najmniej jednej grupy jest obowiązkowe.

Po zakończeniu kliknij *Add*. Twój nowy szablon powinien być widoczny na liście szablonów. Możesz też użyć filtra, aby znaleźć swój szablon.



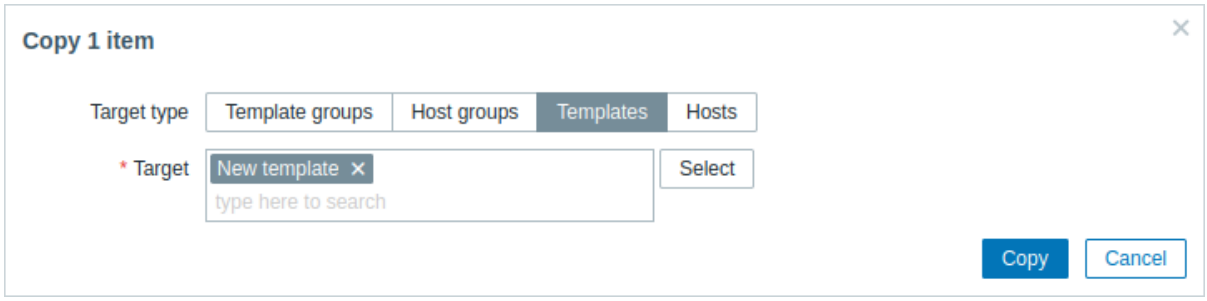
Jak widać, szablon już tam jest, ale nie zawiera niczego - żadnych pozycji, wyzwalaczy ani innych obiektów.

Dodawanie pozycji do szablonu

Aby dodać pozycję do szablonu, otwórz listę pozycji dla „New host", przechodząc do *Data collection* → *Hosts* i klikając *Items* obok „New host".

Następnie:

- Zaznacz pole wyboru przy pozycji „CPU Load" na liście.
- Kliknij *Copy* pod listą.
- Wybierz kartę *Templates*.
- Wybierz szablon, do którego chcesz skopiować pozycję.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

- Kliknij *Copy*.

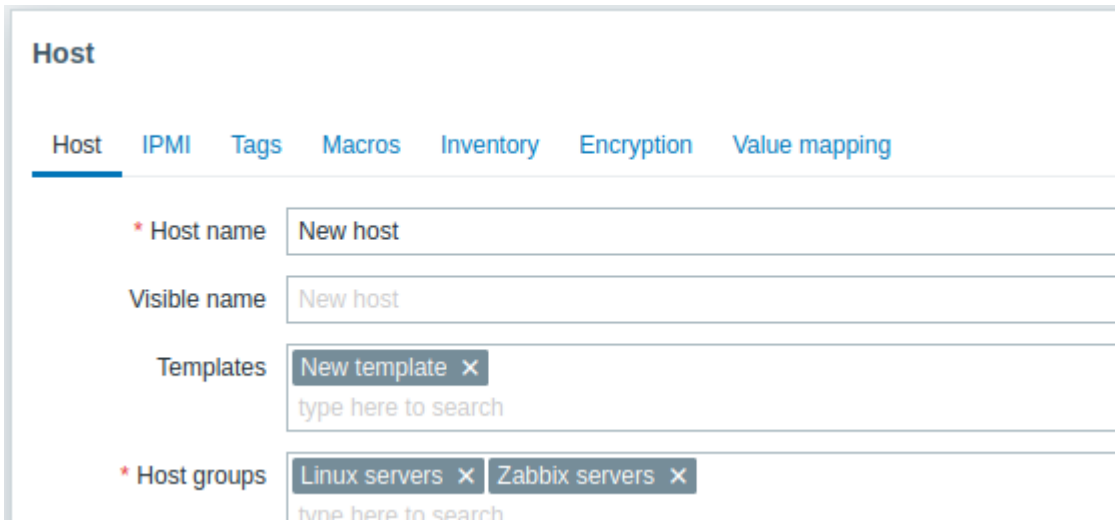
Jeśli teraz przejdziesz do *Data collection* > *Templates*, „New template" powinien zawierać jedną nową pozycję.

Na razie zatrzymamy się na jednej pozycji, ale podobnie możesz dodać do szablonu dowolne inne pozycje, wyzwalacze lub inne elementy, aż będzie on stanowił dość kompletny zestaw obiektów dla danego celu (monitorowanie systemu operacyjnego, monitorowanie pojedynczej aplikacji).

Łączenie szablonu z hostem

Gdy szablon jest gotowy, pozostaje tylko dodać go do hosta. W tym celu przejdź do *Data collection* > *Hosts*, kliknij *New host*, aby otworzyć formularz konfiguracji, i znajdź pole **Templates**.

Zacznij wpisywać *New template* w polu *Templates*. Nazwa utworzonego przez nas szablonu powinna pojawić się na liście rozwijanej. Przewiń w dół, aby ją wybrać. Zobaczysz, że pojawi się w polu *Templates*.



Kliknij *Update* w formularzu, aby zapisać zmiany. Szablon został teraz dodany do hosta wraz ze wszystkimi zawartymi w nim elementami.

W ten sposób można go również zastosować do dowolnego innego hosta. Wszelkie zmiany w pozycjach, wyzwalaczach i innych elementach na poziomie szablonu będą propagowane do hostów, z którymi szablon jest powiązany.

Łączenie predefiniowanych szablonów z hostami

Jak mogłeś zauważyć, Zabbix zawiera zestaw predefiniowanych szablonów dla różnych systemów operacyjnych, urządzeń i aplikacji. Aby bardzo szybko rozpocząć monitorowanie, możesz przypisać odpowiedni z nich do hosta, ale pamiętaj, że te szablony wymagają dostosowania do Twojego środowiska. Niektóre kontrole mogą nie być potrzebne, a interwały odpytywania mogą być zdecydowanie zbyt częste.

Dostępnych jest więcej informacji o szablonach.

6 Urządzenie Zabbix

Przegląd Zabbix appliance umożliwia natychmiastowe wdrożenie serwera Zabbix i frontend, zamiast konfigurowania ich ręcznie lub ponownego wykorzystywania istniejącego serwera dla Zabbix.

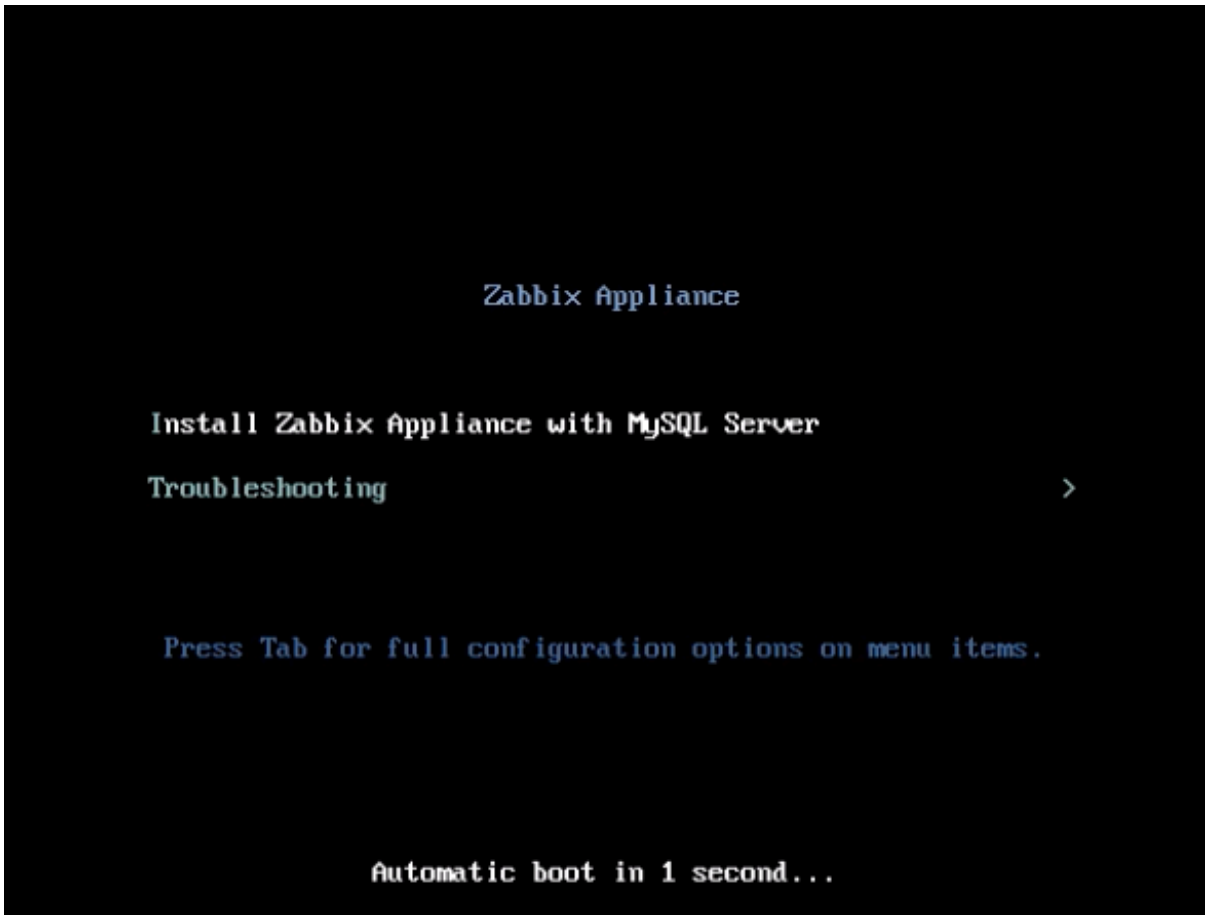
Appliance bazuje na AlmaLinux 8 (x86_64) i zawiera wstępnie skonfigurowany serwer Zabbix działający na MySQL oraz frontend działający na serwerze WWW Nginx.

Attention:
To appliance jest przeznaczone do oceny Zabbix. Nie zaleca się używania go w poważnych środowiskach produkcyjnych.

Obrazy appliance są dostępne do pobrania w następujących formatach:

- Instalacyjny CD/DVD (.iso)
- VMware (.vmx) - zobacz uwagi
- Open virtualization format (.ovf)
- Microsoft Hyper-V (.vhd/.vhdx) - zobacz uwagi
- KVM, Parallels, QEMU, pamięć USB, VirtualBox, Xen (.raw) - zobacz uwagi
- KVM, QEMU (.qcow2)

Menu rozruchowe instalacyjnego CD/DVD Zabbix:



Szybki start Wymagania wstępne

1. Upewnij się, że maszyna hosta ma wystarczające zasoby, aby spełnić wymagania systemowe maszyny wirtualnej:

- *RAM*: 4 GB
- *Miejsce na dysku*: dla maszyny wirtualnej należy przydzielić co najmniej 8 GB
- *CPU*: minimum 2 rdzenie

2. Jeśli oprogramowanie do wirtualizacji służące do uruchamiania obrazu appliance nie jest jeszcze zainstalowane, zainstaluj je (na przykład VirtualBox).
3. Pobierz appliance w formacie obsługiwanym przez używane oprogramowanie do wirtualizacji.
4. Sprawdź ustawienia sieciowe, aby upewnić się, że DHCP jest włączone na maszynie hosta.

Instalacja

1. Uruchom appliance jako maszynę wirtualną z pobranego obrazu.
2. Skonfiguruj ustawienia sieciowe maszyny wirtualnej tak, aby umożliwić dostęp z przeglądarki na maszynie hosta. Można to osiągnąć, włączając *tryb mostkowany*.
3. Zaloguj się do maszyny wirtualnej przy użyciu domyślnych danych uwierzytelniających.
4. Aby uzyskać adres IP, uruchom na maszynie wirtualnej następujące polecenie:

```
ip addr show
```

5. Otwórz przeglądarkę na maszynie hosta i przejdź do adresu IP, który appliance otrzymał przez DHCP.
6. Zaloguj się do Zabbix przy użyciu domyślnych danych uwierzytelniających i rozpocznij monitorowanie.

Konfiguracja Ta sekcja opisuje często wymagane domyślne ustawienia konfiguracji wraz z dostępnymi opcjami dostosowania.

Dane uwierzytelniające

System

- Nazwa użytkownika: root
- Hasło: zabbix

Frontend Zabbix

- Nazwa użytkownika: Admin
- Hasło: zabbix

Po zalogowaniu możesz zmienić domyślne hasło w ustawieniach profilu użytkownika lub utworzyć nowych użytkowników i usunąć domyślnego użytkownika.

Baza danych

Hasła dla wszystkich użytkowników bazy danych są generowane losowo podczas procesu instalacji. Dla bazy danych zdefiniowano następujących użytkowników:

Root:

- Nazwa użytkownika: root
- Hasło: hasło jest przechowywane w pliku `/root/.my.cnf`. Nie jest wymagane podawanie hasła na koncie root.

Serwer Zabbix:

- Nazwa użytkownika: zabbix_srv
- Hasło: hasło jest przechowywane w pliku `/etc/zabbix/zabbix\_server.conf`

Frontend Zabbix:

- Nazwa użytkownika: zabbix_web
- Hasło: hasło jest przechowywane w pliku `/etc/zabbix/web/zabbix.conf.php`

Aby zmienić hasło użytkownika bazy danych, zmodyfikuj je w MySQL oraz w odpowiednim pliku konfiguracyjnym.

Dostęp do frontend

Do frontend Zabbix można uzyskać dostęp pod adresem *http://<IP maszyny wirtualnej>*.

Domyślnie dostęp jest dozwolony z dowolnego miejsca. Aby ograniczyć dostęp, zmodyfikuj plik `/etc/nginx/conf.d/zabbix.conf`. Po zapisaniu zmodyfikowanego pliku uruchom ponownie Nginx, logując się przez SSH jako użytkownik root i wykonując:

```
systemctl restart nginx
```

Statyczny adres IP

Domyślnie appliance używa DHCP do uzyskania adresu IP. Aby ustawić statyczny adres IP:

- Zaloguj się jako użytkownik root.
- Uruchom następujące polecenia, zastępując wartości własnymi adresami IP:

```
nmcli connection modify eth0 ipv4.addresses 192.168.1.10/24    # Adres IP appliance/prefiks CIDR
nmcli connection modify eth0 ipv4.gateway 192.168.1.1     # Adres IP bramy
nmcli connection modify eth0 ipv4.dns 8.8.8.8    # Adres IP serwera DNS
nmcli connection modify eth0 ipv4.method manual
systemctl restart network
```

Alternatywnie możesz otworzyć plik `/etc/NetworkManager/system-connections/eth0.nmconnection` i dodać następujące linie:

```
[ipv4]
address1=192.168.1.10/24,192.168.1.1
dns=8.8.8.8
method=manual
```

Po zapisaniu zmodyfikowanego pliku uruchom polecenie `systemctl restart network`.

Konfiguracja zapory sieciowej

Do zarządzania ustawieniami zapory urządzenie appliance używa iptables ze wstępnie zdefiniowanymi regułami:

- Otwarty port SSH (22 TCP)
- Otwarte porty Zabbix agent (10050 TCP) i Zabbix trapper (10051 TCP)
- Otwarte porty HTTP (80 TCP) i HTTPS (443 TCP)
- Otwarty port pułapek SNMP (162 UDP)
- Otwarte połączenia wychodzące do portu NTP (123 UDP)
- Ograniczenie pakietów ICMP do 5 pakietów na sekundę
- Odrzucanie wszystkich pozostałych połączeń przychodzących

Aby otworzyć dodatkowe porty, zmodyfikuj plik `/etc/sysconfig/iptables` i przeładuj reguły zapory:

```
systemctl reload iptables
```

Repozytoria

Appliance Zabbix używa pakietu *zabbix-release* z repozytorium Zabbix. Repozytoria są skonfigurowane w katalogu `/etc/yum.repos.d/*`.

Strefa czasowa

Domyślnie appliance używa UTC dla zegara systemowego. Aby zmienić strefę czasową, skopiuj odpowiedni plik z `/usr/share/zoneinfo` do `/etc/localtime`, na przykład:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

> **Note:**
> Strefa czasowa frontendu Zabbix jest ustawiana oddzielnie i może zostać zmieniona w ustawieniach frontendu. Domyślna strefa czasowa dla frontendu Zabbix to Europe/Riga.

Lokalizacje plików

- Pliki konfiguracyjne znajdują się w `/etc/zabbix`
- Pliki dziennika serwera Zabbix, proxy i agenta znajdują się w `/var/log/zabbix`
- Zabbix frontend znajduje się w `/usr/share/zabbix`
- Katalog domowy użytkownika *zabbix* to `/var/lib/zabbix`

Usługi systemowe

Usługi systemd są dostępne. Aby wyświetlić listę usług Zabbix, uruchom następujące polecenie na maszynie wirtualnej:

```
systemctl list-units zabbix*
```

Uwagi dotyczące formatu obrazu VMware

Obrazy w formacie *vmdk* mogą być używane bezpośrednio w produktach VMware Player, Server i Workstation. Aby używać ich w ESX, ESXi i vSphere, należy je przekonwertować za pomocą VMware vCenter Converter (do pobrania wymagane jest uwierzytelnienie). Jeśli używasz VMware vCenter Converter, możesz napotkać problemy z hybrydowym adapterem sieciowym. W takim przy-

padku możesz spróbować wskazać adapter E1000 podczas procesu konwersji. Alternatywnie, po zakończeniu konwersji możesz usunąć istniejący adapter i dodać adapter E1000.

Obraz HDD/flash (raw)

Aby uruchomić obraz, wykonaj:

```
dd if=./zabbix_appliance_7.4.0.raw of=/dev/sdc bs=4k conv=fdatasync
```

Zastąp `/dev/sdc` ścieżką urządzenia dysku Flash/HDD.

Hyper-V

Jeśli appliance nie uruchamia się w Hyper-V, spróbuj nacisnąć **Ctrl+Alt+F2**, aby przełączyć sesję TTY.

Rozwiązywanie problemów Jeśli podczas próby zalogowania się do frontend pojawi się komunikat o błędzie *Access denied for user 'replace_user'@'localhost' (using password: YES)*, może to oznaczać, że instalacja jest nadal w toku.



Jeśli błąd nadal występuje po odczekaniu kilku minut lub jeśli zaobserwujesz jakiekolwiek inne nieoczekiwane zachowanie, prawdopodobnie oznacza to, że proces instalacji nie zakończył się pomyślnie. W takim przypadku zalecamy usunięcie bieżącego appliance i ponowne wdrożenie go zgodnie z tą samą instrukcją instalacji. Ten krok zazwyczaj rozwiązuje problem.

Należy pamiętać, że podejmowanie prób ręcznego naprawienia uszkodzonej instalacji nie jest zalecane, ponieważ może prowadzić do dalszych komplikacji.

7 Konfiguracja

Przegląd Konfiguracja w Zabbix obejmuje określenie, które hosty i systemy mają być monitorowane, zdefiniowanie, jakie dane mają być zbierane, oraz ustalenie, w jaki sposób dostarczane są powiadomienia, gdy wystąpią problemy.

Skorzystaj z paska bocznego, aby przejść do odpowiednich aspektów konfiguracji.

Kluczowe zadania konfiguracyjne

- Skonfiguruj nowy host lub grupę hostów → Hosty i grupy hostów
- Zdefiniuj, jakie dane mają być zbierane z hostów → Pozycje
- Utwórz reguły wykrywania problemów → Wykrywanie problemów za pomocą wyzwalaczy
- Skonfiguruj sposób dostarczania powiadomień → Powiadomienia o zdarzeniach
- Wyświetlaj dane monitorowania za pomocą pulpitów, wykresów i map → Wizualizacja
- Używaj ponownie konfiguracji na wielu hostach → Szablony i grupy szablonów, Gotowe szablony
- Kontroluj dostęp użytkowników i uprawnienia → Użytkownicy i grupy użytkowników

1 Konfigurowanie szablonu

Przegląd

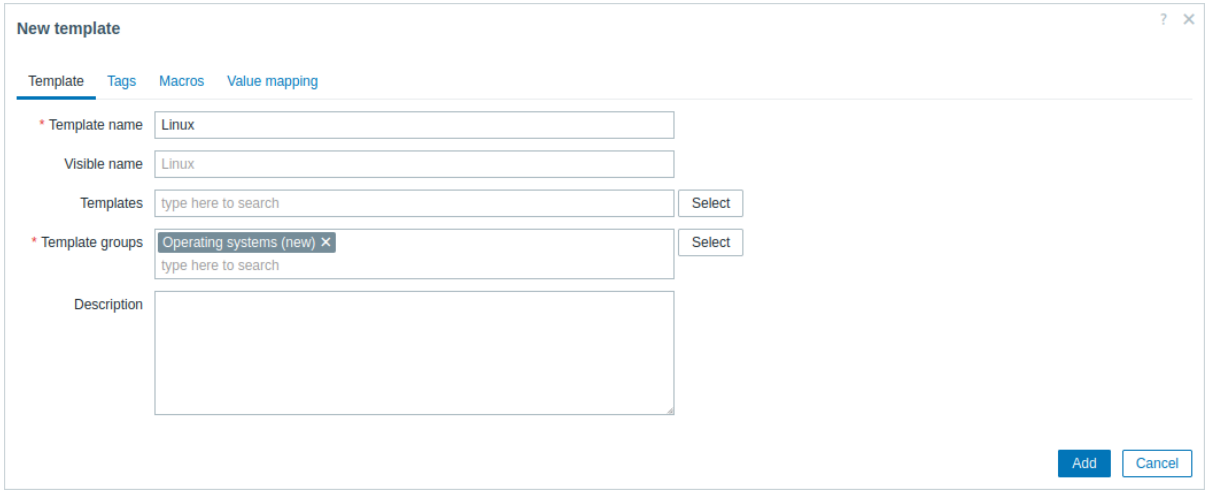
Konfigurowanie szablonu wymaga najpierw utworzenia szablonu przez zdefiniowanie jego ogólnych parametrów, a następnie dodania do niego encji (pozycje, wyzwalacze, wykresy itp.).

Tworzenie szablonu

Aby utworzyć szablon, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Templates*.
2. Kliknij *Create template*.
3. Edytuj atrybuty szablonu.

Karta **Template** zawiera ogólne atrybuty szablonu.

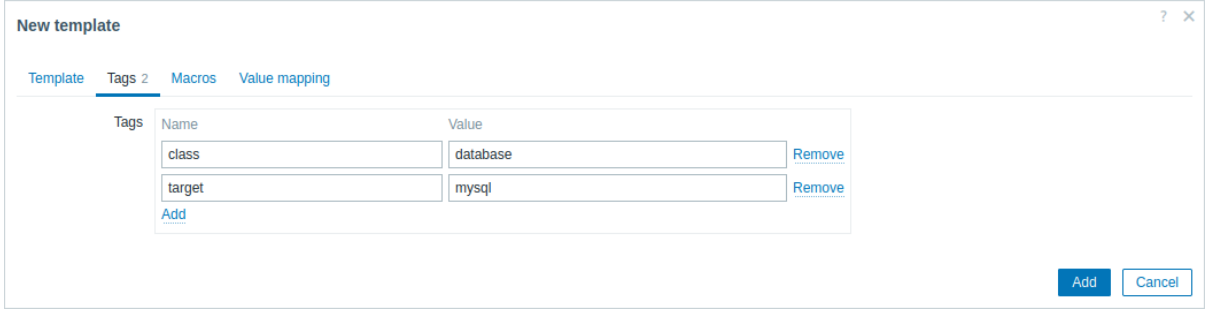


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Atrybuty szablonu:

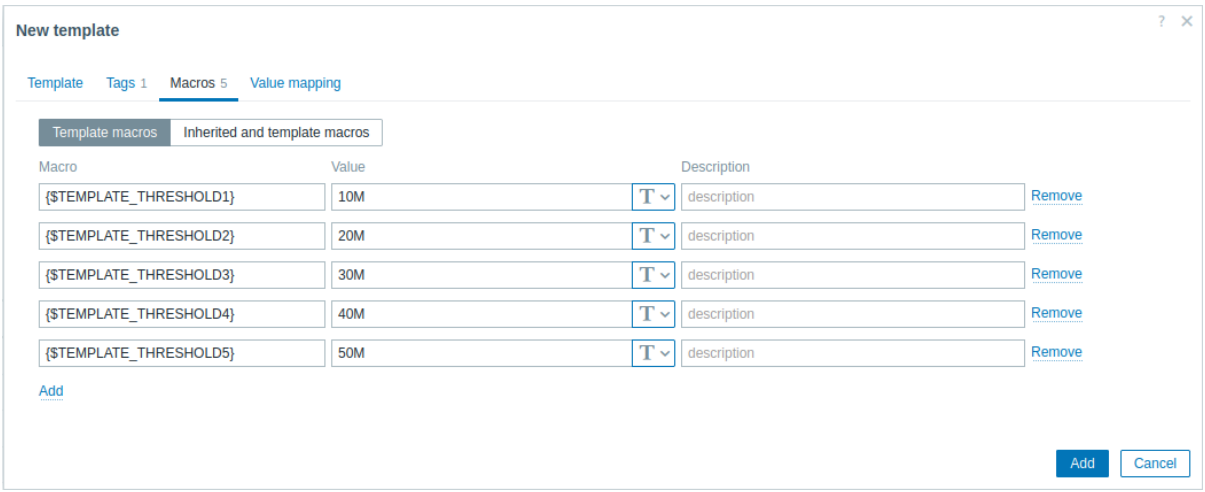
Parameter	Description
Template name	Unikalna nazwa szablonu. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia. Spacje na początku i na końcu nie są dozwolone.
Visible name	Jeśli ustawisz tę nazwę, będzie ona widoczna na listach, mapach itp.
Templates	Połącz z tym szablonem jeden lub więcej szablonów. Wszystkie encje (pozycje, wyzwalacze itp.) zostaną odziedziczone z połączonych szablonów. Aby połączyć nowy szablon, wpisz jego nazwę w polu *Templates* — wyświetli się lista rozwijana pasujących szablonów. Alternatywnie kliknij *Select*, aby otworzyć okno podręczne *Templates*, które początkowo nie zawiera żadnych szablonów. Aby wypełnić listę, wpisz nazwę grupy szablonów w polu *Template group* (i wybierz jedną z sugerowanych pozycji) albo naciśnij przycisk *Select* obok pola *Template group*, aby otworzyć okno podręczne *Template groups*. Po wybraniu grupy szablonów okno *Templates* zostanie zaktualizowane i pokaże szablony należące do tej grupy. Wybrane szablony zostaną połączone po zapisaniu lub zaktualizowaniu konfiguracji. Aby odłączyć szablon, użyj jednej z dwóch opcji w bloku *Templates*: *Unlink* - odłącz szablon, ale zachowaj jego encje (pozycje, wyzwalacze itp.); *Unlink and clear* - odłącz szablon i usuń wszystkie jego encje (pozycje, wyzwalacze itp.).
Template groups	Template groups, do których należy szablon.
Description	Opis szablonu.
Vendor and version	Dostawca i wersja szablonu; wyświetlane tylko podczas aktualizacji istniejących szablonów (out-of-the-box templates dostarczonych przez Zabbix, imported templates lub szablonów zmodyfikowanych za pomocą Template API), jeśli konfiguracja szablonu zawiera takie informacje. Nie można tego zmodyfikować w frontend Zabbix. W przypadku szablonów out-of-the-box wersja jest wyświetlana w następujący sposób: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład 7.0-0, 7.0-5, 7.4-0, 7.4-3.

Karta **Tags** umożliwia zdefiniowanie na poziomie szablonu tags. Wszystkie problemy hostów połączonych z tym szablonem będą oznaczane wartościami wpisanymi tutaj.

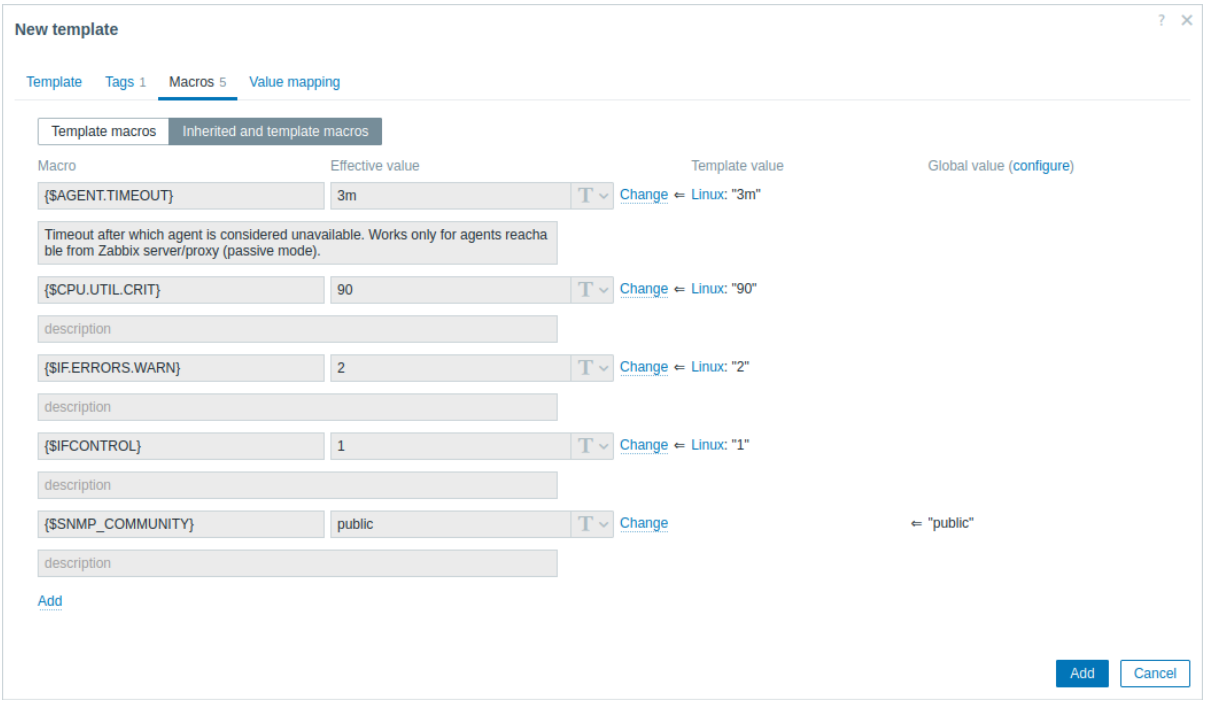


W tagach są obsługiwane makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID}.

Karta **Macros** umożliwia zdefiniowanie na poziomie szablonu user macros jako par nazwa-wartość. Zwróć uwagę, że wartości makr mogą być przechowywane jako zwykły tekst, tekst tajny lub sekret Vault. Obsługiwane jest także dodawanie opisu.



Jeśli wybierzesz opcję *Inherited and template macros*, możesz również wyświetlić makra z połączonych szablonów oraz makra globalne, które szablon odziedziczy, a także wartości, do których te makra zostaną rozwiązane.



Dla wygody dostępne są odnośniki do odpowiednich szablonów, a także odnośnik do konfiguracji makr globalnych. Możliwe jest również edytowanie połączonego makra szablonu lub makra globalnego na poziomie szablonu, co w praktyce tworzy kopię makra w szablonie.

Karta **Value mapping** umożliwia skonfigurowanie przyjaznej dla użytkownika reprezentacji danych pozycji w value mappings.

Przyciski:

Add	Dodaj szablon. Dodany szablon powinien pojawić się na liście.
Update	Zaktualizuj właściwości istniejącego szablonu.
Clone	Utwórz kolejny szablon na podstawie właściwości bieżącego szablonu. Obejmuje to encje (pozycje, wyzwalacze itp.) zarówno odziedziczone z połączonych szablonów, jak i bezpośrednio przypisane do bieżącego szablonu, ale wyklucza dostawcę i wersję bieżącego szablonu, aby sklonowany szablon można było odróżnić od oryginału.

Delete	Usuń szablon; encje szablonu (pozycje, wyzwalacze itp.) pozostaną z połączonymi hostami.
Delete and clear	Usuń szablon i wszystkie jego encje z połączonych hostów.
Cancel	Anuluj edycję właściwości szablonu.

Dodawanie pozycji, wyzwalaczy, wykresów

Attention:
Pozycje muszą najpierw zostać dodane do szablonu. Wyzwalaczy i wykresów nie można dodać bez odpowiedniej pozycji.

Istnieją dwa sposoby dodawania pozycji do szablonu:

1. Aby utworzyć nowe pozycje, postępuj zgodnie z instrukcjami w sekcji Tworzenie pozycji.
2. Aby dodać istniejące pozycje do szablonu:

- Przejdź do *Zbieranie danych* > *Hosty* (lub *Szablony*).
- Kliknij *Pozycje* w wierszu odpowiedniego hosta/szablonu.
- Zaznacz pola wyboru pozycji, które chcesz dodać do szablonu.
- Kliknij *Kopiuj* pod listą pozycji.
- Wybierz szablon (lub grupę szablonów), do którego mają zostać skopiowane pozycje, i kliknij *Kopiuj*.
 Wszystkie zaznaczone pozycje powinny zostać skopiowane do szablonu.

Dodawanie wyzwalaczy i wykresów odbywa się w podobny sposób (z list wyzwalaczy i wykresów), przy czym należy pamiętać, że można je dodać dopiero po wcześniejszym dodaniu wymaganych pozycji.

Dodawanie dashboardów

Aby dodać dashboardy do szablonu w *Zbieranie danych* → *Szablony*, wykonaj następujące czynności:

1. Kliknij *Dashboardy* w wierszu szablonu.
2. Skonfiguruj dashboard zgodnie z wytycznymi dotyczącymi konfigurowania dashboardów.

Attention:
Podczas konfigurowania widżetów na dashboardzie szablonu (zamiast na globalnym dashboardzie) parametry związane z hostem nie są dostępne, a niektóre parametry mają inną etykietę. Dzieje się tak, ponieważ dashboardy szablonów wyświetlają dane tylko z hosta, z którym szablon jest powiązany. Na przykład parametry *Host groups*, *Exclude host groups* i *Hosts* w widżecie *Problems* nie są dostępne, parametr *Host groups* w widżecie *Host availability* nie jest dostępny, a parametr *Show hosts in maintenance* został zmieniony na *Show data in maintenance* itd. Więcej informacji o dostępności parametrów w widżetach dashboardów szablonów można znaleźć w opisie poszczególnych parametrów dla każdego widżetu dashboardu.

Note:
Szczegółowe informacje o dostępie do dashboardów hostów, które są tworzone na podstawie dashboardów szablonów, znajdują się w sekcji dashboardy hostów.

Konfigurowanie reguł niskopoziomowego wykrywania

Zobacz sekcję niskopoziomowe wykrywanie w podręczniku.

Dodawanie scenariuszy web

Aby dodać scenariusze web do szablonu w *Data collection* → *Templates*, wykonaj następujące czynności:

1. Kliknij *Web* w wierszu szablonu.
2. Skonfiguruj scenariusz web, postępując zgodnie ze standardową metodą konfigurowania scenariuszy web.

2 Konfigurowanie grupy szablonów

Overview

Template groups are used for the logical grouping of templates and assigning user permissions to them.

Each template must have at least one template group assigned. A template may belong to multiple template groups, and each template group may contain multiple templates.

Note that in Zabbix, all permissions are based on groups: user groups, host groups, and template groups. So, even if a single user needs access to a single template, it is granted by adding the user to a user group that has permission to access the template group containing that template.

Konfiguracja

> **Attention:**
> Tylko użytkownicy typu Super admin mogą tworzyć grupy szablonów.

Istnieją dwie opcje tworzenia grupy szablonów w frontend Zabbix.

Opcja pierwsza:

- Przejdź do: *Data collection* → *Template groups*
- Kliknij *Create template group* w prawym górnym rogu ekranu
- Wprowadź nazwę grupy w formularzu



Opcja druga: podczas konfigurowania szablonu wprowadź nieistniejącą nazwę grupy w polu wejściowym *Template groups*.

Po utworzeniu grupy szablonów możesz kliknąć nazwę szablonu na liście w sekcji *Data collection* → *Template groups*, aby edytować nazwę grupy, sklonować grupę lub usunąć grupę.

Usunięcie grupy szablonów powoduje usunięcie tylko grupy logicznej, a nie szablonów w tej grupie. Nie można usunąć grupy szablonów, która jest jedyną grupą dla jakiegokolwiek istniejącego szablonu.

Tworzenie podgrup szablonów

Podgrupa szablonu (lub zagnieżdżona grupa szablonów) jest elementem podrzędnym nadrzędnej grupy szablonów, która ją zawiera.

Podgrupę tworzy się przy użyciu ukośnika '/' w polu wprowadzania nazwy grupy, aby określić jej relację do grupy nadrzędnej lub grup nadrzędnych. Na przykład:

- wpisanie `Linux servers/Databases` tworzy podgrupę `Linux servers/Databases` w nadrzędnej grupie `Linux servers`.
- wpisanie `Linux servers/Databases/MySQL/Tokyo` tworzy odpowiednią podgrupę w zagnieżdżonych grupach nadrzędnych `Linux servers`, `Linux servers/Databases`, `Linux servers/Databases/MySQL`.

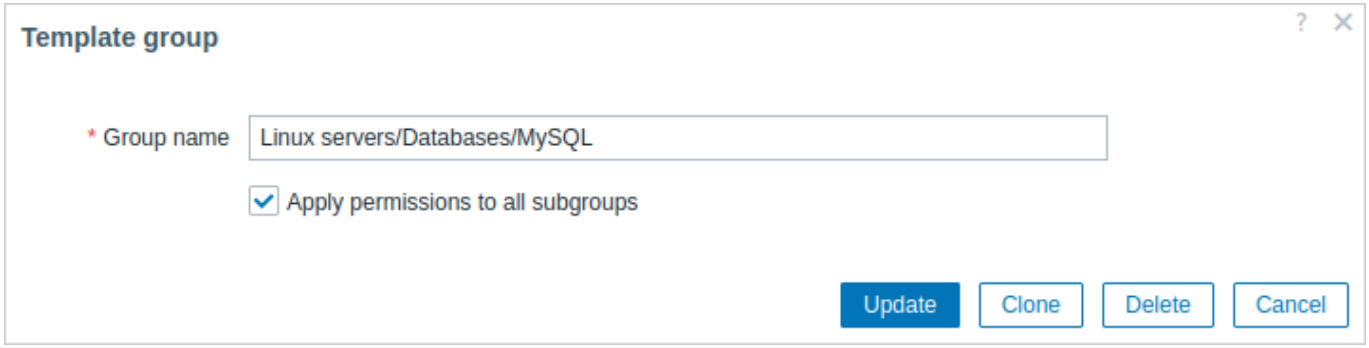
Podczas tworzenia podgrupy nie można używać ukośników na początku ani na końcu nazwy ani kilku ukośników z rzędu. Znak '/' nie obsługuje sekwencji ucieczki.

Nie jest wymagane tworzenie żadnych nadrzędnych grup szablonów przed utworzeniem podgrupy. Możesz zdecydować, czy zacząć od utworzenia podgrupy (na przykład `Linux servers/Databases`), czy dowolnej nadrzędnej grupy szablonów (w naszym przykładzie `Linux servers`). Jeśli zaczniesz od utworzenia podgrupy, nadrzędne grupy szablonów **nie** zostaną utworzone automatycznie.

Uprawnienia do grup szablonów

- Podczas tworzenia podgrupy w istniejącej nadrzędnej grupie szablonów (na przykład tworzenia `Linux servers/Databases`, gdy `Linux servers` już istnieje), uprawnienia grupy użytkowników do podgrupy są dziedziczone z grupy nadrzędnej.
- Podczas tworzenia nadrzędnej grupy szablonów dla istniejącej podgrupy (na przykład tworzenia `Linux servers`, gdy `Linux servers/Databases` już istnieje), dla grupy nadrzędnej nie są ustawiane żadne uprawnienia.

Podczas edycji dowolnej grupy szablonów można również ustawić dodatkową opcję: *Zastosuj uprawnienia do wszystkich podgrup*.



Zaznaczenie tego pola wyboru i kliknięcie *Aktualizuj* spowoduje zastosowanie tego samego poziomu uprawnień do wszystkich bieżących i przyszłych podgrup edytowanej grupy szablonów.

Jeśli więc jakimkolwiek grupom użytkowników nadano różne uprawnienia do podgrup edytowanej grupy szablonów, zaznaczenie tego pola wyboru nada wszystkim bieżącym i przyszłym podgrupom takie same uprawnienia użytkowników jak edytowanej grupie.

Należy pamiętać, że ta opcja nie jest zapisywana w bazie danych i zastąpi istniejące uprawnienia. Wszelkie zmiany wprowadzone za pomocą tej opcji można cofnąć wyłącznie ręcznie.

3 Podłączanie/odłączanie

Przegląd

Podłączanie to proces, w którym szablony są stosowane do hostów, natomiast odłączanie usuwa powiązanie z szablonem z hosta.

Podłączanie szablonu

Aby podłączyć szablon do hosta, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Hosts*.
2. Kliknij wymagany host.
3. Zacznij wpisywać nazwę szablonu w polu *Templates*. Pojawi się lista pasujących szablonów; przewiń w dół, aby wybrać odpowiedni. Alternatywnie możesz kliknąć *Select* obok pola i wybrać jeden lub kilka szablonów z listy w wyskakującym oknie.
4. Kliknij *Add/Update* w formularzu atrybutów hosta.

Host będzie teraz miał wszystkie encje szablonu. Obejmuje to pozycje, wyzwalacze, wykresy, reguły wykrywania niskiego poziomu, scenariusze WWW, a także pulpity.

Attention:
Podłączenie wielu szablonów do tego samego hosta zakończy się niepowodzeniem, jeśli szablony te zawierają pozycje z takim samym kluczem pozycji. Ponadto, ponieważ wyzwalacze i wykresy używają pozycji, również one nie mogą zostać podłączone do jednego hosta z wielu szablonów, jeśli używane są identyczne klucze pozycji.

Gdy encje (pozycje, wyzwalacze itp.) są dodawane z szablonu:

- wcześniej istniejące identyczne encje na hoście są aktualizowane jako encje szablonu, a **wszelkie istniejące dostosowania encji na poziomie hosta zostają utracone**;
- encje z szablonu są dodawane;
- wszelkie bezpośrednio podłączone encje, które przed podłączeniem szablonu istniały tylko na hoście, pozostają bez zmian.

Na listach wszystkie encje pochodzące z szablonu są teraz poprzedzone nazwą szablonu, co wskazuje, że należą do danego szablonu. Sama nazwa szablonu (szarym tekstem) jest linkiem umożliwiającym dostęp do listy tych encji na poziomie szablonu.

Note:
Dla niektórych pozycji, takich jak external checks, HTTP agent checks, simple checks, SSH checks oraz Telnet checks, interfejs hosta jest opcjonalny. Jeśli w momencie podłączania szablonu host nie ma zdefiniowanego interfejsu, te pozycje zostaną dodane bez interfejsu. Jeśli później dodasz interfejs hosta, nie zostanie on automatycznie przypisany do już istniejących pozycji. Aby przypisać nowo dodany interfejs hosta do wszystkich pozycji szablonu jednocześnie, odłącz szablon od hosta, a następnie podłącz go ponownie. Aby zachować historię pozycji, użyj opcji *Unlink*, nie używaj *Unlink and clear*.

Jeśli jakaś encja nie jest poprzedzona nazwą szablonu, oznacza to, że istniała na hoście wcześniej i nie została dodana przez szablon.

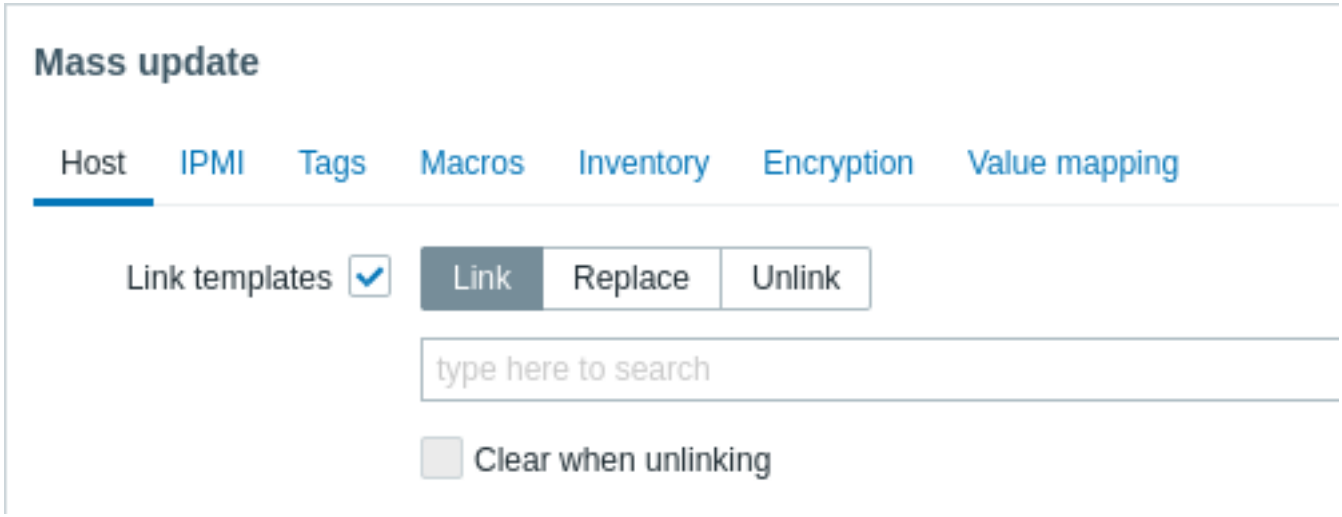
Kryteria unikalności encji

Podczas dodawania encji (pozycji, wyzwalaczy itp.) z szablonu ważne jest, aby wiedzieć, które z tych encji już istnieją na hoście i wymagają aktualizacji, a które encje się różnią. Kryteria unikalności służące do określenia, czy encje są takie same, czy różne, są następujące:

- dla pozycji - klucz pozycji;
- dla wyzwalaczy - nazwa wyzwalacza i wyrażenie;
- dla niestandardowych wykresów - nazwa wykresu i jego pozycje.

Podłączanie szablonów do wielu hostów

Aby zaktualizować powiązania szablonów dla wielu hostów, w *Data collection* → *Hosts* wybierz kilka hostów, zaznaczając ich pola wyboru, a następnie kliknij **Mass update** pod listą i wybierz *Link templates*:



Aby podłączyć dodatkowe szablony, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się rozwijana lista z pasującymi szablonami. Następnie przewiń listę i wybierz szablon do podłączenia.

Opcja *Replace* umożliwia podłączenie nowego szablonu przy jednoczesnym odłączeniu wszystkich szablonów, które były wcześniej podłączone do hostów. Opcja *Unlink* pozwala określić, które szablony mają zostać odłączone. Opcja *Clear when unlinking* pozwala nie tylko odłączyć wcześniej podłączone szablony, ale także usunąć wszystkie encje po nich odziedziczone (pozycje, wyzwalacze itd.).

Note:
Zabbix oferuje obszerny zestaw predefiniowanych szablonów. Można używać ich jako punktu odniesienia, jednak należy uważać z używaniem ich bez zmian w środowisku produkcyjnym, ponieważ mogą zawierać zbyt wiele pozycji i zbyt często odpytywać o dane. Jeśli chcesz ich używać, dostrój je do rzeczywistych potrzeb.

Edycja podlinkowanych encji

Jeśli spróbujesz edytować pozycja lub wyzwalacz, które zostały podlinkowane z szablon, możesz zauważyć, że wiele kluczowych opcji edycji jest wyłączonych. Ma to sens, ponieważ idea szablonów polega na tym, że elementy są edytowane centralnie na poziomie szablonu. Nadal jednak możesz na przykład włączać/wyłączać pozycja na poszczególnych hostach oraz ustawiać interwał aktualizacji, długość historii i niektóre inne parametry.

Jeśli chcesz w pełni edytować encję, musisz zrobić to na poziomie szablonu (skrót do poziomu szablonu jest wyświetlany w nazwie formularza), pamiętając, że te zmiany wpłyną na wszystkie hosty, które mają podlinkowany ten szablon.

Attention:
Wszelkie dostosowania encji wprowadzone na poziomie szablonu zastąpią wcześniejsze dostosowania encji na poziomie hosta.

Odłączanie szablonu

Aby odłączyć szablon od hosta, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Hosts*.
2. Kliknij wymagany host i znajdź pole *Templates*.
3. Kliknij *Unlink* lub *Unlink and clear* obok szablonu, aby odłączyć szablon.
4. Kliknij *Update* w formularzu atrybutów hosta.

Wybranie opcji *Unlink* spowoduje jedynie usunięcie powiązania z szablonem, przy jednoczesnym pozostawieniu wszystkich jego encji przy hoście. Obejmuje to pozycje, wyzwalacze, wykresy, reguły wykrywania niskiego poziomu i scenariusze WWW, ale nie obejmuje pulpitów. Należy pamiętać, że mapy wartości i tagi odziedziczone z podłączonych szablonów również zostaną usunięte.

Wybranie opcji *Unlink and clear* spowoduje usunięcie zarówno powiązania z szablonem, jak i wszystkich jego encji (pozycji, wyzwalaczy itd.).

4 Zagnieżdżanie

Przegląd

Zagnieżdżanie to sposób, w jaki jeden szablon obejmuje jeden lub więcej innych szablonów.

Ponieważ sensowne jest rozdzielenie encji na osobne szablony dla różnych usług, aplikacji itp., możesz ostatecznie mieć całkiem sporo szablonów, z których wszystkie mogą wymagać podłączenia do wielu hostów. Aby uprościć ten obraz, możliwe jest połączenie kilku szablonów razem w jednym szablonie.

Korzyść z zagnieżdżania polega na tym, że do hosta trzeba podłączyć tylko jeden szablon, a host automatycznie odziedziczy wszystkie encje z szablonów, które są podłączone do tego jednego szablonu. Na przykład, jeśli podłączymy *T1* i *T2* do *T3*, uzupełnimy *T3* o wszystkie encje z *T1* i *T2*, ale nie odwrotnie. Jeśli podłączymy *T1* do *T2* i *T3*, uzupełnimy *T2* i *T3* o encje z *T1*.

Konfigurowanie zagnieżdżonych szablonów

Aby połączyć szablony, należy wybrać istniejący szablon (lub utworzyć nowy), a następnie:

1. Otworzyć formularz konfiguracji szablonu.
2. Odszukać pole *Templates*.
3. Kliknąć *Select*, aby otworzyć wyskakujące okno *Templates*.
4. W wyskakującym oknie wybrać wymagane szablony, a następnie kliknąć *Select*, aby dodać szablony do listy.
5. Kliknąć *Add* lub *Update* w formularzu konfiguracji szablonu.

W rezultacie wszystkie encje skonfigurowanego szablonu, a także wszystkie encje połączonych szablonów będą teraz widoczne w konfiguracji szablonu. Obejmuje to pozycje, wyzwalacze, wykresy, reguły wykrywania niskiego poziomu oraz scenariusze WWW, ale nie obejmuje pulpitów. Jednak pulpity połączonych szablonów będą mimo to dziedziczone przez hosty.

Aby odłączyć dowolny z połączonych szablonów, kliknij *Unlink* lub *Unlink and clear* w formularzu konfiguracji szablonu, a następnie kliknij *Update*.

Opcja *Unlink* po prostu usunie powiązanie z połączonym szablonem, nie usuwając żadnych jego encji (pozycji, wyzwalaczy itd.).

Opcja *Unlink and clear* usunie zarówno powiązanie z połączonym szablonem, jak i wszystkie jego encje (pozycje, wyzwalacze itd.).

5 Masowa aktualizacja

Przegląd

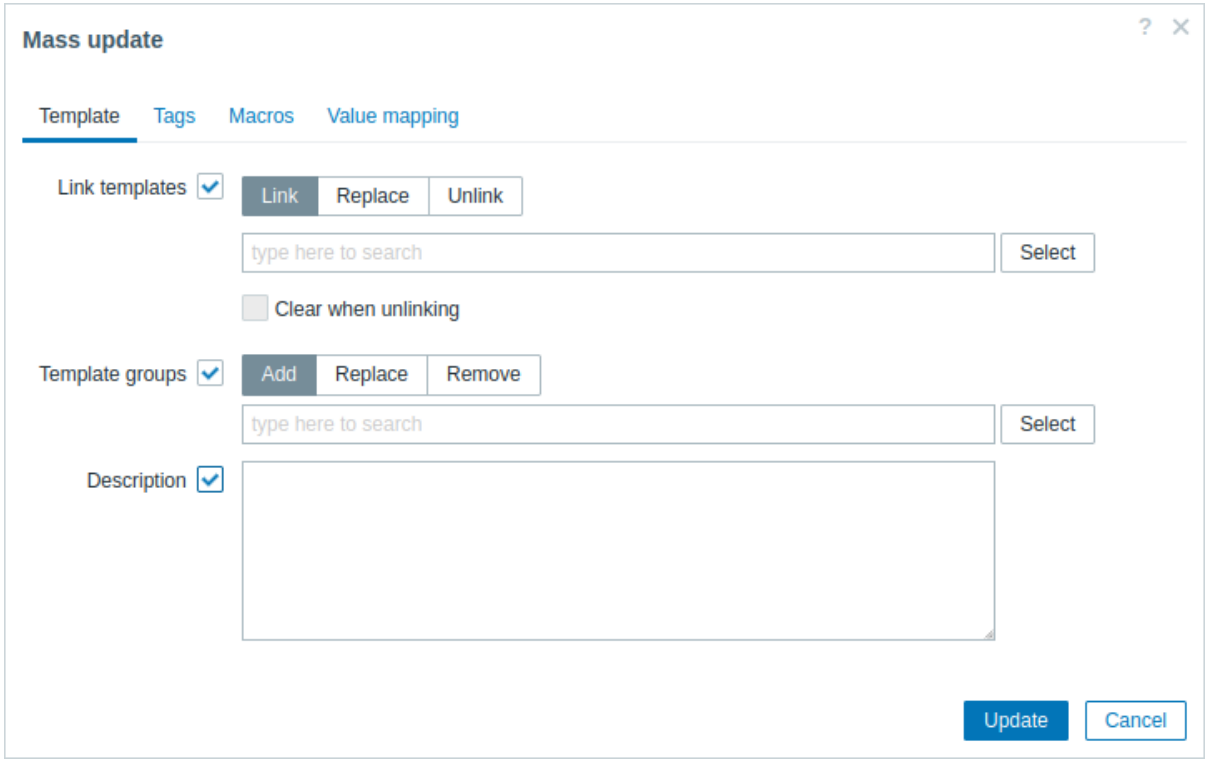
Czasami możesz chcieć zmienić jakiś atrybut dla wielu szablonów jednocześnie. Zamiast otwierać każdy pojedynczy szablon do edycji, możesz w tym celu użyć funkcji masowej aktualizacji.

Używanie masowej aktualizacji

Aby masowo zaktualizować niektóre szablony, wykonaj następujące czynności:

1. Zaznacz pola wyboru przy szablonach, które chcesz zaktualizować, na liście szablonów.
2. Kliknij *Mass update* poniżej listy.
3. Przejdź do zakładki z wymaganymi atrybutami (*Template*, *Tags*, *Macros* lub *Value mapping*).
4. Zaznacz pola wyboru przy atrybutach, które chcesz zaktualizować, i wprowadź dla nich nową wartość.

Karta **Szablon** zawiera ogólne opcje masowej aktualizacji szablonów.



Po wybraniu odpowiedniego przycisku dla aktualizacji *Podłącz szablony* dostępne są następujące opcje:

- *Podłącz* — określa, które dodatkowe szablony należy podłączyć;
- *Zastąp* — określa, które szablony należy podłączyć, jednocześnie odłączając wszystkie wcześniej podłączone szablony;
- *Odłącz* — określa, które szablony należy odłączyć.

Aby określić szablony do podłączenia/odłączenia, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się lista rozwijana z pasującymi szablonami. Wystarczy przewinąć w dół, aby wybrać wymagane szablony.

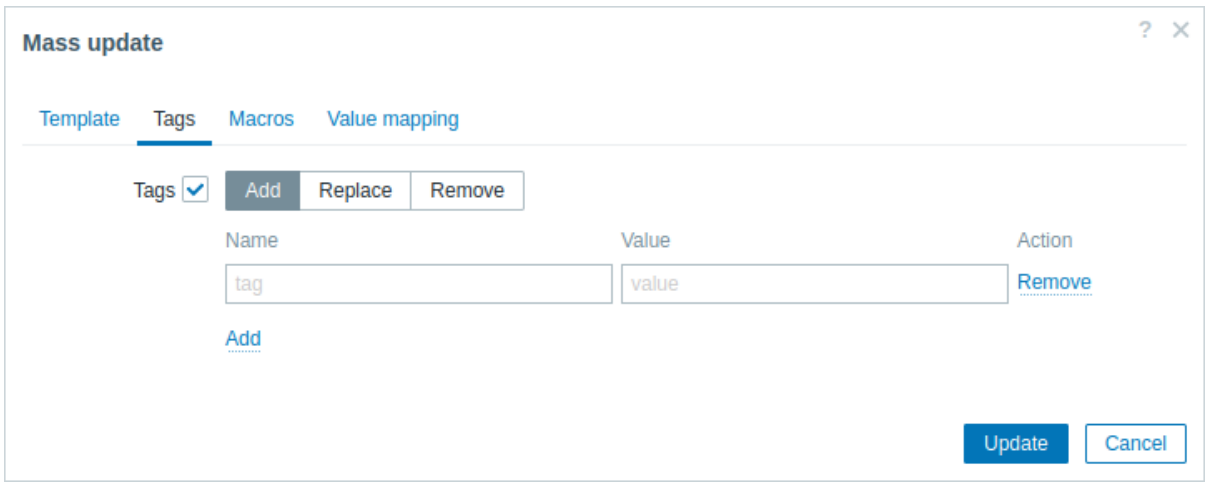
Opcja *Wyczyść przy odłączaniu* umożliwia odłączenie wszystkich wcześniej podłączonych szablonów, a także usunięcie wszystkich elementów odziedziczonych z tych szablonów (pozycji, wyzwalaczy, wykresów itp.).

Po wybraniu odpowiedniego przycisku dla aktualizacji *Grupy szablonów* dostępne są następujące opcje:

- *Dodaj* — umożliwia określenie dodatkowych grup szablonów spośród istniejących lub wprowadzenie całkowicie nowych grup szablonów dla szablonów;
- *Zastąp* — usunie szablon z wszystkich istniejących grup szablonów i zastąpi je grupą/grupami określoną/określonymi w tym polu (istniejącymi lub nowymi grupami szablonów);
- *Usuń* — usunie określone grupy szablonów z szablonów.

Pola te obsługują autouzupełnianie — rozpoczęcie wpisywania powoduje wyświetlenie listy rozwijanej z pasującymi grupami szablonów. Jeśli grupa szablonów jest nowa, również pojawi się na liście rozwijanej i będzie oznaczona dopiskiem *(new)* po nazwie. Wystarczy przewinąć w dół, aby ją wybrać.

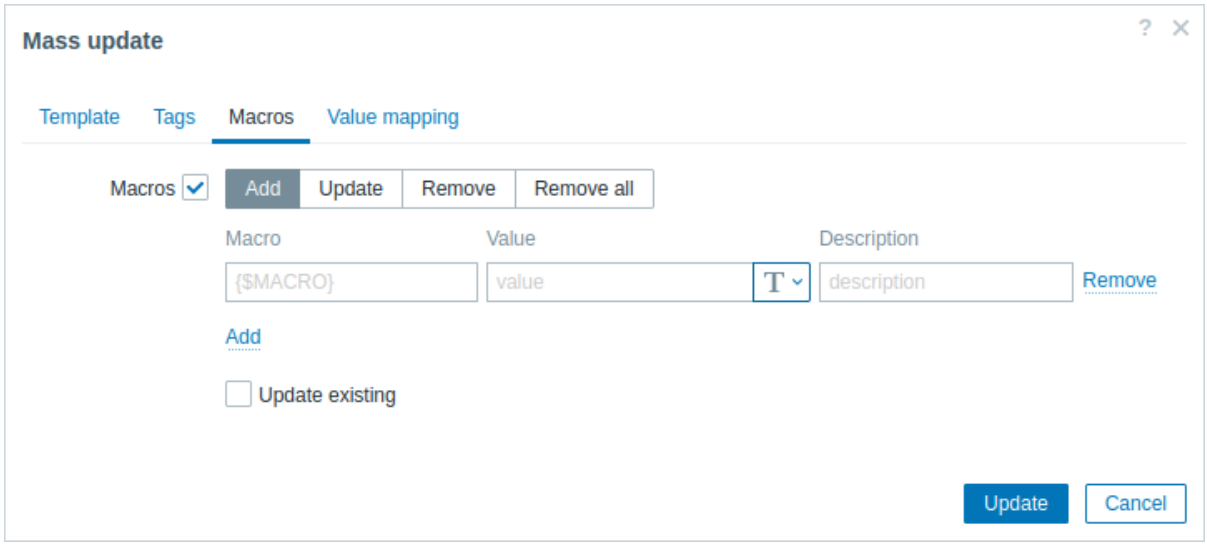
Zakładka **Tagi** umożliwia masową aktualizację tagów na poziomie szablonu.



Makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT}

oraz {HOST.ID} są obsługiwane w tagach. Należy pamiętać, że tagi o tej samej nazwie, ale o różnych wartościach, nie są uznawane za „duplikaty” i mogą zostać dodane do tego samego szablonu.

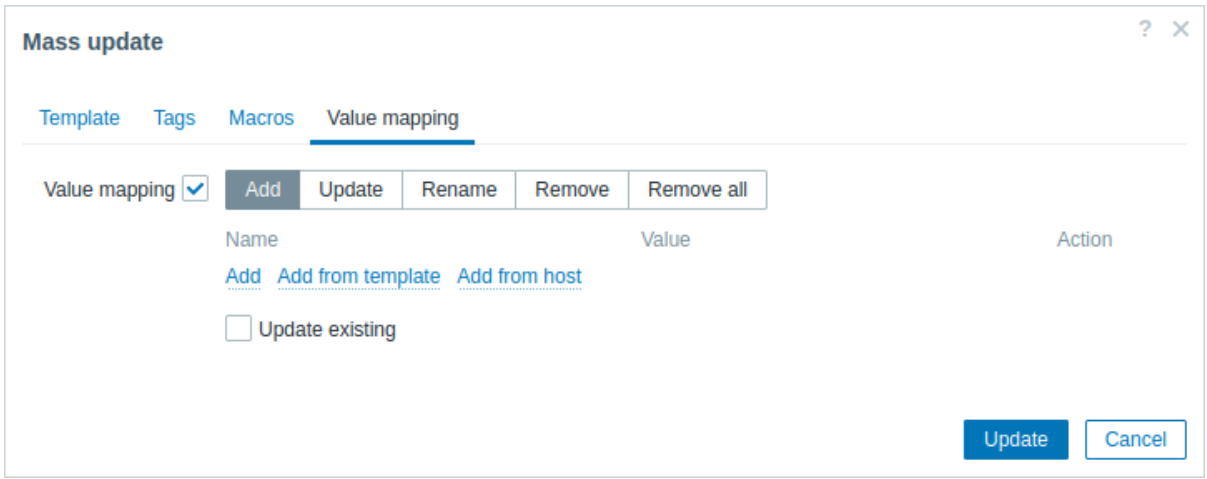
Karta **Makra** umożliwia masową aktualizację makr na poziomie szablonu.



Dostępne są następujące opcje po wybraniu odpowiedniego przycisku aktualizacji makr:

- *Dodaj* — umożliwia określenie dodatkowych makr użytkownika dla szablonów. Jeśli pole wyboru *Aktualizuj istniejące* jest zaznaczone, wartość, typ i opis dla podanej nazwy makra zostaną zaktualizowane. Jeśli nie jest zaznaczone i makro o tej nazwie już istnieje w szablonie(ach), nie zostanie zaktualizowane.
- *Aktualizuj* — zastąpi wartości, typy i opisy makr określonych na tej liście. Jeśli pole wyboru *Dodaj brakujące* jest zaznaczone, makro, które wcześniej nie istniało w szablonie, zostanie dodane jako nowe makro. Jeśli nie jest zaznaczone, zaktualizowane zostaną tylko makra, które już istnieją w szablonie.
- *Usuń* — usunie określone makra z szablonów. Jeśli pole *Z wyjątkiem wybranych* jest zaznaczone, wszystkie makra z wyjątkiem tych określonych na liście zostaną usunięte. Jeśli nie jest zaznaczone, usunięte zostaną tylko makra określone na liście.
- *Usuń wszystko* — usunie wszystkie makra użytkownika z szablonów. Jeśli pole *Potwierdzam usunięcie wszystkich makr* nie jest zaznaczone, otworzy się nowe wyskakujące okno z prośbą o potwierdzenie usunięcia wszystkich makr.

Zakładka **Mapowanie wartości** umożliwia masową aktualizację mapowań wartości.



Dostępne są przyciski z następującymi opcjami aktualizacji mapowań wartości:

- *Dodaj* — dodaje mapowania wartości do szablonów. Jeśli zaznaczysz opcję *Aktualizuj istniejące*, wszystkie właściwości mapowania wartości o tej nazwie zostaną zaktualizowane. W przeciwnym razie, jeśli mapowanie wartości o tej nazwie już istnieje, nie zostanie zaktualizowane.
- *Aktualizuj* — aktualizuje istniejące mapowania wartości. Jeśli zaznaczysz opcję *Dodaj brakujące*, mapowanie wartości, które wcześniej nie istniało w szablonie, zostanie dodane jako nowe mapowanie wartości. W przeciwnym razie zaktualizowane zostaną tylko te mapowania wartości, które już istnieją w szablonie.
- *Zmień nazwę* — nadaje nową nazwę istniejącemu mapowaniu wartości.
- *Usuń* — usuwa określone mapowania wartości z szablonów. Jeśli zaznaczysz opcję *Z wyjątkiem wybranych*, wszystkie mapowania wartości zostaną usunięte **z wyjątkiem** tych, które zostały wskazane.
- *Usuń wszystkie* — usuwa wszystkie mapowania wartości z szablonów. Jeśli pole wyboru *Potwierdzam usunięcie wszystkich mapowań wartości* nie jest zaznaczone, otworzy się nowe wyskakujące okno z prośbą o potwierdzenie usunięcia.

Opcje *Dodaj z szablonu* i *Dodaj z hosta* są dostępne dla operacji dodawania/aktualizacji mapowań wartości. Umożliwiają one wybór mapowań wartości odpowiednio z szablonu lub hosta.

Po wprowadzeniu wszystkich wymaganych zmian kliknij *Aktualizuj*. Atrybuty zostaną odpowiednio zaktualizowane dla wszystkich wybranych szablonów.

1 Hosty i grupy hostów

Czym jest „host”?

W Zabbix „host” oznacza dowolne urządzenie fizyczne lub wirtualne, aplikację, usługę albo dowolny inny logicznie powiązany zbiór monitorowanych parametrów.

Tworzenie hostów jest jednym z pierwszych zadań związanych z monitorowaniem w Zabbix. Na przykład, jeśli chcesz monitorować niektóre parametry na serwerze „x”, musisz najpierw utworzyć host o nazwie, na przykład, „Serwer X”, a następnie możesz dodać do niego pozycje monitorowania.

Hosty są organizowane w grupy hostów.

Przejdź do:

- Kreator hosta — aby skorzystać z prowadzonego krok po kroku interfejsu do tworzenia i konfigurowania nowego lub istniejącego hosta.
- Konfigurowanie hosta — aby skorzystać z klasycznego podejścia do tworzenia i konfigurowania hosta.

1 Kreator hosta

Przegląd

Kreator hosta to interfejs prowadzony krok po kroku, służący do konfigurowania monitorowanego obiektu (urządzenia, aplikacji, usługi itp.) w Zabbix. Prowadzi on przez następujące kroki:

- Wybór szablonu
- Utworzenie lub wybranie hosta
- Instalacja Zabbix agent lub agent 2
- Dodanie interfejsu hosta
- Zastosowanie dodatkowej konfiguracji do monitorowanego obiektu lub hosta Zabbix (jeśli jest wymagana przez szablon)

Aby uzyskać dostęp do Kreatora hosta w frontendzie Zabbix, wykonaj następujące czynności:

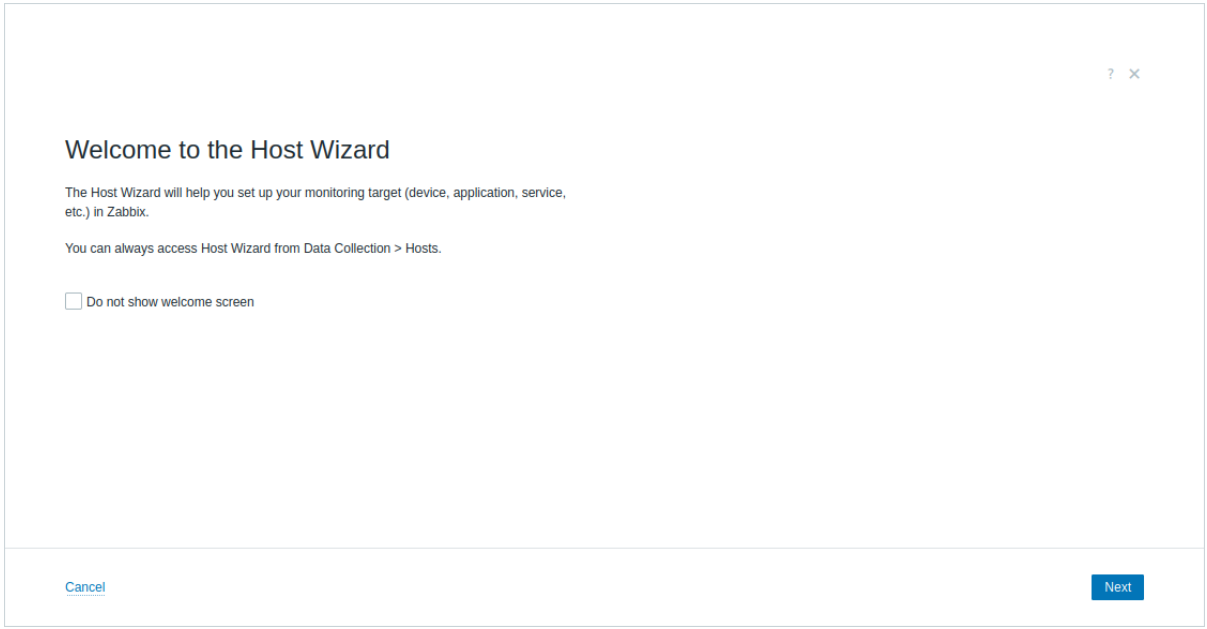
- Przejdź do: *Data collection* > *Hosts*
- Kliknij *Host Wizard* w prawym górnym rogu ekranu
- Postępuj zgodnie z instrukcjami w kreatorze

Możesz także edytować istniejącego hosta, klikając jego nazwę (lub ikonę trzech kropek obok niej) w różnych sekcjach frontendu, aby uzyskać dostęp do Kreatora hosta z poziomu menu hosta. Pamiętaj, że podczas edycji istniejącego hosta niektóre kroki są pomijane, jeśli zawiera on już konfigurację wymaganą przez wybrany szablon.

Alternatywnie możesz skorzystać z klasycznego podejścia do tworzenia i konfigurowania hosta.

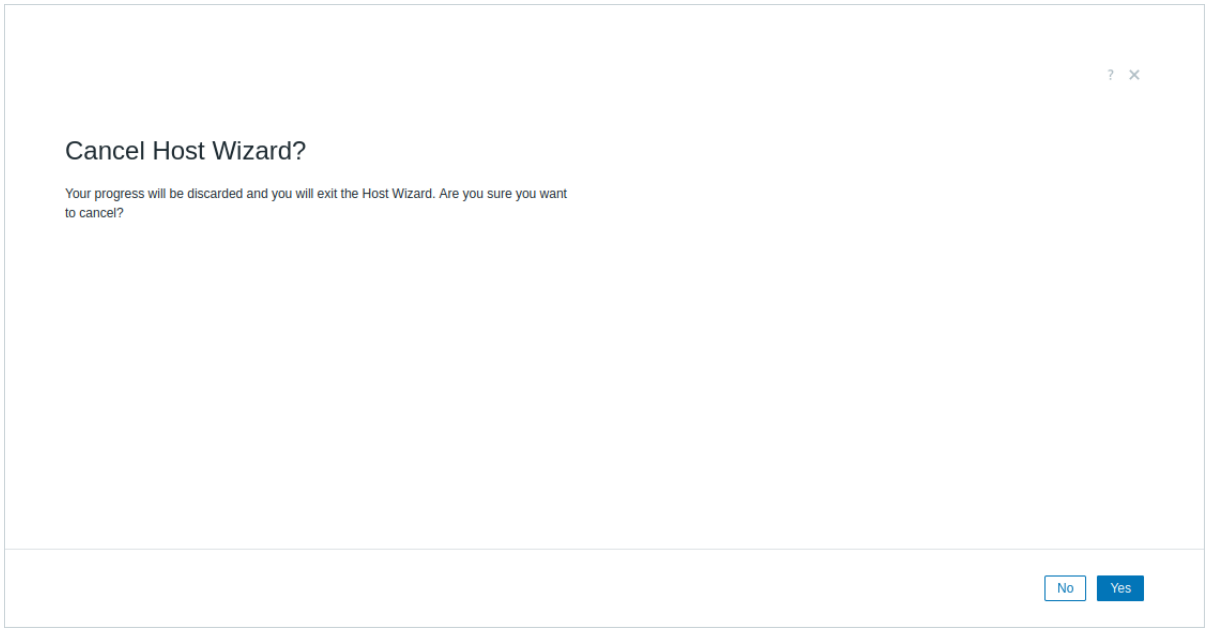
Witamy w kreatorze hosta

Po uruchomieniu kreatora hosta wyświetlany jest ekran powitalny:



Aby pominąć ten ekran w przyszłych sesjach, możesz zaznaczyć pole wyboru *Nie pokazuj ekranu powitalnego* i kliknąć *Dalej*.

Jeśli rozpoczniesz konfigurowanie ustawień w kolejnych krokach kreatora hosta i spróbujesz zamknąć kreator, klikając ikonę zamknięcia lub naciskając ESC, pojawi się ekran potwierdzenia:



Kliknięcie *Tak* spowoduje zamknięcie kreatora hosta bez zapisania postępu. Kliknięcie *Nie*, naciśnięcie ESC lub kliknięcie ikony zamknięcia spowoduje powrót do ostatniego kroku.

Wybierz szablon

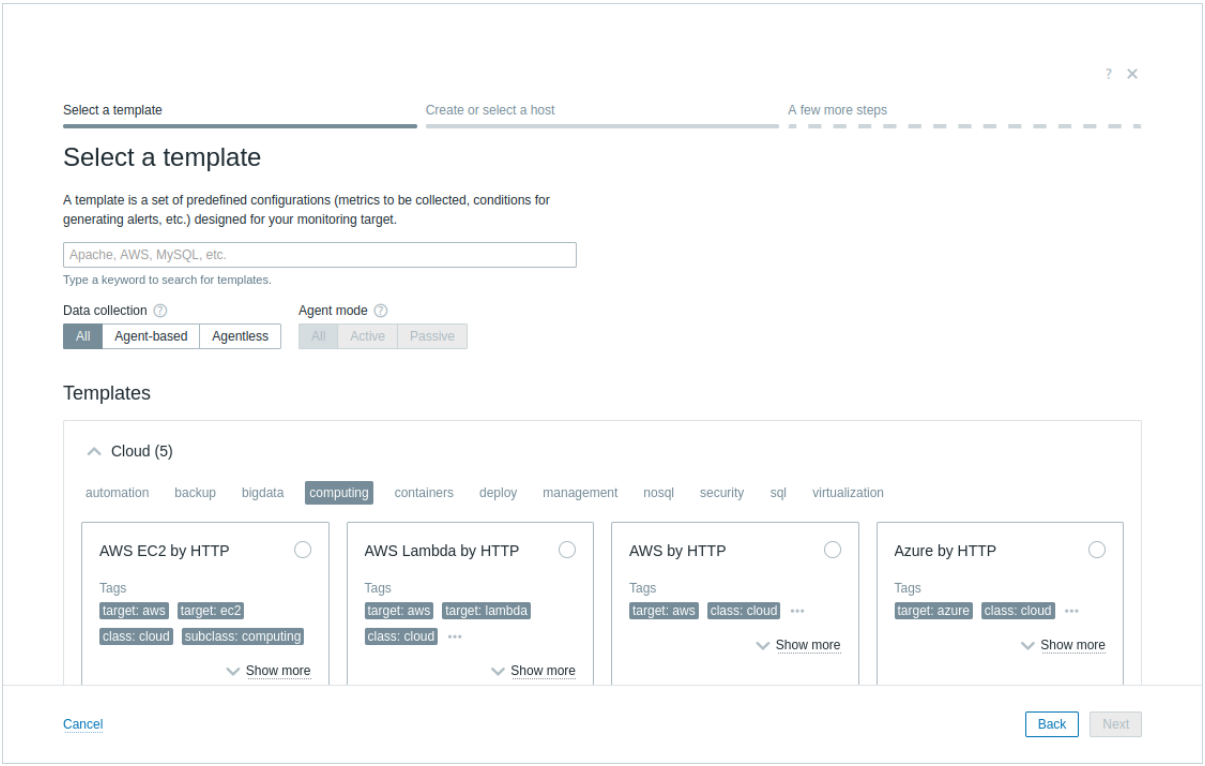
Pierwszym krokiem konfiguracji monitorowanego celu jest wybranie szablonu — zestawu predefiniowanych konfiguracji (metryk do zbierania, warunków generowania alertów itp.) przeznaczonych dla monitorowanego celu.

Gotowe szablony Zabbix oferują różnorodne predefiniowane konfiguracje monitorowania dla systemów operacyjnych, aplikacji, baz danych, urządzeń sieciowych, usług i nie tylko.

W tym kroku możesz:

- Przeglądać szablony według klasy (Cloud, Database, Network itp.) i podklasy (np. automation, backup), opartych na tagach szablonu.
- Wyszukiwać szablony według słów kluczowych w nazwie szablonu lub nazwach i wartościach tagów szablonu.
- Filtrować szablony według metody zbierania danych (z użyciem agenta lub bez agenta; szablony oparte na agencie zawierają co najmniej jedną pozycję Zabbix agent).
- Filtrować szablony według trybu agenta (aktywny lub pasywny; aktywne szablony zawierają co najmniej jedną pozycję Zabbix agent (active), natomiast pasywne szablony zawierają co najmniej jedną pozycję pasywną; szczegóły znajdują się w sekcji Passive and active agent checks).

- Podczas używania Kreatora hosta do konfiguracji istniejącego hosta możesz także filtrować szablony już podłączone do hosta.



Attention:
Podczas wybierania szablonu możesz zobaczyć następujący komunikat: *Niektóre szablony (n) są niezgodne z Kreatorem hosta. Zobacz, jak je zaktualizować. Szablony niestandardowe nie są obsługiwane.* Oznacza to, że niektóre gotowe szablony nie są jeszcze zgodne z Kreatorem hosta i wymagają aktualizacji; zobacz instrukcje aktualizacji szablonów.

Wybrany szablon określi kolejne kroki w Kreatorze hosta. Na przykład, jeśli wybierzesz szablon *MySQL by Zabbix agent*, który używa Zabbix agent do zbierania danych, kreator przeprowadzi Cię przez proces instalacji agenta.

Utwórz lub wybierz host

Szablon musi być podłączony do hosta — encji w Zabbix, która reprezentuje monitorowany obiekt. Po podłączeniu host otrzymuje wszystkie encje szablonu, takie jak pozycje (metryki do zbierania) i wyzwalacze (warunki generowania alertów), które są wymagane do monitorowania.

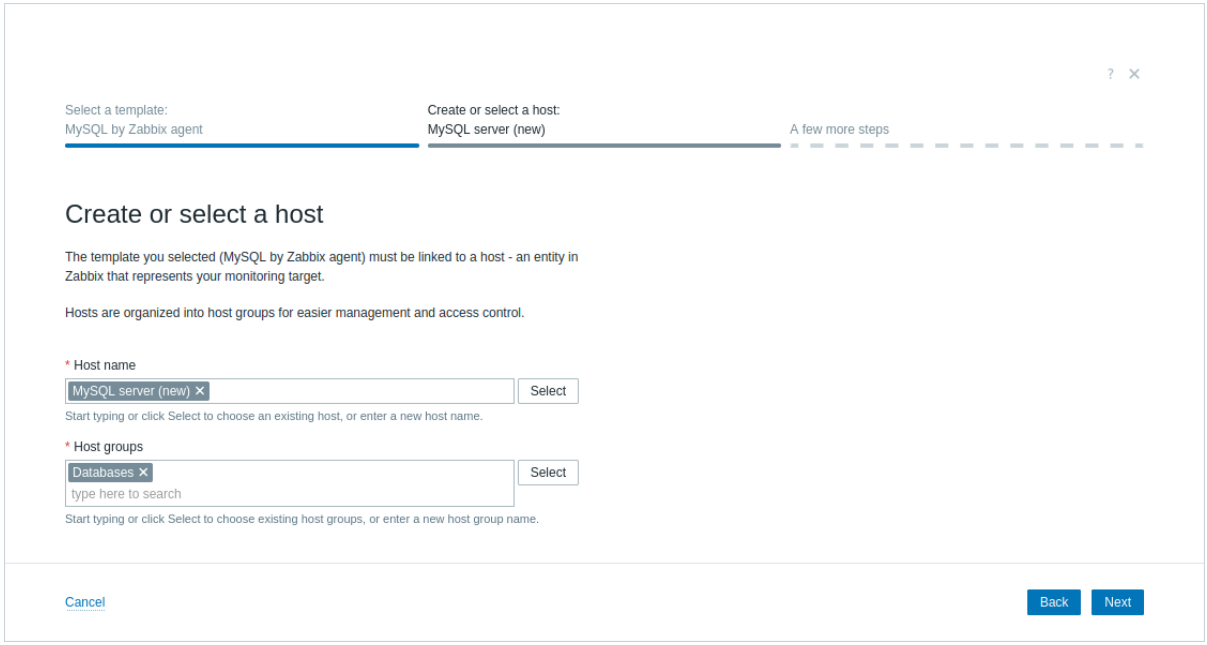
Każdy host musi również należeć do co najmniej jednej grupy hostów, która służy do organizowania hostów i przypisywania do nich uprawnień użytkowników.

W tym kroku możesz:

- Utworzyć nowy host i grupę hostów, wprowadzając nazwy dla obu.
- Utworzyć nowy host, wprowadzając jego nazwę, oraz wybrać istniejącą grupę hostów, do której nowy host zostanie przypisany.
- Wybrać istniejący host; możesz również przypisać tego hosta do dodatkowych grup hostów bez usuwania go z żadnych obecnych grup.

Note:
Wybranie istniejącego hosta nie usuwa ani nie nadpisuje jego bieżącej konfiguracji, chyba że zostanie to wyraźnie wskazane w kolejnych krokach. Wykryte hosty nie mogą zostać wybrane.

Na przykład możesz utworzyć nowy host o nazwie *MySQL server*, aby reprezentował lokalnie zainstalowany serwer MySQL, który chcesz monitorować. Wcześniej wybrany szablon *MySQL by Zabbix agent* zostanie podłączony do tego hosta. Dodatkowo możesz wybrać istniejącą grupę hostów *Databases* (lub utworzyć nową, np. *MySQL servers*), aby uporządkować hosta razem z innymi hostami związanymi z bazami danych i później uprościć zarządzanie uprawnieniami.



Instalacja Zabbix agent

Szablony oparte na agencie (takie jak *MySQL by Zabbix agent*) wymagają zainstalowania Zabbix agent lub agent 2 — procesu wdrożonego na monitorowanym obiekcie, aby aktywnie monitorować lokalne zasoby i aplikacje (jeśli jest to wymagane przez szablon).

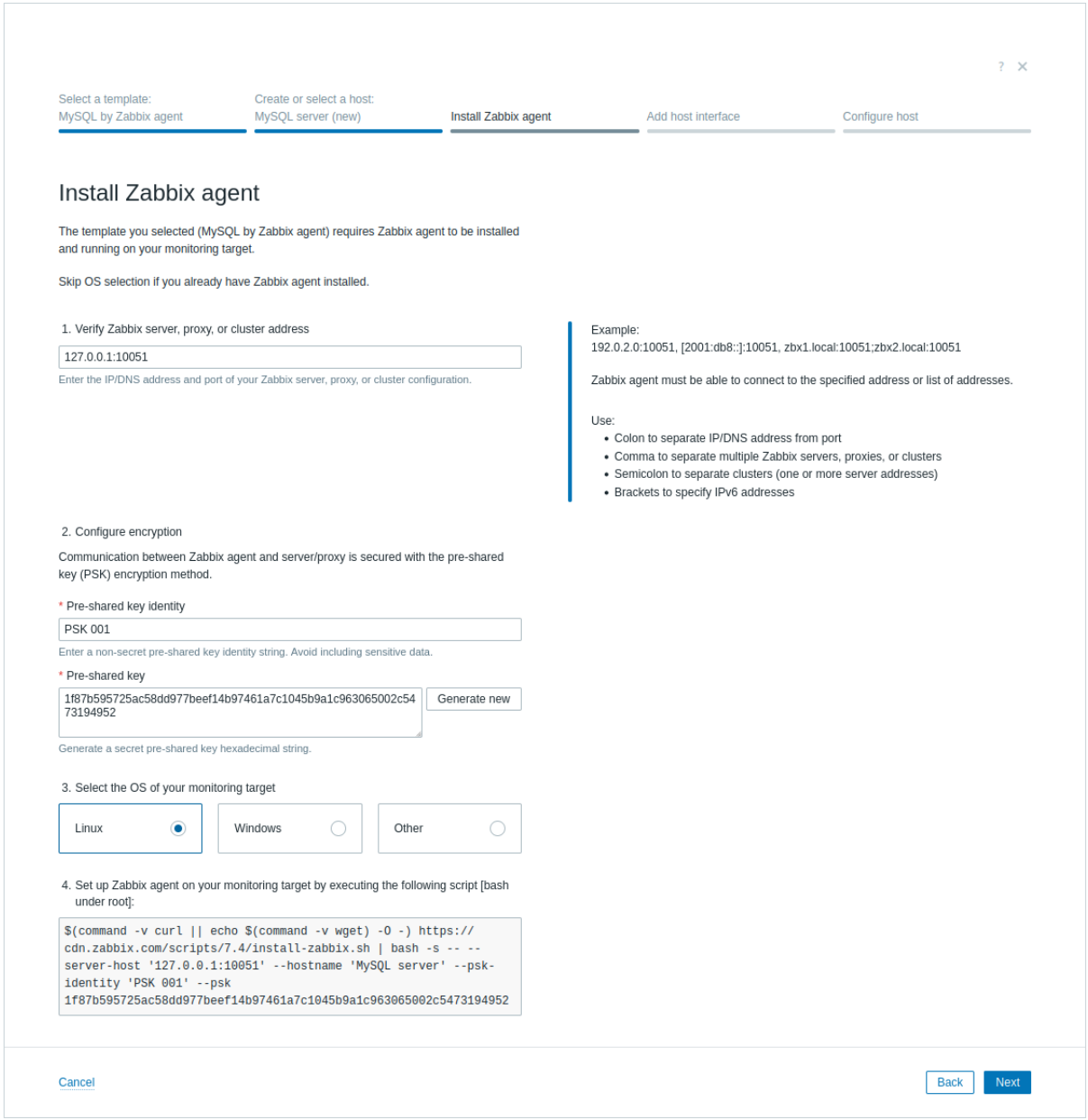
W tym kroku:

- Zweryfikuj adres serwera Zabbix, proxy lub klastra (np. 192.0.2.1:10051). Ta wartość zostanie użyta do skonfigurowania parametru Server i ServerActive dla Zabbix agent.
- Wprowadź nową, unikalną i niejawną tożsamość klucza wstępnie współdzielonego (np. *PSK 001* lub *mysql-agent-psk1*) i wygeneruj klucz wstępnie współdzielony. Te wartości zostaną użyte do skonfigurowania szyfrowania pre-shared key (PSK) dla hosta Zabbix (np. *MySQL server*) i Zabbix agent.

Attention:
Podczas edycji istniejącego hosta konfiguracja PSK zastąpi wszystkie istniejące ustawienia szyfrowania na hoście.

- Wybierz system operacyjny maszyny, na której znajduje się monitorowany obiekt.
- Zainstaluj Zabbix agent w tym systemie, uruchamiając dostarczony skrypt lub postępując zgodnie z instrukcjami instalacji.

Na przykład możesz ustawić *Pre-shared key identity* na *PSK 001* i wygenerować nowy *Pre-shared key*. Następnie możesz wybrać *Linux* jako system operacyjny i uruchomić dostarczony skrypt w tym systemie. Po zainstalowaniu Zabbix agent wróć do Host Wizard.

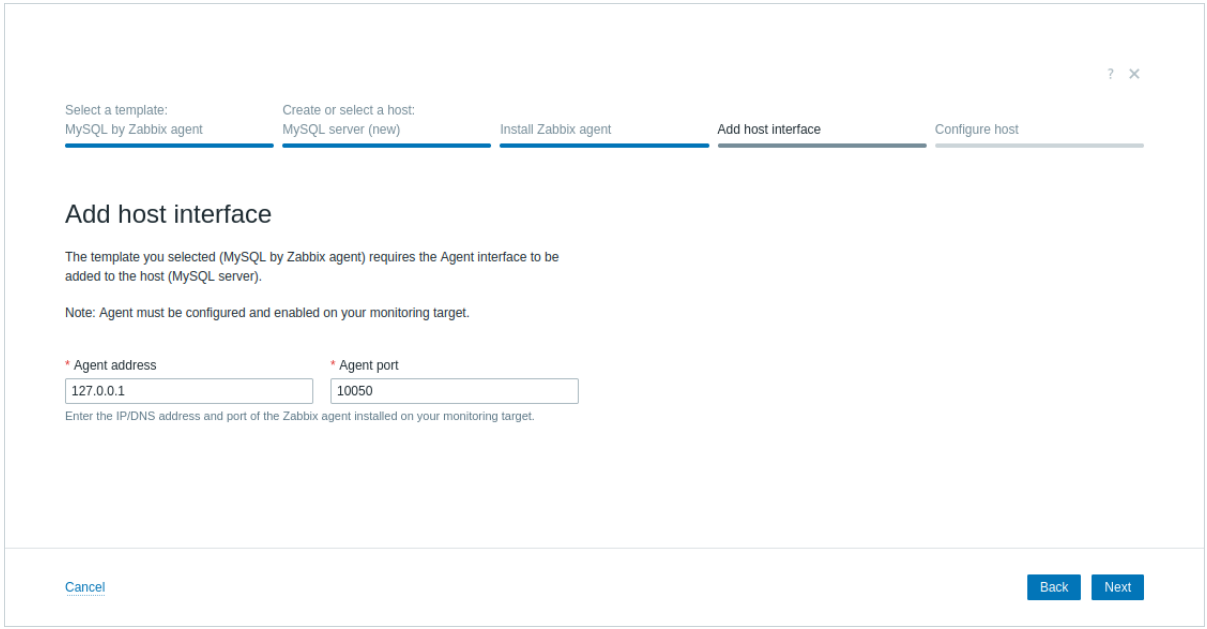


Dodaj interfejs hosta

Interfejs hosta określa, w jaki sposób serwer Zabbix łączy się przez sieć z monitorowanym celem. Definiuje parametry połączenia, takie jak adres IP, nazwa DNS, port oraz typ interfejsu (agent, SNMP, JMX lub IPMI), w zależności od metody zbierania danych wymaganej przez wybrany szablon.

W tym kroku wprowadź szczegóły interfejsu hosta wymagane przez wybrany szablon.

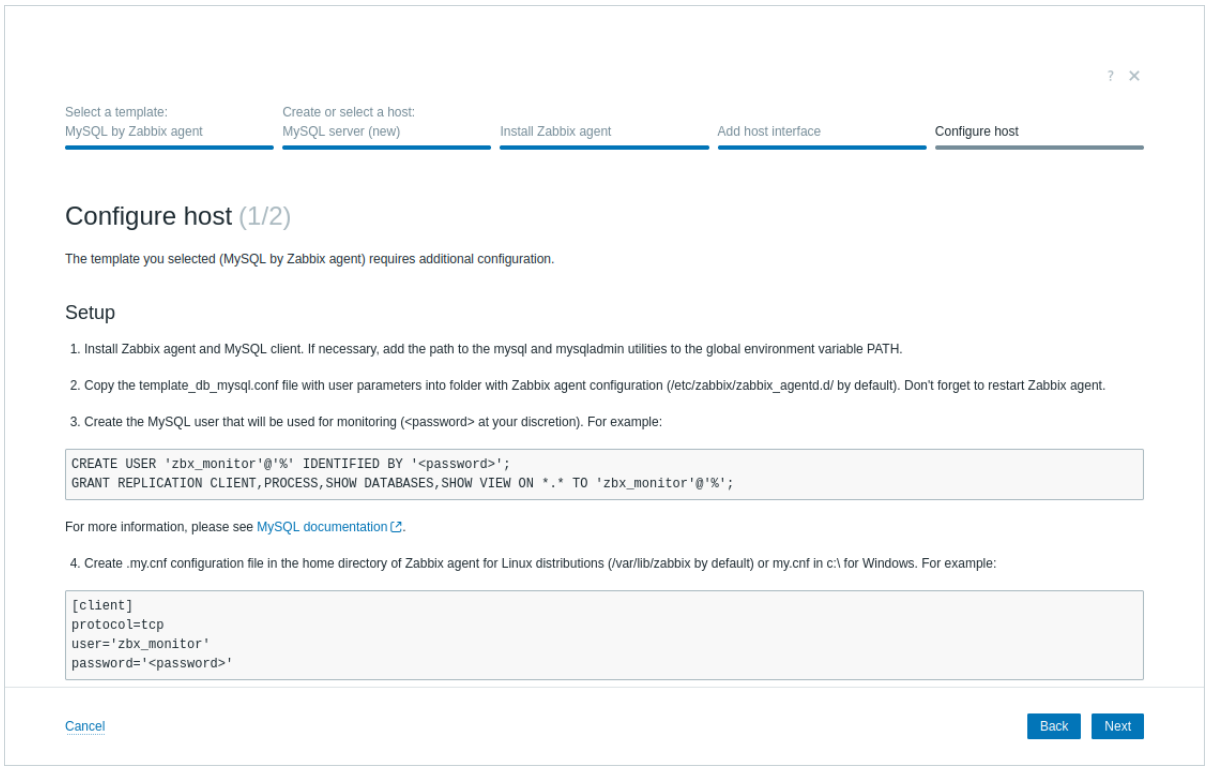
Na przykład możesz użyć domyślnego adresu agenta (127.0.0.1) i portu (10050), jeśli serwer Zabbix, agent Zabbix i serwer MySQL działają lokalnie na tej samej maszynie.



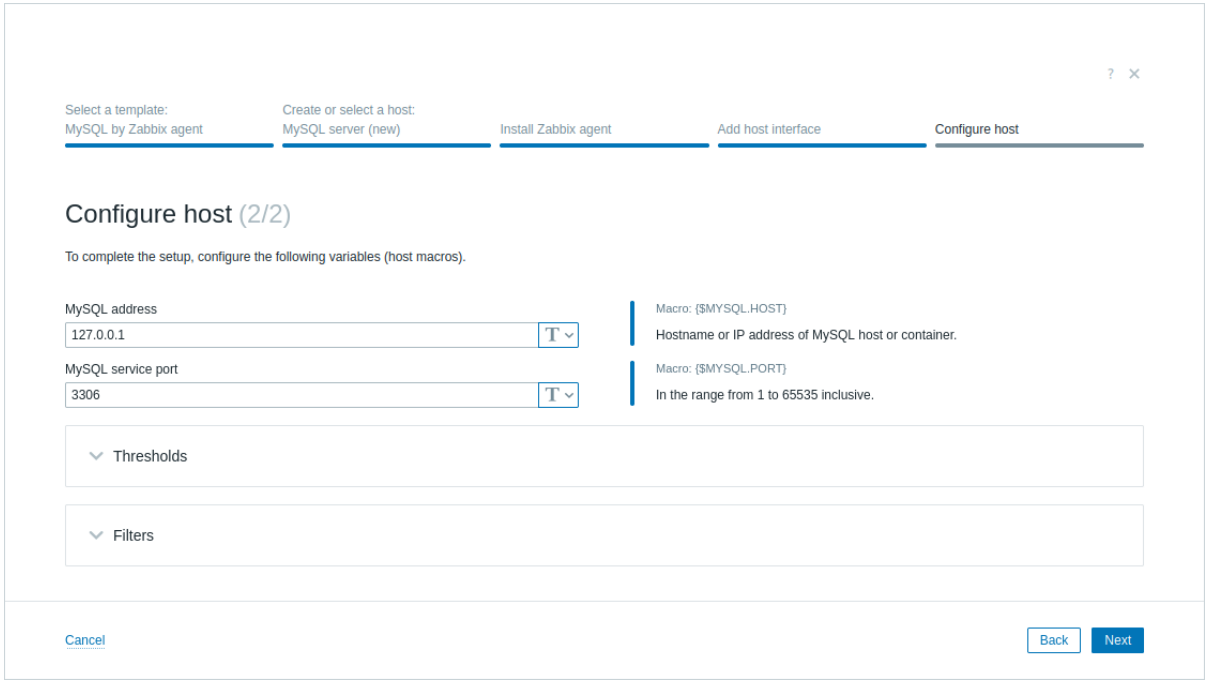
Konfiguracja hosta

Niektóre szablony wymagają dodatkowej konfiguracji, zanim host będzie mógł zostać utworzony. Może to obejmować:

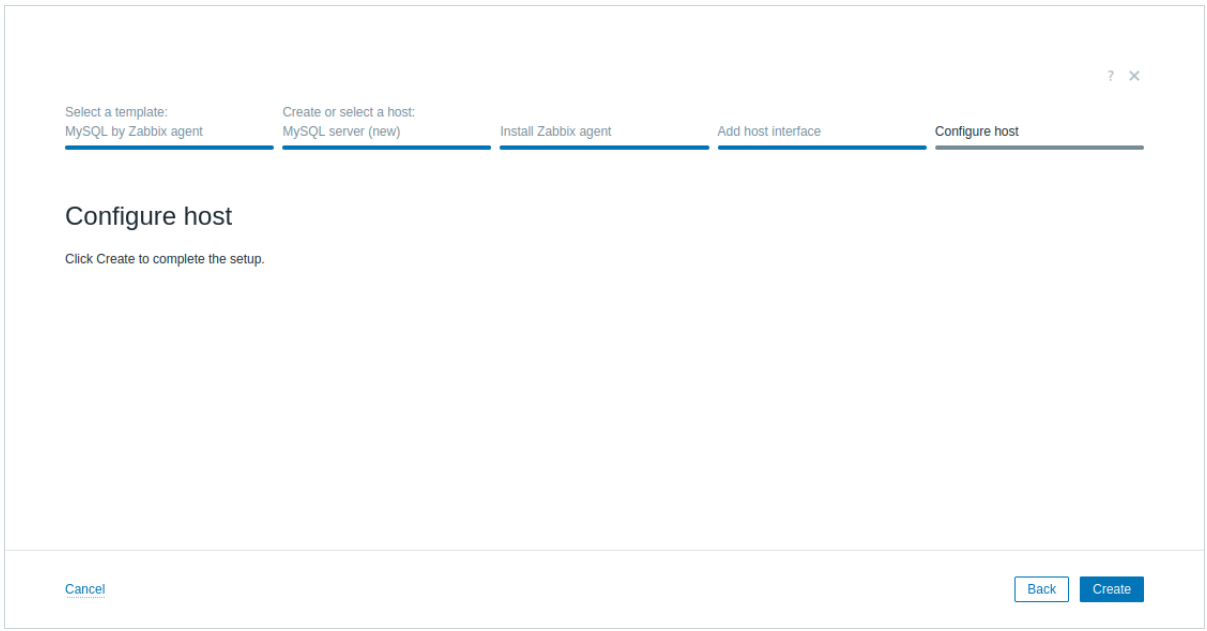
- Ręczną konfigurację monitorowanego celu (np. włączenie określonych usług, utworzenie użytkowników usług lub nadanie uprawnień):



- Definiowanie makr hosta — zmiennych kontrolujących działanie pozycji, ustawienia połączenia, dane uwierzytelniające i inne parametry:



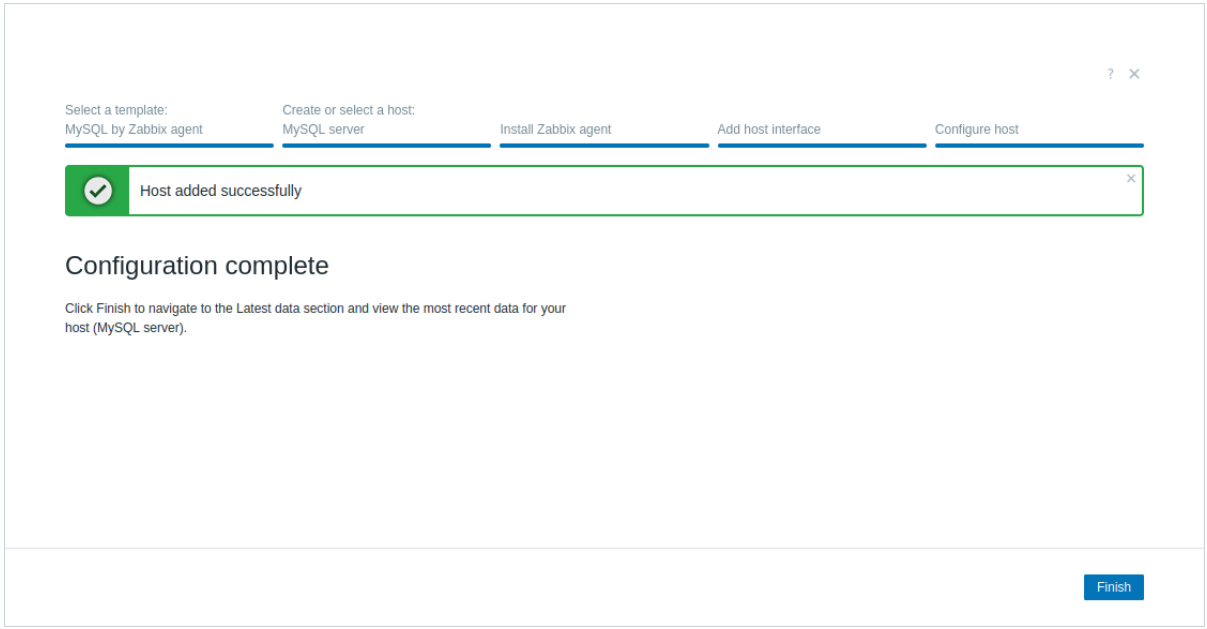
Po wykonaniu wszystkich wymaganych kroków konfiguracji kliknij *Create*, aby dodać host do Zabbix (lub *Update*, aby zaktualizować konfigurację istniejącego hosta):



Konfiguracja zakończona

W tym momencie Zabbix monitoruje już wskazany cel.

W przypadku nowych hostów kliknij *Zakończ*, aby przejść do sekcji Najnowsze dane i wyświetlić najnowsze dane dla swojego hosta. W przypadku istniejących hostów kliknij *Zakończ*, aby zamknąć Kreator hosta i wrócić do ekranu, z którego został otwarty.



2 Konfigurowanie hosta

Przegląd

Aby skonfigurować host w frontend Zabbix, wykonaj następujące czynności:

- Przejdź do: *Data collection* > *Hosts* lub *Monitoring* > *Hosts*
- Kliknij *Create host* w prawym górnym rogu ekranu (lub nazwę hosta, aby edytować istniejący host)
- Wprowadź parametry hosta w formularzu

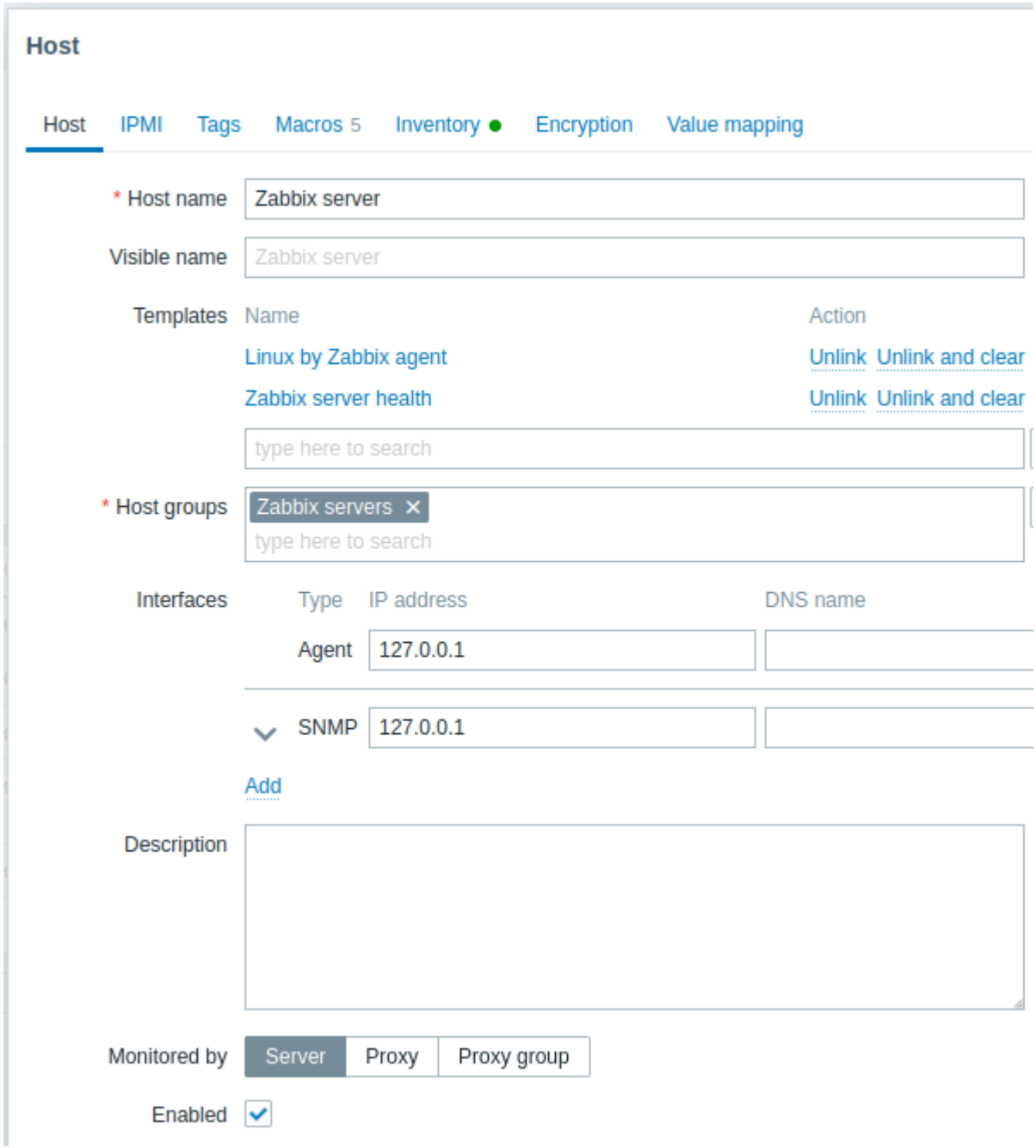
Możesz również użyć przycisku *Clone* w formularzu konfiguracji istniejącego hosta, aby utworzyć nowy host. Ten host będzie miał wszystkie właściwości istniejącego hosta, w tym powiązane szablony, elementy (pozycje, wyzwalacze itp.) z tych szablonów, a także elementy bezpośrednio przypisane do istniejącego hosta.

Należy pamiętać, że po sklonowaniu host zachowa wszystkie elementy szablonu w takiej postaci, w jakiej pierwotnie znajdują się w szablonie. Wszelkie zmiany tych elementów wprowadzone na poziomie istniejącego hosta (takie jak zmieniony interwał pozycji, zmodyfikowane wyrażenie regularne lub dodane prototypy do reguły niskopoziomowego wykrywania) nie zostaną sklonowane do nowego hosta; zamiast tego będą takie jak w szablonie.

Alternatywnie możesz użyć Host Wizard, aby skonfigurować host za pomocą prowadzonego interfejsu krok po kroku.

Konfiguracja

Karta **Host** zawiera ogólne atrybuty hosta:



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

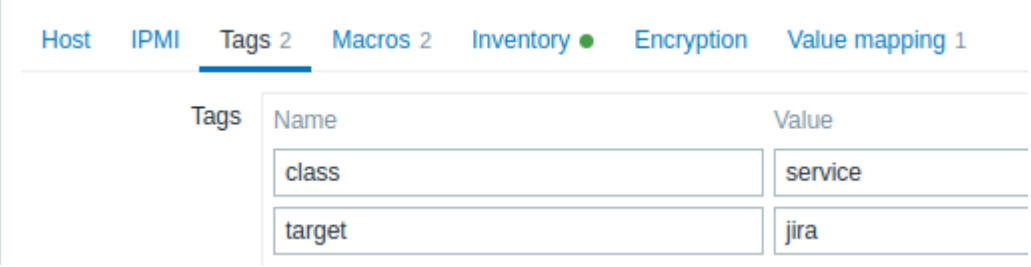
Parameter	Description
Host name	Wprowadź unikalną nazwę hosta. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia. Jednak spacje na początku i na końcu są niedozwolone. *Uwaga:* Jeśli na konfigurowanym hoście działa agent Zabbix, parametr *Hostname* w pliku konfiguracyjnym agenta musi mieć taką samą wartość jak nazwa hosta wpisana tutaj. Nazwa w tym parametrze jest wymagana podczas przetwarzania sprawdzania aktywnego.
Visible name	Wprowadź unikalną nazwę widoczną hosta. Jeśli ustawisz tę nazwę, będzie ona widoczna na listach, mapach itp. zamiast technicznej nazwy hosta. Ten atrybut obsługuje UTF-8.

Parameter	Description
Templates	Połącz szablony z hostem. Wszystkie encje (pozycje, wyzwalacze, itd.) zostaną odziedziczone z szablonu. Aby połączyć nowy szablon, zacznij wpisywać jego nazwę w polu tekstowym. Pojawi się lista pasujących szablonów; przewiń w dół, aby wybrać. Alternatywnie możesz kliknąć *Select* obok pola i wybrać szablony z listy w oknie podręcznym. Wszystkie wybrane szablony zostaną połączone z hostem po zapisaniu lub zaktualizowaniu formularza konfiguracji hosta. Aby odłączyć szablon, użyj jednej z dwóch opcji w bloku *Linked templates*: *Unlink* - odłącz szablon, ale zachowaj jego encje (pozycje, wyzwalacze, itd.); *Unlink and clear* - odłącz szablon i usuń wszystkie jego encje (pozycje, wyzwalacze, itd.). Wyświetlane nazwy szablonów są klikalnymi linkami prowadzącymi do formularza konfiguracji szablonu.
Host groups	Wybierz grupy hostów, do których należy host. Host musi należeć co najmniej do jednej grupy hostów. Nową grupę można utworzyć i połączyć z hostem, dodając nieistniejącą nazwę grupy.
Interfaces	Dla hosta obsługiwanych jest kilka typów interfejsów: *Agent*, *SNMP*, *JMX* i *IPMI*. Domyślnie nie są zdefiniowane żadne interfejsy. Aby dodać nowy interfejs, kliknij *Add* w bloku *Interfaces*, wybierz typ interfejsu i wprowadź informacje *IP/DNS*, *Connect to* oraz *Port*. *Uwaga:* Interfejsów używanych w jakichkolwiek pozycjach nie można usunąć, a link *Remove* jest dla nich wyszarzony. Wartość ”IP” lub ”DNS” z interfejsu SNMP jest również używana dla pułapek SNMP. Podczas dopasowywania używana jest tylko wybrana wartość ”IP” lub ”DNS” w interfejsie hosta. Więcej informacji o konfiguracji interfejsu SNMP (v1, v2 i v3) znajdziesz w sekcji Konfigurowanie monitorowania SNMP.
IP address	Adres IP hosta (opcjonalnie).
DNS name	Nazwa DNS hosta (opcjonalnie).
Connect to	Kliknięcie odpowiedniego przycisku poinformuje serwer Zabbix, czego użyć do pobierania danych od agentów: **IP** - łącz się z adresem IP hosta (zalecane) **DNS** - łącz się z nazwą DNS hosta
Port	Numer portu TCP/UDP. Wartości domyślne to: 10050 dla agenta Zabbix, 161 dla agenta SNMP, 12345 dla JMX i 623 dla IPMI.
Default	Zaznacz przycisk opcji, aby ustawić domyślny interfejs.
Description	Wprowadź opis hosta.
Monitored by	Wybierz, czy host jest monitorowany przez: **Server** - host jest monitorowany przez serwer Zabbix; **Proxy** - host jest monitorowany przez samodzielny proxy; **Proxy group** - host jest monitorowany przez grupę proxy.
Proxy	Wyświetlana jest przypisana nazwa proxy (tylko jeśli serwer Zabbix przypisał ją z wybranej grupy proxy). To pole jest wyświetlane tylko wtedy, gdy host jest monitorowany przez grupę proxy.
Enabled	Gdy pole wyboru jest zaznaczone, host jest włączony - gotowy do monitorowania. Gdy pole wyboru nie jest zaznaczone, host jest wyłączony - nie jest monitorowany: W przypadku pasywnych żądań danych inicjowanych przez serwer/proxy Zabbix (na przykład agent Zabbix, agent SNMP, simple checks), monitorowanie zostaje wyłączone po synchronizacji konfiguracji. Wyzwalacze i akcje powiązane z hostem są również wyłączane dopiero po ponownym wczytaniu konfiguracji. W przypadku sprawdzania aktywnego agenta Zabbix monitorowanie zatrzymuje się w czasie (około 5 sekund), w którym agent Zabbix otrzymuje informację o wyłączeniu hosta. W tym krótkim okresie host będzie nadal lokalnie zbierał dane dla sprawdzeń aktywnych i próbował wysyłać je do serwera/proxy; jednak ponieważ host jest oznaczony jako *Disabled*, serwer/proxy odrzuci dane. Po wyłączeniu hosta jego pozycje są natychmiast usuwane z pamięci podręcznej historii (z wyjątkiem ich ostatnich wartości, które są zachowywane dla logów).

Karta **IPMI** zawiera atrybuty zarządzania IPMI.

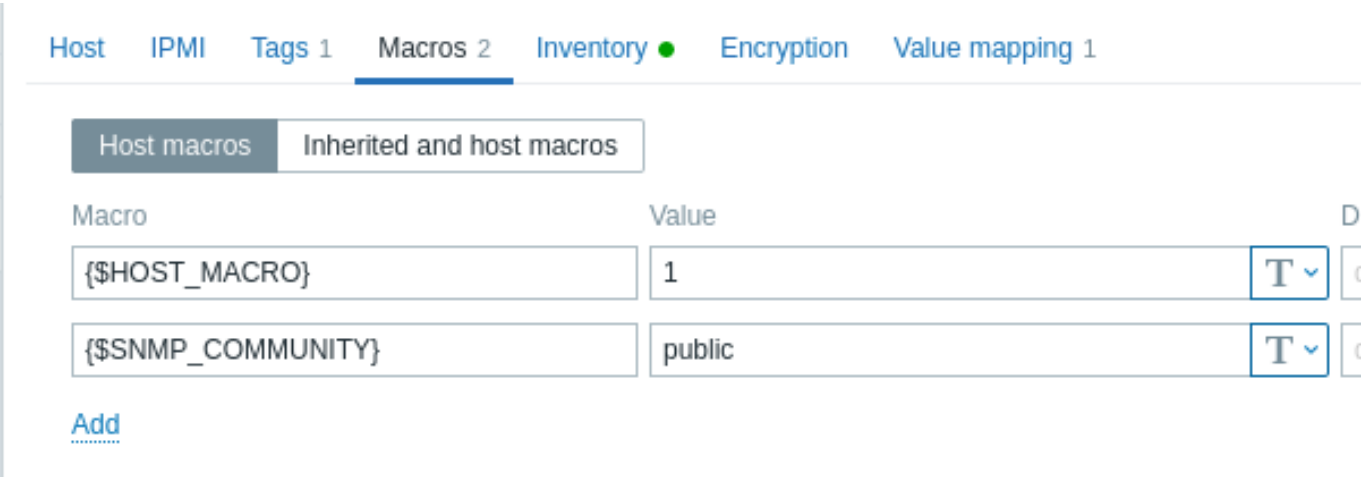
Parameter	Description
Authentication algorithm	Wybierz algorytm uwierzytelniania.
Privilege level	Wybierz poziom uprawnień.
Username	Nazwa użytkownika do uwierzytelniania. Można używać makr użytkownika.
Password	Hasło do uwierzytelniania. Można używać makr użytkownika.

Karta **Tags** umożliwia zdefiniowanie tagów na poziomie hosta tags. Wszystkie problemy tego hosta będą oznaczane wartościami wprowadzonymi tutaj.

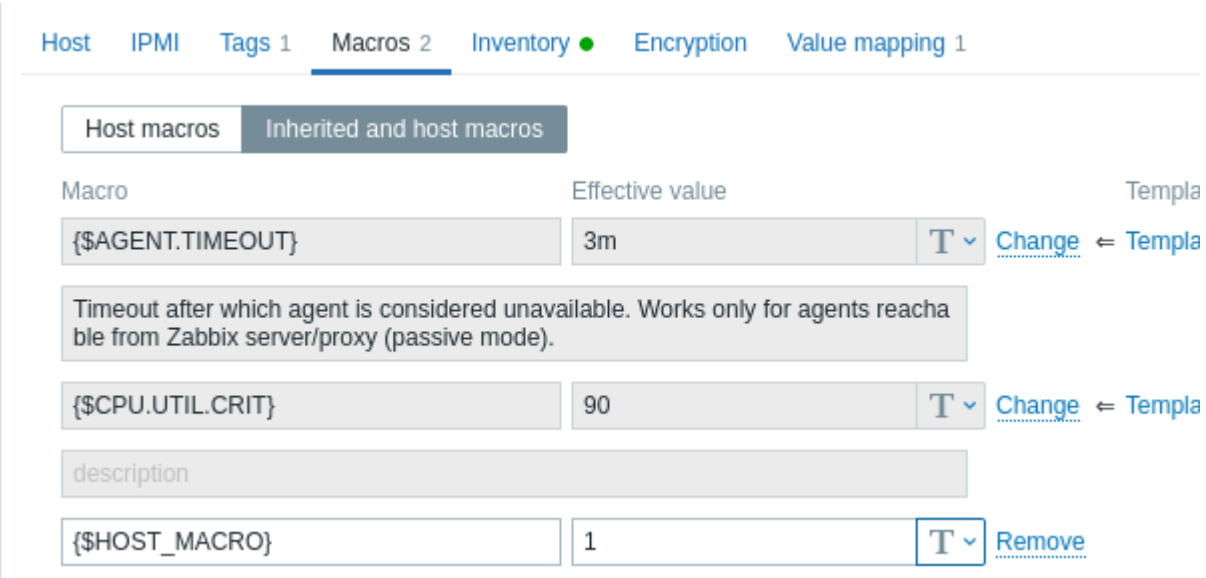


Obsługiwane są makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} i {HOST.ID}.

Karta **Macros** umożliwia zdefiniowanie makr użytkownika na poziomie hosta w postaci par nazwa-wartość. Zwróć uwagę, że wartości makr mogą być przechowywane jako zwykły tekst, tekst tajny lub sekret Vault. Obsługiwane jest również dodawanie opisu.

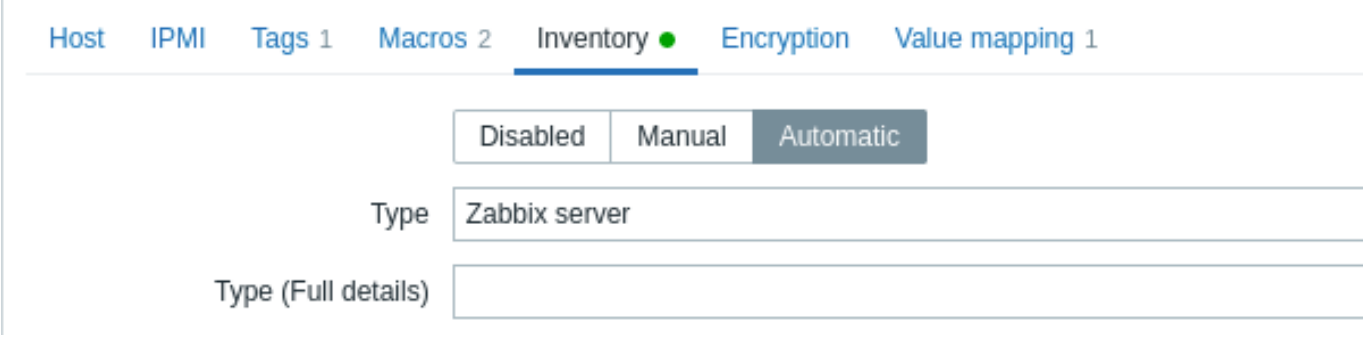


Możesz tu również wyświetlić makra użytkownika na poziomie szablonu i globalne, jeśli wybierzesz opcję *Inherited and host macros*. W tym miejscu wyświetlane są wszystkie zdefiniowane makra użytkownika dla hosta wraz z wartością, do której są rozwijane, oraz ich pochodzeniem.



Dla wygody dostępne są linki do odpowiednich szablonów i konfiguracji globalnych makr. Możliwe jest także edytowanie makra szablonu/globalnego na poziomie hosta, co skutecznie tworzy kopię makra na hoście.

Karta **Inventory** umożliwia ręczne wprowadzenie informacji o inwentarzu dla hosta. Możesz także wybrać włączenie automatycznego uzupełniania inwentarza (*Automatic*) albo wyłączyć uzupełnianie inwentarza dla tego hosta.



Jeśli inwentarz jest włączony (ręcznie lub automatycznie), przy nazwie karty wyświetlana jest zielona kropka.

Szyfrowanie

Karta **Szyfrowanie** umożliwia wymaganie zaszyfrowanych połączeń z hostem.

Parameter	Description
Connections to host	W jaki sposób serwer Zabbix lub proxy łączy się z agentem Zabbix na hoście: bez szyfrowania (domyślnie), z użyciem PSK (pre-shared key) lub certyfikatu.
Connections from host	Wybierz, jaki typ połączeń jest dozwolony z hosta (tj. od agenta Zabbix i Zabbix sendera). Można jednocześnie wybrać kilka typów połączeń (przydatne do testowania i przełączania na inny typ połączenia). Domyślnie: "No encryption".
Issuer	Dozwolony wystawca certyfikatu. Certyfikat jest najpierw weryfikowany za pomocą CA (certificate authority). Jeśli jest poprawny i podpisany przez CA, pole *Issuer* można wykorzystać do dalszego ograniczenia dozwolonego CA. To pole jest przeznaczone do użycia, jeśli instalacja Zabbix korzysta z certyfikatów od wielu CA. Jeśli to pole jest puste, akceptowana jest dowolna CA.
Subject	Dozwolony podmiot certyfikatu. Certyfikat jest najpierw weryfikowany za pomocą CA. Jeśli jest poprawny i podpisany przez CA, pole *Subject* można wykorzystać, aby zezwolić tylko na jedną wartość ciągu *Subject*. Jeśli to pole jest puste, akceptowany jest dowolny poprawny certyfikat podpisany przez skonfigurowaną CA.
PSK identity	Ciąg identyfikacyjny pre-shared key. Nie umieszczaj w identyfikatorze PSK informacji wrażliwych, ponieważ jest on przesyłany przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć.
PSK	Pre-shared key (ciąg szesnastkowy). Maksymalna długość: 512 cyfr szesnastkowych (PSK o rozmiarze 256 bajtów), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, 64 cyfry szesnastkowe (PSK o rozmiarze 32 bajtów), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Mapowanie wartości

Karta **Mapowanie wartości** umożliwia skonfigurowanie przyjaznej dla użytkownika reprezentacji danych pozycji w mapowaniach wartości.

3 Konfigurowanie grupy hostów

Przegląd

Grupy hostów służą do logicznego grupowania hostów oraz przypisywania do nich uprawnień użytkowników.

Każdy host musi mieć przypisaną co najmniej jedną grupę hostów. Host może należeć do wielu grup hostów, a każda grupa hostów może zawierać wiele hostów.

Należy pamiętać, że w Zabbix wszystkie uprawnienia są oparte na grupach: grupach użytkowników, grupach hostów oraz grupach szablonów. Dlatego nawet jeśli pojedynczy użytkownik potrzebuje dostępu do pojedynczego hosta, jest on przyznawany przez dodanie użytkownika do grupy użytkowników, która ma uprawnienia dostępu do grupy hostów zawierającej tego hosta.

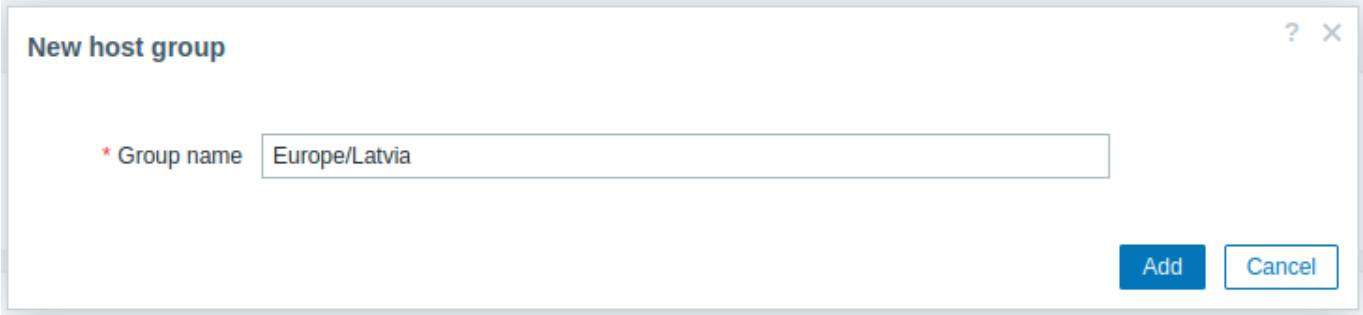
Konfiguracja

Attention:
Tylko użytkownicy typu Super admin mogą tworzyć grupy hostów.

Istnieją dwie opcje tworzenia grupy hostów w frontend Zabbix.

Opcja pierwsza:

- Przejdź do: *Data collection* → *Host groups*
- Kliknij *Create host group* w prawym górnym rogu ekranu
- Wprowadź nazwę grupy w formularzu



Opcja druga: podczas konfigurowania hosta wprowadź nieistniejącą nazwę grupy w polu *Host groups*.

Po utworzeniu grupy hostów możesz kliknąć nazwę grupy na liście w sekcji *Data collection* → *Host groups*, aby edytować nazwę grupy, sklonować grupę lub ją usunąć.

Usunięcie grupy hostów powoduje usunięcie tylko grupy logicznej, a nie hostów w tej grupie. Nie można usunąć grupy hostów, która jest jedyną grupą dla jakiegokolwiek istniejącego hosta.

Tworzenie podgrup hostów

Podgrupa hostów (lub zagnieżdżona grupa hostów) jest elementem podrzędnym nadrzędnej grupy hostów, która ją zawiera.

Podgrupę tworzy się przy użyciu ukośnika '/' w polu wprowadzania nazwy grupy, aby określić jej relację do grupy nadrzędnej lub grup nadrzędnych. Na przykład:

- wpisanie `Europe/Latvia` tworzy podgrupę `Europe/Latvia` w nadrzędnej grupie `Europe`.
- wpisanie `Europe/Latvia/Riga/Zabbix servers` tworzy odpowiednią podgrupę w zagnieżdżonych grupach nadrzędnych `Europe`, `Europe/Latvia`, `Europe/Latvia/Riga`.

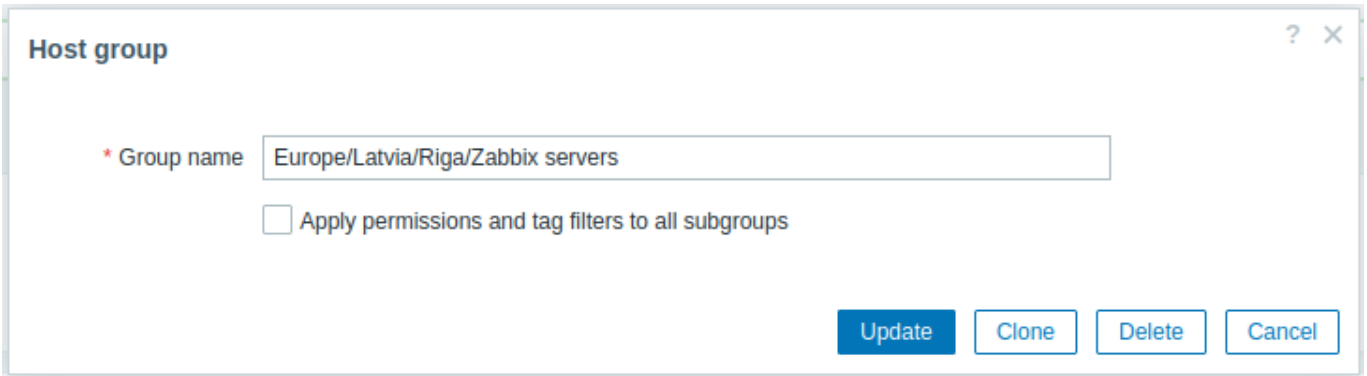
Podczas tworzenia podgrupy nie można używać ukośników na początku ani na końcu nazwy, ani kilku ukośników z rzędu. Escapowanie znaku '/' nie jest obsługiwane.

Nie jest wymagane tworzenie żadnych nadrzędnych grup hostów przed utworzeniem podgrupy. Możesz zdecydować, czy zacząć od utworzenia podgrupy (na przykład `Europe/Latvia`), czy dowolnej nadrzędnej grupy hostów (w naszym przykładzie `Europe`). Jeśli zaczniesz od utworzenia podgrupy, nadrzędne grupy hostów **nie** zostaną utworzone automatycznie.

Uprawnienia do grup hostów

- Podczas tworzenia podgrupy w istniejącej nadrzędnej grupie hostów (na przykład tworzenia `Europe/Latvia`, gdy `Europe` już istnieje), uprawnienia grupy użytkowników do podgrupy są dziedziczone z grupy nadrzędnej.
- Podczas tworzenia nadrzędnej grupy hostów dla istniejącej podgrupy (na przykład tworzenia `Europe`, gdy `Europe/Latvia` już istnieje), żadne uprawnienia do grupy nadrzędnej nie są ustawiane.

Podczas edycji dowolnej grupy hostów można również ustawić dodatkową opcję: *Zastosuj uprawnienia i filtry tagów do wszystkich podgrup*.



Zaznaczenie tego pola wyboru i kliknięcie *Aktualizuj* spowoduje zastosowanie tego samego poziomu uprawnień i filtrów tagów do wszystkich obecnych i przyszłych podgrup edytowanej grupy hostów.

Jeśli więc jakimkolwiek grupom użytkowników przyznano różne uprawnienia do podgrup edytowanej grupy hostów, zaznaczenie tego pola wyboru nada wszystkim obecnym i przyszłym podgrupom takie same uprawnienia użytkowników i uprawnienia oparte na tagach jak w edytowanej grupie.

Należy pamiętać, że ta opcja nie jest zapisywana w bazie danych i zastąpi istniejące uprawnienia. Wszelkie zmiany wprowadzone za pomocą tej opcji można cofnąć wyłącznie ręcznie.

4 Inwentaryzacja

Przegląd

Możesz przechowywać inwentarz urządzeń sieciowych w Zabbix.

W Zabbix frontend dostępne jest specjalne menu *Inwentarz*. Jednak początkowo nie zobaczysz tam żadnych danych i nie jest to miejsce do ich wprowadzania. Tworzenie danych inwentarza odbywa się ręcznie podczas konfigurowania hosta lub automatycznie przy użyciu niektórych opcji automatycznego uzupełniania.

Tworzenie inwentarza

Tryb ręczny

Podczas konfigurowania hosta, na karcie *Inventory* można wprowadzić takie szczegóły, jak typ urządzenia, numer seryjny, lokalizacja, osoba odpowiedzialna, adresy URL itp. — dane, które wypełnią informacje inwentaryzacyjne.

Jeśli adres URL jest zawarty w informacjach inwentaryzacyjnych hosta i zaczyna się od „http” lub „https”, w sekcji *Inventory* będzie wyświetlany jako klikalny link.

Tryb automatyczny

Inwentarz hosta może być również uzupełniany automatycznie. Aby to działało, podczas konfigurowania hosta tryb inwentarza na karcie *Inventory* musi być ustawiony na *Automatic*.

Następnie można skonfigurować pozycje hosta, aby uzupełniały dowolne pole inwentarza hosta swoją wartością, wskazując pole docelowe za pomocą odpowiedniego atrybutu (o nazwie *Item will populate host inventory field*) w konfiguracji pozycji.

Pozycje szczególnie przydatne do automatycznego zbierania danych inwentarzowych:

- system.hw.chassis[full|type|vendor|model|serial] — domyślnie [full], wymagane uprawnienia root
- system.hw.cpu[all|cpunum,full|maxfreq|vendor|model|curfreq] — domyślnie [all,full]
- system.hw.devices[pci|usb] — domyślnie [pci]
- system.hw.macaddr[interface,short|full] — domyślnie [all,full], interface jest wyrażeniem regularnym
- system.sw.arch
- system.sw.os[name|short|full] — domyślnie [name]
- system.sw.packages[regexp,manager,short|full] — domyślnie [all,all,full]

Wybór trybu inwentaryzacji

Tryb inwentaryzacji można wybrać w formularzu konfiguracji hosta.

Domyślny tryb inwentaryzacji dla nowych hostów jest wybierany na podstawie ustawienia *Default host inventory mode* w *Administration → General →* *Other*.

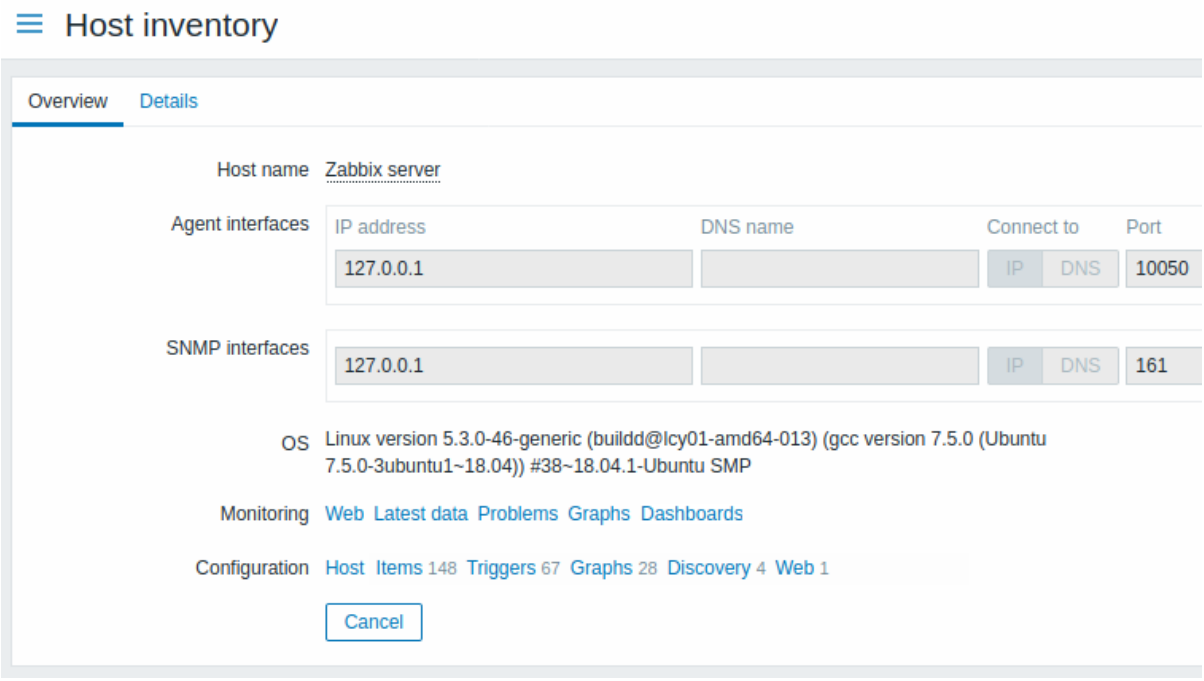
Dla hostów dodanych przez wykrywanie sieci lub akcje autorejestracji można zdefiniować operację *Set host inventory mode*, wybierając tryb ręczny lub automatyczny. Ta operacja zastępuje ustawienie *Default host inventory mode*.

Przegląd inwentaryzacji

Szczegóły wszystkich istniejących danych inwentaryzacyjnych są dostępne w menu *Inventory*.

W *Inventory → Overview* można uzyskać liczbę hostów według różnych pól inwentaryzacji.

W *Inventory → Hosts* można zobaczyć wszystkie hosty, które mają informacje inwentaryzacyjne. Kliknięcie nazwy hosta spowoduje wyświetlenie szczegółów inwentaryzacji w formularzu.



Karta **Overview** pokazuje:

Parameter	Description
Host name	Nazwa hosta. Kliknięcie nazwy otwiera menu ze skryptami zdefiniowanymi dla hosta. Nazwa hosta jest wyświetlana z pomarańczową ikoną, jeśli host jest w trybie konserwacji.
Visible name	Widoczna nazwa hosta (jeśli została zdefiniowana).
Host (Agent, SNMP, JMX, IPMI) interfaces	Ten blok zawiera szczegóły interfejsów skonfigurowanych dla hosta.
OS	Pole inwentaryzacji systemu operacyjnego hosta (jeśli zostało zdefiniowane).
Hardware	Pole inwentaryzacji sprzętu hosta (jeśli zostało zdefiniowane).
Software	Pole inwentaryzacji oprogramowania hosta (jeśli zostało zdefiniowane).
Description	Opis hosta.
Monitoring	Łącza do sekcji monitorowania z danymi dla tego hosta: *Web*, *Latest data*, *Problems*, *Graphs*, *Dashboards*.
Configuration	Łącza do sekcji konfiguracji dla tego hosta: *Host*, *Items*, *Triggers*, *Graphs*, *Discovery*, *Web*. Liczba skonfigurowanych encji jest podana po każdym łączu.

Karta **Details** pokazuje wszystkie wypełnione pola inwentaryzacji (nie są puste).

Makra inwentarza

Dostępne są makra inwentarza hosta {INVENTORY.*}, których można używać w powiadomieniach, na przykład:

„Serwer w {INVENTORY.LOCATION1} ma problem, osobą odpowiedzialną jest {INVENTORY.CONTACT1}, numer telefonu {INVENTORY.POC.PRIMARY.PHONE.A1}.”

Aby uzyskać więcej szczegółów, zobacz stronę Obsługiwane makra.

5 Aktualizacja masowa

Przegląd

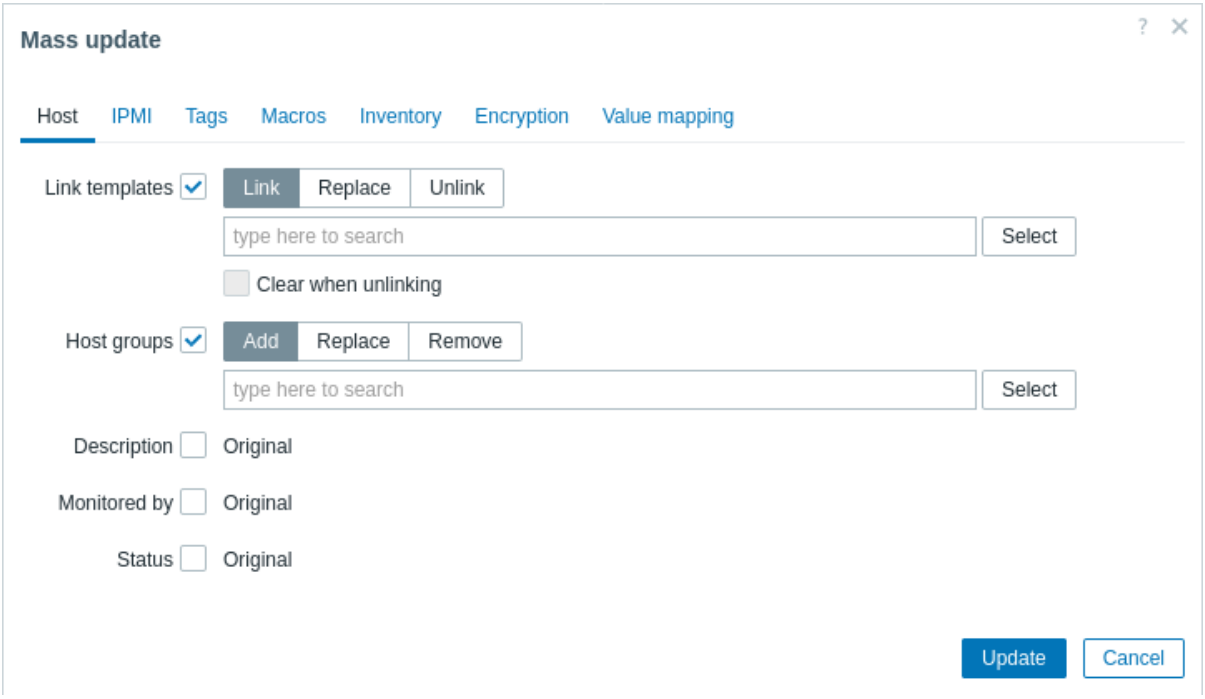
Czasami możesz chcieć naraz zmienić pewien atrybut dla wielu hostów. Zamiast otwierać każdy host do edycji z osobna, możesz skorzystać z funkcji aktualizacji masowej.

Korzystanie z masowej aktualizacji

Aby masowo zaktualizować wybrane hosty, wykonaj następujące czynności:

- Zaznacz pola wyboru przy hostach, które chcesz zaktualizować, na liście hostów
- Kliknij *Mass update* pod listą
- Przejdź do karty z wymaganymi atrybutami (*Host*, *IPMI*, *Tags*, *Macros*, *Inventory*, *Encryption* lub *Value mapping*)

- Zaznacz pola wyboru przy atrybutach, które chcesz zaktualizować, i wprowadź dla nich nowe wartości



Po wybraniu odpowiedniego przycisku aktualizacji powiązań **szablonów** dostępne są następujące opcje:

- *Link* - określa, które dodatkowe szablony mają zostać podłączone
- *Replace* - określa, które szablony mają zostać podłączone, jednocześnie odłączając wszystkie szablony, które były wcześniej podłączone do hostów
- *Unlink* - określa, które szablony mają zostać odłączone

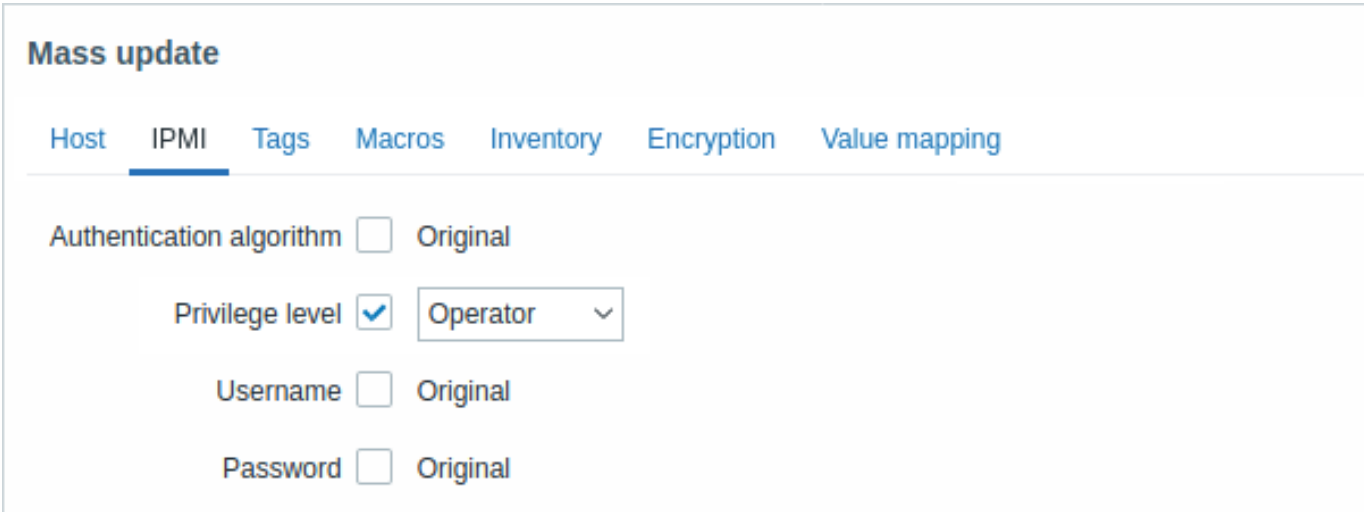
Aby określić szablony do podłączenia/odłączenia, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się rozwijana lista z pasującymi szablonami. Następnie przewiń listę i wybierz wymagany szablon.

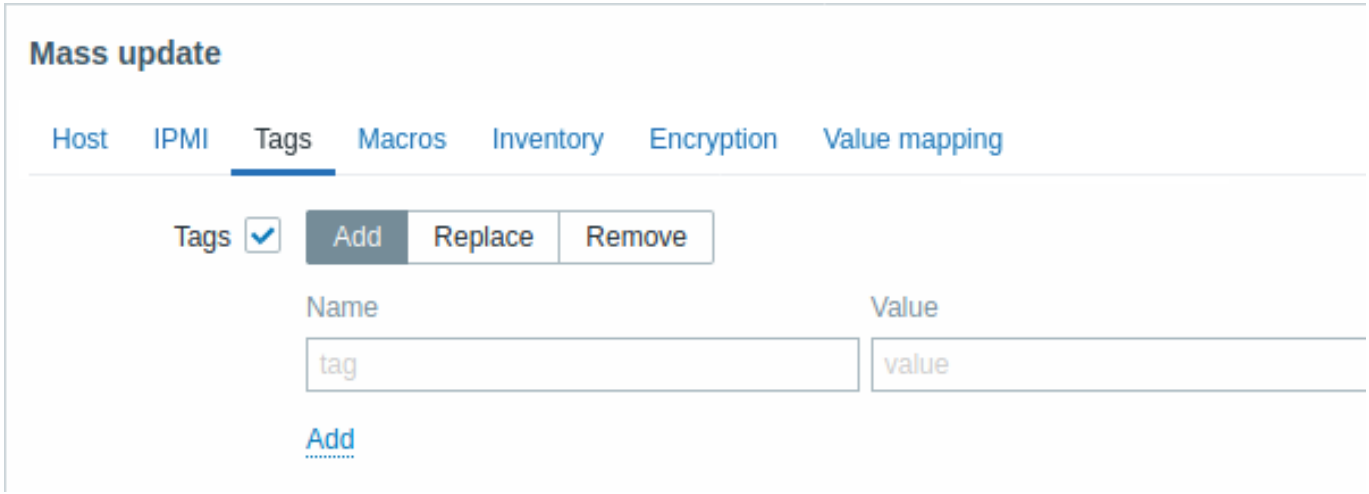
Opcja *Clear when unlinking* pozwala nie tylko odłączyć wcześniej podłączone szablony, ale także usunąć wszystkie elementy odziedziczone po nich (pozycje, wyzwalacze itd.).

Po wybraniu odpowiedniego przycisku aktualizacji **grup hostów** dostępne są następujące opcje:

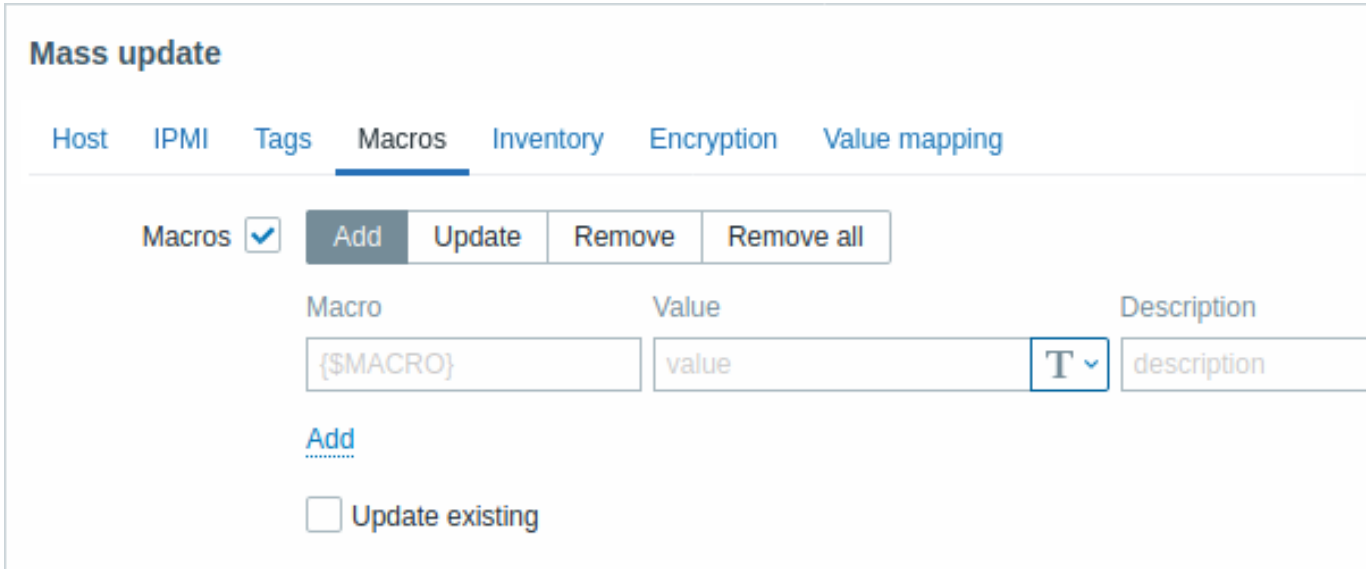
- *Add* - umożliwia określenie dodatkowych grup hostów spośród już istniejących lub wprowadzenie całkowicie nowych grup hostów dla hostów
- *Replace* - usunie host z wszystkich istniejących grup hostów i zastąpi je grupą/grupami określonymi w tym polu (istniejącymi lub nowymi grupami hostów)
- *Remove* - usunie określone grupy hostów z hostów

Pola te obsługują autouzupełnianie — rozpoczęcie wpisywania powoduje wyświetlenie rozwijanej listy pasujących grup hostów. Jeśli grupa hostów jest nowa, również pojawi się na liście i będzie oznaczona dopiskiem *(new)* po ciągu znaków. Wystarczy przewinąć listę i ją wybrać.



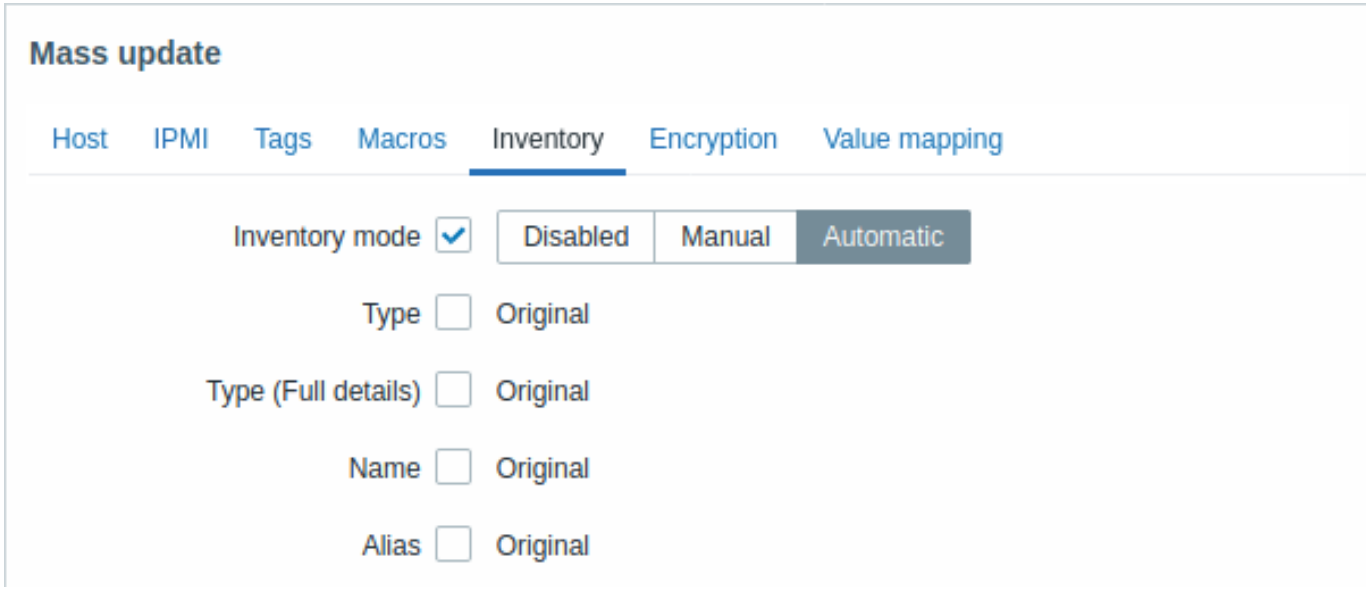


Makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID} są obsługiwane w tagach. Zwróć uwagę, że tagi o tej samej nazwie, ale różnych wartościach, nie są uznawane za „duplikaty” i mogą zostać dodane do tego samego hosta.

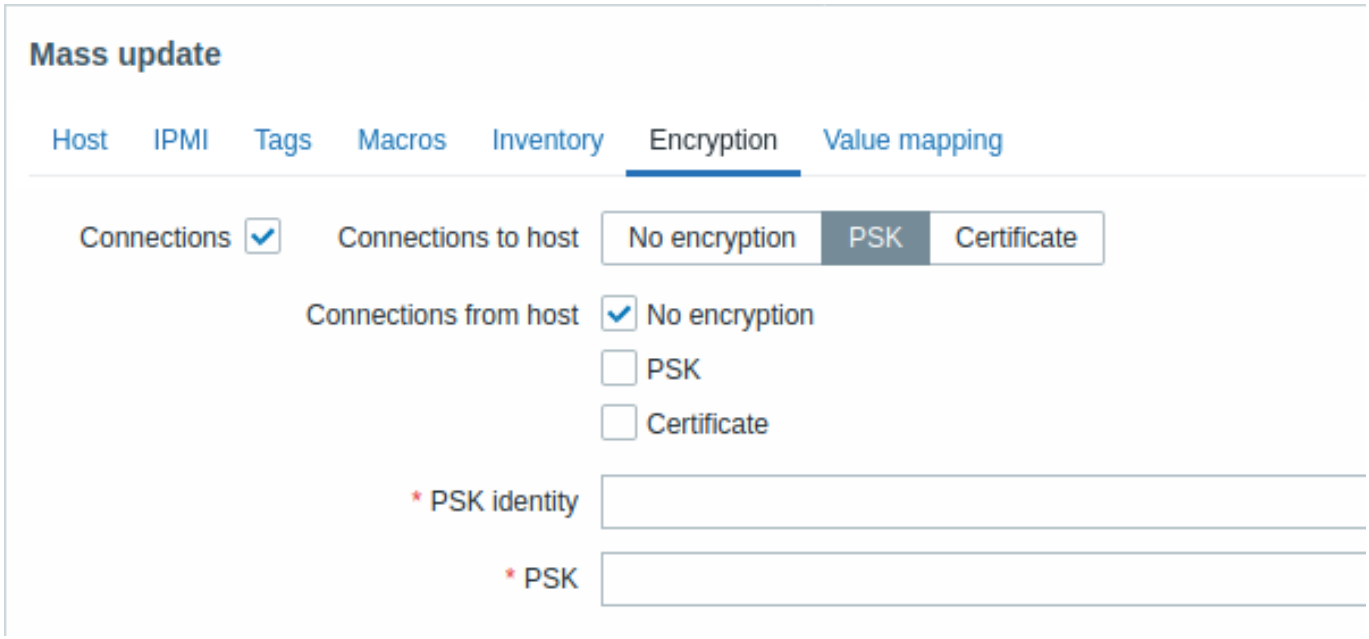


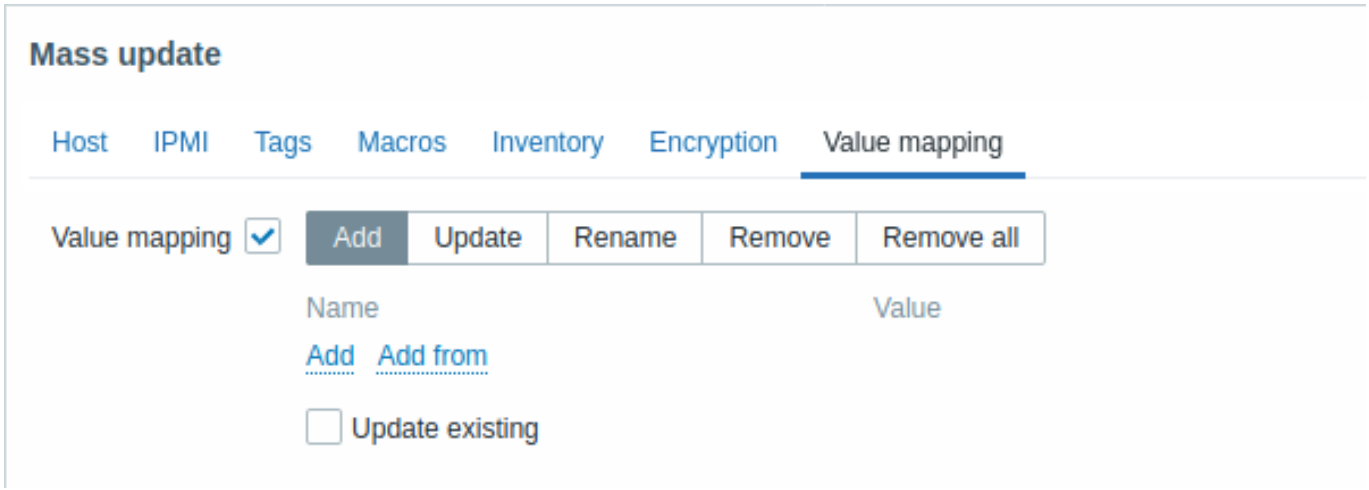
Po wybraniu odpowiedniego przycisku aktualizacji makr dostępne są następujące opcje:

- *Add* - umożliwia określenie dodatkowych makr użytkownika dla hostów. Jeśli zaznaczone jest pole wyboru *Update existing*, wartość, typ i opis dla wskazanej nazwy makra zostaną zaktualizowane. Jeśli pole nie jest zaznaczone i makro o tej nazwie już istnieje na hoście/hostach, nie zostanie zaktualizowane.
- *Update* - zastąpi wartości, typy i opisy makr określonych na tej liście. Jeśli zaznaczone jest pole wyboru *Add missing*, makro, które wcześniej nie istniało na hoście, zostanie dodane jako nowe. Jeśli pole nie jest zaznaczone, zaktualizowane zostaną tylko makra, które już istnieją na hoście.
- *Remove* - usunie określone makra z hostów. Jeśli zaznaczone jest pole *Except selected*, wszystkie makra poza wskazanymi na liście zostaną usunięte. Jeśli pole nie jest zaznaczone, usunięte zostaną tylko makra wskazane na liście.
- *Remove all* - usunie wszystkie makra użytkownika z hostów. Jeśli pole wyboru *I confirm to remove all macros* nie jest zaznaczone, otworzy się nowe wyskakujące okno z prośbą o potwierdzenie usunięcia wszystkich makr.



Aby możliwa była masowa aktualizacja pól inwentarza, *Inventory mode* powinien być ustawiony na „Manual” lub „Automatic”.





W przypadku aktualizacji map wartości dostępne są przyciski z następującymi opcjami:

- *Add* - dodaje mapy wartości do hostów. Jeśli zaznaczysz *Update existing*, wszystkie właściwości mapy wartości o tej nazwie zostaną zaktualizowane. W przeciwnym razie, jeśli mapa wartości o tej nazwie już istnieje, nie zostanie zaktualizowana.
- *Update* - aktualizuje istniejące mapy wartości. Jeśli zaznaczysz *Add missing*, mapa wartości, która wcześniej nie istniała na hoście, zostanie dodana jako nowa mapa wartości. W przeciwnym razie zaktualizowane zostaną tylko mapy wartości, które już istnieją na hoście.

- *Rename* - nadaje nową nazwę istniejącej mapie wartości
- *Remove* - usuwa określone mapy wartości z hostów. Jeśli zaznaczysz *Except selected*, wszystkie mapy wartości zostaną usunięte **z wyjątkiem** tych, które zostały wskazane.
- *Remove all* - usuwa wszystkie mapy wartości z hostów. Jeśli pole wyboru *I confirm to remove all value maps* nie jest zaznaczone, otworzy się nowe wyskakujące okno z prośbą o potwierdzenie usunięcia.

Po wprowadzeniu wszystkich wymaganych zmian kliknij *Update*. Atrybuty zostaną odpowiednio zaktualizowane dla wszystkich wybranych hostów.

2 Pozycje

#####Przegląd

Pozycja to indywidualna metryka.

Pozycje są używane do zbierania danych. Po skonfigurowaniu hosta, musisz dodać pozycje, aby uzyskać rzeczywiste dane. Jednym ze sposobów szybkiego dodawania wielu pozycji jest dołączenie jednego z predefiniowanych szablonów do hosta. Jednakże w celu zoptymalizowania wydajności systemu może zaistnieć potrzeba dostosowania szablonów tak, aby zawierały liczbę pozycji i częstotliwość monitorowania, o ile to konieczne.

Aby określić, jakie dane zbierać z hosta, użyj klucza pozycji. Na przykład, pozycja o kluczu **system.cpu.load** zbierze dane o obciążeniu procesora, podczas gdy pozycja o kluczu **net.if.in** zbierze informacje o przychodzącym ruchu sieciowym.

Dodatkowe parametry można określić w nawiasach kwadratowych po nazwie klucza. Na przykład, system.cpu.load**[avg5]** zwróci średnie obciążenie procesora (load) z ostatnich 5 minut, podczas gdy net.if.in**[eth0]** pokaże ilość przychodzącego ruchu na interfejsie ”eth0”.

> **Note:**
> Zobacz poszczególne sekcje typów pozycji dla wszystkich obsługiwanych typów pozycji i kluczy pozycji.

Przejdź do tworzenia i konfigurowania pozycji.

1 Tworzenie pozycji

Przegląd

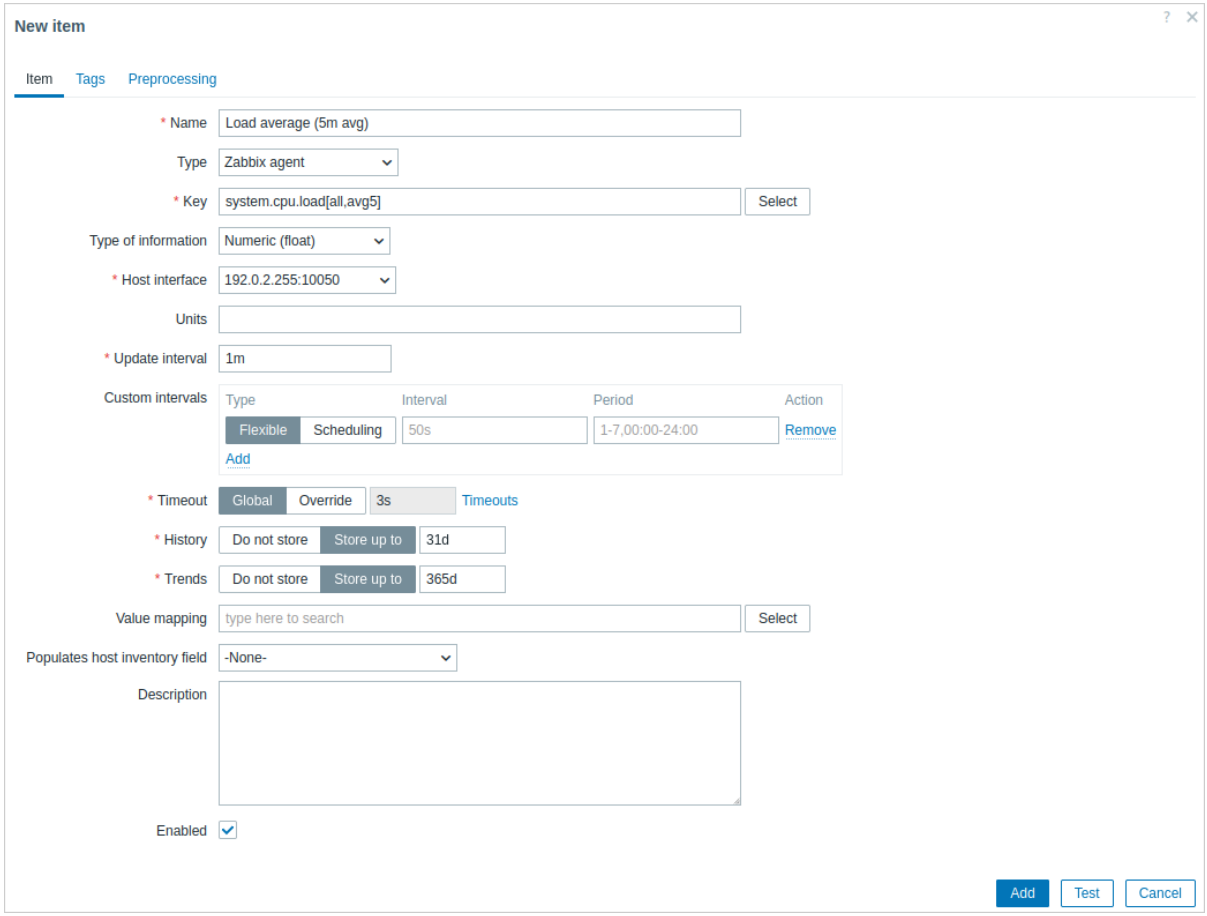
Aby utworzyć pozycję w frontendzie Zabbix, wykonaj następujące czynności:

- Przejdź do: *Data collection* > *Hosts*
- Kliknij *Items* w wierszu hosta
- Kliknij *Create item* w prawym górnym rogu ekranu
- Wprowadź parametry pozycji w formularzu

Możesz również utworzyć pozycję, otwierając istniejącą, naciskając przycisk *Clone*, a następnie zapisując ją pod inną nazwą.

Konfiguracja

Karta **Item** zawiera ogólne atrybuty pozycji.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa pozycji. Obsługiwane są makra użytkownika.
Type	Typ pozycji. Zobacz odpowiednie sekcje dotyczące poszczególnych typów pozycji.
Key	Klucz pozycji (do 2048 znaków). Obsługiwane klucze pozycji można znaleźć w odpowiednich sekcjach dotyczących typów pozycji. Klucz musi być unikalny w obrębie jednego hosta. Jeśli typ klucza to 'Zabbix agent', 'Zabbix agent (active)' lub 'Simple check', wartość klucza musi być obsługiwana przez Zabbix agent lub serwer Zabbix. Zobacz także: poprawny format klucza.
Type of information	Typ danych przechowywanych w bazie danych po wykonaniu ewentualnych konwersji. **Numeric (unsigned)** - 64-bitowa liczba całkowita bez znaku. Należy pamiętać, że wartość zmiennoprzecinkowa, jeśli zostanie odebrana dla pozycji całkowitej, zostanie obcięta do części dziesiętnej. **Numeric (float)** - 64-bitowa liczba zmiennoprzecinkowa. Ten typ zapewnia dokładność około 15 cyfr i zakres od około -1.79E+308 do 1.79E+308 (z wyjątkiem PostgreSQL 11 i wcześniejszych wersji). Obsługiwane jest również przyjmowanie wartości w notacji naukowej, np. 1.23E+7, 1e308, 1.1E-4. **Character** - krótkie dane tekstowe. **Log** - długie dane tekstowe z opcjonalnymi właściwościami związanymi z logami (timestamp, source, severity, logeventid). **Text** - długie dane tekstowe. Zobacz także limity danych tekstowych. **Binary** - liczba binarna (obsługiwana tylko dla pozycji zależnych). Pozycje zwracające wartości w binarnym typie danych nie są obsługiwane w formułach pozycji obliczanych ani w wyrażeniach wyzwalaczy. W przypadku kluczy pozycji, które zwracają dane tylko w jednym określonym formacie, zgodny typ informacji jest wybierany automatycznie.
Host interface	Wybierz interfejs hosta. To pole jest dostępne podczas edycji pozycji na poziomie hosta.

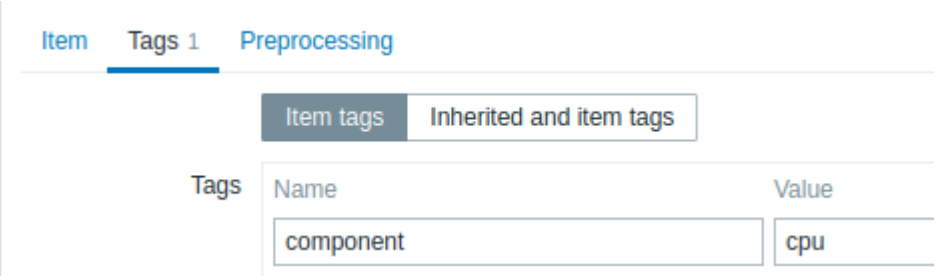
Parameter	Description
Units	Jeśli ustawiono symbol jednostki, Zabbix stosuje przetwarzanie końcowe do odebranej wartości pozycji i wyświetla ją z określonym sufiksem jednostki. Obsługiwane symbole jednostek ze specjalnym formatowaniem (i przykładami odebranej wartości pozycji → wyświetlanej wartości): **B** - bajty (1024 → 1 KB) **Bps** - bajty na sekundę (1024 → 1 KBps) **s** - sekundy, wyświetlane z użyciem maksymalnie trzech największych niezerowych jednostek czasu (881764 → 10d 4h 56m) **uptime** - czas, który upłynął, w formacie hh:mm:ss lub N days, hh:mm:ss (881764 → 10 days, 04:56:04) **unixtime** - znacznik czasu Unix, formatowany jako yyyy.mm.dd hh:mm:ss (881764 → 1970-01-11 04:56:04 AM); aby formatowanie było poprawne, odebrana wartość pozycji musi mieć typ *Numeric (unsigned)*. W przypadku innych jednostek (takich jak Hz, W itp.), jeśli odebrana wartość przekracza 1000, jest dzielona przez 1000 i wyświetlana z odpowiednim prefiksem (5000 → 5 KHz, 881764 → 881.76 KW). Jeśli symbol jednostki ma prefiks `!`, konwersja jednostek i dodawanie prefiksu są wyłączone (1024 !B → 1024 B, 61 !s → 61 s). Więcej przykładów i szczegółów dotyczących symboli jednostek oraz konwersji jednostek można znaleźć w sekcji Sufiksy wartości pozycji.
Update interval	Pobieraj nową wartość dla tej pozycji co N sekund. Maksymalny dozwolony interwał aktualizacji to 86400 sekund (1 dzień). Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Jedno makro musi wypełniać całe pole. Nie są obsługiwane wielokrotne makra w jednym polu ani makra połączone z tekstem. *Uwaga*: Interwał aktualizacji można ustawić na '0' tylko wtedy, gdy istnieją niestandardowe interwały z wartością różną od zera. Jeśli ustawiono '0' i istnieje niestandardowy interwał (elastyczny lub harmonogram) z wartością różną od zera, pozycja będzie odpytywana przez czas trwania niestandardowego interwału. *Uwaga*: pierwsze odpytywanie pozycji po jej aktywacji lub po zmianie interwału aktualizacji może nastąpić wcześniej niż skonfigurowana wartość. Nowe pozycje zostaną sprawdzone w ciągu 60 sekund od utworzenia, chyba że mają ustawiony interwał aktualizacji typu Scheduling lub Flexible, a *Update interval* jest ustawiony na 0. Istniejącą pasywną pozycję można odpytać o wartość natychmiast, naciskając przycisk *Execute now* button.
Custom intervals	Możesz utworzyć niestandardowe reguły sprawdzania pozycji: **Flexible** - utwórz wyjątek od *Update interval* (interwał o innej częstotliwości). **Scheduling** - utwórz niestandardowy harmonogram odpytywania. Szczegółowe informacje znajdziesz w sekcji Niestandardowe interwały. Obsługiwane są sufiksy czasu w polu *Interval*, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Jedno makro musi wypełniać całe pole. Nie są obsługiwane wielokrotne makra w jednym polu ani makra połączone z tekstem.
Timeout	Ustaw limit czasu sprawdzania pozycji (dostępne dla obsługiwanych typów pozycji). Wybierz opcję limitu czasu: **Global** - używany jest limit czasu proxy/globalny (wyświetlany w wyszarzonym polu *Timeout*). **Override** - używany jest niestandardowy limit czasu (ustawiany w polu *Timeout*; dozwolony zakres: 1 - 600s). Obsługiwane są sufiksy czasu, np. 30s, 1m, oraz makra użytkownika. Kliknięcie łącza *Timeouts* umożliwia skonfigurowanie limitów czasu proxy lub limitów czasu globalnych (jeśli proxy nie jest używany). Należy pamiętać, że łącze *Timeouts* jest widoczne tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administration* > *General* lub *Administration* > *Proxies*.

Parameter	Description
History	Wybierz jedną z opcji: **Do not store** - historia pozycji nie jest przechowywana. Przydatne dla pozycji nadrzędnych, jeśli tylko pozycje zależne muszą zachowywać historię. To ustawienie nie może zostać nadpisane przez globalne ustawienia housekeepera settings. **Store up to** - określ czas przechowywania szczegółowej historii w bazie danych (od 1 godziny do 25 lat). Starsze dane zostaną usunięte przez housekeepera. Wartość jest przechowywana w sekundach. Obsługiwane są sufiksy czasu, np. 2h, 1d. Obsługiwane są makra użytkownika. Wartość *Store up to* może zostać globalnie nadpisana w *Administration* > *Housekeeping*. Jeśli istnieje globalne ustawienie nadpisujące, wyświetlana jest pomarańczowa ikona informacji . Po najechaniu na nią kursorem wyświetlany jest komunikat ostrzegawczy, np. *Overridden by global housekeeper settings (1d)*. Zaleca się przechowywanie zarejestrowanych wartości przez możliwie najkrótszy czas, aby zmniejszyć rozmiar historii wartości w bazie danych. Zamiast przechowywać długą historię wartości, można przechowywać dłuższe dane trendów. Zobacz także Historia i trendy.
Trends	Wybierz jedną z opcji: **Do not store** - trendy nie są przechowywane. To ustawienie nie może zostać nadpisane przez globalne ustawienia housekeepera settings. **Store up to** - określ czas przechowywania zagregowanej historii (godzinowe min, max, avg, count) w bazie danych (od 1 dnia do 25 lat). Starsze dane zostaną usunięte przez housekeepera. Wartość jest przechowywana w sekundach. Obsługiwane są sufiksy czasu, np. 24h, 1d. Obsługiwane są makra użytkownika. Wartość *Store up to* może zostać globalnie nadpisana w *Administration* > *Housekeeping*. Jeśli istnieje globalne ustawienie nadpisujące, wyświetlana jest pomarańczowa ikona informacji . Po najechaniu na nią kursorem wyświetlany jest komunikat ostrzegawczy, np. *Overridden by global housekeeper settings (7d)*. *Uwaga:* Przechowywanie trendów nie jest dostępne dla danych nienumerycznych - character, log i text. Zobacz także Historia i trendy.
Value mapping	Zastosuj mapowanie wartości do tej pozycji. Mapowanie wartości nie zmienia odebranych wartości, służy wyłącznie do wyświetlania danych. Działa z pozycjami typu *Numeric(unsigned)*, *Numeric(float)* i *Character*. Na przykład „Windows service states”.
Log time format	Dostępne tylko dla pozycji typu **Log**. Obsługiwane symbole zastępcze: **y**: *Rok (1970-2038)*. **M**: *Miesiąc (01-12)*. **d**: *Dzień (01-31)*. **h**: *Godzina (00-23)*. **m**: *Minuta (00-59)*. **s**: *Sekunda (00-59)*. Jeśli pole pozostanie puste, znacznik czasu zostanie ustawiony na 0 w czasie Unix, co odpowiada 1 stycznia 1970. Na przykład rozważ następującą linię z pliku dziennika Zabbix agent: ” 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).” Zaczyna się ona od sześciu pozycji znakowych dla PID, po których następują data, czas i reszta komunikatu. Format czasu logu dla tej linii byłby następujący: ”pppppp:yyyyMMdd:hhmmss”. Należy pamiętać, że znaki ”p” i ”:” są symbolami zastępczymi i mogą być dowolnymi znakami z wyjątkiem ”yMdhms”.
Populates host inventory field	Możesz wybrać pole inwentarza hosta, które będzie wypełniane wartością pozycji. Będzie to działać, jeśli dla hosta włączone jest automatyczne uzupełnianie inwentarza. To pole nie jest dostępne, jeśli *Type of information* jest ustawione na ’Log’.
Description	Wprowadź opis pozycji. Obsługiwane są makra użytkownika.
Enabled	Zaznacz pole wyboru, aby włączyć pozycję i umożliwić jej przetwarzanie. Po wyłączeniu pozycji jest ona natychmiast usuwana z pamięci podręcznej historii (z wyjątkiem jej ostatniej wartości, która jest zachowywana dla logów).
Latest data	Kliknij łącze, aby wyświetlić najnowsze dane dla pozycji. To łącze jest dostępne tylko podczas edycji już istniejącej pozycji.

Note:
Pola specyficzne dla typu pozycji są opisane na odpowiednich stronach.

Note:
Podczas edycji istniejącej pozycji na poziomie szablon na poziomie hosta, wiele pól jest tylko do odczytu. Możesz użyć łącza w nagłówku formularza i przejść do poziomu szablonu, aby edytować je tam, pamiętając, że zmiany na poziomie szablonu zmienią pozycję dla wszystkich hostów, do których szablon jest podłączony.

Karta **Tags** umożliwia definiowanie tagów na poziomie pozycji.



Wstępne przetwarzanie wartości pozycji

Zakładka **Preprocessing** umożliwia zdefiniowanie reguł transformacji dla otrzymanych wartości.

Testowanie pozycji

Attention:
Aby przeprowadzić testowanie pozycji, upewnij się, że czas systemowy na serwerze i proxy jest zsynchronizowany. W przypadku gdy czas serwera jest opóźniony, testowanie pozycji może zwrócić komunikat o błędzie "The task has been expired." Ustawienie różnych stref czasowych na serwerze i proxy nie wpływa jednak na wynik testu.

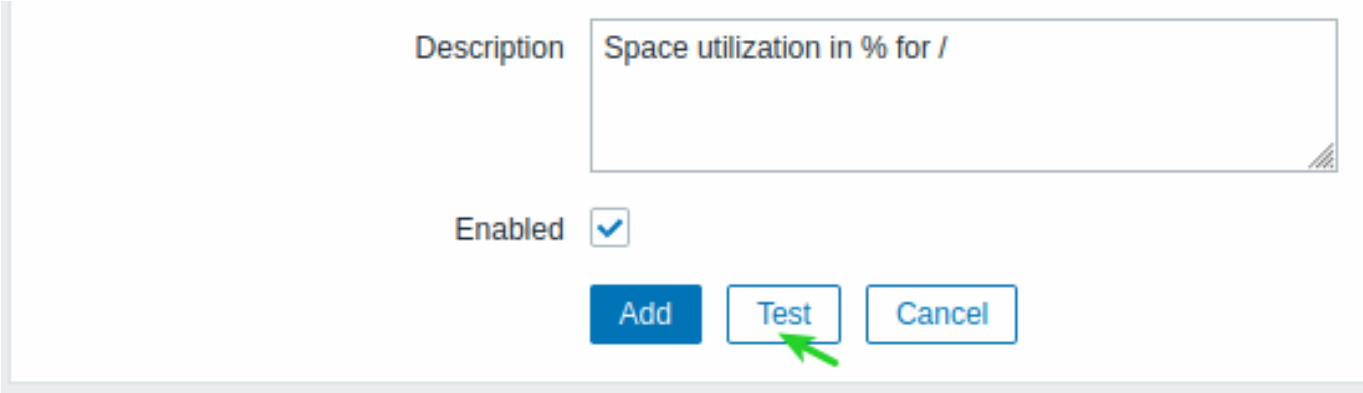
Możliwe jest przetestowanie pozycji i, jeśli jest poprawnie skonfigurowana, uzyskanie w odpowiedzi rzeczywistej wartości. Testowanie może odbywać się nawet przed zapisaniem pozycji.

Testowanie jest dostępne dla pozycji hosta i szablonu, prototypów pozycji oraz reguł wykrywania niskiego poziomu. Testowanie nie jest dostępne dla aktywnych pozycji.

Testowanie pozycji jest dostępne dla następujących pasywnych typów pozycji:

- Zabbix agent
- SNMP agent (v1, v2, v3)
- IPMI agent
- SSH checks
- Telnet checks
- JMX agent
- Simple checks (except `icmpping*`, `vmware.*` items)
- Zabbix internal
- Calculated items
- External checks
- Database monitor
- HTTP agent
- Script
- Browser

Aby przetestować pozycję, kliknij przycisk *Test* na dole formularza konfiguracji pozycji. Należy pamiętać, że przycisk *Test* będzie wyłączony dla pozycji, których nie można testować (takich jak aktywne sprawdzenia, wykluczone proste sprawdzenia).

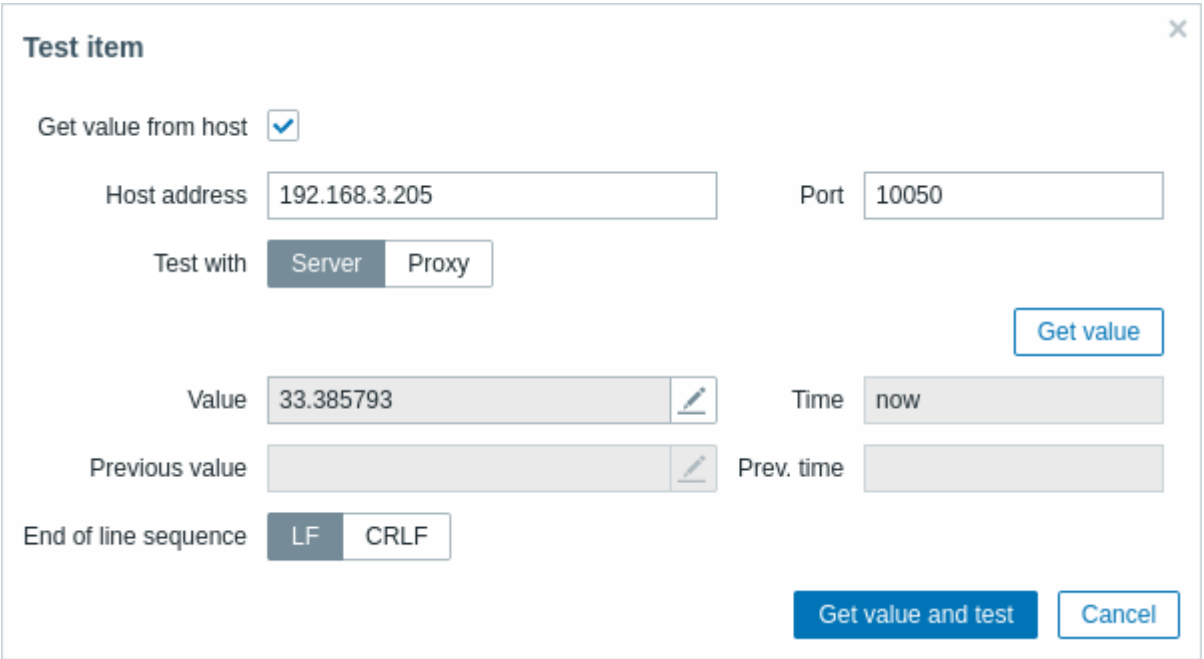


Formularz testowania pozycji zawiera pola wymaganych parametrów hosta (adres hosta, port, test z serwerem/proxy (nazwa proxy)) oraz szczegóły specyficzne dla pozycji (takie jak społeczność SNMPv2 lub poświadczenia bezpieczeństwa SNMPv3). Pola te są zależne od kontekstu:

- Wartości są wstępnie wypełniane, gdy to możliwe, tj. dla pozycji wymagających agenta, na podstawie informacji z wybranego interfejsu agenta hosta.
- Wartości muszą być wprowadzone ręcznie dla pozycji szablonu.
- Wartości makr w postaci zwykłego tekstu są rozwiązywane.
- Pola, w których wartość (lub jej część) jest sekretem lub makrem Vault, są puste i muszą zostać wprowadzone ręcznie. Jeśli dowolny parametr pozycji zawiera wartość makra sekretu, wyświetlany jest następujący komunikat ostrzegawczy: "Item contains user-defined macros with secret values. Values of these macros should be entered manually."
- Pola są wyłączone, gdy nie są potrzebne w kontekście typu pozycji (np. pole adresu hosta i pole proxy są wyłączone dla pozycji obliczanych)

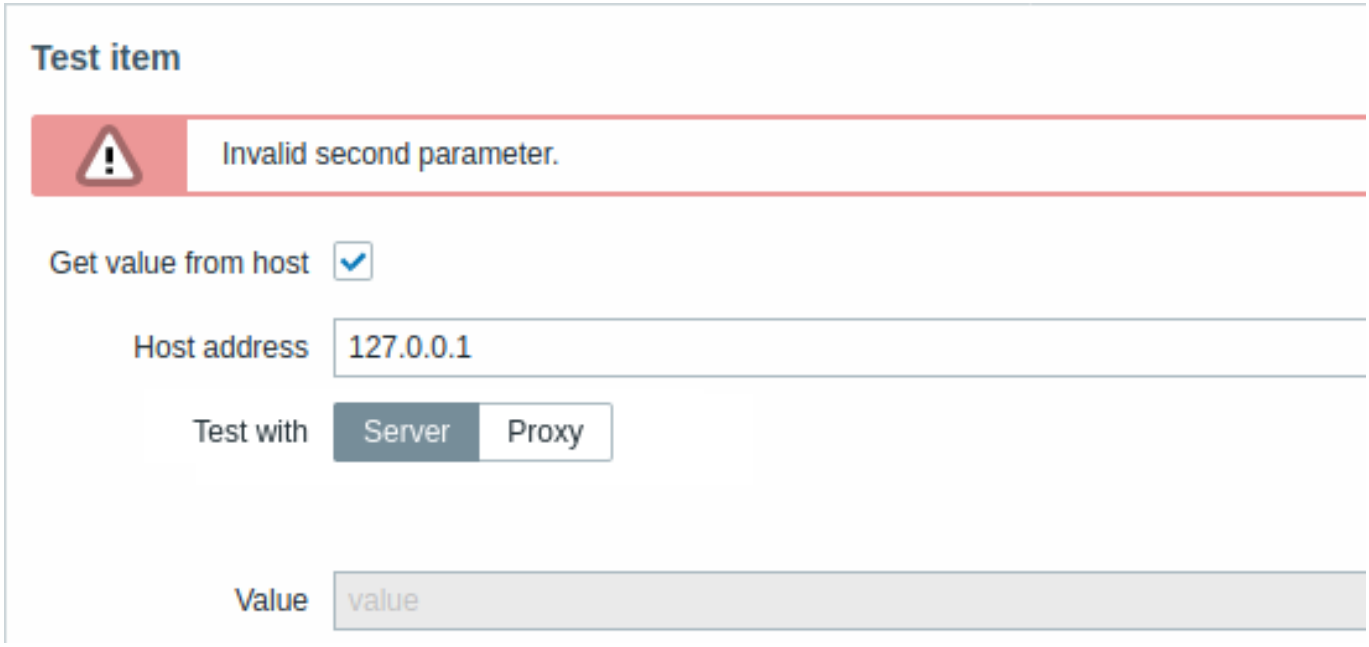
Aby przetestować pozycję, kliknij *Get value*. Jeśli wartość zostanie pobrana pomyślnie, zostanie wpisana do pola *Value*, a bieżąca wartość (jeśli istnieje) zostanie przeniesiona do pola *Previous value*, przy jednoczesnym obliczeniu pola *Prev. time*, tj. różnicy czasu między dwiema wartościami (kliknięciami), oraz próbie wykrycia sekwencji EOL i przełączenia na CRLF, jeśli w pobranej wartości zostanie wykryte "\n\r".

Kliknij *Get value and test*, aby przetestować przetwarzanie wstępne.



Wartości pobrane z hosta oraz wyniki testów są obcinane do maksymalnego rozmiaru 512 KB podczas wysyłania do frontend. Jeśli wynik zostanie obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu myszą. Należy pamiętać, że dane większe niż 512 KB są nadal w pełni przetwarzane przez serwer Zabbix.

Jeśli konfiguracja jest nieprawidłowa, wyświetlany jest komunikat o błędzie opisujący możliwą przyczynę.



Pomyślnie pobrana wartość z hosta może być również użyta do testowania kroków przetwarzania wstępnego.

Przyciski formularza

Przyciski na dole formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj pozycję. Ten przycisk jest dostępny tylko dla nowych pozycji.
Update	Zaktualizuj właściwości pozycji.
Clone	Utwórz kolejną pozycję na podstawie właściwości bieżącej pozycji.
Execute now	Natychmiast wykonaj sprawdzenie nowej wartości pozycji. Obsługiwane tylko dla sprawdzeń **pasywnych** (zobacz więcej szczegółów). *Uwaga:* podczas natychmiastowego sprawdzania wartości pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wartość nie będzie odzwierciedlać bardzo niedawnych zmian w konfiguracji pozycji.
Test	Przetestuj, czy konfiguracja pozycji jest poprawna, pobierając wartość.
Clear history and trends	Usuń historię i trendy pozycji.
Delete	Usuń pozycję.
Cancel	Anuluj edycję właściwości pozycji.

Limity danych tekstowych

Limity danych tekstowych zależą od backendu bazy danych. Przed zapisaniem wartości tekstowych w bazie danych są one przycinane, aby odpowiadały limitowi typu wartości w bazie danych:

Database	Type of information		
	Znakowy	**Log**	**Tekst**
MySQL	255 characters	65536 bytes	65536 bytes
PostgreSQL	255 characters	65536 characters	65536 characters
SQLite (only Zabbix proxy)	255 characters	65536 characters	65536 characters

Limit skryptu niestandardowego

Dostępna długość skryptu niestandardowego zależy od używanej bazy danych:

Baza danych	Limit znaków	Limit bajtów
MySQL	65535	65535
PostgreSQL	65535	bez ograniczeń
SQLite (tylko Zabbix proxy)	65535	bez ograniczeń

Limit czasu pozycji

Limit czasu pozycji określa, jak długo Zabbix powinien czekać przed przerwaniem sprawdzania jako nieudanego.

Po osiągnięciu limitu czasu sprawdzanie zostaje przerwane, nawet jeśli pobieranie danych nie zostało zakończone. Jeśli dane zostaną odebrane częściowo, pozycja staje się niewspierana, a w dzienniku zostaje zapisany komunikat o błędzie (na przykład, jeśli dane zostały pomyślnie pobrane tylko dla jednego z wielu OID-ów w sprawdzaniu SNMP).

W przypadku wielu typów pozycji można ustawić **elastyczne** limity czasu pozycji:

- dla pojedynczej pozycji
- dla typu pozycji (na poziomie proxy)
- dla typu pozycji (na poziomie globalnym)

Niestandardowy limit czasu *dla pojedynczej pozycji* jest przydatny, jeśli chcesz określić dłuższy limit czasu dla konkretnej pozycji, zachowując jednocześnie niskie limity czasu dla innych pozycji.

Priorytet limitów czasu

1. Limit czasu pojedynczej pozycji ma pierwszeństwo przed innymi limitami czasu.
2. Limity czasu na poziomie proxy mają pierwszeństwo przed globalnymi limitami czasu.

Limit czasu z konfiguracji Zabbix serwer lub proxy **nie** ma żadnego znaczenia w przypadku sprawdzeń, dla których ustawiono elastyczne limity czasu.

Obsługa elastycznego limitu czasu

Elastyczne limity czasu dla pozycji są obsługiwane dla następujących typów pozycji:

- Zabbix agent (zarówno dla kontroli pasywnych, jak i aktywnych)
- SNMP agent (z wyjątkiem starszych kontroli SNMP[1])
- Simple check (z wyjątkiem `icmpping*` oraz pozycji VMware[2])
- SSH agent
- Telnet agent
- External check
- Database monitor
- Script
- HTTP agent
- Browser

[1] W przypadku starszych kontroli SNMP (odpytywanie pojedynczego OID) mają zastosowanie ustawienia limitu czasu z konfiguracji serwer lub proxy.
 [2] Dla pozycji `icmpping*` wartość limitu czasu jest określana bezpośrednio w kluczu pozycji. Pozycje monitorowania VMware używają własnego parametru VMwareTimeout z konfiguracji serwer lub proxy.

Niewspierane pozycje

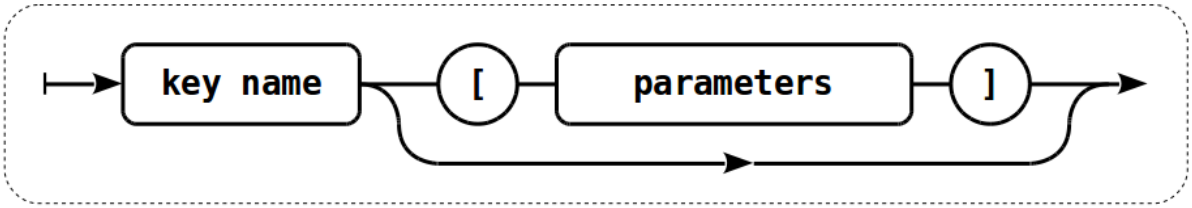
Niewspierane pozycje są zgłaszane ze statusem *Not supported* i nadal są ponownie sprawdzane zgodnie ze standardowym *Update interval*.

Pozycja staje się niewspierana, jeśli z jakiegoś powodu nie można pobrać jej wartości (np. błąd połączenia, brak skonfigurowanych pollerów do przetwarzania pozycji). Ponadto pozycje, które nie otrzymują danych historycznych, pozostają w statusie *Not supported*. Jest to oczekiwane zachowanie — pozycja przechodzi do statusu *Normal* dopiero wtedy, gdy otrzyma nowe, poprawne dane historyczne.

Podobnie jak problemy, niewspierane pozycje są ponownie oceniane tylko wtedy, gdy zostaną odebrane nowe dane — nawet jeśli dla danej pozycji nie są już dostępne żadne dane historyczne. Innymi słowy, pozycje i wyzwalacze zmieniają swój stan wyłącznie po otrzymaniu nowych danych. Jeśli jednak wyrażenie wyzwalacza zawiera funkcję date and time i/lub nodata(), wyzwalacz będzie przeliczany co 30 sekund (szczegóły: triggers calculation time).

1 Format klucza pozycji

Format klucza pozycji, w tym parametry klucza, musi być zgodny z regułami składni. Poniższe ilustracje przedstawiają obsługiwaną składnię. Dozwolone elementy i znaki w każdym miejscu można określić, podążając za strzałkami — jeśli do jakiegoś bloku można dotrzeć po linii, jest on dozwolony, jeśli nie — nie jest dozwolony.



Aby utworzyć prawidłowy klucz pozycji, należy zacząć od określenia nazwy klucza, następnie można wybrać, czy mają występować parametry, czy nie — jak pokazano przez dwie linie, którymi można podążać.

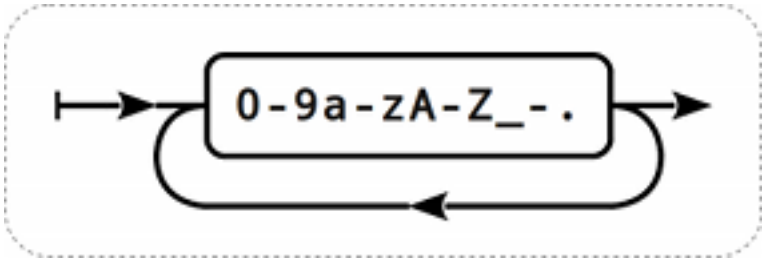
Nazwa klucza

Sama nazwa klucza ma ograniczony zakres dozwolonych znaków, które mogą występować jeden po drugim. Dozwolone znaki to:

`0-9a-zA-Z_-.`

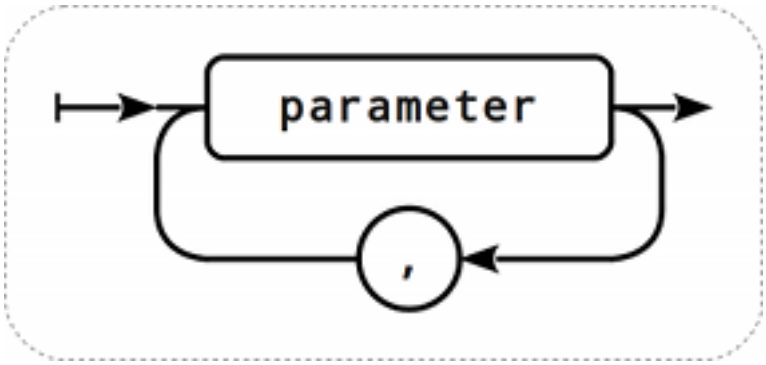
Co oznacza:

- wszystkie cyfry;
- wszystkie małe litery;
- wszystkie wielkie litery;
- podkreślenie;
- myślnik;
- kropka.

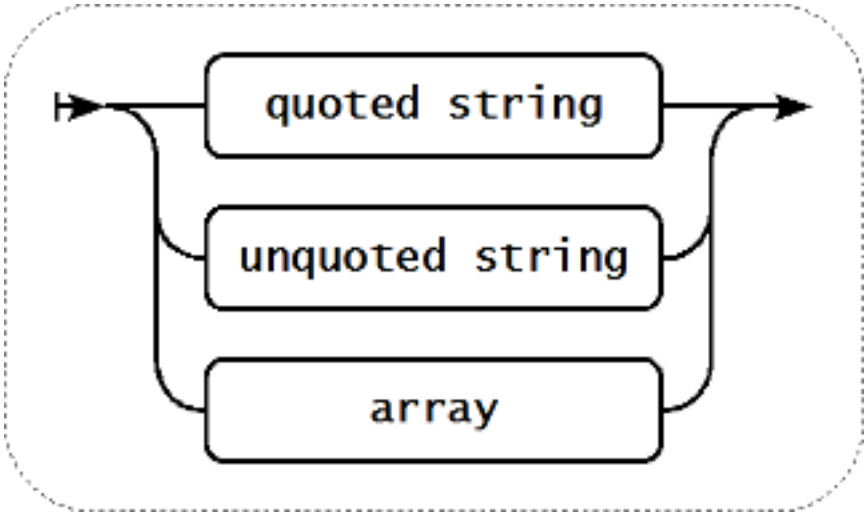


Kluczowe parametry

Klucz pozycji może mieć wiele parametrów oddzielonych przecinkami.



Każdy parametr klucza może być ujętym w cudzysłów ciągiem znaków, ciągiem znaków bez cudzysłowów albo tablicą.

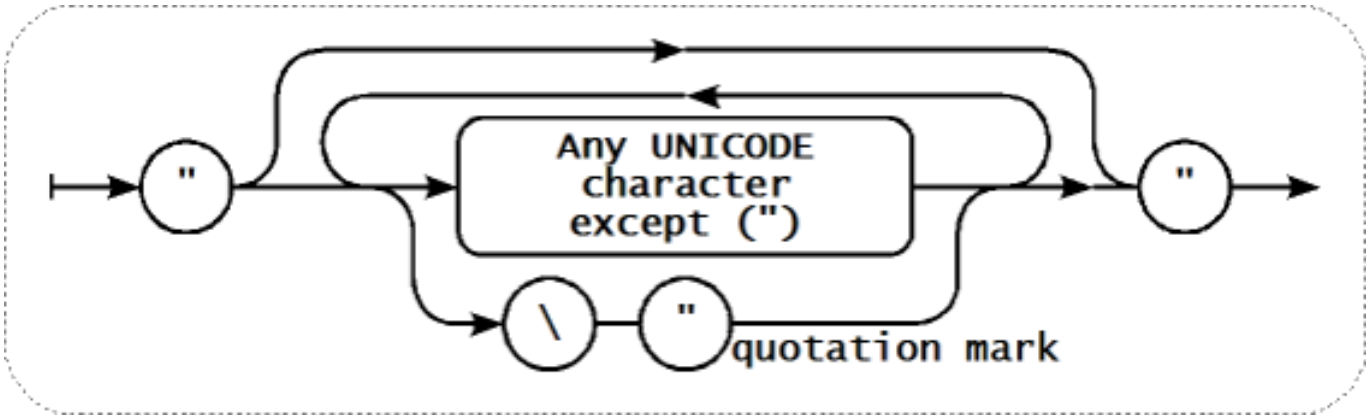


Parametr można również pozostawić pusty, używając w ten sposób wartości domyślnej. W takim przypadku należy dodać odpowiednią liczbę przecinków, jeśli określono jakiekolwiek dalsze parametry. Na przykład klucz pozycji **icmpping[„200„500]** oznacza, że interwał między poszczególnymi pingami wynosi 200 milisekund, limit czasu - 500 milisekund, a wszystkie pozostałe parametry pozostają ustawione na wartości domyślne.

Możliwe jest użycie makr w parametrach. Mogą to być makra użytkownika lub niektóre z wbudowanych makr. Aby sprawdzić, które konkretne wbudowane makra są obsługiwane w parametrach klucza pozycji, wyszukaj na stronie Obsługiwane makra frazę „item key parameters”.

Parametr - ciąg znaków w cudzysłowie

Jeśli parametr klucza jest ciągiem znaków ujętym w cudzysłów, dozwolony jest dowolny znak Unicode. Jeśli ciąg parametru klucza zawiera znak cudzysłowu, parametr ten musi być ujęty w cudzysłów, a każdy cudzysłów będący częścią ciągu parametru musi zostać poprzedzony znakiem ukośnika odwrotnego (\). Jeśli ciąg parametru klucza zawiera przecinek, parametr ten musi być ujęty w cudzysłów.

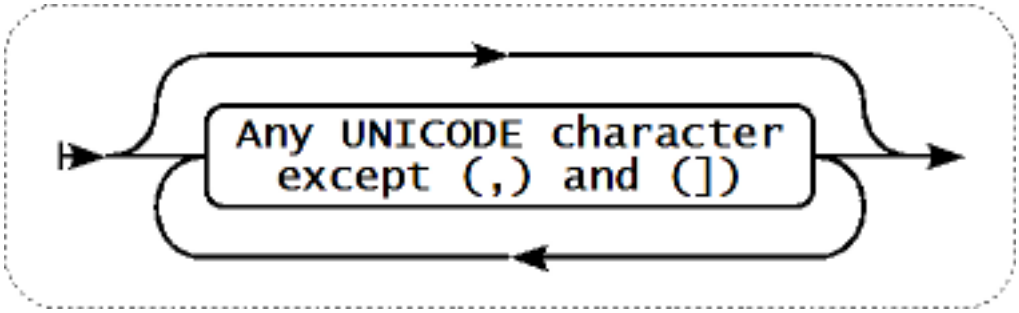


Warning:
Do ujmowania parametrów klucza pozycja w cudzysłów używaj wyłącznie podwójnych cudzysłowów. Pojedyncze cudzysłowy nie są obsługiwane.

Attention:
Wielopoziomowe tablice parametrów, np. `[a,[b,[c,d]],e]`, nie są dozwolone.

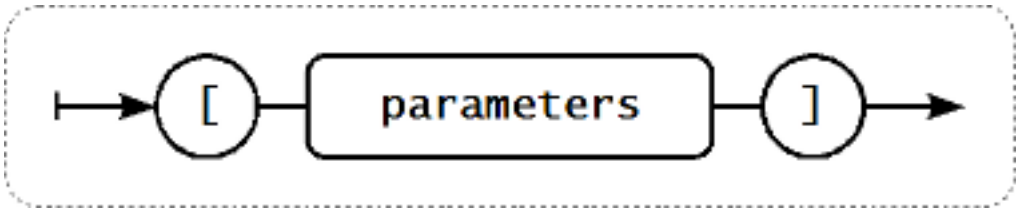
Parametr - ciąg znaków bez cudzysłowu

Jeśli parametr klucza jest ciągiem znaków bez cudzysłowu, dozwolony jest dowolny znak Unicode z wyjątkiem przecinka i prawego nawiasu kwadratowego (]). Parametr bez cudzysłowu nie może zaczynać się od lewego nawiasu kwadratowego ([).



Parametr - array

Jeśli parametr klucza jest tablicą, jest on ponownie ujmowany w nawiasy kwadratowe, gdzie poszczególne parametry są zgodne z zasadami i składnią określania wielu parametrów.



2 Interwały niestandardowe

Przegląd

Możliwe jest tworzenie własnych reguł dotyczących czasu, w którym pozycja jest sprawdzana. Służą do tego dwie metody: *Elastyczne interwały*, które pozwalają na ponowne zdefiniowanie domyślnego interwału aktualizacji, oraz *Harmonogramowanie*, dzięki któremu sprawdzenie pozycji może zostać wykonane o określonej godzinie lub w określonej sekwencji czasów.

Elastyczne interwały

Elastyczne interwały pozwalają przedefiniować domyślny interwał aktualizacji dla określonych okresów czasu. Elastyczny interwał jest definiowany za pomocą *Interwału* i *Okresu*, gdzie:

- *Interwał* — interwał aktualizacji dla określonego okresu czasu. Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d.
- *Okres* — okres czasu, w którym elastyczny interwał jest aktywny (szczegółowy opis formatu *Okresu* znajduje się w sekcji okresy czasu)

Jeśli wiele elastycznych interwałów nakłada się na siebie, dla nakładającego się okresu używana jest najmniejsza wartość *Interwału*. Należy pamiętać, że jeśli najmniejsza wartość nakładających się elastycznych interwałów wynosi „0”, odpytywanie nie będzie wykonywane. Poza elastycznymi interwałami używany jest domyślny interwał aktualizacji.

Należy pamiętać, że jeśli elastyczny interwał jest równy długości okresu, pozycja zostanie sprawdzona dokładnie raz. Jeśli elastyczny interwał jest większy niż okres, pozycja może zostać sprawdzona raz albo może nie zostać sprawdzona wcale (dlatego taka konfiguracja nie jest zalecana). Jeśli elastyczny interwał jest mniejszy niż okres, pozycja zostanie sprawdzona co najmniej raz.

Jeśli elastyczny interwał jest ustawiony na „0”, pozycja nie jest odpytywana w czasie trwania okresu elastycznego interwału i wznawia odpytywanie zgodnie z domyślnym *Interwałem aktualizacji* po zakończeniu tego okresu. Przykłady:

Interwał	Okres	Opis
10	1-5,09:00-18:00	Pozycja będzie sprawdzana co 10 sekund w godzinach pracy.
0	1-7,00:00-7:00	Pozycja nie będzie sprawdzana w nocy.
0	7-7,00:00-24:00	Pozycja nie będzie sprawdzana w niedziele.
60	1-7,12:00-12:01	Pozycja będzie sprawdzana codziennie o 12:00. Należy pamiętać, że było to używane jako obejście dla zaplanowanych kontroli i zaleca się używanie interwałów harmonogramu dla takich kontroli.

Interwały harmonogramu

Interwały harmonogramu są używane do sprawdzania pozycji o określonych porach. Podczas gdy elastyczne interwały służą do redefiniowania domyślnego interwału aktualizacji pozycji, interwały harmonogramu są używane do określenia niezależnego harmonogramu sprawdzania, który jest wykonywany równolegle.

Interwał harmonogramu jest definiowany jako: `md<filter>wd<filter>h<filter>m<filter>s<filter>`, gdzie:

- **md** - dni miesiąca
- **wd** - dni tygodnia
- **h** - godziny
- **m** - minuty
- **s** - sekundy

`<filter>` służy do określania wartości dla swojego prefiksu (dni, godziny, minuty, sekundy) i jest definiowany jako: `[<from>[-<to>]][/<step>][,<filter>]`, gdzie:

- `<from>` i `<to>` definiują zakres pasujących wartości (włącznie). Jeśli `<to>` zostanie pominięte, filtr dopasowuje zakres `<from> - <from>`. Jeśli `<from>` również zostanie pominięte, filtr dopasowuje wszystkie możliwe wartości.
- `<step>` definiuje krok przeskoku wartości liczbowych w obrębie zakresu. Domyślnie `<step>` ma wartość 1, co oznacza, że dopasowywane są wszystkie wartości zdefiniowanego zakresu.

Chociaż definicje filtrów są opcjonalne, musi zostać użyty co najmniej jeden filtr. Filtr musi mieć zdefiniowany albo zakres, albo wartość *<step>*.

Pusty filtr dopasowuje albo „0", jeśli nie zdefiniowano filtra niższego poziomu, albo wszystkie możliwe wartości w przeciwnym razie. Na przykład, jeśli filtr godzin zostanie pominięty, dopasowana zostanie tylko godzina „0", pod warunkiem że filtry minut i sekund również są pominięte; w przeciwnym razie pusty filtr godzin będzie dopasowywał wszystkie wartości godzin.

Prawidłowe wartości `<from>` i `<to>` dla odpowiednich prefiksów filtrów to:

Prefix	Opis	*<from>*	*<to>*
md	Dni miesiąca	1-31	1-31
wd	Dni tygodnia	1-7	1-7
h	Godziny	0-23	0-23
m	Minuty	0-59	0-59
s	Sekundy	0-59	0-59

Wartość `<from>` musi być mniejsza lub równa wartości `<to>`. Wartość `<step>` musi być większa lub równa 1 oraz mniejsza lub równa `<to>` - `<from>`.

Jednocyfrowe wartości dni miesiąca, godzin, minut i sekund mogą być poprzedzone zerem. Na przykład `md01-31` i `h/02` są prawidłowymi interwałami, ale `md01-031` i `wd01-07` już nie.

W Zabbix frontend wiele interwałów harmonogramu wprowadza się w osobnych wierszach. W Zabbix API są one łączone w jeden ciąg znaków z użyciem średnika `;` jako separatora.

Jeśli dany czas pasuje do kilku interwałów, zostanie wykonany tylko raz. Na przykład `wd1h9;h9` zostanie wykonane tylko raz w poniedziałek o 9:00.

Przykłady:

Interwał	Zostanie wykonane
m0-59	co minutę
h9-17/2	co 2 godziny, zaczynając od 9:00 (9:00, 11:00 ...)
m0,30 or m/30	co godzinę o hh:00 i hh:30
m0,5,10,15,20,25,30,35,40,45,50,55 or m/5	co pięć minut
wd1-5h9	w każdy poniedziałek do piątku o 9:00
wd1-5h9-18	w każdy poniedziałek do piątku o 9:00,10:00,...,18:00
h9,10,11 or h9-11	codziennie o 9:00, 10:00 i 11:00
md1h9m30	każdego 1. dnia miesiąca o 9:30
md1wd1h9m30	każdego 1. dnia miesiąca o 9:30, jeśli jest to poniedziałek
h9m/30	codziennie o 9:00, 9:30
h9m0-59/30	codziennie o 9:00, 9:30
h9,10m/30	codziennie o 9:00, 9:30, 10:00, 10:30
h9-10m30	codziennie o 9:30, 10:30
h9m10-40/30	codziennie o 9:10, 9:40
h9,10m10-40/30	codziennie o 9:10, 9:40, 10:10, 10:40
h9-10m10-40/30	codziennie o 9:10, 9:40, 10:10, 10:40
h9m10-40	codziennie o 9:10, 9:11, 9:12, ... 9:40
h9m10-40/1	codziennie o 9:10, 9:11, 9:12, ... 9:40
h9-12,15	codziennie o 9:00, 10:00, 11:00, 12:00, 15:00

Interwał	Zostanie wykonane
h9-12,15m0	codziennie o 9:00, 10:00, 11:00, 12:00, 15:00
h9-12,15m0s30	codziennie o 9:00:30, 10:00:30, 11:00:30, 12:00:30, 15:00:30
h9-12s30	codziennie o 9:00:30, 9:01:30, 9:02:30 ... 12:58:30, 12:59:30
h9m/30;h10 (*składnia specyficzna dla API*)	codziennie o 9:00, 9:30, 10:00
h9m/30 h10 (*dodaj to jako kolejny wiersz w frontend*)	codziennie o 9:00, 9:30, 10:00

Wyrównanie stref czasowych dla proxy i agent

Należy pamiętać, że proxy i agent Zabbix używają swoich lokalnych stref czasowych podczas przetwarzania interwałów harmonogramu.

Z tego powodu, gdy interwały harmonogramu są stosowane do pozycji monitorowanych przez proxy Zabbix lub aktywnych pozycji agent, zaleca się ustawienie strefy czasowej odpowiednich proxy lub agent na taką samą jak na serwerze Zabbix, w przeciwnym razie kolejka może nieprawidłowo zgłaszać opóźnienia pozycji.

Strefę czasową dla proxy Zabbix lub agent można ustawić za pomocą zmiennej środowiskowej TZ w pliku jednostki `systemd`:

```
[Service]
...
Environment="TZ=Europe/Amsterdam"
```

2 Wstępne przetwarzanie wartości pozycji

Przegląd

Wstępne przetwarzanie umożliwia stosowanie przekształceń do odebranych wartości pozycji przed zapisaniem ich do bazy danych. Te przekształcenia/kroki wstępnego przetwarzania są wykonywane przez serwer Zabbix lub proxy (jeśli pozycje są monitorowane przez proxy).

Ta funkcja obsługuje różne przypadki użycia, takie jak:

- konwersja bajtów na bity (np. mnożenie wartości ruchu sieciowego przez „8”);
- obliczanie statystyk na sekundę dla wartości narastających przyrostowo;
- stosowanie wyrażeń regularnych do wyodrębniania lub modyfikowania wartości;
- wykonywanie własnych skryptów na wartościach;
- odrzucanie niezmienionych wartości w celu optymalizacji przechowywania w bazie danych.

Dla pozycji można skonfigurować jeden lub więcej kroków wstępnego przetwarzania. Kroki te są wykonywane w kolejności, w jakiej zostały skonfigurowane.

Attention:
Jeśli krok wstępnego przetwarzania zakończy się niepowodzeniem, pozycja staje się nieobsługiwana. Można tego uniknąć dzięki obsłudze błędów *Custom on fail* (dostępnej dla większości przekształceń), która pozwala odrzucać wartości lub ustawiać wartości niestandardowe.

 W przypadku pozycji logów metadane logu (bez wartości) zawsze resetują stan nieobsługiwanej pozycji, ponownie czyniąc ją obsługiwaną. Dzieje się tak nawet wtedy, gdy początkowy błąd wystąpił po odebraniu wartości logu od agent.

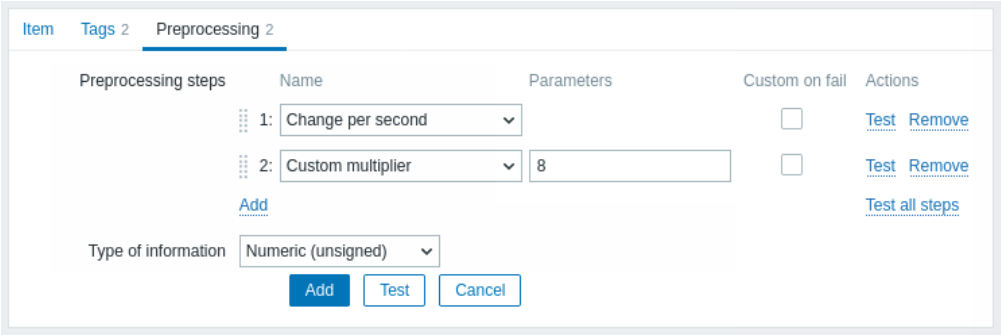
Wszystkie wartości przekazywane do wstępnego przetwarzania są początkowo traktowane jako ciągi znaków. Konwersja do żądanego typu wartości (zgodnie z definicją w konfiguracji pozycji) jest wykonywana na końcu potoku wstępnego przetwarzania. Jednak określone kroki wstępnego przetwarzania mogą wymuszać wcześniejsze konwersje, jeśli jest to wymagane. Szczegółowe informacje techniczne można znaleźć w sekcji Szczegóły wstępnego przetwarzania.

Aby upewnić się, że konfiguracja wstępnego przetwarzania działa zgodnie z oczekiwaniami, możesz ją przetestować.

Zobacz także: Przykłady wstępnego przetwarzania

Konfiguracja

Kroki przetwarzania wstępnego są definiowane na karcie **Preprocessing** formularza konfiguracji pozycji.



Kliknij *Add*, aby wybrać obsługiwaną transformację.

Pole *Type of information* jest wyświetlane na dole karty, gdy zdefiniowano co najmniej jeden krok przetwarzania wstępnego. W razie potrzeby można zmienić typ informacji bez opuszczania karty *Preprocessing*. Szczegółowy opis parametrów znajduje się w sekcji Tworzenie pozycji.

Obsługiwane transformacje

Wszystkie obsługiwane transformacje są wymienione poniżej. Kliknij nazwę transformacji, aby zobaczyć jej pełny opis.

Nazwa	Opis	Typ
Wyrażenie regularne	Dopasuj wartość do wyrażenia regularnego i zastąp ją wymaganym wynikiem.	Tekst
Zamień	Znajdź wyszukiwany ciąg i zastąp go innym (lub niczym).	
Przytnij	Usuń określone znaki z początku i końca wartości.	
Przytnij z prawej	Usuń określone znaki z końca wartości.	
Przytnij z lewej	Usuń określone znaki z początku wartości.	
XML XPath	Wyodrębnij wartość lub fragment z danych XML przy użyciu funkcji XPath.	Dane strukturalne
JSON Path	Wyodrębnij wartość lub fragment z danych JSON przy użyciu funkcji JSONPath.	
CSV do JSON	Konwertuj dane pliku CSV do formatu JSON.	
XML do JSON	Konwertuj dane w formacie XML do JSON.	
Wartość SNMP walk	Wyodrębnij wartość według określonej nazwy OID/MIB i zastosuj opcje formatowania.	SNMP
SNMP walk do JSON	Konwertuj wartości SNMP do JSON.	
Wartość SNMP get	Zastosuj opcje formatowania do wartości SNMP get.	
Mnożnik niestandardowy	Pomnóż wartość przez określoną liczbę całkowitą lub zmiennoprzecinkową.	Arytmetyka
Prosta zmiana	Oblicz różnicę między bieżącą a poprzednią wartością.	Zmiana
Zmiana na sekundę	Oblicz szybkość zmiany wartości na sekundę (różnicę między bieżącą a poprzednią wartością).	
Wartość logiczna na dziesiętną	Konwertuj wartość z formatu logicznego na dziesiętny.	Systemy liczbowe
Ósemkowy na dziesiętny	Konwertuj wartość z formatu ósemkowego na dziesiętny.	
Szesnastkowy na dziesiętny	Konwertuj wartość z formatu szesnastkowego na dziesiętny.	
JavaScript	Wprowadź kod JavaScript.	Skrypty niestandardowe
W zakresie	Zdefiniuj zakres, w którym powinna znajdować się wartość.	Walidacja
Pasuje do wyrażenia regularnego	Określ wyrażenie regularne, do którego wartość musi pasować.	
Nie pasuje do wyrażenia regularnego	Określ wyrażenie regularne, do którego wartość nie może pasować.	
Sprawdź błąd w JSON	Sprawdź komunikat błędu na poziomie aplikacji znajdujący się pod JSONPath.	
Sprawdź błąd w XML	Sprawdź komunikat błędu na poziomie aplikacji znajdujący się pod XPath.	
Sprawdź błąd przy użyciu wyrażenia regularnego	Sprawdź komunikat błędu na poziomie aplikacji przy użyciu wyrażenia regularnego.	
Sprawdź nieobsługiwaną wartość	Sprawdź, czy nie udało się pobrać żadnej wartości pozycji.	

Nazwa	Opis	Typ
Odrzuć niezmienione	Odrzuć wartość, jeśli nie uległa zmianie.	Ograniczanie częstotliwości
Odrzuć niezmienione z heartbeat	Odrzuć wartość, jeśli nie uległa zmianie w zdefiniowanym okresie czasu.	
Wzorzec Prometheus	Użyj poniższego zapytania, aby wyodrębnić wymagane dane z metryk Prometheus.	Prometheus
Prometheus do JSON	Konwertuj wymagane metryki Prometheus do JSON.	

Należy pamiętać, że dla kroków preprocessing typu *Zmiana* i *Ograniczanie częstotliwości* Zabbix musi zapamiętać ostatnią wartość, aby obliczyć/porównać nową wartość zgodnie z wymaganiami. Te poprzednie wartości są obsługiwane przez menedżera preprocessing. Jeśli serwer Zabbix lub proxy zostanie zrestartowany albo zostaną wprowadzone jakiekolwiek zmiany w krokach preprocessing, ostatnia wartość odpowiedniej pozycji zostanie zresetowana, co skutkuje:

- dla kroków *Prosta zmiana*, *Zmiana na sekundę* — następna wartość zostanie zignorowana, ponieważ nie ma poprzedniej wartości, na podstawie której można obliczyć zmianę;
- dla kroków *Odrzuć niezmienione*, *Odrzuć niezmienione z heartbeat* — następna wartość nigdy nie zostanie odrzucona, nawet jeśli powinna zostać odrzucona zgodnie z regułami odrzucania.

Wyrażenie regularne

Dopasuj wartość do wyrażenia regularnego i zastąp ją wymaganym wynikiem.

Parametry:

- **pattern** - wyrażenie regularne;
- **output** - szablon formatowania danych wyjściowych. Sekwencja ucieczki \N (gdzie N=1…9) jest zastępowana N-tą dopasowaną grupą. Sekwencja ucieczki \0 jest zastępowana dopasowanym tekstem.

Uwagi:

- Brak dopasowania wartości wejściowej spowoduje, że pozycja stanie się nieobsługiwana;
- Wyrażenie regularne obsługuje wyodrębnienie maksymalnie 10 przechwyconych grup za pomocą sekwencji \N;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.
- Przykłady można znaleźć w sekcji regular expressions.

Zamień

Znajdź wyszukiwany ciąg i zastąp go innym ciągiem (lub niczym).

Parametry:

- **wyszukiwany ciąg** - ciąg do znalezienia i zastąpienia, z rozróżnianiem wielkości liter (wymagane);
- **zamiennik** - ciąg, którym zostanie zastąpiony wyszukiwany ciąg. Ciąg zamiennika może być również pusty, co w praktyce pozwala usunąć wyszukiwany ciąg po jego znalezieniu.

Uwagi:

- Wszystkie wystąpienia wyszukiwanego ciągu zostaną zastąpione;
- Możliwe jest użycie sekwencji ucieczki do wyszukiwania lub zastępowania znaków nowej linii, powrotu karetki, tabulatorów i spacji ”\n \r \t \s”; ukośnik odwrotny można zapisać jako ”\\”, a sekwencje ucieczki można zapisać jako ”\\n”;
- Escapowanie znaków nowej linii, powrotu karetki i tabulatorów jest wykonywane automatycznie podczas wykrywania niskiego poziomu.

Przytnij

Usuń określone znaki z początku i końca wartości.

Przycinanie z prawej strony

Usuń określone znaki z końca wartości.

Przycinanie z lewej

Usuń określone znaki z początku wartości.

XML XPath

Wyodrębnij wartość lub fragment z danych XML przy użyciu funkcji XPath.

Uwagi:

- Aby ta opcja działała, serwer Zabbix (lub Zabbix proxy) musi być skompilowany z obsługą libxml;
- Przestrzenie nazw nie są obsługiwane;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Przykłady:

```
number(/document/item/value) #wyodrębni '10' z <document><item><value>10</value></item></document>
number(/document/item/@attribute) #wyodrębni '10' z <document><item attribute="10"></item></document>
/document/item #wyodrębni '<item><value>10</value></item>' z <document><item><value>10</value></item></doc
```

Ścieżka JSON

Wyodrębnij wartość lub fragment z danych JSON przy użyciu funkcjonalności JSONPath.

Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

CSV do JSON

Konwertuj dane pliku CSV do formatu JSON.

Więcej informacji znajduje się tutaj: Wstępne przetwarzanie CSV do JSON.

XML do JSON

Konwertuj dane w formacie XML do JSON.

Więcej informacji znajduje się tutaj: Reguły serializacji.

Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Wartość SNMP walk

Wyodrębnij wartość według określonej nazwy OID/MIB i zastosuj opcje formatowania:

- **Bez zmian** — zwróć ciąg szesnastkowy jako nieescapowany ciąg szesnastkowy (*uwaga*: wskazówki wyświetlania są nadal stosowane);
- **UTF-8 z hex-STRING** — przekonwertuj ciąg szesnastkowy na ciąg UTF-8;
- **MAC z hex-STRING** — zweryfikuj ciąg szesnastkowy jako adres MAC i zwróć poprawny ciąg adresu MAC (gdzie `' '` są zastępowane przez `':'`);
- **Liczba całkowita z BITS** — przekonwertuj pierwsze 8 bajtów ciągu bitowego wyrażonego jako sekwencja znaków szesnastkowych (np. „1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitowych dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane.

Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku nieudanego kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

SNMP walk do JSON

Konwertuj wartości SNMP do formatu JSON.

Określ nazwę pola w JSON oraz odpowiadającą jej ścieżkę SNMP OID. Wartości pól zostaną wypełnione wartościami z określonej ścieżki SNMP OID.

Uwagi:

- Dostępne są opcje formatowania wartości podobne do tych w kroku *Wartość SNMP walk*;
- Możesz użyć tego kroku przetwarzania wstępnego do wykrywania SNMP OID;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe będzie określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Wartość SNMP get

Zastosuj opcje formatowania do wartości SNMP get:

- **UTF-8 from Hex-STRING** — konwertuje ciąg szesnastkowy na ciąg UTF-8;
- **MAC from Hex-STRING** — sprawdza poprawność ciągu szesnastkowego jako adresu MAC i zwraca prawidłowy ciąg adresu MAC (gdzie ' ' są zastępowane przez ':');
- **Integer from BITS** — konwertuje pierwsze 8 bajtów ciągu bitowego wyrażonego jako sekwencja znaków szesnastkowych (np. „1A 2B 3C 4D") na 64-bitową liczbę całkowitą bez znaku. W ciągach bitowych dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane.

Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Mnożnik niestandardowy

Pomnóż wartość przez określoną liczbę całkowitą lub zmiennoprzecinkową.

Uwagi:

- Użyj tej opcji, aby konwertować wartości otrzymywane w KB, MBps itp. na B, Bps. W przeciwnym razie Zabbix nie będzie mógł poprawnie ustawić prefiksów (K, M, G itd.).
- Należy pamiętać, że jeśli typ informacji pozycji to *Numeric (unsigned)*, przychodzące wartości z częścią ułamkową zostaną obcięte (tj. „0.9" stanie się „0") przed zastosowaniem mnożnika niestandardowego;
- Jeśli używasz mnożnika niestandardowego lub przechowujesz wartość jako *Change per second* dla pozycji z typem informacji ustawionym na *Numeric (unsigned)*, a wynikowa obliczona wartość jest w rzeczywistości liczbą zmiennoprzecinkową, obliczona wartość nadal jest akceptowana jako poprawna po obcięciu części dziesiętnej i zapisaniu wartości jako liczby całkowitej;
- Obsługiwane: notacja naukowa, na przykład 1e+70; makra użytkownika i makra LLD; ciągi znaków zawierające makra, na przykład `{#MACRO}e+10`, `{$MACRO1}e+{$MACRO2}`. Makra muszą zostać rozwinięte do liczby całkowitej lub zmiennoprzecinkowej.
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Zmiana prosta

Oblicza różnicę między bieżącą a poprzednią wartością.

Uwagi:

- Ten krok może być przydatny do pomiaru stale rosnącej wartości;
- Obliczane jako **value**-**prev_value**, gdzie *value* — wartość bieżąca; *prev_value* — poprzednio odebrana wartość;
- Dozwolona jest tylko jedna operacja zmiany na pozycja („Zmiana prosta" lub „Zmiana na sekundę");
- Jeśli bieżąca wartość jest mniejsza od poprzedniej, Zabbix odrzuca tę różnicę (nic nie zapisuje) i czeka na kolejną wartość;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku nieudanego kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Zmiana na sekundę

Oblicza szybkość zmiany wartości (różnicę między bieżącą a poprzednią wartością) na sekundę.

Uwagi:

- Ten krok jest przydatny do obliczania szybkości na sekundę dla stale rosnącej wartości;
- Ponieważ to obliczenie może dawać liczby zmiennoprzecinkowe, zaleca się ustawienie pola „Type of information" na *Numeric (float)*, nawet jeśli przychodzące surowe wartości są liczbami całkowitymi. Jest to szczególnie istotne w przypadku małych liczb, gdzie część dziesiętna ma znaczenie. Jeśli wartości zmiennoprzecinkowe są duże i mogą przekroczyć długość pola „float", co może spowodować utratę całej wartości, zaleca się użycie *Numeric (unsigned)*, a tym samym obcięcie jedynie części dziesiętnej.
- Obliczane jako (**value**-**prev_value**)/(**time**-**prev_time**), gdzie *value* — bieżąca wartość; *prev_value* — poprzednio odebrana wartość; *time* — bieżący znacznik czasu; *prev_time* — znacznik czasu poprzedniej wartości;
- Dozwolona jest tylko jedna operacja zmiany na pozycja („Simple change" lub „Change per second");
- Jeśli bieżąca wartość jest mniejsza od poprzedniej, Zabbix odrzuca tę różnicę (nic nie zapisuje) i oczekuje na kolejną wartość. Pomaga to poprawnie działać na przykład w przypadku zawijania (przepełnienia) 32-bitowych liczników SNMP.
- Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Wartość logiczna na dziesiętną

Konwertuje wartość z formatu logicznego na dziesiętny.

Uwagi:

- Reprezentacja tekstowa jest tłumaczona odpowiednio na 0 lub 1. W związku z tym „TRUE” jest zapisywane jako 1, a „FALSE” jako 0. Wszystkie wartości są dopasowywane bez rozróżniania wielkości liter. Obecnie rozpoznawane wartości to, dla *TRUE* — true, t, yes, y, on, up, running, enabled, available, ok, master; dla *FALSE* — false, f, no, n, off, down, unused, disabled, unavailable, err, slave. Dodatkowo każda niezerowa wartość liczbowa jest uznawana za TRUE, a zero jest uznawane za FALSE.
- Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku nieudanego kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Ósemkowy na dziesiętny

Konwertuje wartość z formatu ósemkowego na dziesiętny.

Jeśli zaznaczysz pole wyboru *Niestandardowo w przypadku błędu*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Szesnastkowy na dziesiętny

Konwertuje wartość z formatu szesnastkowego na dziesiętny.

Jeśli zaznaczysz pole wyboru *Custom on fail*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

JavaScript

Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w polu parametru lub ikonie ołówka obok niego.

> **Attention:**
> Nie używaj niezadeklarowanych przypisań w kodzie JavaScript przetwarzania wstępnego. Użyj `var`, aby zadeklarować zmienne lokalne.

Uwagi:

- Dostępna długość kodu JavaScript zależy od używanej bazy danych;
- Więcej informacji można znaleźć tutaj: Przetwarzanie wstępne JavaScript.

W zakresie

Zdefiniuj zakres, w którym powinna znajdować się wartość, określając wartości minimalne/maksymalne (włącznie).

Uwagi:

- Akceptowane są wartości liczbowe (w tym dowolna liczba cyfr, opcjonalna część dziesiętna i opcjonalna część wykładnicza, wartości ujemne);
- Wartość minimalna powinna być mniejsza od wartości maksymalnej;
- Musi istnieć co najmniej jedna wartość;
- Można używać makr użytkownika i makr wykrywania niskiego poziomu;
- Jeśli zaznaczysz pole wyboru *Niestandardowo w przypadku błędu*, można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku nieudanego kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Pasuje do wyrażenia regularnego

Określ wyrażenie regularne, do którego wartość musi pasować.

Jeśli zaznaczysz pole wyboru *Niestandardowe przy błędzie*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Nie pasuje do wyrażenia regularnego

Określ wyrażenie regularne, do którego wartość nie może pasować.

Jeśli zaznaczysz pole wyboru *Niestandardowo w przypadku błędu*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Sprawdzenie błędu w JSON

Sprawdza komunikat błędu na poziomie aplikacji znajdujący się pod JSONPath. Przetwarzanie zostanie zatrzymane, jeśli operacja się powiedzie, a komunikat nie będzie pusty; w przeciwnym razie przetwarzanie będzie kontynuowane z wartością sprzed tego kroku przetwarzania wstępnego.

Uwagi:

- Te błędy usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku przetwarzania wstępnego;
- W przypadku niepowodzenia parsowania nieprawidłowego JSON nie zostanie zgłoszony żaden błąd;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie własnych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku nieudanego kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Sprawdzenie błędu w XML

Sprawdza komunikat błędu na poziomie aplikacji znajdujący się pod XPath. Przetwarzanie zostanie zatrzymane, jeśli zakończy się powodzeniem i komunikat nie jest pusty; w przeciwnym razie przetwarzanie będzie kontynuowane z wartością sprzed tego kroku wstępnego przetwarzania.

Uwagi:

- Błędy tych usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku wstępnego przetwarzania;
- W przypadku niepowodzenia parsowania nieprawidłowego XML nie zostanie zgłoszony żaden błąd;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku nieudanego kroku wstępnego przetwarzania pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Sprawdzenie błędu za pomocą wyrażenia regularnego

Sprawdź komunikat błędu na poziomie aplikacji za pomocą wyrażenia regularnego. Zatrzymaj przetwarzanie, jeśli zakończy się powodzeniem i komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością, która była przed tym krokiem przetwarzania wstępnego.

Parametry:

- **pattern** - wyrażenie regularne;
- **output** - szablon formatowania danych wyjściowych. Sekwencja ucieczki \N (gdzie N=1…9) jest zastępowana N-tą dopasowaną grupą. Sekwencja ucieczki \0 jest zastępowana dopasowanym tekstem.

Uwagi:

- Te błędy usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku przetwarzania wstępnego;
- Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu błędu. W przypadku niepowodzenia kroku przetwarzania wstępnego pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.

Sprawdzenie nieobsługiwanej wartości

Sprawdź, czy nie udało się pobrać żadnej wartości pozycji. Określ, jak przetworzyć błąd, na podstawie analizy zwróconego komunikatu o błędzie.

Parametry:

- **scope** — wybierz zakres przetwarzania błędu:
dowolny błąd — dowolny błąd;
błąd pasuje — tylko błąd pasujący do wyrażenia regularnego określonego w *pattern*;
błąd nie pasuje — tylko błąd, który nie pasuje do wyrażenia regularnego określonego w *pattern*
- **pattern** — wyrażenie regularne, do którego ma pasować błąd. Jeśli w parametrze scope wybrano *dowolny błąd*, to pole to nie jest wyświetlane. Jeśli jest wyświetlane, to pole jest wymagane.

Uwagi:

- Zwykle brak wartości lub niepowodzenie pobrania wartości powoduje, że pozycja staje się nieobsługiwana. Ten krok preprocessing umożliwia zmianę tego zachowania. Jeśli zaznaczysz pole wyboru *Custom on fail* (dla tego kroku preprocessing zawsze zaznaczone i wyszarzone), można określić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie. W przypadku niepowodzenia kroku preprocessing pozycja nie stanie się nieobsługiwana, jeśli wybrano opcję odrzucenia wartości lub ustawienia określonej wartości.
- Ten krok preprocessing sprawdza tylko, czy nie udało się pobrać wartości pozycji. Nie sprawdza na przykład, czy typ pobranej wartości (np. string) odpowiada typowi informacji pozycji (np. numeric); szczegóły znajdują się w Przykładach preprocessing. Jeśli występuje niezgodność typów, pozycja może nadal stać się nieobsługiwana po wykonaniu wszystkich kroków preprocessing. Aby sprawdzić niezgodność typów, można użyć na przykład kroku preprocessing *Custom multiplier*; zobacz Przykłady preprocessing.
- Przechwytywanie grup wyrażeń regularnych jest obsługiwane w polach *Set value to* lub *Set error to*. Użyj \N (gdzie N=1…9), aby pobrać N-tą dopasowaną grupę; użyj \0, aby pobrać dopasowany tekst;
- Te kroki są zawsze wykonywane jako pierwsze kroki preprocessing i po zapisaniu zmian w pozycji są umieszczane nad wszystkimi pozostałymi;
- Obsługiwanych jest wiele kroków *Check for not supported value*, w określonej kolejności. Krok dla *dowolny błąd* zostanie automatycznie umieszczony jako ostatni krok w tej grupie.

Odrzuć niezmienione

Odrzuć wartość, jeśli nie uległa zmianie.

Uwagi:

- Jeśli wartość zostanie odrzucona, nie jest zapisywana w bazie danych, a serwer Zabbix nie ma wiedzy o tym, że ta wartość została odebrana. Wyrażenia wyzwalaczy nie będą oceniane, w rezultacie nie zostaną utworzone/rozwiązane żadne problemy dla powiązanych wyzwalaczy. Funkcje będą działać wyłącznie na podstawie danych, które są faktycznie zapisane w bazie danych. Ponieważ trendy są tworzone na podstawie danych w bazie danych, jeśli przez godzinę nie zostanie zapisana żadna wartość, to dla tej godziny również nie będzie danych trendów.
- Dla jednej pozycji można określić tylko jedną opcję ograniczania.

Odrzuć niezmienione z heartbeat

Odrzuć wartość, jeśli nie zmieniła się w zdefiniowanym okresie czasu (w sekundach).

Uwagi:

- Obsługiwane są dodatnie wartości całkowite do określenia liczby sekund (minimum — 1 sekunda);
- Można używać sufiksów czasu (np. 30s, 1m, 2h, 1d);
- Można używać makr użytkownika i makr wykrywania niskiego poziomu;
- Jeśli wartość zostanie odrzucona, nie jest zapisywana w bazie danych, a serwer Zabbix nie ma informacji, że ta wartość została odebrana. Żadne wyrażenia wyzwalaczy nie będą obliczane, w rezultacie nie zostaną utworzone/rozwiązane żadne problemy dla powiązanych wyzwalaczy. Funkcje będą działać wyłącznie na podstawie danych, które są faktycznie zapisane w bazie danych. Ponieważ trendy są tworzone na podstawie danych w bazie danych, jeśli przez godzinę nie zostanie zapisana żadna wartość, to również nie będzie danych trendów dla tej godziny.
- Dla jednej pozycji można określić tylko jedną opcję ograniczania.

Wzorzec Prometheus

Użyj następującego zapytania, aby wyodrębnić wymagane dane z metryk Prometheus.

Więcej informacji znajdziesz w sekcji Sprawdzanie Prometheus.

Prometheus do JSON

Konwertuje wymagane metryki Prometheus do formatu JSON.

Więcej informacji znajdziesz w sekcji Sprawdzanie Prometheus.

Obsługa makr

Makra użytkownika oraz makra użytkownika z kontekstem są obsługiwane w:

- parametrach kroku przetwarzania wstępnego, w tym w kodzie JavaScript;
- parametrach niestandardowej obsługi błędów (pola *Ustaw wartość na* i *Ustaw błąd na*).

> **Note:**
> Kontekst makra jest ignorowany, gdy makro jest zastępowane swoją wartością. Wartość makra jest wstawiana do kodu w postaci niezmienionej; nie ma możliwości dodania dodatkowego escapingu przed umieszczeniem wartości w kodzie JavaScript. Należy pamiętać, że w niektórych przypadkach może to powodować błędy JavaScript.

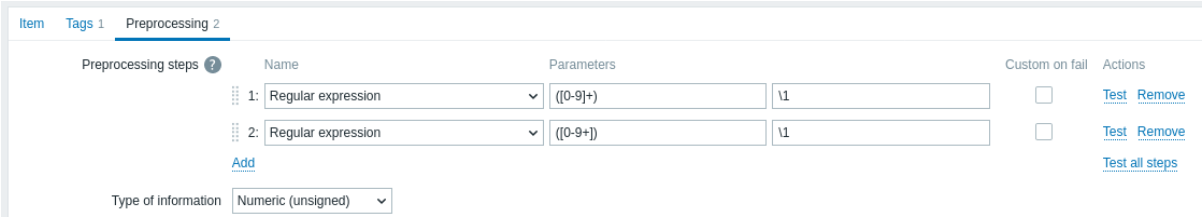
Testowanie

Zobacz testowanie preprocessing.

1 Testowanie przetwarzania wstępnego

Testowanie

Testowanie kroków przetwarzania wstępnego jest przydatne, aby upewnić się, że złożone potoki przetwarzania wstępnego dają oczekiwane wyniki, bez konieczności czekania na odebranie i wstępne przetworzenie wartości pozycji.

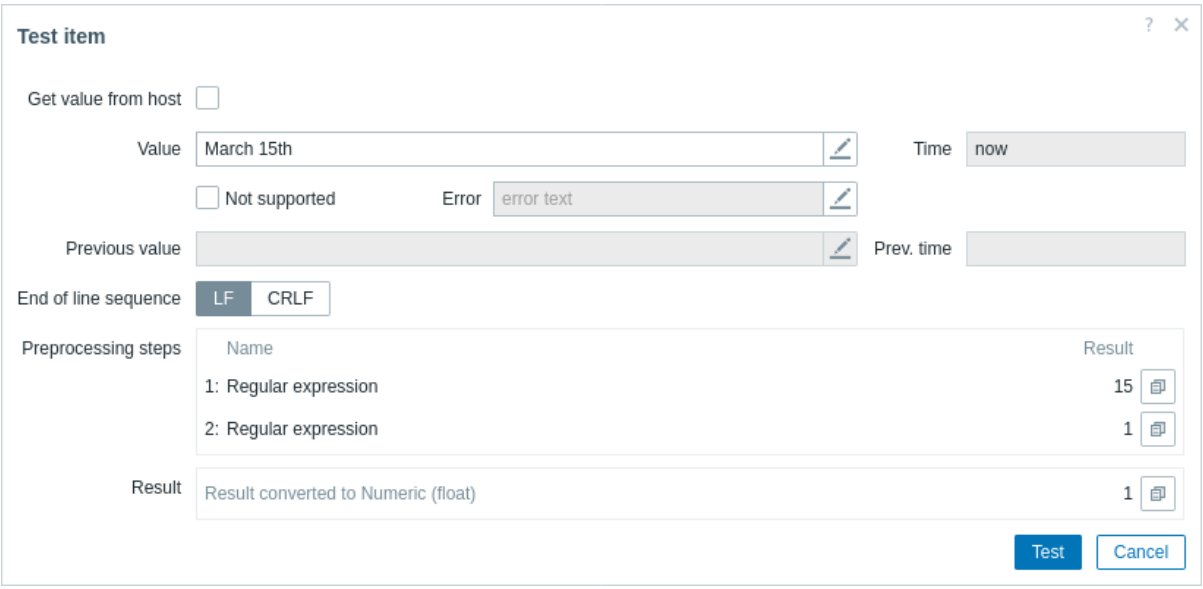


Możliwe jest testowanie:

- względem hipotetycznej wartości
- względem rzeczywistej wartości z hosta

Każdy krok przetwarzania wstępnego można testować indywidualnie, a także można testować wszystkie kroki razem. Po kliknięciu odpowiednio przycisku *Test* lub *Test all steps* w bloku Actions otwierane jest okno testowania.

Testowanie wartości hipotetycznej



Parametr	Opis
Pobierz wartość z hosta	Jeśli chcesz przetestować wartość hipotetyczną, pozostaw to pole wyboru niezaznaczone. Zobacz także: Testowanie wartości rzeczywistej.
Wartość	Wprowadź wartość wejściową do przetestowania. Kliknięcie w polu parametru lub przycisku podglądu/edycji otworzy okno obszaru tekstowego do wprowadzenia wartości lub bloku kodu.
Nieobsługiwane	Zaznacz to pole wyboru, aby przetestować nieobsługiwaną wartość. Ta opcja jest przydatna do testowania kroku wstępnego przetwarzania *Sprawdź pod kątem nieobsługiwanej wartości*.
Błąd	Wprowadź tekst błędu. To pole jest włączone, gdy opcja *Pobierz wartość z hosta* jest odznaczona, a opcja *Nieobsługiwane* jest zaznaczona. Jeśli opcja *Pobierz wartość z hosta* jest zaznaczona, to pole zostanie wypełnione rzeczywistym komunikatem błędu (tylko do odczytu) z hosta.
Czas	Wyświetlany jest czas wartości wejściowej: `now` (tylko do odczytu).
Poprzednia wartość	Wprowadź poprzednią wartość wejściową do porównania. Tylko dla kroków wstępnego przetwarzania *Zmiana* i *Ograniczanie*.

Parametr	Opis
Poprzedni czas	Wprowadź czas poprzedniej wartości wejściowej do porównania. Tylko dla kroków wstępnego przetwarzania *Zmiana* i *Ograniczanie*. Wartość domyślna jest oparta na wartości pola „Interwał aktualizacji" pozycji (jeśli „1m", to to pole zostanie wypełnione wartością `now-1m`). Jeśli nic nie określono lub użytkownik nie ma dostępu do hosta, wartością domyślną jest `now-30s`.
Makra	Jeśli używane są jakiekolwiek makra, są one wyświetlane wraz z ich wartościami. Wartości można edytować do celów testowych, ale zmiany zostaną zapisane tylko w kontekście testowania.
Sekwencja końca linii	Wybierz sekwencję końca linii dla wielowierszowych wartości wejściowych: **LF** - sekwencja LF (line feed) **CRLF** - sekwencja CRLF (carriage-return line-feed).
Kroki wstępnego przetwarzania	Wyświetlane są kroki wstępnego przetwarzania; wynik testu jest wyświetlany dla każdego kroku po kliknięciu przycisku *Test*. Wyniki testu są obcinane do maksymalnego rozmiaru 512KB podczas wysyłania do frontend. Wyniki testu można kopiować (nie więcej niż obcięte 512KB). Jeśli wynik zostanie obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu kursorem. Należy pamiętać, że dane większe niż 512KB są nadal w pełni przetwarzane przez serwer Zabbix. Jeśli krok zakończył się niepowodzeniem podczas testowania, wyświetlana jest ikona błędu. Opis błędu jest wyświetlany po najechaniu kursorem. Jeśli dla kroku określono „Custom on fail" i to działanie zostało wykonane, bezpośrednio po wierszu kroku testu wstępnego przetwarzania pojawi się nowy wiersz pokazujący, jakie działanie zostało wykonane i jaki wynik zostało uzyskany (błąd lub wartość).
Wynik	Końcowy wynik testowania kroków wstępnego przetwarzania jest wyświetlany we wszystkich przypadkach, gdy wszystkie kroki są testowane razem (po kliknięciu przycisku *Test all steps*). Wyświetlany jest także typ konwersji do typu wartości pozycji, na przykład `Result converted to Numeric (unsigned)`. Wyniki testu są obcinane do maksymalnego rozmiaru 512KB podczas wysyłania do frontend. Końcowy wynik można kopiować (nie więcej niż obcięte 512KB). Jeśli wynik zostanie obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu kursorem. Należy pamiętać, że dane większe niż 512KB są nadal w pełni przetwarzane przez serwer Zabbix.

Kliknij *Test*, aby zobaczyć wynik po każdym kroku wstępnego przetwarzania.

Wartości testowe są przechowywane między sesjami testowymi zarówno dla poszczególnych kroków, jak i dla wszystkich kroków, co pozwala użytkownikowi zmieniać kroki wstępnego przetwarzania lub konfigurację pozycji, a następnie wrócić do okna testowania bez konieczności ponownego wprowadzania informacji. Wartości są jednak tracone po odświeżeniu strony.

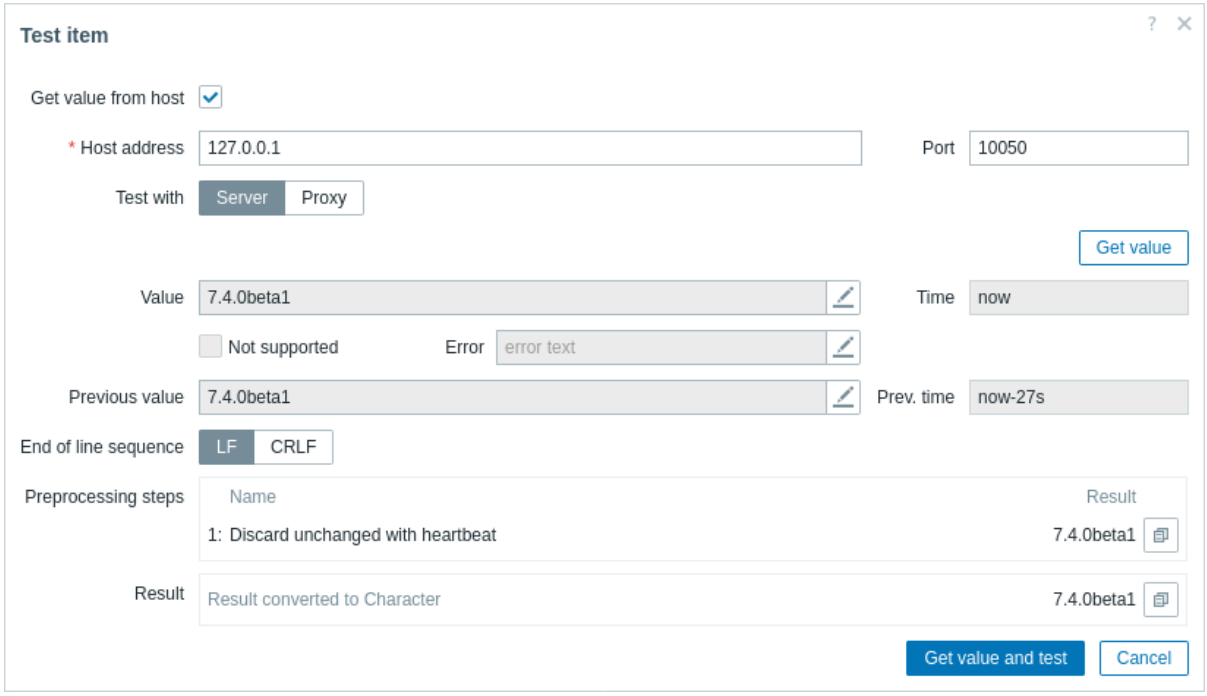
Testowanie jest wykonywane przez serwer Zabbix. frontend wysyła odpowiednie żądanie do serwera i oczekuje na wynik. Żądanie zawiera wartość wejściową i kroki wstępnego przetwarzania (z rozwiniętymi makrami użytkownika). Dla kroków *Zmiana* i *Ograniczanie* można opcjonalnie określić poprzednią wartość i czas. Serwer odpowiada wynikami dla każdego kroku wstępnego przetwarzania.

Wszystkie błędy techniczne lub błędy walidacji danych wejściowych są wyświetlane w polu błędów u góry okna testowania.

Testowanie rzeczywistej wartości

Aby przetestować przetwarzanie wstępne względem rzeczywistej wartości:

- Zaznacz pole wyboru *Get value from host*
- Wprowadź lub zweryfikuj parametry hosta (adres hosta, port, nazwa proxy/brak proxy) oraz szczegóły specyficzne dla pozycji (takie jak społeczność SNMPv2 lub poświadczenia bezpieczeństwa SNMPv3). Te pola są zależne od kontekstu:
  - Wartości są wstępnie uzupełniane, gdy to możliwe, tj. dla pozycji wymagających agenta, poprzez pobranie informacji z wybranego interfejsu agenta hosta
  - Wartości muszą zostać wypełnione ręcznie dla pozycji szablonu
  - Wartości makr w postaci zwykłego tekstu są rozwijane
  - Jeśli wartość pola (lub część wartości) jest sekretem lub makrem Vault, pole będzie puste i trzeba je będzie wypełnić ręcznie. Jeśli którykolwiek parametr pozycji zawiera wartość sekretnego makra, wyświetlany jest następujący komunikat ostrzegawczy: "Item contains user-defined macros with secret values. Values of these macros should be entered manually."
  - Pola są wyłączone, gdy nie są potrzebne w kontekście typu pozycji (np. pola adresu hosta i proxy są wyłączone dla pozycji obliczanych)
- Kliknij *Get value and test*, aby przetestować przetwarzanie wstępne



Jeśli w formularzu konfiguracji pozycji określono mapowanie wartości (pole „Show value”), okno dialogowe testu pozycji wyświetli po wyniku końcowym dodatkowy wiersz o nazwie „Result with value map applied”.

Parametry specyficzne dla pobierania rzeczywistej wartości z hosta:

Parameter	Description
Get value from host	Zaznacz to pole wyboru, aby pobrać rzeczywistą wartość z hosta.
Host address	Wprowadź adres hosta. To pole jest automatycznie wypełniane adresem interfejsu hosta pozycji.
Port	Wprowadź port hosta. To pole jest automatycznie wypełniane portem interfejsu hosta pozycji.
Additional fields for SNMP interfaces (SNMP version, SNMP community, Context name, etc.)	Zobacz Configuring SNMP monitoring, aby uzyskać dodatkowe informacje na temat konfiguracji interfejsu SNMP (v1, v2 i v3). Pola te są automatycznie wypełniane na podstawie interfejsu hosta pozycji.
Proxy	Określ proxy, jeśli host jest monitorowany przez proxy. To pole jest automatycznie wypełniane proxy hosta (jeśli istnieje).
Value	Wartość pobrana z hosta. Kliknięcie w polu parametru lub przycisku podglądu/edycji otworzy okno obszaru tekstowego z wartością lub blokiem kodu. Wartości są obcinane do maksymalnego rozmiaru 512KB i tylko we frontendzie. Jeśli wynik zostanie obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu kursorem. Należy pamiętać, że dane większe niż 512KB są nadal w pełni przetwarzane przez serwer Zabbix.

Dla pozostałych parametrów zobacz Testing hypothetical value powyżej.

2 Szczegóły przetwarzania wstępnego

Przegląd

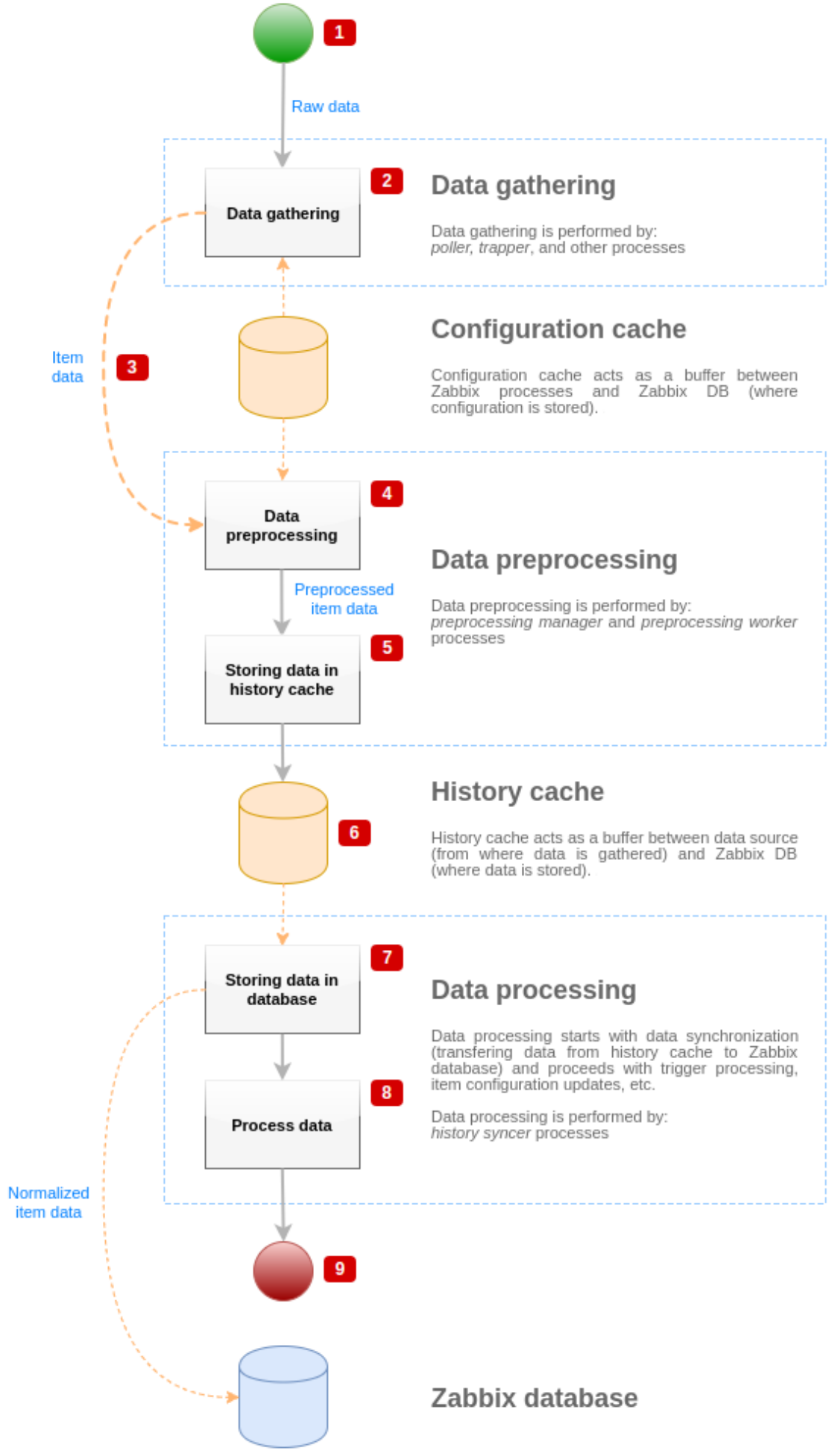
Ta sekcja zawiera szczegóły dotyczące wstępnego przetwarzania wartości pozycji. Wstępne przetwarzanie wartości pozycji umożliwia definiowanie i wykonywanie reguł transformacji dla otrzymanych wartości pozycji.

Wstępne przetwarzanie jest zarządzane przez proces menedżera wstępnego przetwarzania wraz z workerami wstępnego przetwarzania, które wykonują poszczególne kroki wstępnego przetwarzania. Wszystkie wartości z wstępnym przetwarzaniem (przed Zabbix 7.4.1, wszystkie wartości), otrzymywane od różnych zbieraczy danych, przechodzą przez menedżera wstępnego przetwarzania przed dodaniem do pamięci podręcznej historii. Do komunikacji IPC opartej na gniazdach używany jest kanał między

zbieraczami danych (pollery, trappery itp.) a procesem wstępnego przetwarzania. Kroki wstępnego przetwarzania wykonuje albo serwer Zabbix, albo proxy Zabbix (dla pozycji monitorowanych przez proxy).

Przetwarzanie wartości pozycji

Aby zobrazować przepływ danych od źródła danych do bazy danych Zabbix, możemy użyć następującego uproszczonego diagramu:


1
Raw data
Data gathering
2
Data gathering
Data gathering is performed by:
poller, trapper, and other processes
Item data
3
Configuration cache
Configuration cache acts as a buffer between Zabbix processes and Zabbix DB (where configuration is stored).
Data preprocessing
4
Data preprocessing
Data preprocessing is performed by:
preprocessing manager and preprocessing worker processes
Preprocessed item data
Storing data in history cache
5
History cache
6
History cache acts as a buffer between data source (from where data is gathered) and Zabbix DB (where data is stored).
Storing data in database
7
Data processing
Data processing starts with data synchronization (transfering data from history cache to Zabbix database) and proceeds with trigger processing, item configuration updates, etc.
Process data
8
Data processing is performed by:
history syncer processes
Normalized item data
9
Zabbix database

Powyższy diagram pokazuje tylko procesy, obiekty i działania związane z przetwarzaniem wartości pozycji w formie **uproszczonej**. Diagram nie pokazuje warunkowych zmian kierunku, obsługi błędów ani pętli. Lokalna pamięć podręczna danych menedżera preprocesingu również nie jest pokazana, ponieważ nie wpływa bezpośrednio na przepływ danych. Celem tego diagramu jest pokazanie procesów biorących udział w przetwarzaniu wartości pozycji oraz sposobu, w jaki ze sobą współdziałają.

- Zbieranie danych rozpoczyna się od surowych danych ze źródła danych. Na tym etapie dane zawierają tylko identyfikator, znacznik czasu i wartość (mogą to być również wiele wartości).
- Niezależnie od tego, jaki typ zbieracza danych jest używany, zasada pozostaje taka sama dla sprawdzeń aktywnych lub pasywnych, pozycji trapper itp., ponieważ zmienia się jedynie format danych i inicjator komunikacji (albo zbieracz danych oczekuje na połączenie i dane, albo zbieracz danych inicjuje komunikację i żąda danych). Surowe dane są weryfikowane, konfiguracja pozycji jest pobierana z pamięci podręcznej konfiguracji (dane są wzbogacane o dane konfiguracyjne).
- Do przekazywania danych od zbieraczy danych do menedżera preprocesingu używany jest mechanizm IPC oparty na gniazdach. Na tym etapie zbieracz danych kontynuuje zbieranie danych bez oczekiwania na odpowiedź od menedżera preprocesingu.
- Wykonywany jest preprocesing danych. Obejmuje to wykonanie kroków preprocesingu oraz przetwarzanie pozycji zależnych.

Note:
Pozycja może zmienić swój stan na NOT SUPPORTED podczas wykonywania preprocesingu, jeśli którykolwiek z kroków preprocesingu zakończy się niepowodzeniem.

- Dane historyczne z lokalnej pamięci podręcznej menedżera preprocesingu są opróżniane do pamięci podręcznej historii.
- Na tym etapie przepływ danych zatrzymuje się do czasu następnej synchronizacji pamięci podręcznej historii (gdy proces history syncer wykonuje synchronizację danych).
- Proces synchronizacji rozpoczyna się od normalizacji danych przed zapisaniem ich w bazie danych Zabbix. Normalizacja danych wykonuje konwersje do docelowego typu pozycji (typu zdefiniowanego w konfiguracji pozycji), w tym obcinanie danych tekstowych na podstawie predefiniowanych rozmiarów dozwolonych dla tych typów (HISTORY_STR_VALUE_LEN dla string, HISTORY_TEXT_VALUE_LEN dla text oraz HISTORY_LOG_VALUE_LEN dla wartości log). Po zakończeniu normalizacji dane są wysyłane do bazy danych Zabbix.

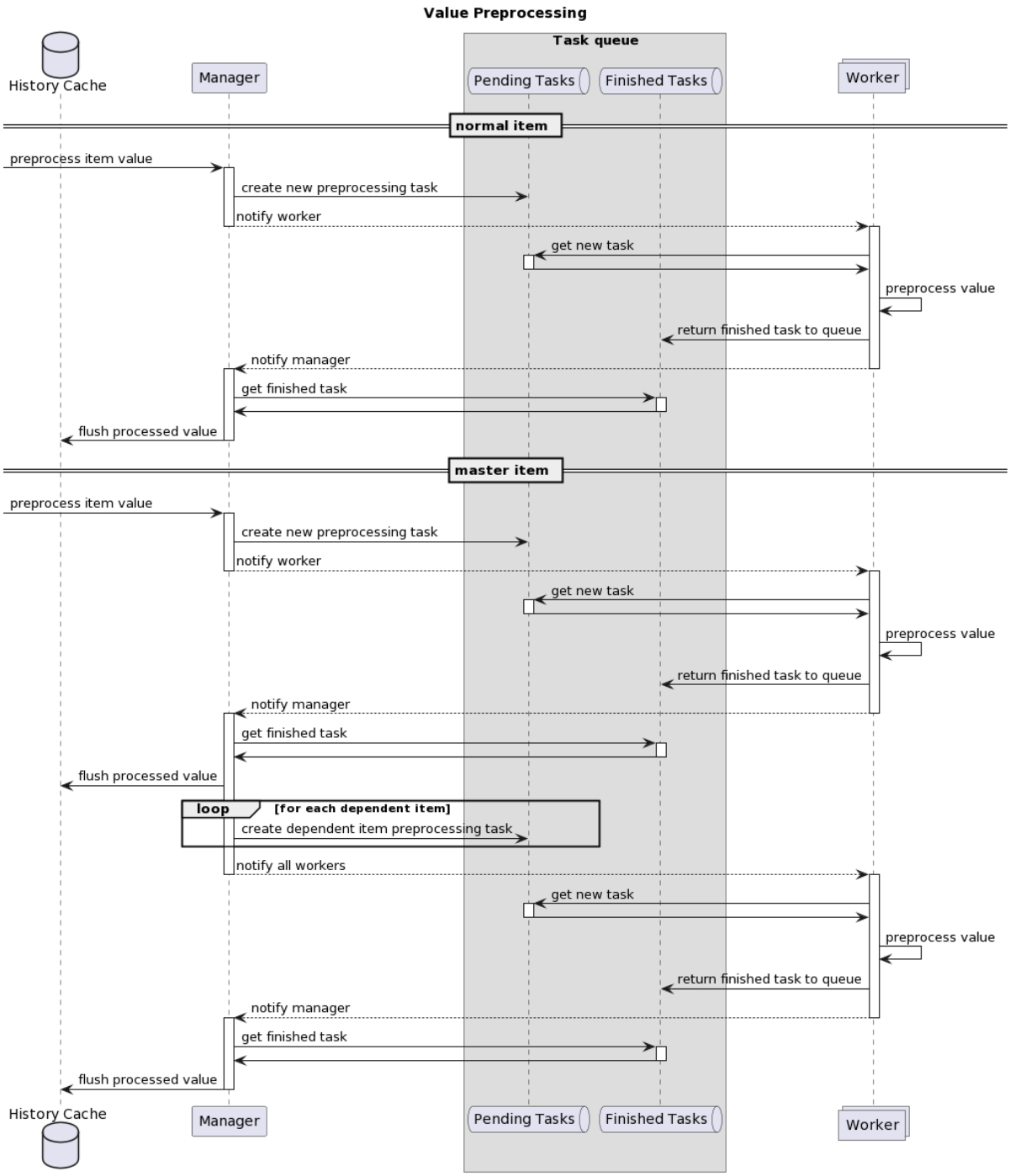
Note:
Pozycja może zmienić swój stan na NOT SUPPORTED, jeśli normalizacja danych zakończy się niepowodzeniem (na przykład, gdy wartość tekstowa nie może zostać przekonwertowana na liczbę).

- Zebrane dane są przetwarzane - sprawdzane są wyzwalacze, konfiguracja pozycji jest aktualizowana, jeśli pozycja stanie się NOT SUPPORTED, itd.
- Jest to uznawane za koniec przepływu danych z punktu widzenia przetwarzania wartości pozycji.

Wstępne przetwarzanie wartości pozycji

Wstępne przetwarzanie danych odbywa się w następujących krokach:

- Jeśli pozycja nie ma ani wstępnego przetwarzania, ani pozycji zależnych, jej wartość jest albo dodawana do pamięci podręcznej historii, albo wysyłana do menedżera LLD. W przeciwnym razie wartość pozycji jest przekazywana do menedżera wstępnego przetwarzania za pomocą mechanizmu IPC opartego na gnieździe UNIX (przed Zabbix 7.4.1 wszystkie wartości były przekazywane przez menedżera wstępnego przetwarzania przed dodaniem ich do pamięci podręcznej historii lub wysłaniem do menedżera LLD).
- Tworzone jest zadanie wstępnego przetwarzania, które zostaje dodane do kolejki, a pracownicy wstępnego przetwarzania są powiadamiani o nowym zadaniu.
- W tym momencie przepływ danych zatrzymuje się, dopóki nie będzie co najmniej jeden wolny (tj. niewykonujący żadnych zadań) pracownik wstępnego przetwarzania.
- Gdy pracownik wstępnego przetwarzania jest dostępny, pobiera następne zadanie z kolejki.
- Po zakończeniu wstępnego przetwarzania (zarówno w przypadku niepowodzenia, jak i powodzenia wykonania kroków wstępnego przetwarzania) wstępnie przetworzona wartość jest dodawana do kolejki ukończonych zadań, a menedżer jest powiadamiany o nowym ukończonym zadaniu.
- Menedżer wstępnego przetwarzania konwertuje wynik do wymaganego formatu (określonego przez typ wartości pozycji) i albo dodaje go do pamięci podręcznej historii, albo wysyła do menedżera LLD.
- Jeśli dla przetwarzanej pozycji istnieją pozycje zależne, są one dodawane do kolejki wstępnego przetwarzania z wstępnie przetworzoną wartością pozycji nadrzędnej. Pozycje zależne są kolejkowane z pominięciem zwykłych żądań wstępnego przetwarzania wartości, ale tylko dla pozycji nadrzędnych z ustawioną wartością i niebędących w stanie NOT SUPPORTED.



Należy zauważyć, że na diagramie wstępne przetwarzanie pozycji nadrzędnej jest nieco uproszczone przez pominięcie buforowania wstępnego przetwarzania.

Kolejka przetwarzania wstępnego

Kolejka przetwarzania wstępnego jest zorganizowana następująco:

- lista oczekujących zadań:
  - zadania utworzone bezpośrednio na podstawie żądań przetwarzania wstępnego wartości, w kolejności ich otrzymania
- lista zadań natychmiastowych (przetwarzanych przed zadaniami oczekującymi):
  - zadania testowe (tworzone w odpowiedzi na żądania testowania pozycja/przetwarzania wstępnego przez frontend)
  - zadania zależnych pozycji
  - zadania sekwencyjne (zadania, które muszą być wykonywane w ścisłej kolejności):
    * posiadające kroki przetwarzania wstępnego wykorzystujące ostatnią wartość:
      · zmiana
      · ograniczanie
      · JavaScript (buforowanie kodu bajtowego)
    * buforowanie przetwarzania wstępnego zależnych pozycji
- lista zakończonych zadań

Buforowanie przetwarzania wstępnego

Buforowanie przetwarzania wstępnego zostało wprowadzone w celu poprawy wydajności przetwarzania wstępnego dla wielu zależnych pozycji mających podobne kroki przetwarzania wstępnego (co jest częstym rezultatem LLD).

Buforowanie polega na przetworzeniu wstępnym jednej zależnej pozycji i ponownym wykorzystaniu części wewnętrznych danych przetwarzania wstępnego dla pozostałych zależnych pozycji. Pamięć podręczna przetwarzania wstępnego jest obsługiwana tylko dla pierwszego kroku przetwarzania wstępnego następujących typów:

- Wzorzec Prometheus (indeksuje dane wejściowe według metryk)
- JSONPath (parsuje dane do drzewa obiektów i indeksuje pierwsze wyrażenie `[?(@.path == "value")]`)

Workery przetwarzania wstępnego

Plik konfiguracyjny serwera Zabbix pozwala użytkownikom ustawić liczbę wątków workerów przetwarzania wstępnego. Parametr konfiguracyjny StartPreprocessors powinien zostać użyty do ustawienia liczby wstępnie uruchamianych instancji workerów przetwarzania wstępnego, która powinna co najmniej odpowiadać liczbie dostępnych rdzeni CPU.

Jeśli zadania przetwarzania wstępnego nie są ograniczone wydajnością CPU i obejmują częste żądania sieciowe, zaleca się skonfigurowanie dodatkowych workerów. Optymalna liczba workerów przetwarzania wstępnego może zależeć od wielu czynników, w tym liczby „przetwarzalnych" pozycji (pozycji, które wymagają wykonania jakichkolwiek kroków przetwarzania wstępnego), liczby procesów zbierania danych, średniej liczby kroków przetwarzania wstępnego dla pozycji itd. Niewystarczająca liczba workerów może prowadzić do wysokiego zużycia pamięci. Informacje o rozwiązywaniu problemów z nadmiernym zużyciem pamięci w instalacji Zabbix można znaleźć w sekcji Profiling excessive memory usage with tcmalloc.

Zakładając jednak, że nie występują ciężkie operacje przetwarzania wstępnego, takie jak analizowanie dużych fragmentów XML/JSON, liczba workerów przetwarzania wstępnego może odpowiadać łącznej liczbie procesów zbierających dane. W ten sposób w większości przypadków (z wyjątkiem sytuacji, gdy dane od procesu zbierającego napływają zbiorczo) będzie dostępny co najmniej jeden nieużywany worker przetwarzania wstępnego dla zebranych danych.

Warning:
Zbyt duża liczba procesów zbierania danych (pollers, unreachable pollers, ODBC pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) wraz z IPMI manager, SNMP trapper i workerami przetwarzania wstępnego może wyczerpać limit deskryptorów plików na proces dla menedżera przetwarzania wstępnego.

 Wyczerpanie limitu deskryptorów plików na proces spowoduje zatrzymanie serwera Zabbix, zwykle krótko po uruchomieniu, choć czasami może to nastąpić później. Aby uniknąć takich problemów, przejrzyj plik konfiguracyjny serwera Zabbix, aby zoptymalizować liczbę równoczesnych kontroli i procesów. Dodatkowo, w razie potrzeby upewnij się, że limit deskryptorów plików jest ustawiony wystarczająco wysoko, sprawdzając i dostosowując limity systemowe.

Potok przetwarzania wartości

Przetwarzanie wartości pozycji jest wykonywane w wielu krokach (lub fazach) przez wiele procesów. Może to powodować:

- Zależna pozycja może otrzymywać wartości, podczas gdy wartość główna NIE może. Można to osiągnąć przy użyciu następującego przypadku użycia:
  - Główna pozycja ma typ wartości `UINT` (można użyć trapper item), zależna pozycja ma typ wartości `TEXT`.
  - Dla głównej i zależnej pozycji nie są wymagane żadne kroki preprocessing.
  - Wartość tekstowa (na przykład „abc") powinna zostać przekazana do głównej pozycji.
  - Ponieważ nie ma kroków preprocessing do wykonania, menedżer preprocessing sprawdza, czy główna pozycja nie jest w stanie NOT SUPPORTED oraz czy wartość jest ustawiona (oba warunki są spełnione), a następnie umieszcza zależną pozycję w kolejce z tą samą wartością co główna pozycja (ponieważ nie ma kroków preprocessing).
  - Gdy zarówno główna, jak i zależna pozycja osiągną fazę synchronizacji historii, główna pozycja przechodzi w stan NOT SUPPORTED z powodu błędu konwersji wartości (dane tekstowe nie mogą zostać przekonwertowane na liczbę całkowitą bez znaku).

W rezultacie zależna pozycja otrzymuje wartość, podczas gdy główna pozycja zmienia swój stan na NOT SUPPORTED.

- Zależna pozycja otrzymuje wartość, która nie jest obecna w historii głównej pozycji. Ten przypadek użycia jest bardzo podobny do poprzedniego, z wyjątkiem typu głównej pozycji. Na przykład, jeśli dla głównej pozycji używany jest typ `CHAR`, to wartość głównej pozycji zostanie obcięta w fazie synchronizacji historii, podczas gdy zależne pozycje otrzymają swoje wartości z początkowej (nieobciętej) wartości głównej pozycji.

3 Przykłady przetwarzania wstępnego

Przegląd

W tej sekcji przedstawiono przykłady użycia kroków przetwarzania wstępnego do realizacji niektórych praktycznych zadań.

Filtrowanie rekordów dziennika zdarzeń VMware

Ten przykład wykorzystuje krok przetwarzania wstępnego Matches regular expression do odfiltrowania niepotrzebnych zdarzeń z dziennika zdarzeń VMware.

1. Na działającym hoście VMware Hypervisor sprawdź, czy pozycja vmware.eventlog jest obecna i działa prawidłowo. Zwróć uwagę, że pozycja dziennika zdarzeń może już być obecna na hypervisorze, jeśli podczas tworzenia hosta został podłączony szablon VMware.

2. Na hoście VMware Hypervisor utwórz pozycję zależną typu *Log* i ustaw pozycję dziennika zdarzeń jako jej pozycję nadrzędną.

3. Na karcie *Preprocessing* pozycji zależnej kliknij *Add*, aby utworzyć krok przetwarzania wstępnego, i wybierz *Matches regular expression* z listy rozwijanej. Następnie określ jeden z następujących wzorców:

- Aby filtrować wszystkie zdarzenia dziennika:

```
.* logged in .*
```

- Aby filtrować wiersze zawierające nazwy użytkowników po "User":

```
\bUser\s+\K\S+
```

Attention:
Jeśli wyrażenie regularne nie zostanie dopasowane, pozycja zależna stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie. Aby tego uniknąć, zaznacz pole wyboru *Custom on fail* i wybierz opcję, taką jak odrzucenie wartości lub ustawienie wartości niestandardowej. Pamiętaj, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są oceniane, a dane trendu nie są generowane.

Alternatywnie możesz użyć kroku przetwarzania wstępnego Regular expression, aby wyodrębnić pasujące grupy i kontrolować dane wyjściowe:

- Aby wyodrębnić i zwrócić całe zdarzenie dziennika zawierające "logged in", podaj następujące parametry:

```
Pattern: .*logged in.*
Output: \0
```

- Aby wyodrębnić i zwrócić nazwy użytkowników występujące po "User":

```
Pattern: User (.*?)(?=\ )
Output: \1
```

Sprawdzanie typu pobranej wartości

W tym przykładzie użyto kroku przetwarzania wstępnego Custom multiplier, aby sprawdzić, czy typ pobranej wartości pozycji jest numeryczny.

Na karcie *Preprocessing* pozycji wybierz krok przetwarzania wstępnego *Custom multiplier* i określ następujący parametr (mnoży pobraną wartość przez 1):

```
1
```

Attention:
Jeśli przetwarzanie wstępne zakończy się niepowodzeniem (np. dane wejściowe nie są numeryczne), pozycja stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie. Aby tego uniknąć, zaznacz pole wyboru *Custom on fail* i wybierz opcję, taką jak odrzucenie wartości lub ustawienie wartości niestandardowej. Należy pamiętać, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są oceniane, a dane trendu nie są generowane.

Sprawdzanie nieobsługiwanej wartości

Ten przykład wykorzystuje krok preprocessing Check for not supported value do sprawdzenia, czy nie udało się pobrać wartości pozycji.

Gdy proces pollera serwera/proxy Zabbix próbuje pobrać wartość pozycji, może:

- Zwrócić prawidłowy wynik.
- Zwrócić wynik, który początkowo wydaje się prawidłowy, ale później może stać się nieobsługiwany (np. z powodu niezgodności typu wartości po preprocessing).
- Zwrócić błąd podczas pobierania wartości, powodując, że pozycja stanie się nieobsługiwana. Typowe przyczyny obejmują:
  - Nieznany klucz pozycji (dla pozycji Zabbix agent, Simple check lub Zabbix internal)
  - Nieznany OID (SNMP agent), nieznany sensor (IPMI agent) lub brak metryki JMX (JMX agent)
  - Nie można odczytać pliku trap (SNMP trap)
  - Nie znaleziono skryptu (External check)

- Taki URL nie istnieje (HTTP agent, Browser)
- Logowanie nie powiodło się (SSH agent, TELNET agent)
- Nieprawidłowa składnia formuły (Calculated), błąd składni JavaScript (Script) lub nieprawidłowe SQL (Database monitor)

Aby wykrywać i obsługiwać błędy pobierania wartości pozycji, można użyć kroku preprocessing *Check for not supported value*. Należy pamiętać, że ten krok jest zawsze wykonywany jako pierwszy i wykrywa tylko błędy, które występują przed rozpoczęciem preprocessing.

Na karcie *Preprocessing* pozycji wybierz krok preprocessing *Check for not supported value* i określ jeden z następujących parametrów:

- Dla dowolnych błędów:

```
Parameter: any error
```

- Dla błędów zawierających "cannot connect":

```
Parameter: error matches
Pattern: (?i)cannot connect
```

Następnie użyj opcji *Custom on fail*, aby odrzucić wartość (w tym przypadku błąd), ustawić wartość niestandardową lub zwrócić niestandardowy komunikat o błędzie. Pamiętaj, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są obliczane, a dane trendu nie są generowane.

4 Funkcjonalność JSONPath

Przegląd

Ta sekcja przedstawia obsługiwaną funkcjonalność JSONPath w krokach wstępnego przetwarzania wartości pozycji.

JSONPath składa się z segmentów oddzielonych kropkami. Segment może mieć postać prostego słowa, reprezentującego nazwę wartości JSON, znaku wieloznacznego (`*`) lub bardziej złożonej konstrukcji ujętej w nawiasy kwadratowe. Kropka przed segmentem ujętym w nawiasy jest opcjonalna i może zostać pominięta.

Przykład JSONPath	Opis
`$.object.name`	Zwraca zawartość `object.name`.
`$.object['name']`	Zwraca zawartość `object.name`.
`$.object.['name']`	Zwraca zawartość `object.name`.
`$["object"]['name']`	Zwraca zawartość `object.name`.
`$.['object'].["name"]`	Zwraca zawartość `object.name`.
`$.object.history.length()`	Zwraca liczbę elementów tablicy `object.history`.
`$[?(@.name == 'Object')].price.first()`	Zwraca wartość właściwości `price` z pierwszego obiektu o nazwie „Object".
`$[?(@.name == 'Object')].history.first().length()`	Zwraca liczbę elementów tablicy historii z pierwszego obiektu o nazwie „Object".
`$[?(@.price > 10)].length()`	Zwraca liczbę obiektów z ceną większą niż 10.

Zobacz także: Escaping special characters from LLD macro values in JSONPath.

Obsługiwane segmenty

Segment	Opis
`<name>`	Dopasowuje właściwość obiektu według nazwy.
`*`	Dopasowuje wszystkie właściwości obiektu.
`['<name>']`	Dopasowuje właściwość obiektu według nazwy.
`['<name>', '<name>', ...]`	Dopasowuje właściwość obiektu według dowolnej z podanych nazw.
`[<index>]`	Dopasowuje element tablicy według indeksu.
`[<number>, <number>, ...]`	Dopasowuje element tablicy według dowolnego z podanych indeksów.
`[*]`	Dopasowuje wszystkie właściwości obiektu lub elementy tablicy.

Segment	Opis
`[<start>:<end>]`	Dopasowuje elementy tablicy według zdefiniowanego zakresu: **<start>** - pierwszy dopasowywany indeks (włącznie); jeśli nie został określony, dopasowywane są wszystkie elementy tablicy od początku; jeśli jest ujemny, określa przesunięcie początkowe liczone od końca tablicy; **<end>** - ostatni dopasowywany indeks (z wyłączeniem); jeśli nie został określony, dopasowywane są wszystkie elementy tablicy do końca; jeśli jest ujemny, określa przesunięcie początkowe liczone od końca tablicy.
`[?(<expression>)]`	Dopasowuje obiekty/elementy tablicy przez zastosowanie wyrażenia filtra.

Aby znaleźć pasujący segment z pominięciem jego pochodzenia (segment odłączony), należy poprzedzić go dwiema kropkami (`..`). Na przykład `$..name` lub `$..['name']` zwracają wartości wszystkich właściwości `name`.

Nazwy dopasowanych elementów można wyodrębnić, dodając do JSONPath sufiks tyldy (`~`). Zwraca on nazwę dopasowanego obiektu lub indeks dopasowanego elementu tablicy w formacie ciągu znaków. Format wyjściowy jest zgodny z tymi samymi zasadami co w przypadku innych zapytań JSONPath — wyniki ścieżki określonej są zwracane „tak jak są", a wyniki ścieżki nieokreślonej są zwracane w tablicy. Jednak wyodrębnianie nazwy elementu dopasowanego przez ścieżkę określoną ma niewielką wartość, ponieważ jest ona już znana.

Wyrażenie filtra

Wyrażenie filtra jest wyrażeniem arytmetycznym w notacji infiksowej.

Obsługiwane operandy:

Operand	Opis

`"<text>"`
`'<text>'`
	Stała tekstowa.
	
	Przykład:
	`'value: \\'1\\''`
	`"value: '1'"`

`<number>`	Stała liczbowa obsługująca notację naukową. Przykład: 123
`<jsonpath starting with $>`	Wartość wskazywana przez JSONPath od węzła głównego dokumentu wejściowego; obsługiwane są tylko ścieżki jednoznaczne. Przykład: `$.object.name`
`<jsonpath starting with @>`	Wartość wskazywana przez JSONPath od bieżącego obiektu/elementu; obsługiwane są tylko ścieżki jednoznaczne. Przykład: `@.name`

Obsługiwane operatory:

Operator	Typ	Opis	Wynik
–	Binarny	Odejmowanie	Liczba
+	Binarny	Dodawanie	Liczba
/	Binarny	Dzielenie	Liczba
*	Binarny	Mnożenie	Liczba
==	Binarny	Równość	Wartość logiczna (1/0)
!=	Binarny	Nierówność	Wartość logiczna (1/0)
	Binarny	Mniejsze niż	Wartość logiczna (1/0)
<=	Binarny	Mniejsze niż lub równe	Wartość logiczna (1/0)
>	Binarny	Większe niż	Wartość logiczna (1/0)
>=	Binarny	Większe niż lub równe	Wartość logiczna (1/0)
=~	Binarny	Pasuje do wyrażenia regularnego	Wartość logiczna (1/0)
!	Unarny	Logiczne NOT	Wartość logiczna (1/0)
||	Binarny	Logiczne OR	Wartość logiczna (1/0)
&&	Binarny	Logiczne AND	Wartość logiczna (1/0)

Funkcje

Funkcje mogą być używane na końcu JSONPath. Można łączyć wiele funkcji w łańcuch, jeśli poprzedzająca funkcja zwraca wartość akceptowaną przez następną funkcję.

Obsługiwane funkcje:

Funkcja	Opis	Wejście	Wyjście
`avg`	Średnia wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`min`	Minimalna wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`max`	Maksymalna wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`sum`	Suma liczb w tablicy wejściowej	Tablica liczb	Liczba
`length`	Liczba elementów w tablicy wejściowej	Tablica	Liczba
`first`	Pierwszy element tablicy	Tablica	Konstrukcja JSON (obiekt, tablica, wartość) zależnie od zawartości tablicy wejściowej

Funkcje agregujące JSONPath akceptują ujęte w cudzysłów wartości liczbowe. Wartości te są automatycznie konwertowane z ciągów znaków na typy liczbowe, gdy wymagana jest agregacja. Niezgodne dane wejściowe spowodują wygenerowanie błędu przez funkcję.

Wartość wyjściowa

Ścieżki JSONPath można podzielić na ścieżki określone i nieokreślone. Ścieżka określona może zwrócić tylko null lub jedno dopasowanie. Ścieżka nieokreślona może zwrócić wiele dopasowań: JSONPath z odłączonymi segmentami, wieloma listami nazw/indeksów, wycinkami tablic lub segmentami wyrażeń. Jednak gdy używana jest funkcja, JSONPath staje się ścieżką określoną, ponieważ funkcje zawsze zwracają pojedynczą wartość.

Ścieżka określona zwraca obiekt/tablicę/wartość, do których się odwołuje. Natomiast ścieżka nieokreślona zwraca tablicę dopasowanych obiektów/tablic/wartości.

> **Attention:**
> Kolejność właściwości w wynikach zapytania JSONPath może nie odpowiadać oryginalnej kolejności właściwości w JSON ze względu na wewnętrzne metody optymalizacji. Na przykład JSONPath `$.books[1]["author", "title"]` może zwrócić `["title", "author"]`. Jeśli zachowanie oryginalnej kolejności właściwości jest istotne, należy rozważyć alternatywne metody przetwarzania po wykonaniu zapytania.

Zasady formatowania ścieżki

Białe znaki (spacja, znak tabulacji) mogą być używane w segmentach notacji nawiasowej i wyrażeniach, na przykład: `$[ 'a' ][ 0 ][ ?( $.b == 'c' ) ][ : -1 ].first( )`.

Ciągi znaków powinny być ujmowane w pojedyncze (') lub podwójne (") cudzysłowy. Wewnątrz ciągów znaków pojedyncze lub podwójne cudzysłowy (w zależności od tego, które są używane do ich ujmowania) oraz ukośniki odwrotne (\) są poprzedzane znakiem ukośnika odwrotnego (\).

Przykład

```
{
  "books": [
    {
      "category": "reference",
      "author": "Nigel Rees",
      "title": "Powiedzenia stulecia",
      "price": 8.95,
      "id": 1
    },
    {
      "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Miecz honoru",
      "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": "Władca Pierścieni",
    "isbn": "0-395-19395-8",
    "price": 22.99,
    "id": 4
  }
],
"services": {
  "delivery": {
    "servicegroup": 1000,
    "description": "Dostawa następnego dnia na terenie lokalnego miasta",
    "active": true,
    "price": 5
  },
  "bookbinding": {
    "servicegroup": 1001,
    "description": "Druk i składanie książki w formacie A5",
    "active": true,
    "price": 154.99
  },
  "restoration": {
    "servicegroup": 1002,
    "description": "Różne metody renowacji",
    "active": false,
    "methods": [
      {
        "description": "Czyszczenie chemiczne",
        "price": 46
      },
      {
        "description": "Prasowanie stron uszkodzonych przez wilgoć",
        "price": 24.5
      },
      {
        "description": "Ponowne oprawienie rozdartej książki",
        "price": 99.49
      }
    ]
  }
},
"filters": {
  "price": 10,
  "category": "fiction",
  "no filters": "brak \"filtrów\""
},
"closed message": "Sklep jest zamknięty",
"tags": [
  "a",
  "b",
  "c",
  "d",
  "e"
]
```

```
}
```

JSONPath	Type	Wynik
`$.filters.price`	definite	10
`$.filters.category`	definite	fiction
`$.filters['no filters']`	definite	brak ”filtrów”
`$.filters`	definite	{ ”price”: 10, ”category”: ”fiction”, ”no filters”: ”brak \”filtrów\”” }
`$.books[1].title`	definite	Miecz honoru
`$.books[-1].author`	definite	J. R. R. Tolkien
`$.books.length()`	definite	4
`$.tags[:]`	indefinite	[”a”, ”b”, ”c”, ”d”, ”e”]
`$.tags[2:]`	indefinite	[”c”, ”d”, ”e”]
`$.tags[:3]`	indefinite	[”a”, ”b”, ”c”]
`$.tags[1:4]`	indefinite	[”b”, ”c”, ”d”]
`$.tags[-2:]`	indefinite	[”d”, ”e”]
`$.tags[:-3]`	indefinite	[”a”, ”b”]
`$.tags[:-3].length()`	definite	2
`$.books[0, 2].title`	indefinite	[”Moby Dick”, ”Powiedzenia stulecia”]
`$.books[1]['author', "title"]`	indefinite	[”Miecz honoru”, ”Evelyn Waugh”]
`$..id`	indefinite	[1, 2, 3, 4]
`$.services..price`	indefinite	[154.99, 5, 46, 24.5, 99.49]
`$.books[?(@.id == 4 - 0.4 * 5)].title`	indefinite	[”Miecz honoru”] Uwaga: To zapytanie pokazuje, że w zapytaniach można używać operacji arytmetycznych; można je uprościć do `$.books[?(@.id == 2)].title`
`$.books[?(@.id == 2 || @.id == 4)].title`	indefinite	[”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(!(@.id == 2))].title`	indefinite	[”Powiedzenia stulecia”, ”Moby Dick”, ”Władca Pierścieni”]
`$.books[?(@.id != 2)].title`	indefinite	[”Powiedzenia stulecia”, ”Moby Dick”, ”Władca Pierścieni”]
`$.books[?(@.title =~ " of ")].title`	indefinite	[”Powiedzenia stulecia”, ”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(@.price > 12.99)].title`	indefinite	[”Władca Pierścieni”]
`$.books[?(@.author > "Herman Melville")].title`	indefinite	[”Powiedzenia stulecia”, ”Władca Pierścieni”]
`$.books[?(@.price > $.filters.price)].title`	indefinite	[”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(@.category == $.filters.category)].title`	indefinite	[”Miecz honoru”,”Moby Dick”,”Władca Pierścieni”]
`$.books[?(@.category == "fiction" && @.price < 10)].title`	indefinite	[”Moby Dick”]

JSONPath	Type	Wynik

`$..[?(@.id)]`	indefinite	[
		{
		”price”: 8.95,
		”id”: 1,
		”category”: ”reference”,
		”author”: ”Nigel Rees”,
		”title”: ”Powiedzenia stulecia”
		},
		{
		”price”: 12.99,
		”id”: 2,
		”category”: ”fiction”,
		”author”: ”Evelyn Waugh”,
		”title”: ”Miecz honoru”
		},
		{
		”price”: 8.99,
		”id”: 3,
		”category”: ”fiction”,
		”author”: ”Herman Melville”,
		”title”: ”Moby Dick”,
		”isbn”: ”0-553-21311-3”
		},
		{
		”price”: 22.99,
		”id”: 4,
		”category”: ”fiction”,
		”author”: ”J. R. R. Tolkien”,
		”title”: ”Władca Pierścieni”,
		”isbn”: ”0-395-19395-8”
		}
		]

`$.services..[?(@.price > 50)].description`	indefinite	[”Druk i składanie książki w formacie A5”, ”Ponowne oprawienie rozdartej książki”]
`$..id.length()`	definite	4
`$.books[?(@.id == 2)].title.first()`	definite	Miecz honoru
`$..tags.first().length()`	definite	5 Uwaga: `$..tags` jest ścieżką indefinite, więc zwraca tablicę dopasowanych elementów, tj. `[["a", "b", "c", "d", "e" ]]`; `first()` zwraca pierwszy element, tj. `["a", "b", "c", "d", "e"]`; `length()` oblicza długość elementu, tj. 5.
`$.books[*].price.min()`	definite	8.95
`$..price.max()`	definite	154.99
`$.books[?(@.category == "fiction")].price.avg()`	definite	14.99
`$.books[?(@.category == $.filters.xyz)].title`	indefinite	Uwaga: Zapytanie bez dopasowania zwraca NULL dla ścieżek definite i indefinite.
`$.services[?(@.active=="true")].servicegroup`	indefinite	[1001,1000] Uwaga: W porównaniach wartości logicznych należy używać stałych tekstowych.
`$.services[?(@.active=="false")].servicegroup`	indefinite	[1002] Uwaga: W porównaniach wartości logicznych należy używać stałych tekstowych.
`$.services[?(@.servicegroup=="1002")].description`	definite	test

1 Escapowanie znaków specjalnych z wartości makr LLD w JSONPath

Gdy makra wykrywania niskiego poziomu są używane we wstępnym przetwarzaniu JSONPath i ich wartości są rozwijane, stosowane są następujące reguły escapowania znaków specjalnych:

- tylko znaki backslash (\) i cudzysłów (") są brane pod uwagę przy escapowaniu;
- jeśli rozwinięta wartość makra zawiera te znaki, każdy z nich jest poprzedzany znakiem backslash;
- jeśli są już poprzedzone znakiem backslash, nie jest to traktowane jako escapowanie i zarówno backslash, jak i następujące po nim znaki specjalne są escapowane ponownie.

Na przykład:

JSONPath	Wartość makra LLD	Po podstawieniu
$.[?(@.value == "{#MACRO}")]	special "value"	$.[?(@.value == "special \"value\"")]
	c:\temp	$.[?(@.value == "c:\\temp")]
	a\\b	$.[?(@.value == "a\\\\b")]

W przypadku użycia w wyrażeniu makro, które może zawierać znaki specjalne, powinno być ujęte w cudzysłowy:

JSONPath	Wartość makra LLD	Po podstawieniu	Wynik
$.[?(@.value == "{#MACRO}")]	special "value"	$.[?(@.value == "special \"value\"")]	OK
$.[?(@.value == {#MACRO})]		$.[?(@.value == special \"value\")]	**Nieprawidłowe wyrażenie JSONPath**

W przypadku użycia w ścieżce makro, które może zawierać znaki specjalne, powinno być ujęte w nawiasy kwadratowe **oraz** cudzysłowy:

JSONPath	Wartość makra LLD	Po podstawieniu	Wynik
$.["{#MACRO}"].value	c:\temp	$.["c:\\temp"].value	OK
$.{#MACRO}.value		$.c:\\temp.value	**Nieprawidłowe wyrażenie JSONPath**

5 Wstępne przetwarzanie JavaScript

Przegląd

Ta sekcja zawiera szczegółowe informacje dotyczące preprocessingu za pomocą JavaScript.

> **Attention:**
> Nie używaj niezadeklarowanych przypisań w JavaScript używanym do preprocessingu. Użyj `var`, aby deklarować zmienne lokalne.

Wstępne przetwarzanie JavaScript

Wstępne przetwarzanie JavaScript jest wykonywane przez wywołanie funkcji JavaScript z pojedynczym parametrem „value" oraz treścią funkcji dostarczoną przez użytkownika. Wynikiem kroku wstępnego przetwarzania jest wartość zwrócona przez tę funkcję; na przykład, aby wykonać konwersję ze stopni Fahrenheita na stopnie Celsjusza, wprowadź:

```
return (value - 32)  * 5 / 9
```

w parametrach wstępnego przetwarzania JavaScript; zostanie to opakowane przez serwer w funkcję JavaScript:

```
function (value)
{
   return (value - 32) * 5 / 9
}
```

Parametr wejściowy „value" jest zawsze przekazywany jako ciąg znaków. Wartość zwracana jest automatycznie konwertowana do ciągu znaków za pomocą metody toString() (jeśli to się nie powiedzie, błąd zostanie zwrócony jako wartość tekstowa), z kilkoma wyjątkami:

- zwrócenie wartości undefined spowoduje błąd;
- zwrócenie wartości null spowoduje odrzucenie wartości wejściowej, podobnie jak wstępne przetwarzanie „Discard value” przy akcji „Custom on fail”.

Błędy mogą być zwracane przez zgłaszanie wartości/obiektów (zwykle są to ciągi znaków albo obiekty Error).

Na przykład:

```
if (value == 0)
    throw "Zero input value"
return 1/value
```

Każdy skrypt ma limit czasu wykonania wynoszący 10 sekund (w zależności od skryptu zadziałanie limitu czasu może potrwać dłużej); jego przekroczenie spowoduje zwrócenie błędu. Obowiązuje limit sterty wynoszący 512 megabajtów.

Kod bajtowy kroku wstępnego przetwarzania JavaScript jest buforowany i używany ponownie przy następnym zastosowaniu tego kroku. Wszelkie zmiany w krokach wstępnego przetwarzania pozycji spowodują zresetowanie skryptu w pamięci podręcznej i jego późniejszą ponowną kompilację.

Kolejne błędy wykonania w czasie działania (3 z rzędu) spowodują ponowną inicjalizację silnika, aby ograniczyć możliwość, że jeden skrypt uszkodzi środowisko wykonywania kolejnych skryptów (to działanie jest rejestrowane przy DebugLevel 4 i wyższym).

Wstępne przetwarzanie JavaScript jest zaimplementowane przy użyciu silnika JavaScript Duktape.

Zobacz także: Dodatkowe obiekty JavaScript i globalne funkcje

Używanie makr w skryptach

Można używać makr użytkownika (a także makr LLD w kontekście wykrywania niskopoziomowego) w kodzie JavaScript. Jeśli skrypt zawiera makra, są one rozwijane przez serwer/proxy przed wykonaniem określonych kroków przetwarzania wstępnego. Należy pamiętać, że podczas testowania kroków przetwarzania wstępnego w frontendzie wartości makr nie zostaną pobrane i trzeba je wprowadzić ręcznie.

> **Note:**
> Kontekst jest ignorowany, gdy makro zostaje zastąpione swoją wartością. Wartość makra jest wstawiana do kodu bez zmian; nie jest możliwe dodanie dodatkowego escapowania przed umieszczeniem wartości w kodzie JavaScript. Należy pamiętać, że w niektórych przypadkach może to powodować błędy JavaScript.

W poniższym przykładzie, jeśli otrzymana wartość przekracza wartość makra {$THRESHOLD}, zostanie zwrócona wartość progu (jeśli jest obecna):

```
var threshold = '{$THRESHOLD}';
return (!isNaN(threshold) && value > threshold) ? threshold : value;
```

Przykłady

Poniższe przykłady ilustrują, jak można używać preprocessingu JavaScript.

Każdy przykład zawiera krótki opis, treść funkcji dla parametrów preprocessingu JavaScript oraz wynik kroku preprocessingu — wartość zwracaną przez funkcję.

Przykład 1: Konwersja liczby (notacja naukowa na liczbę całkowitą)

Przekonwertuj liczbę „2.62128e+07” z notacji naukowej na liczbę całkowitą.

```
return (Number(value))
```

Wartość zwrócona przez funkcję: 26212800.

Przykład 2: Konwersja liczby (z systemu binarnego na dziesiętny)

Przekonwertuj liczbę binarną „11010010” na liczbę dziesiętną.

```
return(parseInt(value,2))
```

Wartość zwrócona przez funkcję: 210.

Przykład 3: Zaokrąglanie liczby

Zaokrąglij liczbę „18.2345” do 2 cyfr.

```
return(Math.round(value* 100) / 100)
```

Wartość zwrócona przez funkcję: 18.23.

Przykład 4: Zliczanie liter w ciągu znaków

Policz liczbę liter w ciągu znaków „Zabbix”.

```
return (value.length)
```

Wartość zwrócona przez funkcję: 6.

Przykład 5: Pobierz pozostały czas

Pobierz pozostały czas (w sekundach) do daty wygaśnięcia certyfikatu (Feb 12 12:33:56 2022 GMT).

```
var split = value.split(' '),
    MONTHS_LIST = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'],
    month_index = ('0' + (MONTHS_LIST.indexOf(split[0]) + 1)).slice(-2),
    ISOdate = split[3] + '-' + month_index + '-' + split[1] + 'T' + split[2],
    now = Date.now();

return parseInt((Date.parse(ISOdate) - now) / 1000);
```

Wartość zwracana przez funkcję: 44380233.

Przykład 6: Usuń właściwości JSON

Zmodyfikuj strukturę danych JSON, usuwając wszystkie właściwości z kluczem "data_size" lub "index_size".

```
var obj=JSON.parse(value);

for (i = 0; i < Object.keys(obj).length; i++) {
    delete obj[i]["data_size"];
    delete obj[i]["index_size"];
}

return JSON.stringify(obj)
```

Wartość przyjmowana przez funkcję:

```
[
    {
        "table_name":"history",
        "data_size":"326.05",
        "index_size":"174.34"
    },
    {
        "table_name":"history_log",
        "data_size":"6.02",
        "index_size":"3.45"
    }
]
```

Wartość zwracana przez funkcję:

```
[
    {
        "table_name":"history"
    },
    {
        "table_name":"history_log"
    }
]
```

Przykład 7: Konwersja statusu Apache do formatu JSON

Przekonwertuj wartość otrzymaną z pozycji Zabbix agent web.page.get (np. web.page.get[http://127.0.0.1:80/server-status?auto]) na obiekt JSON.

```
// Konwersja statusu Apache do formatu JSON

// Podziel wartość na podciągi i umieść te podciągi w tablicy
var lines = value.split('\n');
```

```
// Utwórz pusty obiekt "output"
var output = {};

// Utwórz obiekt "workers" ze wstępnie zdefiniowanymi właściwościami
var workers = {
    '_': 0, 'S': 0, 'R': 0, 'W': 0,
    'K': 0, 'D': 0, 'C': 0, 'L': 0,
    'G': 0, 'I': 0, '.': 0
};

// Dodaj podciągi z tablicy "lines" do obiektu "output" jako właściwości (pary klucz-wartość)
for (var i = 0; i < lines.length; i++) {
    var line = lines[i].match(/([A-z0-9 ]+): (.*)/);

    if (line !== null) {
        output[line[1]] = isNaN(line[2]) ? line[2] : Number(line[2]);
    }
}

// Metryki dla wielu wersji
output.ServerUptimeSeconds = output.ServerUptimeSeconds || output.Uptime;
output.ServerVersion = output.ServerVersion || output.Server;

// Przeanalizuj właściwość "Scoreboard", aby uzyskać liczbę workerów
if (typeof output.Scoreboard === 'string') {
    for (var i = 0; i < output.Scoreboard.length; i++) {
        var char = output.Scoreboard[i];

        workers[char]++;
    }
}

// Dodaj dane workerów do obiektu "output"
output.Workers = {
    waiting: workers['_'], starting: workers['S'], reading: workers['R'],
    sending: workers['W'], keepalive: workers['K'], dnslookup: workers['D'],
    closing: workers['C'], logging: workers['L'], finishing: workers['G'],
    cleanup: workers['I'], slot: workers['.']
};

// Zwróć ciąg JSON
return JSON.stringify(output);
```

Wartość akceptowana przez funkcję:

```
HTTP/1.1 200 OK
Date: Mon, 27 Mar 2023 11:08:39 GMT
Server: Apache/2.4.52 (Ubuntu)
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 405
Content-Type: text/plain; charset=ISO-8859-1

127.0.0.1
ServerVersion: Apache/2.4.52 (Ubuntu)
ServerMPM: prefork
Server Built: 2023-03-08T17:32:01
CurrentTime: Monday, 27-Mar-2023 14:08:39 EEST
RestartTime: Monday, 27-Mar-2023 12:19:59 EEST
ParentServerConfigGeneration: 1
ParentServerMPMGeneration: 0
ServerUptimeSeconds: 6520
ServerUptime: 1 hour 48 minutes 40 seconds
```

```
Load1: 0.56
Load5: 0.33
Load15: 0.28
Total Accesses: 2476
Total kBytes: 8370
Total Duration: 52718
CPUUser: 8.16
CPUSystem: 3.44
CPUChildrenUser: 0
CPUChildrenSystem: 0
CPULoad: .177914
Uptime: 6520
ReqPerSec: .379755
BytesPerSec: 3461.58
BytesPerReq: 3461.58
DurationPerReq: 21.2916
BusyWorkers: 2
IdleWorkers: 6
Scoreboard: ____KW__.....................................................................................
```

Wartość zwracana przez funkcję:


```
{
    "Date": "Mon, 27 Mar 2023 11:08:39 GMT",
    "Server": "Apache/2.4.52 (Ubuntu)",
    "Vary": "Accept-Encoding",
    "Encoding": "gzip",
    "Length": 405,
    "Type": "text/plain; charset=ISO-8859-1",
    "ServerVersion": "Apache/2.4.52 (Ubuntu)",
    "ServerMPM": "prefork",
    "Server Built": "2023-03-08T17:32:01",
    "CurrentTime": "Monday, 27-Mar-2023 14:08:39 EEST",
    "RestartTime": "Monday, 27-Mar-2023 12:19:59 EEST",
    "ParentServerConfigGeneration": 1,
    "ParentServerMPMGeneration": 0,
    "ServerUptimeSeconds": 6520,
    "ServerUptime": "1 hour 48 minutes 40 seconds",
    "Load1": 0.56,
    "Load5": 0.33,
    "Load15": 0.28,
    "Total Accesses": 2476,
    "Total kBytes": 8370,
    "Total Duration": 52718,
    "CPUUser": 8.16,
    "CPUSystem": 3.44,
    "CPUChildrenUser": 0,
    "CPUChildrenSystem": 0,
    "CPULoad": 0.177914,
    "Uptime": 6520,
    "ReqPerSec": 0.379755,
    "BytesPerSec": 1314.55,
    "BytesPerReq": 3461.58,
    "DurationPerReq": 21.2916,
    "BusyWorkers": 2,
    "IdleWorkers": 6,
    "Scoreboard": "____KW__.....................................................................................
    "Workers": {
        "waiting": 6,
        "starting": 0,
        "reading": 0,
        "sending": 1,
        "keepalive": 1,
```

```
        "dnslookup": 0,
        "closing": 0,
        "logging": 0,
        "finishing": 0,
        "cleanup": 0,
        "slot": 142
    }
}
```

1 Dodatkowe obiekty JavaScript

Przegląd

Ta sekcja opisuje dodatki Zabbix do języka JavaScript zaimplementowanego przy użyciu Duktape oraz obsługiwane globalne funkcje JavaScript.

Attention:
Nie używaj niezadeklarowanych przypisań w skryptach JavaScript przetwarzania wstępnego. Używaj `var` do deklarowania zmiennych lokalnych.

Wbudowane obiekty

Zabbix

Obiekt Zabbix umożliwia interakcję z wewnętrzną funkcjonalnością Zabbix.

Method	Description
`log(loglevel, message)`	Zapisuje <message> do logu Zabbix z użyciem poziomu logowania <loglevel> (zobacz parametr DebugLevel w pliku konfiguracyjnym).

Przykład:

```
Zabbix.log(3, "this is a log entry written with 'Warning' log level")
```

Możesz używać następujących aliasów:

Alias	Alias to
console.log(object)	Zabbix.log(4, JSON.stringify(object))
console.warn(object)	Zabbix.log(3, JSON.stringify(object))
console.error(object)	Zabbix.log(2, JSON.stringify(object))

Attention:
Łączny rozmiar wszystkich rejestrowanych komunikatów jest ograniczony do 8 MB na jedno wykonanie skryptu.

Method	Description
`sleep(delay)`	Opóźnia wykonanie JavaScript o `delay` milisekund.

Przykład (opóźnienie wykonania o 15 sekund):

```
Zabbix.sleep(15000)
```

HttpRequest

Ten obiekt hermetyzuje uchwyt cURL, umożliwiając wykonywanie prostych żądań HTTP. Błędy są zgłaszane jako wyjątki.

Attention:
Inicjalizacja wielu obiektów `HttpRequest` jest ograniczona do 10 na jedno wykonanie skryptu.

Method	Description
`addHeader(value)`	Dodaje pole nagłówka HTTP. To pole jest używane dla wszystkich kolejnych żądań, dopóki nie zostanie wyczyszczone metodą `clearHeader()`. Łączna długość pól nagłówka, które można dodać do pojedynczego obiektu `HttpRequest`, jest ograniczona do 128 KB (łącznie ze znakami specjalnymi i nazwami nagłówków).
`clearHeader()`	Czyści nagłówek HTTP. Jeśli nie ustawiono żadnych pól nagłówka, `HttpRequest` ustawi Content-Type na application/json, jeśli wysyłane dane są w formacie JSON; w przeciwnym razie text/plain.
`connect(url)`	Wysyła żądanie HTTP CONNECT do URL i zwraca odpowiedź.
`customRequest(method, url, data)`	Umożliwia określenie dowolnej metody HTTP w pierwszym parametrze. Wysyła żądanie wskazaną metodą do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`delete(url, data)`	Wysyła żądanie HTTP DELETE do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`getHeaders(<asArray>)`	Zwraca obiekt odebranych pól nagłówka HTTP. Parametr `asArray` może być ustawiony na "true" (np. `getHeaders(true)`), "false" lub może być niezdefiniowany. Jeśli jest ustawiony na "true", wartości odebranych pól nagłówka HTTP zostaną zwrócone jako tablice; należy tego używać do pobierania wartości pól wielu nagłówków o tej samej nazwie. Jeśli parametr nie jest ustawiony lub jest ustawiony na "false", wartości odebranych pól nagłówka HTTP zostaną zwrócone jako ciągi znaków.
`get(url, data)`	Wysyła żądanie HTTP GET do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`head(url)`	Wysyła żądanie HTTP HEAD do URL i zwraca odpowiedź.
`options(url)`	Wysyła żądanie HTTP OPTIONS do URL i zwraca odpowiedź.
`patch(url, data)`	Wysyła żądanie HTTP PATCH do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`put(url, data)`	Wysyła żądanie HTTP PUT do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`post(url, data)`	Wysyła żądanie HTTP POST do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`getStatus()`	Zwraca kod statusu ostatniego żądania HTTP.
`setProxy(proxy)`	Ustawia HTTP proxy na wartość "proxy". Jeśli ten parametr jest pusty, proxy nie jest używany.

`setHttpAuth(bitmask, username, password)`	Ustawia włączone metody uwierzytelniania HTTP (HTTPAUTH_BASIC, HTTPAUTH_DIGEST, HTTPAUTH_NEGOTIATE, HTTPAUTH_NTLM, HTTPAUTH_NONE) w parametrze 'bitmask'.
	Flaga HTTPAUTH_NONE umożliwia wyłączenie uwierzytelniania HTTP.
	Przykłady:
	`request.setHttpAuth(HTTPAUTH_NTLM | HTTPAUTH_BASIC, username, password)`
	`request.setHttpAuth(HTTPAUTH_NONE)`

`trace(url, data)`	Wysyła żądanie HTTP TRACE do URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.

Przykład:

```
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://jira.example.com/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

Obiekt XML umożliwia przetwarzanie danych XML we wstępnym przetwarzaniu pozycji i wykrywaniu niskopoziomowym oraz w webhookach.

> **Attention:**
> Aby używać obiektu XML, serwer/proxy musi być skompilowany z obsługą libxml2.

Method	Description
`XML.query(data, expression)`	Pobiera zawartość węzła przy użyciu XPath. Zwraca null, jeśli węzeł nie zostanie znaleziony. **expression** - wyrażenie XPath; **data** - dane XML jako ciąg znaków.
`XML.toJson(data)`	Konwertuje dane w formacie XML do JSON.
`XML.fromJson(object)`	Konwertuje dane w formacie JSON do XML.

Przykład:

Wejście:

```
<menu>
    <food type = "breakfast">
        <name>Chocolate</name>
        <price>$5.95</price>
        <description></description>
        <calories>650</calories>
    </food>
</menu>
```

Wyjście:

```
{
    "menu": {
        "food": {
            "@type": "breakfast",
            "name": "Chocolate",
            "price": "$5.95",
            "description": null,
            "calories": "650"
        }
    }
}
```

Zasady serializacji

Konwersja XML do JSON będzie przetwarzana zgodnie z następującymi zasadami (w przypadku konwersji JSON do XML stosowane są zasady odwrotne):

1. Atrybuty XML zostaną przekonwertowane na klucze, których nazwy są poprzedzone znakiem '@'.

Przykład:

Wejście:

```
<xml foo="FOO">
  <bar>
    <baz>BAZ</baz>
  </bar>
</xml>
```

Wyjście:

```
{
  "xml": {
    "@foo": "FOO",
    "bar": {
      "baz": "BAZ"
    }
  }
}
```

2. Elementy samozamykające się (<foo/>) zostaną przekonwertowane tak, jakby miały wartość 'null'.

Przykład:

Wejście:

```
<xml>
  <foo/>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": null
  }
}
```

3. Puste atrybuty (o wartości "") zostaną przekonwertowane tak, jakby miały wartość pustego ciągu znaków (").

Przykład:

Wejście:

```
<xml>
  <foo bar="" />
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": {
      "@bar": ""
    }
  }
}
```

4. Wiele węzłów podrzędnych o tej samej nazwie elementu zostanie przekonwertowanych na pojedynczy klucz, którego wartością będzie tablica wartości.

Przykład:

Wejście:

```
<xml>
  <foo>BAR</foo>
  <foo>BAZ</foo>
  <foo>QUX</foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": ["BAR", "BAZ", "QUX"]
  }
}
```

5. Jeśli element tekstowy nie ma atrybutów ani elementów podrzędnych, zostanie przekonwertowany jako ciąg znaków.

Przykład:

Wejście:

```
<xml>
    <foo>BAZ</foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": "BAZ"
   }
}
```

6. Jeśli element tekstowy nie ma elementów podrzędnych, ale ma atrybuty, zawartość tekstowa zostanie przekonwertowana na element z kluczem '#text', a zawartość będzie jego wartością; atrybuty zostaną przekonwertowane zgodnie z zasadą serializacji 1.

Przykład:

Wejście:

```
<xml>
  <foo bar="BAR">
    BAZ
  </foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": {
      "@bar": "BAR",
      "#text": "BAZ"
    }
  }
}
```

Globalne funkcje JavaScript

Dodatkowe globalne funkcje JavaScript zostały zaimplementowane przy użyciu Duktape:

- `btoa(data)` - koduje dane do ciągu base64.
- `atob(base64_string)` - dekoduje ciąg base64 jako bufor Uint8Array.

```
try {
    b64 = btoa("test string");
    buffer = atob(b64);

    // Należy pamiętać, że logika dekodowania zależy od formatu danych w buforze.
    decoded = String.fromCharCode.apply(this, [].slice.call(buffer));
}
catch (error) {
    return {'error.name' : error.name, 'error.message' : error.message};
}
```

- `md5(data)` - oblicza skrót MD5 danych.
- `sha256(data)` - oblicza skrót SHA256 danych.

- `hmac('<hash type>',key,data)` - zwraca skrót HMAC jako ciąg w formacie szesnastkowym; jako `hash type` obsługiwane są `md5` i `sha256`; parametry `key` i `data` obsługują dane binarne.

  Przykłady:

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

- `sign(hash,key,data)` - zwraca obliczony podpis (podpis RSA z SHA-256) jako ciąg, gdzie:
 hash - dozwolone jest tylko `sha256`, w przeciwnym razie zostanie zgłoszony błąd.
 key - klucz prywatny. Powinien być zgodny ze standardem PKCS#1 lub PKCS#8. Klucz może być podany w różnych formach:

  - ze spacjami zamiast znaków nowej linii
  - z poprzedzonymi znakiem ucieczki lub niepoprzedzonymi znakiem ucieczki `\n` zamiast znaków nowej linii
  - bez żadnych znaków nowej linii, jako ciąg jednoliniowy
  - jako ciąg w formacie JSON

  Klucz może być również wczytany z makra użytkownika/tajnego makra/sejfu.

  data - dane, które zostaną podpisane. Mogą to być ciąg znaków (obsługiwane są również dane binarne) lub bufor (Uint8Array/ArrayBuffer).

  Przykład:

  - `sign('sha256',key,data)`

  Do obliczania podpisów używany jest OpenSSL lub GnuTLS. Jeśli Zabbix został zbudowany bez którejkolwiek z tych bibliotek szyfrowania, zostanie zgłoszony błąd ('missing OpenSSL or GnuTLS library').

2 Obiekty JavaScript pozycji przeglądarki

Przegląd

Ta sekcja opisuje rozszerzenia języka JavaScript w Zabbix zaimplementowane przy użyciu Duktape do użycia w skrypcie pozycji Browser. Rozszerzenia te uzupełniają obiekty JavaScript opisane na stronie Dodatkowe obiekty JavaScript.

Browser

Obiekt `Browser` zarządza sesjami WebDriver, inicjalizując sesję podczas tworzenia i kończąc ją podczas niszczenia obiektu. Pojedynczy skrypt może obsługiwać maksymalnie cztery obiekty `Browser`.

Aby utworzyć obiekt `Browser`, użyj składni `new Browser(options)`. Parametr `options` (*obiekt JSON*) określa opcje przeglądarki, zwykle wynik metody opcji WebDriver (na przykład `Browser.chromeOptions()`).

Obiekt `Browser` obsługuje następujące metody.

Method	Description
`navigate(url)`	Przejdź do określonego adresu URL. Parametry: `url` - (string) Adres URL, do którego należy przejść.
`getUrl()`	Zwraca ciąg znaków z adresem URL otwartej strony.
`getPageSource()`	Zwraca ciąg znaków z kodem źródłowym otwartej strony.
`findElement(strategy, selector)`	Zwraca obiekt `Element` z jednym elementem na otwartej stronie (lub zwraca `null`, jeśli żadne elementy nie pasują do `strategy` i `selector`). Parametry: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Strategia lokalizacji; `selector` - (string) Selektor elementu używający określonej strategii lokalizacji.
`findElements(strategy, target)`	Zwraca tablicę obiektów `Element` z wieloma elementami na otwartej stronie (lub zwraca pustą tablicę, jeśli żadne elementy nie pasują do strategii lokalizacji i celu). Parametry: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Strategia lokalizacji; `target` - (string) Selektor elementu używający określonej strategii lokalizacji.
`getCookies()`	Zwraca tablicę obiektów `Cookie`.

Method	Description
addCookie(cookie)	Ustawia pliki cookie. Parametry: cookie - (Cookie object) Plik cookie do ustawienia.
getScreenshot()	Zwraca ciąg znaków zakodowany w base64, reprezentujący obraz obszaru widoku przeglądarki.
setScreenSize(x,y)	Ustawia rozmiar obszaru widoku przeglądarki. Parametry: x - (string) Szerokość w pikselach; y - (string) Wysokość w pikselach.
setScriptTimeout(timeout)	Ustawia limit czasu ładowania skryptu. Parametry: timeout - (integer) Wartość limitu czasu w milisekundach.
setSessionTimeout(timeout)	Ustawia limit czasu sesji (ładowania strony). Parametry: timeout - (integer) Wartość limitu czasu w milisekundach.
setElementWaitTimeout(timeout)	Ustawia limit czasu strategii lokalizacji elementu (implicit). Parametry: timeout - (integer) Wartość limitu czasu w milisekundach.
collectPerfEntries(mark)	Zbiera wpisy wydajności do pobrania metodą getResult(). Parametry: mark - (string, optional) Znacznik migawki wydajności.
getRawPerfEntries()	Zwraca tablicę obiektów wpisów wydajności.
getResult()	Zwraca obiekt Result ze statystykami sesji przeglądarki (informacje o błędach, migawki wydajności itp.).
getError()	Zwraca obiekt BrowserError z błędami przeglądarki (lub zwraca null, jeśli nie ma błędów przeglądarki).
setError(message)	Ustawia niestandardowy komunikat błędu, który ma zostać uwzględniony w obiekcie Result. Parametry: message - (string) Komunikat błędu.
discardError()	Odrzuca błąd, który ma zostać zwrócony w obiekcie Result.
getAlert()	Zwraca obiekt Alert z alertami przeglądarki (lub zwraca null, jeśli nie ma alertów przeglądarki).
chromeOptions()	Zwraca obiekt chromeOptions ze wstępnie zdefiniowanymi opcjami przeglądarki Chrome.
firefoxOptions()	Zwraca obiekt firefoxOptions ze wstępnie zdefiniowanymi opcjami przeglądarki Firefox.
safariOptions()	Zwraca obiekt safariOptions ze wstępnie zdefiniowanymi opcjami przeglądarki Safari.
edgeOptions()	Zwraca obiekt edgeOptions ze wstępnie zdefiniowanymi opcjami przeglądarki Edge.
switchFrame(target)	Przełącza do określonej ramki. Parametry: target - (browser element or integer, optional) Ramka docelowa. Aby wybrać ramkę według elementu, przekaż element. Aby wybrać ramkę według indeksu, przekaż liczbę. Jeśli pozostawiono puste, nastąpi przełączenie do kontekstu przeglądania najwyższego poziomu.

Wszystkie metody Browser mogą zgłaszać następujące błędy:

- BrowserError - dziedziczony z obiektu Error, zgłaszany, jeśli konstruktor Browser zakończy się niepowodzeniem; zawiera dodatkową właściwość browser z obiektem Browser, który zgłosił ten BrowserError.
- WebdriverError - dziedziczony z BrowserError; zawiera te same właściwości co obiekt BrowserError, które wskazują, czy błąd został wygenerowany w odpowiedzi na błąd w odpowiedzi WebDriver.

Element

Obiekt `Element` jest zwracany przez metody `findElement()`/`findElements()` obiektu `Browser` i nie może być tworzony bezpośrednio.

Obiekt `Element` reprezentuje element na stronie internetowej i udostępnia metody do interakcji z nim.

Poniższe metody są obsługiwane przez obiekt `Element`.

Method	Description
`getAttribute(name)`	Zwraca ciąg wartości atrybutu elementu (lub zwraca `null`, jeśli określony atrybut nie został znaleziony). Parametry: `name` - (string) Nazwa atrybutu.
`getProperty(name)`	Zwraca ciąg wartości właściwości elementu (lub zwraca `null`, jeśli określona właściwość nie została znaleziona). Parametry: `name` - (string) Nazwa właściwości.
`getText()`	Zwraca ciąg wartości tekstowej tekstu elementu.
`click()`	Kliknij element.
`clear()`	Wyczyść zawartość edytowalnego elementu.
`sendKeys(keys)`	Wysyła klawisze. Parametry: `keys` - (string) Klawisze do wysłania.

Cookie

Obiekt `Cookie` jest zwracany przez metodę `getCookies()` obiektu `Browser` i przekazywany do metody `addCookie()`.

Chociaż obiekt `Cookie` nie ma żadnych metod, może zawierać następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa pliku cookie.
value	string	Wartość pliku cookie.
path	string	Ścieżka, dla której plik cookie jest prawidłowy. Domyślnie `"/"`, jeśli zostanie pominięta podczas dodawania pliku cookie.
domain	string	Domena, dla której plik cookie jest widoczny. Domyślnie domena URL aktywnego dokumentu bieżącego kontekstu przeglądania sesji, jeśli zostanie pominięta podczas dodawania pliku cookie.
secure	boolean	Wskazuje, czy plik cookie jest bezpieczny. Domyślnie `false`, jeśli zostanie pominięte podczas dodawania pliku cookie.
httpOnly	boolean	Wskazuje, czy plik cookie jest dostępny tylko przez HTTP. Domyślnie `false`, jeśli zostanie pominięte podczas dodawania pliku cookie.
expiry	integer	Czas wygaśnięcia pliku cookie (w sekundach od epoki Unix). Nie może być ustawiony, jeśli zostanie pominięty podczas dodawania pliku cookie.
sameSite	string	Atrybut `sameSite` pliku cookie, który określa, czy plik cookie powinien być ograniczony do kontekstu first-party lub same-site. Może być ustawiony na `"Lax"` lub `"Strict"`. Domyślnie `"None"`, jeśli zostanie pominięty podczas dodawania pliku cookie.

Alert

Obiekt `Alert` reprezentuje alert strony internetowej, jest zwracany przez metodę `getAlert()` obiektu `Browser` i nie może być tworzony bezpośrednio.

Obiekt `Alert` zawiera właściwość `text` z tekstem alertu (lub `null`, jeśli nie ma alertów).

Następujące metody są obsługiwane przez obiekt `Alert`.

Method	Description
`accept()`	Akceptuje alert.
`dismiss()`	Odrzuca alert.

Wynik

Obiekt `Result` zawiera statystyki sesji i jest zwracany przez metodę `getResult()` obiektu `Browser`.

Zazwyczaj obiekt `Result` jest konwertowany do postaci ciągu znaków i zwracany ze skryptu, a następnie przetwarzany na wartości zależnych pozycji za pomocą preprocessing.

Chociaż obiekt `Result` nie ma żadnych metod, może zawierać następujące właściwości.

Właściwość	Typ	Opis
duration	string	Czas trwania sesji od utworzenia sesji do pobrania wyniku.
error	object	Informacje o błędzie.
http_status	integer	Status HTTP zwrócony przez WebDriver (lub 0, jeśli nie ma błędów WebDriver).
error_code	string	Błąd zwrócony przez WebDriver (lub pusty ciąg znaków, jeśli nie ma błędów WebDriver).
message	string	Komunikat błędu WebDriver (lub pusty ciąg znaków, jeśli nie ma błędów WebDriver).
performance_data	object	Statystyki wydajności.
summary	object	Podsumowanie wydajności.
navigation	object	Podsumowanie nawigacji.
resource	object	Podsumowanie zasobów.
details	array of objects	Statystyki wydajności po każdej operacji, która mogła skutkować nawigacją.
mark	string	*(opcjonalnie)* Oznaczenie migawki wydajności określone za pomocą metody `collectPerfEntries()`.
navigation	object	Statystyki nawigacji.
resource	object	Podsumowanie zasobów dla tego kroku.
user	array of objects	Tablica statystyk typu mark/measure.
marks	array of objects	Indeksy oznaczonych migawek wydajności.
name	string	Nazwa oznaczenia migawki wydajności.
index	integer	Indeks migawki wydajności w tablicy details.

6 Przetwarzanie wstępne CSV do JSON

Przegląd

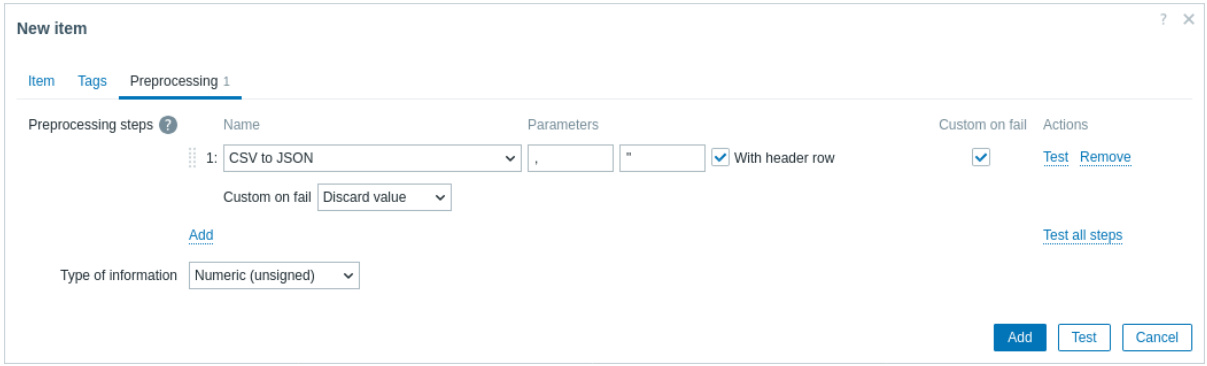
W tym kroku przetwarzania wstępnego można przekonwertować dane pliku CSV do formatu JSON. Jest to obsługiwane w:

- pozycjach (prototypach pozycji)
- regułach wykrywania niskiego poziomu

Konfiguracja

Aby skonfigurować krok przetwarzania wstępnego CSV do JSON:

- Przejdź do zakładki Przetwarzanie wstępne w konfiguracji pozycji/reguły wykrywania
- Kliknij *Dodaj*
- Wybierz opcję *CSV do JSON*



Pierwszy parametr umożliwia ustawienie niestandardowego separatora. Należy pamiętać, że jeśli pierwszy wiersz danych wejściowych CSV zaczyna się od „Sep=" i po nim występuje pojedynczy znak UTF-8, wówczas ten znak zostanie użyty jako separator, jeśli pierwszy parametr nie jest ustawiony. Jeśli pierwszy parametr nie jest ustawiony i separator nie zostanie odczytany z wiersza „Sep=", jako separator zostanie użyty przecinek.

Drugi opcjonalny parametr umożliwia ustawienie znaku cudzysłowu.

Jeśli pole wyboru *Z wierszem nagłówka* jest zaznaczone, wartości wiersza nagłówka będą interpretowane jako nazwy kolumn (więcej informacji znajduje się w sekcji Przetwarzanie nagłówka).

Jeśli pole wyboru *Niestandardowo w przypadku błędu* jest zaznaczone, pozycja nie stanie się nieobsługiwana w przypadku niepowodzenia kroku przetwarzania wstępnego. Dodatkowo można ustawić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.

Header processing

The CSV file header line can be processed in two different ways:

- If the *With header row* checkbox is marked - header line values are interpreted as column names. In this case the column names must be unique and the data row should not contain more columns than the header row;
- If the *With header row* checkbox is not marked - the header line is interpreted as data. Column names are generated automatically (1,2,3,4...)

CSV file example:

```
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:
A quotation character within a quoted field in the input must be escaped by preceding it with another quotation character.

Processing header line

JSON output when a header line is expected:

```
[
   {
      "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"
```

```
    }
]
```

No header line processing

JSON output when a header line is not expected:

```
[
    {
        "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"
    }
]
```

3 Typy pozycji

Przegląd

Typy pozycji obejmują różne metody pozyskiwania danych z systemu. Każdy typ pozycji ma własny zestaw obsługiwanych kluczy pozycji oraz wymaganych parametrów.

Obecnie Zabbix oferuje następujące typy pozycji:

- agent Zabbix
  - Monitorowanie plików dziennika
- Prosty check
  - Monitorowanie VMware
- agent SNMP
- SNMP trap
- Wewnętrzny Zabbix
- Zabbix trapper
- Zewnętrzny check
- Monitor bazy danych
- agent HTTP
  - Check Prometheus
- agent IPMI
- agent SSH
- agent Telnet
- agent JMX
- Obliczana pozycja
  - Obliczenia agregujące
- Pozycja zależna
- Pozycja skryptowa
- Pozycja przeglądarki

Szczegóły dotyczące wszystkich typów pozycji znajdują się na podstronach tej sekcji. Mimo że typy pozycji oferują wiele opcji zbierania danych, istnieją dodatkowe możliwości za pomocą parametrów użytkownika lub ładowalnych modułów.

Niektóre kontrole są wykonywane wyłącznie przez serwer Zabbix (jako monitorowanie bez agenta), podczas gdy inne wymagają agenta Zabbix, a nawet Zabbix Java gateway (w przypadku monitorowania JMX).

Attention:
Jeśli dany typ pozycji wymaga określonego interfejsu (na przykład check IPMI wymaga interfejsu IPMI na hoście), taki interfejs musi istnieć w definicji hosta.

W definicji hosta można ustawić wiele interfejsów: agent Zabbix, agent SNMP, JMX i IPMI. Jeśli pozycja może używać więcej niż jednego interfejsu, będzie wyszukiwać dostępne interfejsy hosta (w kolejności: Agent→SNMP→JMX→IPMI) w celu wybrania pierwszego odpowiedniego do powiązania.

Wszystkie pozycje, które zwracają tekst (informacje typu znakowego, dziennika, tekstu), mogą również zwracać wyłącznie białe znaki (tam, gdzie ma to zastosowanie), ustawiając wartość zwracaną na pusty ciąg znaków (obsługiwane od wersji 2.0).

1 Agent Zabbix

Przegląd

Ta sekcja zawiera szczegółowe informacje o kluczach pozycji, które wykorzystują komunikację z agent do zbierania danych.

Istnieją pasywne i aktywne kontrole agent. Podczas konfigurowania pozycji można wybrać wymagany typ:

- *Zabbix agent* — dla kontroli pasywnych
- *Zabbix agent (active)* — dla kontroli aktywnych

Należy pamiętać, że wszystkie klucze pozycji obsługiwane przez Zabbix agent są również obsługiwane przez nowej generacji Zabbix agent 2. Zobacz dodatkowe klucze pozycji, których można używać wyłącznie z agent 2.

Obsługiwane klucze pozycji

Klucze pozycji, których można używać z agentem Zabbix, są wymienione poniżej.

Klucze pozycji są podane bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Item key	Description	Item group
kernel.maxfiles	Maksymalna liczba otwartych plików obsługiwana przez system operacyjny.	Kernel
kernel.maxproc	Maksymalna liczba procesów obsługiwana przez system operacyjny.	
kernel.openfiles	Liczba aktualnie otwartych deskryptorów plików.	
log	Monitorowanie pliku dziennika.	Log monitoring
log.count	Liczba dopasowanych wierszy w monitorowanym pliku dziennika.	
logrt	Monitorowanie rotowanego pliku dziennika.	
logrt.count	Liczba dopasowanych wierszy w monitorowanym rotowanym pliku dziennika.	
modbus.get	Odczytuje dane Modbus.	Modbus
net.dns	Sprawdza stan usługi DNS.	Network
net.dns.perf	Sprawdza wydajność usługi DNS.	
net.dns.record	Wykonuje zapytanie DNS.	
net.if.collisions	Liczba kolizji poza oknem czasowym.	
net.if.discovery	Lista interfejsów sieciowych.	
net.if.in	Statystyki ruchu przychodzącego na interfejsie sieciowym.	
net.if.out	Statystyki ruchu wychodzącego na interfejsie sieciowym.	
net.if.total	Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym.	
net.tcp.listen	Sprawdza, czy ten port TCP ma stan LISTEN.	
net.tcp.port	Sprawdza, czy można nawiązać połączenie TCP z określonym portem.	
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP.	
net.tcp.service.perf	Sprawdza wydajność usługi TCP.	
net.tcp.socket.count	Zwraca liczbę gniazd TCP pasujących do parametrów.	
net.udp.listen	Sprawdza, czy ten port UDP ma stan LISTEN.	
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.	
net.udp.service.perf	Sprawdza wydajność usługi UDP.	
net.udp.socket.count	Zwraca liczbę gniazd UDP pasujących do parametrów.	
proc.cpu.util	Procent wykorzystania CPU przez proces.	Processes
proc.get	Lista procesów systemu operacyjnego i ich parametrów.	

Item key	Description	Item group
proc.mem	Pamięć używana przez proces w bajtach.	
proc.num	Liczba procesów.	
sensor	Odczyt czujnika sprzętowego.	Sensors
system.boottime	Czas uruchomienia systemu.	System
system.cpu.discovery	Lista wykrytych procesorów/rdzeni CPU.	
system.cpu.intr	Przerwania urządzeń.	
system.cpu.load	Obciążenie CPU.	
system.cpu.num	Liczba procesorów CPU.	
system.cpu.switches	Liczba przełączeń kontekstu.	
system.cpu.util	Procent wykorzystania CPU.	
system.hostname	Nazwa hosta systemu.	
system.hw.chassis	Informacje o obudowie.	
system.hw.cpu	Informacje o CPU.	
system.hw.devices	Lista urządzeń PCI lub USB.	
system.hw.macaddr	Lista adresów MAC.	
system.localtime	Czas systemowy.	
system.run	Uruchamia określone polecenie na hoście.	
system.stat	Statystyki systemowe.	
system.sw.arch	Informacje o architekturze oprogramowania.	
system.sw.os	Informacje o systemie operacyjnym.	
system.sw.os.get	Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja minor i major itp).	
system.sw.packages	Lista zainstalowanych pakietów.	
system.sw.packages.get	Szczegółowa lista zainstalowanych pakietów.	
system.swap.in	Statystyki swap-in (z urządzenia do pamięci).	
system.swap.out	Statystyki swap-out (z pamięci na urządzenie).	
system.swap.size	Rozmiar przestrzeni swap w bajtach lub jako procent całości.	
system.uname	Identyfikacja systemu.	
system.uptime	Czas działania systemu w sekundach.	
system.users.num	Liczba zalogowanych użytkowników.	
vfs.dev.discovery	Lista urządzeń blokowych i ich typów.	Virtual file systems
vfs.dev.read	Statystyki odczytu dysku.	
vfs.dev.write	Statystyki zapisu dysku.	
vfs.dir.count	Liczba wpisów katalogu.	
vfs.dir.get	Lista wpisów katalogu.	
vfs.dir.size	Rozmiar katalogu.	
vfs.file.cksum	Suma kontrolna pliku, obliczana algorytmem UNIX cksum.	
vfs.file.contents	Pobieranie zawartości pliku.	
vfs.file.exists	Sprawdza, czy plik istnieje.	
vfs.file.get	Zwraca informacje o pliku.	
vfs.file.md5sum	Suma kontrolna MD5 pliku.	
vfs.file.owner	Pobiera właściciela pliku.	
vfs.file.permissions	Zwraca 4-cyfrowy ciąg zawierający liczbę ósemkową z uprawnieniami UNIX.	
vfs.file.regexp	Pobiera ciąg znaków z pliku.	
vfs.file.regmatch	Znajduje ciąg znaków w pliku.	
vfs.file.size	Rozmiar pliku.	
vfs.file.time	Informacje o czasie pliku.	
vfs.fs.discovery	Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania.	
vfs.fs.get	Lista zamontowanych systemów plików wraz z ich typem, dostępnym miejscem na dysku, statystykami inode i opcjami montowania.	
vfs.fs.inode	Liczba lub procent inode.	
vfs.fs.size	Miejsce na dysku w bajtach lub jako procent całości.	
vm.memory.size	Rozmiar pamięci w bajtach lub jako procent całości.	Virtual memory
web.page.get	Pobiera zawartość strony internetowej.	Web monitoring
web.page.perf	Czas ładowania pełnej strony internetowej.	
web.page.regexp	Znajduje ciąg znaków na stronie internetowej.	
agent.hostmetadata	Metadane hosta agenta.	Zabbix
agent.hostname	Nazwa hosta agenta.	

Item key	Description	Item group
agent.ping	Sprawdzenie dostępności agenta.	
agent.variant	Wariant agenta Zabbix (Zabbix agent lub Zabbix agent 2).	
agent.version	Wersja agenta Zabbix.	
zabbix.stats	Zwraca zestaw wewnętrznych metryk serwera Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.	
zabbix.stats (queue)	Zwraca liczbę monitorowanych pozycji w kolejce, które są opóźnione na serwerze Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.	

Obsługiwane platformy

O ile w szczegółach pozycji nie określono inaczej, pozycje agenta (oraz wszystkie parametry) są obsługiwane na:

- **Linux**
- **FreeBSD**
- **Solaris**
- **HP-UX**
- **AIX**
- **MacOS X**
- **OpenBSD**
- **NetBSD**

Wiele pozycji agenta jest również obsługiwanych w systemie **Windows**. Szczegóły można znaleźć na stronie Windows agent item.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

kernel.maxfiles

 Maksymalna liczba otwartych plików obsługiwana przez system operacyjny.
 Zwracana wartość: *Liczba całkowita*.
 Obsługiwane platformy: Linux, FreeBSD, MacOS X, OpenBSD, NetBSD.

kernel.maxproc

 Maksymalna liczba procesów obsługiwana przez system operacyjny.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux 2.6 i nowsze, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

kernel.openfiles

 Liczba aktualnie otwartych deskryptorów plików.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux (pozycja może działać na innych platformach typu UNIX).

log[file,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 Monitorowanie pliku dziennika.
 Zwracana wartość: *Log*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka i nazwa pliku dziennika;
- **regexp** - regularne wyrażenie opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej;
- **maxlines** - maksymalna liczba nowych linii na sekundę, które agent wyśle do serwer lub proxy. Ten parametr zastępuje wartość 'MaxLinesPerSecond' w zabbix_agentd.conf;
- **mode** - możliwe wartości: *all* (domyślnie) lub *skip* - pomiń przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji);
- **output** - opcjonalny szablon formatowania danych wyjściowych. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, od którego zaczyna się dopasowanie, do znaku, na którym dopasowanie się kończy), natomiast sekwencja ucieczki **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup);
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości: 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika; > 0.0 - ignoruj starsze linie, aby analizować najnowsze linie w ciągu "maxdelay" sekund. Przed użyciem przeczytaj uwagi dotyczące maxdelay!
- **options** - dodatkowe opcje:
mtime-noreread - nieunikalne rekordy, odczytuj ponownie tylko wtedy, gdy zmienia się rozmiar pliku (ignoruj zmianę czasu modyfikacji). (Ten parametr jest przestarzały od wersji 5.0.2, ponieważ obecnie czas modyfikacji jest ignorowany.)

- **persistent dir** (tylko w zabbix_agentd w systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka do katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące plików trwałych.

Komentarze:

- Pozycja musi być skonfigurowana jako sprawdzenie aktywne.
- Jeśli plik nie istnieje lub uprawnienia nie pozwalają na dostęp, pozycja staje się nieobsługiwana.
- Jeśli `output` pozostanie puste - zwracana jest cała linia zawierająca dopasowany tekst. Zwróć uwagę, że wszystkie typy globalnych wyrażeń regularnych z wyjątkiem 'Result is TRUE' zawsze zwracają całą dopasowaną linię, a parametr `output` jest ignorowany.
- Ekstrakcja zawartości przy użyciu parametru `output` odbywa się po stronie agenta.

Przykłady:

```
log[/var/log/syslog]
log[/var/log/syslog,error]
log[/home/zabbix/logs/logfile,,,100]
```

Przykład użycia parametru `output` do wyodrębnienia liczby z rekordu dziennika:

```
log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #ta pozycja dop
```

Przykład użycia parametru `output` do przepisania rekordu dziennika przed wysłaniem do serwer:

```
log[/app1/app.log,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",,,,"\1
```

log.count[file,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent dir>]

 Liczba dopasowanych linii w monitorowanym pliku dziennika.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** — pełna ścieżka i nazwa pliku dziennika;
- **regexp** — wyrażenie regularne opisujące wymagany wzorzec;
- **encoding** — identyfikator strony kodowej;
- **maxproclines** — maksymalna liczba nowych linii na sekundę, które agent przeanalizuje (nie może przekraczać 10000). Wartość domyślna to 10*'MaxLinesPerSecond' w zabbix_agentd.conf.
- **mode** — możliwe wartości: *all* (domyślnie) lub *skip* — pomiń przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).
- **maxdelay** — maksymalne opóźnienie w sekundach. Typ: float. Wartości: 0 — (domyślnie) nigdy nie ignoruj linii pliku dziennika; > 0.0 — ignoruj starsze linie, aby analizować najnowsze linie w ciągu „maxdelay" sekund. Przed użyciem przeczytaj uwagi dotyczące maxdelay!
- **options** — dodatkowe opcje:
mtime-noreread — nieunikalne rekordy, odczytuj ponownie tylko wtedy, gdy zmieni się rozmiar pliku (ignoruj zmianę czasu modyfikacji). (Ten parametr jest przestarzały od wersji 5.0.2, ponieważ obecnie mtime jest ignorowany.)
- **persistent dir** (tylko w zabbix_agentd w systemach Unix; nieobsługiwane w Zabbix agent 2) — bezwzględna ścieżka katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące plików trwałych.

Komentarze:

- Pozycja musi być skonfigurowana jako sprawdzenie aktywne.
- Dopasowane linie są zliczane wśród nowych linii od ostatniego sprawdzenia dziennika przez agent, a zatem zależą od interwału aktualizacji pozycji.
- Jeśli plik nie istnieje lub uprawnienia nie pozwalają na dostęp, pozycja staje się nieobsługiwana.

logrt[file regexp,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 Monitorowanie pliku dziennika, który jest rotowany.
 Wartość zwracana: *Log*.
 Zobacz obsługiwane platformy.

Parametry:

- **file regexp** - bezwzględna ścieżka do pliku, z nazwą pliku określoną za pomocą wyrażenia regularnego expression. Należy pamiętać, że wyrażenie regularne ma zastosowanie tylko do nazwy pliku i nie musi dopasowywać całej nazwy (np. /path/to/agent dopasuje zabbix_agentd.log).
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec zawartości.
- **encoding** - identyfikator strony kodowej identifier.
- **maxlines** - maksymalna liczba nowych linii na sekundę, którą agent wyśle do serwer lub proxy. Ten parametr zastępuje wartość 'MaxLinesPerSecond' w zabbix_agentd.conf.
- **mode** - możliwe wartości: *all* (domyślnie) lub *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

- **output** - opcjonalny szablon formatowania wyjścia. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, w którym zaczyna się dopasowanie, do znaku, w którym się kończy), natomiast sekwencja **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup).
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości: 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika; > 0.0 - ignoruj starsze linie, aby najnowsze analizowane linie mieściły się w czasie "maxdelay" sekund. Przed użyciem przeczytaj uwagi dotyczące maxdelay!
- **options** - typ rotacji pliku dziennika i inne opcje. Możliwe wartości:
rotate (domyślnie),
copytruncate - należy pamiętać, że *copytruncate* nie może być używane razem z *maxdelay*. W takim przypadku *maxdelay* musi mieć wartość 0 lub nie może być określone; zobacz uwagi dotyczące copytruncate,
mtime-reread - rekordy nieunikalne, odczyt ponownie, jeśli zmieni się czas modyfikacji lub rozmiar (używane domyślnie, jeśli żadna opcja `mtime-*` nie została jawnie ustawiona),
mtime-noreread - rekordy nieunikalne, odczyt ponownie tylko jeśli zmieni się rozmiar (ignoruj zmianę czasu modyfikacji).
- **persistent dir** (tylko w zabbix_agentd w systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące persistent files.

Uwagi:

- Pozycja musi być skonfigurowana jako active check.
- Rotacja pliku dziennika opiera się na czasie ostatniej modyfikacji plików.
- Należy pamiętać, że logrt jest przeznaczony do pracy z jednym aktualnie aktywnym plikiem dziennika oraz kilkoma innymi pasywnymi plikami spełniającymi warunek, które zostały zrotowane. Jeśli na przykład katalog zawiera wiele aktywnych plików dziennika, należy utworzyć osobną pozycję logrt dla każdego z nich. W przeciwnym razie, jeśli jedna pozycja logrt obejmie zbyt wiele plików, może to doprowadzić do wyczerpania pamięci i awarii monitorowania.
- Jeśli `output` pozostanie puste - zwracana jest cała linia zawierająca dopasowany tekst. Należy pamiętać, że wszystkie globalne typy wyrażeń regularnych z wyjątkiem 'Result is TRUE' zawsze zwracają całą dopasowaną linię, a parametr `output` jest ignorowany.
- Wyodrębnianie zawartości przy użyciu parametru `output` odbywa się na agencie.
- W parametrze `file regexp` ścieżka katalogu dziennika i wyrażenie regularne pliku dziennika muszą być oddzielone poprawnym separatorem katalogów:
  - w systemie Windows separatorem musi być ukośnik odwrotny (\); ukośniki ukośne (/) mogą być tolerowane w innych miejscach, z wyjątkiem tego, który oddziela ścieżkę katalogu dziennika i wyrażenie regularne pliku dziennika (zobacz przykłady poniżej).
  - w innych systemach musi to być ukośnik ukośny (/).

Przykłady dla Windows:

```
logrt["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja będzie zbierać dane z plików dzi
logrt["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja będzie zbierać
```

Przykłady dla innych systemów:

```
logrt["/home/zabbix/logs/^logfile[0-9]{1,3}$",,,100] #ta pozycja dopasuje plik taki jak "logfile1" (nie do
logrt["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #ta pozycja będzie zbierać dane
```

Przykład użycia parametru `output` do wyodrębnienia liczby z rekordu dziennika:

```
logrt[/app1/^test.*log$,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #ta pozyc
```

Przykład użycia parametru `output` do przekształcenia rekordu dziennika przed wysłaniem do serwer:

```
logrt[/app1/^test.*log$,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",
```

logrt.count[file regexp,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent dir>]

 Liczba dopasowanych wierszy w monitorowanym pliku dziennika, który jest rotowany.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file regexp** - bezwzględna ścieżka do pliku, z nazwą pliku określoną za pomocą wyrażenia regularnego expression. Należy pamiętać, że wyrażenie regularne ma zastosowanie tylko do nazwy pliku i nie musi dopasowywać całej nazwy (np. /path/to/agent dopasuje zabbix_agentd.log).
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec.
- **encoding** - identyfikator strony kodowej identifier.
- **maxproclines** - maksymalna liczba nowych wierszy na sekundę, które agent będzie analizował (nie może przekroczyć 10000). Wartość domyślna to 10*'MaxLinesPerSecond' w zabbix_agentd.conf.
- **mode** - możliwe wartości: *all* (domyślnie) lub *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości: 0 - (domyślnie) nigdy nie ignoruj wierszy pliku dziennika; > 0.0 - ignoruj starsze wiersze, aby najnowsze wiersze zostały przeanalizowane w ciągu "maxdelay" sekund. Przed użyciem przeczytaj uwagi dotyczące maxdelay!
- **options** - typ rotacji pliku dziennika i inne opcje. Możliwe wartości:
rotate (domyślnie),
copytruncate - należy pamiętać, że *copytruncate* nie może być używane razem z *maxdelay*. W takim przypadku *maxdelay* musi mieć wartość 0 lub nie może być określone; zobacz uwagi dotyczące copytruncate,
mtime-reread - rekordy nieunikalne, ponowne odczytanie, jeśli zmieni się czas modyfikacji lub rozmiar (domyślnie),
mtime-noreread - rekordy nieunikalne, ponowne odczytanie tylko jeśli zmieni się rozmiar (ignoruj zmianę czasu modyfikacji).
- **persistent dir** (tylko w zabbix_agentd w systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka do katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące persistent files.

Uwagi:

- Pozycja musi być skonfigurowana jako active check.
- Dopasowane wiersze są zliczane w nowych wierszach od ostatniego sprawdzenia pliku dziennika przez agenta, a więc zależą od interwału aktualizacji pozycji.
- Rotacja pliku dziennika opiera się na czasie ostatniej modyfikacji plików.
- W parametrze `file regexp` ścieżka do katalogu dziennika i wyrażenie regularne pliku dziennika muszą być oddzielone poprawnym separatorem katalogów:
  - w systemie Windows separatorem musi być ukośnik odwrotny (\); ukośniki ukośne mogą być tolerowane w innych miejscach, z wyjątkiem tego, który oddziela ścieżkę katalogu dziennika i wyrażenie regularne pliku dziennika (zobacz przykłady poniżej).
  - w innych systemach musi to być ukośnik ukośny (/).

Przykłady dla systemu Windows:

```
logrt.count["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja zliczy liczbę dopasowań wz
logrt.count["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja zliczy li
```

Przykłady dla innych systemów:

```
logrt.count["/home/zabbix/logs/.*","^logfile[0-9]{1,3}$",,100] #ta pozycja zliczy liczbę dopasowań wzorca
logrt.count["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #ta pozycja zliczy liczbę
```

modbus.get[endpoint,<slave id>,<function>,<address>,<count>,<type>,<endianness>,<offset>]

 Odczytuje dane Modbus.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **endpoint** - punkt końcowy zdefiniowany jako `protocol://connection_string`;
- **slave id** - identyfikator slave;
- **function** - funkcja Modbus;
- **address** - adres pierwszego rejestru, cewki lub wejścia;
- **count** - liczba rekordów do odczytu;
- **type** - typ danych;
- **endianness** - konfiguracja kolejności bajtów;
- **offset** - liczba rejestrów, począwszy od „address", których wyniki zostaną odrzucone.

W systemie Windows ta pozycja obsługuje ścieżki UNC w parametrze **endpoint**, co może stanowić zagrożenie bezpieczeństwa.

Zobacz szczegółowy opis parametrów.

net.dns[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Sprawdza stan usługi DNS.
 Zwracane wartości: 0 - rozwiązywanie DNS nie powiodło się (serwer DNS nie odpowiedział lub zwrócił błąd); 1 - rozwiązywanie DNS zakończyło się powodzeniem.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - adres IP serwera DNS (pozostaw puste dla domyślnego serwera DNS);
- **name** - nazwa DNS do odpytania;
- **type** - typ rekordu, o który zostanie wykonane zapytanie (domyślnie *SOA*);
- **timeout** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości parametru `type` to: *ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane przez Zabbix agent w systemie Windows, Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawione na *PTR*) można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (zobacz przykłady poniżej). Pamiętaj, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Międzynarodowe nazwy domen nie są obsługiwane; zamiast tego użyj nazw zakodowanych w IDNA.

Przykłady:

```
net.dns[198.51.100.1,example.com,MX,2,1]

net.dns[,198.51.100.1,PTR]
net.dns[,1.100.51.198.in-addr.arpa,PTR]

net.dns[,2a00:1450:400f:800::200e,PTR]
net.dns[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.dns.perf[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Sprawdza wydajność usługi DNS.
 Zwracana wartość: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund spędzonych na oczekiwaniu na odpowiedź z usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - adres IP serwera DNS (pozostaw puste dla domyślnego serwera DNS);
- **name** - nazwa DNS do odpytania;
- **type** - typ rekordu, o który zostanie wykonane zapytanie (domyślnie *SOA*);
- **timeout** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - liczba prób wykonania żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości parametru `type` to:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane przez Zabbix agent w systemie Windows, Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawione na *PTR*) można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (zobacz przykłady poniżej). Należy pamiętać, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Międzynarodowe nazwy domen nie są obsługiwane; zamiast tego należy używać nazw zakodowanych w IDNA.
- pozycja zwraca czas odpowiedzi zamiast 0, gdy serwer DNS odpowiada kodem błędu (na przykład `NXDOMAIN` lub `SERVFAIL`).

Przykłady:

```
net.dns.perf[198.51.100.1,example.com,MX,2,1]

net.dns.perf[,198.51.100.1,PTR]
net.dns.perf[,1.100.51.198.in-addr.arpa,PTR]

net.dns.perf[,2a00:1450:400f:800::200e,PTR]
net.dns.perf[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.dns.record[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Wykonuje zapytanie DNS.

Zabbix agent 2 udostępnia również net.dns.get, które oferuje dodatkowe funkcje, takie jak większa liczba typów rekordów oraz większa kontrola nad monitorowaniem DNS. Wartość zwracana: ciąg znaków zawierający wymagany typ informacji.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS, o którą ma zostać wykonane zapytanie;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie *SOA*);

- **timeout** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - liczba prób wykonania żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości parametru `type` to:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane przez Zabbix agent w systemie Windows oraz przez Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawione na *PTR*) można podać nazwę DNS w formacie odwróconym lub nieodwróconym (zobacz przykłady poniżej). Należy pamiętać, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Międzynarodowe nazwy domen nie są obsługiwane; zamiast tego należy używać nazw zakodowanych w IDNA.

Przykłady:

```
net.dns.record[198.51.100.1,example.com,MX,2,1]

net.dns.record[,198.51.100.1,PTR]
net.dns.record[,1.100.51.198.in-addr.arpa,PTR]

net.dns.record[,2a00:1450:400f:800::200e,PTR]
net.dns.record[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.if.collisions[if]

 Liczba kolizji poza oknem czasowym.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD. Uprawnienia root są wymagane w NetBSD.

Parametr:

- **if** - nazwa interfejsu sieciowego

net.if.discovery

 Lista interfejsów sieciowych. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

net.if.in[if,<mode>]

 Statystyki ruchu przychodzącego na interfejsie sieciowym.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Uprawnienia root są wymagane w systemie NetBSD.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęty w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);
- **mode** - możliwe wartości:
bytes - liczba bajtów (domyślnie)
packets - liczba pakietów
errors - liczba błędów
dropped - liczba odrzuconych pakietów
overruns (fifo) - liczba błędów bufora FIFO
frame - liczba błędów ramkowania pakietów
compressed - liczba skompresowanych pakietów odebranych przez sterownik urządzenia
multicast - liczba ramek multicast odebranych przez sterownik urządzenia

Uwagi:

- Możesz użyć tego klucza razem z krokiem przetwarzania wstępnego *Change per second*, aby uzyskać statystyki liczby bajtów na sekundę.
- Tryb *dropped* jest obsługiwany tylko w systemach Linux, FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Tryby *overruns*, *frame*, *compressed*, *multicast* są obsługiwane tylko w systemie Linux.
- W systemie HP-UX ta pozycja nie dostarcza szczegółów dotyczących interfejsów loopback (np. lo0).

Przykłady:

```
net.if.in[eth0]
net.if.in[eth0,errors]
```

net.if.out[if,<mode>]

 Statystyki ruchu wychodzącego na interfejsie sieciowym.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Uprawnienia root są wymagane w systemie NetBSD.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęty w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);
- **mode** - możliwe wartości:
bytes - liczba bajtów (domyślnie)
packets - liczba pakietów
errors - liczba błędów
dropped - liczba odrzuconych pakietów
overruns (fifo) - liczba błędów bufora FIFO
collisions (colls) - liczba kolizji wykrytych na interfejsie
carrier - liczba utrat nośnej wykrytych przez sterownik urządzenia
compressed - liczba skompresowanych pakietów przesłanych przez sterownik urządzenia

Uwagi:

- Możesz użyć tego klucza z krokiem przetwarzania wstępnego *Zmiana na sekundę*, aby uzyskać statystyki bajtów na sekundę.
- Tryb *dropped* jest obsługiwany tylko w systemach Linux, HP-UX.
- Tryby *overruns*, *collision*, *carrier*, *compressed* są obsługiwane tylko w systemie Linux.
- W systemie HP-UX ta pozycja nie dostarcza szczegółów dotyczących interfejsów loopback (np. lo0).

Przykłady:

```
net.if.out[eth0]
net.if.out[eth0,errors]
```

net.if.total[if,<mode>]

 Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Uprawnienia root są wymagane w systemie NetBSD.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęty w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);
- **mode** - możliwe wartości:
bytes - liczba bajtów (domyślnie)
packets - liczba pakietów
errors - liczba błędów
dropped - liczba odrzuconych pakietów
overruns (fifo) - liczba błędów bufora FIFO
collisions (colls) - liczba kolizji wykrytych na interfejsie
compressed - liczba skompresowanych pakietów wysłanych lub odebranych przez sterownik urządzenia

Uwagi:

- Możesz użyć tego klucza z krokiem preprocessing *Change per second*, aby uzyskać statystyki bajtów na sekundę.
- Tryb *dropped* jest obsługiwany tylko w systemach Linux i HP-UX. Odrzucone pakiety są obsługiwane tylko wtedy, gdy zarówno `net.if.in`, jak i `net.if.out` działają dla odrzuconych pakietów na Twojej platformie.
- Tryby *overruns*, *collision*, *compressed* są obsługiwane tylko w systemie Linux.
- W systemie HP-UX ta pozycja nie dostarcza szczegółów dotyczących interfejsów loopback (np. lo0).

Przykłady:

```
net.if.total[eth0]
net.if.total[eth0,errors]
```

net.tcp.listen[port]

 Sprawdza, czy ten port TCP jest w stanie LISTEN.
 Zwracane wartości: 0 - nie jest w stanie LISTEN; 1 - jest w stanie LISTEN.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X, Windows.

Parametr:

- **port** - numer portu TCP

W jądrach Linux 2.6.14 i nowszych informacje o nasłuchujących gniazdach TCP są pobierane, jeśli to możliwe, z interfejsu NETLINK jądra. W przeciwnym razie informacje są pobierane z plików /proc/net/tcp i /roc/net/tcp6.

Przykład:

```
net.tcp.listen[80]
```

net.tcp.port[<ip>,port]

 Sprawdza, czy możliwe jest nawiązanie połączenia TCP z określonym portem.
 Zwracane wartości: 0 - nie można się połączyć; 1 - można się połączyć.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - adres IP lub nazwa DNS (domyślnie 127.0.0.1);
- **port** - numer portu.

Uwagi:

- Do prostego testowania wydajności TCP użyj `net.tcp.service.perf[tcp,<ip>,<port>]`.
- Te testy mogą powodować pojawianie się dodatkowych komunikatów w plikach dziennika demona systemowego (zwykle rejestrowane są sesje SMTP i SSH).

Przykład:

```
net.tcp.port[,80] #tej pozycji można użyć do testowania dostępności serwera WWW działającego na porcie 80
```

net.tcp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i akceptuje połączenia TCP.
 Zwracane wartości: 0 - usługa jest niedostępna; 1 - usługa działa.
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* lub *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie 127.0.0.1);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Te testy mogą powodować pojawianie się dodatkowych komunikatów w plikach dziennika demonów systemowych (zwykle rejestrowane są sesje SMTP i SSH).
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście należy użyć `net.tcp.port[]` do takich testów.
- Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.
- Test telnet szuka monitu logowania (':' na końcu).

Przykład:

```
net.tcp.service[ftp,,45] #tej pozycji można użyć do testowania dostępności serwera FTP na porcie TCP 45
```

net.tcp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi TCP.
 Zwracane wartości: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund oczekiwania na odpowiedź od usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* lub *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie 127.0.0.1);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście, do takich kontroli użyj `net.tcp.service.perf[tcp,<ip>,<port>]`.
- Kontrola telnet szuka monitu logowania (':' na końcu).

Przykład:

```
net.tcp.service.perf[ssh] #ta pozycja może być użyta do testowania szybkości początkowej odpowiedzi z serw
```

net.tcp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>]

 Zwraca liczbę gniazd TCP pasujących do parametrów.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **laddr** - lokalny adres IPv4/6 lub podsieć CIDR;
- **lport** - numer portu lokalnego lub nazwa usługi;
- **raddr** - zdalny adres IPv4/6 lub podsieć CIDR;
- **rport** - numer portu zdalnego lub nazwa usługi;
- **state** - stan połączenia (*established*, *syn_sent*, *syn_recv*, *fin_wait1*, *fin_wait2*, *time_wait*, *close*, *close_wait*, *last_ack*, *listen*, *closing*).

Przykład:

```
net.tcp.socket.count[,80,,,established] #liczba połączeń z lokalnym portem TCP 80 w stanie established
```

net.udp.listen[port]

 Sprawdza, czy ten port UDP jest w stanie LISTEN.
 Zwracane wartości: 0 - nie jest w stanie LISTEN; 1 - jest w stanie LISTEN.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X.

Parametr:

- **port** - numer portu UDP

Przykład:

```
net.udp.listen[68]
```

net.udp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i odpowiada na żądania UDP.
 Zwracane wartości: 0 - usługa jest niedostępna; 1 - usługa działa.
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie 127.0.0.1);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service[ntp,,45] #this item can be used to test the availability of NTP service on UDP port 45
```

net.udp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi UDP.
 Zwracane wartości: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund oczekiwania na odpowiedź z usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie 127.0.0.1);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service.perf[ntp] #tej pozycji można użyć do testowania czasu odpowiedzi usługi NTP
```

net.udp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>]

 Zwraca liczbę gniazd UDP pasujących do parametrów.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **laddr** - lokalny adres IPv4/6 lub podsieć CIDR;
- **lport** - numer portu lokalnego lub nazwa usługi;
- **raddr** - zdalny adres IPv4/6 lub podsieć CIDR;
- **rport** - numer portu zdalnego lub nazwa usługi;
- **state** - stan połączenia (*established*, *unconn*).

Przykład:

```
net.udp.socket.count[,,,,established] #zwraca liczbę gniazd UDP w stanie połączenia
```

proc.cpu.util[<name>,<user>,<type>,<cmdline>,<mode>,<zone>]

 Procent wykorzystania CPU procesu.
 Zwracana wartość: *Float*.
 Obsługiwane platformy: Linux, Solaris[6].

Parametry:

- **name** - nazwa procesu (domyślnie *wszystkie procesy*);
- **user** - nazwa użytkownika (domyślnie *wszyscy użytkownicy*);
- **type** - typ wykorzystania CPU: *total* (domyślnie), *user* lub *system*;
- **cmdline** - filtrowanie według wiersza poleceń (jest to wyrażenie regularne);
- **mode** - tryb zbierania danych: *avg1* (domyślnie), *avg5* lub *avg15*;
- **zone** - docelowa strefa: *current* (domyślnie) lub *all*. Ten parametr jest obsługiwany tylko w systemie Solaris.

Komentarze:

- Zwracana wartość jest oparta na procencie wykorzystania pojedynczego rdzenia CPU. Na przykład wykorzystanie CPU przez proces w pełni używający dwóch rdzeni wynosi 200%.
- Dane o wykorzystaniu CPU procesu są zbierane przez kolektor, który obsługuje maksymalnie 1024 unikalne zapytania (według nazwy, użytkownika i wiersza poleceń). Zapytania, do których nie odwoływano się w ciągu ostatnich 24 godzin, są usuwane z kolektora.

- Jeśli parametr zone jest ustawiony na *current* (lub domyślnie), a agent został skompilowany w systemie Solaris bez obsługi stref, ale działa na nowszym systemie Solaris, w którym strefy są obsługiwane, agent zwróci NOTSUPPORTED (agent nie może ograniczyć wyników tylko do bieżącej strefy). Jednak w takim przypadku *all* jest obsługiwane.

Przykłady:

```
proc.cpu.util[,root] #wykorzystanie CPU wszystkich procesów uruchomionych przez użytkownika "root"
proc.cpu.util[zabbix_server,zabbix] #wykorzystanie CPU wszystkich procesów zabbix_server uruchomionych prz
```

proc.get[<name>,<user>,<cmdline>,<mode>]

 Lista procesów systemu operacyjnego i ich parametrów. Może być używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Windows, OpenBSD, NetBSD.

Parametry:

- **name** - nazwa procesu (domyślnie *wszystkie procesy*);
- **user** - nazwa użytkownika (domyślnie *wszyscy użytkownicy*);
- **cmdline** - filtrowanie według wiersza poleceń (jest to regularne wyrażenie). Ten parametr nie jest obsługiwany w systemie Windows; na innych platformach nie jest obsługiwany, jeśli mode jest ustawiony na 'summary'.
- **mode** - możliwe wartości:
process (domyślnie), *thread* (nieobsługiwane w NetBSD), *summary*. Zobacz listę parametrów procesu zwracanych dla każdego trybu i systemu operacyjnego.

Uwagi:

- Jeśli nie można pobrać wartości, na przykład z powodu błędu (proces już zakończył działanie, brak uprawnień, błąd wywołania systemowego), zostanie zwrócone −1.
- Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko systemu Linux).

Przykłady:

```
proc.get[zabbix_server,zabbix,,process] #lista wszystkich procesów zabbix_server uruchomionych przez użytk
proc.get[java,,,thread] #lista wszystkich procesów Java, zwraca jeden wpis na wątek
proc.get[,zabbix,,summary] #połączone dane dla procesów każdego typu uruchomionych przez użytkownika zabbi
```

proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]

 Pamięć używana przez proces w bajtach.
 Wartość zwracana: *Integer* - gdy `mode` ma wartość *max*, *min*, *sum*; *Float* - gdy `mode` ma wartość *avg*
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

Parametry:

- **name** - nazwa procesu (domyślnie *wszystkie procesy*);
- **user** - nazwa użytkownika (domyślnie *wszyscy użytkownicy*);
- **mode** - możliwe wartości: *avg*, *max*, *min* lub *sum* (domyślnie);
- **cmdline** - filtrowanie według linii poleceń (jest to regularne wyrażenie);
- **memtype** - typ pamięci używanej przez proces

Uwagi:

- Parametr `memtype` jest obsługiwany tylko w systemach Linux, FreeBSD, Solaris[6], AIX.
- Gdy kilka procesów używa pamięci współdzielonej, suma pamięci używanej przez procesy może dawać duże, nierealistyczne wartości.

Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko Linuxa).

Gdy ta pozycja jest wywoływana z wiersza poleceń i zawiera parametr linii poleceń (np. przy użyciu trybu testowego agent: `zabbix_agentd -t proc.mem[,,,apache2]`), zostanie zliczony jeden dodatkowy proces, ponieważ agent policzy również samego siebie.

Przykłady:

```
proc.mem[,root] #pamięć używana przez wszystkie procesy uruchomione przez użytkownika "root"
proc.mem[zabbix_server,zabbix] #pamięć używana przez wszystkie procesy zabbix_server uruchomione przez uży
proc.mem[,oracle,max,oracleZABBIX] #pamięć używana przez proces zużywający najwięcej pamięci, uruchomiony
```

proc.num[<name>,<user>,<state>,<cmdline>,<zone>]

 Liczba procesów.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[6], HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parametry:

- **name** - nazwa procesu (domyślnie *all processes*);
- **user** - nazwa użytkownika (domyślnie *all users*);

- **state** - możliwe wartości:
all (domyślnie),
disk - nieprzerywalny sen,
run - uruchomiony,
sleep - przerywalny sen,
trace - zatrzymany,
zomb - zombie;
- **cmdline** - filtruj według linii poleceń (jest to wyrażenie regularne expression);
- **zone** - strefa docelowa: *current* (domyślnie) lub *all*. Ten parametr jest obsługiwany tylko w Solaris.

Komentarze:

- Parametry stanu *disk* i *trace* są obsługiwane tylko w Linux, FreeBSD, OpenBSD, NetBSD.
- W Windows obsługiwane są tylko parametry `name` i `user`.
- Gdy ta pozycja jest wywoływana z wiersza poleceń i zawiera parametr linii poleceń (np. przy użyciu trybu testowego agent: `zabbix_agentd -t proc.num[,,,apache2]`), zostanie zliczony jeden dodatkowy proces, ponieważ agent policzy również siebie.
- Ustawienie parametru `zone` na *current* (lub domyślnie) w przypadku, gdy agent został skompilowany w Solaris bez obsługi stref, ale działa na nowszym Solaris, gdzie strefy są obsługiwane, spowoduje zwrócenie NOTSUPPORTED (agent nie może ograniczyć wyników wyłącznie do bieżącej strefy). Jednak *all* jest w tym przypadku obsługiwane.
- Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko Linux).

Przykłady:

```
proc.num[,mysql] #liczba procesów uruchomionych pod użytkownikiem mysql
proc.num[apache2,www-data] #liczba procesów apache2 uruchomionych pod użytkownikiem www-data
proc.num[,oracle,sleep,oracleZABBIX] #liczba procesów w stanie sleep uruchomionych pod Oracle, mających or
```

sensor[device,sensor,<mode>]

 Odczyt czujnika sprzętowego.
 Zwracana wartość: *Float*.
 Obsługiwane platformy: Linux, OpenBSD.

Parametry:

- **device** - nazwa urządzenia;
- **sensor** - nazwa czujnika;
- **mode** - możliwe wartości: *avg*, *max* lub *min*.

Uwagi:

- Odczytuje /proc/sys/dev/sensors w systemie Linux 2.4.
- Odczytuje /sys/class/hwmon w systemie Linux 2.6+. Zobacz bardziej szczegółowy opis pozycji sensor w systemie Linux.
- Odczytuje MIB *hw.sensors* w systemie OpenBSD.
- W systemie Linux 2.4, jeśli określisz mode, wtedy device i sensor są traktowane jako wyrażenia regularne.
- W systemach Linux 2.6+ i OpenBSD lub jeśli mode zostanie pominięty, device i sensor są traktowane dosłownie.

Przykłady:

```
sensor[w83781d-i2c-0-2d,temp1]
sensor[cpu0,temp0] #the temperature of one CPU
sensor["cpu[0-2]$",temp,avg] #the average temperature of the first three CPUs
```

system.boottime

 Czas uruchomienia systemu.
 Zwracana wartość: *Liczba całkowita (znacznik czasu Unix)*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

system.cpu.discovery

 Lista wykrytych procesorów CPU/rdzeni CPU. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Zobacz obsługiwane platformy.

system.cpu.intr

 Przerwania urządzenia.
 Zwracana wartość: *Liczba całkowita*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.load[<cpu>,<mode>]

 Obciążenie CPU.
 Zwracana wartość: *Float*.
 Zobacz obsługiwane platformy.

Parametry:

- **cpu** — możliwe wartości: *all* (domyślnie) lub *percpu* (całkowite obciążenie podzielone przez liczbę aktywnych CPU);
- **mode** — możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*.

Przykład:

```
system.cpu.load[,avg5]
```

system.cpu.num[<type>]

 Liczba procesorów.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Parametr:

- **type** — możliwe wartości: *online* (domyślnie) lub *max*

Parametr typu *max* jest obsługiwany tylko w systemach Linux, FreeBSD, Solaris, MacOS X.

Przykład:

```
system.cpu.num
```

system.cpu.switches

 Liczba przełączeń kontekstu.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.util[<cpu>,<type>,<mode>,<logical or physical>]

 Procent wykorzystania CPU.
 Zwracana wartość: *Float*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parametry:

- **cpu** - <*numer CPU*> lub *all* (domyślnie);
- **type** - możliwe wartości: *user* (domyślnie), *idle*, *nice*, *system*, *iowait*, *interrupt*, *softirq*, *steal*, *spin* (w OpenBSD), *guest* (w jądrach Linux 2.6.24 i nowszych) lub *guest_nice* (w jądrach Linux 2.6.33 i nowszych);
- **mode** - możliwe wartości: *avg1* (średnia jednominutowa, domyślnie), *avg5* lub *avg15*;
- **logical or physical** - możliwe wartości: *logical* (domyślnie) lub *physical*. Ten parametr jest obsługiwany tylko w AIX.

Uwagi:

- Parametr typu *nice* jest obsługiwany tylko w systemach Linux, FreeBSD, HP-UX, OpenBSD, NetBSD.
- Parametr typu *iowait* jest obsługiwany tylko w systemach Linux 2.6 i nowszych, Solaris, AIX.
- Parametr typu *interrupt* jest obsługiwany tylko w systemach Linux 2.6 i nowszych, FreeBSD, OpenBSD.
- Parametry typu *softirq*, *steal*, *guest*, *guest_nice* są obsługiwane tylko w systemach Linux 2.6 i nowszych.
- Parametry trybu *avg5* i *avg15* są obsługiwane w systemach Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD.

Przykład:

```
system.cpu.util[0,user,avg5]
```

system.hostname[<type>,<transform>]

 Nazwa hosta systemu.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Parametry:

- **type** - możliwe wartości: *netbios* (domyślnie w systemie Windows), *host* (domyślnie w systemie Linux), *shorthost* (zwraca część nazwy hosta przed pierwszą kropką, pełny ciąg dla nazw bez kropek), *fqdn* (zwraca w pełni kwalifikowaną nazwę domenową, Fully Qualified Domain Name);
- **transform** - możliwe wartości: *none* (domyślnie) lub *lower* (konwertuje na małe litery).

Wartość jest pobierana z `nodename` w danych wyjściowych systemowego API uname().

Przykłady zwracanych wartości:

```
system.hostname → linux-w7x1
system.hostname → example.com
system.hostname[shorthost] → example
system.hostname → WIN-SERV2008-I6
system.hostname[host] → Win-Serv2008-I6LonG
system.hostname[host,lower] → win-serv2008-i6long
system.hostname[fqdn,lower] → blog.zabbix.com
```

system.hw.chassis[<info>]

 Informacje o obudowie.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux.

Parametr:

- **info** - możliwe wartości: *full* (domyślnie), *model*, *serial*, *type* lub *vendor*

Uwagi:

- Ten klucz pozycji zależy od dostępności tabeli SMBIOS.
- Nastąpi próba odczytu tabeli DMI z sysfs; jeśli dostęp do sysfs się nie powiedzie, zostanie podjęta próba bezpośredniego odczytu z pamięci.
- Wymagane są **uprawnienia root**, ponieważ wartość jest uzyskiwana przez odczyt z sysfs lub pamięci.

Przykład:

```
system.hw.chassis[full] → Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXX Desktop
```

system.hw.cpu[<cpu>,<info>]

 Informacje o CPU.
 Zwracana wartość: *String* lub *Integer*.
 Obsługiwane platformy: Linux.

Parametry:

- **cpu** — *<numer CPU>* lub *all* (domyślnie);
- **info** — możliwe wartości: *full* (domyślnie), *curfreq*, *maxfreq*, *model* lub *vendor*.

Uwagi:

- Zbiera informacje z `/proc/cpuinfo` oraz `/sys/devices/system/cpu/[cpunum]/cpufreq/cpuinfo_max_freq`.
- Jeśli podano numer CPU oraz *curfreq* lub *maxfreq*, zwracana jest wartość liczbowa (Hz).

Przykład:

```
system.hw.cpu[0,vendor] → AuthenticAMD
```

system.hw.devices[<type>]

 Lista urządzeń PCI lub USB.
 Zwracana wartość: *Tekst*.
 Obsługiwane platformy: Linux.

Parametr:

- **type** - *pci* (domyślnie) lub *usb*

Zwraca dane wyjściowe narzędzia lspci lub lsusb (uruchamianego bez żadnych parametrów).

Przykład:

```
system.hw.devices → 00:00.0 Host bridge: Advanced Micro Devices [AMD] RS780 Host Bridge
```

system.hw.macaddr[<interface>,<format>]

 Lista adresów MAC.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux.

Parametry:

- **interface** - *all* (domyślnie) lub regularne wyrażenie;
- **format** - *full* (domyślnie) lub *short*

Uwagi:

- Wyświetla adresy MAC interfejsów, których nazwa pasuje do podanego regularnego wyrażenia `interface` (*all* wyświetla dla wszystkich interfejsów).
- Jeśli `format` jest określony jako *short*, nazwy interfejsów i identyczne adresy MAC nie są wyświetlane.

Przykład:

```
system.hw.macaddr["eth0$",full] → [eth0] 00:11:22:33:44:55
```

system.localtime[<type>]

 Czas systemowy.
 Zwracana wartość: *Integer* - gdy `type` ma wartość *utc*; *String* - gdy `type` ma wartość *local*.
 Zobacz obsługiwane platformy.

Parametry:

- **type** - możliwe wartości: *utc* - (domyślnie) czas od Epoki (00:00:00 UTC, 1 stycznia 1970), mierzony w sekundach, lub *local* - czas w formacie 'yyyy-mm-dd,hh:mm:ss.nnn,+hh:mm'

Musi być używane jako sprawdzenie pasywne dla Zabbix agent; w Zabbix agent 2 może być skonfigurowane jako sprawdzenie aktywne.

Przykład:

```
system.localtime[local] #utwórz pozycja używając tego klucza, a następnie użyj jej do wyświetlenia czasu h
```

system.run[command,<mode>]

 Uruchamia określone polecenie na hoście.
 Zwracana wartość: wynik *Text* polecenia lub 1 - przy `mode` ustawionym na *nowait* (niezależnie od wyniku polecenia).
 Zobacz obsługiwane platformy.

Parametry:

- **command** - polecenie do wykonania;
- **mode** - możliwe wartości: *wait* - czekaj na zakończenie wykonania (domyślnie) lub *nowait* - nie czekaj.

Uwagi:

- Ta pozycja jest domyślnie wyłączona. Dowiedz się, jak włączyć je.
- Zwracana wartość pozycji to standardowe wyjście wraz ze standardowym błędem wygenerowanym przez polecenie. Sprawdzanie kodu zakończenia nie jest wykonywane.
- Aby przetwarzanie przebiegało poprawnie, zwracana wartość polecenia musi mieć typ danych `text`. Pusty wynik jest również dozwolony.
- Zwracana wartość jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.
- Zobacz także: Wykonywanie poleceń.

Przykład:

```
system.run[ls -l /] #zwraca szczegółową listę plików katalogu głównego
```

system.stat[resource,<type>]

 Statystyki systemu.
 Zwracana wartość: *Integer* lub *float*.
 Obsługiwane platformy: AIX.

Parametry:

- **ent** - liczba jednostek procesora, do których ta partycja jest uprawniona (float);
- **kthr,<type>** - informacje o stanach wątków jądra:
r - średnia liczba wątków jądra gotowych do uruchomienia (float)
b - średnia liczba wątków jądra umieszczonych w kolejce oczekiwania Menedżera Pamięci Wirtualnej (float)
- **memory,<type>** - informacje o użyciu pamięci wirtualnej i rzeczywistej:
avm - aktywne strony wirtualne (integer)
fre - rozmiar listy wolnych stron (integer)
- **page,<type>** - informacje o błędach stron i aktywności stronicowania:
fi - liczba stron plików wczytanych na sekundę (float)
fo - liczba stron plików zapisanych na sekundę (float)
pi - strony wczytane z przestrzeni stronicowania (float)
po - strony zapisane do przestrzeni stronicowania (float)
fr - zwolnione strony (zastępowanie stron) (float)
sr - strony przeskanowane przez algorytm zastępowania stron (float)
- **faults,<type>** - częstotliwość pułapek i przerwań:
in - przerwania urządzeń (float)
sy - wywołania systemowe (float)
cs - przełączenia kontekstu wątków jądra (float)
- **cpu,<type>** - podział procentowego wykorzystania czasu procesora:
us - czas użytkownika (float)
sy - czas systemowy (float)
id - czas bezczynności (float)
wa - czas bezczynności, podczas którego system miał oczekujące żądania wejścia/wyjścia dysku/NFS (float)
pc - liczba wykorzystanych fizycznych procesorów (float)
ec - procent wykorzystanej przydzielonej mocy obliczeniowej (float)
lbusy - wskazuje procent wykorzystania procesorów logicznych, który wystąpił podczas wykonywania na poziomie użytkownika i systemu (float)
app - wskazuje dostępne procesory fizyczne we wspólnej puli (float)
- **disk,<type>** - statystyki dysku:
bps - wskazuje ilość danych przesłanych (odczytanych lub zapisanych) na dysk w bajtach na sekundę (integer)
tps - wskazuje liczbę transferów na sekundę skierowanych do fizycznego dysku/taśmy (float)

Uwagi:

- Zwróć uwagę na następujące ograniczenia tych pozycji:
 `system.stat[cpu,app]` - obsługiwane tylko w AIX LPAR typu "Shared";
 `system.stat[cpu,ec]` - obsługiwane w AIX LPAR typu "Shared" i "Dedicated" ("Dedicated" zawsze zwraca 100 (procent));
 `system.stat[cpu,lbusy]` - obsługiwane tylko w AIX LPAR typu "Shared";
 `system.stat[cpu,pc]` - obsługiwane w AIX LPAR typu "Shared" i "Dedicated";
 `system.stat[ent]` - obsługiwane w AIX LPAR typu "Shared" i "Dedicated".

system.sw.arch

 Informacja o architekturze oprogramowania.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Informacja jest pobierana za pomocą funkcji `uname()`.

Przykład:

```
system.sw.arch → i686
```

system.sw.os[<info>]

 Informacje o systemie operacyjnym.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux, Windows.

Parametr:

- **info** - możliwe wartości: *full* (domyślnie), *short* lub *name*

Informacje są pobierane z (należy pamiętać, że nie wszystkie pliki i opcje są obecne we wszystkich dystrybucjach):

- `/proc/version` (*full*) w systemie Linux;
- `/proc/version_signature` (*short*) w systemie Linux;
- parametr PRETTY_NAME z `/etc/os-release` w systemach Linux, które go obsługują, lub `/etc/issue.net` (*name*);
- klucz rejestru `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion` w systemie Windows.

Przykłady:

```
system.sw.os[short] → Ubuntu 2.6.35-28.50-generic 2.6.35.11
system.sw.os[full] → [s|Windows 10 Enterprise 22621.1.asd64fre.ni_release.220506-1250 Build 22621.963]
```

system.sw.os.get

 Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja główna i pomocnicza itp.).
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, Windows.

system.sw.packages[<regexp>,<manager>,<format>]

 Lista zainstalowanych pakietów.
 Wartość zwracana: *Text*.
 Obsługiwane platformy: Linux.

Parametry:

- **regexp** - *all* (domyślnie) lub wyrażenie regularne expression;
- **manager** - *all* (domyślnie) lub menedżer pakietów;
- **format** - *full* (domyślnie) lub *short*.

Uwagi:

- Wyświetla alfabetycznie zainstalowane pakiety, których nazwa pasuje do podanego wyrażenia regularnego expression (*all* wyświetla wszystkie).
- Obsługiwane menedżery pakietów (wykonywane polecenie):
dpkg (dpkg --get-selections)
pkgtool (ls /var/log/packages)
rp (rpm -qa)
pacman (pacman -Q)
portage
- Jeśli `format` ma wartość *full*, pakiety są grupowane według menedżerów pakietów (każdy menedżer w osobnym wierszu rozpoczynającym się od jego nazwy w nawiasach kwadratowych).
- Jeśli `format` ma wartość *short*, pakiety nie są grupowane i są wyświetlane w jednym wierszu.

Przykład:

```
system.sw.packages[mini,dpkg,short] → python-minimal, python2.6-minimal, ubuntu-minimal
```

system.sw.packages.get[<regexp>,<manager>]

 Szczegółowa lista zainstalowanych pakietów.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux.

Parametry:

- **regexp** — *all* (domyślnie) lub regularne wyrażenie;
- **manager** — *all* (domyślnie) lub menedżer pakietów (możliwe wartości: *rpm*, *dpkg*, *pkgtool*, *pacman* lub *portage*).

Uwagi:

- Zwraca niesformatowany JSON z zainstalowanymi pakietami, których nazwa pasuje do podanego wyrażenia regularnego.
- Dane wyjściowe są tablicą obiektów, z których każdy zawiera następujące klucze: name, manager, version, size, architecture, buildtime oraz installtime (zobacz więcej szczegółów).

system.swap.in[<device>,<type>]

 Statystyki swap-in (z urządzenia do pamięci).
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, OpenBSD.

Parametry:

- **device** - określa urządzenie używane do swapowania (tylko Linux) lub *all* (domyślnie);
- **type** - możliwe wartości: *count* (liczba swapinów, domyślnie na platformach innych niż Linux), *sectors* (sektory wczytane do swap), lub *pages* (strony wczytane do swap, domyślnie na Linux).

Komentarze:

- Źródłem tych informacji są:
/proc/swaps, /proc/partitions, /proc/stat (Linux 2.4)
/proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)
- Należy pamiętać, że *pages* będzie działać tylko wtedy, gdy nie określono device.
- Parametr typu *sectors* jest obsługiwany tylko na Linux.

Przykład:

```
system.swap.in[,pages]
```

system.swap.out[<device>,<type>]

 Statystyki swap-out (z pamięci na urządzenie).
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, OpenBSD.

Parametry:

- **device** - określa urządzenie używane do swapowania (tylko Linux) lub *all* (domyślnie);
- **type** - możliwe wartości: *count* (liczba operacji swap-out, domyślnie na platformach innych niż Linux), *sectors* (sektory przeniesione do obszaru wymiany) lub *pages* (strony przeniesione do obszaru wymiany, domyślnie w systemie Linux).

Uwagi:

- Źródłem tych informacji są:
`/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4)
`/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6)
- Należy pamiętać, że *pages* będzie działać tylko wtedy, gdy nie określono urządzenia.
- Parametr typu *sectors* jest obsługiwany tylko w systemie Linux.

Przykład:

```
system.swap.out[,pages]
```

system.swap.size[<device>,<type>]

 Rozmiar przestrzeni wymiany w bajtach lub jako procent całości.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, Windows.

Parametry:

- **device** - określa urządzenie używane do swapu (tylko FreeBSD) lub *all* (domyślnie);
- **type** - możliwe wartości: *free* (wolna przestrzeń wymiany, domyślnie), *pfree* (wolna przestrzeń wymiany, w procentach), *pused* (używana przestrzeń wymiany, w procentach), *total* (całkowita przestrzeń wymiany) lub *used* (używana przestrzeń wymiany).

Uwagi:

- Należy pamiętać, że *pfree*, *pused* nie są obsługiwane w systemie Windows, jeśli rozmiar swapu wynosi 0.
- Jeśli urządzenie nie zostanie określone, agent Zabbix uwzględni tylko urządzenia (pliki) swapu, a pamięć fizyczna zostanie pominięta. Na przykład w systemach Solaris polecenie `swap -s` uwzględnia część pamięci fizycznej oraz urządzenia swapu (w przeciwieństwie do `swap -l`).

Przykład:

```
system.swap.size[,pfree] → procent wolnej przestrzeni wymiany
```

system.uname

 Identyfikacja systemu.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Uwagi:

- W systemach UNIX wartość tej pozycji jest uzyskiwana za pomocą wywołania systemowego uname().
- W systemie Windows pozycja zwraca architekturę systemu operacyjnego, natomiast w systemach UNIX zwraca architekturę CPU.

Przykłady:

```
system.uname → FreeBSD localhost 4.2-RELEASE FreeBSD 4.2-RELEASE #0: Mon Nov i386
system.uname → Windows ZABBIX-WIN 6.0.6001 Microsoft® Windows Server® 2008 Standard Service Pack 1 x86
```

system.uptime

 Czas działania systemu w sekundach.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD, Windows.

W konfiguracji pozycji użyj jednostek **s** lub **uptime**, aby uzyskać czytelne wartości.

system.users.num

 Liczba zalogowanych użytkowników.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Do uzyskania tej wartości po stronie agent używane jest polecenie **who**.

vfs.dev.discovery

 Lista urządzeń blokowych i ich typów. Używane do wykrywania niskiego poziomu.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux.

vfs.dev.read[<device>,<type>,<mode>]

 Statystyki odczytu dysku.
 Wartość zwracana: *Integer* - dla `type` ustawionego na *sectors*, *operations*, *bytes*; *Float* - dla `type` ustawionego na *sps*, *ops*, *bps*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parametry:

- **device** - urządzenie dyskowe (domyślnie *all* [3]);
- **type** - możliwe wartości: *sectors*, *operations*, *bytes*, *sps*, *ops* lub *bps* (*sps*, *ops*, *bps* oznaczają odpowiednio: sektory, operacje, bajty na sekundę);
- **mode** - możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*. Ten parametr jest obsługiwany tylko przy `type` ustawionym na: sps, ops, bps.

Uwagi:

- Jeśli używany jest interwał aktualizacji wynoszący trzy godziny lub więcej[2], ta pozycja zawsze zwróci '0'.
- Parametry typu *sectors* i *sps* są obsługiwane tylko w systemie Linux.
- Parametr typu *ops* jest obsługiwany tylko w systemach Linux i FreeBSD.
- Parametr typu *bps* jest obsługiwany tylko w systemie FreeBSD.
- Parametr typu *bytes* jest obsługiwany tylko w systemach FreeBSD, Solaris, AIX, OpenBSD.
- Parametr `mode` jest obsługiwany tylko w systemach Linux, FreeBSD.
- Można używać nazw względnych urządzeń (na przykład `sda`) oraz opcjonalnego prefiksu /dev/ (na przykład `/dev/sda`).
- Obsługiwane są logiczne woluminy LVM.
- Domyślne wartości parametru 'type' dla różnych systemów operacyjnych:
AIX - operations
FreeBSD - bps
Linux - sps
OpenBSD - operations
Solaris - bytes
- *sps*, *ops* i *bps* na obsługiwanych platformach są ograniczone do 1024 urządzeń (1023 indywidualnych i jednego dla *all*).

Przykład:

```
vfs.dev.read[,operations]
```

vfs.dev.write[<device>,<type>,<mode>]

 Statystyki zapisu na dysku.
 Wartość zwracana: *Integer* - dla `type` w *sectors*, *operations*, *bytes*; *Float* - dla `type` w *sps*, *ops*, *bps*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parametry:

- **device** - urządzenie dyskowe (domyślnie *all* [3]);
- **type** - możliwe wartości: *sectors*, *operations*, *bytes*, *sps*, *ops* lub *bps* (*sps*, *ops*, *bps* oznaczają odpowiednio: sektory, operacje, bajty na sekundę);
- **mode** - możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*. Ten parametr jest obsługiwany tylko wtedy, gdy `type` ma wartość: sps, ops, bps.

Uwagi:

- Jeśli używany jest interwał aktualizacji wynoszący trzy godziny lub więcej[2], ta pozycja zawsze zwróci '0'.
- Parametry typu *sectors* i *sps* są obsługiwane tylko w systemie Linux.
- Parametr typu *ops* jest obsługiwany tylko w systemie Linux i FreeBSD.
- Parametr typu *bps* jest obsługiwany tylko w systemie FreeBSD.
- Parametr typu *bytes* jest obsługiwany tylko w systemie FreeBSD, Solaris, AIX, OpenBSD.
- Parametr `mode` jest obsługiwany tylko w systemie Linux, FreeBSD.
- Można używać nazw urządzeń względnych (na przykład `sda`) oraz opcjonalnego prefiksu /dev/ (na przykład `/dev/sda`).
- Obsługiwane są logiczne woluminy LVM.
- Wartości domyślne parametru 'type' dla różnych systemów operacyjnych:
AIX - operations
FreeBSD - bps
Linux - sps
OpenBSD - operations
Solaris - bytes
- *sps*, *ops* i *bps* na obsługiwanych platformach są ograniczone do 1024 urządzeń (1023 pojedynczych i jednego dla *all*).

Przykład:

```
vfs.dev.write[,operations]
```

vfs.dir.count[dir,<regex incl>,<regex excl>,<types incl>,<types excl>,<max depth>,<min size>,<max size>,<min age>,<max age>,<regex excl dir>]

 Liczba wpisów katalogu.
 Zwracana wartość: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **dir** - ścieżka bezwzględna do katalogu;
- **regex incl** - regularne wyrażenie opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego), które mają zostać uwzględnione; jeśli puste, uwzględniane są wszystkie (wartość domyślna);
- **regex excl** - regularne wyrażenie opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego), które mają zostać wykluczone; jeśli puste, nic nie jest wykluczane (wartość domyślna);
- **types incl** - typy wpisów katalogu do zliczania, możliwe wartości: *file* - zwykły plik, *dir* - podkatalog, *sym* - dowiązanie symboliczne, *sock* - gniazdo, *bdev* - urządzenie blokowe, *cdev* - urządzenie znakowe, *fifo* - FIFO, *dev* - synonim "bdev,cdev", *all* - wszystkie typy (domyślnie), tj. "file,dir,sym,sock,bdev,cdev,fifo". Wiele typów musi być rozdzielonych przecinkami i ujętych w cudzysłów.
- **types excl** - typy wpisów katalogu (zobacz `types incl`), które NIE mają być zliczane. Jeśli jakiś typ wpisu występuje zarówno w `types incl`, jak i `types excl`, wpisy katalogu tego typu NIE są zliczane.
- **max depth** - maksymalna głębokość przeszukiwania podkatalogów:
-1 (domyślnie) - bez ograniczeń,
0 - bez schodzenia do podkatalogów.
- **min size** - minimalny rozmiar (w bajtach), aby plik został zliczony. Mniejsze pliki nie będą zliczane. Można używać sufiksów pamięci.
- **max size** - maksymalny rozmiar (w bajtach), aby plik został zliczony. Większe pliki nie będą zliczane. Można używać sufiksów pamięci.
- **min age** - minimalny wiek (w sekundach) wpisu katalogu, aby został zliczony. Nowsze wpisy nie będą zliczane. Można używać sufiksów czasu.
- **max age** - maksymalny wiek (w sekundach) wpisu katalogu, aby został zliczony. Wpisy starsze lub w tym wieku nie będą zliczane (czas modyfikacji). Można używać sufiksów czasu.
- **regex excl dir** - regularne wyrażenie opisujące wzorzec nazwy katalogu, który ma zostać wykluczony. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do regex_excl)

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Zmienne środowiskowe, np. %APP_HOME%, $HOME i %TEMP%, nie są obsługiwane.
- Pseudokatalogi "." i ".." nigdy nie są zliczane.
- Dowiązania symboliczne nigdy nie są śledzone podczas przeszukiwania katalogów.
- Zarówno `regex incl`, jak i `regex excl` są stosowane do plików i katalogów przy obliczaniu liczby wpisów, ale są ignorowane przy wybieraniu podkatalogów do przeszukiwania (jeśli `regex incl` ma wartość "(?i)^.+\.zip$" i `max depth` nie jest ustawione, wtedy wszystkie podkatalogi zostaną przeszukane, ale zliczone zostaną tylko pliki typu zip).
- Czas wykonania jest ograniczony wartością timeout w konfiguracji agent (3 sek.). Ponieważ przeszukiwanie dużych katalogów może trwać dłużej, żadne dane nie zostaną zwrócone, a pozycja stanie się nieobsługiwana. Częściowy wynik zliczania nie zostanie zwrócony.
- Przy filtrowaniu według rozmiaru tylko zwykłe pliki mają znaczące rozmiary. W systemach Linux i BSD katalogi również mają rozmiary różne od zera (zwykle kilka KB). Urządzenia mają rozmiar zero, np. rozmiar **/dev/sda1** nie odzwierciedla rozmiaru odpowiedniej partycji. Dlatego przy używaniu <min_size> i <max_size> zaleca się określenie <types_incl> jako "*file*", aby uniknąć niespodzianek.

Przykłady:

```
vfs.dir.count[/dev] #monitoruje liczbę urządzeń w /dev (Linux)
vfs.dir.count["C:\Users\ADMINI~1\AppData\Local\Temp"] #monitoruje liczbę plików w katalogu tymczasowym
```

vfs.dir.get[dir,<regex incl>,<regex excl>,<types incl>,<types excl>,<max depth>,<min size>,<max size>,<min age>,<max age>,<regex excl dir>]

 Lista wpisów katalogu.
 Zwracana wartość: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **dir** - ścieżka bezwzględna do katalogu;
- **regex incl** - regularne wyrażenie opisujące wzorzec nazwy obiektu (pliku, katalogu, dowiązania symbolicznego), który ma zostać uwzględniony; jeśli puste, uwzględniane są wszystkie (wartość domyślna);
- **regex excl** - regularne wyrażenie opisujące wzorzec nazwy obiektu (pliku, katalogu, dowiązania symbolicznego), który ma zostać wykluczony; jeśli puste, nic nie jest wykluczane (wartość domyślna);

- **types incl** - typy wpisów katalogu do wyświetlenia, możliwe wartości: *file* - zwykły plik, *dir* - podkatalog, *sym* - dowiązanie symboliczne, *sock* - gniazdo, *bdev* - urządzenie blokowe, *cdev* - urządzenie znakowe, *fifo* - FIFO, *dev* - synonim „bdev,cdev", *all* - wszystkie typy (domyślnie), tj. „file,dir,sym,sock,bdev,cdev,fifo". Wiele typów musi być oddzielonych przecinkami i ujętych w cudzysłów.
- **types excl** - typy wpisów katalogu (zobacz `types incl`), które NIE mają być wyświetlane. Jeśli jakiś typ wpisu znajduje się zarówno w `types incl`, jak i w `types excl`, wpisy katalogu tego typu NIE będą wyświetlane.
- **max depth** - maksymalna głębokość przeszukiwania podkatalogów:
-1 (domyślnie) - bez ograniczeń,
0 - bez schodzenia do podkatalogów.
- **min size** - minimalny rozmiar pliku (w bajtach), aby został wyświetlony. Mniejsze pliki nie będą wyświetlane. Można używać sufiksów pamięci.
- **max size** - maksymalny rozmiar pliku (w bajtach), aby został wyświetlony. Większe pliki nie będą wyświetlane. Można używać sufiksów pamięci.
- **min age** - minimalny wiek wpisu katalogu (w sekundach), aby został wyświetlony. Nowsze wpisy nie będą wyświetlane. Można używać sufiksów czasu.
- **max age** - maksymalny wiek wpisu katalogu (w sekundach), aby został wyświetlony. Wpisy starsze oraz o dokładnie takim wieku nie będą wyświetlane (czas modyfikacji). Można używać sufiksów czasu.
- **regex excl dir** - regularne wyrażenie opisujące wzorzec nazwy katalogu, który ma zostać wykluczony. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do `regex excl`)

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Zmienne środowiskowe, np. %APP_HOME%, $HOME i %TEMP%, nie są obsługiwane.
- Pseudokatalogi "." i ".." nigdy nie są wyświetlane.
- Dowiązania symboliczne nigdy nie są śledzone podczas przeszukiwania katalogów.
- Zarówno `regex incl`, jak i `regex excl` są stosowane do plików i katalogów podczas generowania listy wpisów, ale są ignorowane przy wybieraniu podkatalogów do przeszukania (jeśli `regex incl` ma wartość "(?i)^.+\.zip$" i `max depth` nie jest ustawione, wtedy wszystkie podkatalogi zostaną przeszukane, ale uwzględnione zostaną tylko pliki typu zip).
- Czas wykonania jest ograniczony wartością timeout w konfiguracji agent. Ponieważ przeszukiwanie dużych katalogów może trwać dłużej, żadne dane nie zostaną zwrócone, a pozycja stanie się nieobsługiwana. Częściowa lista nie zostanie zwrócona.
- Podczas filtrowania według rozmiaru tylko zwykłe pliki mają sensowne rozmiary. W systemach Linux i BSD katalogi również mają rozmiary różne od zera (zwykle kilka KB). Urządzenia mają rozmiar zero, np. rozmiar **/dev/sda1** nie odzwierciedla rozmiaru odpowiedniej partycji. Dlatego przy używaniu `min size` i `max size` zaleca się ustawienie `types incl` na "*file*", aby uniknąć niespodzianek.

Przykłady:

```
vfs.dir.get[/dev] #pobiera listę urządzeń w /dev (Linux)
vfs.dir.get["C:\Users\ADMINI~1\AppData\Local\Temp"] #pobiera listę plików w katalogu tymczasowym
```

vfs.dir.size[dir,<regex incl>,<regex excl>,<mode>,<max depth>,<regex excl dir>]

 Rozmiar katalogu (w bajtach).
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, Windows. Ta pozycja może działać także na innych platformach typu UNIX.

Parametry:

- **dir** - ścieżka bezwzględna do katalogu;
- **regex incl** - wyrażenie regularne opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego), która ma zostać uwzględniona; jeśli puste, uwzględniane są wszystkie (wartość domyślna);
- **regex excl** - wyrażenie regularne opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego), która ma zostać wykluczona; jeśli puste, nic nie jest wykluczane (wartość domyślna);
- **mode** - możliwe wartości: *apparent* (domyślnie) - pobiera pozorne rozmiary plików zamiast użycia dysku (działa jak `du -sb dir`), *disk* - pobiera użycie dysku (działa jak `du -s -B1 dir`). W przeciwieństwie do polecenia du, pozycja vfs.dir.size uwzględnia ukryte pliki przy obliczaniu rozmiaru katalogu (działa jak `du -sb .[^.]* *` w obrębie dir).
- **max depth** - maksymalna głębokość przeszukiwanych podkatalogów: **-1** (domyślnie) - bez ograniczeń, **0** - bez schodzenia do podkatalogów.
- **regex excl dir** - wyrażenie regularne opisujące wzorzec nazwy katalogu, który ma zostać wykluczony. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do `regex excl`)

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Obliczane są tylko katalogi, do których użytkownik *zabbix* ma co najmniej uprawnienie odczytu. Dla katalogów mających tylko uprawnienie odczytu obliczany jest rozmiar samego katalogu. Dla katalogów mających uprawnienia odczytu i wykonania obliczany jest rozmiar wraz z zawartością.
- W przypadku dużych katalogów lub wolnych dysków dla tej pozycji może wystąpić przekroczenie czasu z powodu ustawienia Timeout w plikach konfiguracyjnych agent oraz serwer/proxy. W razie potrzeby zwiększ wartości timeout.

- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
vfs.dir.size[/tmp,log] #oblicza rozmiar wszystkich plików w /tmp zawierających 'log' w nazwie
vfs.dir.size[/tmp,log,^.+\.old$] #oblicza rozmiar wszystkich plików w /tmp zawierających 'log' w nazwie, z
```

vfs.file.cksum[file,<mode>]

 Suma kontrolna pliku, obliczana za pomocą algorytmu UNIX cksum.
 Zwracana wartość: *Integer* - gdy mode to *crc32*, *String* - gdy mode to *md5*, *sha256*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - *crc32* (domyślnie), *md5* lub *sha256*.

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.cksum[/etc/passwd]
```

Przykłady zwracanych wartości (odpowiednio crc32/md5/sha256):

```
675436101
9845acf68b73991eb7fd7ee0ded23c44
ae67546e4aac995e5c921042d0cf0f1f7147703aa42bfbfb65404b30f238f2dc
```

vfs.file.contents[file,<encoding>]

 Pobieranie zawartości pliku[7].
 Wartość zwracana: *Text*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **encoding** - identyfikator strony kodowej identyfikator.

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.
- Zwracany jest pusty ciąg, jeśli plik jest pusty lub zawiera wyłącznie znaki LF/CR.
- Znacznik kolejności bajtów (BOM) jest wykluczony z wyniku.

Przykład:

```
vfs.file.contents[/etc/passwd]
```

vfs.file.exists[file,<types incl>,<types excl>]

 Sprawdza, czy plik istnieje.
 Zwracana wartość: 0 - nie znaleziono; 1 - istnieje plik określonego typu.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **types incl** - lista typów plików do uwzględnienia, możliwe wartości: *file* (zwykły plik, domyślnie, jeśli types_excl nie jest ustawiony), *dir* (katalog), *sym* (dowiązanie symboliczne), *sock* (gniazdo), *bdev* (urządzenie blokowe), *cdev* (urządzenie znakowe), *fifo* (FIFO), *dev* (synonim „bdev,cdev”), *all* (wszystkie wymienione typy, domyślnie, jeśli types_excl jest ustawiony).
- **types excl** - lista typów plików do wykluczenia, możliwe wartości jak dla types_incl (domyślnie żadne typy nie są wykluczane)

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Wiele typów musi być oddzielonych przecinkiem, a cały zestaw musi być ujęty w cudzysłowy ””.
- Jeśli ten sam typ występuje zarówno w <types_incl>, jak i <types_excl>, pliki tego typu są wykluczane.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
vfs.file.exists[/tmp/application.pid]
vfs.file.exists[/tmp/application.pid,"file,dir,sym"]
vfs.file.exists[/tmp/application_dir,dir]
```

vfs.file.get[file]

 Zwraca informacje o pliku.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametr:

- **file** — pełna ścieżka do pliku

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Obsługiwane typy plików w systemach typu UNIX: plik zwykły, katalog, dowiązanie symboliczne, gniazdo, urządzenie blokowe, urządzenie znakowe, FIFO.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.get[/etc/passwd] #zwraca JSON z informacjami o pliku /etc/passwd (typ, użytkownik, uprawnienia, S
```

vfs.file.md5sum[file]

 Suma kontrolna MD5 pliku.
 Zwracana wartość: Ciąg znaków (skrót MD5 pliku).
 Zobacz obsługiwane platformy.

Parametr:

- **file** - pełna ścieżka do pliku

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.md5sum[/usr/local/etc/zabbix_agentd.conf]
```

Przykład zwracanej wartości:

```
b5052decb577e0fffd622d6ddc017e82
```

vfs.file.owner[file,<ownertype>,<resulttype>]

 Pobiera właściciela pliku.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **ownertype** - *user* (domyślnie) lub *group* (tylko Unix);
- **resulttype** - *name* (domyślnie) lub *id*; dla id - zwraca uid/gid w systemie Unix, SID w systemie Windows.

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
vfs.file.owner[/tmp/zabbix_server.log] #zwraca właściciela pliku /tmp/zabbix_server.log
vfs.file.owner[/tmp/zabbix_server.log,,id] #zwraca identyfikator właściciela pliku /tmp/zabbix_server.log
```

vfs.file.permissions[file]

 Zwraca 4-cyfrowy ciąg zawierający liczbę ósemkową z uprawnieniami UNIX.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux. Pozycja może działać na innych platformach typu UNIX.

Parametry:

- **file** — pełna ścieżka do pliku

Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.permissions[/etc/passwd] #zwraca uprawnienia dla /etc/passwd, na przykład '0644'
```

vfs.file.regexp[file,regexp,<encoding>,<start line>,<end line>,<output>]

 Pobiera ciąg znaków z pliku[7].
 Wartość zwracana: Wiersz zawierający dopasowany ciąg znaków lub wartość określona przez opcjonalny parametr `output`.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **regexp** - regularne wyrażenie opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej;
- **start line** - numer pierwszego wiersza do przeszukania (domyślnie pierwszy wiersz pliku);
- **end line** - numer ostatniego wiersza do przeszukania (domyślnie ostatni wiersz pliku);
- **output** - opcjonalny szablon formatowania danych wyjściowych. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, od którego zaczyna się dopasowanie, do znaku, na którym dopasowanie się kończy), natomiast sekwencja ucieczki **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem znaków, jeśli N przekracza liczbę przechwyconych grup).

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.
- Zwracany jest tylko pierwszy pasujący wiersz.
- Jeśli żaden wiersz nie pasuje do wyrażenia, zwracany jest pusty ciąg znaków.
- Znacznik kolejności bajtów (BOM) jest wykluczany z danych wyjściowych.
- Wyodrębnianie zawartości przy użyciu parametru `output` odbywa się po stronie agent.

Przykłady:

```
vfs.file.regexp[/etc/passwd,zabbix]
vfs.file.regexp[/path/to/some/file,"([0-9]+)$",,3,5,\1]
vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1] → pobranie identyfikatora użytkownika *zabbix*
```

vfs.file.regmatch[file,regexp,<encoding>,<start line>,<end line>]

 Znajdź ciąg znaków w pliku[7].
 Zwracane wartości: 0 - nie znaleziono dopasowania; 1 - znaleziono.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **regexp** - regularne wyrażenie opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej;
- **start line** - numer pierwszego wiersza do przeszukania (domyślnie pierwszy wiersz pliku);
- **end line** - numer ostatniego wiersza do przeszukania (domyślnie ostatni wiersz pliku).

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.
- Znacznik kolejności bajtów (BOM) jest ignorowany.

Przykład:

```
vfs.file.regmatch[/var/log/app.log,error]
```

vfs.file.size[file,<mode>]

 Rozmiar pliku (w bajtach).
 Zwracana wartość: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - możliwe wartości: *bytes* (domyślnie) lub *lines* (liczone są również puste wiersze).

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Plik musi mieć uprawnienia do odczytu dla użytkownika *zabbix*.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.size[/var/log/syslog]
```

vfs.file.time[file,<mode>]

 Informacje o czasie pliku.
 Zwracana wartość: *Integer* (znacznik czasu Unix).
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - możliwe wartości:
modify (domyślnie) - czas ostatniej modyfikacji zawartości pliku,
access - czas ostatniego odczytu pliku,
change - czas ostatniej zmiany właściwości pliku

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.time[/etc/passwd,modify]
```

vfs.fs.discovery

 Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

vfs.fs.get

 Lista zamontowanych systemów plików wraz z ich typem, dostępnym miejscem na dysku, statystykami inode oraz opcjami montowania. Może być używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Uwagi:

- Systemy plików z liczbą inode równą zero, co może mieć miejsce w przypadku systemów plików z dynamicznymi inode (np. btrfs), są również raportowane.
- Zobacz także: Wykrywanie zamontowanych systemów plików.

vfs.fs.inode[fs,<mode>]

 Liczba lub wartość procentowa inode'ów.
 Zwracana wartość: *Integer* - dla liczby; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametry:

- **fs** - system plików;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *used*, *pfree* (wolne, procentowo) lub *pused* (użyte, procentowo).

Jeśli liczba inode'ów jest równa zero, co może mieć miejsce w przypadku systemów plików z dynamicznymi inode'ami (np. btrfs), wartości pfree/pused będą raportowane odpowiednio jako „100" i „0".

Przykład:

```
vfs.fs.inode[/,pfree]
```

vfs.fs.size[fs,<mode>]

 Miejsce na dysku w bajtach lub jako wartość procentowa całkowitej pojemności.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametry:

- **fs** - system plików;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *used*, *pfree* (wolne, procentowo) lub *pused* (użyte, procentowo).

Uwagi:

- Jeśli system plików nie jest zamontowany, zwracany jest rozmiar lokalnego systemu plików, w którym znajduje się punkt montowania.
- Uwzględniana jest zarezerwowana przestrzeń systemu plików i nie jest ona wliczana przy użyciu trybu *free*.

Przykład:

```
vfs.fs.size[/tmp,free]
```

vm.memory.size[<mode>]

 Rozmiar pamięci w bajtach lub jako wartość procentowa całkowitej pamięci.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametr:

- **mode** - możliwe wartości: *total* (domyślnie), *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*, *used*, *pused* (używana, wartość procentowa), *available* lub *pavailable* (dostępna, wartość procentowa).

Komentarze:

- Ta pozycja akceptuje trzy kategorie parametrów:
  - *total* - całkowita ilość pamięci;
  - specyficzne dla platformy typy pamięci: *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*;
  - oszacowania na poziomie użytkownika dotyczące ilości używanej i dostępnej pamięci: *used*, *pused*, *available*, *pavailable*.
- Parametr trybu *active* jest obsługiwany tylko w systemach FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Parametry trybu *anon*, *exec*, *file* są obsługiwane tylko w systemie NetBSD.
- Parametr trybu *buffers* jest obsługiwany tylko w systemach Linux, FreeBSD, OpenBSD, NetBSD.
- Parametr trybu *cached* jest obsługiwany tylko w systemach Linux, FreeBSD, AIX, OpenBSD, NetBSD.
- Parametry trybu *inactive*, *wired* są obsługiwane tylko w systemach FreeBSD, MacOS X, OpenBSD, NetBSD.
- Parametr trybu *pinned* jest obsługiwany tylko w systemie AIX.
- Parametr trybu *shared* jest obsługiwany tylko w systemach Linux 2.4, FreeBSD, OpenBSD, NetBSD.
- Zobacz także dodatkowe szczegóły dotyczące tej pozycji.

Przykład:

```
vm.memory.size[pavailable]
```

web.page.get[host,<path>,<port>]

 Pobiera zawartość strony internetowej.
 Wartość zwracana: kod źródłowy strony internetowej jako tekst (łącznie z nagłówkami).
 Zobacz obsługiwane platformy.

Parametry:

- **host** — nazwa hosta lub URL (w postaci `scheme://host:port/path`, gdzie wymagany jest tylko *host*). Dozwolone schematy URL: *http*, *https*[4]. Brakujący schemat będzie traktowany jako *http*. Jeśli podano URL, parametry `path` i `port` muszą być puste. Podanie nazwy użytkownika/hasła podczas łączenia z serwerami wymagającymi uwierzytelniania, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Punycode jest obsługiwany w nazwach hostów.
- **path** — ścieżka do dokumentu HTML (domyślnie /);
- **port** — numer portu (domyślnie 80 dla HTTP)

Uwagi:

- Ta pozycja staje się nieobsługiwana, jeśli zasób określony w `host` nie istnieje lub jest niedostępny.
- `host` może być nazwą hosta, nazwą domenową, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi być skompilowany z włączoną obsługą IPv6.

Przykłady:

```
web.page.get[www.example.com,index.php,80]
web.page.get[https://www.example.com]
web.page.get[https://blog.example.com/?s=zabbix]
web.page.get[localhost:80]
web.page.get["[::1]/server-status"]
```

web.page.perf[host,<path>,<port>]

 Czas ładowania pełnej strony WWW (w sekundach).
 Zwracana wartość: *Float*.
 Zobacz obsługiwane platformy.

Parametry:

- **host** — nazwa hosta lub URL (w postaci `scheme://host:port/path`, gdzie tylko *host* jest obowiązkowy). Dozwolone schematy URL: *http*, *https*[4]. Brakujący schemat będzie traktowany jako *http*. Jeśli podano URL, `path` i `port` muszą być puste. Podanie nazwy użytkownika/hasła podczas łączenia z serwerami wymagającymi uwierzytelnienia, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Punycode jest obsługiwany w nazwach hostów.
- **path** — ścieżka do dokumentu HTML (domyślnie /);

- **port** — numer portu (domyślnie 80 dla HTTP)

Uwagi:

- Ta pozycja staje się nieobsługiwana, jeśli zasób określony w `host` nie istnieje lub jest niedostępny.
- `host` może być nazwą hosta, nazwą domenową, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi być skompilowany z włączoną obsługą IPv6.

Przykłady:

```
web.page.perf[www.example.com,index.php,80]
web.page.perf[https://www.example.com]
```

web.page.regexp[host,<path>,<port>,regexp,<length>,<output>]

 Znajdź ciąg znaków na stronie internetowej.
 Zwracana wartość: Dopasowany ciąg znaków lub wartość określona przez opcjonalny parametr `output`.
 Zobacz obsługiwane platformy.

Parametry:

- **host** — nazwa hosta lub URL (w postaci `scheme://host:port/path`, gdzie tylko *host* jest obowiązkowy). Dozwolone schematy URL: *http*, *https*[4]. Brakujący schemat będzie traktowany jako *http*. Jeśli podano URL, parametry `path` i `port` muszą być puste. Podanie nazwy użytkownika/hasła podczas łączenia z serwerami wymagającymi uwierzytelniania, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Punycode jest obsługiwany w nazwach hostów.
- **path** — ścieżka do dokumentu HTML (domyślnie /);
- **port** — numer portu (domyślnie 80 dla HTTP)
- **regexp** — regularne wyrażenie opisujące wymagany wzorzec;
- **length** — maksymalna liczba znaków do zwrócenia;
- **output** — opcjonalny szablon formatowania danych wyjściowych. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, od którego zaczyna się dopasowanie, do znaku, na którym dopasowanie się kończy), natomiast sekwencja ucieczki **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup).

Uwagi:

- Ta pozycja staje się nieobsługiwana, jeśli zasób określony w `host` nie istnieje lub jest niedostępny.
- `host` może być nazwą hosta, nazwą domenową, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi być skompilowany z włączoną obsługą IPv6.
- Ekstrakcja zawartości przy użyciu parametru `output` odbywa się po stronie agenta.

Przykłady:

```
web.page.regexp[www.example.com,index.php,80,OK,2]
web.page.regexp[https://www.example.com,,,OK,2]
```

agent.hostmetadata

 Metadane hosta agenta.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Zwraca wartość parametrów HostMetadata lub HostMetadataItem, albo pusty ciąg znaków, jeśli żaden z nich nie jest zdefiniowany.

agent.hostname

 Nazwa hosta agenta.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Zwraca:

- Jako sprawdzenie pasywne — nazwę pierwszego hosta wymienionego w parametrze Hostname pliku konfiguracyjnego agenta;
- Jako sprawdzenie aktywne — nazwę bieżącego hosta.

agent.ping

 Sprawdzenie dostępności agenta.
 Zwracana wartość: Nic — niedostępny; 1 — dostępny.
 Zobacz obsługiwane platformy.

Użyj funkcji wyzwalacza **nodata()**, aby sprawdzić niedostępność hosta.

agent.variant

 Wariant agenta Zabbix (Zabbix agent lub Zabbix agent 2).
 Zwracana wartość: 1 - Zabbix agent; 2 - Zabbix agent 2.
 Zobacz obsługiwane platformy.

agent.version

 Wersja agenta Zabbix.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Przykład zwracanej wartości:

```
6.0.3
```

zabbix.stats[<ip>,<port>]

 Zwraca zestaw wewnętrznych metryk serwera Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - lista masek IP/DNS/sieci serwerów/proxy, które mają być odpytywane zdalnie (domyślnie 127.0.0.1);
- **port** - port serwera/proxy, który ma być odpytywany zdalnie (domyślnie 10051)

Uwagi:

- Ta pozycja zwraca wybrany zestaw wewnętrznych metryk. Szczegóły można znaleźć w sekcji Zdalne monitorowanie statystyk Zabbix.
- Należy pamiętać, że żądanie statystyk zostanie zaakceptowane tylko z adresów wymienionych w parametrze 'StatsAllowedIP' serwer/proxy w instancji docelowej.

zabbix.stats[<ip>,<port>,queue,<from>,<to>]

 Zwraca liczbę monitorowanych pozycji w kolejce, które są opóźnione na serwerze Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - lista masek IP/DNS/sieci serwerów/proxy, które mają być odpytywane zdalnie (domyślnie 127.0.0.1);
- **port** - port serwera/proxy, który ma być odpytywany zdalnie (domyślnie 10051)
- **queue** - stała (należy użyć bez zmian)
- **from** - opóźnienie co najmniej o (domyślnie 6 sekund)
- **to** - opóźnienie co najwyżej o (domyślnie nieskończoność)

Należy pamiętać, że żądanie statystyk zostanie zaakceptowane tylko z adresów wymienionych w parametrze 'StatsAllowedIP' serwera/proxy w instancji docelowej.

Przypisy

[1] Uwaga specyficzna dla systemu Linux. Zabbix agent musi mieć dostęp tylko do odczytu do systemu plików */proc*. Łaty jądra z www.grsecurity.org ograniczają prawa dostępu użytkowników nieuprzywilejowanych.

[2] `vfs.dev.read[]`, `vfs.dev.write[]`: Zabbix agent zakończy „nieaktywne" połączenia urządzeń, jeśli wartości pozycji nie będą odczytywane przez ponad 3 godziny. Może się tak zdarzyć, jeśli system ma urządzenia z dynamicznie zmieniającymi się ścieżkami lub jeśli urządzenie zostanie ręcznie usunięte. Należy również pamiętać, że te pozycje, jeśli używają interwału aktualizacji wynoszącego 3 godziny lub więcej, zawsze zwrócą „0".

[3] `vfs.dev.read[]`, `vfs.dev.write[]`: Jeśli dla pierwszego parametru użyta jest domyślna wartość *all*, klucz zwróci statystyki zbiorcze obejmujące wszystkie urządzenia blokowe, takie jak sda, sdb i ich partycje (sda1, sda2, sdb3...), a także urządzenia wielodyskowe (MD raid) oparte na tych urządzeniach blokowych/partycjach oraz woluminy logiczne (LVM) oparte na tych urządzeniach blokowych/partycjach. W takich przypadkach zwracane wartości należy traktować wyłącznie jako wartości względne (zmienne w czasie), a nie jako wartości bezwzględne.

[4] SSL (HTTPS) jest obsługiwane tylko wtedy, gdy agent został skompilowany z obsługą cURL. W przeciwnym razie pozycja stanie się nieobsługiwana.

[5] Wartości `bytes` i `errors` nie są obsługiwane dla interfejsów loopback w systemach Solaris do wersji Solaris 10 6/06 włącznie, ponieważ statystyki bajtów, błędów i wykorzystania nie są przechowywane i/lub raportowane przez jądro. Jeśli jednak monitorujesz system Solaris przez net-snmp, wartości mogą być zwracane, ponieważ net-snmp zawiera starszy kod z cmu-snmp, datowany nawet na 1997 rok, który w przypadku niepowodzenia odczytu wartości bajtów ze statystyk interfejsu zwraca licznik pakietów (który istnieje na interfejsach loopback) pomnożony przez arbitralną wartość 308. Zakłada to, że średnia długość pakietu wynosi 308 oktetów, co jest bardzo przybliżonym oszacowaniem, ponieważ limit MTU dla interfejsów loopback w systemach Solaris wynosi 8892 bajty. Nie należy zakładać, że te wartości są poprawne ani nawet w przybliżeniu dokładne. Są to jedynie szacunki. Zabbix agent nie stosuje żadnych przybliżeń, ale net-snmp zwróci wartość dla tych pól.

[6] Wiersz poleceń w systemie Solaris, pobierany z /proc/pid/psinfo, jest ograniczony do 80 bajtów i zawiera wiersz poleceń w postaci z momentu uruchomienia procesu.

[7] Pozycje `vfs.file.contents[]`, `vfs.file.regexp[]`, `vfs.file.regmatch[]` mogą być używane do pobierania zawartości plików. Jeśli chcesz ograniczyć dostęp do określonych plików zawierających informacje wrażliwe, uruchom Zabbix agent jako użytkownika, który nie ma uprawnień dostępu do przeglądania tych plików.

Użycie z narzędziami wiersza poleceń

Należy pamiętać, że podczas testowania lub używania kluczy pozycji za pomocą `zabbix_agentd` lub `zabbix_get` z wiersza poleceń należy również uwzględnić składnię powłoki.

Na przykład, jeśli określony parametr klucza musi być ujęty w podwójne cudzysłowy, należy jawnie poprzedzić podwójne cudzysłowy znakiem ucieczki, w przeciwnym razie zostaną one usunięte przez powłokę jako znaki specjalne i nie zostaną przekazane do narzędzia Zabbix.

Przykłady:

```
zabbix_agentd -t 'vfs.dir.count[/var/log,,,"file,dir",,0]'

zabbix_agentd -t vfs.dir.count[/var/log,,,\"file,dir\",,0]
```

Ustawienia kodowania

Aby upewnić się, że pobrane dane nie są uszkodzone, można określić poprawne kodowanie do przetwarzania sprawdzenia (np. 'vfs.file.contents') w parametrze `encoding`. Listę obsługiwanych kodowań (identyfikatorów stron kodowych) można znaleźć w dokumentacji libiconv (GNU Project) lub w dokumentacji Microsoft Windows SDK dla "Code Page Identifiers". Należy pamiętać, że Microsoft czasami oznacza niektóre strony kodowe jako „dostępne tylko dla aplikacji zarządzanych" — tzn. udostępniane wyłącznie przez środowisko uruchomieniowe .NET i niedostępne przez natywne interfejsy API Win32. agent Zabbix implementuje własną logikę konwersji kodowania, dlatego te strony kodowe są obsługiwane przez agent Zabbix nawet wtedy, gdy natywne funkcje systemu Windows ich nie udostępniają.

Jeśli w parametrze `encoding` nie zostanie określone kodowanie, stosowane są następujące strategie rozpoznawania:

- Jeśli kodowanie nie jest określone (lub jest pustym ciągiem), przyjmuje się UTF-8, a dane są przetwarzane „tak jak są";
- Analiza BOM - dotyczy pozycji 'vfs.file.contents', 'vfs.file.regexp', 'vfs.file.regmatch'. Podejmowana jest próba określenia poprawnego kodowania na podstawie znacznika kolejności bajtów (BOM) na początku pliku. Jeśli BOM nie jest obecny - zamiast tego stosowane jest standardowe rozpoznawanie (patrz wyżej).

Rozwiązywanie problemów z pozycjami agenta

W przypadku sprawdzeń pasywnych, aby zapobiec sytuacji, w której pozycja nie otrzyma żadnej wartości, ponieważ żądanie serwera do agenta najpierw przekroczy limit czasu, należy zwrócić uwagę na następujące kwestie:

- Jeśli wersja agenta jest starsza niż wersja serwera, wartość *Timeout* w konfiguracji pozycji (lub globalny limit czasu) może wymagać ustawienia na wyższą niż wartość `Timeout` w pliku konfiguracyjnym agenta.
- Jeśli wersja agenta jest nowsza niż wersja serwera, wartość `Timeout` w pliku konfiguracyjnym serwera może wymagać ustawienia na wyższą niż wartość `Timeout` w pliku konfiguracyjnym agenta.

1 Zabbix agent 2

Zabbix agent 2 obsługuje wszystkie klucze pozycja obsługiwane przez Zabbix agent dla Unix i Windows. Ta strona zawiera szczegóły dotyczące dodatkowych kluczy pozycji, których można używać wyłącznie z Zabbix agent 2, pogrupowanych według wtyczki, do której należą.

Klucze pozycji są wymienione bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Klucz pozycji	Opis	Wtyczka
ceph.df.details	Wykorzystanie danych klastra i ich rozkład między pulami.	Ceph
ceph.osd.stats	Zagregowane statystyki oraz statystyki dla poszczególnych OSD.	
ceph.osd.discovery	Lista wykrytych OSD.	
ceph.osd.dump	Progi wykorzystania i stany OSD.	
ceph.ping	Sprawdza, czy można nawiązać połączenie z Ceph.	
ceph.pool.discovery	Lista wykrytych pul.	
ceph.status	Ogólny stan klastra.	
docker.container_info	Informacje niskiego poziomu o kontenerze.	Docker
docker.container_stats	Statystyki wykorzystania zasobów kontenera.	
docker.containers	Zwraca listę kontenerów.	
docker.containers.discovery	Zwraca listę kontenerów. Używane do niskopoziomowego wykrywania.	
docker.data.usage	Informacje o bieżącym wykorzystaniu danych.	

Klucz pozycji	Opis	Wtyczka
docker.images	Zwraca listę obrazów.	
docker.images.discovery	Zwraca listę obrazów. Używane do niskopoziomowego wykrywania.	
docker.info	Informacje o systemie.	
docker.ping	Sprawdza, czy demon Docker działa.	
ember.get	Zwraca wynik dla wymaganego urządzenia.	Ember+
memcached.ping	Sprawdza, czy połączenie jest aktywne.	Memcached
memcached.stats	Pobiera wynik polecenia STATS.	
mongodb.collection.stats	Zwraca różne statystyki magazynu dla określonej kolekcji.	MongoDB
mongodb.collections.discovery	Zwraca listę wykrytych kolekcji.	
mongodb.collections.usage	Zwraca statystyki wykorzystania kolekcji.	
mongodb.connpool.stats	Zwraca informacje o otwartych połączeniach wychodzących z bieżącej instancji bazy danych do innych członków klastra shardowanego lub zestawu replik.	
mongodb.db.stats	Zwraca statystyki odzwierciedlające stan określonego systemu bazy danych.	
mongodb.db.discovery	Zwraca listę wykrytych baz danych.	
mongodb.jumbo_chunks.count	Zwraca liczbę jumbo chunks.	
mongodb.oplog.stats	Zwraca stan zestawu replik na podstawie danych odczytanych z oplog.	
mongodb.ping	Sprawdza, czy połączenie jest aktywne.	
mongodb.rs.config	Zwraca bieżącą konfigurację zestawu replik.	
mongodb.rs.status	Zwraca stan zestawu replik z punktu widzenia członka, na którym uruchomiono metodę.	
mongodb.server.status	Zwraca stan bazy danych.	
mongodb.sh.discovery	Zwraca listę wykrytych shardów obecnych w klastrze.	
mongodb.version	Zwraca wersję serwera bazy danych.	
mqtt.get	Subskrybuje określony temat lub tematy (z symbolami wieloznacznymi) podanego brokera i czeka na publikacje.	MQTT
mssql.availability.group.get	Zwraca grupy dostępności.	MSSQL
mssql.custom.query	Zwraca wynik zapytania niestandardowego.	
mssql.db.get	Zwraca wszystkie dostępne bazy danych MSSQL.	
mssql.job.status.get	Zwraca stan zadań.	
mssql.last.backup.get	Zwraca czas ostatniej kopii zapasowej dla wszystkich baz danych.	
mssql.local.db.get	Zwraca bazy danych uczestniczące w grupie dostępności Always On i replikach (podstawowej lub pomocniczej), które znajdują się na serwerze, z którym nawiązano połączenie.	
mssql.mirroring.get	Zwraca informacje o mirroringu.	
mssql.nonlocal.db.get	Zwraca bazy danych uczestniczące w grupie dostępności Always On i replikach (podstawowej lub pomocniczej), które znajdują się na innych serwerach (baza danych nie jest lokalna względem instancji SQL Server, z którą nawiązano połączenie).	
mssql.perfcounter.get	Zwraca liczniki wydajności.	
mssql.ping	Sprawdza, czy połączenie jest aktywne.	
mssql.quorum.get	Zwraca informacje o kworum.	
mssql.quorum.member.get	Zwraca członków kworum.	
mssql.replica.get	Zwraca repliki.	
mssql.version	Zwraca wersję MSSQL.	
mysql.custom.query	Zwraca wynik zapytania niestandardowego.	MySQL
mysql.db.discovery	Zwraca listę baz danych MySQL.	
mysql.db.size	Rozmiar bazy danych w bajtach.	
mysql.get_status_variables	Wartości globalnych zmiennych stanu.	
mysql.ping	Sprawdza, czy połączenie jest aktywne.	
mysql.replication.discovery	Zwraca listę replikacji MySQL.	
mysql.replication.get_slave_status	Stan replikacji.	
mysql.version	Wersja MySQL.	
net.dns.get	Wykonuje zapytanie DNS i zwraca szczegółowe informacje o rekordach DNS.	Network
nvml.device.count	Liczba urządzeń GPU.	NVIDIA GPU
nvml.device.decoder.utilization	Wykorzystanie dekodera urządzenia GPU.	
nvml.device.ecc.mode	Tryb ECC urządzenia GPU.	
nvml.device.encoder.stats.get	Statystyki enkodera urządzenia GPU.	
nvml.device.encoder.utilization	Wykorzystanie enkodera urządzenia GPU.	
nvml.device.energy.consumption	Zużycie energii przez urządzenie GPU.	
nvml.device.errors.memory	Informacje o błędach pamięci ECC urządzenia GPU.	

Klucz pozycji	Opis	Wtyczka
nvml.device.errors.register	Informacje o błędach rejestrów ECC urządzenia GPU.	
nvml.device.fan.speed.avg	Średnia prędkość wentylatora urządzenia GPU.	
nvml.device.get	Wykryte GPU z UUID i nazwą.	
nvml.device.graphics.frequency	Częstotliwość zegara grafiki urządzenia GPU.	
nvml.device.memory.bar1.get	Informacje o pamięci BAR1 urządzenia GPU.	
nvml.device.memory.fb.get	Informacje o pamięci framebuffer urządzenia GPU.	
nvml.device.memory.frequency	Częstotliwość zegara pamięci urządzenia GPU.	
nvml.device.pci.utilization	Wykorzystanie PCI urządzenia GPU.	
nvml.device.performance.state	Stan wydajności urządzenia GPU.	
nvml.device.power.limit	Limit mocy urządzenia GPU.	
nvml.device.power.usage	Zużycie mocy urządzenia GPU.	
nvml.device.serial	Numer seryjny urządzenia GPU.	
nvml.device.sm.frequency	Częstotliwość zegara multiprocesora strumieniowego urządzenia GPU.	
nvml.device.temperature	Temperatura urządzenia GPU.	
nvml.device.utilization	Statystyki wykorzystania urządzenia GPU.	
nvml.device.video.frequency	Częstotliwość zegara wideo urządzenia GPU.	
nvml.system.driver.version	Wersja sterownika NVIDIA.	
nvml.version	Wersja biblioteki NVML.	
oracle.diskgroups.stats	Zwraca statystyki grup dysków Automatic Storage Management (ASM).	Oracle
oracle.diskgroups.discovery	Zwraca listę grup dysków ASM.	
oracle.archive.info	Statystyki dzienników archiwalnych.	
oracle.cdb.info	Informacje o Container Databases (CDBs).	
oracle.custom.query	Wynik zapytania niestandardowego.	
oracle.datafiles.stats	Zwraca statystyki plików danych.	
oracle.db.discovery	Zwraca listę baz danych.	
oracle.fra.stats	Zwraca statystyki Fast Recovery Area (FRA).	
oracle.instance.info	Statystyki instancji.	
oracle.pdb.info	Informacje o Pluggable Databases (PDBs).	
oracle.pdb.discovery	Zwraca listę PDB.	
oracle.pga.stats	Zwraca statystyki Program Global Area (PGA).	
oracle.ping	Sprawdza, czy można nawiązać połączenie z Oracle.	
oracle.proc.stats	Zwraca statystyki procesów.	
oracle.redolog.info	Informacje o pliku dziennika z pliku kontrolnego.	
oracle.sga.stats	Zwraca statystyki System Global Area (SGA).	
oracle.sessions.stats	Zwraca statystyki sesji.	
oracle.sys.metrics	Zwraca zestaw wartości metryk systemowych.	
oracle.sys.params	Zwraca zestaw wartości parametrów systemowych.	
oracle.ts.stats	Zwraca statystyki tablespace’ów.	
oracle.ts.discovery	Zwraca listę tablespace’ów.	
oracle.user.info	Zwraca informacje o użytkowniku Oracle.	
oracle.version	Zwraca wersję serwera bazy danych.	
pgsql.autovacuum.count	Liczba procesów roboczych autovacuum.	PostgreSQL
pgsql.archive	Informacje o zarchiwizowanych plikach.	
pgsql.bgwriter	Łączna liczba checkpointów dla klastra bazy danych, z podziałem według typu checkpointu.	
pgsql.cache.hit	Współczynnik trafień w buforowej pamięci podręcznej PostgreSQL.	
pgsql.connections	Zwraca połączenia według typu.	
pgsql.custom.query	Zwraca wynik zapytania niestandardowego.	
pgsql.db.age	Wiek najstarszego FrozenXID bazy danych.	
pgsql.db.bloating_tables	Liczba spuchniętych tabel w poszczególnych bazach danych.	
pgsql.db.discovery	Lista baz danych PostgreSQL.	
pgsql.db.size	Rozmiar bazy danych w bajtach.	
pgsql.dbstat	Zbiera statystyki dla poszczególnych baz danych.	
pgsql.dbstat.sum	Zestawione dane dla wszystkich baz danych w klastrze.	
pgsql.locks	Informacje o przyznanych blokadach dla poszczególnych baz danych.	
pgsql.oldest.xid	Wiek najstarszego XID.	
pgsql.ping	Sprawdza, czy połączenie jest aktywne.	
pgsql.queries	Metryki zapytań według czasu wykonania.	
pgsql.replication.count	Liczba serwerów standby.	
pgsql.replication.process	Opóźnienie flush, opóźnienie write i opóźnienie replay dla każdego procesu wysyłającego.	

Klucz pozycji	Opis	Wtyczka
pgsql.replication.process.discovery	Wykrywanie nazw procesów replikacji.	
pgsql.replication.recovery_role	Stan odzyskiwania.	
pgsql.replication.status	Stan replikacji.	
pgsql.replication_lag.b	Opóźnienie replikacji w bajtach.	
pgsql.replication_lag.sec	Opóźnienie replikacji w sekundach.	
pgsql.uptime	Czas działania PostgreSQL w milisekundach.	
pgsql.version	Zwraca wersję PostgreSQL.	
pgsql.wal.stat	Statystyki WAL.	
redis.config	Pobiera parametry konfiguracji instancji Redis pasujące do wzorca.	Redis
redis.info	Pobiera wynik polecenia INFO.	
redis.ping	Sprawdza, czy połączenie jest aktywne.	
redis.slowlog.count	Liczba wpisów w dzienniku wolnych operacji od uruchomienia Redis.	
smart.attribute.discovery	Zwraca listę atrybutów urządzeń S.M.A.R.T.	S.M.A.R.T.
smart.disk.discovery	Zwraca listę urządzeń S.M.A.R.T.	
smart.disk.get	Zwraca wszystkie dostępne właściwości urządzeń S.M.A.R.T.	
systemd.unit.get	Zwraca wszystkie właściwości jednostki systemd.	Systemd
systemd.unit.info	Informacje o jednostce systemd.	
systemd.unit.discovery	Lista jednostek systemd i ich szczegółów.	
web.certificate.get	Weryfikuje certyfikaty i zwraca szczegóły certyfikatu.	Web certificates

Zobacz także:

- Wbudowane wtyczki
- Ładowalne wtyczki

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

ceph.df.details[connString,<user>,<apikey>]

 Wykorzystanie danych klastra i ich dystrybucja pomiędzy pulami.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.stats[connString,<user>,<apikey>]

 Statystyki zagregowane i dla poszczególnych OSD.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.discovery[connString,<user>,<apikey>]

 Lista wykrytych OSD. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.dump[connString,<user>,<apikey>]

 Progi wykorzystania i statusy OSD.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.ping[connString,<user>,<apikey>]

 Sprawdza, czy można nawiązać połączenie z Ceph.
 Wartość zwracana: *0* - połączenie jest przerwane (jeśli występuje jakikolwiek błąd, w tym problemy z AUTH i konfiguracją); *1* - połączenie zostało nawiązane pomyślnie.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do łączenia z Ceph.

ceph.pool.discovery[connString,<user>,<apikey>]

 Lista wykrytych puli. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.status[connString,<user>,<apikey>]

 Ogólny status klastra.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

docker.container_info[<ID>,<info>]

 Informacje niskiego poziomu o kontenerze.
 Zwracana wartość: Dane wyjściowe wywołania API ContainerInspect zserializowane jako JSON.

Parametry:

- **ID** - identyfikator lub nazwa kontenera;
- **info** - ilość zwracanych informacji. Obsługiwane wartości: *short* (domyślnie) lub *full*.

Użytkownik Agent 2 (’zabbix’) musi zostać dodany do grupy ’docker’, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.container_stats[<ID>]

 Statystyki użycia zasobów kontenera.
 Wartość zwracana: dane wyjściowe wywołania API ContainerStats oraz procent użycia CPU zserializowane jako JSON.

Parametr:

- **ID** — identyfikator lub nazwa kontenera.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby mieć wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.containers

 Lista kontenerów.
 Zwracana wartość: Dane wyjściowe wywołania API ContainerList zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do group „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.containers.discovery[<options>]

 Zwraca listę kontenerów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **options** — określa, czy mają zostać wykryte wszystkie kontenery, czy tylko uruchomione. Obsługiwane wartości: *true* — zwraca wszystkie kontenery; *false* — zwraca tylko uruchomione kontenery (domyślnie).

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.data.usage

 Informacje o bieżącym użyciu danych.
 Zwracana wartość: Dane wyjściowe wywołania API SystemDataUsage zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do group „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.images

 Zwraca listę obrazów.
 Wartość zwracana: dane wyjściowe wywołania API ImageList zserializowane do formatu JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.images.discovery

 Zwraca listę obrazów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.info

 Informacje o systemie.
 Wartość zwracana: Dane wyjściowe wywołania API SystemInfo zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.ping

 Sprawdza, czy demon Docker działa, czy nie.
 Zwracana wartość: *1* - połączenie działa; *0* - połączenie jest zerwane.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby mieć wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

ember.get[<uri>,<path>]

 Zwraca wynik wymaganego urządzenia.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI urządzenia Ember+. Domyślnie: 127.0.0.1:9998;
- **path** - ścieżka OID do urządzenia. Domyślnie pusta, zwraca dane kolekcji głównej.

memcached.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne.
 Zwracana wartość: *1* — połączenie jest aktywne; *0* — połączenie jest zerwane (jeśli wystąpi dowolny błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do Memcached.

memcached.stats[connString,<user>,<password>,<type>]

 Pobiera dane wyjściowe polecenia STATS.
 Wartość zwracana: *JSON* - dane wyjściowe są serializowane do formatu JSON.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do Memcached;
- **type** - typ statystyk do zwrócenia: *items*, *sizes*, *slabs* lub *settings* (domyślnie pusty, zwraca statystyki ogólne).

mongodb.collection.stats[connString,<user>,<password>,<database>,collection]

 Zwraca różne statystyki pamięci masowej dla danej kolekcji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB;
- **database** - nazwa bazy danych (domyślnie: admin);
- **collection** - nazwa kolekcji.

mongodb.collections.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych kolekcji. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.collections.usage[connString,<user>,<password>]

 Zwraca statystyki użycia dla kolekcji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.connpool.stats[connString,<user>,<password>]

 Zwraca informacje dotyczące otwartych połączeń wychodzących z bieżącej instancji bazy danych do innych członków klastra shardingu lub zestawu replik.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.db.stats[connString,<user>,<password>,<database>]

 Zwraca statystyki odzwierciedlające stan danego systemu bazy danych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB;
- **database** - nazwa bazy danych (domyślnie: admin).

mongodb.db.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych baz danych. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.jumbo_chunks.count[connString,<user>,<password>]

 Zwraca liczbę jumbo chunks.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.oplog.stats[connString,<user>,<password>]

 Zwraca stan zestawu replik, używając danych pobranych z oplog.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne, czy nie.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.rs.config[connString,<user>,<password>]

 Zwraca bieżącą konfigurację zestawu replik.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.rs.status[connString,<user>,<password>]

 Zwraca status zestawu replik z punktu widzenia członka, na którym uruchomiono tę metodę.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.server.status[connString,<user>,<password>]

 Zwraca stan bazy danych.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.sh.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych shardów obecnych w klastrze.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.version[connString,<user>,<password>]

 Zwraca wersję serwera bazy danych.
 Wartość zwracana: *String*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mqtt.get[<broker url>,topic,<user>,<password>]

 Subskrybuje określony temat lub tematy (z symbolami wieloznacznymi) podanego brokera i oczekuje na publikacje.
 Wartość zwracana: Zależy od zawartości tematu. Jeśli używane są symbole wieloznaczne, zwracana jest zawartość tematu w formacie JSON.

Parametry:

- **broker url** - URL brokera MQTT w formacie `protocol://host:port` bez parametrów zapytania (obsługiwane protokoły: `tcp`, `ssl`, `ws`). Jeśli nie podano wartości, agent użyje `tcp://localhost:1883`. Jeśli protokół lub port zostaną pominięte, zostanie użyty domyślny protokół (`tcp`) lub port (1883);
- **topic** - temat MQTT (wymagany). Obsługiwane są symbole wieloznaczne (+,#);
- **user, password** - dane uwierzytelniające (jeśli są wymagane).

Uwagi:

- Pozycja musi być skonfigurowana jako sprawdzenie aktywne (typ pozycji „Zabbix agent (active)”);
- Certyfikaty szyfrowania TLS mogą być używane po zapisaniu ich w domyślnej lokalizacji (np. katalog `/etc/ssl/certs/` w Ubuntu). Dla TLS użyj schematu `tls://`.

mssql.availability.group.get[URI,<user>,<password>]

 Zwraca grupy dostępności.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **URI** - URI serwera MSSQL (jedynym obsługiwanym schematem jest `sqlserver://`). Osadzone poświadczenia zostaną zignorowane. Można określić nazwę instancji jako część URI, np.: `sqlserver://localhost/InstanceName` (bez portu). Jeśli port jest określony, nazwa instancji jest ignorowana;
- **user, password** - nazwa użytkownika, hasło do wysłania do chronionego serwera MSSQL.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.custom.query[URI,<user>,<password>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **URI** - URI serwera MSSQL (jedynym obsługiwanym schematem jest `sqlserver://`). Osadzone poświadczenia zostaną zignorowane. Można określić nazwę instancji jako część URI, tj.: `sqlserver://localhost/InstanceName` (bez portu). Jeśli port jest określony, nazwa instancji jest ignorowana;
- **user, password** - nazwa użytkownika, hasło do wysłania do chronionego serwera MSSQL;
- **queryName** - nazwa niestandardowego zapytania skonfigurowanego w `Plugins.MSSQL.CustomQueriesDir` bez rozszerzenia .sql;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.db.get

 Zwraca wszystkie dostępne bazy danych MSSQL.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.job.status.get

 Zwraca status zadań.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.last.backup.get

 Zwraca czas ostatniej kopii zapasowej dla wszystkich baz danych.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.local.db.get

 Zwraca bazy danych, które uczestniczą w grupie dostępności Always On i replikacji (głównej lub pomocniczej) oraz znajdują się na serwerze, z którym nawiązano połączenie.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme MSSQL plugin.

mssql.mirroring.get

 Zwraca informacje o mirroringu.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.nonlocal.db.get

 Zwraca bazy danych, które uczestniczą w grupie dostępności Always On oraz replikę (podstawową lub pomocniczą) znajdującą się na innych serwerach (baza danych nie jest lokalna względem wystąpienia SQL Server, z którym nawiązano połączenie).
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.perfcounter.get

 Zwraca liczniki wydajności.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.ping

 Sprawdza dostępność bazy danych. Testuje, czy połączenie jest poprawnie skonfigurowane.
 Zwracana wartość: *1* - aktywne, *0* - nieaktywne.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.quorum.get

 Zwraca informacje o quorum.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.quorum.member.get

 Zwraca członków kworum.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.replica.get

 Zwraca repliki.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.version

 Zwraca wersję MSSQL.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mysql.custom.query[connString,<user>,<password>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MySQL;

- **queryName** - nazwa niestandardowego zapytania, musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Więcej informacji można znaleźć w pliku readme wtyczki MySQL, w sekcji *Custom queries*.

mysql.db.discovery[connString,<user>,<password>]

 Zwraca listę baz danych MySQL. Używane w odkrywaniu niskiego poziomu.
 Wartość zwracana: wynik zapytania SQL „show databases” w formacie JSON LLD.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MySQL.

mysql.db.size[connString,<user>,<password>,<database name>]

 Rozmiar bazy danych w bajtach.
 Zwracana wartość: wynik zapytania SQL „select coalesce(sum(data_length + index_length),0) as size from information_schema.tables where table_schema=?” dla określonej bazy danych, w bajtach.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania MySQL;
- **database name** — nazwa bazy danych.

mysql.get_status_variables[connString,<user>,<password>]

 Wartości globalnych zmiennych statusu.
 Wartość zwracana: Wynik zapytania SQL „show global status” w formacie JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania MySQL.

mysql.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MySQL.

mysql.replication.discovery[connString,<user>,<password>]

 Zwraca listę replikacji MySQL. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: wynik zapytania SQL `SHOW SLAVE STATUS` lub `SHOW REPLICA STATUS` w formacie LLD JSON.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania MySQL.

mysql.replication.get_slave_status[connString,<user>,<password>,<master host>]

 Status replikacji.
 Zwracana wartość: wynik zapytania SQL `SHOW SLAVE STATUS` lub `SHOW REPLICA STATUS` w formacie JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MySQL;
- **master host** — nazwa hosta głównego replikacji. Jeśli nie znaleziono żadnego, zwracany jest błąd. Jeśli ten parametr nie jest określony, zwracane są wszystkie hosty.

mysql.version[connString,<user>,<password>]

 Wersja MySQL.
 Zwracana wartość: *String* (z wersją instancji MySQL).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MySQL.

net.dns.get[<ip>,name,<type>,<timeout>,<count>,<protocol>,"<flags>"]

Wykonuje zapytanie DNS i zwraca szczegółowe informacje o rekordzie DNS.
 Ta pozycja jest rozszerzoną wersją pozycji agenta Zabbix `net.dns.record` z obsługą większej liczby typów rekordów i konfigurowalnych flag.
 Zwracane wartości: *obiekt JSON*

Parametry:

- **ip** - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS do zapytania;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie *SOA*);
- **timeout** - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*;
- **flags** - jeden lub więcej argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Uwagi:

- Możliwe wartości dla `type` to: *A*, *NS*, *MD*, *MF*, *CNAME*, *SOA*, *MB*, *MG*, *MR*, *NULL*, *PTR*, *HINFO*, *MINFO*, *MX*, *TXT*, *RP*, *AFSDB*, *X25*, *ISDN*, *RT*, *NSAPPTR*, *SIG*, *KEY*, *PX*, *GPOS*, *AAAA*, *LOC*, *NXT*, *EID*, *NIMLOC*, *SRV*, *ATMA*, *NAPTR*, *KX*, *CERT*, *DNAME*, *OPT*, *APL*, *DS*, *SSHFP*, *IPSECKEY*, *RRSIG*, *NSEC*, *DNSKEY*, *DHCID*, *NSEC3*, *NSEC3PARAM*, *TLSA*, *SMIMEA*, *HIP*, *NINFO*, *RKEY*, *TALINK*, *CDS*, *CDNSKEY*, *OPENPGPKEY*, *CSYNC*, *ZONEMD*, *SVCB*, *HTTPS*, *SPF*, *UINFO*, *UID*, *GID*, *UNSPEC*, *NID*, *L32*, *L64*, *LP*, *EUI48*, *EUI64*, *URI*, *CAA*, *AVC*, *AMTRELAY*. Zwróć uwagę, że wartości muszą być podane wyłącznie wielkimi literami; małe litery lub mieszana wielkość liter nie są obsługiwane.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawiony na *PTR*), można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (patrz przykłady poniżej). Zwróć uwagę, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Możliwe wartości dla `flags` to: *cdflag* lub *nocdflag* (domyślnie), *rdflag* (domyślnie) lub *nordflag*, *dnssec* lub *nodnssec* (domyślnie), *nsid* lub *nonsid* (domyślnie), *edns0* (domyślnie) lub *noedns0*, *aaflag* lub *noaaflag* (domyślnie), *adflag* lub *noadflag* (domyślnie). Flag `dnssec` i `nsid` nie mogą być używane razem z *noedns0*, ponieważ obie wymagają *edns0*. Zwróć uwagę, że wartości muszą być podane wyłącznie małymi literami; wielkie litery lub mieszana wielkość liter nie są obsługiwane.
- Zinternacjonalizowane nazwy domen nie są obsługiwane, użyj zamiast tego nazw zakodowanych w IDNA.
- Wynik jest obiektem zawierającym informacje o rekordzie DNS na podstawie podanych parametrów (zobacz więcej szczegółów).

Przykłady:

```
net.dns.get[192.0.2.1,example.com,DNSKEY,3,3,tcp,"cdflag,rdflag,nsid"]

net.dns.get[,198.51.100.1,PTR,,,,"cdflag,rdflag,nsid"]
net.dns.get[,1.100.51.198.in-addr.arpa,PTR,,,,"cdflag,rdflag,nsid"]

net.dns.get[,2a00:1450:400f:800::200e,PTR,,,,"cdflag,rdflag,nsid"]
net.dns.get[,e.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR,,,,"cdflag,rdfla
```

nvml.device.count

 Liczba urządzeń GPU.
 Wartość zwracana: *Integer*.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.decoder.utilization[<deviceUUID>]

 Wykorzystanie dekodera urządzenia GPU w procentach.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.ecc.mode[<deviceUUID>]

 Informacje o trybie ECC urządzenia GPU (bieżący, oczekujący).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.encoder.stats.get[<deviceUUID>]

 Statystyki enkodera urządzenia GPU.
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.encoder.utilization[<deviceUUID>]

 Wykorzystanie enkodera urządzenia GPU w procentach.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.energy.consumption[<deviceUUID>]

 Łączne zużycie energii przez urządzenie GPU w milidżulach (mJ) od momentu ostatniego ponownego załadowania sterownika.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.errors.memory[<deviceUUID>]

 Informacje o błędach pamięci ECC urządzenia GPU (skorygowane, nieskorygowane).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.errors.register[<deviceUUID>]

 Informacje o błędach rejestru ECC urządzenia GPU (skorygowane, nieskorygowane).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.fan.speed.avg[<deviceUUID>]

 Średnia prędkość wentylatora urządzenia GPU wyrażona jako procent maksymalnej prędkości.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.get

 Wykryte GPU z UUID i nazwą.
 Zwracana wartość: *JSON*.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.graphics.frequency[<deviceUUID>]

 Prędkość zegara grafiki urządzenia GPU w MHz.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.memory.fb.get[<deviceUUID>]

 Statystyki pamięci framebuffer urządzenia GPU (całkowita, zarezerwowana, wolna, używana).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.memory.bar1.get[<deviceUUID>]

 Statystyki pamięci BAR1 urządzenia GPU (całkowita, wolna, używana).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.memory.frequency[<deviceUUID>]

 Prędkość zegara pamięci urządzenia GPU w MHz.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.pci.utilization[<deviceUUID>]

 Wykorzystanie PCI urządzenia GPU (przepustowość wysyłania/odbierania w KB/s).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.performance.state[<deviceUUID>]

 Stan wydajności urządzenia GPU.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.power.limit[<deviceUUID>]

 Limit mocy urządzenia GPU w miliwatach.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.power.usage[<deviceUUID>]

 Aktualne zużycie mocy urządzenia GPU w miliwatach.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.serial[<deviceUUID>]

 Numer seryjny urządzenia GPU.
 Zwracana wartość: *String*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.sm.frequency[<deviceUUID>]

 Częstotliwość zegara multiprocesora strumieniowego urządzenia GPU w MHz.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.temperature[<deviceUUID>]

 Temperatura urządzenia GPU w stopniach Celsjusza.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.utilization[<deviceUUID>]

 Statystyki wykorzystania urządzenia GPU (wykorzystanie GPU/pamięci w procentach).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.device.video.frequency[<deviceUUID>]

 Prędkość zegara wideo urządzenia GPU w MHz.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.system.driver.version

 Wersja sterownika NVIDIA.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

nvml.version

 Wersja biblioteki NVML.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme NVIDIA GPU plugin.

oracle.diskgroups.stats[connString,<user>,<password>,<service>,<diskgroup>]

 Zwraca statystyki grup dysków Automatic Storage Management (ASM).
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracyjnym pluginu;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu ”(” (z pominięciem początkowych spacji). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **diskgroup** - nazwa grupy dysków ASM, dla której mają zostać pobrane dane.

oracle.diskgroups.discovery[connString,<user>,<password>,<service>]

 Zwraca listę grup dysków ASM. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;

    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania jest nieczuła na wielkość liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.archive.info[connString,<user>,<password>,<service>,<destination>]

 Statystyki dzienników archiwum.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **destination** - nazwa miejsca docelowego do odpytywania.

oracle.cdb.info[connString,<user>,<password>,<service>,<database>]

 Informacje o bazach danych kontenerowych (CDB).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **database** - nazwa bazy danych, o którą ma zostać wykonane zapytanie.

oracle.custom.query[connString,<user>,<password>,<service>,queryName,<args...>]

 Wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;

    - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie mają zastosowania żadne z powyższych warunków;
    - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **queryName** - nazwa niestandardowego zapytania, musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Uwagi: - Zwrócone dane są automatycznie konwertowane do JSON. - Unikaj zwracania JSON bezpośrednio z zapytań, ponieważ zostanie on uszkodzony, gdy wtyczka spróbuje ponownie przekonwertować go do JSON.

oracle.datafiles.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki plików danych.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione sesji o podanej nazwie i `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.db.discovery[connString,<user>,<password>,<service>]

 Zwraca listę baz danych. Używane do low-level discovery.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie sesji o podanej nazwie i `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.fra.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki Fast Recovery Area (FRA).
 Wartość zwracana: *obiekt JSON*.

Parametry: - **connString** - może to być:
 - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
 - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
 - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
 - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli

zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
 - **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
 - **password** - hasło Oracle;
 - **service** - nazwa usługi Oracle.

oracle.instance.info[connString,<user>,<password>,<service>]

 Statystyki instancji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione połączenie sesji o podanej nazwie i `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.pdb.info[connString,<user>,<password>,<service>,<database>]

 Informacje o bazach danych Pluggable Databases (PDB).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **database** - nazwa bazy danych do odpytywania.

oracle.pdb.discovery[connString,<user>,<password>,<service>]

 Zwraca listę PDB. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne sesja o podanej nazwie i `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości

liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.pga.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki Program Global Area (PGA).
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji pluginu;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego ”(” (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.ping[connString,<user>,<password>,<service>]

 Sprawdza, czy można nawiązać połączenie z Oracle.
 Wartość zwracana: *1* - połączenie zostało nawiązane pomyślnie; *0* - połączenie zostało przerwane (jeśli wystąpił jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego ”(” (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania jest nieczuła na wielkość liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

Przykłady:

```
oracle.ping[tcp://127.0.0.1:1521,ZABBIX_MON,zabbix,xe]
oracle.ping[localhost,ZABBIX_MON,zabbix,xe]
oracle.ping[zbx_tns_example,ZABBIX_MON,zabbix,xe]
oracle.ping["(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=x
```

oracle.proc.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki procesów.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie mają zastosowania żadne z powyższych warunków;

    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.redolog.info[connString,<user>,<password>,<service>]

 Informacje o pliku dziennika z pliku kontrolnego.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. `1521`) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dodanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.sga.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki System Global Area (SGA).
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie i `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. `1521`) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
    - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.sessions.stats[connString,<user>,<password>,<service>,<lockMaxTime>]

 Zwraca statystyki sesji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. `1521`) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.

- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **lockMaxTime** - maksymalny czas blokady sesji w sekundach, po którym sesja jest liczona jako długotrwale zablokowana. Domyślnie: 600 sekund.

oracle.sys.metrics[connString,<user>,<password>,<service>,<duration>]

 Zwraca zestaw wartości metryk systemowych.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i w każdym przypadku będzie traktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu ”(” (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **duration** - interwał przechwytywania (w sekundach) wartości metryk systemowych. Możliwe wartości: *60* — długi czas trwania (domyślnie), *15* — krótki czas trwania.

oracle.sys.params[connString,<user>,<password>,<service>]

 Zwraca zestaw wartości parametrów systemowych.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie mają zastosowania żadne z powyższych warunków;
  - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego ”(” (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania jest nieczuła na wielkość liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.ts.stats[connString,<user>,<password>,<service>,<tablespace>,<type>,<conname>]

 Zwraca statystyki tablespace’ów.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;

    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **tablespace** - nazwa tablespace'u, dla którego ma zostać wykonane zapytanie. Domyślnie (jeśli pozostanie puste i ustawiono `type`):
    - "TEMP" (jeśli `type` ma wartość "TEMPORARY");
    - "USERS" (jeśli `type` ma wartość "PERMANENT").
- **type** - typ tablespace'u, dla którego ma zostać wykonane zapytanie. Domyślnie (jeśli ustawiono `tablespace`): "PERMANENT".
- **conname** - nazwa kontenera, dla którego są wymagane informacje.

Jeśli `tablespace`, `type` lub `conname` zostanie pominięty, pozycja zwróci statystyki tablespace'ów dla wszystkich pasujących kontenerów (w tym PDB i CDB).

oracle.ts.discovery[connString,<user>,<password>,<service>]

 Zwraca listę tablespace'ów. Używane do low-level discovery.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione żadne połączenie sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i i tak zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwane od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwane od Zabbix 7.4.3) - jeśli zaczyna się od otwartego nawiasu "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.user.info[connString,<user>,<password>,<service>,<username>]

 Zwraca informacje o użytkowniku Oracle.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie zostanie znalezione sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma ustawioną wartość false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
    - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
    - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma ustawioną wartość true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego "(" (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane były tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **username** - nazwa użytkownika, dla którego potrzebne są informacje. Nazwy użytkowników zapisane małymi literami nie są obsługiwane. Domyślnie: bieżący użytkownik.

oracle.version[connString,<user>,<password>,<service>]

 Zwraca wersję serwera bazy danych.
 Wartość zwracana: *String*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie zostanie znalezione żadne połączenie o podanej nazwie, a `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowany jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracji wtyczki;
  - klucz TNS (obsługiwany od Zabbix 7.4.3) - jeśli opcja ResolveTNS ma wartość true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS (obsługiwana od Zabbix 7.4.3) - jeśli zaczyna się od nawiasu otwierającego ”(” (wiodące spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle, obsługuje dopisanie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji). Należy pamiętać, że przed Zabbix 7.4.2 obsługiwane są tylko opcje logowania `as sysdba`, `as sysoper` lub `as sysasm`;
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

pgsql.autovacuum.count[uri,<username>,<password>,<database name>]

 Liczba procesów autovacuum.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.archive[uri,<username>,<password>,<database name>]

 Informacje o zarchiwizowanych plikach.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.bgwriter[uri,<username>,<password>,<database name>]

 Łączna liczba checkpointów dla klastra bazy danych, z podziałem według typu checkpointu.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.cache.hit[uri,<username>,<password>,<database name>]

 Współczynnik trafień w buforze PostgreSQL.
 Wartość zwracana: *Float*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.connections[uri,<username>,<password>,<database name>]

 Zwraca połączenia według typu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.custom.query[uri,<username>,<password>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **queryName** - nazwa niestandardowego zapytania; musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów oddzielonych przecinkami, przekazywanych do zapytania.

pgsql.db.age[uri,<username>,<password>,<database name>]

 Wiek najstarszego FrozenXID w bazie danych.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.db.bloating_tables[uri,<username>,<password>,<database name>]

 Liczba bloating tables w bazie danych.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.db.discovery[uri,<username>,<password>,<database name>]

 Lista baz danych PostgreSQL. Używane do odkrywania niskopoziomowego.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.db.size[uri,<username>,<password>,<database name>]

 Rozmiar bazy danych w bajtach.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.dbstat[uri,<username>,<password>,<database name>]

 Zbiera statystyki dla każdej bazy danych. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.dbstat.sum[uri,<username>,<password>,<database name>]

 Zestawione dane dla wszystkich baz danych w klastrze.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.locks[uri,<username>,<password>,<database name>]

 Informacje o przyznanych blokadach dla każdej bazy danych. Używane do low-level discovery.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.oldest.xid[uri,<username>,<password>,<database name>]

 Wiek najstarszego XID.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.ping[uri,<username>,<password>,<database name>]

 Sprawdza, czy połączenie jest aktywne, czy nie.
 Wartość zwracana: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpił jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.queries[uri,<username>,<password>,<database name>,time period]

 Zwraca metryki zapytań według czasu wykonania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf);
- **time period** - limit czasu wykonania dla liczby wolnych zapytań (musi być dodatnią liczbą całkowitą).

pgsql.replication.count[uri,<username>,<password>]

 Liczba serwerów standby.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL.

pgsql.replication.process[uri,<username>,<password>]

 Opóźnienie flush, opóźnienie zapisu i opóźnienie odtwarzania dla każdego procesu nadawcy.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication.process.discovery[uri,<username>,<password>]

 Odkrywanie nazw procesów replikacji.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL.

pgsql.replication.recovery_role[uri,<username>,<password>]

 Status odzyskiwania.
 Zwracana wartość: *0* - tryb master; *1* - odzyskiwanie jest nadal w toku (tryb standby).

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication.status[uri,<username>,<password>]

 Status replikacji.
 Zwracana wartość: *0* - streaming jest wyłączony; *1* - streaming jest włączony; *2* - tryb master.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication_lag.b[uri,<username>,<password>]

 Opóźnienie replikacji w bajtach.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL.

pgsql.replication_lag.sec[uri,<username>,<password>]

 Opóźnienie replikacji w sekundach.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.uptime[uri,<username>,<password>,<database name>]

 Czas działania PostgreSQL w milisekundach.
 Wartość zwracana: *Float*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.version[uri,<username>,<password>,<database name>]

 Zwraca wersję PostgreSQL.
 Wartość zwracana: *String*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

pgsql.wal.stat[uri,<username>,<password>,<database name>]

 Statystyki WAL.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf).

redis.config[connString,<password>,<pattern>]

 Pobiera parametry konfiguracyjne instancji Redis pasujące do wzorca.
 Zwracana wartość: *JSON* - jeśli użyto wzorca w stylu glob; pojedyncza wartość - jeśli wzorzec nie zawierał żadnego znaku wieloznacznego.

Parametry:

- **connString** - URI lub nazwa sesji;
- **password** - hasło Redis;

- **pattern** - wzorzec w stylu glob (domyślnie *).

Komentarz:

- Ostrzeżenie dotyczące bezpieczeństwa: `redis.config` uruchamia polecenie Redis `CONFIG GET` i może zwracać wrażliwe parametry konfiguracyjne (na przykład `requirepass`, `masterauth`, ustawienia związane z TLS oraz inne sekrety). Użycie domyślnego wzorca (`*`) lub szerokich wzorców może ujawnić te sekrety każdemu, kto ma uprawnienia do przeglądania wartości pozycji w Zabbix.

redis.info[connString,<password>,<section>]

 Pobiera dane wyjściowe polecenia INFO.
 Wartość zwracana: *JSON* — dane wyjściowe są serializowane do formatu JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **password** — hasło Redis;
- **section** — sekcja informacji (domyślnie *default*).

redis.ping[connString,<password>]

 Sprawdza, czy połączenie jest aktywne, czy nie.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **password** - hasło Redis.

redis.slowlog.count[connString,<password>]

 Liczba wpisów w logu wolnych operacji od momentu uruchomienia Redis.
 Zwracana wartość: *Integer*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **password** — hasło Redis.

smart.attribute.discovery

 Zwraca listę atrybutów urządzenia S.M.A.R.T.
 Zwracana wartość: *obiekt JSON*.

Uwagi:

- Zwracane są następujące makra i ich wartości: {#NAME}, {#DISKTYPE}, {#ID}, {#ATTRNAME}, {#THRESH};
- Obsługiwane są typy dysków HDD, SSD i NVME. Dyski mogą występować samodzielnie lub być połączone w RAID. W przypadku RAID do {#NAME} zostanie dodany dodatkowy element, np.: {"{#NAME}": "/dev/sda cciss,2"}.

smart.disk.discovery[<type>]

 Zwraca listę urządzeń S.M.A.R.T.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **type** - określa wartość, według której mają być skanowane dyski. Możliwe wartości: *id*, *name* (domyślnie). *id* nie jest obsługiwane w systemie Windows i po wybraniu tej wartości zostanie zwrócony błąd

Uwagi:

- Zwracane są następujące makra i ich wartości: {#NAME}, {#DISKTYPE}, {#MODEL}, {#SN}, {#PATH}, {#ATTRIBUTES}, {#RAIDTYPE};
- Obsługiwane są typy dysków HDD, SSD i NVME. Jeśli dysk nie należy do RAID, {#RAIDTYPE} będzie puste. {#NAME} będzie zawierać dodatkowy element w przypadku RAID, np.: {"{#NAME}": "/dev/sda cciss,2"}.

smart.disk.get[<path>,<raid type>]

 Zwraca wszystkie dostępne właściwości urządzeń S.M.A.R.T.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **path** - ścieżka dysku; jako wartości można użyć makra {#PATH};
- **raid_type** - typ RAID; jako wartości można użyć makra {#RAID}

Uwagi:

- Obsługiwane są dyski typu HDD, SSD i NVME. Dyski mogą działać samodzielnie lub być połączone w RAID;

- Dane obejmują wersję smartctl i argumenty wywołania oraz dodatkowe pola:
disk_name - zawiera nazwę z wymaganymi dodatkami do wykrywania RAID, np.: {"disk_name": "/dev/sda cciss,2"}
disk_type - zawiera typ dysku: HDD, SSD lub NVME, np.: {"disk_type": "ssd"};
- Jeśli nie określono parametrów, pozycja zwróci informacje o wszystkich dyskach.

systemd.unit.get[nazwa jednostki,<interface>]

 Zwraca wszystkie właściwości jednostki systemd.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **nazwa jednostki** - nazwa jednostki (możesz chcieć użyć makra {#UNIT.NAME} w prototypie pozycji, aby wykryć nazwę);
- **interface** - typ interfejsu jednostki, możliwe wartości: *Unit* (domyślnie), *Service*, *Socket*, *Device*, *Mount*, *Automount*, *Swap*, *Target*, *Path*.

Komentarze:

- Ta pozycja jest obsługiwana tylko na platformie Linux;
- LoadState, ActiveState i UnitFileState dla interfejsu Unit są zwracane jako tekst i liczba całkowita: `"ActiveState":{"state":1,"tex`

systemd.unit.info[nazwa jednostki,<właściwość>,<interfejs>]

 Informacja o jednostce systemd.
 Zwracana wartość: *String*.

Parametry:

- **nazwa jednostki** - nazwa jednostki (możesz chcieć użyć makra {#UNIT.NAME} w prototypie pozycji, aby wykryć nazwę);
- **właściwość** - właściwość jednostki (np. ActiveState (domyślnie), LoadState, Description);
- **interfejs** - typ interfejsu jednostki (np. Unit (domyślnie), Socket, Service).

Uwagi:

- Ta pozycja jest obsługiwana tylko na platformie Linux;
- Ta pozycja umożliwia pobranie określonej właściwości z określonego typu interfejsu, jak opisano w dbus API.

Przykłady:

```
systemd.unit.info["{#UNIT.NAME}"] #zbiera informacje o stanie aktywności (active, reloading, inactive, fai
systemd.unit.info["{#UNIT.NAME}",LoadState] #zbiera informacje o stanie załadowania wykrytych jednostek sy
systemd.unit.info[mysqld.service,Id] #pobiera techniczną nazwę usługi (mysqld.service)
systemd.unit.info[mysqld.service,Description] #pobiera opis usługi (MySQL Server)
systemd.unit.info[mysqld.service,ActiveEnterTimestamp] #pobiera ostatni czas przejścia usługi do stanu act
systemd.unit.info[dbus.socket,NConnections,Socket] #zbiera liczbę połączeń z tej jednostki gniazda
```

systemd.unit.discovery[<type>]

 Lista jednostek systemd i ich szczegółów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **type** — możliwe wartości: *all*, *automount*, *device*, *mount*, *path*, *service* (domyślnie), *socket*, *swap*, *target*.

Ta pozycja jest obsługiwana tylko na platformie Linux.

web.certificate.get[hostname,<port>,<address>]

 Weryfikuje certyfikaty i zwraca szczegóły certyfikatu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **hostname** — może być adresem IP lub nazwą DNS.
Może zawierać schemat URL (*tylko https*), ścieżkę (zostanie zignorowana) oraz port.
Jeśli port zostanie podany zarówno w pierwszym, jak i w drugim parametrze, ich wartości muszą być zgodne.
Jeśli zostanie określony address (3. parametr), hostname jest używany tylko do SNI i weryfikacji nazwy hosta;
- **port** — numer portu (domyślnie 443 dla HTTPS);
- **address** — może być adresem IP lub nazwą DNS. Jeśli zostanie określony, będzie używany do połączenia, a hostname (1. parametr) będzie używany do SNI i weryfikacji hosta. W przypadku gdy 1. parametr jest adresem IP, a 3. parametr jest nazwą DNS, 1. parametr będzie używany do połączenia, a 3. parametr będzie używany do SNI i weryfikacji hosta.

Uwagi:

- Ta pozycja staje się nieobsługiwana, jeśli miejsce docelowe określone w konfiguracji hosta nie istnieje, jest niedostępne lub jeśli uzgadnianie TLS zakończy się błędem innym niż nieprawidłowy certyfikat;

- Obecnie rozszerzenie X.509 AIA (Authority Information Access), listy CRL i OCSP (w tym OCSP stapling) oraz Certificate Transparency nie są obsługiwane;
- Pola odpowiedzi JSON:
  - *x509*: zawiera szczegóły certyfikatu X.509.
    - *version*: wersja X.509 (np. 3).
    - *serial_number*: numer seryjny certyfikatu.
    - *signature_algorithm*: algorytm użyty do podpisania certyfikatu (np. `SHA256-RSA`).
    - *issuer*: wystawca certyfikatu.
    - *not_before*: data rozpoczęcia ważności certyfikatu.
    - *not_after*: data wygaśnięcia certyfikatu.
    - *subject*: podmiot certyfikatu.
    - *public_key_algorithm*: algorytm użyty dla klucza publicznego (np. `RSA`).
    - *alternative_names*: alternatywne nazwy podmiotu (jeśli występują), w przeciwnym razie `null`.
  - result: zawiera wynik weryfikacji.
    - *value*: status weryfikacji (zobacz możliwe wartości poniżej).
    - *message*: szczegółowy komunikat weryfikacji (np. `"certificate verified successfully"`).
  - *sha1_fingerprint*: odcisk SHA-1 certyfikatu.
  - *sha256_fingerprint*: odcisk SHA-256 certyfikatu.
- Pole `$.result.value` wskazuje wynik weryfikacji certyfikatu. Możliwe wartości obejmują:
  - *valid* — certyfikat jest prawidłowy i zaufany.
  - *valid-but-self-signed* — certyfikat jest prawidłowy, ale samopodpisany, co oznacza, że jego podmiot jest zgodny z jego wystawcą.
  - *invalid* — certyfikat jest nieprawidłowy z powodu problemu, takiego jak wygaśnięcie, nieprawidłowa nazwa hosta lub nieznany urząd podpisujący.

Przykład:

```
web.certificate.get[example.com,443]
```

Odpowiedź JSON:


```
{
  "x509": {
    "version": 3,
    "serial_number": "0ad893bafa68b0b7fb7a404f06ecaf9a",
    "signature_algorithm": "ECDSA-SHA384",
    "issuer": "CN=DigiCert Global G3 TLS ECC SHA384 2020 CA1,O=DigiCert Inc,C=US",
    "not_before": {
      "value": "Jan 15 00:00:00 2025 GMT",
      "timestamp": 1736899200
    },
    "not_after": {
      "value": "Jan 15 23:59:59 2026 GMT",
      "timestamp": 1768521599
    },
    "subject": "CN=*.example.com,O=Internet Corporation for Assigned Names and Numbers,L=Los Angeles,ST=Ca
    "public_key_algorithm": "ECDSA",
    "alternative_names": [
      "*.example.com",
      "example.com"
    ]
  },
  "result": {
    "value": "valid",
    "message": "certificate verified successfully"
  },
  "sha1_fingerprint": "310db7af4b2bc9040c8344701aca08d0c69381e3",
  "sha256_fingerprint": "455943cf819425761d1f950263ebf54755d8d684c25535943976f488bc79d23b"
}
```


2 Windows agent Zabbix

Przegląd

Pozycje agenta Zabbix dla systemu Windows są przedstawione na dwóch listach:

- Pozycje współdzielone — klucze pozycji współdzielone z agentem Zabbix dla systemów UNIX;
- Pozycje specyficzne dla Windows — klucze pozycji obsługiwane **tylko** w systemie Windows.

Należy pamiętać, że wszystkie klucze pozycji obsługiwane przez agent Zabbix w systemie Windows są również obsługiwane przez nowej generacji agent Zabbix 2. Zobacz dodatkowe klucze pozycji, których można używać wyłącznie z agentem 2.

Zobacz także: Minimalne uprawnienia dla pozycji Windows

Współdzielone pozycje

Poniższa tabela zawiera pozycje agenta Zabbix obsługiwane w systemie Windows i współdzielone z agentem Zabbix dla UNIX:

- Klucz pozycji jest łączem do pełnych szczegółów pozycji agenta Zabbix dla UNIX
- Uwzględniono komentarze dotyczące pozycji istotne dla systemu Windows

Item key	Description	Item group
log	Monitorowanie pliku dziennika. Ta pozycja nie jest obsługiwana dla dziennika zdarzeń systemu Windows. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	Monitorowanie dziennika
log.count	Liczba dopasowanych wierszy w monitorowanym pliku dziennika. Ta pozycja nie jest obsługiwana dla dziennika zdarzeń systemu Windows. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
logrt	Monitorowanie rotowanego pliku dziennika. Ta pozycja nie jest obsługiwana dla dziennika zdarzeń systemu Windows. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
logrt.count	Liczba dopasowanych wierszy w monitorowanym rotowanym pliku dziennika. Ta pozycja nie jest obsługiwana dla dziennika zdarzeń systemu Windows. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
modbus.get	Odczytuje dane Modbus.	Modbus
net.dns	Sprawdza, czy usługa DNS działa. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	Network
net.dns.perf	Sprawdza wydajność usługi DNS. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	
net.dns.record	Wykonuje zapytanie DNS. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	
net.if.discovery	Lista interfejsów sieciowych. Niektóre wersje systemu Windows (na przykład Server 2008) mogą wymagać zainstalowania najnowszych aktualizacji, aby obsługiwać znaki niebędące ASCII w nazwach interfejsów.	
net.if.in	Statystyki ruchu przychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	
net.if.out	Statystyki ruchu wychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	

Item key	Description	Item group
net.if.total	Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	
net.tcp.listen	Sprawdza, czy ten port TCP znajduje się w stanie LISTEN.	
net.tcp.port	Sprawdza, czy można nawiązać połączenie TCP z określonym portem.	
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP. Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.	
net.tcp.service.perf	Sprawdza wydajność usługi TCP. Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.	
net.tcp.socket.count	Zwraca liczbę gniazd TCP pasujących do parametrów. Ta pozycja jest obsługiwana w systemie Linux przez agenta Zabbix, ale w systemie Windows jest obsługiwana tylko przez Zabbix agent 2 w 64-bitowym systemie Windows.	
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.	
net.udp.service.perf	Sprawdza wydajność usługi UDP.	
net.udp.socket.count	Zwraca liczbę gniazd UDP pasujących do parametrów. Ta pozycja jest obsługiwana w systemie Linux przez agenta Zabbix, ale w systemie Windows jest obsługiwana tylko przez Zabbix agent 2 w 64-bitowym systemie Windows.	
proc.get	Lista procesów systemu operacyjnego i ich parametrów. Parametr `cmdline` nie jest obsługiwany w systemie Windows.	Processes
proc.num	Liczba procesów. W systemie Windows obsługiwane są tylko parametry `name` i `user`.	
system.cpu.discovery	Lista wykrytych procesorów/rdzeni procesora.	System
system.cpu.load	Obciążenie CPU. Po uruchomieniu procesu kolektora w agencie Zabbix inicjowane są następujące liczniki wydajności, które są później używane przez tę pozycję: `\System\Processor Queue Length`	
system.cpu.num	Liczba procesorów.	
system.cpu.util	Procentowe użycie CPU. Wartość jest pobierana przy użyciu licznika wydajności *Processor Time*. Należy zauważyć, że od Windows 8 Menedżer zadań pokazuje użycie CPU na podstawie licznika wydajności *Processor Utility*, podczas gdy we wcześniejszych wersjach był to licznik *Processor Time* (zobacz więcej szczegółów). *system* jest jedynym parametrem `type` obsługiwanym w systemie Windows.	
system.hostname	Nazwa hosta systemu. Wartość jest pobierana w systemie Windows za pomocą funkcji GetComputerName() (dla **netbios**), GetComputerNameExA() (dla **fqdn**) lub gethostname() (dla **host**). Zobacz także bardziej szczegółowy opis.	
system.localtime	Czas systemowy.	
system.run	Uruchamia określone polecenie na host.	
system.sw.arch	Informacje o architekturze oprogramowania.	
system.sw.os	Informacje o systemie operacyjnym.	
system.sw.os.get	Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja minor i major itp.).	
system.swap.size	Rozmiar przestrzeni wymiany w bajtach lub jako procent całości. Parametr typu `pused` jest obsługiwany w systemie Linux przez agenta Zabbix, ale w systemie Windows jest obsługiwany tylko przez Zabbix agent 2. Należy zauważyć, że ten klucz może zwracać nieprawidłowy rozmiar/procent przestrzeni wymiany na zwirtualizowanych platformach Windows (VMware ESXi, VirtualBox). W takim przypadku można użyć klucza `perf_counter[\700(_Total)\702]`, aby uzyskać prawidłowy procent przestrzeni wymiany.	

Item key	Description	Item group
system.uname	Identyfikacja systemu. W systemie Windows wartość dla tej pozycji jest pobierana z klas WMI Win32_OperatingSystem i Win32_Processor. Nazwa systemu operacyjnego (wraz z edycją) może być przetłumaczona na język wyświetlania użytkownika. W niektórych wersjach systemu Windows zawiera symbole znaków towarowych i dodatkowe spacje.	
system.uptime	Czas działania systemu w sekundach.	
vfs.dir.count	Liczba wpisów katalogu. W systemie Windows dowiązania symboliczne do katalogów są pomijane, a dowiązania twarde są liczone tylko raz.	Virtual file systems
vfs.dir.get	Lista wpisów katalogu. W systemie Windows dowiązania symboliczne do katalogów są pomijane, a dowiązania twarde są liczone tylko raz.	
vfs.dir.size	Rozmiar katalogu. W systemie Windows dowolne dowiązanie symboliczne jest pomijane, a dowiązania twarde są uwzględniane tylko raz.	
vfs.file.cksum	Suma kontrolna pliku, obliczana algorytmem UNIX cksum.	
vfs.file.contents	Pobieranie zawartości pliku.	
vfs.file.exists	Sprawdza, czy plik istnieje. W systemie Windows podwójne cudzysłowy muszą być poprzedzone znakiem ucieczki \\, a cały klucz pozycji musi być ujęty w podwójne cudzysłowy podczas korzystania z narzędzia wiersza poleceń do wywoływania zabbix_get.exe lub agent2. Należy zauważyć, że pozycja może stać się nieobsługiwana w systemie Windows, jeśli katalog jest wyszukiwany w nieistniejącym katalogu, np. `vfs.file.exists[C:\no\dir,dir]` (gdzie 'no' nie istnieje).	
vfs.file.get	Zwraca informacje o pliku. Obsługiwane typy plików w systemie Windows: zwykły plik, katalog, dowiązanie symboliczne	
vfs.file.md5sum	Suma kontrolna MD5 pliku.	
vfs.file.owner	Pobiera właściciela pliku.	
vfs.file.regexp	Pobiera ciąg znaków z pliku.	
vfs.file.regmatch	Wyszukuje ciąg znaków w pliku.	
vfs.file.size	Rozmiar pliku.	
vfs.file.time	Informacje o czasie pliku. W systemie Windows XP `vfs.file.time[file,change]` może być równe `vfs.file.time[file,access]`.	
vfs.fs.discovery	Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania. Makro {#FSLABEL} jest obsługiwane w systemie Windows.	
vfs.fs.get	Lista zamontowanych systemów plików wraz z ich typem, dostępnym miejscem na dysku, statystykami inode i opcjami montowania. Makro {#FSLABEL} jest obsługiwane w systemie Windows.	
vfs.fs.size	Miejsce na dysku w bajtach lub jako procent całości.	
vm.memory.size	Rozmiar pamięci w bajtach lub jako procent całości.	Virtual memory
web.page.get	Pobiera zawartość strony WWW.	Web monitoring
web.page.perf	Czas ładowania pełnej strony WWW.	
web.page.regexp	Wyszukuje ciąg znaków na stronie WWW.	
agent.hostmetadata	Metadane hosta agenta.	Zabbix
agent.hostname	Nazwa hosta agenta.	
agent.ping	Sprawdzenie dostępności agenta.	
agent.variant	Wariant agenta Zabbix (Zabbix agent lub Zabbix agent 2).	
agent.version	Wersja agenta Zabbix.	
zabbix.stats	Zwraca zdalnie zestaw wewnętrznych metryk serwer Zabbix lub proxy.	
zabbix.stats (queue)	Zwraca zdalnie liczbę monitorowanych pozycji w kolejce, które są opóźnione na serwer Zabbix lub proxy.	

Pozycje specyficzne dla systemu Windows

Tabela zawiera szczegółowe informacje o kluczach pozycji, które są obsługiwane **wyłącznie** przez agenta Zabbix dla systemu

Windows.

Pozycje specyficzne dla systemu Windows są czasami przybliżonym odpowiednikiem podobnej pozycji agenta, na przykład `proc_info`, obsługiwana w systemie Windows, w przybliżeniu odpowiada pozycji `proc.mem`, nieobsługiwanej w systemie Windows.

Klucz pozycji jest łączem do pełnych informacji o kluczu pozycji.

Klucz pozycji	Opis	Grupa pozycji
eventlog	Monitorowanie dziennika zdarzeń systemu Windows.	Monitorowanie logów
eventlog.count	Liczba wierszy w dzienniku zdarzeń systemu Windows.	
net.if.list	Lista interfejsów sieciowych (obejmuje typ interfejsu, stan, adres IPv4, opis).	Sieć
perf_counter	Wartość dowolnego licznika wydajności systemu Windows.	Liczniki wydajności
perf_counter_en	Wartość dowolnego licznika wydajności systemu Windows w języku angielskim.	
perf_instance.discovery	Lista instancji obiektów liczników wydajności systemu Windows.	
perf_instance_en.discovery	Lista instancji obiektów liczników wydajności systemu Windows, wykrytych przy użyciu nazw obiektów w języku angielskim.	
proc_info	Różne informacje o określonym procesie lub procesach.	Procesy
registry.data	Zwraca dane dla określonej nazwy wartości w kluczu Rejestru systemu Windows.	Rejestr
registry.get	Lista wartości lub kluczy Rejestru systemu Windows znajdujących się pod podanym kluczem.	
service.discovery	Lista usług systemu Windows.	Usługi
service.info	Informacje o usłudze.	
services	Wykaz usług.	
vm.vmemory.size	Rozmiar pamięci wirtualnej w bajtach lub jako wartość procentowa całkowitej pamięci.	Pamięć wirtualna
wmi.get	Wykonuje zapytanie WMI i zwraca pierwszy wybrany obiekt.	WMI
wmi.getall	Wykonuje zapytanie WMI i zwraca całą odpowiedź.	

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

eventlog[name,<regexp>,<severity>,<source>,<eventid>,<maxlines>,<mode>]

 Monitorowanie dziennika zdarzeń.
 Wartość zwracana: *Log*.

Parametry:

- **name** - nazwa kanału dziennika zdarzeń (*Log Name* w interfejsie GUI Podglądu zdarzeń);
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec (uwzględnia wielkość liter);
- **severity** - wyrażenie regularne opisujące poziom ważności (nie uwzględnia wielkości liter). Ten parametr akceptuje wyrażenie regularne oparte na następujących wartościach: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (uruchomione w Windows Vista lub nowszym).
- **source** - wyrażenie regularne opisujące identyfikator źródła (nie uwzględnia wielkości liter);
- **eventid** - wyrażenie regularne opisujące identyfikator(y) zdarzenia (uwzględnia wielkość liter);
- **maxlines** - maksymalna liczba nowych linii na sekundę, które agent wyśle do serwer Zabbix lub proxy. Ten parametr zastępuje wartość ’MaxLinesPerSecond’ w zabbix_agentd.conf.
- **mode** - możliwe wartości: *all* (domyślnie) lub *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

Komentarze:

- Pozycja musi być skonfigurowana jako active check;
- agent nie jest w stanie wysyłać zdarzeń z dziennika ”Forwarded events”;
- Obsługiwany jest Windows Eventing 6.0;
- Wybranie dla tej pozycji nie-Log type of information spowoduje utratę lokalnego znacznika czasu, a także informacji o ważności dziennika i źródle;
- Zobacz także dodatkowe informacje o log monitoring.

Przykłady:

```
eventlog[Application]
eventlog[Microsoft-Windows-Application-Experience/Program-Compatibility-Assistant]
```

```
eventlog[Security,,"Failure Audit",,^(529|680)$]
eventlog[System,,"Warning|Error"]
eventlog[System,,,,^1$]
eventlog[Windows PowerShell,,,,,,skip]
eventlog[System,,,,@TWOSHORT] #here a custom regular expression named `TWOSHORT` is referenced (defined as
```

eventlog.count[name,<regexp>,<severity>,<source>,<eventid>,<maxproclines>,<mode>]

 Liczba wierszy w dzienniku zdarzeń systemu Windows.
 Wartość zwracana: *Integer*.

Parametry:

- **name** - nazwa kanału dziennika zdarzeń (*Log Name* w interfejsie GUI Podglądu zdarzeń);
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec (uwzględnia wielkość liter);
- **severity** - wyrażenie regularne opisujące poziom ważności (nie uwzględnia wielkości liter). Ten parametr akceptuje wyrażenie regularne oparte na następujących wartościach: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (działa w systemie Windows Vista lub nowszym).
- **source** - wyrażenie regularne opisujące identyfikator źródła (nie uwzględnia wielkości liter);
- **eventid** - wyrażenie regularne opisujące identyfikator(y) zdarzenia (uwzględnia wielkość liter);
- **maxproclines** - maksymalna liczba nowych wierszy na sekundę, które agent będzie analizować (nie może przekroczyć 10000). Wartość domyślna to 10*’MaxLinesPerSecond’ w zabbix_agentd.conf.
- **mode** - możliwe wartości: *all* (domyślnie) lub *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

Komentarze:

- Pozycja musi być skonfigurowana jako active check;
- agent nie jest w stanie odbierać zdarzeń z dziennika ”Forwarded events”;
- Obsługiwany jest Windows Eventing 6.0;
- Wybranie dla tej pozycji innego type of information niż Log spowoduje utratę lokalnego znacznika czasu, a także informacji o ważności i źródle dziennika;
- Zobacz także dodatkowe informacje o monitorowaniu logów.

Przykłady:

```
eventlog.count[System,,"Warning|Error"]
eventlog.count[Windows PowerShell,,,,,,skip]
```

net.if.list

 Lista interfejsów sieciowych (zawiera typ interfejsu, status, adres IPv4, opis).
 Zwracana wartość: *Tekst*.

Uwagi:

- Obsługiwane są nazwy interfejsów wielobajtowych;
- Wyłączone interfejsy nie są wyświetlane;
- Włączanie/wyłączanie niektórych komponentów może zmienić ich kolejność w nazwie interfejsu systemu Windows;
- Niektóre wersje systemu Windows (na przykład Server 2008) mogą wymagać zainstalowania najnowszych aktualizacji, aby obsługiwać znaki spoza ASCII w nazwach interfejsów.

perf_counter[counter,<interval>]

 Wartość dowolnego licznika wydajności systemu Windows.
 Zwracana wartość: *Integer*, *float*, *string* lub *text* (w zależności od żądania).

Parametry:

- **counter** — ścieżka do licznika;
- **interval** — ostatnie N sekund do przechowywania wartości średniej. `interval` musi mieścić się w zakresie od 1 do 900 sekund (włącznie), a wartość domyślna to 1.

Uwagi:

- Ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa;
- `interval` jest używany dla liczników wymagających więcej niż jednej próbki (takich jak wykorzystanie CPU), dlatego sprawdzenie za każdym razem zwraca wartość średnią z ostatnich „interval” sekund;
- Do uzyskania listy dostępnych liczników można użyć Monitora wydajności.
- Zobacz także: Liczniki wydajności systemu Windows.

perf_counter_en[counter,<interval>]

 Wartość dowolnego licznika wydajności systemu Windows w języku angielskim.
 Zwracana wartość: *Integer*, *float*, *string* lub *text* (w zależności od żądania).

Parametry:

- **counter** — ścieżka do licznika w języku angielskim;
- **interval** — ostatnie N sekund do przechowywania wartości średniej. Wartość `interval` musi mieścić się w zakresie od 1 do 900 sekund (włącznie), a wartość domyślna to 1.

Uwagi:

- Ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa;
- `interval` jest używany dla liczników wymagających więcej niż jednej próbki (takich jak użycie CPU), dlatego sprawdzenie za każdym razem zwraca wartość średnią z ostatnich „interval" sekund;
- Ta pozycja jest obsługiwana tylko w systemach **Windows Server 2008/Vista** i nowszych;
- Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

perf_instance.discovery[object]

 Lista instancji obiektów liczników wydajności systemu Windows. Używana do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **object** — nazwa obiektu (zlokalizowana).

perf_instance_en.discovery[object]

 Lista instancji obiektów liczników wydajności systemu Windows, wykrywanych przy użyciu nazw obiektów w języku angielskim. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **object** - nazwa obiektu (w języku angielskim).

proc_info[process,<attribute>,<type>]

 Różne informacje o określonym procesie(-ach).
 Wartość zwracana: *Float*.

Parametry:

- **process** — nazwa procesu;
- **attribute** — żądany atrybut procesu;
- **type** — typ reprezentacji (ma znaczenie, gdy istnieje więcej niż jeden proces o tej samej nazwie)

Uwagi:

- Obsługiwane są następujące `attributes`:
vmsize (domyślnie) — rozmiar pamięci wirtualnej procesu w KB
wkset — rozmiar zestawu roboczego procesu (ilość pamięci fizycznej używanej przez proces) w KB
pf — liczba błędów strony
ktime — czas jądra procesu w milisekundach
utime — czas użytkownika procesu w milisekundach
io_read_b — liczba bajtów odczytanych przez proces podczas operacji wejścia/wyjścia
io_read_op — liczba operacji odczytu wykonanych przez proces
io_write_b — liczba bajtów zapisanych przez proces podczas operacji wejścia/wyjścia
io_write_op — liczba operacji zapisu wykonanych przez proces
io_other_b — liczba bajtów przesłanych przez proces podczas operacji innych niż odczyt i zapis
io_other_op — liczba operacji wejścia/wyjścia wykonanych przez proces, innych niż odczyt i zapis
gdiobj — liczba obiektów GDI używanych przez proces
userobj — liczba obiektów USER używanych przez proces;
- Prawidłowe `types` to:
avg (domyślnie) — wartość średnia dla wszystkich procesów o nazwie <process>
min — wartość minimalna spośród wszystkich procesów o nazwie <process>
max — wartość maksymalna spośród wszystkich procesów o nazwie <process>
sum — suma wartości dla wszystkich procesów o nazwie <process>;
- W systemie 64-bitowym do poprawnego działania tej pozycji wymagany jest 64-bitowy agent Zabbix.

Przykłady:

```
proc_info[iexplore.exe,wkset,sum] #pobierz ilość pamięci fizycznej zajmowanej przez wszystkie procesy Inte
proc_info[iexplore.exe,pf,avg] #pobierz średnią liczbę błędów strony dla procesów Internet Explorer
```

registry.data[key,<value name>]

 Zwraca dane dla określonej nazwy wartości w kluczu Rejestru Windows.
 Zwracana wartość: *Integer*, *string* lub *text* (w zależności od typu wartości)

Parametry:

- **key** - klucz rejestru wraz z kluczem głównym; dozwolone są skróty kluczy głównych (np. HKLM);

- **value name** - nazwa wartości rejestru w kluczu (domyślnie pusty ciąg ""); Jeśli nazwa wartości nie zostanie podana, zwracana jest wartość domyślna.

Uwagi:

- Obsługiwane skróty kluczy głównych:
HKCR - HKEY_CLASSES_ROOT
HKCC - HKEY_CURRENT_CONFIG
HKCU - HKEY_CURRENT_USER
HKCULS - HKEY_CURRENT_USER_LOCAL_SETTINGS
HKLM - HKEY_LOCAL_MACHINE
HKPD - HKEY_PERFORMANCE_DATA
HKPN - HKEY_PERFORMANCE_NLSTEXT
HKPT - HKEY_PERFORMANCE_TEXT
HKU - HKEY_USERS
- Klucze zawierające spacje muszą być ujęte w podwójny cudzysłów.

Przykłady:

```
registry.data["HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting"] #zwraca dane wartoś
registry.data["HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting","EnableZip"] #zwraca dane wartości
```

registry.get[key,<mode>,<name regexp>]

 Lista wartości lub kluczy Rejestru Windows znajdujących się w podanym kluczu.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **key** - klucz rejestru wraz z kluczem głównym; dozwolone są skróty kluczy głównych (np. HKLM) (pełną listę skrótów można znaleźć w komentarzach dla registry.data[]);
- **mode** - możliwe wartości:
values (domyślnie) lub *keys*;
- **name regexp** - wykrywaj tylko wartości, których nazwy pasują do wyrażenia regularnego (domyślnie - wykrywane są wszystkie wartości). Dozwolone tylko z *values* jako `mode`.

Klucze zawierające spacje muszą być ujęte w podwójny cudzysłów.

Przykłady:

```
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values,"^DisplayName|DisplayVersion$
JSON będzie zawierał szczegóły klucza, ostatniego podklucza, nazwy wartości, typu wartości i danych wartoś
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values] #zwraca dane wszystkich wart
JSON będzie zawierał szczegóły klucza, ostatniego podklucza, nazwy wartości, typu wartości i danych wartoś
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,keys] #zwraca wszystkie podklucze te
JSON będzie zawierał szczegóły klucza i ostatniego podklucza.
```

service.discovery

 Lista usług Windows. Używana do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

service.info[service,<param>]

 Informacje o usłudze.
 Wartość zwracana: *Integer* - gdy `param` to *state*, *startup*; *String* - gdy `param` to *displayname*, *path*, *user*; *Text* - gdy `param` to *description*
Dokładnie dla *state*: 0 - uruchomiona, 1 - wstrzymana, 2 - oczekiwanie na uruchomienie, 3 - oczekiwanie na wstrzymanie, 4 - oczekiwanie na wznowienie, 5 - oczekiwanie na zatrzymanie, 6 - zatrzymana, 7 - nieznany, 255 - brak takiej usługi
Dokładnie dla *startup*: 0 - automatyczny, 1 - automatyczny opóźniony, 2 - ręczny, 3 - wyłączony, 4 - nieznany, 5 - automatyczny start wyzwalany, 6 - automatyczny opóźniony start wyzwalany, 7 - ręczny start wyzwalany

Parametry:

- **service** - rzeczywista nazwa usługi lub jej nazwa wyświetlana widoczna w przystawce MMC Services;
- **param** - *state* (domyślnie), *displayname*, *path*, *user*, *startup* lub *description*.

Uwagi:

- Pozycje takie jak `service.info[service,state]` i `service.info[service]` zwrócą te same informacje;
- Tylko gdy `param` to *state*, ta pozycja zwraca wartość dla nieistniejących usług (255).

Przykłady:

```
service.info[SNMPTRAP] - stan usługi SNMPTRAP;
service.info[SNMP Trap] - stan tej samej usługi, ale z podaną nazwą wyświetlaną;
service.info[EventLog,startup] - typ uruchamiania usługi EventLog
```

services[<type>,<state>,<exclude>]

 Lista usług.
 Zwracana wartość: *0* - jeśli pusta; *Text* - lista usług rozdzielonych znakiem nowej linii.

Parametry:

- **type** - *all* (domyślnie), *automatic*, *manual* lub *disabled*;

- **state** - *all* (domyślnie), *stopped*, *started*, *start_pending*, *stop_pending*, *running*, *continue_pending*, *pause_pending* lub *paused*;
- **exclude** - usługi, które należy wykluczyć z wyniku. Wykluczane usługi powinny być podane w podwójnych cudzysłowach, rozdzielone przecinkami, bez spacji.

Przykłady:

```
services[,started] #zwraca listę uruchomionych usług;
services[automatic, stopped] #zwraca listę zatrzymanych usług, które powinny być uruchomione;
services[automatic, stopped, "service1,service2,service3"] #zwraca listę zatrzymanych usług, które powinny
```

vm.vmemory.size[<type>]

 Rozmiar pamięci wirtualnej w bajtach lub jako procent całości.
 Zwracana wartość: *Integer* - dla bajtów; *float* - dla wartości procentowej.

Parametr:

- **type** - możliwe wartości: *available* (dostępna pamięć wirtualna), *pavailable* (dostępna pamięć wirtualna, w procentach), *pused* (używana pamięć wirtualna, w procentach), *total* (całkowita pamięć wirtualna, domyślnie) lub *used* (używana pamięć wirtualna)

Komentarze:

- Monitorowanie statystyk pamięci wirtualnej opiera się na:
  - Całkowitej pamięci wirtualnej w systemie Windows (całkowita pamięć fizyczna + rozmiar pliku stronicowania);
  - Maksymalnej ilości pamięci, którą agent Zabbix może przydzielić;
  - Bieżącym limicie pamięci przydzielonej dla systemu lub agent Zabbix, w zależności od tego, która wartość jest mniejsza.

Przykład:

```
vm.vmemory.size[pavailable] #zwraca dostępną pamięć wirtualną, w procentach
```

wmi.get[<namespace>,<query>]

 Wykonuje zapytanie WMI i zwraca pierwszy wybrany obiekt.
 Wartość zwracana: *Integer*, *float*, *string* lub *text* (w zależności od żądania).

Parametry:

- **namespace** - przestrzeń nazw WMI;
- **query** - zapytanie WMI zwracające pojedynczy obiekt.

Zapytania WMI są wykonywane przy użyciu WQL.

Przykład:

```
wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] #zwraca status p
```

wmi.getall[<namespace>,<query>]

 Wykonuje zapytanie WMI i zwraca całą odpowiedź. Może być używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*

Parametry:

- **namespace** - przestrzeń nazw WMI;
- **query** - zapytanie WMI.

Uwagi:

- Zapytania WMI są wykonywane przy użyciu WQL.
- Przetwarzanie wstępne JSONPath może być używane do wskazania bardziej szczegółowych wartości w zwróconym JSON.

Przykład:

```
wmi.getall[root\cimv2,select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'] #zwraca informacje
```

Monitorowanie usług Windows

Ten samouczek zawiera instrukcje krok po kroku dotyczące konfiguracji monitorowania usług Windows. Zakłada się, że serwer Zabbix i agent są skonfigurowane i działają poprawnie.

Krok 1

Pobierz nazwę usługi.

Nazwę usługi można uzyskać, przechodząc do przystawki MMC Services i otwierając właściwości usługi. Na karcie *General* powinno być widoczne pole o nazwie „Service name". Następująca po nim wartość to nazwa, której użyjesz podczas konfigurowania pozycji do monitorowania. Na przykład, jeśli chcesz monitorować usługę „workstation", nazwa usługi może być następująca: **lanmanworkstation**.

Krok 2

Skonfiguruj pozycję do monitorowania usługi.

Pozycja `service.info[service,<param>]` pobiera informacje o określonej usłudze. W zależności od potrzebnych informacji należy określić opcję `param`, która przyjmuje następujące wartości: *displayname*, *state*, *path*, *user*, *startup* lub *description*. Wartością domyślną jest *state*, jeśli `param` nie zostanie określony (`service.info[service]`).

Typ zwracanej wartości zależy od wybranego `param`: integer dla *state* i *startup*; character string dla *displayname*, *path* i *user*; text dla *description*.

Przykład:

- *Klucz:* `service.info[lanmanworkstation]`
- *Typ informacji:* Numeryczny (bez znaku)

Pozycja `service.info[lanmanworkstation]` pobierze informacje o stanie usługi jako wartość liczbową. Aby odwzorować wartość liczbową na reprezentację tekstową we frontendzie („0" jako „Running", „1" jako „Paused" itd.), można skonfigurować mapowanie wartości na hoście, na którym skonfigurowano pozycję. Aby to zrobić, należy albo podłączyć szablon *Windows services by Zabbix agent* lub *Windows services by Zabbix agent active* do hosta, albo skonfigurować na hoście nową mapę wartości opartą na mapie wartości *Windows service state* skonfigurowanej we wspomnianych szablonach.

Należy pamiętać, że oba wspomniane szablony mają skonfigurowaną regułę wykrywania, która będzie automatycznie wykrywać usługi. Jeśli nie jest to pożądane, można wyłączyć regułę wykrywania na poziomie hosta po podłączeniu szablonu do hosta.

Wykrywanie usług Windows

Wykrywanie niskiego poziomu zapewnia sposób na automatyczne tworzenie pozycji, wyzwalaczy i wykresów dla różnych obiektów na komputerze. Zabbix może automatycznie rozpocząć monitorowanie usług Windows na komputerze, bez potrzeby znajomości dokładnej nazwy usługi ani ręcznego tworzenia pozycji dla każdej usługi. Filtr może zostać użyty do generowania rzeczywistych pozycji, wyzwalaczy i wykresów tylko dla interesujących usług.

3 Monitorowanie plików dziennika

Przegląd

Zabbix może być używany do scentralizowanego monitorowania i analizy plików dziennika z obsługą lub bez obsługi rotacji logów.

Powiadomienia mogą służyć do ostrzegania użytkowników, gdy plik dziennika zawiera określone ciągi znaków lub wzorce ciągów.

Aby monitorować plik dziennika, musisz mieć:

- uruchomiony agent Zabbix na hoście
- skonfigurowaną pozycję monitorowania logów

> **Attention:**
> Limit rozmiaru monitorowanego pliku dziennika zależy od obsługi dużych plików.

Konfiguracja

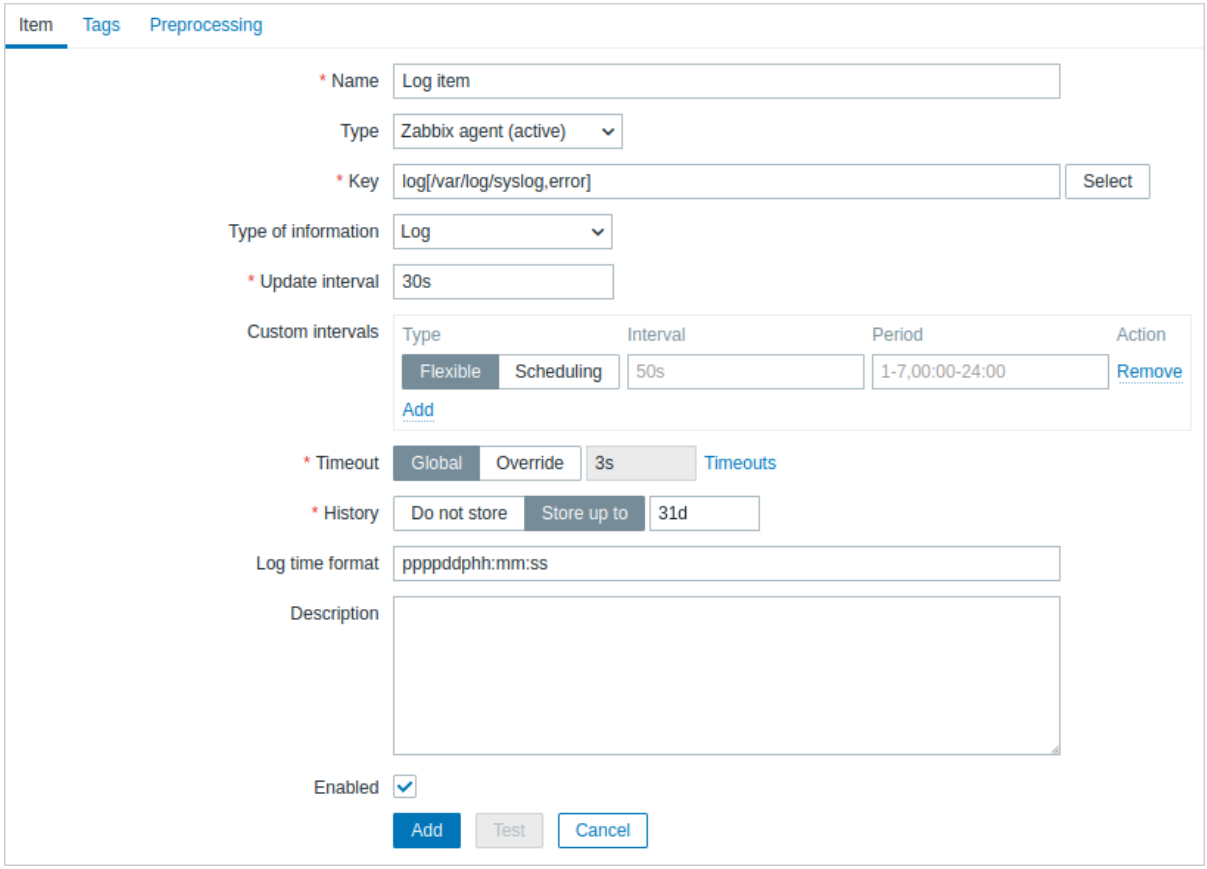
Zweryfikuj parametry agenta

Upewnij się, że w pliku konfiguracyjnym agenta:

- parametr `Hostname` odpowiada nazwie hosta we frontendzie.
- serwery w parametrze `ServerActive` są określone do przetwarzania aktywnych kontroli.

Konfiguracja pozycja

Skonfiguruj pozycję do monitorowania logów.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W przypadku pozycji monitorujących logi należy wprowadzić:

Type	Wybierz tutaj **Zabbix agent (active)**.
Key	Użyj jednego z następujących kluczy pozycji: **log[]** lub **logrt[]**: Te dwa klucze pozycji umożliwiają monitorowanie logów i filtrowanie wpisów logów według wyrażenia regularnego zawartości, jeśli jest obecne. Na przykład: `log[/var/log/syslog,error]`. Upewnij się, że plik ma uprawnienia do odczytu dla użytkownika 'zabbix', w przeciwnym razie status pozycji zostanie ustawiony na 'unsupported'. **log.count[]** lub **logrt.count[]**: Te dwa klucze pozycji umożliwiają zwracanie wyłącznie liczby pasujących wierszy. Szczegóły dotyczące używania tych kluczy pozycji i ich parametrów znajdziesz w sekcji obsługiwanych kluczy pozycji Zabbix agent.
Type of information	Wypełniane automatycznie: Dla pozycji `log[]` lub `logrt[]` - `Log`; Dla pozycji `log.count[]` lub `logrt.count[]` - `Numeric (unsigned)`. Jeśli opcjonalnie używasz parametru `output`, możesz ręcznie wybrać odpowiedni typ informacji inny niż `Log`. Pamiętaj, że wybranie typu informacji innego niż `Log` spowoduje utratę lokalnego znacznika czasu.
Update interval (in sec)	Ten parametr określa, jak często agent Zabbix będzie sprawdzać zmiany w pliku logu. Ustawienie go na 1 sekundę zapewni otrzymywanie nowych rekordów tak szybko, jak to możliwe.

Log time format	W tym polu możesz opcjonalnie określić wzorzec do parsowania znacznika czasu w wierszu logu. Obsługiwane symbole zastępcze: * **y**: *Rok (1970-2038)* * **M**: *Miesiąc (01-12)* * **d**: *Dzień (01-31)* * **h**: *Godzina (00-23)* * **m**: *Minuta (00-59)* * **s**: *Sekunda (00-59)* Jeśli pozostawisz to pole puste, znacznik czasu zostanie ustawiony na 0 w czasie Unix, co odpowiada 1 stycznia 1970. Na przykład rozważ następujący wiersz z pliku logu agenta Zabbix: ” 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).” Zaczyna się on od sześciu znaków dla PID, po których następują data, czas i reszta komunikatu. Format czasu logu dla tego wiersza byłby następujący: ”ppppppp:yyyyMMdd:hhmmss”. Pamiętaj, że znaki ”p” i ”:” są symbolami zastępczymi i mogą być dowolnymi znakami z wyjątkiem ”yMdhms”.

Ważne uwagi

- Serwer i agent przechowują ślad rozmiaru monitorowanego logu oraz czasu ostatniej modyfikacji (dla logrt) w dwóch licznikach. Dodatkowo:
  - Agent wewnętrznie używa także numerów inode (na UNIX/GNU/Linux), indeksów plików (na Microsoft Windows) oraz sum MD5 pierwszych 512 bajtów pliku logu, aby poprawić decyzje, gdy pliki logów są obcinane i rotowane.
  - W systemach UNIX/GNU/Linux zakłada się, że systemy plików, na których są przechowywane pliki logów, zwracają numery inode, które można wykorzystać do śledzenia plików.
  - W systemie Microsoft Windows agent Zabbix określa typ systemu plików, na którym znajdują się pliki logów, i używa:
    * W systemach plików NTFS 64-bitowych indeksów plików.
    * W systemach plików ReFS (tylko od Microsoft Windows Server 2012) 128-bitowych identyfikatorów plików.
    * W systemach plików, w których indeksy plików się zmieniają (np. FAT32, exFAT), stosowany jest algorytm awaryjny, aby przyjąć rozsądne podejście w niepewnych warunkach, gdy rotacja pliku logu powoduje wiele plików logów z tym samym czasem ostatniej modyfikacji.
  - Numery inode, indeksy plików i sumy MD5 są wewnętrznie zbierane przez agent Zabbix. Nie są przesyłane do serwer Zabbix i są tracone po zatrzymaniu agenta Zabbix.
  - Nie modyfikuj czasu ostatniej modyfikacji pliku logu (na przykład za pomocą `touch`) i nie zastępuj monitorowanego pliku logu przez skopiowanie pliku z powrotem pod jego oryginalną nazwę (tworzy to nowy inode). W obu przypadkach Zabbix może potraktować plik jako inny plik i odczytać go ponownie od początku, co może spowodować duplikaty alertów.
  - Jeśli dla pozycji `logrt[]` istnieje kilka pasujących plików logów, a agent Zabbix śledzi najnowszy z nich i ten najnowszy plik logu zostanie usunięty, w dzienniku zostanie zapisany komunikat ostrzegawczy `"there are no files matching "<regexp mask>" in "<directory>"`. Agent Zabbix ignoruje pliki logów z czasem modyfikacji wcześniejszym niż najnowszy czas modyfikacji widziany przez agenta dla sprawdzanej pozycji `logrt[]`.
- Agent rozpoczyna odczyt pliku logu od miejsca, w którym zakończył poprzednio.
- Liczba już przeanalizowanych bajtów (licznik rozmiaru) oraz czas ostatniej modyfikacji (licznik czasu) są przechowywane w bazie danych Zabbix i są wysyłane do agenta, aby upewnić się, że agent rozpocznie odczyt pliku logu od tego miejsca w przypadkach, gdy agent został właśnie uruchomiony lub otrzymał pozycje, które wcześniej były wyłączone lub nieobsługiwane. Jednak jeśli agent otrzymał od serwer niezerowy licznik rozmiaru, ale pozycja logrt[] lub logrt.count[] nie może znaleźć pasujących plików, licznik rozmiaru jest resetowany do 0, aby analizować od początku, jeśli pliki pojawią się później.
- Za każdym razem, gdy plik logu staje się mniejszy niż znany agentowi licznik rozmiaru logu, licznik jest resetowany do zera, a agent rozpoczyna odczyt pliku logu od początku, biorąc pod uwagę licznik czasu.
- Jeśli w katalogu znajduje się kilka pasujących plików o tym samym czasie ostatniej modyfikacji, agent próbuje poprawnie przeanalizować wszystkie pliki logów z tym samym czasem modyfikacji i uniknąć pomijania danych lub analizowania tych samych danych dwukrotnie, choć nie można tego zagwarantować we wszystkich sytuacjach. Agent nie zakłada żadnego konkretnego schematu rotacji plików logów ani go nie określa. W przypadku wielu plików logów z tym samym czasem ostatniej modyfikacji agent przetworzy je w kolejności leksykograficznie malejącej. W związku z tym dla niektórych schematów rotacji pliki logów będą analizowane i raportowane w ich oryginalnej kolejności. Dla innych schematów rotacji oryginalna kolejność plików logów nie będzie zachowana, co może prowadzić do raportowania dopasowanych rekordów pliku logu w zmienionej kolejności (problem nie występuje, jeśli pliki logów mają różne czasy ostatniej modyfikacji).
- Agent Zabbix przetwarza nowe rekordy pliku logu raz na *Update interval* sekund.
- Agent Zabbix nie wysyła więcej niż **maxlines** pliku logu na sekundę. Limit ten zapobiega przeciążeniu zasobów sieci i CPU oraz zastępuje wartość domyślną podaną przez parametr **MaxLinesPerSecond** w pliku konfiguracyjnym agenta.
- Aby znaleźć wymagany ciąg, Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w MaxLinesPerSecond. Tak więc na przykład, jeśli pozycja `log[]` lub `logrt[]` ma *Update interval* równy 1 sekundzie, domyślnie agent przeanalizuje nie

więcej niż 200 rekordów pliku logu i wyśle nie więcej niż 20 pasujących rekordów do serwer Zabbix w jednym sprawdzeniu. Zwiększając **MaxLinesPerSecond** w pliku konfiguracyjnym agenta lub ustawiając parametr **maxlines** w kluczu pozycji, można zwiększyć limit do 10000 przeanalizowanych rekordów pliku logu i 1000 pasujących rekordów wysłanych do serwer Zabbix w jednym sprawdzeniu. Jeśli *Update interval* jest ustawiony na 2 sekundy, limity dla jednego sprawdzenia będą ustawione 2 razy wyżej niż przy *Update interval* równym 1 sekundzie.

- Dodatkowo wartości log i log.count są zawsze ograniczone do 50% rozmiaru bufora wysyłania agenta, nawet jeśli nie ma w nim żadnych wartości innych niż log. Aby więc wartości **maxlines** mogły zostać wysłane w jednym połączeniu (a nie w kilku), parametr BufferSize agenta musi wynosić co najmniej maxlines x 2. Agent Zabbix może przesyłać dane podczas zbierania logów i tym samym zwalniać bufor, natomiast agent Zabbix 2 wstrzyma zbieranie logów do czasu przesłania danych i zwolnienia bufora, co odbywa się asynchronicznie.
- W przypadku braku pozycji log cały rozmiar bufora agenta jest używany dla wartości innych niż log. Gdy pojawiają się wartości log, zastępują starsze wartości inne niż log w razie potrzeby, aż do wyznaczonych 50%.
- Dla rekordów pliku logu dłuższych niż 256kB tylko pierwsze 256kB są dopasowywane do wyrażenia regularnego, a reszta rekordu jest ignorowana. Jednak jeśli agent Zabbix zostanie zatrzymany podczas obsługi długiego rekordu, wewnętrzny stan agenta zostaje utracony i długi rekord może zostać przeanalizowany ponownie i inaczej po ponownym uruchomieniu agenta.
- Uwaga specjalna dotycząca separatorów ścieżek "\": jeśli file_format ma wartość "file\.log", to nie powinien istnieć katalog "file", ponieważ nie można jednoznacznie określić, czy "." jest znakiem ucieczki, czy pierwszym symbolem nazwy pliku.
- Wyrażenia regularne dla `logrt` są obsługiwane tylko w nazwie pliku, dopasowywanie wyrażeniem regularnym dla katalogu nie jest obsługiwane.
- Na platformach UNIX pozycja `logrt[]` staje się NOTSUPPORTED, jeśli katalog, w którym oczekuje się plików logów, nie istnieje.
- W systemie Microsoft Windows, jeśli katalog nie istnieje, pozycja nie stanie się NOTSUPPORTED (na przykład, jeśli katalog jest błędnie wpisany w kluczu pozycji).
- Brak plików logów dla pozycji `logrt[]` nie powoduje stanu NOTSUPPORTED. Błędy odczytu plików logów dla pozycji `logrt[]` są zapisywane jako ostrzeżenia w pliku logu agenta Zabbix, ale nie powodują stanu NOTSUPPORTED.
- Plik logu agenta Zabbix może być pomocny w ustaleniu, dlaczego pozycja `log[]` lub `logrt[]` stała się NOTSUPPORTED. Zabbix może monitorować plik logu swojego agenta, z wyjątkiem sytuacji przy DebugLevel=4 lub DebugLevel=5.
- Wyszukiwanie znaku zapytania za pomocą wyrażenia regularnego, np. \?, może dawać wyniki fałszywie dodatnie, jeśli plik tekstowy zawiera symbole NUL, ponieważ są one zastępowane przez "?" przez Zabbix, aby kontynuować przetwarzanie linii do znaku nowej linii.

Wyodrębnianie pasującej części wyrażenia regularnego

Czasami chcemy wyodrębnić z pliku docelowego tylko interesującą wartość zamiast zwracać cały wiersz, gdy zostanie znalezione dopasowanie wyrażenia regularnego.

Pozycje typu log mają możliwość wyodrębniania żądanych wartości z dopasowanych wierszy. Jest to realizowane za pomocą dodatkowego parametru **output** w pozycjach `log` i `logrt`.

Użycie parametru `output` pozwala wskazać "grupę przechwytującą" dopasowania, która może nas interesować.

Na przykład

```
log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+)",,,,\1]
```

powinno umożliwić zwrócenie liczby wpisów znalezionych w zawartości:

```
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
```

Zwrócona zostanie tylko liczba, ponieważ **\1** odnosi się do pierwszej i jedynej grupy przechwytującej: **([0-9]+)**.

A dzięki możliwości wyodrębniania i zwracania liczby, wartość ta może być użyta do definiowania wyzwalaczy.

Używanie parametru maxdelay

Parametr `maxdelay` w pozycjach logów pozwala ignorować niektóre starsze wiersze z plików logów, aby w ramach `maxdelay` sekund analizowane były najnowsze wiersze.

> **Warning:**
> Określenie `maxdelay` > 0 może prowadzić do **ignorowania ważnych rekordów plików logów i pominiętych alertów**. Używaj go ostrożnie, na własne ryzyko, tylko wtedy, gdy jest to konieczne.

Domyślnie pozycje do monitorowania logów przetwarzają wszystkie nowe wiersze pojawiające się w plikach logów. Istnieją jednak aplikacje, które w niektórych sytuacjach zaczynają zapisywać ogromną liczbę komunikatów do swoich plików logów. Na przykład, jeśli baza danych lub serwer DNS jest niedostępny, takie aplikacje zalewają pliki logów tysiącami niemal identycznych komunikatów

o błędach, aż do przywrócenia normalnego działania. Domyślnie wszystkie te komunikaty będą sumiennie analizowane, a pasujące wiersze będą wysyłane do serwera zgodnie z konfiguracją w pozycjach `log` i `logrt`.

Wbudowana ochrona przed przeciążeniem składa się z konfigurowalnego parametru `maxlines` (chroni serwer przed zbyt dużą liczbą przychodzących pasujących wierszy logów) oraz limitu 10*'maxlines' (chroni CPU i I/O hosta przed przeciążeniem przez agent w jednym sprawdzeniu). Mimo to istnieją 2 problemy związane z wbudowaną ochroną. Po pierwsze, do serwera zgłaszana jest duża liczba potencjalnie mało informacyjnych komunikatów, które zajmują miejsce w bazie danych. Po drugie, z powodu ograniczonej liczby wierszy analizowanych na sekundę agent może przez wiele godzin pozostawać w tyle za najnowszymi rekordami logów. Najprawdopodobniej wolisz zostać wcześniej poinformowany o bieżącej sytuacji w plikach logów, zamiast przez wiele godzin przeglądać stare rekordy.

Rozwiązaniem obu problemów jest użycie parametru `maxdelay`. Jeśli zostanie określone `maxdelay > 0`, podczas każdego sprawdzenia mierzona jest liczba przetworzonych bajtów, liczba pozostałych bajtów oraz czas przetwarzania. Na podstawie tych wartości agent oblicza szacowane opóźnienie - ile sekund zajęłoby przeanalizowanie wszystkich pozostałych rekordów w pliku logu.

Jeśli opóźnienie nie przekracza `maxdelay`, agent kontynuuje analizę pliku logu jak zwykle.

Jeśli opóźnienie jest większe niż `maxdelay`, agent **pomija fragment pliku logu, "przeskakując" nad nim** do nowej szacowanej pozycji, tak aby pozostałe wiersze mogły zostać przeanalizowane w ciągu `maxdelay` sekund.

Należy zauważyć, że agent nawet nie odczytuje pominiętych wierszy do bufora, lecz oblicza przybliżoną pozycję, do której ma przeskoczyć w pliku.

Fakt pomijania wierszy pliku logu jest zapisywany w pliku logu agenta w następujący sposób:

```
14287:20160602:174344.206 item:"logrt["/home/zabbix32/test[0-9].log",ERROR,,1000,,,120.0]"
logfile:"/home/zabbix32/test1.log" skipping 679858 bytes
(from byte 75653115 to byte 76332973) to meet maxdelay
```

Wartość „to byte" jest przybliżona, ponieważ po „przeskoczeniu" agent dostosowuje pozycję w pliku do początku wiersza logu, który może znajdować się dalej w pliku lub wcześniej.

W zależności od tego, jak szybko rośnie log w porównaniu z szybkością jego analizowania, możesz nie zobaczyć żadnych „przeskoków", rzadkie lub częste „przeskoki", duże lub małe „przeskoki", a nawet niewielki „przeskok" przy każdym sprawdzeniu. Wahania obciążenia systemu i opóźnienia sieciowe również wpływają na obliczanie opóźnienia, a tym samym na „przeskakiwanie" do przodu, aby nadążyć za parametrem `maxdelay`.

Nie zaleca się ustawiania `maxdelay < update interval` (może to skutkować częstymi małymi „przeskokami").

Uwagi dotyczące obsługi rotacji plików dziennika `copytruncate`

`logrt` z opcją `copytruncate` zakłada, że różne pliki dziennika mają różne rekordy (co najmniej ich znaczniki czasu są różne), dlatego sumy MD5 początkowych bloków (do pierwszych 512 bajtów) będą różne. Dwa pliki z takimi samymi sumami MD5 początkowych bloków oznaczają, że jeden z nich jest oryginałem, a drugi - kopią.

`logrt` z opcją `copytruncate` dokłada starań, aby poprawnie przetwarzać kopie plików dziennika bez zgłaszania duplikatów. Nie zaleca się jednak takich sytuacji, jak tworzenie wielu kopii pliku dziennika z tym samym znacznikiem czasu, rotacja pliku dziennika częściej niż interwał aktualizacji pozycji `logrt\[\]` oraz częste ponowne uruchamianie agent. Agent stara się obsługiwać wszystkie te sytuacje w rozsądny sposób, ale nie można zagwarantować dobrych wyników w każdych okolicznościach.

Uwagi dotyczące trwałych plików dla pozycji log*[]

Cel plików trwałych

Gdy agent Zabbix jest uruchamiany, otrzymuje listę aktywnych kontroli z serwera Zabbix lub proxy. W przypadku metryk log*[] otrzymuje przetworzony rozmiar logu oraz czas modyfikacji, aby ustalić, od którego miejsca rozpocząć monitorowanie pliku logu. W zależności od rzeczywistego rozmiaru pliku logu i czasu modyfikacji zgłoszonych przez system plików agent decyduje, czy kontynuować monitorowanie pliku logu od przetworzonego rozmiaru logu, czy przeanalizować plik logu ponownie od początku.

Działający agent utrzymuje większy zestaw atrybutów do śledzenia wszystkich monitorowanych plików logu pomiędzy kontrolami. Ten stan przechowywany w pamięci jest tracony po zatrzymaniu agenta.

Nowy opcjonalny parametr **persistent_dir** określa katalog do przechowywania tego stanu pozycji log[], log.count[], logrt[] lub logrt.count[] w pliku. Stan pozycji log jest przywracany z pliku trwałego po ponownym uruchomieniu agenta Zabbix.

Głównym przypadkiem użycia jest monitorowanie pliku logu znajdującego się na lustrzanym systemie plików. Do pewnego momentu plik logu jest zapisywany na obu lustrach. Następnie lustra są rozdzielane. Na aktywnej kopii plik logu nadal rośnie, otrzymując nowe rekordy. Agent Zabbix analizuje go i wysyła przetworzony rozmiar logu oraz czas modyfikacji do serwera. Na pasywnej kopii plik logu pozostaje bez zmian, daleko za aktywną kopią. Później system operacyjny i agent Zabbix są uruchamiane ponownie z pasywnej kopii. Przetworzony rozmiar logu i czas modyfikacji, które agent Zabbix otrzymuje z serwera, mogą nie być

prawidłowe dla sytuacji na pasywnej kopii. Aby kontynuować monitorowanie pliku logu od miejsca, w którym agent zakończył pracę w momencie rozdzielenia lustra systemu plików, agent przywraca swój stan z pliku trwałego.

Działanie agent z plikami trwałymi

Przy uruchomieniu agent Zabbix nie wie nic o plikach trwałych. Dopiero po otrzymaniu listy aktywnych kontroli od serwer Zabbix (proxy) agent widzi, że niektóre pozycje logów powinny być obsługiwane przez pliki trwałe w określonych katalogach.

Podczas działania agent pliki trwałe są otwierane do zapisu (za pomocą fopen(filename, ”w”)) i nadpisywane najnowszymi danymi. Prawdopodobieństwo utraty danych pliku trwałego, jeśli nadpisywanie i rozdzielenie kopii lustrzanej systemu plików nastąpią w tym samym czasie, jest bardzo małe; nie jest wymagane żadne specjalne postępowanie. Zapisywanie do pliku trwałego NIE jest poprzedzane wymuszonym synchronizowaniem z nośnikiem danych (fsync() nie jest wywoływane).

Nadpisywanie najnowszymi danymi odbywa się po pomyślnym zgłoszeniu pasującego rekordu pliku logu lub metadanych (przetworzony rozmiar logu i czas modyfikacji) do serwer Zabbix. Może to następować nawet przy każdym sprawdzeniu pozycji, jeśli plik logu nadal się zmienia.

Brak specjalnych działań podczas zamykania agent.

Po otrzymaniu listy aktywnych kontroli agent oznacza przestarzałe trwałe pliki do usunięcia. Trwały plik staje się przestarzały, jeśli:

1. Odpowiednia pozycja logu nie jest już monitorowana.
2. Pozycja logu zostaje ponownie skonfigurowana z inną lokalizacją **persistent_dir** niż wcześniej.

Usunięcie jest wykonywane z opóźnieniem 24 godzin, ponieważ pliki logów w stanie NOTSUPPORTED nie są uwzględniane na liście aktywnych kontroli, ale mogą później stać się SUPPORTED, a ich trwałe pliki będą wtedy przydatne.

Jeśli agent zostanie zatrzymany przed upływem 24 godzin, przestarzałe pliki nie zostaną usunięte, ponieważ agent Zabbix nie otrzymuje już od serwer Zabbix informacji o ich lokalizacji.

Warning:
Ponowne skonfigurowanie **persistent_dir** pozycji logu z powrotem na starą lokalizację **persistent_dir** podczas zatrzymania agenta, bez usunięcia przez użytkownika starego trwałego pliku, spowoduje odtworzenie stanu agenta ze starego trwałego pliku, co doprowadzi do pominiętych komunikatów lub fałszywych alertów.

Nazewnictwo i lokalizacja plików trwałych

Zabbix agent rozróżnia aktywne kontrole na podstawie ich kluczy. Na przykład logrt[/home/zabbix/test.log] i logrt[/home/zabbix/test.log,] są różnymi pozycjami. Zmodyfikowanie pozycji logrt[/home/zabbix/test.log„,10] w frontend na logrt[/home/zabbix/test.log„,20] spowoduje usunięcie pozycji logrt[/home/zabbix/test.log„,10] z listy aktywnych kontroli agenta i utworzenie pozycji logrt[/home/zabbix/test.log„,20] (niektóre atrybuty są przenoszone podczas modyfikacji w frontend/serwer, ale nie w agencie).

Nazwa pliku jest tworzona z sumy MD5 klucza pozycji z dołączoną długością klucza pozycji, aby zmniejszyć prawdopodobieństwo kolizji. Na przykład stan pozycji logrt[/home/zabbix50/test.log„„„„„/home/zabbix50/agent_private] będzie przechowywany w pliku trwałym c963ade4008054813bbc0a650bb8e09266.

Wiele pozycji log może używać tej samej wartości **persistent_dir**.

persistent_dir jest określany z uwzględnieniem konkretnego układu systemu plików, punktów montowania i opcji montowania oraz konfiguracji mirroringu pamięci masowej - plik trwały powinien znajdować się na tym samym zmirroringowanym systemie plików co monitorowany plik log.

Jeśli katalog **persistent_dir** nie może zostać utworzony lub nie istnieje, albo uprawnienia dostępu dla Zabbix agent nie pozwalają na tworzenie/zapis/odczyt/usuwanie plików, pozycja log staje się NOTSUPPORTED.

Jeśli prawa dostępu do plików trwałej pamięci masowej zostaną usunięte podczas działania agenta lub wystąpią inne błędy (np. brak miejsca na dysku), błędy są zapisywane w pliku dziennika agenta, ale pozycja logu nie przechodzi w stan NOTSUPPORTED.

Obciążenie I/O

Trwały plik pozycja jest aktualizowany po pomyślnym wysłaniu każdej partii danych (zawierającej dane pozycji) do serwer. Na przykład domyślna wartość `BufferSize` wynosi 100. Jeśli pozycja logu znalazła 70 pasujących rekordów, to pierwsze 50 rekordów zostanie wysłanych w jednej partii, trwały plik zostanie zaktualizowany, a następnie pozostałe 20 rekordów zostanie wysłanych (być może z pewnym opóźnieniem, gdy zostanie zgromadzonych więcej danych) w drugiej partii, po czym trwały plik zostanie ponownie zaktualizowany.

Akcje w przypadku awarii komunikacji między agentem a serwerem

Każda pasująca linia z pozycji `log[]` i `logrt[]` oraz wynik każdego sprawdzenia pozycji `log.count[]` i `logrt.count[]` wymaga wolnego slotu w wyznaczonej 50% części bufora wysyłania agenta. Elementy bufora są regularnie wysyłane do serwera (lub proxy), a sloty bufora są ponownie zwalniane.

Gdy w wyznaczonej części logów w buforze wysyłania agenta są wolne sloty i komunikacja między agentem a serwerem (lub proxy) zawodzi, wyniki monitorowania logów są gromadzone w buforze wysyłania. Pomaga to ograniczyć skutki krótkotrwałych awarii komunikacji.

Podczas dłuższych awarii komunikacji wszystkie sloty logów zostają zajęte i podejmowane są następujące działania:

- Sprawdzanie pozycji `log[]` i `logrt[]` zostaje zatrzymane. Po przywróceniu komunikacji i dostępności wolnych slotów w buforze sprawdzanie jest wznawiane od poprzedniego miejsca. Żadne pasujące linie nie są tracone, są jedynie raportowane później.
- Sprawdzanie `log.count[]` i `logrt.count[]` zostaje zatrzymane, jeśli `maxdelay = 0` (domyślnie). Zachowanie jest podobne do pozycji `log[]` i `logrt[]`, jak opisano powyżej. Należy zauważyć, że może to wpływać na wyniki `log.count[]` i `logrt.count[]`: na przykład jedno sprawdzenie zlicza 100 pasujących linii w pliku logu, ale ponieważ nie ma wolnych slotów w buforze, sprawdzanie zostaje zatrzymane. Po przywróceniu komunikacji agent zlicza te same 100 pasujących linii, a także 70 nowych pasujących linii. Agent wysyła teraz count = 170, tak jakby zostały znalezione w jednym sprawdzeniu.
- Sprawdzanie `log.count[]` i `logrt.count[]` z `maxdelay > 0`: jeśli podczas sprawdzania nie nastąpił "skok", zachowanie jest podobne do opisanego powyżej. Jeśli nastąpił "skok" o linie pliku logu, pozycja po "skoku" jest zachowywana, a wynik zliczania jest odrzucany. W ten sposób agent próbuje nadążać za rosnącym plikiem logu nawet w przypadku awarii komunikacji.

Obsługa błędów kompilacji i błędów wykonania wyrażeń regularnych

Jeśli wyrażenie regularne używane w pozycji `log[]`, `logrt[]`, `log.count[]` lub `logrt.count[]` nie może zostać skompilowane przez bibliotekę PCRE lub PCRE2, pozycja przechodzi do stanu NOTSUPPORTED z komunikatem o błędzie.
Aby kontynuować monitorowanie pozycji logu, należy poprawić wyrażenie regularne.

Jeśli wyrażenie regularne skompiluje się pomyślnie, ale zakończy się niepowodzeniem w czasie wykonywania (dla niektórych lub wszystkich rekordów logu), pozycja logu pozostaje obsługiwana, a monitorowanie jest kontynuowane.
Błąd wykonania jest zapisywany w pliku logu agenta Zabbix (bez rekordu z pliku logu).

Częstotliwość logowania jest ograniczona do jednego błędu wykonania na jedno sprawdzenie, aby umożliwić agentowi Zabbix monitorowanie własnego pliku logu.
Na przykład, jeśli analizowanych jest 10 rekordów, a 3 rekordy kończą się błędem wykonania regexp, w logu agenta zostanie wygenerowany jeden wpis.

Wyjątek: jeśli `MaxLinesPerSecond=1` i interwał aktualizacji wynosi 1 (tylko 1 rekord może zostać przeanalizowany na jedno sprawdzenie), błędy wykonania regexp nie są logowane.

`zabbix_agentd` zapisuje klucz pozycji w przypadku błędu wykonania, a `zabbix_agent2` zapisuje identyfikator pozycji, aby ułatwić identyfikację, która pozycja logu ma błędy wykonania.
W przypadku błędów wykonania zaleca się przeprojektowanie wyrażenia regularnego.

2 Prosty check

Przegląd

Proste sprawdzenia są zwykle używane do zdalnych sprawdzeń usług bez użycia agenta.

Należy pamiętać, że agent Zabbix nie jest potrzebny do prostych sprawdzeń. Serwer/proxy Zabbix odpowiada za przetwarzanie prostych sprawdzeń (nawiązywanie połączeń zewnętrznych itp.).

Przykłady użycia prostych sprawdzeń:

```
net.tcp.service[ftp,,155]
net.tcp.service[http]
net.tcp.service.perf[http,,8080]
net.udp.service.perf[ntp]
```

> **Note:**
> Pola *Nazwa użytkownika* i *Hasło* (ograniczone do 255 znaków) w konfiguracji pozycji prostego sprawdzenia są używane dla pozycji monitorowania VMware; w pozostałych przypadkach są ignorowane.

Obsługiwane sprawdzenia

Klucze pozycji są wymienione bez opcjonalnych parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Zobacz także klucze pozycji monitorowania VMware.

Klucz pozycji	Opis
icmpping	Dostępność hosta za pomocą ping ICMP.
icmppingloss	Procent utraconych pakietów.
icmppingretry	Dostępność hosta za pomocą ping ICMP z ponawianiem prób.
icmppingsec	Czas odpowiedzi ping ICMP.
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP.
net.tcp.service.perf	Sprawdza wydajność usługi TCP.
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.
net.udp.service.perf	Sprawdza wydajność usługi UDP.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

icmpping[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 Dostępność hosta za pomocą ICMP ping.
 Zwracana wartość: *0* - ICMP ping nie powiódł się; *1* - ICMP ping zakończył się powodzeniem.

Parametry:

- **target** - adres IP hosta lub nazwa DNS;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **options** - używane do zezwolenia na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

Przykład:

```
icmpping[,4] #Jeśli zostanie zwrócony co najmniej jeden z czterech pakietów, pozycja zwróci 1.
```

icmppingloss[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 Procent utraconych pakietów.
 Zwracana wartość: *Float*.

Parametry:

- **target** - adres IP hosta lub nazwa DNS;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **options** - używane do zezwolenia na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]

 Dostępność hosta za pomocą ICMP ping z ponowieniami. Jeśli pierwszy pakiet zakończy się powodzeniem, dalsze próby są wstrzymywane; jeśli pakiet nie powiedzie się, próby są ponawiane aż do osiągnięcia maksymalnej liczby prób określonej w parametrze `retries`. Ta pozycja jest przydatna do zmniejszenia liczby pakietów wysyłanych przez sieć.
 Wartość zwracana: *0* - ICMP ping nie powiódł się; *1* - ICMP ping zakończył się powodzeniem.

Parametry:

- **target** - adres IP lub nazwa DNS hosta;
- **retries** - liczba ponowień próby pingowania celu, bez uwzględnienia pierwszej próby (0 lub więcej; domyślnie 1);
- **backoff** - wartość, przez którą mnożony jest czas oczekiwania przy każdym kolejnym żądaniu (zakres 1.0-5.0; domyślnie 1.0);
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;

- **options** - używane do zezwalania na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

icmppingsec[<target>,<packets>,<interval>,<size>,<timeout>,<mode>,<options>]

 Czas odpowiedzi ping ICMP (w sekundach).
 Zwracana wartość: *Float*.

Parametry:

- **target** - adres IP lub nazwa DNS hosta;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **mode** - możliwe wartości: *min*, *max* lub *avg* (domyślnie);
- **options** - używane do zezwolenia na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Uwagi:

- Pakiety utracone lub takie, dla których upłynął limit czasu, nie są uwzględniane w obliczeniach;
- Jeśli host jest niedostępny (osiągnięto limit czasu), pozycja zwróci 0;
- Jeśli zwracana wartość jest mniejsza niż 0.0001 sekundy, wartość zostanie ustawiona na 0.0001 sekundy;
- Zobacz także tabelę wartości domyślnych.

net.tcp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i akceptuje połączenia TCP.
 Zwracana wartość: *0* - usługa jest niedostępna; *1* - usługa działa.

Parametry:

- **service** - możliwe wartości: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Należy pamiętać, że dla usługi *tcp* podanie portu jest obowiązkowe;
- Te testy mogą powodować dodatkowe komunikaty w plikach dziennika demonów systemowych (zwykle rejestrowane są sesje SMTP i SSH);
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście należy użyć `net.tcp.service[tcp,<ip>,port]` do takich testów.

Przykład:

```
net.tcp.service[ftp,,45] #Ta pozycja może być użyta do testowania dostępności serwera FTP na porcie TCP 45
```

Attention:
Jeśli SELinux działa w trybie enforced, niestandardowe proste testy TCP/UDP mogą być blokowane przez politykę. Aby zweryfikować i zezwolić na nowe połączenie wychodzące, sprawdź odmowy w audycie: używając `grep denied /var/log/audit/audit.log`

net.tcp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi TCP.
 Wartość zwracana: *Float*: *0.000000* - usługa jest niedostępna; *seconds* - liczba sekund poświęconych na połączenie z usługą.

Parametry:

- **service** - możliwe wartości: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Należy pamiętać, że w przypadku usługi *tcp* podanie portu jest obowiązkowe;
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście należy użyć `net.tcp.service[tcp,<ip>,port]` do takich kontroli.

Przykład:

```
net.tcp.service.perf[ssh] #Ta pozycja może być użyta do testowania szybkości początkowej odpowiedzi serwer
```

net.udp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i odpowiada na żądania UDP.
 Zwracana wartość: *0* - usługa jest niedostępna; *1* - usługa działa.

Parametry:

- **service** - możliwe wartości: *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service[ntp,,45] #Ta pozycja może być użyta do testowania dostępności usługi NTP na porcie UDP 45.
```

net.udp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi UDP.
 Zwracana wartość: *Float*: *0.000000* - usługa jest niedostępna; *seconds* - liczba sekund oczekiwania na odpowiedź z usługi.

Parametry:

- **service** - możliwe wartości: *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service.perf[ntp] #Ta pozycja może być użyta do testowania czasu odpowiedzi usługi NTP.
```

Attention:
Do obsługi SourceIP w prostych kontrolach LDAP (np. `net.tcp.service[ldap]`) wymagana jest wersja OpenLDAP 2.6.1 lub nowsza.

Przetwarzanie limitu czasu

Zabbix nie będzie przetwarzać prostego sprawdzenia dłużej niż przez liczbę sekund *Timeout* zdefiniowaną w formularzu konfiguracji pozycji. W przypadku pozycji VMware items Zabbix nie będzie przetwarzać prostego sprawdzenia dłużej niż przez liczbę sekund `Timeout` zdefiniowaną w pliku konfiguracyjnym Zabbix serwer lub proxy. W przypadku pozycji icmpping* wartości limitu czasu i liczby ponowień są określane bezpośrednio w kluczu pozycji i nie są zależne od globalnego parametru `Timeout`. Upewnij się, że wartości te są odpowiednio skonfigurowane w kluczu pozycji. Należy pamiętać, że dla pozycji icmpping* stosowany jest na sztywno ustawiony maksymalny możliwy limit czasu wynoszący 600 sekund (od Zabbix 7.4.6).

Pingi ICMP

Zabbix używa zewnętrznego narzędzia **fping** do przetwarzania pingów ICMP (**icmpping**, **icmppingloss**, **icmppingretry**, **icmppingsec**).

Instalacja

fping nie jest dołączony do Zabbix i musi zostać zainstalowany osobno:

- Różne platformy oparte na Uniksie mają pakiet fping w swoich domyślnych repozytoriach, ale nie jest on preinstalowany. W takim przypadku można użyć menedżera pakietów do zainstalowania fping.
- Zabbix udostępnia pakiety fping dla RHEL i jego pochodnych. Należy pamiętać, że pakiety te są dostarczane bez oficjalnego wsparcia.
- fping można również skompilować ze źródeł.

Konfiguracja

Określ lokalizację fping w parametrze *FpingLocation* pliku konfiguracyjnego serwera/proxy Zabbix (lub parametrze *Fping6Location* w przypadku używania adresów IPv6).

fping powinien być wykonywalny dla użytkownika, z uprawnieniami którego działa serwer/proxy Zabbix, a użytkownik ten powinien mieć wystarczające uprawnienia.

Zobacz także: Znane problemy dotyczące przetwarzania prostych testów z wersjami fping poniżej 3.10.

Wartości domyślne

Wartości domyślne, limity i opis wartości parametrów sprawdzania ICMP:

Parametr	Jednostka	Opis	Flaga fping	Wartości domyślne ustawiane przez		Dozwolone limity przez Zabbix	
				fping	**Zabbix**	**min**	**max**
packets	liczba	Liczba pakietów żądania wysyłanych do celu	-C		3	1	10000
interval	milisekundy	Czas oczekiwania między kolejnymi pakietami wysyłanymi do pojedynczego celu	-p	1000		20	bez ograniczeń
size	bajty	Rozmiar pakietu w bajtach 56 bajtów na x86, 68 bajtów na x86_64	-b	56 lub 68		24	65507
timeout	milisekundy	**fping v3.x** - limit czasu oczekiwania po wysłaniu ostatniego pakietu, zależny od flagi *-C* **fping v4.x** - indywidualny limit czasu dla każdego pakietu	-t	**fping v3.x** - 500 **fping v4.x** i nowsze - dziedziczone z flagi *-p*, ale nie więcej niż 2000		50	bez ograniczeń
retries	liczba	Liczba prób pingowania celu, nie licząc pierwszej próby	-r	3	1	0	bez ograniczeń
backoff factor	liczba	Liczba, przez którą mnożony jest czas oczekiwania przy każdym kolejnym żądaniu	-B	1.5	1.0	1.0	5.0

Wartości domyślne mogą się nieznacznie różnić w zależności od platformy i wersji.

Dodatkowo Zabbix używa opcji fping *-i interval ms* (nie należy mylić z parametrem pozycji *interval* wymienionym w tabeli powyżej, który odpowiada opcji fping *-p*) oraz *-S source IP address* (lub *-I* w starszych wersjach fping). Opcje te są wykrywane automatycznie przez uruchamianie testów z różnymi kombinacjami opcji. Zabbix próbuje wykryć minimalną wartość w milisekundach, której fping pozwala użyć z opcją *-i*, testując 3 wartości: 0, 1 i 10. Wartość, która jako pierwsza zakończy się powodzeniem, jest następnie używana w kolejnych testach ICMP. Proces ten jest wykonywany indywidualnie przez każdy proces ICMP pinger.

Automatycznie wykryte opcje fping są unieważniane co godzinę i wykrywane ponownie przy następnej próbie wykonania testu ICMP. Ustaw DebugLevel>=4, aby wyświetlić szczegóły tego procesu w pliku dziennika serwera lub proxy.

Zabbix zapisuje adresy IP do sprawdzenia przez dowolny z kluczy *icmpping** do pliku tymczasowego, który jest następnie przekazywany do fping. Jeśli pozycje mają różne parametry klucza, do jednego pliku zapisywane są tylko te z identycznymi parametrami klucza. Wszystkie adresy IP zapisane w jednym pliku będą sprawdzane przez fping równolegle, więc proces Zabbix ICMP pinger poświęci stałą ilość czasu niezależnie od liczby adresów IP w pliku.

1 Klucze pozycji monitorowania VMware

Lista kluczy pozycji monitorowania VMware została przeniesiona do sekcji Monitorowanie VMware.

3 SNMP agent

Omówienie

Możesz chcieć używać monitorowania SNMP na urządzeniach takich jak drukarki, przełączniki sieciowe, routery lub UPS, które zwykle obsługują SNMP i na których próba skonfigurowania pełnych systemów operacyjnych oraz agentów Zabbixa byłaby niepraktyczna.

Aby możliwe było pobieranie danych dostarczanych przez agentów SNMP na tych urządzeniach, serwer Zabbixa musi zostać początkowo skonfigurowany z obsługą SNMP przez podanie flagi `--with-net-snmp`. Zaleca się również zainstalowanie plików MIB, aby zapewnić wyświetlanie wartości pozycji w prawidłowym formacie. Bez plików MIB mogą wystąpić problemy z formatowaniem, takie jak wyświetlanie wartości w formacie HEX zamiast UTF-8 lub odwrotnie.

Sprawdzenia SNMP są wykonywane wyłącznie przez protokół UDP.

Demony serwera Zabbixa i proxy zapisują w logach wiersze podobne do poniższego, jeśli otrzymają nieprawidłową odpowiedź SNMP:

```
SNMP response from host "gateway" does not contain all of the requested variable bindings
```

Chociaż nie obejmują wszystkich problematycznych przypadków, są przydatne do identyfikowania pojedynczych urządzeń SNMP, dla których należy wyłączyć łączone żądania.

Serwer/proxy Zabbixa ponowi próbę do 5 razy dla pozycji SNMP `walk` i `get`. Mechanizm ponawiania nie ma zastosowania do błędów rozwiązywania DNS.

W przypadku starszych sprawdzeń SNMP (pojedynczy numer OID lub ciąg) serwer/proxy Zabbixa ponowi próbę co najmniej jeden raz po nieudanej próbie zapytania: albo za pośrednictwem mechanizmu ponawiania biblioteki SNMP, albo poprzez wewnętrzny mechanizm łączonego przetwarzania.

Warning:
Jeśli monitorujesz urządzenia SNMPv3, upewnij się, że msgAuthoritativeEngineID (znany również jako snmpEngineID lub "Engine ID") nigdy nie jest współdzielony przez dwa urządzenia. Zgodnie z RFC 2571 (sekcja 3.1.1.1) musi być unikalny dla każdego urządzenia.

Warning:
RFC3414 wymaga, aby urządzenia SNMPv3 zachowywały swoje engineBoots. Niektóre urządzenia tego nie robią, co powoduje, że ich komunikaty SNMP są odrzucane jako nieaktualne po ponownym uruchomieniu. W takiej sytuacji pamięć podręczną SNMP należy ręcznie wyczyścić na serwerze/proxy (używając -R snmp_cache_reload) albo należy ponownie uruchomić serwer/proxy.

Konfigurowanie monitorowania SNMP

Aby rozpocząć monitorowanie urządzenia za pomocą SNMP, należy wykonać następujące kroki:

Krok 1

Ustal ciąg SNMP (lub OID) pozycji, którą chcesz monitorować.

Aby uzyskać listę ciągów SNMP, użyj polecenia **snmpwalk** (części oprogramowania net-snmp, które powinno być zainstalowane w ramach instalacji Zabbix) lub równoważnego narzędzia:

```
snmpwalk -v 2c -c public <host IP> .
```

Ponieważ '2c' oznacza tutaj wersję SNMP, możesz również zastąpić ją wartością '1', aby wskazać SNMP Version 1 na urządzeniu.

Powinno to zwrócić listę ciągów SNMP i ich ostatnią wartość. Jeśli tak się nie stanie, możliwe jest, że SNMP 'community' różni się od standardowego 'public'; w takim przypadku trzeba ustalić, jaka ona jest.

Następnie możesz przejrzeć listę, aż znajdziesz ciąg, który chcesz monitorować, np. jeśli chcesz monitorować bajty przychodzące do przełącznika na porcie 3, użyj ciągu `IF-MIB::ifHCInOctets.3` z tego wiersza:

```
IF-MIB::ifHCInOctets.3 = Counter64: 3409739121
```

Możesz teraz użyć polecenia **snmpget**, aby ustalić numeryczny OID dla 'IF-MIB::ifHCInOctets.3':

```
snmpget -v 2c -c public -On <host IP> IF-MIB::ifHCInOctets.3
```

Zwróć uwagę, że ostatnia liczba w ciągu to numer portu, który chcesz monitorować. Zobacz także: Indeksy dynamiczne.

Powinno to zwrócić coś podobnego do poniższego:

```
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 3472126941
```

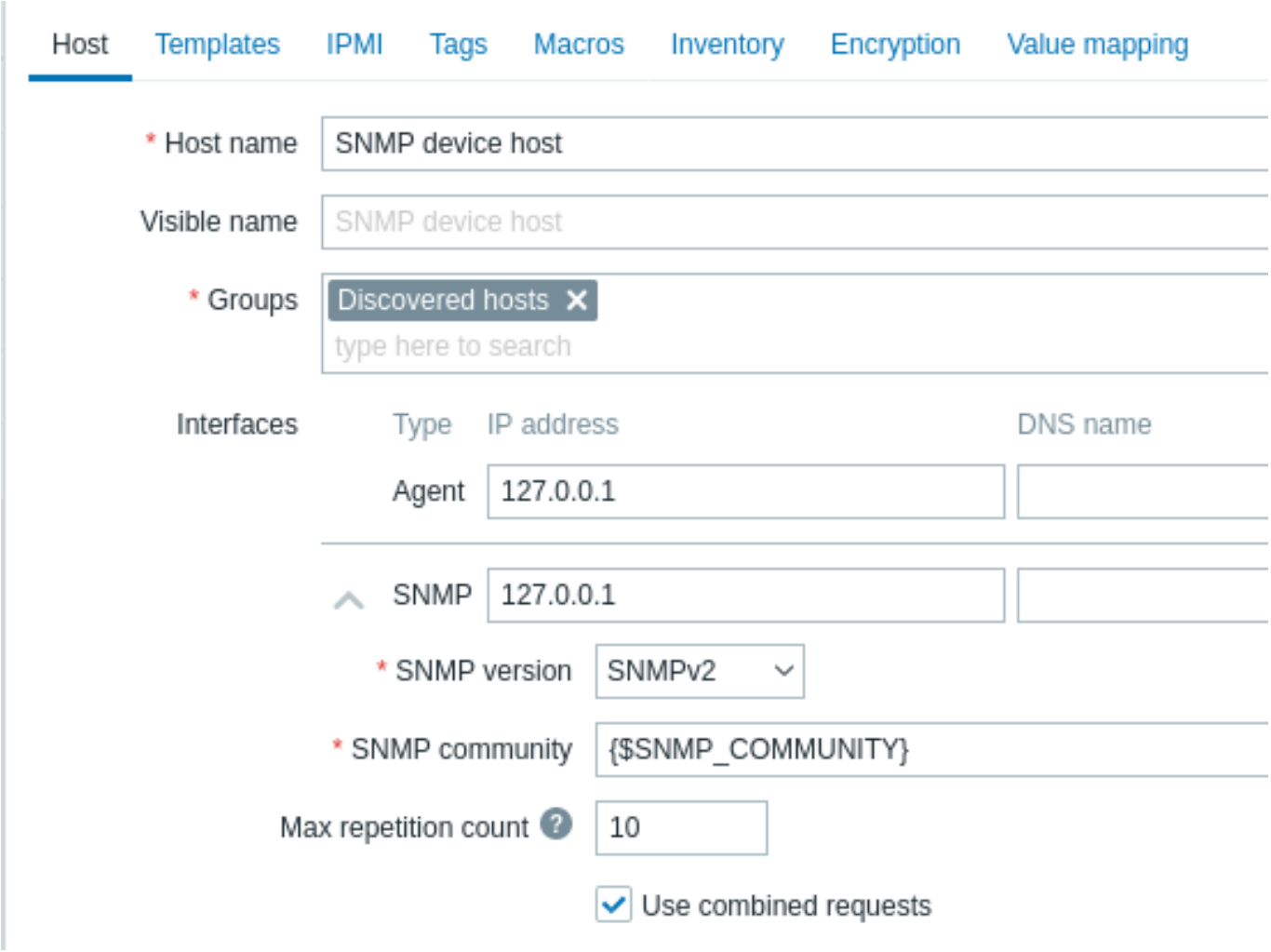
Ponownie, ostatnia liczba w OID to numer portu.

Note:
Niektóre z najczęściej używanych OID-ów SNMP są automatycznie tłumaczone na reprezentację numeryczną przez Zabbix.

W ostatnim przykładzie powyżej typ wartości to ”Counter64”, co wewnętrznie odpowiada typowi ASN_COUNTER64. Pełna lista obsługiwanych typów to ASN_COUNTER, ASN_COUNTER64, ASN_UINTEGER, ASN_UNSIGNED64, ASN_INTEGER, ASN_INTEGER64, ASN_FLOAT, ASN_DOUBLE, ASN_TIMETICKS, ASN_GAUGE, ASN_IPADDRESS, ASN_OCTET_STR oraz ASN_OBJECT_ID. Typy te w przybliżeniu odpowiadają wartościom ”Counter32”, ”Counter64”, ”UInteger32”, ”INTEGER”, ”Float”, ”Double”, ”Timeticks”, ”Gauge32”, ”IpAddress”, ”OCTET STRING”, ”OBJECT IDENTIFIER” w danych wyjściowych **snmpget**, ale mogą być również wyświetlane jako ”STRING”, ”Hex-STRING”, ”OID” i inne, w zależności od obecności wskazówki wyświetlania.

Krok 2

Utwórz hosta odpowiadającego urządzeniu.



Dodaj interfejs SNMP dla hosta:

- Wprowadź adres IP/nazwę DNS oraz numer portu.
- Wybierz *Wersję SNMP* z listy rozwijanej.
- Dodaj poświadczenia interfejsu zależnie od wybranej wersji SNMP:
  - SNMPv1, v2 wymagają tylko community (zwykle ’public’).
  - SNMPv3 wymaga bardziej szczegółowych opcji (zobacz poniżej).
- Określ wartość maksymalnej liczby powtórzeń (domyślnie: 10) dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs); tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Zwróć uwagę, że ustawienie zbyt wysokiej wartości może spowodować przekroczenie limitu czasu sprawdzania agenta SNMP.
- Zaznacz pole wyboru *Użyj żądań łączonych*, aby zezwolić na przetwarzanie łączone żądań SNMP (niezwiązane z natywnymi zbiorczymi żądaniami SNMP „walk” i „get”).

Parametr SNMPv3	Opis
Nazwa kontekstu	Wprowadź nazwę kontekstu, aby zidentyfikować pozycję w podsieci SNMP. Makra użytkownika są rozwijane w tym polu.
Nazwa bezpieczeństwa	Wprowadź nazwę bezpieczeństwa. Makra użytkownika są rozwijane w tym polu.

Parametr SNMPv3	Opis
Poziom bezpieczeństwa	Wybierz poziom bezpieczeństwa: **noAuthNoPriv** — nie są używane ani protokoły uwierzytelniania, ani prywatności **AuthNoPriv** — używany jest protokół uwierzytelniania, protokół prywatności nie jest używany **AuthPriv** — używane są zarówno protokoły uwierzytelniania, jak i prywatności
Protokół uwierzytelniania	Wybierz protokół uwierzytelniania — *MD5*, *SHA1*; z net-snmp 5.8 i nowszymi także *SHA224*, *SHA256*, *SHA384* lub *SHA512*.
Hasło uwierzytelniania	Wprowadź hasło uwierzytelniania. Makra użytkownika są rozwijane w tym polu.
Protokół prywatności	Wybierz protokół prywatności — *DES*, *AES128*, *AES192*, *AES256*, *AES192C* (Cisco) lub *AES256C* (Cisco). Zobacz uwagi dotyczące obsługi protokołu prywatności
Hasło prywatności	Wprowadź hasło prywatności. Makra użytkownika są rozwijane w tym polu.

W przypadku nieprawidłowych poświadczeń SNMPv3 (nazwa bezpieczeństwa, protokół/hasło uwierzytelniania, protokół prywatności):

- Zabbix otrzymuje ERROR z net-snmp, z wyjątkiem nieprawidłowego *Hasła prywatności*, w którym to przypadku Zabbix otrzymuje z net-snmp błąd TIMEOUT.
- Dostępność interfejsu SNMP zmieni się na czerwoną (niedostępny).

Note:
Zmiany w *Protokole uwierzytelniania*, *Haśle uwierzytelniania*, *Protokole prywatności* lub *Haśle prywatności*, wprowadzone bez zmiany *Nazwy bezpieczeństwa*, są zwykle stosowane automatycznie po zaktualizowaniu odpowiedniego interfejsu SNMPv3 w Zabbixie. W przypadkach, gdy zmieniana jest również *Nazwa bezpieczeństwa*, wszystkie parametry zostaną zaktualizowane natychmiast.

Możesz użyć jednego z dostarczonych szablonów SNMP, który automatycznie doda zestaw pozycji. Przed użyciem szablonu sprawdź, czy jest on zgodny z hostem.

Kliknij *Dodaj*, aby zapisać hosta.

Obsługa protokołów prywatności

W zależności od systemu operacyjnego i konfiguracji net-snmp niektóre protokoły prywatności mogą nie być dostępne:

- W niektórych nowszych systemach operacyjnych (na przykład RHEL9) obsługa DES została usunięta z pakietu net-snmp.
- Protokoły szyfrowania AES192 i silniejsze nie są obsługiwane od razu po instalacji w systemach operacyjnych starszych niż RHEL 8, CentOS 8, Oracle Linux 8, Debian 12, Ubuntu LTS 22.04, openSUSE Leap 15.5.

Aby sprawdzić, czy biblioteka net-snmp obsługuje AES192+, użyj jednej z poniższych opcji:

1. `net-snmp-config`:

```
net-snmp-config --configure-options
```

Jeśli dane wyjściowe zawierają `--enable-blumenthal-aes`, AES192+ jest obsługiwane.

Pamiętaj, że net-snmp-config jest częścią pakietu deweloperskiego dla SNMP (libsnmp-dev dla Debian/Ubuntu, net-snmp-devel dla CentOS/RHEL/OL/SUSE) i może nie być domyślnie zainstalowany.

2. `snmpget`:

```
snmpget -v 3 -x AES-256
```

Jeśli dane wyjściowe zawierają `Invalid privacy protocol specified after -3x flag: AES-256`, AES192+ nie jest obsługiwane. Jeśli dane wyjściowe zawierają `No hostname specified.`, AES192+ nie jest obsługiwane.

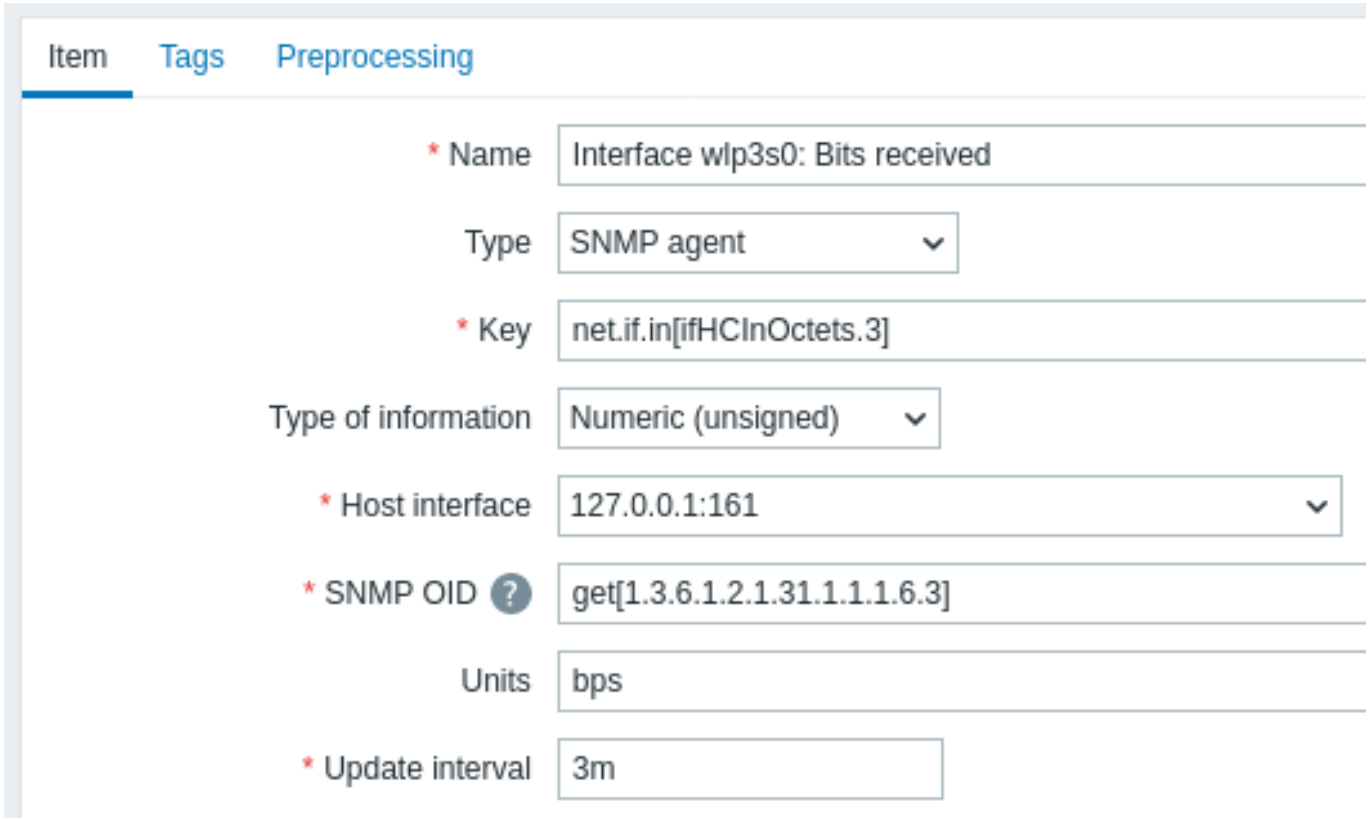
Jeśli biblioteka net-snmp nie obsługuje protokołów AES192 i wyższych, skompiluj ponownie net-snmp z opcją `--enable-blumenthal-aes`, a następnie skompiluj ponownie serwer Zabbix, podając opcję `--with-net-snmp=/home/user/yourcustomnetsnmp/bin/net-snmp-co`

Krok 3

Utwórz pozycję do monitorowania.

Wróć teraz do Zabbix i kliknij *Items* dla hosta SNMP, który utworzyłeś wcześniej. W zależności od tego, czy podczas tworzenia hosta użyłeś szablonu, będziesz mieć albo listę pozycji SNMP powiązanych z hostem, albo po prostu pustą listę. Założymy, że pozycję utworzysz samodzielnie, korzystając z informacji właśnie zebranych za pomocą snmpwalk i snmpget, więc kliknij *Create item*.

Wypełnij wymagane parametry w nowym formularzu pozycji:



Parameter	Description
Name	Wprowadź nazwę pozycji.
Type	Wybierz tutaj **SNMP agent**.
Key	Wprowadź klucz w sposób opisowy.
Host interface	Upewnij się, że wybierasz interfejs SNMP, np. dla przełącznika/routera.

Parameter	Description
SNMP OID	Użyj jednego z obsługiwanych formatów, aby wprowadzić wartość lub wartości OID: **walk[OID1,OID2,...]** - pobiera poddrzewo wartości. Na przykład: `walk[1.3.6.1.2.1.2.2.1.2,1.3.6.1.2.1.2.2.1.3]`. Ta opcja korzysta z natywnych zbiorczych zapytań SNMP (GetBulkRequest-PDU) asynchronicznie. Ustawienia limitu czasu dla tej pozycji można skonfigurować w formularzu konfiguracji pozycji. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest niedostępne, ponieważ zostanie wykonanych do 5 ponowień, jeśli wcześniejsze zakończą się przekroczeniem limitu czasu lub niepowodzeniem (np. limit 3 sekund może skutkować czasem oczekiwania 15 sekund). Możesz użyć tego jako pozycji nadrzędnej, z pozycjami zależnymi, które wyodrębniają dane z pozycji nadrzędnej za pomocą przetwarzania wstępnego. Możliwe jest określenie wielu OID w jednym snmp walk, na przykład `walk[OID1,OID2,...]`, aby asynchronicznie przetwarzać jeden OID naraz. Jeśli żądanie bulk nie zwróci wyników, podejmowana jest próba pobrania pojedynczego rekordu bez użycia bulk request. Obsługiwane są nazwy MIB jako parametry; zatem `walk[1.3.6.1.2.1.2.2.1.2]` i `walk[ifDescr]` zwrócą ten sam wynik. Jeśli określono kilka OID/MIB, tj. `walk[ifDescr,ifType,ifPhysAddress]`, wynik będzie stanowił połączoną listę. Żądania GetBulk są używane z interfejsami SNMPv2 i v3, a GetNext dla interfejsów SNMPv1; maksymalna liczba powtórzeń dla żądań bulk jest konfigurowana na poziomie interfejsu. Parametr maksymalnej liczby powtórzeń wpływa na żądania bulk, określając maksymalną liczbę OID zwracanych w jednej odpowiedzi bulk. Wyższa wartość skutkuje większymi odpowiedziami bulk, zmniejszając liczbę wymaganych transmisji. Jednak nie wszystkie urządzenia mogą obsługiwać bardzo wysokie wartości, co może powodować problemy. Ta pozycja zwraca wynik narzędzia snmpwalk z parametrami -Oe -Ot -On. Możesz użyć tej pozycji jako pozycji nadrzędnej w odkrywaniu SNMP. **get[OID]** - pobiera *pojedynczą* wartość asynchronicznie. Na przykład: `get[1.3.6.1.2.1.31.1.1.1.6.3]` Ustawienia limitu czasu dla tej pozycji można skonfigurować w formularzu konfiguracji pozycji. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest niedostępne, ponieważ zostanie wykonanych do 5 ponowień, jeśli wcześniejsze zakończą się przekroczeniem limitu czasu lub niepowodzeniem (np. limit 3 sekund może skutkować czasem oczekiwania 15 sekund). **OID** - (starsza metoda) wprowadź pojedynczy tekstowy lub numeryczny OID, aby pobrać jedną wartość synchronicznie, opcjonalnie w połączeniu z innymi wartościami. Na przykład: `1.3.6.1.2.1.31.1.1.1.6.3`. W przypadku tej opcji czas sprawdzania pozycji będzie równy wartości ustawionej w pliku konfiguracyjnym serwera. **Zaleca się** używanie pozycji `walk[OID]` i `get[OID]` dla lepszej wydajności. Wszystkie pozycje `walk[OID]` i `get[OID]` są wykonywane asynchronicznie - nie jest wymagane otrzymanie odpowiedzi na jedno żądanie przed rozpoczęciem kolejnych sprawdzeń. Rozwiązywanie nazw DNS również odbywa się asynchronicznie. Maksymalna współbieżność asynchronicznych sprawdzeń wynosi 1000 (zdefiniowana przez parametr MaxConcurrentChecksPerPoller). Liczba asynchronicznych pollerów SNMP jest określona przez parametr StartSNMPPollers. Pamiętaj, że dla statystyk ruchu sieciowego zwracanych dowolną z metod należy dodać krok *Change per second* w zakładce *Preprocessing*; w przeciwnym razie otrzymasz wartość skumulowaną z urządzenia SNMP zamiast najnowszej zmiany.

Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Teraz zapisz pozycję i przejdź do *Monitoring* > *Latest data* dla danych SNMP.

Przykład 1

Ogólny przykład:

Parametr	Opis
OID	1.2.3.45.6.7.8.0 (lub .1.2.3.45.6.7.8.0)
Key	<Unikalny ciąg znaków używany jako odniesienie do wyzwalaczy> Na przykład „my_param”.

Zwróć uwagę, że OID może być podany zarówno w postaci numerycznej, jak i tekstowej. Jednak w niektórych przypadkach tekstowy OID musi zostać przekonwertowany do postaci numerycznej. W tym celu można użyć narzędzia snmpget:

```
snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Przykład 2

Monitorowanie czasu działania:

Parametr	Opis
OID	MIB::sysUpTime.0
Key	router.uptime
Typ wartości	Float
Jednostki	uptime
Krok przetwarzania wstępnego: Mnożnik niestandardowy	0.01

Natywne żądania zbiorcze SNMP

Pozycja **walk[OID1,OID2,...]** umożliwia użycie natywnej funkcjonalności SNMP do żądań zbiorczych (GetBulkRequest-PDUs), dostępnej w wersjach SNMP 2/3.

Żądanie GetBulk w SNMP wykonuje wiele żądań GetNext i zwraca wynik w jednej odpowiedzi. Może to być używane zarówno dla zwykłych pozycji SNMP, jak i dla wykrywania SNMP, aby zminimalizować liczbę wymian z siecią.

Pozycja SNMP **walk[OID1,OID2,...]** może być używana jako pozycja nadrzędna, która zbiera dane w jednym żądaniu, wraz z pozycjami zależnymi, które analizują odpowiedź w razie potrzeby przy użyciu preprocessing.

Należy pamiętać, że użycie natywnych żądań zbiorczych SNMP nie jest związane z opcją łączenia żądań SNMP, która jest własnym sposobem Zabbix na łączenie wielu żądań SNMP (zobacz następną sekcję).

Dla zbiorczych pozycji SNMP zostanie wykonanych maksymalnie pięć ponowień, aby uniknąć niepowodzenia, jeśli jeden z pakietów zostanie utracony. Limit czasu dla pozycji SNMP z `get` i `walk` (ustawiany w formularzu konfiguracji pozycji) jest ustawiany dla całej sesji. Limit czasu jest stosowany niezależnie od tego, czy dane zostały pobrane w całości; jeśli dane zostaną odebrane częściowo (na przykład dane zostaną pomyślnie zebrane tylko dla jednego z wielu OID), pozycja stanie się nieobsługiwana z komunikatem „Only partial data received”. Jeśli limit czasu zostanie osiągnięty, nastąpi ponowienie, limit czasu zostanie zresetowany, a ostatnie żądanie zostanie wysłane ponownie, co pozwoli kontynuować sesję od ostatniego żądania, jeśli pojedynczy pakiet został utracony lub dotarł zbyt późno. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest nieosiągalne, ponieważ zostanie wykonanych do 5 ponowień, jeśli wcześniejsze przekroczą limit czasu lub zakończą się niepowodzeniem (np. 3-sekundowy limit czasu może skutkować 15-sekundowym czasem oczekiwania).

Wewnętrzne działanie przetwarzania łączonego

serwer Zabbix i proxy mogą odpytywać urządzenia SNMP o wiele wartości w jednym żądaniu. Wpływa to na kilka typów pozycji SNMP:

- zwykłe pozycje SNMP
- pozycje SNMP z dynamicznymi indeksami
- reguły niskopoziomowego wykrywania SNMP

Wszystkie pozycje SNMP na pojedynczym interfejsie o identycznych parametrach są planowane do odpytywania w tym samym czasie. Pierwsze dwa typy pozycji są pobierane przez pollery partiami po maksymalnie 128 pozycji, natomiast reguły niskopoziomowego wykrywania są przetwarzane pojedynczo, tak jak wcześniej.

Na niższym poziomie wykonywane są dwa rodzaje operacji służących do odpytywania wartości: pobieranie wielu określonych obiektów oraz przechodzenie drzewa OID.

Do „pobierania” używane jest GetRequest-PDU z maksymalnie 128 powiązaniami zmiennych. Do „przechodzenia” używane jest GetNextRequest-PDU dla SNMPv1, a dla SNMPv2 i SNMPv3 używane jest GetBulkRequest z polem „max-repetitions” o wartości maksymalnie 128.

W związku z tym korzyści z przetwarzania łączonego dla każdego typu pozycji SNMP przedstawiają się następująco:

- zwykłe pozycje SNMP korzystają z usprawnień „pobierania”;
- pozycje SNMP z dynamicznymi indeksami korzystają zarówno z usprawnień „pobierania”, jak i „przechodzenia”: „pobieranie” jest używane do weryfikacji indeksów, a „przechodzenie” do budowania pamięci podręcznej;
- reguły niskopoziomowego wykrywania SNMP korzystają z usprawnień „przechodzenia”.

Istnieje jednak problem techniczny polegający na tym, że nie wszystkie urządzenia są w stanie zwrócić 128 wartości na żądanie. Niektóre zawsze zwracają prawidłową odpowiedź, ale inne albo odpowiadają błędem „tooBig(1)”, albo nie odpowiadają wcale, gdy potencjalna odpowiedź przekracza pewien limit.

Aby znaleźć optymalną liczbę obiektów do odpytywania dla danego urządzenia, Zabbix stosuje następującą strategię. Zaczyna ostrożnie od odpytywania 1 wartości w żądaniu. Jeśli to się powiedzie, odpytywane są 2 wartości w żądaniu. Jeśli to ponownie się powiedzie, odpytywane są 3 wartości w żądaniu, a następnie proces jest kontynuowany w podobny sposób przez mnożenie liczby odpytywanych obiektów przez 1,5, co daje następującą sekwencję rozmiarów żądań: 1, 2, 3, 4, 6, 9, 13, 19, 28, 42, 63, 94, 128.

Jednak gdy urządzenie odmówi zwrócenia prawidłowej odpowiedzi (na przykład dla 42 zmiennych), Zabbix robi dwie rzeczy.

Po pierwsze, dla bieżącej partii pozycji zmniejsza o połowę liczbę obiektów w pojedynczym żądaniu i odpytuje 21 zmiennych. Jeśli urządzenie działa, to zapytanie powinno zadziałać w zdecydowanej większości przypadków, ponieważ wiadomo, że 28 zmiennych działało, a 21 to znacznie mniej. Jeśli jednak to również się nie powiedzie, Zabbix wraca do odpytywania wartości pojedynczo. Jeśli nadal się nie powiedzie, oznacza to, że urządzenie zdecydowanie nie odpowiada i rozmiar żądania nie jest problemem.

Drugą rzeczą, jaką Zabbix robi dla kolejnych partii pozycji, jest rozpoczęcie od ostatniej pomyślnej liczby zmiennych (28 w naszym przykładzie) i dalsze zwiększanie rozmiaru żądań o 1 aż do osiągnięcia limitu. Na przykład, zakładając, że największy rozmiar odpowiedzi to 32 zmienne, kolejne żądania będą miały rozmiary 29, 30, 31, 32 i 33. Ostatnie żądanie zakończy się niepowodzeniem i Zabbix nigdy więcej nie wyśle żądania o rozmiarze 33. Od tego momentu Zabbix będzie odpytywać to urządzenie o maksymalnie 32 zmienne.

Jeśli duże zapytania zawodzą przy tej liczbie zmiennych, może to oznaczać jedną z dwóch rzeczy. Nie da się dokładnie określić kryteriów, których urządzenie używa do ograniczania rozmiaru odpowiedzi, ale staramy się to przybliżyć za pomocą liczby zmiennych. Pierwsza możliwość jest taka, że ta liczba zmiennych znajduje się blisko rzeczywistego limitu rozmiaru odpowiedzi urządzenia w ogólnym przypadku: czasami odpowiedź jest mniejsza od limitu, a czasami go przekracza. Druga możliwość jest taka, że pakiet UDP w jednym z kierunków po prostu został utracony. Z tych powodów, jeśli Zabbix otrzyma nieudane zapytanie, zmniejsza maksymalną liczbę zmiennych, aby spróbować wejść głębiej w bezpieczny zakres pracy urządzenia, ale robi to najwyżej dwa razy.

W powyższym przykładzie, jeśli zapytanie z 32 zmiennymi akurat się nie powiedzie, Zabbix zmniejszy tę liczbę do 31. Jeśli to również się nie powiedzie, Zabbix zmniejszy tę liczbę do 30. Jednak Zabbix nie zmniejszy tej liczby poniżej 30, ponieważ założy, że dalsze niepowodzenia wynikają z utraty pakietów UDP, a nie z limitu urządzenia.

Jeśli jednak urządzenie nie potrafi poprawnie obsługiwać żądań łączonych z innych powodów i opisana powyżej heurystyka nie działa, dla każdego interfejsu dostępne jest ustawienie „Use combined requests”, które pozwala wyłączyć żądania łączone dla tego urządzenia.

Jeśli żądania łączone powodują częściowe lub nieprawidłowo sformowane odpowiedzi, które skutkują błędnymi obliczeniami wartości na sekundę (delta) (na przykład pozornymi skokami liczników interfejsu), wyłącz opcję *Use combined requests* dla danego interfejsu, aby wymusić osobne zapytania dla każdej pozycji; często zapobiega to fałszywym skokom. Alternatywnie rozważ użycie asynchronicznych pozycji `get[]` lub `walk[]`, które są wykonywane asynchronicznie i nie podlegają grupowaniu według interfejsu przez opcję *Use combined requests* — mogą być używane zamiast starszych synchronicznych kontroli OID, aby uniknąć problemów związanych z żądaniami łączonymi. Szukaj wpisów w logach serwera/proxy podobnych do pokazanego w sekcji Przegląd, aby pomóc w identyfikacji urządzeń, których dotyczy problem.

Dodatkowo, jeśli interfejs często staje się niedostępny, może być konieczne zwiększenie parametru `UnavailableDelay` w plikach konfiguracyjnych serwera Zabbix lub proxy Zabbix, aby zmniejszyć częstotliwość żądań. Pozycje mogą stać się nieobsługiwane, jeśli podczas wykrywania lub przejść OID zostaną odebrane częściowe dane.

1 Indeksy dynamiczne

Przegląd

Chociaż wymagany numer indeksu (na przykład interfejsu sieciowego) można znaleźć wśród OID-ów SNMP, czasami nie można całkowicie polegać na tym, że numer indeksu zawsze pozostanie taki sam.

Numery indeksów mogą być dynamiczne — mogą zmieniać się z upływem czasu, a w konsekwencji Twoja pozycja może przestać działać.

Aby uniknąć takiej sytuacji, można zdefiniować OID, który uwzględnia możliwość zmiany numeru indeksu.

Na przykład, jeśli chcesz pobrać wartość indeksu do dołączenia do **ifInOctets**, która odpowiada interfejsowi **GigabitEthernet0/1** na urządzeniu Cisco, użyj następującego OID:

```
ifInOctets["index","ifDescr","GigabitEthernet0/1"]
```

Składnia

Używana jest specjalna składnia dla OID:

<OID danych>[”index”,”<bazowy OID indeksu>”,”<szukany ciąg znaków>”]

Parametr	Opis
OID danych	Główny OID używany do pobierania danych dla pozycji.
index	Metoda przetwarzania. Obecnie obsługiwana jest jedna metoda: **index** - wyszukuje indeks i dołącza go do OID danych
bazowy OID indeksu	Ten OID zostanie sprawdzony w celu uzyskania wartości indeksu odpowiadającej ciągowi znaków.
szukany ciąg znaków	Ciąg znaków używany do dokładnego dopasowania wartości podczas wyszukiwania. Wielkość liter ma znaczenie.

Przykład

Pobieranie użycia pamięci procesu *apache*.

Jeśli używana jest ta składnia OID:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index","HOST-RESOURCES-MIB::hrSWRunPath", "/usr/sbin/apache2"]
```

numer indeksu zostanie tutaj wyszukany:

```
...
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"
...
```

Teraz mamy indeks 5388. Indeks zostanie dołączony do OID danych, aby otrzymać interesującą nas wartość:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes
```

Buforowanie wyszukiwania indeksów

Gdy żądana jest pozycja dynamicznego indeksu, Zabbix pobiera i buforuje całą tabelę SNMP pod bazowym OID dla indeksu, nawet jeśli dopasowanie zostałoby znalezione wcześniej. Jest to wykonywane na wypadek, gdyby inna pozycja odwoływała się później do tego samego bazowego OID — Zabbix wyszuka indeks w pamięci podręcznej zamiast ponownie odpytywać monitorowany host. Należy pamiętać, że każdy proces pollera używa oddzielnej pamięci podręcznej.

We wszystkich kolejnych operacjach pobierania wartości weryfikowany jest tylko znaleziony indeks. Jeśli nie uległ zmianie, żądana jest wartość. Jeśli uległ zmianie, pamięć podręczna jest przebudowywana — każdy poller, który napotka zmieniony indeks, ponownie przechodzi tabelę indeksów SNMP.

2 Specjalne OID

Niektóre z najczęściej używanych OID SNMP są automatycznie tłumaczone przez Zabbix na reprezentację numeryczną. Na przykład **ifIndex** jest tłumaczone na **1.3.6.1.2.1.2.2.1.1**, a **ifIndex.0** jest tłumaczone na **1.3.6.1.2.1.2.2.1.1.0**.

Tabela zawiera listę specjalnych OID.

Specjalny OID	Identyfikator	Opis
ifIndex	1.3.6.1.2.1.2.2.1.1	Unikalna wartość dla każdego interfejsu.
ifDescr	1.3.6.1.2.1.2.2.1.2	Ciąg tekstowy zawierający informacje o interfejsie. Ten ciąg powinien zawierać nazwę producenta, nazwę produktu oraz wersję interfejsu sprzętowego.
ifType	1.3.6.1.2.1.2.2.1.3	Typ interfejsu, rozróżniany zgodnie z protokołem(ami) fizycznym(i)/łącza bezpośrednio „poniżej” warstwy sieciowej w stosie protokołów.
ifMtu	1.3.6.1.2.1.2.2.1.4	Rozmiar największego datagramu, który może zostać wysłany/odebrany przez interfejs, określony w oktetach.
ifSpeed	1.3.6.1.2.1.2.2.1.5	Szacunkowa bieżąca przepustowość interfejsu w bitach na sekundę.

Specjalny OID	Identyfikator	Opis
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	Adres interfejsu na warstwie protokołu bezpośrednio ‘poniżej’ warstwy sieciowej w stosie protokołów.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	Bieżący administracyjny stan interfejsu.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	Bieżący operacyjny stan interfejsu.
ifInOctets	1.3.6.1.2.1.2.2.1.10	Łączna liczba oktetów odebranych przez interfejs, w tym znaków ramkowania.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	Liczba pakietów unicast podsieci dostarczonych do protokołu wyższej warstwy.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Liczba pakietów nie-unicast (tj. rozgłoszeniowych podsieci lub multicastowych podsieci) dostarczonych do protokołu wyższej warstwy.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	Liczba pakietów przychodzących, które zostały wybrane do odrzucenia, mimo że nie wykryto błędów, które uniemożliwiałyby ich dostarczenie do protokołu wyższej warstwy. Jednym z możliwych powodów odrzucenia takiego pakietu może być zwolnienie miejsca w buforze.
ifInErrors	1.3.6.1.2.1.2.2.1.14	Liczba pakietów przychodzących zawierających błędy uniemożliwiające ich dostarczenie do protokołu wyższej warstwy.
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	Liczba pakietów odebranych przez interfejs, które zostały odrzucone z powodu nieznanego lub nieobsługiwanego protokołu.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	Łączna liczba oktetów wysłanych z interfejsu, w tym znaków ramkowania.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	Łączna liczba pakietów, których wysłania zażądały protokoły wyższej warstwy i które nie były adresowane do adresu multicast lub broadcast na tej podwarstwie, w tym tych, które zostały odrzucone lub niewysłane.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	Łączna liczba pakietów, których wysłania zażądały protokoły wyższej warstwy i które były adresowane do adresu multicast lub broadcast na tej podwarstwie, w tym tych, które zostały odrzucone lub niewysłane.
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	Liczba pakietów wychodzących, które zostały wybrane do odrzucenia, mimo że nie wykryto błędów, które uniemożliwiałyby ich transmisję. Jednym z możliwych powodów odrzucenia takiego pakietu może być zwolnienie miejsca w buforze.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	Liczba pakietów wychodzących, które nie mogły zostać przesłane z powodu błędów.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	Długość kolejki pakietów wyjściowych (w pakietach).

3 pliki MIB

Wprowadzenie

MIB oznacza bazę informacji zarządzania (Management Information Base). Pliki MIB umożliwiają używanie tekstowej reprezentacji OID (Object Identifier). Podczas monitorowania urządzeń SNMP za pomocą Zabbix można używać surowych OID, ale jeśli wygodniej jest używać reprezentacji tekstowej, należy zainstalować pliki MIB.

Na przykład

```
ifHCOutOctets
```

jest tekstową reprezentacją OID

```
1.3.6.1.2.1.31.1.1.1.10
```

Instalacja plików MIB

W systemach opartych na Debianie:

```
apt install snmp-mibs-downloader
download-mibs
```

W systemach opartych na RedHat:

```
dnf install net-snmp-libs
```

Włączanie plików MIB

W systemach opartych na RedHat pliki MIB powinny być domyślnie włączone. W systemach opartych na Debian należy edytować plik `/etc/snmp/snmp.conf` i zakomentować linię `mibs :`

```
# Ponieważ pakiety snmp są dostarczane bez plików MIB z powodów licencyjnych, ładowanie
# plików MIB jest domyślnie wyłączone. Jeśli dodano pliki MIB, można ponownie włączyć
# ich ładowanie, zakomentowując poniższą linię.
mibs :
```

Testowanie plików MIB

Testowanie SNMP MIB można wykonać za pomocą narzędzia `snmpwalk`. Jeśli nie jest ono zainstalowane, skorzystaj z poniższych instrukcji.

W systemach opartych na Debianie:

```
apt install snmp
```

W systemach opartych na RedHat:

```
dnf install net-snmp-utils
```

Następnie poniższe polecenie nie powinno zwracać błędu podczas odpytywania urządzenia sieciowego:

```
$ 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
[...]
```

Używanie MIB-ów w Zabbixie

Najważniejsze, o czym należy pamiętać, to fakt, że procesy Zabbix nie są informowane o zmianach wprowadzonych w plikach MIB. Dlatego po każdej zmianie należy ponownie uruchomić serwer lub proxy Zabbix, na przykład:

```
systemctl restart zabbix-server
```

Po tym zmiany wprowadzone w plikach MIB zaczną obowiązywać.

Używanie niestandardowych plików MIB

Istnieją standardowe pliki MIB dostarczane z każdą dystrybucją GNU/Linux. Jednak niektórzy producenci urządzeń udostępniają własne.

Załóżmy, że chcesz użyć pliku MIB CISCO-SMI. Poniższe instrukcje pobiorą go i zainstalują:

```
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/loca
cp /tmp/CISCO-SMI.my /usr/local/share/snmp/mibs
```

Teraz powinno być możliwe jego użycie. Spróbuj przetłumaczyć nazwę obiektu *ciscoProducts* z pliku MIB na OID:

```
snmptranslate -IR -On CISCO-SMI::ciscoProducts
.1.3.6.1.4.1.9.1
```

Jeśli zamiast OID otrzymasz błędy, upewnij się, że wszystkie poprzednie polecenia zostały wykonane bez błędów.

Tłumaczenie nazwy obiektu zadziałało, więc możesz już używać niestandardowego pliku MIB. Zwróć uwagę na prefiks nazwy MIB (*CISCO-SMI::*) użyty w zapytaniu. Będzie on potrzebny zarówno podczas korzystania z narzędzi wiersza poleceń, jak i Zabbix.

Nie zapomnij zrestartować serwera/proxy Zabbix przed użyciem tego pliku MIB w Zabbix.

> **Attention:**
> Pamiętaj, że pliki MIB mogą mieć zależności. Oznacza to, że jeden MIB może wymagać innego. Aby spełnić te zależności, musisz zainstalować wszystkie powiązane pliki MIB.

4 SNMP trap

Omówienie

Odbieranie pułapek SNMP jest przeciwieństwem odpytywania urządzeń obsługujących SNMP.

W tym przypadku informacje są wysyłane z urządzenia obsługującego SNMP do snmptrapd i są zbierane lub „przechwytywane" przez serwer Zabbix albo proxy Zabbix z pliku.

Zazwyczaj pułapki są wysyłane po zmianie pewnego warunku, a agent łączy się z serwerem na porcie 162 (w przeciwieństwie do portu 161 po stronie agenta, który jest używany do zapytań). Używanie pułapek może wykrywać krótkotrwałe problemy, które występują pomiędzy interwałami zapytań i mogą zostać pominięte przez dane z zapytań.

Odbieranie pułapek SNMP w Zabbix zostało zaprojektowane do współpracy z **snmptrapd** oraz jednym z mechanizmów przekazywania pułapek do Zabbix - albo skryptem Bash lub Perl, albo SNMPTT.

Note:
Najprostszym sposobem skonfigurowania monitorowania pułapek po skonfigurowaniu Zabbix jest użycie rozwiązania ze skryptem Bash, ponieważ Perl i SNMPTT często brakuje w nowoczesnych dystrybucjach i wymagają bardziej złożonej konfiguracji. Jednak to rozwiązanie używa skryptu skonfigurowanego jako `traphandle`. Dla lepszej wydajności w systemach produkcyjnych użyj wbudowanego rozwiązania Perl (albo skryptu z opcją `do perl`, albo SNMPTT).

Przebieg odbierania pułapki:

1. `snmptrapd` odbiera pułapkę.
2. `snmptrapd` przekazuje pułapkę do skryptu odbierającego (Bash, Perl) lub SNMPTT.
3. Odbiornik analizuje, formatuje i zapisuje pułapkę do pliku.
4. Trapper SNMP Zabbix odczytuje i analizuje plik z pułapkami.
5. Dla każdej pułapki Zabbix znajduje wszystkie pozycje trappera SNMP z interfejsami hosta pasującymi do otrzymanego adresu pułapki. Należy pamiętać, że podczas dopasowywania używany jest tylko wybrany *IP* lub *DNS* w interfejsie hosta.
6. Dla każdej znalezionej pozycji pułapka jest porównywana z wyrażeniem regularnym w `snmptrap[regexp]`. Pułapka jest ustawiana jako wartość **wszystkich** dopasowanych pozycji. Jeśli nie zostanie znaleziona żadna pasująca pozycja i istnieje pozycja `snmptrap.fallback`, pułapka jest ustawiana jako jej wartość.
7. Jeśli pułapka nie została ustawiona jako wartość żadnej pozycji, Zabbix domyślnie rejestruje niedopasowaną pułapkę. (Jest to konfigurowane przez opcję *Log unmatched SNMP traps* w *Administration* > *General* > *Other*.)

Uwagi dotyczące przełączenia awaryjnego HA

Podczas przełączania węzła wysokiej dostępności (HA) Zabbix będzie kontynuował przetwarzanie od ostatniego rekordu w ramach ostatniego znacznika czasu ISO 8601; jeśli ten sam rekord nie zostanie znaleziony, do określenia ostatniej pozycji zostanie użyty tylko znacznik czasu.

Konfigurowanie trapów SNMP

Ten typ pozycji wymaga następującej konfiguracji frontend:

Krok 1: Utwórz interfejs SNMP dla hosta

1. W *Data collection* > *Hosts*, utwórz host lub edytuj istniejący.
2. W polu *Interfaces* wybierz typ interfejsu *SNMP*.
3. Wprowadź adres IP/nazwę DNS oraz numer portu.
4. Wybierz *wersję SNMP* z listy rozwijanej: (*SNMPv1*, *SNMPv2*, lub *SNMPv3*) i dodaj dane uwierzytelniające interfejsu w zależności od wybranej wersji SNMP.

Adres z każdego odebranego trapu będzie porównywany z adresami IP i DNS wszystkich interfejsów SNMP, aby znaleźć odpowiadające hosty.

Krok 2: Skonfiguruj pozycję

1. Dla hosta utwórz pozycję lub edytuj istniejącą.
2. W polu *Key* użyj jednego z kluczy trapów SNMP:
   - `snmptrap[regexp]` - przechwytuje wszystkie trapy SNMP, które pasują do wyrażenia regularnego określonego w parametrze `regexp`; jeśli `regexp` nie jest określony, przechwytuje dowolny trap.
 W parametrze są obsługiwane makra użytkownika i globalne wyrażenia regularne.
 Wartość zwracana: trap SNMP.
 Tę pozycję można ustawić tylko dla interfejsów SNMP.
   - `snmptrap.fallback` - przechwytuje wszystkie trapy SNMP, które nie zostały przechwycone przez żadne pozycje `snmptrap[]` dla tego interfejsu.
 Wartość zwracana: trap SNMP.
 Tę pozycję można ustawić tylko dla interfejsów SNMP.

Note:
Obecnie nie jest obsługiwane dopasowywanie wielowierszowych wyrażeń regularnych.

3. Ustaw *Type of information* na *Log*, aby znaczniki czasu mogły zostać przeanalizowane. Inne formaty, takie jak *Numeric*, są również akceptowalne, ale mogą wymagać niestandardowego modułu obsługi trapów.

Konfigurowanie monitorowania pułapek SNMP

Konfigurowanie serwera/proxy Zabbix

Aby odczytywać trapy, serwer Zabbix lub proxy musi być skonfigurowany tak, aby uruchamiał proces SNMP trapper i wskazywał plik trapów, do którego zapisuje SNMPTT lub odbiornik trapów Bash/Perl. W tym celu należy edytować plik konfiguracyjny (`zabbix\_server.conf` lub `zabbix\_proxy.conf`):

```
StartSNMPTrapper=1
SNMPTrapperFile=[TRAP FILE]
```

Warning:
Jeśli używany jest parametr systemd `PrivateTmp`, ten plik prawdopodobnie nie będzie działał w `/tmp`.

Konfigurowanie odbiornika trapów Bash

Wymagania: tylko snmptrapd.

Skrypt odbiornika trapów Bash script może być użyty do przekazywania trapów do serwera Zabbix z snmptrapd za pomocą pliku trapper. Aby go skonfigurować, dodaj opcję `traphandle` do pliku konfiguracyjnego snmptrapd (`snmptrapd.conf`), zobacz przykład.

Note:
Może być konieczne ponowne uruchomienie snmptrapd, aby uwzględnił zmiany w konfiguracji.

Konfigurowanie odbiornika trapów Perl

Wymagania: Perl, Net-SNMP skompilowany z `--enable-embedded-perl` (domyślnie od Net-SNMP 5.4)

Odbiornik trapów Perl (szukaj `misc/snmptrap/zabbix\_trap\_receiver.pl`) może być używany do przekazywania trapów do serwera Zabbix bezpośrednio z snmptrapd. Aby go skonfigurować:

- Dodaj skrypt Perl do pliku konfiguracyjnego snmptrapd (`snmptrapd.conf`), np.:

```
perl do "[FULL PATH TO PERL RECEIVER SCRIPT]";
```

- skonfiguruj odbiornik, np.:

```
$SNMPTrapperFile = '[TRAP FILE]';
$DateTimeFormat = '[DATE TIME FORMAT]';
```

Note:
snmptrapd może wymagać ponownego uruchomienia, aby uwzględnić zmiany w konfiguracji.

Note:
Jeśli nazwa skryptu nie jest ujęta w cudzysłów, snmptrapd odmówi uruchomienia i wyświetli komunikaty podobne do poniższych:


```
Regexp modifiers "/l" and "/a" are mutually exclusive at (eval 2) line 1, at end of line
Regexp modifier "/l" may not appear twice at (eval 2) line 1, at end of line
```

Konfigurowanie SNMPTT

Na początku należy skonfigurować snmptrapd do używania SNMPTT.

Note:
Dla najlepszej wydajności SNMPTT powinien być skonfigurowany jako demon z użyciem `snmptthandler-embedded`, aby przekazywać do niego pułapki. Instrukcje znajdziesz w sekcji konfigurowania SNMPTT.

Gdy SNMPTT jest skonfigurowany do odbierania pułapek, skonfiguruj `snmptt.ini`:

1. Włącz użycie modułu Perl z pakietu `NET-SNMP`:

```
net_snmp_perl_enable = 1
```

2. Zapisuj pułapki do pliku pułapek, który będzie odczytywany przez Zabbix:

```
log_enable = 1
log_file = [TRAP FILE]
```

3. Ustaw format daty i czasu:

```
date_time_format = %Y-%m-%dT%H:%M:%S%z
```

Warning:
Pakiet `net-snmp-perl` został usunięty w RHEL 8.0-8.2; dodano go ponownie w RHEL 8.3. Więcej informacji znajdziesz w sekcji znane problemy.

Teraz sformatuj pułapki tak, aby Zabbix mógł je rozpoznać (edytuj `snmptt.conf`):

1. Każda instrukcja `FORMAT` powinna zaczynać się od `ZBXTRAP [address]`, gdzie `[address]` będzie porównywany z adresami IP i DNS interfejsów SNMP w Zabbix. Na przykład:

```
EVENT coldStart .1.3.6.1.6.3.1.1.5.1 "Status Events" Normal
FORMAT ZBXTRAP $aA Device reinitialized (coldStart)
```

2. Więcej informacji o formacie pułapek SNMP znajdziesz poniżej.

Attention:
Nie używaj nieznanych pułapek - Zabbix nie będzie w stanie ich rozpoznać. Nieznane pułapki można obsłużyć, definiując ogólne zdarzenie w `snmptt.conf`:


```
EVENT general .* "General event" Normal
```

Format trapu SNMP

Wszystkie niestandardowe odbiorniki trapów Perl oraz konfiguracja trapów SNMPTT muszą formatować trap w następujący sposób:

```
[timestamp] [the trap, part 1] ZBXTRAP [address] [the trap, part 2]
```

gdzie

- `[timestamp]` - znacznik czasu w formacie `%Y-%m-%dT%H:%M:%S%z`.
- `ZBXTRAP` - nagłówek wskazujący, że w tym wierszu zaczyna się nowy trap.
- `[address]` - adres IP używany do znalezienia hosta dla tego trapu.

Zwróć uwagę, że `ZBXTRAP` i `[address]` zostaną usunięte z komunikatu podczas przetwarzania. Jeśli trap ma inny format, Zabbix może zinterpretować trapy w nieoczekiwany sposób.

Przykładowy trap:

```
2024-01-11T15:28:47+0200 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events" localhost - ZBXTRAP 192.168.1.1 Link
```

Spowoduje to utworzenie następującego trapu dla interfejsu SNMP z IP=`192.168.1.1`:

```
2024-01-11T15:28:47+0200 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events"
localhost - Link down on interface 2. Admin state: 1. Operational state: 2
```

Wymagania systemowe

Note:
Zaleca się zainstalowanie plików MIB, aby zapewnić, że wartości pozycji będą wyświetlane w prawidłowym formacie. Bez plików MIB mogą wystąpić problemy z formatowaniem, takie jak wyświetlanie wartości w formacie HEX zamiast UTF-8 lub odwrotnie.

Obsługa dużych plików

Zabbix obsługuje duże pliki dla plików SNMP trapper. Maksymalny rozmiar pliku, jaki Zabbix może odczytać, wynosi 2^63 (8 EiB). Należy pamiętać, że system plików może narzucać niższy limit rozmiaru pliku.

Rotacja logów

Zabbix nie zapewnia własnego mechanizmu rotacji logów — powinien być on obsługiwany przez użytkownika. Rotacja logów powinna najpierw zmienić nazwę starego pliku, a dopiero później go usunąć, aby nie utracić żadnych trapów:

1. Zabbix otwiera plik trapów w ostatnio znanej lokalizacji i przechodzi do kroku 3.
2. Zabbix sprawdza, czy aktualnie otwarty plik został zrotowany, porównując numer inode z numerem inode zdefiniowanego pliku trapów. Jeśli nie ma otwartego pliku, Zabbix resetuje ostatnią lokalizację i przechodzi do kroku 1.

3. Zabbix odczytuje dane z aktualnie otwartego pliku i ustawia nową lokalizację.
4. Nowe dane są analizowane. Jeśli był to zrotowany plik, plik zostaje zamknięty i następuje powrót do kroku 2.
5. Jeśli nie było nowych danych, Zabbix usypia na 1 sekundę i wraca do kroku 2.

System plików

Ze względu na implementację pliku trap, Zabbix wymaga, aby system plików obsługiwał i-węzły, aby rozróżniać pliki (informacje są pobierane za pomocą wywołania `stat()`).

Przykłady konfiguracji z użyciem różnych wersji protokołu SNMP

Ten przykład wykorzystuje snmptrapd oraz skrypt odbierający Bash do przekazywania pułapek do serwer Zabbix.

Konfiguracja:

1. Skonfiguruj Zabbix tak, aby uruchamiał SNMP trapper i ustaw plik pułapek. Dodaj do `zabbix_server.conf`:

```
StartSNMPTrapper=1
SNMPTrapperFile=/var/lib/zabbix/snmptraps/snmptraps.log
```

2. Pobierz skrypt Bash do `/usr/sbin/zabbix_trap_handler.sh`:

```
curl -o /usr/sbin/zabbix_trap_handler.sh https://raw.githubusercontent.com/zabbix/zabbix-docker/7.4/templa
```

W razie potrzeby dostosuj zmienną `ZABBIX_TRAPS_FILE` w skrypcie. Aby użyć wartości domyślnej, najpierw utwórz katalog nadrzędny:

```
mkdir -p /var/lib/zabbix/snmptraps
```

3. Dodaj następującą konfigurację do `snmtrapd.conf` (zobacz działający przykład)

```
traphandle default /bin/bash /usr/sbin/zabbix_trap_handler.sh
```

Note:
Może być konieczne ponowne uruchomienie snmptrapd, aby uwzględnił zmiany w konfiguracji.

4. Utwórz pozycję SNMP TEST (pamiętaj o początkowych wymaganiach konfiguracyjnych):

   Type: *SNMP trap*
 Type of information: *Log*
 Host interface: *SNMP 127.0.0.1*
 Key: `snmptrap["linkUp"]`
 Log time format: `yyyyMMdd.hhmmss`

Zwróć uwagę, że używany jest format daty i czasu ISO 8601.

5. Następnie skonfigurujemy `snmptrapd` dla wybranej wersji protokołu SNMP i wyślemy testowe pułapki za pomocą narzędzia `snmptrap`.

SNMPv1, SNMPv2

Protokoły SNMPv1 i SNMPv2 opierają się na uwierzytelnianiu za pomocą "community string". W poniższym przykładzie użyjemy "secret" jako community string. Musi on mieć tę samą wartość na nadawcach pułapek SNMP.

Należy pamiętać, że chociaż SNMPv2 jest nadal szeroko używany w środowiskach produkcyjnych, nie zapewnia szyfrowania ani rzeczywistego uwierzytelniania nadawcy. Dane są przesyłane jako zwykły tekst, dlatego tych wersji protokołu należy używać wyłącznie w bezpiecznych środowiskach, takich jak sieć prywatna, i nigdy nie należy ich używać w żadnej publicznej ani zewnętrznej sieci.

Wersja SNMP 1 nie jest obecnie właściwie używana, ponieważ nie obsługuje liczników 64-bitowych i jest uznawana za protokół przestarzały.

Aby włączyć odbieranie pułapek SNMPv1 lub SNMPv2, należy dodać następującą linię do `snmptrapd.conf`. Zastąp `secret` ciągiem community SNMP skonfigurowanym na nadawcach pułapek SNMP:

```
authCommunity log,execute,net secret
```

Następnie możemy wysłać testową pułapkę za pomocą `snmptrap`. W tym przykładzie użyjemy popularnego OID "link up":

```
snmptrap -v 2c -c secret localhost '' linkUp.0
```

SNMPv3

SNMPv3 rozwiązuje problemy bezpieczeństwa SNMPv1/v2 i zapewnia uwierzytelnianie oraz szyfrowanie. Możesz użyć metod uwierzytelniania MD5 lub wielu SHA oraz DES/wielu AES jako szyfru.

Aby włączyć odbieranie SNMPv3, dodaj następujące linie do `snmptrapd.conf`:

```
createUser -e 0x8000000001020304 traptest SHA mypassword AES
authuser log,execute traptest
```

Attention:
Zwróć uwagę na słowo kluczowe "execute", które umożliwia wykonywanie skryptów dla tego modelu bezpieczeństwa użytkownika.

```
snmptrap -v 3 -n "" -a SHA -A mypassword -x AES -X mypassword -l authPriv -u traptest -e 0x800000000102030
```

Warning:
Jeśli chcesz używać silnych metod szyfrowania, takich jak AES192 lub AES256, użyj net-snmp w wersji 5.8 lub nowszej. Może być konieczna rekompilacja z opcją `configure --enable-blumenthal-aes`. Starsze wersje net-snmp nie obsługują AES192/AES256. Zobacz także: Strong Authentication or Encryption.

Weryfikacja

W obu przykładach zobaczysz podobne wiersze w `/var/lib/zabbix/snmptraps/snmptraps.log`:

```
2024-01-30T10:04:23+0200 ZBXTRAP 127.0.0.1
UDP: [127.0.0.1]:56585->[127.0.0.1]:162
DISMAN-EVENT-MIB::sysUpTimeInstance = 2538834
SNMPv2-MIB::snmpTrapOID.0 = IF-MIB::linkUp.0
```

Wartość pozycji w Zabbix będzie następująca:

```
2024-01-30 10:04:23 2024-01-30 10:04:21

2024-01-30T10:04:21+0200 UDP: [127.0.0.1]:56585->[127.0.0.1]:162
DISMAN-EVENT-MIB::sysUpTimeInstance = 2538834
SNMPv2-MIB::snmpTrapOID.0 = IF-MIB::linkUp.0
```

Przykład z Perlem:

```
2024-01-30T11:42:54+0200 ZBXTRAP 127.0.0.1
PDU INFO:
  receivedfrom                   UDP: [127.0.0.1]:58649->[127.0.0.1]:162
  notificationtype               TRAP
  version                        1
  community                      public
  errorstatus                    0
  transactionid                  1
  requestid                      2101882550
  messageid                      0
  errorindex                     0
VARBINDS:
  DISMAN-EVENT-MIB::sysUpTimeInstance type=67 value=Timeticks: (457671) 1:16:16.71
  SNMPv2-MIB::snmpTrapOID.0      type=6  value=OID: IF-MIB::linkUp.0
```

Zobacz także

- Artykuł na blogu Zabbix o pułapkach SNMP
- Konfigurowanie snmptrapd (oficjalna dokumentacja net-snmp)
- Konfigurowanie snmptrapd do odbierania powiadomień SNMPv3 (oficjalna dokumentacja net-snmp)

5 Wewnętrzne Zabbix

Przegląd

Kontrole wewnętrzne umożliwiają monitorowanie wewnętrznych procesów Zabbix. Innymi słowy, można monitorować to, co dzieje się z serwer Zabbix lub Zabbix proxy.

Kontrole wewnętrzne są obliczane:

- na serwer Zabbix — jeśli host jest monitorowany przez serwer
- na Zabbix proxy — jeśli host jest monitorowany przez proxy

Kontrole wewnętrzne są przetwarzane przez serwer lub proxy niezależnie od statusu konserwacji hosta.

Aby użyć tej pozycji, wybierz typ pozycji **Zabbix internal**.

> **Note:**
> Kontrole wewnętrzne są przetwarzane przez pollery Zabbix.

Wydajność

Używanie niektórych wewnętrznych pozycji może negatywnie wpływać na wydajność. Są to następujące pozycje:

- `zabbix[host,,items]`
- `zabbix[host,,items_unsupported]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[queue,,]`
- `zabbix[requiredperformance]`
- `zabbix[stats,,,queue,,]`
- `zabbix[triggers]`

Dotyczy to również sekcji frontend Informacje o systemie oraz Kolejka.

Obsługiwane kontrole

Klucze pozycji są wymienione bez parametrów konfigurowalnych i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Item key	Opis
zabbix[boottime]	Czas uruchomienia procesu serwera Zabbix lub Zabbix proxy w sekundach.
zabbix[cluster,discovery,nodes]	Wykrywa węzły klastra wysokiej dostępności.
zabbix[connector_queue]	Liczba wartości umieszczonych w kolejce connector.
zabbix[discovery_queue]	Liczba kontroli sieciowych umieszczonych w kolejce wykrywania.
zabbix[host,,items]	Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych) na hoście.
zabbix[host,,items_unsupported]	Liczba włączonych nieobsługiwanych pozycji na hoście.
zabbix[host,maintenance]	Bieżący stan konserwacji hosta.
zabbix[host,active_agent,available]	Dostępność aktywnych kontroli agent na hoście.
zabbix[host,discovery,interfaces]	Szczegóły wszystkich skonfigurowanych interfejsów hosta w frontendzie Zabbix.
zabbix[host,,available]	Dostępność głównego interfejsu określonego typu kontroli na hoście.
zabbix[hosts]	Liczba monitorowanych hostów.
zabbix[items]	Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych).
zabbix[items_unsupported]	Liczba nieobsługiwanych pozycji.
zabbix[java,,]	Informacje o Zabbix Java gateway.
zabbix[lld_queue]	Liczba wartości umieszczonych w kolejce przetwarzania wykrywania niskiego poziomu.
zabbix[preprocessing]	Statystyki wartości odebranych przez menedżera przetwarzania wstępnego.
zabbix[preprocessing_queue]	Liczba wartości umieszczonych w kolejce przetwarzania wstępnego.
zabbix[process,,,]	Procent czasu, jaki określony proces Zabbix lub grupa procesów (identyfikowana przez <type> i <mode>) spędziła w stanie <state>.
zabbix[proxy,,]	Informacje o Zabbix proxy.
zabbix[proxy,discovery]	Lista Zabbix proxy.
zabbix[proxy group,,available]	Liczba proxy online w grupie proxy.
zabbix[proxy group,,pavailable]	Procent proxy online w grupie proxy.
zabbix[proxy group,,proxies]	Lista Zabbix proxy w grupie proxy.
zabbix[proxy group,,state]	Stan grupy proxy.
zabbix[proxy group,discovery]	Zwraca listę grup proxy z danymi konfiguracyjnymi i danymi czasu rzeczywistego.
zabbix[proxy_buffer,buffer,]	Zwraca statystyki użycia bufora pamięci proxy.
zabbix[proxy_buffer,state,changes]	Zwraca liczbę zmian stanu między trybami bufora dysk/pamięć od uruchomienia.
zabbix[proxy_buffer,state,current]	Zwraca bieżący stan pracy, w którym przechowywane są nowe dane.
zabbix[proxy_history]	Liczba wartości w tabeli historii proxy oczekujących na wysłanie do serwera.

Item key	Opis
zabbix[queue„]	Liczba monitorowanych pozycji w kolejce, które są opóźnione co najmniej o <from> sekund, ale mniej niż o <to> sekund.
zabbix[rcache„]	Statystyki dostępności pamięci podręcznej konfiguracji Zabbix.
zabbix[requiredperformance]	Wymagana wydajność serwera Zabbix lub Zabbix proxy, wyrażona jako oczekiwana liczba nowych wartości na sekundę.
zabbix[stats„]	Zwraca wewnętrzne metryki serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
zabbix[stats„,queue„]	Zwraca wewnętrzne metryki kolejki serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
zabbix[tcache„]	Statystyki efektywności pamięci podręcznej funkcji trendów Zabbix.
zabbix[triggers]	Liczba włączonych wyzwalaczy w bazie danych Zabbix, ze wszystkimi pozycjami włączonymi na włączonych hostach.
zabbix[uptime]	Czas działania procesu serwera Zabbix lub proxy w sekundach.
zabbix[vcache,buffer,]	Statystyki dostępności pamięci podręcznej wartości Zabbix.
zabbix[vcache,cache,]	Statystyki efektywności pamięci podręcznej wartości Zabbix.
zabbix[version]	Wersja serwera Zabbix lub proxy.
zabbix[vmware,buffer,]	Statystyki dostępności pamięci podręcznej vmware Zabbix.
zabbix[vps,written]	Łączna liczba wartości historii zapisanych do bazy danych.
zabbix[wcache„]	Statystyki i dostępność pamięci podręcznej zapisu Zabbix.

Szczegóły klucza pozycji

- Parametry bez nawiasów ostrych są obowiązkowe i muszą być użyte *dokładnie w tej postaci* (na przykład „host” i „available” w `zabbix[host,<type>,available]`).
- Parametry ujęte w nawiasy ostre **< >** muszą zostać zastąpione prawidłową wartością. Jeśli parametr ma wartość domyślną, można go pominąć.
- Wartości dla pozycji i parametrów pozycji oznaczonych jako „not supported on proxy” mogą być pobierane tylko wtedy, gdy host jest monitorowany przez serwer. Z kolei wartości oznaczone jako „not supported on server” mogą być pobierane tylko wtedy, gdy host jest monitorowany przez proxy.

zabbix[boottime]

 Czas uruchomienia procesu serwera Zabbix lub procesu Zabbix proxy w sekundach.
 Zwracana wartość: *Integer*.

zabbix[cluster,discovery,nodes]

 Wykrywa węzły klastra wysokiej dostępności.
 Zwracana wartość: *obiekt JSON*.

Uwagi:

- Ta pozycja może być używana w wykrywaniu niskiego poziomu.
- Zwracane pola w większości odpowiadają właściwościom obiektu węzła wysokiej dostępności.
- Dodatkowe pola:
  - `db_timestamp` — bieżący czas serwera (znacznik czasu Unix) uwzględniony w odpowiedzi
  - `lastaccess_age` — liczba sekund od ostatniego heartbeat węzła (db_timestamp - lastaccess)
  - `status` — stan węzła:
    - 0 — zapasowy
    - 1 — zatrzymany ręcznie
    - 2 — niedostępny
    - 3 — aktywny

Przykład zwracanych wartości:

```
[
  {
    "id": "ckvupihk70001z8mkpw5cg0u3",
    "name": "zabbix-prod-01",
    "status": 3,
    "address": "10.0.4.12:10051",
    "port": 10051,
    "lastaccess": 1756115995,
    "db_timestamp": 1756116000,
    "lastaccess_age": 5
  },
  {
```

```
    "id": "ckvx2a9k70004b1nq2hz9d7f",
    "name": "zabbix-standby-02",
    "status": 0,
    "address": "10.0.4.13:10051",
    "port": 10051,
    "lastaccess": 1756115550,
    "db_timestamp": 1756116000,
    "lastaccess_age": 450
  },
  {
    "id": "ckw0bq3l70007y4r1a0m5kz8",
    "name": "zabbix-backup-eu1",
    "status": 1,
    "address": "backup.example.com:10051",
    "port": 10051,
    "lastaccess": 1756105080,
    "db_timestamp": 1756116000,
    "lastaccess_age": 10920
  }
]
```


zabbix[connector_queue]

 Liczba wartości umieszczonych w kolejce connector queue.
 Zwracana wartość: *Integer*.

zabbix[discovery_queue]

 Liczba kontroli sieciowych umieszczonych w kolejce wykrywania.
 Wartość zwracana: *Integer*.

zabbix[host,„items]

 Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych) na hoście.
 Zwracana wartość: *Integer*.

zabbix[host,„items_unsupported]

 Liczba włączonych nieobsługiwanych pozycji na hoście.
 Zwracana wartość: *Integer*.

zabbix[host,„maintenance]

 Bieżący status konserwacji hosta.
 Zwracane wartości: *0* - stan normalny; *1* - konserwacja z gromadzeniem danych; *2* - konserwacja bez gromadzenia danych.

Uwagi:

- Ta pozycja jest zawsze przetwarzana przez serwer Zabbix niezależnie od lokalizacji hosta (na serwerze lub proxy). Proxy nie otrzyma tej pozycji wraz z danymi konfiguracyjnymi.
- Drugi parametr musi być pusty i jest zarezerwowany do wykorzystania w przyszłości.

zabbix[host,active_agent,available]

 Dostępność aktywnych kontroli agenta na hoście.
 Zwracane wartości: *0* - nieznana; *1* - dostępny; *2* - niedostępny.

zabbix[host,discovery,interfaces]

 Szczegóły wszystkich skonfigurowanych interfejsów hosta w Zabbix frontend.
 Zwracana wartość: *obiekt JSON*.

Uwagi:

- Ta pozycja może być używana w odkrywaniu niskiego poziomu.
- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[host,<type>,available]

 Dostępność głównego interfejsu określonego typu kontroli na hoście.
 Zwracane wartości: *0* - niedostępny; *1* - dostępny; *2* - nieznany.

Parametry:

- **type** - *agent*, *snmp*, *ipmi* lub *jmx*.

Komentarze:

- Wartość pozycji jest obliczana zgodnie z parametrami konfiguracyjnymi dotyczącymi nieosiągalności/niedostępności hosta.

zabbix[hosts]

 Liczba monitorowanych hostów.
 Zwracana wartość: *Integer*.

zabbix[items]

 Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych).
 Zwracana wartość: *Integer*.

zabbix[items_unsupported]

 Liczba nieobsługiwanych pozycji.
 Zwracana wartość: *Integer*.

zabbix[java„<param>]

 Informacje o Zabbix Java gateway.
 Zwracane wartości: *1* - jeśli <param> to *ping*; *wersja Java gateway* - jeśli <param> to *version* (na przykład: ”7.4.0”).

Parametry:

- **param** - *ping* lub *version*.

Komentarze:

- Ta pozycja może być używana do sprawdzania dostępności Java gateway za pomocą funkcji wyzwalacza `nodata()`.
- Drugi parametr musi być pusty i jest zarezerwowany do przyszłego użycia.

zabbix[lld_queue]

 Liczba wartości umieszczonych w kolejce przetwarzania wykrywania niskiego poziomu.
 Zwracana wartość: *Integer*.

Uwagi:

- Ta pozycja może być używana do monitorowania długości kolejki przetwarzania wykrywania niskiego poziomu.

zabbixpreprocessing

 Statystyki wartości otrzymanych przez menedżera preprocessing:

- *queued* — liczba i rozmiar wartości oczekujących w kolejce, które wymagają preprocessing (licznik)
- *direct* — liczba i rozmiar wartości oczekujących w kolejce, które nie wymagają preprocessing (licznik)
- *queue* — liczba wartości umieszczonych w kolejce preprocessing (to samo co `zabbix[preprocessing_queue]`)

Zwracana wartość: *JSON*.

Przykład zwracanych wartości:

```
{"data":
    {
    "queued": {
        "count": 106,
        "size": 58620
    },
    "direct": {
        "count": 395,
        "size": 33843
    },
    "queue": 0
    }
}
```

zabbix[preprocessing_queue]

 Liczba wartości umieszczonych w kolejce przetwarzania wstępnego.
 Zwracana wartość: *Integer*.

Uwagi:

- Ta pozycja może być używana do monitorowania długości kolejki przetwarzania wstępnego.

zabbix[process,<type>,<mode>,<state>]

 Procent czasu, jaki określony proces Zabbix lub grupa procesów (identyfikowana przez <type> i <mode>) spędziły w stanie <state>. Jest obliczany tylko dla ostatniej minuty.
 Zwracana wartość: *Float*.

Parametry:

- **type** - dla procesów serwera: *agent poller*, *alert manager*, *alert syncer*, *alerter*, *availability manager*, *browser poller*, *configuration syncer*, *configuration syncer worker*, *connector manager*, *connector worker*, *discovery manager*, *discovery worker*, *escalator*, *ha manager*, *history poller*, *history syncer*, *housekeeper*, *http agent poller*, *http poller*, *icmp pinger*, *internal poller ipmi manager*, *ipmi poller*, *java poller*, *lld manager*, *lld worker*, *odbc poller*, *poller*, *preprocessing manager*, *preprocessing worker*, *proxy group manager*, *proxy poller*, *self-monitoring*, *service manager*, *snmp poller*, *snmp trapper*, *task manager*, *timer*, *trapper*, *trigger housekeeper*, *unreachable poller*, *vmware collector*;
dla procesów proxy: *agent poller*, *availability manager*, *browser poller*, *configuration syncer*, *data sender*, *discovery manager*, *discovery worker*, *history syncer*, *housekeeper*, *http agent poller*, *http poller*, *icmp pinger*, *internal poller ipmi manager*, *ipmi poller*, *java poller*, *odbc poller*, *poller*, *preprocessing manager*, *preprocessing worker*, *self-monitoring*, *snmp poller*, *snmp trapper*, *task manager*, *trapper*, *unreachable poller*, *vmware collector*;
- **mode** - *avg* - wartość średnia dla wszystkich procesów danego typu (domyślnie);
count - zwraca liczbę forków dla danego typu procesu, nie należy podawać <state>;
max - wartość maksymalna;
min - wartość minimalna;
<*process number*> - numer procesu (od 1 do liczby wstępnie utworzonych instancji; na przykład, jeśli działają 4 trappers, wartość mieści się w zakresie od 1 do 4);
- **state** - *busy* - proces jest w stanie zajętości, na przykład przetwarza żądanie (domyślnie);
idle - proces jest w stanie bezczynności i nic nie robi.

Uwagi:

- Jeśli <mode> jest numerem procesu Zabbix, który nie jest uruchomiony (na przykład przy 5 działających pollerach w <mode> podano 6), taka pozycja stanie się nieobsługiwana.
- Wartości minimalna i maksymalna odnoszą się do procentu wykorzystania pojedynczego procesu. Tak więc, jeśli w grupie 3 pollerów procenty wykorzystania dla każdego procesu wynosiły 2, 18 i 66, to min zwróci 2, a max zwróci 66.
- Procesy raportują, co robią, do pamięci współdzielonej, a proces self-monitoring podsumowuje te dane co sekundę. Zmiany stanu (busy/idle) są rejestrowane przy zmianie — dlatego proces, który staje się busy, zostaje tak zarejestrowany i nie zmienia ani nie aktualizuje stanu, dopóki nie stanie się idle. Zapewnia to poprawne zarejestrowanie nawet całkowicie zawieszonych procesów jako 100% busy.
- Obecnie „busy" oznacza „nie śpi", ale w przyszłości mogą zostać wprowadzone dodatkowe stany — oczekiwanie na blokady, wykonywanie zapytań do bazy danych itp. Należy pamiętać, że asynchroniczne pollery są uznawane za busy, jeśli osiągnęły limit ustawiony przez parametr konfiguracyjny MaxConcurrentChecksPerPoller server/proxy.
- W systemie Linux i większości innych systemów rozdzielczość wynosi 1/100 sekundy.

Przykłady:

```
zabbix[process,poller,avg,busy] #średni czas, jaki procesy poller spędziły na wykonywaniu czegoś w ciągu c
zabbix[process,"icmp pinger",max,busy] #maksymalny czas spędzony na wykonywaniu czegoś przez dowolny proce
zabbix[process,"history syncer",2,busy] #czas spędzony na wykonywaniu czegoś przez history syncer numer 2
zabbix[process,trapper,count] #liczba aktualnie uruchomionych procesów trapper
```

zabbix[proxy,<name>,<param>]

 Informacje o proxy Zabbix.
 Wartość zwracana: *Integer*.

Parametry:

- **name** - nazwa proxy;
- **param** - *lastaccess* - znacznik czasu ostatniej wiadomości heartbeat odebranej z proxy;
delay - jak długo zebrane wartości pozostają niewysłane; obliczane jako "opóźnienie proxy" + ("bieżący czas serwera" - "ostatni lastaccess proxy"), gdzie "opóźnienie proxy" to różnica między bieżącym czasem proxy a znacznikiem czasu najstarszej niewysłanej wartości na proxy.

Komentarze:

- Ta pozycja jest zawsze przetwarzana przez serwer Zabbix niezależnie od lokalizacji hosta (na serwerze lub proxy).
- Funkcji `fuzzytime()` można użyć do sprawdzenia dostępności proxy.

Przykład:

```
zabbix[proxy,"Germany",lastaccess] #znacznik czasu ostatniej wiadomości heartbeat odebranej z proxy "Germa
```

zabbix[proxy,discovery]

 Lista proxy Zabbix zawierająca nazwę, tryb, szyfrowanie, kompresję, wersję, czas ostatniego połączenia, liczbę hostów, liczbę pozycji, wymaganą liczbę wartości na sekundę (vps), status wersji (aktualna/przestarzała/niewspierana), limity czasu według typu pozycji, nazwę grupy proxy (jeśli proxy należy do grupy), stan (nieznany/offline/online).
 Wartość zwracana: *obiekt JSON*.

zabbix[proxy group,<name>,available]

 Liczba proxy online w grupie proxy.
 Wartość zwracana: *Integer*.

Parametry:

- **name** - nazwa grupy proxy.

zabbix[proxy group,<name>,pavailable]

 Procent proxy online w grupie proxy.
 Zwracana wartość: *Float*.

Parametry:

- **name** — nazwa grupy proxy.

zabbix[proxy group,<name>,proxies]

 Lista proxy Zabbix w grupie proxy z nazwą, trybem, szyfrowaniem, kompresją, wersją, czasem ostatniej aktywności, liczbą hostów, liczbą pozycji, wymaganą liczbą wartości na sekundę (vps), statusem wersji (current/outdated/unsupported), limitami czasu, nazwą grupy proxy, stanem (unknown/offline/online).
 Zwracana wartość: *JSON*.

Parametry:

- **name** - nazwa grupy proxy.

zabbix[proxy group,<name>,state]

 Stan grupy proxy.
 Zwracana wartość: *0* - nieznany; *1* - offline; *2* - odzyskiwanie; *3* - online; *4* - pogarszanie się.

Parametry:

- **name** - nazwa grupy proxy.

zabbix[proxy group,discovery]

 Zwraca listę grup proxy z danymi konfiguracyjnymi i danymi czasu rzeczywistego. Dane konfiguracyjne obejmują nazwę grupy proxy, opóźnienie przełączenia awaryjnego oraz minimalną wymaganą liczbę proxy online. Dane czasu rzeczywistego obejmują stan grupy proxy (szczegóły w komentarzach), liczbę proxy online oraz procent proxy online.
 Wartość zwracana: *JSON*.

Komentarze:

- Ta pozycja nie zwraca proxy nienależących do żadnej grupy.
- Jeśli dla ”failover_delay” lub ”min_online” występuje nieprawidłowa wartość, zgłaszana jest specjalna wartość *-1*, aby to zasygnalizować. Nieprawidłowe wartości mogą wystąpić, jeśli do konfiguracji używane są makra i nie można ich rozwinąć do prawidłowej wartości.
- Stan grupy proxy jest raportowany jako liczba całkowita: *0* - nieznany; *1* - offline; *2* - odzyskiwanie; *3* - online; *4* - pogarszanie.

Przykład wartości zwracanych:

```
{
    "groups": [
       { "name": "Riga", "failover_delay": 60, "min_online": 1 },
       { "name": "Tokyo", "failover_delay": 60, "min_online": 2 },
       { "name": "Porto Alegre", "failover_delay": 60, "min_online": 3 }
    ],
    "details": {
        "Riga": { "state": 3, "available": 10, "pavailable": 20 },
        "Tokyo": { "state": 3, "available": 10, "pavailable": 20 },
        "Porto Alegre": { "state": 1, "available": 0, "pavailable": 0 }
    }
}
```

zabbix[proxy_buffer,buffer,<mode>]

 Statystyki użycia bufora pamięci proxy.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **mode**: *total* - całkowity rozmiar bufora (może być używany do sprawdzenia, czy bufor pamięci jest włączony);
free - rozmiar wolnego bufora;
pfree - procent wolnego bufora;
used - rozmiar używanego bufora;
pused - procent używanego bufora.

Uwagi:

- Zwraca błąd „Proxy memory buffer is disabled”, gdy bufor pamięci jest wyłączony;
- Ta pozycja nie jest obsługiwana na serwerze Zabbix.

zabbix[proxy_buffer,state,changes]

 Zwraca liczbę zmian stanu między trybami bufora dysk/pamięć od momentu uruchomienia.
 Zwracane wartości: *Integer*; *0* - bufor pamięci jest wyłączony.

Uwagi:

- Częste zmiany stanu wskazują, że należy zwiększyć rozmiar lub wiek bufora pamięci.
- Jeśli stan bufora pamięci jest monitorowany rzadko (na przykład raz na minutę), bufor może zmienić swój stan bez zarejestrowania tego faktu.

zabbix[proxy_buffer,state,current]

 Zwraca bieżący stan roboczy, w którym przechowywane są nowe dane.
 Zwracane wartości: *0* - dysk; *1* - pamięć.

Uwagi:

- „0” jest również zwracane, gdy bufor pamięci jest wyłączony.

zabbix[proxy_history]

 Liczba wartości w tabeli historii proxy oczekujących na wysłanie do serwer.
 Zwracane wartości: *Integer*.

Komentarze:

- Ta pozycja nie jest obsługiwana na serwerze Zabbix.

zabbix[queue,<from>,<to>]

 Liczba monitorowanych pozycji w kolejce, które są opóźnione co najmniej o <from> sekund, ale mniej niż o <to> sekund.
 Zwracana wartość: *Integer*.

Parametry:

- **from** — opóźnienie co najmniej o (domyślnie 6 sekund);
- **to** — opóźnienie co najwyżej o (domyślnie nieskończoność).

Uwagi:

- Sufiksy czasu (s,m,h,d,w) są obsługiwane w parametrach.

> **Attention:**
> Proxy Zabbix nie uwzględnia okresów konserwacji; szczegóły znajdują się w sekcji Obliczanie kolejek podczas konserwacji.

zabbix[rcache,<cache>,<mode>]

 Statystyki dostępności pamięci podręcznej konfiguracji Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **cache** - *buffer*;
- **mode** - *total* - całkowity rozmiar bufora;
free - rozmiar wolnego bufora;
pfree - procent wolnego bufora;
used - rozmiar używanego bufora;
pused - procent używanego bufora.

zabbix[requiredperformance]

 Wymagana wydajność serwera Zabbix lub proxy Zabbix, wyrażona jako oczekiwana liczba nowych wartości na sekundę.
 Zwracana wartość: *Float*.

Komentarze:

- W przybliżeniu odpowiada „Required server performance, new values per second” w *Raporty* > *Informacje o systemie*.

zabbix[stats,<ip>,<port>]

 Zwraca wewnętrzne metryki serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
 Zwracane wartości: *obiekt JSON*.

Parametry:

- **ip** — lista IP/DNS/masek sieciowych serwerów/proxy, które mają być odpytywane zdalnie (domyślnie 127.0.0.1);
- **port** — port serwera/proxy, który ma być odpytywany zdalnie (domyślnie 10051).

Uwagi:

- Żądanie statystyk będzie akceptowane tylko z adresów wymienionych w parametrze ’StatsAllowedIP’ server/proxy na instancji docelowej.
- Ta pozycja zwraca wybrany zestaw wewnętrznych metryk. Szczegóły znajdują się w sekcji Zdalne monitorowanie statystyk Zabbix.

zabbix[stats,<ip>,<port>,queue,<from>,<to>]

 Zwraca metryki wewnętrznej kolejki (zob. `zabbix[queue,<from>,<to>]`) serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
 Zwracane wartości: *obiekt JSON*.

Parametry:

- **ip** - lista IP/DNS/masek sieciowych serwerów/proxy, które mają być odpytywane zdalnie (domyślnie 127.0.0.1);
- **port** - port serwera/proxy, który ma być odpytywany zdalnie (domyślnie 10051);
- **from** - opóźnione co najmniej o (domyślnie 6 sekund);
- **to** - opóźnione co najwyżej o (domyślnie nieskończoność).

Komentarze:

- Żądanie statystyk będzie akceptowane wyłącznie z adresów wymienionych w parametrze 'StatsAllowedIP' server/proxy na instancji docelowej.
- Ten item zwraca wybrany zestaw metryk wewnętrznych. Szczegóły znajdują się w sekcji Zdalne monitorowanie statystyk Zabbix.

Attention:
Proxy Zabbix nie uwzględnia okresów konserwacji; szczegóły znajdują się w sekcji Obliczanie kolejek podczas konserwacji.

zabbix[tcache,cache,<parameter>]

 Statystyki efektywności pamięci podręcznej funkcji trendów Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **parameter** - *all* - łączna liczba żądań do pamięci podręcznej (domyślnie);
hits - trafienia pamięci podręcznej;
phits - procent trafień pamięci podręcznej;
misses - chybienia pamięci podręcznej;
pmisses - procent chybień pamięci podręcznej;
items - liczba elementów w pamięci podręcznej;
requests - liczba żądań w pamięci podręcznej;
pitems - procent elementów w pamięci podręcznej względem sumy elementów i żądań w pamięci podręcznej. Niski procent najprawdopodobniej oznacza, że rozmiar pamięci podręcznej można zmniejszyć.

Uwagi:

- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[triggers]

 Liczba włączonych wyzwalaczy w bazie danych Zabbix, ze wszystkimi pozycjami włączonymi na włączonych hostach.
 Zwracana wartość: *Integer*.

Komentarze:

- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[uptime]

 Czas działania procesu serwera lub proxy Zabbix w sekundach.
 Zwracana wartość: *Integer*.

zabbix[vcache,buffer,<mode>]

 Statystyki dostępności pamięci podręcznej wartości Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **mode** - *total* - całkowity rozmiar bufora;
free - rozmiar wolnego bufora;
pfree - procent wolnego bufora;
used - rozmiar używanego bufora;
pused - procent używanego bufora.

Uwagi:

- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[vcache,cache,<parameter>]

 Statystyki efektywności pamięci podręcznej wartości Zabbix.
 Zwracana wartość: *Integer*. Jeśli <parameter> jest ustawiony na *mode*, zwracana wartość to: *0* - tryb normalny; *1* - tryb niskiej pamięci.

Parametry:

- **parameter** - *requests* - całkowita liczba żądań;
hits - liczba trafień pamięci podręcznej (wartości historii pobrane z pamięci podręcznej);
misses - liczba chybień pamięci podręcznej (wartości historii pobrane z bazy danych);
mode - tryb działania pamięci podręcznej wartości.

Uwagi:

- Po włączeniu trybu niskiej pamięci pamięć podręczna wartości pozostanie w tym stanie przez 24 godziny, nawet jeśli problem, który wyzwolił ten tryb, zostanie rozwiązany wcześniej.
- Możesz użyć tego klucza z krokiem przetwarzania wstępnego *Change per second*, aby uzyskać statystyki wartości na sekundę.
- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbixversion

 Wersja serwera Zabbix lub proxy.
 Zwracana wartość: *String*. Na przykład: `7.4.0`.

zabbix[vmware,buffer,<mode>]

 Statystyki dostępności pamięci podręcznej vmware Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **mode** - *total* - całkowity rozmiar bufora;
free - rozmiar wolnego bufora;
pfree - procent wolnego bufora;
used - rozmiar używanego bufora;
pused - procent używanego bufora.

zabbix[vps,written]

 Łączna liczba wartości historii zapisanych do bazy danych.
 Zwracana wartość: *Integer*.

zabbix[wcache,<cache>,<mode>]

 Statystyki i dostępność cache zapisu Zabbix.
 Wartości zwracane: *Integer* (dla liczby/rozmiaru); *Float* (dla procentu).

Parametry:

- **cache** - *values*, *history*, *index* lub *trend*;
- **mode** - (dla *values*) *all* (domyślnie) - łączna liczba wartości przetworzonych przez serwer/proxy Zabbix, z wyjątkiem nieobsługiwanych pozycji (licznik);
float - liczba przetworzonych wartości typu float (licznik);
uint - liczba przetworzonych niepodpisanych liczb całkowitych (licznik);
str - liczba przetworzonych wartości znakowych/łańcuchowych (licznik);
log - liczba przetworzonych wartości logów (licznik);
text - liczba przetworzonych wartości tekstowych (licznik);
bin - liczba przetworzonych wartości binarnych (licznik);
not supported - liczba przypadków, w których przetwarzanie pozycji zakończyło się oznaczeniem jej jako nieobsługiwana lub utrzymaniem tego stanu (licznik);
(dla cache *history*, *index*, *trend*) *pfree* (domyślnie) - procent wolnego bufora;
total - całkowity rozmiar bufora;
free - rozmiar wolnego bufora;
used - rozmiar użytego bufora;
pused - procent użytego bufora.

Komentarze:

- Parametr cache `trend` nie jest obsługiwany przez proxy Zabbix.
- Cache history jest używany do przechowywania wartości pozycji. Niska wartość wskazuje na problemy wydajnościowe po stronie bazy danych.
- Cache history index jest używany do indeksowania wartości przechowywanych w cache history.
- Po zapełnieniu cache history i jego wyczyszczeniu cache history index nadal będzie przechowywać część danych. To zachowanie jest oczekiwane i pomaga systemowi działać wydajniej, ponieważ unika dodatkowego przetwarzania wymaganego do ciągłego zmieniania rozmiaru pamięci.
- Cache trend przechowuje agregat dla bieżącej godziny dla wszystkich pozycji, które otrzymują dane.
- Możesz użyć klucza zabbix[wcache,values] wraz z krokiem przetwarzania *Change per second*, aby uzyskać statystyki wartości na sekundę.

6 Zabbix trapper

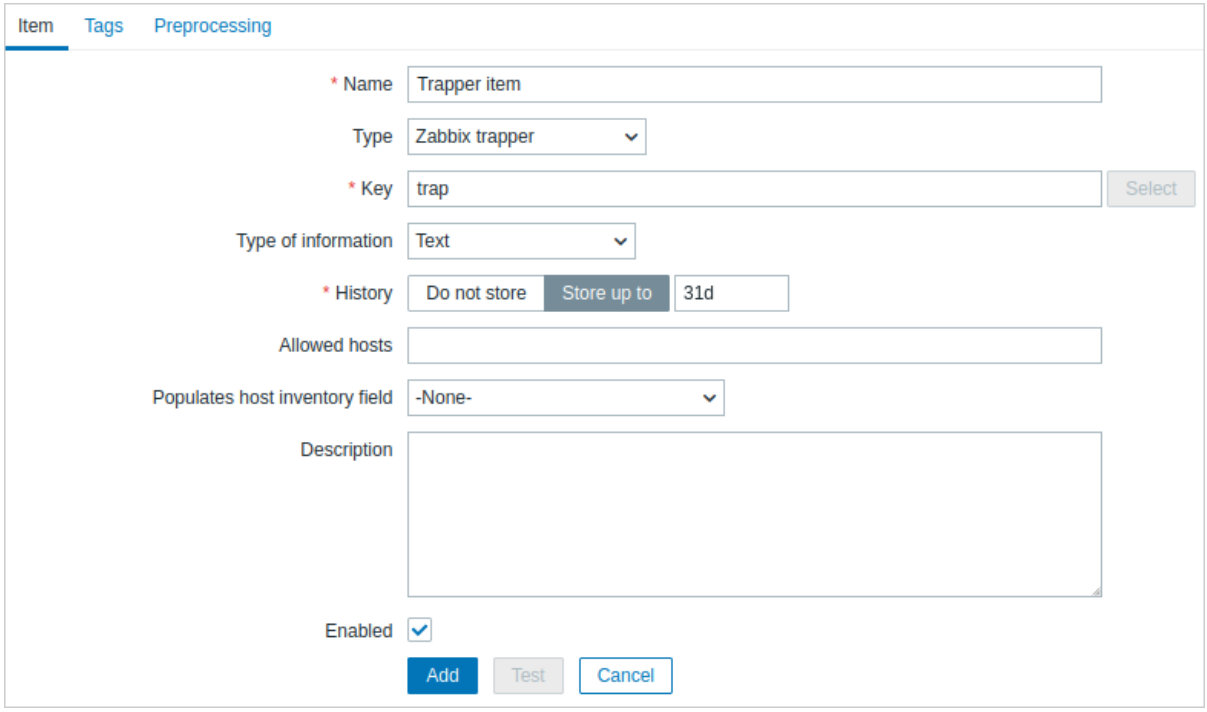
Przegląd

Pozycje trapper przyjmują dane przychodzące zamiast je odpytywać. Jest to przydatne dla wszelkich danych, które chcesz wysyłać do Zabbix.

Konfiguracja

Aby skonfigurować pozycję typu trapper:

1. Przejdź do *Data collection* → *Hosts*.
2. Kliknij *Items* w wierszu hosta.
3. Kliknij *Create item*.
4. Wprowadź parametry pozycji w formularzu konfiguracji.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji typu trapper to:

Type	Wybierz ”Zabbix trapper”.
Key	Wprowadź klucz, który będzie używany do rozpoznawania pozycji podczas wysyłania danych do serwer Zabbix.
Type of information	Wybierz typ informacji, który będzie odpowiadał formatowi wysyłanych danych.
Allowed hosts	Lista adresów IP rozdzielonych przecinkami (opcjonalnie w notacji CIDR) lub nazw DNS. Jeśli zostanie określona, przychodzące połączenia będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, to ’127.0.0.1’, ’::127.0.0.1’, ’::ffff:127.0.0.1’ są traktowane jednakowo, a ’::/0’ zezwoli na dowolny adres IPv4 lub IPv6. ’0.0.0.0/0’ może być użyte do zezwolenia na dowolny adres IPv4. Należy pamiętać, że ”IPv4-compatible IPv6 addresses” (prefiks 0000::/96) są obsługiwane, ale uznane za przestarzałe przez RFC4291. Przykład: 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} Obsługiwane są spacje, makra użytkownika oraz makra hosta {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}.

Note:
Przed wysłaniem wartości może być konieczne odczekanie do 60 sekund po zapisaniu pozycji, aż serwer Zabbix pobierze zmiany z aktualizacji pamięci podręcznej konfiguracji.

Wysyłanie danych

Wysyłanie danych do serwera lub proxy Zabbix jest możliwe przy użyciu narzędzia Zabbix sender lub protokołu Zabbix sender. Wysyłanie danych do serwera Zabbix jest również możliwe przy użyciu metody API `history.push`.

Zabbix sender

Aby wysłać dane do serwera lub proxy Zabbix przy użyciu narzędzia Zabbix sender, możesz uruchomić następujące polecenie, aby wysłać „wartość testową”:

```
zabbix_sender -z <server IP address> -p 10051 -s "New host" -k trap -o "test value"
```

Aby wysłać „wartość testową", używane są następujące opcje polecenia:

- `-z` — określa adres IP serwera Zabbix
- `-p` — określa numer portu serwera Zabbix (domyślnie 10051)
- `-s` — określa hosta (upewnij się, że używasz technicznej, a nie widocznej nazwy hosta)
- `-k` — określa klucz pozycji skonfigurowanej w pozycji trapper
- `-o` — określa wartość do wysłania

Attention:
Proces Zabbix trapper nie rozwija makr użytych w kluczu pozycji podczas sprawdzania istnienia odpowiedniego klucza pozycji dla docelowego hosta.

Więcej informacji o komunikacji między Zabbix sender a serwerem lub proxy Zabbix można znaleźć w sekcji *protokół Zabbix sender*.

history.push

Aby wysłać dane do serwera Zabbix za pomocą metody API `history.push`, można wykonać następujące żądanie HTTP POST zawierające przykładowe wartości testowe:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"history.push","params":[{"itemid":10600,"value":"test value 1"},{"ite
```

Jeśli żądanie jest poprawne, odpowiedź zwrócona przez API może wyglądać następująco:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "Item is disabled."
            },
            {
                "error": "No permissions to referred object or it does not exist."
            }
        ]
    },
    "id": 1
}
```

Błędy w danych odpowiedzi wskazują, że wysyłanie danych dla określonych pozycji nie przeszło walidacji przez serwer Zabbix. Może się tak zdarzyć z następujących powodów:

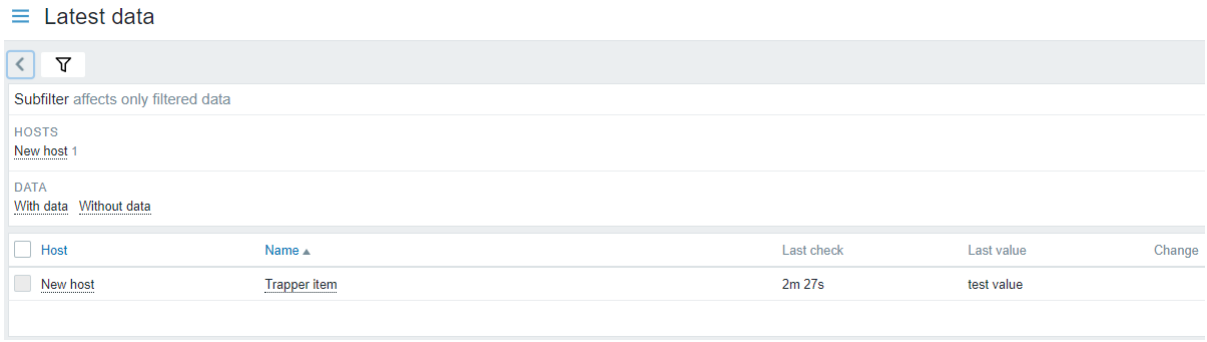
- użytkownik wysyłający dane nie ma uprawnień *read* do hosta danej pozycji;
- host jest wyłączony lub znajduje się w trybie maintenance bez zbierania danych;
- pozycja nie istnieje lub nie została jeszcze uwzględniona w pamięci podręcznej konfiguracji serwera;
- pozycja jest wyłączona lub jej typ jest inny niż Zabbix trapper lub HTTP agent (z włączonym trappingiem);
- adres IP lub DNS użytkownika nie jest ustawiony na liście *Allowed hosts* pozycji;
- inna pozycja ma wartość z duplikującym się znacznikiem czasu na poziomie nanosekund.

Brak błędów oznacza, że wysłane wartości zostały przyjęte do przetwarzania, które obejmuje preprocessing (jeśli występuje), przetwarzanie wyzwalaczy oraz zapis do bazy danych. Należy pamiętać, że przetwarzanie przyjętej wartości również może się nie powieść (na przykład podczas preprocessing), co spowoduje odrzucenie wartości.

Więcej informacji o pracy z Zabbix API można znaleźć w sekcji API.

Wyświetlanie danych

Po wysłaniu danych możesz przejść do *Monitoring* → *Latest data*, aby zobaczyć wynik:



Note:
Jeśli zostanie wysłana pojedyncza wartość liczbowa, wykres danych pokaże poziomą linię po lewej i prawej stronie punktu czasowego tej wartości.

7 Zewnętrzny check

Omówienie

Zewnętrzne sprawdzenie to sprawdzenie wykonywane przez serwer Zabbix przez uruchomienie skryptu powłoki lub pliku binarnego. Jednak gdy hosty są monitorowane przez proxy Zabbix, zewnętrzne sprawdzenia są wykonywane przez proxy.

Zewnętrzne sprawdzenia nie wymagają uruchomionego agenta na monitorowanym hoście.

Składnia klucza pozycji jest następująca:

```
script[<parameter1>,<parameter2>,...]
```

Gdzie:

ARGUMENT	DEFINICJA
script	Nazwa skryptu powłoki lub pliku binarnego.
parameter(s)	Opcjonalne parametry wiersza poleceń.

Jeśli nie chcesz przekazywać żadnych parametrów do skryptu, możesz użyć:

```
script[] or
script
```

Serwer Zabbix lub proxy wyszuka katalog określony dla skryptów zewnętrznych i wykona polecenie (zobacz parametr `ExternalScripts` w pliku konfiguracyjnym Zabbix serwer/proxy). Polecenie zostanie wykonane z uprawnieniami tego samego użytkownika co serwer Zabbix/proxy, więc w razie potrzeby wszelkie uprawnienia dostępu lub zmienne środowiskowe powinny być obsłużone w skrypcie opakowującym. Uprawnienia do polecenia powinny również umożliwiać temu użytkownikowi jego wykonanie. Do wykonania dostępne są tylko polecenia znajdujące się w określonym katalogu.

Warning:
Nie nadużywaj zewnętrznych sprawdzeń, ponieważ każdy skrypt wymaga uruchomienia procesu fork przez serwer/proxy Zabbix, a uruchamianie wielu skryptów może znacząco obniżyć wydajność Zabbix.

Przykład użycia

Wykonanie skryptu **check_oracle.sh** z pierwszym parametrem '-h'. Drugi parametr zostanie zastąpiony adresem IP lub nazwą DNS, w zależności od wyboru we właściwościach hosta.

```
check_oracle.sh["-h","{HOST.CONN}"]
```

Zakładając, że host jest skonfigurowany do używania adresu IP, serwer/proxy Zabbix wykona:

```
check_oracle.sh '-h' '192.168.1.4'
```

Wynik sprawdzenia zewnętrznego

Wartość zwracana przez sprawdzenie zewnętrzne to standardowe wyjście wraz ze standardowym wyjściem błędów wygenerowanym przez sprawdzenie.

Attention:
Pozycja, która zwraca tekst (informacje typu character, log lub text), nie stanie się nieobsługiwana w przypadku standardowego wyjścia błędów.

Wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.

Jeśli żądany skrypt nie zostanie znaleziony lub serwer/proxy Zabbixa nie ma uprawnień do jego wykonania, pozycja stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie.

W przypadku przekroczenia limitu czasu pozycja stanie się nieobsługiwana, zostanie wyświetlony odpowiedni komunikat o błędzie, a proces utworzony przez fork dla skryptu zostanie zakończony.

8 Monitorowanie bazy danych

Omówienie

Typ pozycja *Database monitor* w frontendzie Zabbix jest używany do monitorowania ODBC (sprawdzeń ODBC).

ODBC to pośredniczące API języka programowania C służące do uzyskiwania dostępu do systemów zarządzania bazami danych (DBMS). Koncepcja ODBC została opracowana przez firmę Microsoft, a następnie przeniesiona na inne platformy.

Zabbix może wykonywać zapytania do dowolnej bazy danych obsługiwanej przez ODBC. Aby to zrobić, Zabbix nie łączy się bezpośrednio z bazami danych, lecz korzysta z interfejsu ODBC oraz sterowników skonfigurowanych w ODBC. Umożliwia to bardziej efektywne monitorowanie różnych baz danych do wielu celów (na przykład sprawdzanie określonych kolejek bazy danych, statystyk użycia itp.).

Zabbix obsługuje unixODBC, który jest jedną z najczęściej używanych otwartoźródłowych implementacji API ODBC.

Attention:
Zobacz także: znane problemy dotyczące sprawdzeń ODBC.

Instalowanie unixODBC

Zalecanym sposobem instalacji unixODBC jest korzystanie z domyślnych repozytoriów pakietów systemu operacyjnego Linux. W najpopularniejszych dystrybucjach Linux unixODBC jest domyślnie dołączony do repozytorium pakietów. Jeśli pakiety nie są dostępne, pliki źródłowe można pobrać ze strony głównej unixODBC: http://www.unixodbc.org/download.html.

Aby zainstalować unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install unixodbc unixodbc-dev

##### Dla systemów opartych na RedHat/Fedora:
dnf install unixODBC unixODBC-devel

##### Dla systemów opartych na SUSE:
zypper in unixODBC-devel
```

Attention:
Pakiet `unixodbc-dev` lub `unixODBC-devel` jest niezbędny do skompilowania Zabbixa z obsługą unixODBC. Aby włączyć obsługę ODBC, Zabbix powinien zostać skompilowany z następującą opcją konfiguracji:

`--with-unixodbc[=ARG] # Use ODBC driver against unixODBC package.`

Instalowanie sterowników unixODBC

Sterownik bazy danych unixODBC powinien być zainstalowany dla bazy danych, która będzie monitorowana. Listę obsługiwanych baz danych i sterowników można znaleźć na stronie głównej unixODBC: http://www.unixodbc.org/drivers.html.

Note:
W niektórych dystrybucjach Linuksa sterowniki baz danych są dołączone do repozytoriów pakietów.

MySQL

Aby zainstalować sterownik bazy danych MySQL dla unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install odbc-mariadb

##### Dla systemów opartych na RedHat/Fedora:
dnf install mariadb-connector-odbc

##### Dla systemów opartych na SUSE:
zypper install mariadb-connector-odbc
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z dokumentacją MySQL dla `mysql-connector-odbc` lub z dokumentacją MariaDB dla `mariadb-connector-odbc`.

PostgreSQL

Aby zainstalować sterownik bazy danych PostgreSQL dla unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install odbc-postgresql

##### Dla systemów opartych na RedHat/Fedora:
dnf install postgresql-odbc

##### Dla systemów opartych na SUSE:
zypper install psqlODBC
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z dokumentacją PostgreSQL.

Oracle

Aby zainstalować sterownik bazy danych unixODBC, zapoznaj się z dokumentacją Oracle.

MSSQL

Aby zainstalować sterownik bazy danych MSSQL unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install tdsodbc

##### Dla systemów opartych na RedHat/Fedora (pakiety EPEL: https://docs.fedoraproject.org/en-US/epel/):
dnf install epel-release
dnf install freetds

##### Dla systemów opartych na SUSE:
zypper install libtdsodbc0
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z przewodnikiem użytkownika FreeTDS.

Konfigurowanie unixODBC

Aby skonfigurować unixODBC, należy edytować pliki `odbcinst.ini` i `odbc.ini`. Lokalizację tych plików można sprawdzić, wykonując następujące polecenie:

```
odbcinst -j
```

Wynik polecenia powinien zawierać informacje podobne do poniższych:

```
unixODBC 2.3.9
DRIVERS............: /etc/odbcinst.ini
SYSTEM DATA SOURCES: /etc/odbc.ini
FILE DATA SOURCES..: /etc/ODBCDataSources
```

odbcinst.ini

Plik `odbcinst.ini` zawiera listę zainstalowanych sterowników baz danych ODBC. Jeśli plik `odbcinst.ini` nie istnieje, należy utworzyć go ręcznie.

```
[TEST_MYSQL]
Description=ODBC for MySQL
Driver=/usr/lib/libmyodbc5.so
FileUsage=1
```

Parameter	Description
TEST_MYSQL	Nazwa sterownika bazy danych.
Description	Opis sterownika bazy danych.
Driver	Lokalizacja biblioteki sterownika bazy danych.
FileUsage	Określa, czy sterownik bazy danych obsługuje połączenie z serwerem bazy danych bez obsługi dostępu do plików lokalnych (0); obsługuje odczyt danych z plików (1); obsługuje zapis danych do plików (2).
Threading	Poziom serializacji wątków. Obsługiwane dla PostgreSQL. Od wersji 1.6, jeśli menedżer sterowników został zbudowany z obsługą wątków, można dodać kolejny wpis sterownika.

odbc.ini

Plik `odbc.ini` służy do konfigurowania źródeł danych. Należy pamiętać, że lista obsługiwanych parametrów zależy od sterownika bazy danych (na przykład bazy danych Oracle mogą używać parametru ServerName zamiast Server itd.).

```
[TEST_MYSQL]
Description=MySQL Test Database
Driver=mysql
Server=127.0.0.1
User=root
Password=
Port=3306
Socket=
Database=zabbix
```

Parameter	Description
TEST_MYSQL	Nazwa źródła danych (DSN).
Description	Opis źródła danych.
Driver	Nazwa sterownika bazy danych (zgodnie z określeniem w `odbcinst.ini`).
Server	IP/DNS serwera bazy danych.
User	Użytkownik bazy danych używany do połączenia.
Password	Hasło użytkownika bazy danych.
Port	Port połączenia z bazą danych.
Socket	Gniazdo połączenia z bazą danych.
Database	Nazwa bazy danych.

Aby poznać inne możliwe opcje parametrów konfiguracyjnych, zobacz dokumentację MySQL.

Plik `odbc.ini` dla PostgreSQL może zawierać dodatkowe parametry:

```
[TEST_PSQL]
Description=PostgreSQL Test Database
Driver=postgresql
Username=zbx_test
Password=zabbix
Servername=127.0.0.1
Database=zabbix
Port=5432
ReadOnly=No
Protocol=7.4+
ShowOidColumn=No
FakeOidIndex=No
RowVersioning=No
ShowSystemTables=No
Fetch=Yes
BoolsAsChar=Yes
SSLmode=Require
ConnSettings=
```

Parameter	Description
ReadOnly	Określa, czy połączenie z bazą danych zezwala wyłącznie na operacje odczytu (SELECT) i ogranicza modyfikacje (INSERT, UPDATE oraz DELETE); przydatne w scenariuszach, w których dane powinny pozostać niezmienione.
Protocol	Wersja protokołu backendu PostgreSQL (ignorowana podczas korzystania z połączeń SSL).
ShowOidColumn	Określa, czy w SQLColumns ma być uwzględniany Object ID (OID).
FakeOidIndex	Określa, czy ma zostać utworzony fikcyjny unikalny indeks na OID.
RowVersioning	Określa, czy aplikacje mają mieć możliwość wykrywania, czy dane zostały zmodyfikowane przez innych użytkowników podczas próby aktualizacji wiersza. Należy zauważyć, że ten parametr może przyspieszyć proces aktualizacji, ponieważ do zaktualizowania wiersza nie trzeba podawać każdej pojedynczej kolumny w klauzuli WHERE.
ShowSystemTables	Określa, czy sterownik bazy danych powinien traktować tabele systemowe jako zwykłe tabele w SQLTables; przydatne ze względu na dostępność, umożliwiając widoczność tabel systemowych.
Fetch	Określa, czy sterownik powinien automatycznie używać declare cursor/fetch do obsługi instrukcji SELECT i utrzymywać pamięć podręczną 100 wierszy.
BoolsAsChar	Kontroluje mapowanie typów logicznych. Jeśli ustawione na "Yes", wartości logiczne są mapowane na SQL_CHAR; w przeciwnym razie są mapowane na SQL_BIT.
SSLmode	Określa tryb SSL dla połączenia.
ConnSettings	Dodatkowe ustawienia wysyłane do backendu podczas nawiązywania połączenia.

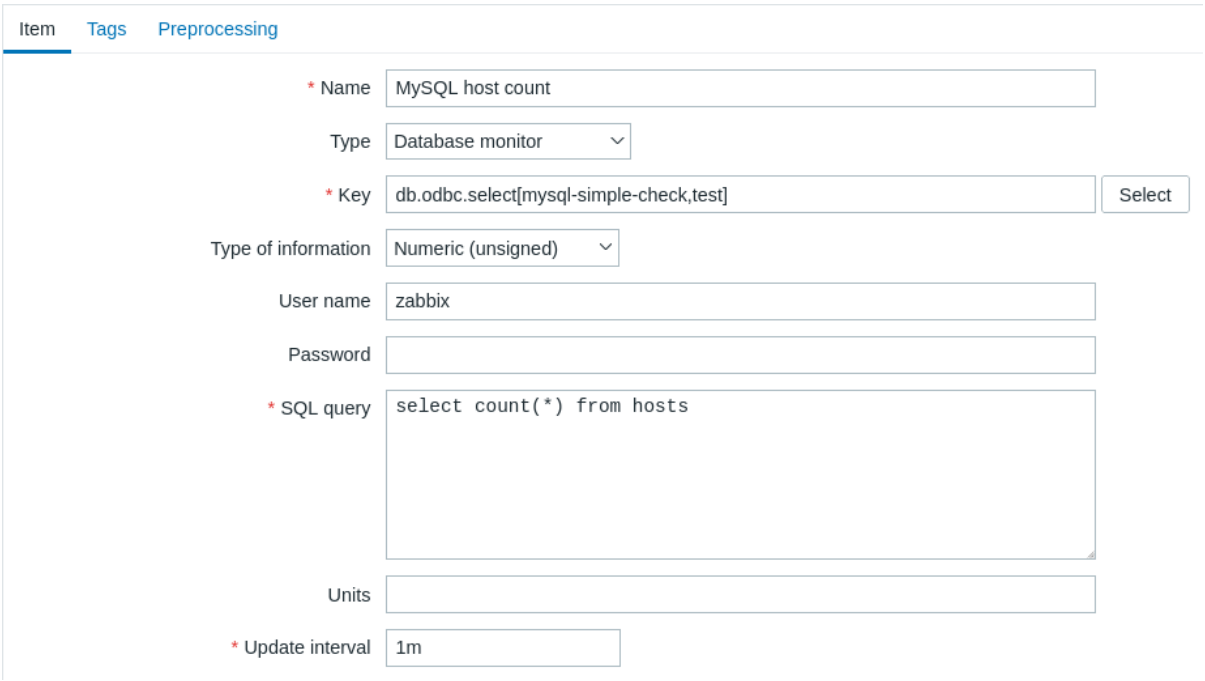
Testowanie połączenia ODBC

Aby sprawdzić, czy połączenie ODBC działa poprawnie, można użyć narzędzia `isql` (dołączonego do pakietu `unixODBC`):

```
isql test
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
|                                       |
+---------------------------------------+
```

Konfiguracja pozycji w frontendzie Zabbix

Skonfiguruj pozycję monitorowania **Database monitoring** item.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Dla pozycji monitorowania bazy danych należy określić:

Type	Wybierz tutaj "Database monitor".
Key	Wprowadź jeden z obsługiwanych kluczy pozycji: **db.odbc.select[]** - ta pozycja zwraca jedną wartość (pierwszą kolumnę pierwszego wiersza wyniku zapytania SQL); **db.odbc.get[]** - ta pozycja zwraca wiele wierszy/kolumn w formacie JSON; **db.odbc.discovery[]** - ta pozycja zwraca dane wykrywania niskiego poziomu.
User name	Wprowadź nazwę użytkownika bazy danych (do 255 znaków). Ten parametr jest opcjonalny, jeśli nazwa użytkownika bazy danych jest określona w pliku `odbc.ini`. Jeśli używany jest connection string, a pole *User name* nie jest puste, zostanie on dołączony do connection string jako `UID=<user>`.
Password	Wprowadź hasło użytkownika bazy danych (do 255 znaków). Ten parametr jest opcjonalny, jeśli hasło jest określone w pliku `odbc.ini`. Jeśli używany jest connection string, a pole *Password* nie jest puste, zostanie on dołączony do connection string jako `PWD=<password>`. Znaki specjalne są obsługiwane w tym polu. Hasło zostanie dołączone do connection string po nazwie użytkownika, na przykład jako `UID=<username>;PWD=P?;)*word`. Aby przetestować wynikowy ciąg, możesz uruchomić następujące polecenie: `isql -v -k 'Driver=libmaodbc.so;Database=zabbix;UID=zabbix;PWD=P?;)*word'`
SQL query	Wprowadź zapytanie SQL. Zwróć uwagę, że w przypadku `db.odbc.select[]` zapytanie musi zwracać tylko jedną wartość.
Type of information	Wybierz tutaj typ informacji, który będzie zwracany przez zapytanie. Jeśli typ informacji zostanie wybrany nieprawidłowo, pozycja stanie się nieobsługiwana.

Ważne uwagi

- Pozycje monitorowania bazy danych staną się nieobsługiwane, jeśli w konfiguracji serwera lub proxy nie zostaną uruchomione żadne procesy *odbc poller*. Aby aktywować pollery ODBC, ustaw parametr `StartODBCPollers` w pliku konfiguracyjnym Zabbix server lub, w przypadku kontroli wykonywanych przez proxy, w pliku konfiguracyjnym Zabbix proxy.
- Wartość parametru *Timeout* w formularzu konfiguracji pozycji jest używana jako limit czasu logowania ODBC oraz limit czasu wykonania zapytania. Należy pamiętać, że te ustawienia limitu czasu mogą zostać zignorowane, jeśli zainstalowany sterownik ODBC ich nie obsługuje.
- Polecenie SQL musi zwracać zestaw wyników, jak każde zapytanie używające instrukcji `select`. Składnia zapytania będzie zależeć od systemu RDBMS, który będzie je przetwarzał. Składnia żądania do procedury składowanej musi rozpoczynać się od słowa kluczowego `call`.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

db.odbc.select[<unikalny krótki opis>,<dsn>,<connection string>]

 Zwraca jedną wartość, czyli pierwszą kolumnę pierwszego wiersza wyniku zapytania SQL.
 Wartość zwracana: Zależy od zapytania SQL.

Parametry:

- **unikalny krótki opis** — unikalny krótki opis identyfikujący pozycja (do użycia w wyzwalaczach itp.);
- **dsn** — nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** — ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Uwagi:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli zdefiniowano oba, `dsn` zostanie zignorowany.
- Jeśli zapytanie zwraca więcej niż jedną kolumnę, odczytywana jest tylko pierwsza kolumna. Jeśli zapytanie zwraca więcej niż jeden wiersz, odczytywany jest tylko pierwszy wiersz.

db.odbc.get[<unique short description>,<dsn>,<connection string>]

 Przekształca wynik zapytania SQL do tablicy JSON.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **unique short description** - unikalny krótki opis służący do identyfikacji pozycji (do użycia w wyzwalaczach itp.);
- **dsn** - nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** - ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Uwagi:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli oba są zdefiniowane, `dsn` zostanie zignorowany.
- W formacie JSON mogą zostać zwrócone wiele wierszy/kolumn. Ta pozycja może być używana jako pozycja nadrzędna, która zbiera wszystkie dane w jednym wywołaniu systemowym, natomiast przetwarzanie JSONPath może być używane w pozycjach zależnych do wyodrębniania poszczególnych wartości. Więcej informacji można znaleźć w przykładzie zwracanego formatu, używanego w niskopoziomowym wykrywaniu.

Przykład:

```
# Połączenie dla sterownika MySQL ODBC 5:
db.odbc.get[MySQL example,,"Driver=/usr/local/lib/libmyodbc5a.so;Database=master;Server=127.0.0.1;Port=330
```

db.odbc.discovery[<unikalny krótki opis>,<dsn>,<connection string>]

 Przekształca wynik zapytania SQL w tablicę JSON, używaną do niskopoziomowego wykrywania. Nazwy kolumn z wyniku zapytania są przekształcane w nazwy makr niskopoziomowego wykrywania sparowane z wartościami wykrytych pól. Makra te mogą być używane podczas tworzenia prototypów pozycji, wyzwalaczy itp.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **unikalny krótki opis** — unikalny krótki opis identyfikujący pozycję (do użycia w wyzwalaczach itp.);
- **dsn** — nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** — ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Uwagi:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli zdefiniowano oba, `dsn` zostanie zignorowany.

Komunikaty o błędach

Komunikaty o błędach ODBC są podzielone na pola, aby dostarczyć szczegółowych informacji. Na przykład komunikat o błędzie może wyglądać tak:

```
Cannot execute ODBC query: [SQL_ERROR]:[42601][7][ERROR: syntax error at or near ";"; Error while executin
```

- "`Cannot execute ODBC query`" — komunikat Zabbix
- "`[SQL_ERROR]`" — kod zwrotny ODBC
- "`[42601]`" — SQLState
- "`[7]`" — natywny kod błędu
- "`[ERROR: syntax error at or near ";"; Error while executing the query]`" — natywny komunikat błędu

Należy pamiętać, że długość komunikatu o błędzie jest ograniczona do 2048 bajtów, więc komunikat może zostać obcięty. Jeśli istnieje więcej niż jeden rekord diagnostyczny ODBC, Zabbix próbuje je połączyć (oddzielając je znakiem |) w zakresie dozwolonym przez limit długości.

9 HTTP agent

Omówienie

Ten typ pozycji umożliwia odpytywanie danych przy użyciu protokołu HTTP/HTTPS. Możliwe jest również trapowanie za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender (do wysyłania danych do serwera Zabbix lub proxy), albo przy użyciu metody API `history.push` (do wysyłania danych do serwera Zabbix).

Kontrole pozycji HTTP są wykonywane przez serwer Zabbix. Jednak gdy hosty są monitorowane przez proxy Zabbix, kontrole pozycji HTTP są wykonywane przez proxy.

Kontrole pozycji HTTP nie wymagają uruchomionego agenta na monitorowanym hoście.

HTTP agent obsługuje zarówno HTTP, jak i HTTPS. Zabbix opcjonalnie będzie podążać za przekierowaniami (zobacz opcję *Follow redirects* poniżej). Maksymalna liczba przekierowań jest na stałe ustawiona na 10 (z użyciem opcji cURL CURLOPT_MAXREDIRS).

> **Attention:**
> Serwer/proxy Zabbix musi być początkowo skonfigurowany z obsługą cURL (libcurl).

Kontrole HTTP są wykonywane asynchronicznie - nie jest wymagane otrzymanie odpowiedzi na jedno żądanie przed rozpoczęciem innych kontroli. Rozwiązywanie DNS również odbywa się asynchronicznie.

Maksymalna współbieżność kontroli asynchronicznych wynosi 1000 (zdefiniowana przez parametr MaxConcurrentChecksPerPoller).

Liczba asynchronicznych pollerów HTTP agent jest określana przez parametr StartHTTPAgentPollers.

Funkcja trwałych połączeń cURL została dodana do kontroli HTTP agent od Zabbix 7.0.

Konfiguracja

Aby skonfigurować pozycję HTTP:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij *Items* w wierszu hosta.
3. Kliknij *Create item*.
4. Wprowadź parametry pozycji w formularzu.

Item | Tags | Preprocessing

* Name: HTTP agent item

Type: HTTP agent

* Key: http_value_search [Select]

Type of information: Numeric (unsigned)

* URL: http://localhost:9200/str/values/_search [Parse]

Query fields:

Name		Value	
scroll	⇒	10s	Remove

Add

Request type: POST

Request body type: Raw data | JSON data | XML data

Request body:

```
{
  "query":{
    "bool":{
      "must":{
       "match":{
         "itemid":28275
}
```

Headers:

Name		Value	
name	⇒	value	Remove

Add

Required status codes: 200

Follow redirects: ☑

Retrieve mode: Body | Headers | Body and headers

Convert to JSON: ☐

HTTP proxy: [protocol://][user[:password]@]proxy.example.com[:port]

HTTP authentication: None

SSL verify peer: ☐

SSL verify host: ☐

SSL certificate file:

SSL key file:

SSL key password:

Host interface: 127.0.0.1:10050

Units:

* Update interval: 1m

Custom intervals:

Type	Interval	Period	Action
Flexible | Scheduling	50s	1-7,00:00-24:00	Remove

Add

* Timeout: Global | Override 3 Timeouts

* History: Do not store | Store up to 31d

* Trends: Do not store | Store up to 365d

Value mapping: type here to search [Select]

Enable trapping: ☐

Populates host inventory field: -None-

Description:

Enabled: ☑

Add | Test | Cancel

Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji HTTP to:

Parameter	Description
Type	Wybierz tutaj **HTTP agent**.
Key	Wprowadź unikalny klucz pozycji.
URL	Adres URL do połączenia i pobrania danych. Na przykład: https://www.example.com http://www.example.com/download Nazwy domen mogą być podane w znakach Unicode. Podczas wykonywania sprawdzenia HTTP są one automatycznie konwertowane do ASCII w formacie punycode. Przycisk *Parse* można użyć do oddzielenia opcjonalnych pól zapytania (takich jak ?name=Admin&password=mypassword) od adresu URL, przenosząc atrybuty i wartości do *Query fields* w celu automatycznego kodowania URL. Limit: 2048 znaków. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_URL.
Query fields	Zmienne dla adresu URL (patrz wyżej). Określane jako pary atrybut i wartość. Wartości są automatycznie kodowane URL. Wartości z makr są najpierw rozwijane, a następnie automatycznie kodowane URL. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_URL.
Request type	Wybierz typ metody żądania: *GET*, *POST*, *PUT* lub *HEAD*
Request body type	Wybierz typ treści żądania: **Raw data** - niestandardowa treść żądania HTTP, makra są podstawiane, ale nie jest wykonywane kodowanie **JSON data** - treść żądania HTTP w formacie JSON. Makra mogą być używane jako string, number, true i false; makra używane jako stringi muszą być ujęte w podwójne cudzysłowy. Wartości z makr są najpierw rozwijane, a następnie automatycznie escape'owane. Jeśli nagłówek "Content-Type" nie zostanie określony, domyślnie zostanie ustawiony "Content-Type: application/json" **XML data** - treść żądania HTTP w formacie XML. Makra mogą być używane jako węzeł tekstowy, atrybut lub sekcja CDATA. Wartości z makr są najpierw rozwijane, a następnie automatycznie escape'owane w węźle tekstowym i atrybucie. Jeśli nagłówek "Content-Type" nie zostanie określony, domyślnie zostanie ustawiony "Content-Type: application/xml" *Uwaga*: wybranie *XML data* wymaga libxml2.
Request body	Wprowadź treść żądania. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania.
Headers	Niestandardowe nagłówki HTTP, które zostaną wysłane podczas wykonywania żądania. Określane jako pary atrybut i wartość. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_HTTPHEADER.
Required status codes	Lista oczekiwanych kodów statusu HTTP. Jeśli Zabbix otrzyma kod, którego nie ma na liście, pozycja stanie się nieobsługiwana. Jeśli pole jest puste, nie jest wykonywane żadne sprawdzenie. Na przykład: 200,201,210-299 Obsługiwane makra na liście: makra użytkownika, makra niskopoziomowego wykrywania. Używa opcji cURL CURLINFO_RESPONSE_CODE.
Follow redirects	Zaznacz pole wyboru, aby podążać za przekierowaniami HTTP. Ustawia opcję cURL CURLOPT_FOLLOWLOCATION.
Retrieve mode	Wybierz część odpowiedzi, która ma zostać pobrana: **Body** - tylko treść **Headers** - tylko nagłówki **Body and headers** - treść i nagłówki

Parameter	Description
Convert to JSON	Nagłówki są zapisywane jako pary atrybut i wartość pod kluczem ”header”, jeśli *Retrieve mode* nie jest ustawiony na *Body*. Jeśli zostanie wykryte ’Content-Type: application/json’, treść jest zapisywana jako obiekt, w przeciwnym razie jest przechowywana jako string, na przykład: `{ "header": { "<key>": "<value>", "<key2>": "<value>" }, "body": <body> }`
HTTP proxy	Możesz określić HTTP proxy do użycia, stosując format `[protocol://][username[:password]@]proxy.example.com[:port]`. Opcjonalny prefiks `protocol://` może być użyty do określenia alternatywnych protokołów proxy (np. https, socks4, socks5; zobacz dokumentację; obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli nie określono protokołu, proxy będzie traktowany jako HTTP proxy. Jeśli podasz niewłaściwy protokół, połączenie nie powiedzie się, a pozycja stanie się nieobsługiwana. Domyślnie używany będzie port 1080. Jeśli zostanie określony, proxy nadpisze zmienne środowiskowe związane z proxy, takie jak http_proxy, HTTPS_PROXY. Jeśli nie zostanie określony, proxy nie nadpisze zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana ”as is”, bez sprawdzania poprawności. *Uwaga*: z HTTP proxy obsługiwane jest tylko proste uwierzytelnianie. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_PROXY.
HTTP authentication	Wybierz opcję uwierzytelniania: **None** - bez uwierzytelniania; **Basic** - używane jest uwierzytelnianie podstawowe; **NTLM** - używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** - używane jest uwierzytelnianie Kerberos (zobacz także: Konfigurowanie Kerberos z Zabbix); **Digest** - używane jest uwierzytelnianie Digest. Ustawia opcję cURL CURLOPT_HTTPAUTH.
User name	Wprowadź nazwę użytkownika (do 255 znaków). To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika i makra niskopoziomowego wykrywania.
Password	Wprowadź hasło użytkownika (do 255 znaków). To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika i makra niskopoziomowego wykrywania.
SSL verify peer	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera WWW. Certyfikat serwera zostanie automatycznie pobrany z systemowej lokalizacji urzędu certyfikacji (CA). Możesz zastąpić lokalizację plików CA, używając parametru konfiguracji Zabbix serwer lub proxy SSLCALocation. Ustawia opcję cURL CURLOPT_SSL_VERIFYPEER.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole Common Name lub pole Subject Alternate Name w certyfikacie serwera WWW jest zgodne. Ustawia opcję cURL CURLOPT_SSL_VERIFYHOST.
SSL certificate file	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole pliku klucza SSL puste. Jeśli klucz jest zaszyfrowany, podaj hasło w polu hasła klucza SSL. Katalog zawierający ten plik jest określany przez parametr konfiguracji Zabbix serwer lub proxy SSLCertLocation. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_SSLCERT.

Parameter	Description
SSL key file	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Katalog zawierający ten plik jest określany przez parametr konfiguracji Zabbix serwer lub proxy SSLKeyLocation. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_SSLKEY.
SSL key password	Hasło prywatnego klucza SSL. Obsługiwane makra: makra użytkownika, makra niskopoziomowego wykrywania. Ustawia opcję cURL CURLOPT_KEYPASSWD.
Timeout	Zabbix nie poświęci na przetwarzanie adresu URL więcej niż ustawiony czas (1-600 sekund). W praktyce ten parametr określa maksymalny czas na nawiązanie połączenia z adresem URL oraz maksymalny czas wykonania żądania HTTP. W związku z tym Zabbix nie poświęci na jedno sprawdzenie więcej niż 2 x *Timeout* sekund. Ustawia opcję cURL CURLOPT_TIMEOUT. Więcej informacji o parametrze *Timeout* znajdziesz w sekcji ogólne atrybuty pozycji.
Enable trapping	Po zaznaczeniu tego pola wyboru pozycja będzie również działać jako pozycja trapper i będzie akceptować dane wysyłane do serwer lub proxy Zabbix za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender protocol, albo będzie akceptować dane wysyłane do serwer Zabbix za pomocą metody API `history.push`. Więcej informacji o wysyłaniu danych znajdziesz w sekcji: Pozycje trapper.
Allowed hosts	Widoczne tylko wtedy, gdy zaznaczono pole wyboru *Enable trapping*. Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS. Jeśli zostanie określona, połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' pozwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może być użyte, aby zezwolić na dowolny adres IPv4. Zwróć uwagę, że "IPv4-compatible IPv6 addresses" (prefiks 0000::/96) są obsługiwane, ale zostały uznane za przestarzałe przez RFC4291. Przykład: 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} W tym polu dozwolone są spacje i makra użytkownika. W tym polu dozwolone są makra hosta: {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}.

Note:
Jeśli pole *HTTP proxy* pozostanie puste, innym sposobem użycia HTTP proxy jest ustawienie zmiennych środowiskowych związanych z proxy.
Dla HTTP - ustaw zmienną środowiskową `http_proxy` dla użytkownika serwer Zabbix. Na przykład:
 `http_proxy=http://proxy_ip:proxy_port`.
Dla HTTPS - ustaw zmienną środowiskową `HTTPS_PROXY`. Na przykład:
 `HTTPS_PROXY=http://proxy_ip:proxy_port`.
Więcej szczegółów można uzyskać, uruchamiając polecenie powłoki: *# man curl*.

Attention:
[1] Zabbix obsługuje pliki certyfikatów i prywatnych kluczy wyłącznie w formacie PEM. Jeśli certyfikat i prywatny klucz są zapisane w pliku w formacie PKCS #12 (zwykle z rozszerzeniem *.p12 lub *.pfx), możesz wygenerować z niego plik PEM za pomocą następujących poleceń:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

Przykłady

Przykład 1

Wyślij proste żądania GET, aby pobrać dane z usług takich jak Elasticsearch:

- Utwórz pozycja GET z adresem URL: `localhost:9200/?pretty`
- Zwróć uwagę na odpowiedź:

```
{
  "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" : "Wiesz, do wyszukiwania"
}
```

- Teraz wyodrębnij numer wersji, używając kroku przetwarzania JSONPath: `$.version.number`

Przykład 2

Wyślij proste żądania POST, aby pobrać dane z usług takich jak Elasticsearch:

- Utwórz pozycję POST z adresem URL: `http://localhost:9200/str/values/_search?scroll=10s`
- Skonfiguruj następujące ciało POST, aby uzyskać obciążenie procesora (średnia z 1 minuty na rdzeń)

```
{
    "query": {
        "bool": {
            "must": [{
                "match": {
                    "itemid": 28275
                }
            }],
            "filter": [{
                "range": {
                    "clock": {
                        "gt": 1517565836,
                        "lte": 1517566137
                    }
                }
            }]
        }
    }
}
```

- Otrzymano:

```
{
    "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAkFllRMlZBWS1UU1pxTmdEeGVwQjRBTFEAAAAAAAAAJRZZUTJWQVk
    "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": {
```

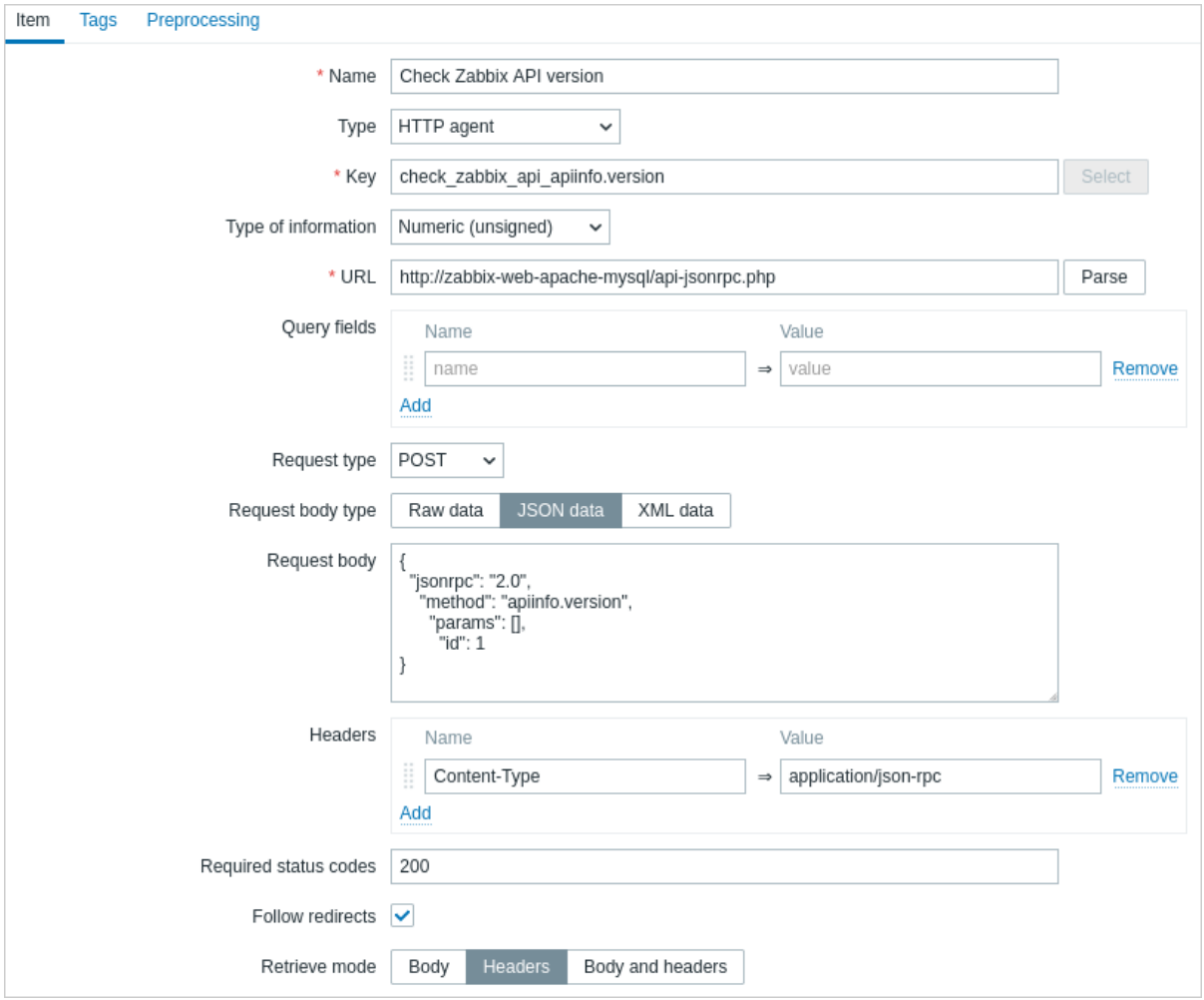
```
                "itemid": 28275,
                "value": "0.138750",
                "clock": 1517566136,
                "ns": 25388713,
                "ttl": 604800
            }
        }]
    }
}
```

- Teraz użyj kroku przetwarzania JSONPath, aby pobrać wartość pozycji: `$.hits.hits[0]._source.value`

Przykład 3

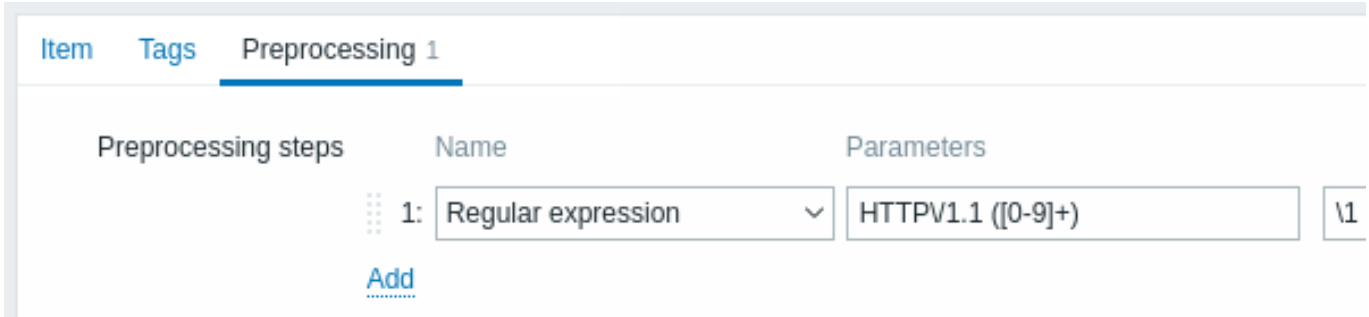
Sprawdzanie, czy Zabbix API działa, z użyciem apiinfo.version.

- Konfiguracja pozycji:

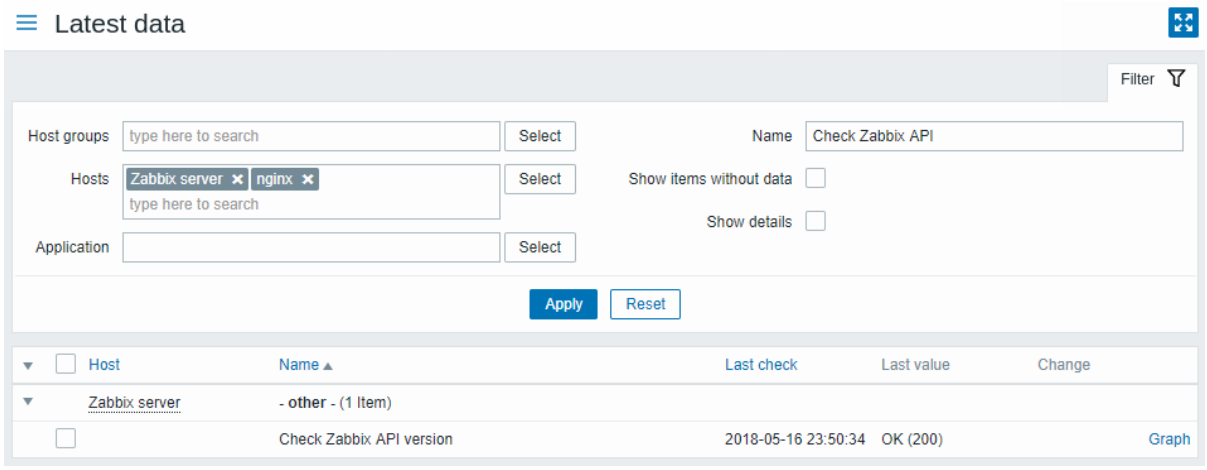


Zwróć uwagę na użycie metody POST z danymi JSON, ustawienie nagłówków żądania i żądanie zwrócenia tylko nagłówków:

- Wstępne przetwarzanie wartości pozycji za pomocą wyrażenia regularnego w celu uzyskania kodu HTTP:



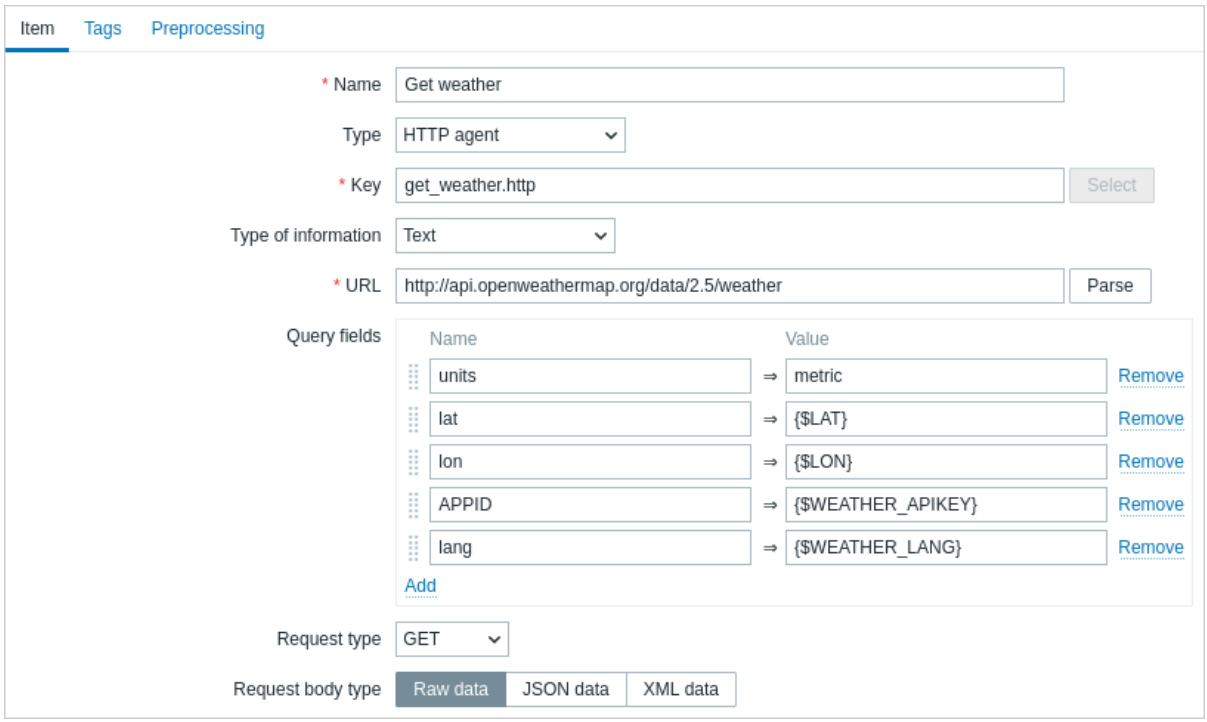
- Sprawdzanie wyniku w *Latest data*:



Przykład 4

Pobieranie informacji o pogodzie przez połączenie z publiczną usługą Openweathermap.

- Skonfiguruj pozycję nadrzędną do zbiorczego pobierania danych w jednym JSON:



Zwróć uwagę na użycie makr w polach zapytania. Informacje o tym, jak je wypełnić, znajdziesz w Openweathermap API.

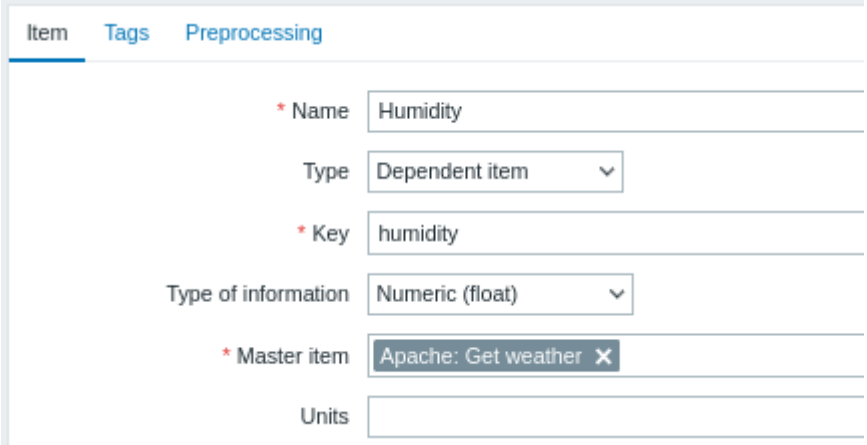
Przykładowy JSON zwracany w odpowiedzi przez agent HTTP:

```
{
    "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
    }
}
```

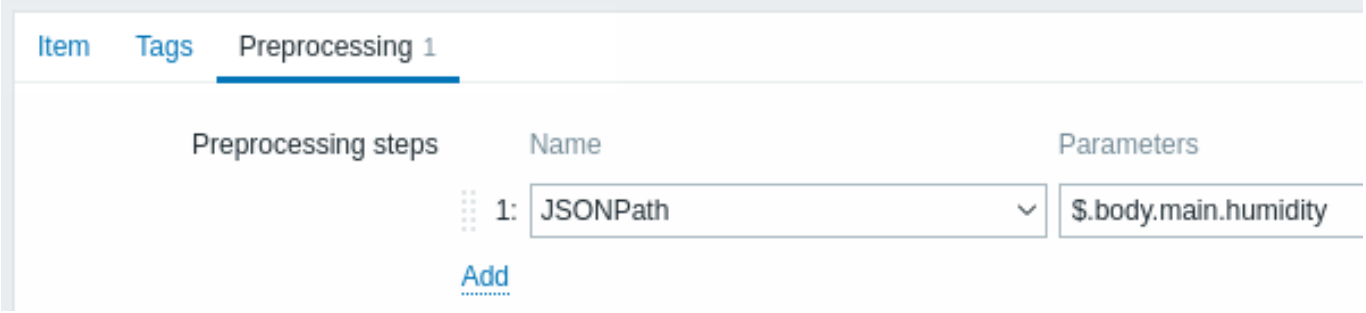

Następnym zadaniem jest skonfigurowanie pozycji zależnych, które wyodrębniają dane z JSON.

- Skonfiguruj przykładową pozycję zależną dla wilgotności:

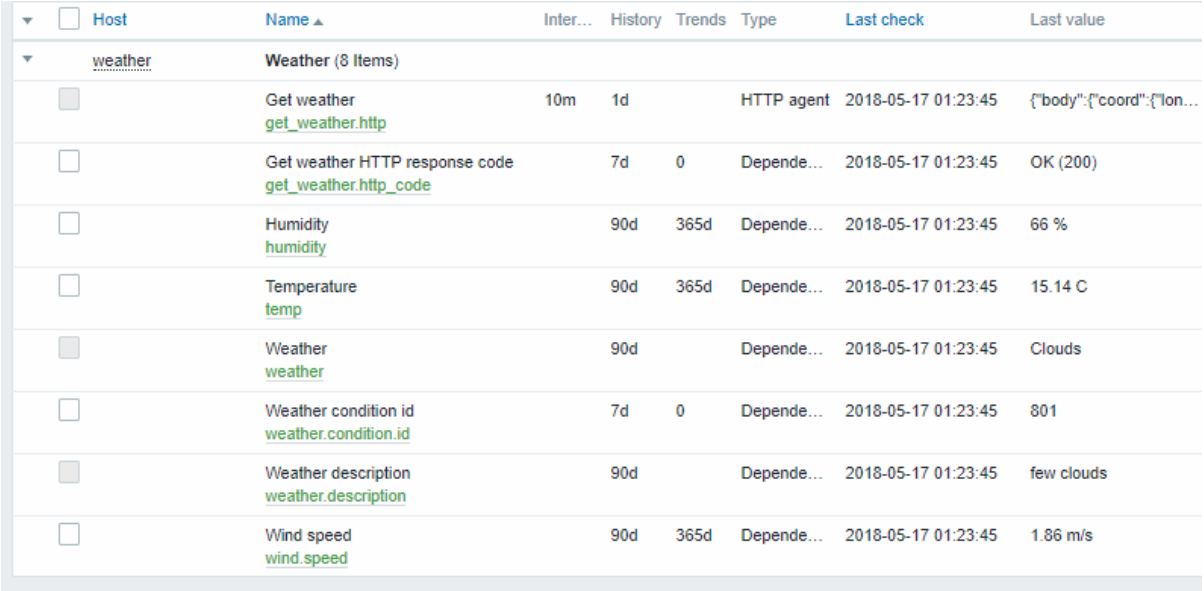


Pozostałe metryki pogodowe, takie jak „Temperatura”, dodaje się w ten sam sposób.

- Przykładowe przetwarzanie wstępne wartości pozycji zależnej z użyciem JSONPath:



- Sprawdź wynik danych pogodowych w *Latest data*:

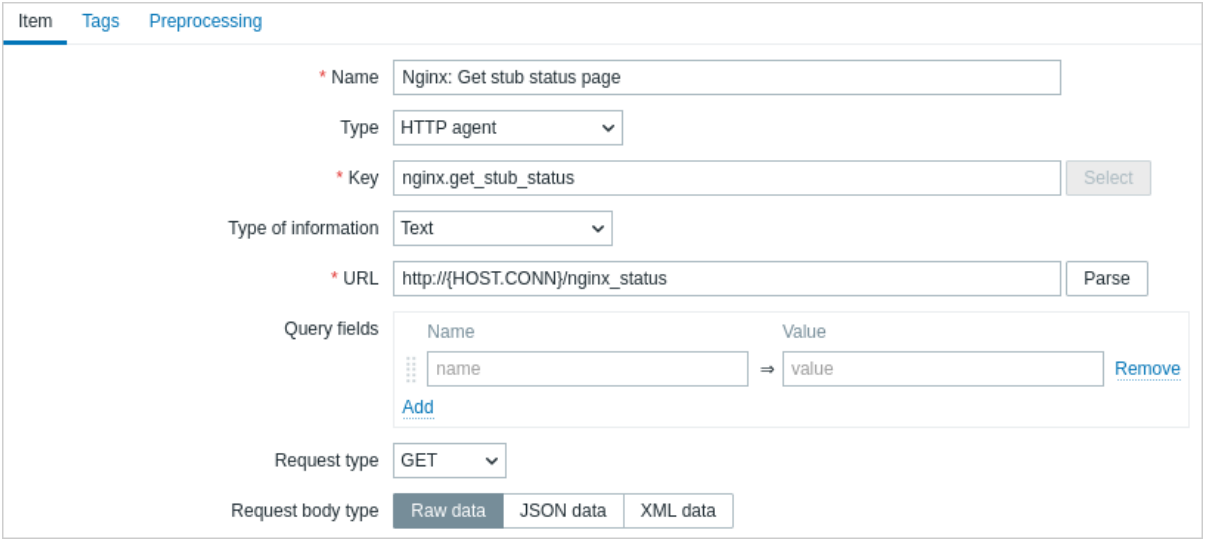


Host	Name ▲	Inter...	History	Trends	Type	Last check	Last value
weather	**Weather** (8 Items)						
	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{"body":{"coord":{"lon...
	Get weather HTTP response code get_weather.http_code		7d	0	Depende...	2018-05-17 01:23:45	OK (200)
	Humidity humidity		90d	365d	Depende...	2018-05-17 01:23:45	66 %
	Temperature temp		90d	365d	Depende...	2018-05-17 01:23:45	15.14 C
	Weather weather		90d		Depende...	2018-05-17 01:23:45	Clouds
	Weather condition id weather.condition.id		7d	0	Depende...	2018-05-17 01:23:45	801
	Weather description weather.description		90d		Depende...	2018-05-17 01:23:45	few clouds
	Wind speed wind.speed		90d	365d	Depende...	2018-05-17 01:23:45	1.86 m/s

Przykład 5

Łączenie się ze stroną statusu Nginx i pobieranie jej metryk zbiorczo.

- Skonfiguruj Nginx zgodnie z oficjalnym przewodnikiem.
- Skonfiguruj pozycję nadrzędną do zbiorczego zbierania danych:

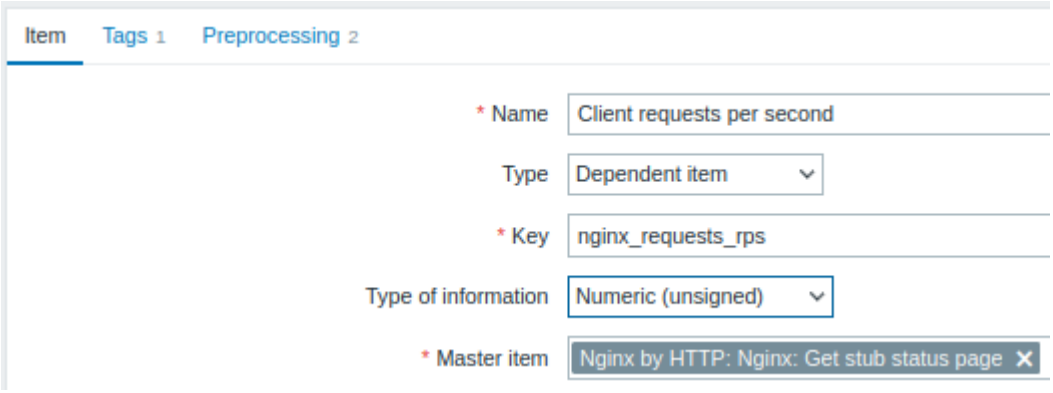


Przykładowy wynik stub status Nginx:

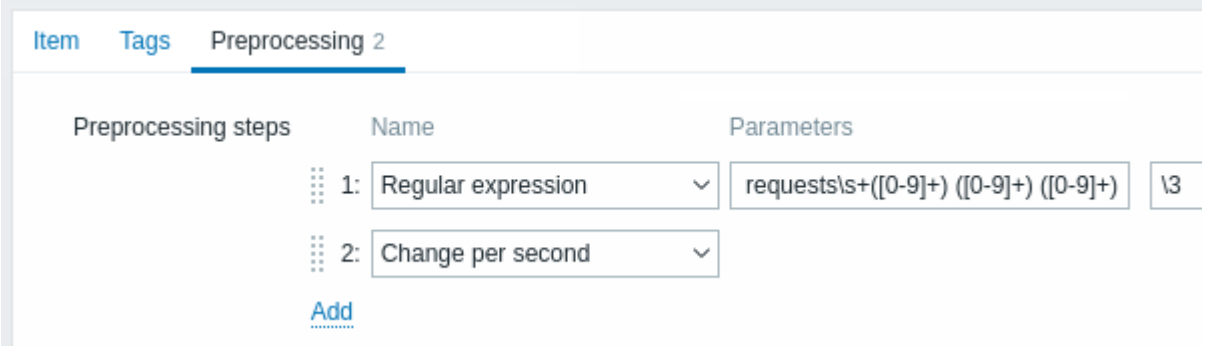
```
Active connections: 1 Active connections:
server accepts handled requests
 52 52 52
Reading: 0 Writing: 1 Waiting: 0
```

Następnym zadaniem jest skonfigurowanie pozycji zależnych, które wyodrębniają dane.

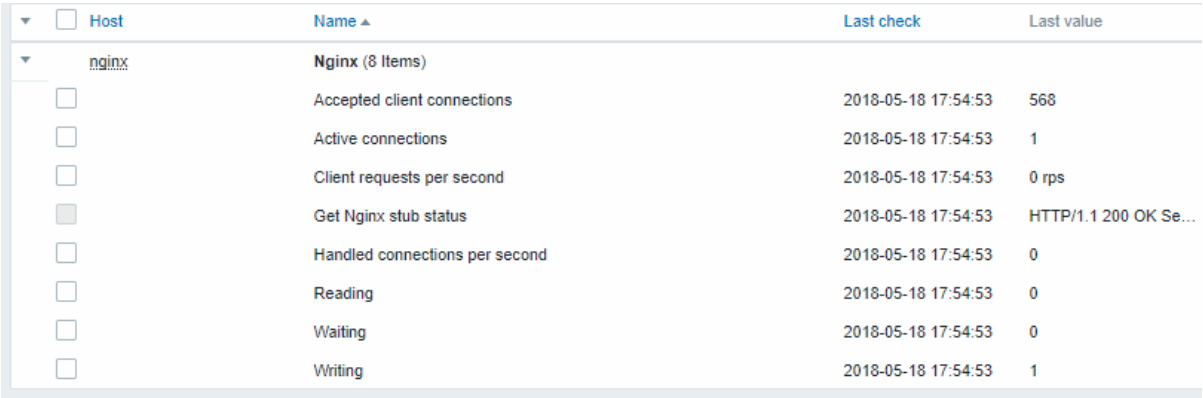
- Skonfiguruj przykładową pozycję zależną dla liczby żądań na sekundę:



- Przykładowe przetwarzanie wstępne wartości pozycji zależnej z użyciem wyrażenia regularnego `server accepts handled requests\s+([0-9]+) ([0-9]+) ([0-9]+)`:



- Sprawdź pełny wynik z modułu stub w *Latest data*:



1 Sprawdzenie Prometheusa

Przegląd

Zabbix może odpytywać metryki udostępniane w formacie liniowym Prometheus.

Aby rozpocząć zbieranie danych Prometheus, wymagane są dwa kroki:

- główna pozycja HTTP wskazująca na odpowiedni punkt końcowy danych, np. `https://<prometheus host>/metrics`
- pozycje zależne korzystające z opcji przetwarzania wstępnego Prometheus do odpytywania wymaganych danych z metryk zebranych przez główną pozycję

Istnieją dwie opcje przetwarzania wstępnego danych Prometheus:

- *Wzorzec Prometheus* - używany w zwykłych pozycjach do odpytywania danych Prometheus
- *Prometheus do JSON* - używany w zwykłych pozycjach oraz do wykrywania niskiego poziomu. W tym przypadku odpytywane dane Prometheus są zwracane w formacie JSON.

Przetwarzanie zbiorcze

Przetwarzanie zbiorcze jest obsługiwane dla pozycji zależnych. Aby włączyć buforowanie i indeksowanie, preprocessing *Prometheus pattern* musi być **pierwszym** krokiem preprocessing. Gdy *Prometheus pattern* jest pierwszym krokiem preprocessing, przeanalizowane dane Prometheus są buforowane i indeksowane przez pierwszy warunek `<label>==<value>` w kroku

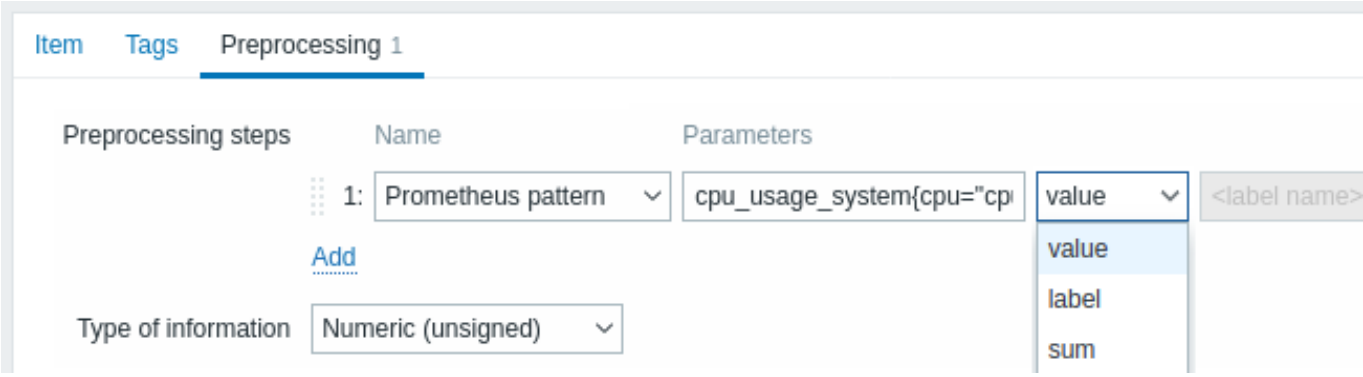
preprocessing *Prometheus pattern*. Ta pamięć podręczna jest ponownie używana podczas przetwarzania innych pozycji zależnych w tym wsadzie. Aby uzyskać optymalną wydajność, pierwsza etykieta powinna być tą, która ma najwięcej różnych wartości.

Jeśli przed pierwszym krokiem trzeba wykonać inne preprocessing, należy go przenieść albo do pozycji głównej, albo do nowej pozycji zależnej, która będzie używana jako pozycja główna dla pozycji zależnych.

Konfiguracja

Zakładając, że skonfigurowano nadrzędną pozycję HTTP, należy utworzyć pozycję zależną, która używa kroku przetwarzania Prometheus:

- Wprowadź ogólne parametry pozycji zależnej w formularzu konfiguracji
- Przejdź do zakładki Przetwarzanie wstępne
- Wybierz opcję przetwarzania wstępnego Prometheus (*Prometheus pattern* lub *Prometheus to JSON*)



Poniższe parametry są specyficzne dla opcji przetwarzania wstępnego *Prometheus pattern*:

Parameter	Description	Examples
Pattern	Aby zdefiniować wymagany wzorzec danych, możesz użyć języka zapytań podobnego do języka zapytań Prometheus (zob. tabela porównawcza), np.: <metric name> - wybór według nazwy metryki {__name__=”<metric name>”} - wybór według nazwy metryki {__name__=~”<regex>”} - wybór według nazwy metryki pasującej do wyrażenia regularnego {<label name>=”<label value>”,...} - wybór według nazwy etykiety {<label name>=~”<regex>”,...} - wybór według nazwy etykiety pasującej do wyrażenia regularnego {__name__=~”.*”}==<value> - wybór według wartości metryki Lub kombinację powyższych: <metric name>{<label1 name>=”<label1 value>”,<label2 name>=~”<regex>”,...}==<value> Wartość etykiety może być dowolnym ciągiem znaków UTF-8, jednak znak ukośnika odwrotnego, cudzysłów i znak nowej linii muszą być odpowiednio poprzedzone znakami \\, \" i \n; pozostałych znaków nie należy poprzedzać.	*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*

Parameter	Description	Examples
Result processing	Określ, czy zwrócić wartość, etykietę, czy zastosować odpowiednią funkcję (jeśli wzorzec pasuje do kilku wierszy i wynik musi zostać zagregowany): **value** - zwróć wartość metryki (błąd, jeśli dopasowano wiele wierszy) **label** - zwróć wartość etykiety określonej w polu *Label* (błąd, jeśli dopasowano wiele metryk) **sum** - zwróć sumę wartości **min** - zwróć wartość minimalną **max** - zwróć wartość maksymalną **avg** - zwróć wartość średnią **count** - zwróć liczbę wartości To pole jest dostępne tylko dla opcji *Prometheus pattern*.	Zobacz także przykłady użycia parametrów poniżej.
Output	Zdefiniuj nazwę etykiety (opcjonalnie). W takim przypadku zwracana jest wartość odpowiadająca nazwie etykiety. To pole jest dostępne tylko dla opcji *Prometheus pattern*, jeśli w polu *Result processing* wybrano opcję 'Label'.	

Przykłady użycia parametrów

1. Najczęstszym przypadkiem użycia jest zwrócenie **wartości**. Aby zwrócić wartość `/var/db` z:

   `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`

użyj następujących parametrów:

- *Pattern* - `node_disk_usage_bytes{path="/var/db"}`
- *Result processing* - wybierz 'value'

2. Możesz być również zainteresowany **średnią** wartością wszystkich parametrów `node_disk_usage_bytes`:

- *Pattern* - `node_disk_usage_bytes`
- *Result processing* - wybierz 'avg'

3. Chociaż Prometheus obsługuje wyłącznie dane liczbowe, popularnym obejściem jest zwracanie również odpowiedniego opisu tekstowego. Można to osiągnąć za pomocą filtra i określenia etykiety. Zatem, aby zwrócić wartość etykiety 'color' z

   `elasticsearch_cluster_health_status{cluster="elasticsearch",color="green"} 1<br> elasticsearch_cluster 0<br> elasticsearch_cluster_health_status{cluster="elasticsearch",color="yellow"} 0`

użyj następujących parametrów:

- *Pattern* - `elasticsearch_cluster_health_status {cluster="elasticsearch"} == 1`
- *Result processing* - wybierz 'label'
- *Label* - określ 'color'

Filtr (oparty na wartości liczbowej '1') dopasuje odpowiedni wiersz, natomiast etykieta zwróci opis stanu zdrowia (obecnie 'green'; ale potencjalnie także 'red' lub 'yellow').

Prometheus do JSON

Dane z Prometheus mogą być używane do niskopoziomowego wykrywania. W takim przypadku potrzebne są dane w formacie JSON, a opcja wstępnego przetwarzania *Prometheus to JSON* zwróci dokładnie taki wynik.

Więcej informacji można znaleźć w sekcji Wykrywanie z użyciem danych Prometheus.

Porównanie języka zapytań

Poniższa tabela przedstawia różnice i podobieństwa między PromQL a językiem zapytań przetwarzania wstępnego Zabbix Prometheus.

Selektor wektora chwilowego PromQL		Zabbix Prometheus preprocessing
Różnice		
Cel zapytania	serwer Prometheus	Zwykły tekst w formacie ekspozycji Prometheus
Zwraca	Wektor chwilowy	Wartość metryki lub etykiety (wzorzec Prometheus) Tablica metryk dla pojedynczej wartości w JSON (Prometheus to JSON)
Operatory dopasowania etykiet	**=**, **!=**, **=~**, **!~**	**=**, **!=**, **=~**, **!~**
Wyrażenie regularne używane w dopasowywaniu nazwy etykiety lub metryki	RE2	PCRE
Operatory porównania	Zobacz listę	Obsługiwany jest tylko operator **==** (równe) do filtrowania wartości
Podobieństwa		
Wybieranie według nazwy metryki równej ciągowi znaków	<metric name> lub {__name__=”<metric name>”}	<metric name> lub {__name__=”<metric name>”}
Wybieranie według nazwy metryki pasującej do wyrażenia regularnego	{__name__=~”<regex>”}	{__name__=~”<regex>”}
Wybieranie według wartości <label name> równej ciągowi znaków	{<label name>=”<label value>”,...}	{<label name>=”<label value>”,...}
Wybieranie według wartości <label name> pasującej do wyrażenia regularnego	{<label name>=~”<regex>”,...}	{<label name>=~”<regex>”,...}
Wybieranie według wartości równej ciągowi znaków	{__name__=~”.*”} == <value>	{__name__=~”.*”} == <value>

10 IPMI agent

Omówienie

Możesz monitorować stan i dostępność urządzeń Intelligent Platform Management Interface (IPMI) w Zabbix. Aby wykonywać kontrole IPMI, serwer Zabbix musi być początkowo skonfigurowany z obsługą IPMI.

IPMI to standardowy interfejs do zdalnego zarządzania typu ”lights-out” lub ”out-of-band” systemami komputerowymi. Umożliwia monitorowanie stanu sprzętu bezpośrednio z tak zwanych kart zarządzania ”out-of-band”, niezależnie od systemu operacyjnego lub od tego, czy maszyna jest w ogóle włączona.

Monitorowanie IPMI w Zabbix działa tylko dla urządzeń obsługujących IPMI (HP iLO, DELL DRAC, IBM RSA, Sun SSP itp.).

Proces menedżera IPMI planuje kontrole IPMI wykonywane przez pollery IPMI. Host jest zawsze odpytywany w danym momencie tylko przez jeden poller IPMI, co zmniejsza liczbę otwartych połączeń do kontrolerów BMC. Dzięki temu można bezpiecznie zwiększyć liczbę pollerów IPMI bez obaw o przeciążenie kontrolerów BMC. Proces menedżera IPMI jest uruchamiany automatycznie, gdy uruchomiony zostanie co najmniej jeden poller IPMI.

Zobacz także znane problemy dotyczące kontroli IPMI.

Konfiguracja

Konfiguracja hosta

Aby przetwarzać kontrole IPMI, host musi być skonfigurowany. Należy dodać interfejs IPMI z odpowiednim adresem IP i numerem portu oraz zdefiniować parametry uwierzytelniania IPMI.

Więcej informacji można znaleźć w sekcji konfiguracja hostów.

Konfiguracja serwera

Domyślnie serwer Zabbix nie jest skonfigurowany do uruchamiania żadnych procesów odpytywania IPMI, dlatego wszelkie dodane pozycje IPMI nie będą działać. Aby to zmienić, otwórz jako root plik konfiguracyjny serwera Zabbix (zabbix_server.conf) i znajdź następujący wiersz:

```
# StartIPMIPollers=0
```

Odkomentuj go i ustaw liczbę procesów odpytywania na przykład na 3, tak aby wyglądał następująco:

```
StartIPMIPollers=3
```

Zapisz plik, a następnie uruchom ponownie zabbix_server.

Konfiguracja pozycji

Podczas konfigurowania pozycji na poziomie hosta:

- Wybierz „IPMI agent” jako *Typ*
- Wprowadź klucz pozycji, który jest unikalny w obrębie hosta (na przykład `ipmi.fan.rpm`)
- W polu *Interfejs hosta* wybierz odpowiedni interfejs IPMI (adres IP i port). Pamiętaj, że interfejs IPMI musi istnieć na hoście.
- Określ *Czujnik IPMI* (na przykład „FAN MOD 1A RPM” w Dell Poweredge), z którego ma być pobierana metryka. Domyślnie należy podać identyfikator czujnika. Możliwe jest również użycie prefiksów przed wartością:
  - `id:` — aby określić identyfikator czujnika;
  - `name:` — aby określić pełną nazwę czujnika. Może to być przydatne w sytuacjach, gdy czujniki można rozróżnić tylko przez podanie pełnej nazwy.
- Wybierz odpowiedni typ informacji („Numeryczny (zmiennoprzecinkowy)” w tym przypadku; dla czujników dyskretnych — „Numeryczny (bez znaku)”), jednostki (najprawdopodobniej „rpm”) oraz wszelkie inne wymagane atrybuty pozycji

Obsługiwane testy

agent IPMI obsługuje wbudowaną pozycję **ipmi.get**, która zwraca informacje związane z czujnikami IPMI i może być używana do wykrywania czujników IPMI.
 Wartość zwracana: *obiekt JSON*

Timeout i zakończenie sesji

Timeouty komunikatów IPMI oraz liczba ponowień są zdefiniowane w bibliotece OpenIPMI. Ze względu na obecną konstrukcję OpenIPMI nie ma możliwości skonfigurowania tych wartości w Zabbix, ani na poziomie interfejsu, ani pozycja.

Timeout bezczynności sesji IPMI dla LAN wynosi 60 +/-3 sekundy. Obecnie nie jest możliwe zaimplementowanie okresowego wysyłania polecenia Activate Session przy użyciu OpenIPMI. Jeśli z Zabbix do określonego BMC nie są wykonywane żadne kontrole pozycja IPMI przez czas dłuższy niż timeout sesji skonfigurowany w BMC, to następna kontrola IPMI po upływie tego timeoutu zakończy się timeoutem z powodu timeoutów pojedynczych komunikatów, ponowień lub błędu odbioru. Następnie otwierana jest nowa sesja i inicjowane jest pełne ponowne skanowanie BMC. Może zostać otwarty nowy port UDP do zarządzania nową sesją. Bezczynność jest definiowana jako brak zarówno żądań wychodzących, jak i odpowiedzi przychodzących. Jeśli chcesz uniknąć niepotrzebnych ponownych skanowań BMC, zaleca się ustawienie interwału odpytywania pozycja IPMI poniżej timeoutu bezczynności sesji IPMI skonfigurowanego w BMC.

Uwagi dotyczące dyskretnych czujników IPMI

Aby znaleźć czujniki na hoście, uruchom serwer Zabbix z włączonym **DebugLevel=4**. Poczekaj kilka minut i znajdź rekordy wykrywania czujników w pliku dziennika serwera Zabbix:

```
$ 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
```

Aby zdekodować typy i stany czujników IPMI, dostępna jest kopia specyfikacji IPMI 2.0 (należy pamiętać, że nie są planowane dalsze aktualizacje specyfikacji IPMI).

Pierwszym parametrem, od którego należy zacząć, jest ”reading_type”. Użyj ”Table 42-1, Event/Reading Type Code Ranges” ze specyfikacji, aby zdekodować kod ”reading_type”. Większość czujników w naszym przykładzie ma ”reading_type:0x1”, co oznacza czujnik typu ”threshold”. ”Table 42-3, Sensor Type Codes” pokazuje, że ”type:0x1” oznacza czujnik temperatury, ”type:0x2” -

czujnik napięcia, ”type:0x4” - wentylator itd. Czujniki threshold są czasami nazywane czujnikami ”analogowymi”, ponieważ mierzą parametry ciągłe, takie jak temperatura, napięcie czy obroty na minutę.

Inny przykład - czujnik z ”reading_type:0x3”. ”Table 42-1, Event/Reading Type Code Ranges” mówi, że kody typu od 02h do 0Ch oznaczają czujnik ”Generic Discrete”. Czujniki dyskretne mają do 15 możliwych stanów (innymi słowy - do 15 znaczących bitów). Na przykład dla czujnika ‘CATERR’ z ”type:0x7” tabela ”Table 42-3, Sensor Type Codes” pokazuje, że ten typ oznacza ”Processor”, a znaczenie poszczególnych bitów jest następujące: 00h (najmłodszy bit) - IERR, 01h - Thermal Trip itd.

W naszym przykładzie jest kilka czujników z ”reading_type:0x6f”. Dla tych czujników tabela ”Table 42-1, Event/Reading Type Code Ranges” zaleca użycie ”Table 42-3, Sensor Type Codes” do dekodowania znaczenia bitów. Na przykład czujnik ’Power Unit Stat’ ma typ ”type:0x9”, co oznacza ”Power Unit”. Offset 00h oznacza ”PowerOff/Power Down”. Innymi słowy, jeśli najmłodszy bit ma wartość 1, to serwer jest wyłączony. Aby sprawdzić ten bit, można użyć funkcji `bitand` z maską ’1’. Wyrażenie wyzwalacza może wyglądać następująco:

```
bitand(last(/www.example.com/Power Unit Stat,#1),1)=1
```

aby ostrzec o wyłączeniu zasilania serwera.

Uwagi dotyczące nazw czujników dyskretnych w OpenIPMI-2.0.16, 2.0.17, 2.0.18 i 2.0.19

Nazwy czujników dyskretnych w OpenIPMI-2.0.16, 2.0.17 i 2.0.18 często mają na końcu dodane dodatkowe ”0” (lub inną cyfrę albo literę). Na przykład, podczas gdy `ipmitool` i OpenIPMI-2.0.19 wyświetlają nazwy czujników jako ”`PhysicalSecurity`” lub ”`CATERR`”, w OpenIPMI-2.0.16, 2.0.17 i 2.0.18 nazwy te to odpowiednio ”`PhysicalSecurity0`” lub ”`CATERR0`”.

Podczas konfigurowania pozycji IPMI w serwerze Zabbix z użyciem OpenIPMI-2.0.16, 2.0.17 i 2.0.18 należy używać tych nazw zakończonych na ”0” w polu *IPMI sensor* pozycji agenta IPMI. Gdy serwer Zabbix zostanie zaktualizowany do nowej dystrybucji Linuksa, która używa OpenIPMI-2.0.19 (lub nowszej), pozycje z tymi dyskretnymi czujnikami IPMI staną się ”NOT SUPPORTED”. Należy zmienić ich nazwy *IPMI sensor* (usunąć końcowe ’0’) i odczekać pewien czas, aż ponownie staną się ”Enabled”.

Uwagi dotyczące jednoczesnej dostępności czujnika progowego i czujnika dyskretnego

Niektóre agenty IPMI udostępniają zarówno czujnik progowy, jak i czujnik dyskretny pod tą samą nazwą. Zawsze preferowany jest czujnik progowy.

Uwagi dotyczące zakończenia połączenia

Jeśli kontrole IPMI nie są wykonywane (z dowolnego powodu: wszystkie pozycje IPMI hosta są wyłączone/nieobsługiwane, host jest wyłączony/usunięty, host jest w trybie konserwacji itp.), połączenie IPMI zostanie zakończone przez serwer Zabbix lub proxy w ciągu 3 do 4 godzin, w zależności od czasu uruchomienia serwera/proxy Zabbix.

11 SSH agent

Omówienie

Kontrole SSH są wykonywane jako monitorowanie bez użycia agenta. Agent Zabbix nie jest potrzebny do kontroli SSH.

Aby wykonywać kontrole SSH, serwer Zabbix musi być początkowo skonfigurowany z obsługą SSH2 (libssh lub libssh2). Zobacz także: Wymagania.

> **Attention:**
> Począwszy od RHEL 8 obsługiwany jest tylko libssh. W przypadku innych dystrybucji zalecany jest libssh zamiast libssh2.

Konfiguracja

Uwierzytelnianie za pomocą hasła

Kontrole SSH udostępniają dwie metody uwierzytelniania — parę użytkownik/hasło oraz metodę opartą na pliku klucza.

Jeśli nie zamierzasz używać kluczy, nie jest wymagana żadna dodatkowa konfiguracja, poza podlinkowaniem biblioteki libssh lub libssh2 do Zabbix, jeśli kompilujesz ze źródeł.

Uwierzytelnianie za pomocą pliku klucza

Aby używać uwierzytelniania opartego na kluczu dla pozycji SSH, wymagane są pewne zmiany w konfiguracji serwera.

Otwórz plik konfiguracyjny serwera Zabbix (*zabbix_server.conf*) jako `root` i znajdź następującą linię:

```
##### SSHKeyLocation=
```

Usuń znak komentarza i ustaw pełną ścieżkę do folderu, w którym będą znajdować się klucze publiczny i prywatny:

```
SSHKeyLocation=/home/zabbix/.ssh
```

Zapisz plik, a następnie uruchom ponownie serwer Zabbix.

Ścieżka */home/zabbix* oznacza tutaj katalog domowy konta użytkownika *zabbix*, a *.ssh* to katalog, w którym domyślnie będą generowane klucze publiczny i prywatny przez polecenie ssh-keygen wewnątrz katalogu domowego.

Zazwyczaj pakiety instalacyjne serwera Zabbix z różnych dystrybucji systemów operacyjnych tworzą konto użytkownika *zabbix* z katalogiem domowym w innym miejscu, na przykład */var/lib/zabbix* (jak w przypadku kont systemowych).

Przed wygenerowaniem kluczy można przenieść katalog domowy do */home/zabbix*, tak aby odpowiadał parametrowi konfiguracyjnemu `SSHKeyLocation` serwera Zabbix wspomnianemu powyżej.

Note:
Poniższe kroki można pominąć, jeśli konto *zabbix* zostało dodane ręcznie zgodnie z sekcją instalacji. W takim przypadku katalog domowy konta *zabbix* najprawdopodobniej już znajduje się w */home/zabbix*.

Aby zmienić katalog domowy konta użytkownika *zabbix*, należy zatrzymać wszystkie działające procesy, które z niego korzystają:

```
systemctl stop zabbix-agent
systemctl stop zabbix-server
```

Aby zmienić lokalizację katalogu domowego z próbą przeniesienia go (jeśli istnieje), należy wykonać następujące polecenie:

```
usermod -m -d /home/zabbix zabbix
```

Możliwe jest również, że katalog domowy nie istniał w starej lokalizacji, więc należy go utworzyć w nowym miejscu. Bezpieczna próba wykonania tego to:

```
test -d /home/zabbix || mkdir /home/zabbix
```

Aby mieć pewność, że wszystko jest zabezpieczone, można wykonać dodatkowe polecenia, aby ustawić uprawnienia do katalogu domowego:

```
chown zabbix:zabbix /home/zabbix
chmod 700 /home/zabbix
```

Wcześniej zatrzymane procesy można teraz ponownie uruchomić:

```
systemctl start zabbix-agent
systemctl start zabbix-server
```

Teraz można wykonać kroki generowania kluczy publicznego i prywatnego za pomocą następujących poleceń (dla lepszej czytelności monity poleceń zostały zakomentowane):

```
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):
/home/zabbix/.ssh/id_rsa
##### Enter passphrase (empty for no passphrase):
<Leave empty>
##### Enter same passphrase again:
<Leave empty>
##### 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+        |
##### +-----------------+
```

Note:
Klucze publiczny i prywatny (*id_rsa.pub* i *id_rsa*) zostały domyślnie wygenerowane w katalogu */home/zabbix/.ssh*, co odpowiada parametrowi konfiguracyjnemu `SSHKeyLocation` serwera Zabbix.

Attention:
Typy kluczy inne niż ”rsa” mogą być obsługiwane przez narzędzie ssh-keygen i serwery SSH, ale mogą nie być obsługiwane przez libssh2 używany przez Zabbix.

Formularz konfiguracji powłoki

Ten krok należy wykonać tylko raz dla każdego hosta, który będzie monitorowany za pomocą kontroli SSH.

Korzystając z poniższych poleceń, plik klucza **publicznego** można zainstalować na zdalnym hoście *10.10.10.10*, tak aby kontrole SSH mogły być wykonywane przy użyciu konta *root* (dla lepszej czytelności znaki zachęty wiersza poleceń zostały zakomentowane):

```
sudo -u zabbix ssh-copy-id root@10.10.10.10
##### The authenticity of host '10.10.10.10 (10.10.10.10)' can't be established.
##### RSA key fingerprint is 38:ba:f2:a4:b5:d9:8f:52:00:09:f7:1f:75:cc:0b:46.
##### Are you sure you want to continue connecting (yes/no)?
yes
##### Warning: Permanently added '10.10.10.10' (RSA) to the list of known hosts.
##### root@10.10.10.10's password:
<Enter root password>
##### Now try logging into the machine, with "ssh 'root@10.10.10.10'",
##### and check to make sure that only the key(s) you wanted were added.
```

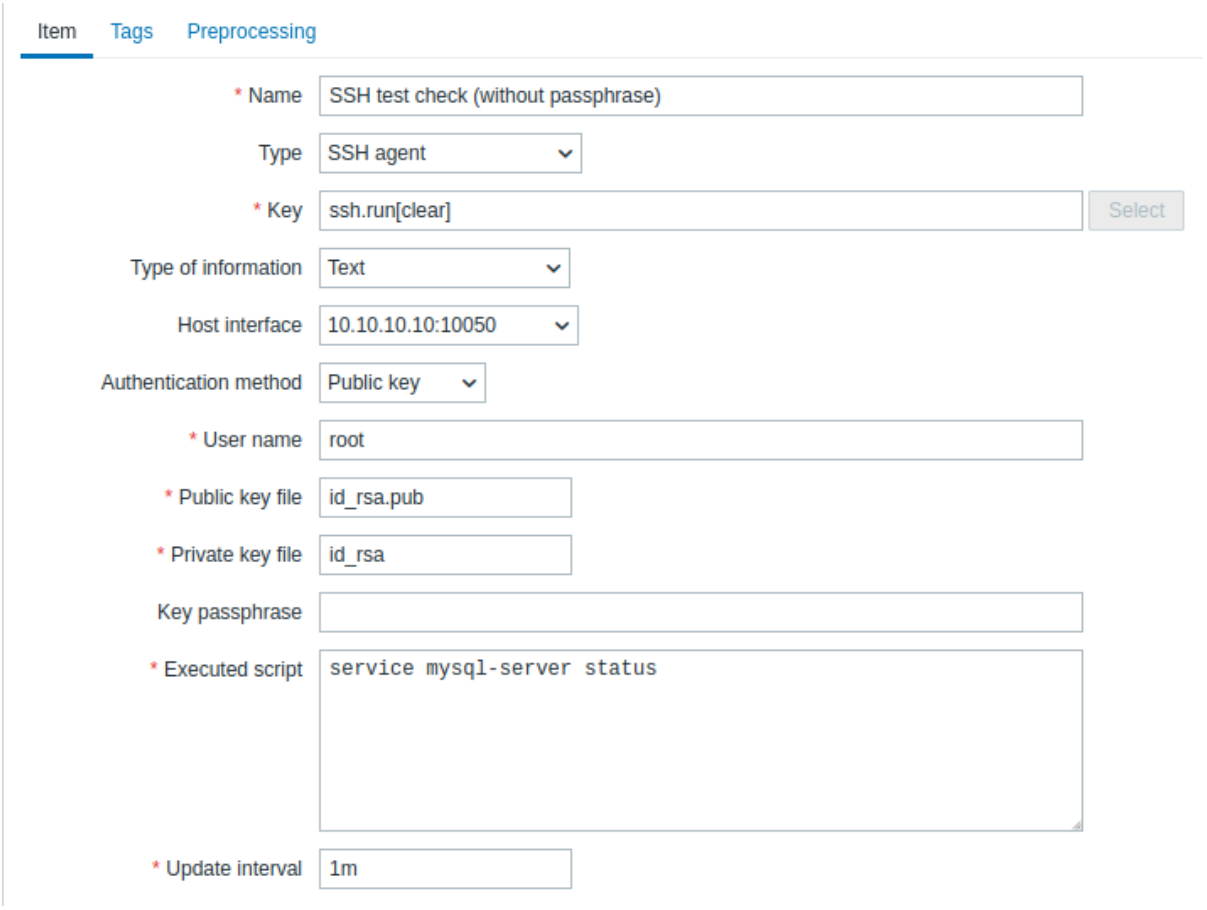
Teraz można sprawdzić logowanie SSH przy użyciu domyślnego klucza prywatnego (*/home/zabbix/.ssh/id_rsa*) dla konta użytkownika *zabbix*:

```
sudo -u zabbix ssh root@10.10.10.10
```

Jeśli logowanie powiedzie się, część konfiguracyjna w powłoce jest zakończona i zdalną sesję SSH można zamknąć.

Konfiguracja pozycji

Rzeczywiste polecenie(a), które mają zostać wykonane, należy umieścić w polu *Executed script* w konfiguracji pozycji. Kilka poleceń można wykonać jedno po drugim, umieszczając je w nowej linii. W takim przypadku zwracane wartości będą również formatowane jako wielowierszowe.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji SSH to:

Parameter	Description	Comments
Type	Wybierz tutaj **SSH agent**.	

Parameter	Description	Comments

Key	Unikalny (dla host) klucz pozycji w formacie **ssh.run[unique short description,<ip>,<port>,<encoding>,<ssh options>,<subsystem>]**	**unique short description** jest wymagany i powinien być unikalny dla każdej pozycji SSH dla host.
		
		Domyślny port to 22, a nie port określony w interfejsie, do którego przypisana jest ta pozycja.
		
		ssh options umożliwiają przekazanie dodatkowych opcji SSH w formacie *key1=value1;key2=value2,value3*. Wiele wartości dla jednego klucza można przekazać, oddzielając je przecinkiem (w takim przypadku parametr musi być cytowany); wiele kluczy opcji można przekazać, oddzielając je średnikiem.
		
		Obsługiwane są następujące klucze opcji: `KexAlgorithms`, `HostkeyAlgorithms`, `Ciphers`, `MACs`, `PubkeyAcceptedKeyTypes`. Obsługa klucza i wartości zależy od biblioteki SSH (na przykład `PubkeyAcceptedKeyTypes` jest obsługiwany tylko w libssh); jeśli opcja nie jest obsługiwana, zostanie zwrócony błąd, a pozycja stanie się nieobsługiwana.
		
		Zwróć uwagę, że znak "+" do dołączania ustawień szyfrów oraz "!" do wyłączania określonych ustawień szyfrów (jak w GnuTLS i OpenSSL) nie są obsługiwane.
		
		Przykłady:
		=>
		`ssh.run[KexAlgorithms,127.0.0.1,,,Ciphers=aes128`
		=>
		`ssh.run[KexAlgorithms,,,,"KexAlgorithms=diffie-h`
		=>
		`ssh.run[PubkeyAcceptedKeyTypes,127.0.0.1,,,Pubke`
		
		subsystem umożliwia przekazanie podsystemu SSH, ograniczając połączenie SSH do określonych operacji dozwolonych przez ten podsystem (np. transferów plików z użyciem SFTP lub zarządzania urządzeniami sieciowymi z użyciem NETCONF). Zwróć uwagę, że użycie podsystemu może również wymagać zastosowania określonej składni skryptu w parametrze *Executed script*.
		
		Przykłady:
		=>
		`ssh.run[SFTPBackup,192.0.2.18,,,,sftp]`
		=>
		`ssh.run[Cisco1234,192.0.2.18,,,,netconf]`

Authentication method	Jedna z opcji "Password" lub "Public key".	
User name	Nazwa użytkownika (do 255 znaków) do uwierzytelniania na zdalnym host. Wymagane.	
Public key file	Nazwa pliku klucza publicznego, jeśli *Authentication method* to "Public key". Wymagane.	Przykład: *id_rsa.pub* - domyślna nazwa pliku klucza publicznego generowana przez polecenie ssh-keygen.
Private key file	Nazwa pliku klucza prywatnego, jeśli *Authentication method* to "Public key". Wymagane.	Przykład: *id_rsa* - domyślna nazwa pliku klucza prywatnego.

Parameter	Description	Comments
Password or *Key passphrase*	Hasło (do 255 znaków) do uwierzytelniania lub hasło **jeśli** było użyte dla klucza prywatnego.	Pozostaw pole *Key passphrase* puste, jeśli hasło nie było użyte. Zobacz także znane problemy dotyczące użycia hasła.

Executed script	Wykonywane polecenie(a) powłoki przy użyciu zdalnej sesji SSH.	Zwracana wartość wykonanego polecenia(a) powłoki jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.
		
		Zwróć uwagę, że biblioteka libssh2 może obcinać wykonywalne skrypty do ~32 kB.
		
		Przykłady:
		`date +%s`
		`systemctl status mysql-server`
		`ps auxww | grep httpd | wc -l`
		
		Przykład (dla podsystemu NETCONF):
		`<rpc>`
		`<get-software-information/>`
		`</rpc>`
		`]]>]]>`
		`<rpc>`
		`<close-session/>`
		`</rpc>`
		`]]>]]>`

12 Telnet agent

Przegląd

Kontrole Telnet są wykonywane jako monitorowanie bez agent. Zabbix agent nie jest wymagany do kontroli Telnet.

Konfigurowalne pola

Rzeczywiste polecenie(a) do wykonania należy umieścić w polu **Executed script** w konfiguracji pozycji.
 Wiele poleceń można wykonać jedno po drugim, umieszczając je w nowej linii. W takim przypadku zwracana wartość również będzie sformatowana jako wielowierszowa.

Obsługiwane znaki, którymi może kończyć się znak zachęty powłoki:

- $
- #
- >
- %

Note:
Linia znaku zachęty telnet zakończona jednym z tych znaków zostanie usunięta ze zwracanej wartości, ale tylko dla pierwszego polecenia na liście poleceń, tj. tylko na początku sesji telnet.

Klucz	Opis
telnet.run[<unique short description>,<ip>,<port>,<encoding>]	Uruchamia polecenie na zdalnym urządzeniu przy użyciu połączenia telnet

Attention:
Jeśli sprawdzenie telnet zwraca wartość zawierającą znaki spoza ASCII i w kodowaniu innym niż UTF8, wówczas parametr *<encoding>* w kluczu powinien zostać odpowiednio określony. Więcej szczegółów można znaleźć na stronie encoding of returned values.

13 JMX agent

Przegląd

Monitorowanie JMX może być używane do monitorowania liczników JMX aplikacji Java.

Monitorowanie JMX ma natywne wsparcie w Zabbix w postaci demona Zabbix o nazwie „Zabbix Java gateway".

Aby pobrać wartość konkretnego licznika JMX na hoście, serwer Zabbix odpytuje **Java gateway** Zabbix, który z kolei używa API zarządzania JMX do zdalnego odpytywania interesującej aplikacji.

Więcej szczegółów oraz informacje o konfiguracji można znaleźć w sekcji Zabbix Java gateway.

Warning:
Komunikacja między Java gateway a monitorowaną aplikacją JMX nie powinna być blokowana przez zaporę sieciową.

Włączanie zdalnego monitorowania JMX dla aplikacji Java

Aplikacja Java nie wymaga instalacji żadnego dodatkowego oprogramowania, ale musi zostać uruchomiona z opcjami wiersza poleceń podanymi poniżej, aby obsługiwać zdalne monitorowanie JMX.

Jako absolutne minimum, jeśli chcesz po prostu rozpocząć monitorowanie prostej aplikacji Java na lokalnym hoście bez wymuszonych zabezpieczeń, uruchom ją z następującymi opcjami:

```
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 /path/to/your/application.jar
```

Spowoduje to, że Java będzie nasłuchiwać przychodzących połączeń JMX na porcie 12345, wyłącznie z lokalnego hosta, oraz nie będzie wymagać uwierzytelniania ani SSL.

Jeśli chcesz zezwolić na połączenia na innym interfejsie, ustaw parametr -Djava.rmi.server.hostname na adres IP tego interfejsu.

Jeśli chcesz bardziej rygorystycznie podejść do kwestii bezpieczeństwa, masz do dyspozycji wiele innych opcji Java. Na przykład kolejny przykład uruchamia aplikację z bardziej wszechstronnym zestawem opcji i udostępnia ją w szerszej sieci, a nie tylko na lokalnym hoście.

```
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 /path/to/your/application.jar
```

Większość (jeśli nie wszystkie) z tych ustawień można określić w pliku `$JRE/lib/management/management.properties` (lub tam, gdzie ten plik znajduje się w twoim systemie).

Zwróć uwagę, że jeśli chcesz używać SSL, musisz zmodyfikować skrypt startup.sh, dodając opcje `-Djavax.net.ssl.*` do Java gateway, aby wiedział, gdzie znaleźć magazyny kluczy i zaufania.

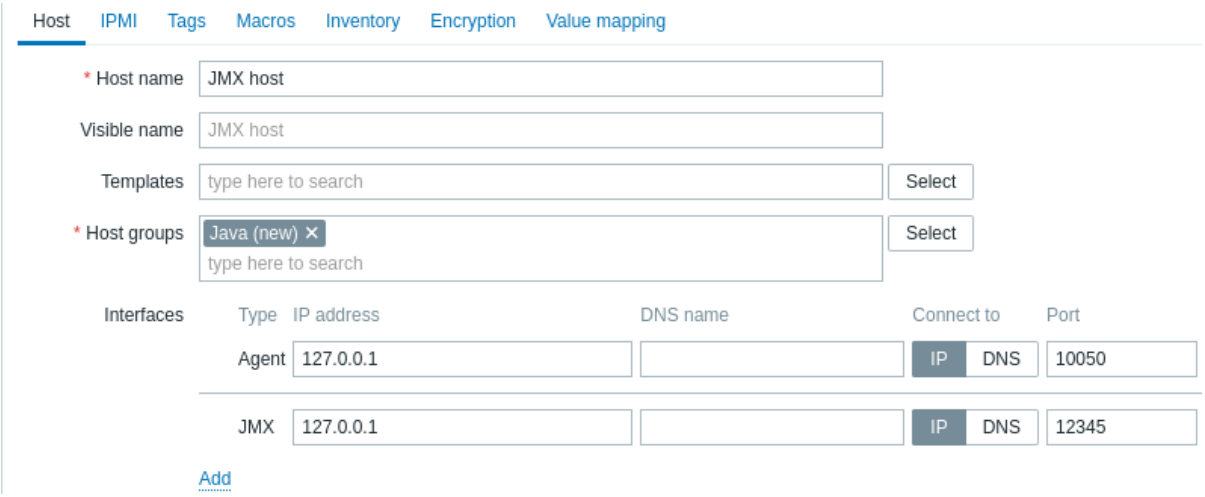
Szczegółowy opis znajduje się w Monitoring and Management Using JMX.

Konfigurowanie interfejsów JMX i pozycji w frontend Zabbix

Gdy Java gateway jest uruchomiony, serwer wie, gdzie go znaleźć, a aplikacja Java została uruchomiona z obsługą zdalnego monitorowania JMX, nadszedł czas, aby skonfigurować interfejsy i pozycje w GUI Zabbix.

Konfigurowanie interfejsu JMX

Zacznij od utworzenia interfejsu typu JMX na interesującym Cię hoście.

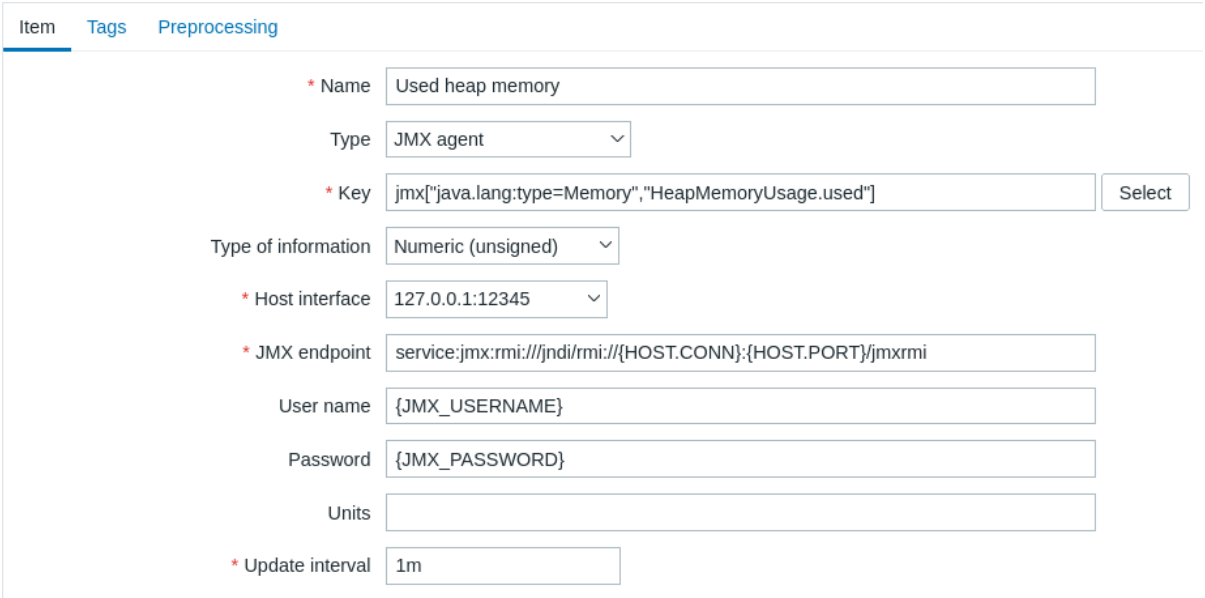


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Dodawanie pozycji agenta JMX

Dla każdego licznika JMX, który chcesz monitorować, dodajesz pozycję **agent JMX** przypisaną do tego interfejsu.

Klucz na zrzucie ekranu poniżej to `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji JMX to:

Type	Ustaw tutaj **agent JMX**.
Key	Klucz pozycji `jmx[]` zawiera trzy parametry: **nazwa obiektu** — nazwa obiektu MBean **nazwa atrybutu** — nazwa atrybutu MBean z opcjonalnymi nazwami pól danych złożonych oddzielonymi kropkami **unikalny krótki opis** — unikalny opis, który pozwala na istnienie wielu pozycji JMX z tą samą nazwą obiektu i nazwą atrybutu na hoście (opcjonalnie) Poniżej znajduje się więcej szczegółów dotyczących kluczy pozycji JMX. Możesz wykrywać MBeany i atrybuty MBean przy użyciu pozycji low-level discovery `jmx.discovery[]`.
JMX endpoint	Możesz określić niestandardowy punkt końcowy JMX. Upewnij się, że parametry połączenia punktu końcowego JMX są zgodne z interfejsem JMX. Można to osiągnąć, używając makr {HOST.*}, tak jak w domyślnym punkcie końcowym JMX. Obsługiwane są makra {HOST.*} oraz makra użytkownika.
User name	Podaj nazwę użytkownika (do 255 znaków), jeśli w aplikacji Java skonfigurowano uwierzytelnianie. Obsługiwane są makra użytkownika.

Password	Podaj hasło (do 255 znaków), jeśli w aplikacji Java skonfigurowano uwierzytelnianie. Obsługiwane są makra użytkownika.

Jeśli chcesz monitorować licznik logiczny, który przyjmuje wartość „true” albo „false”, ustaw typ informacji na „Numeric (unsigned)” i wybierz krok przetwarzania wstępnego „Boolean to decimal” na karcie Preprocessing. Serwer będzie przechowywać wartości logiczne odpowiednio jako 1 lub 0.

Klucze pozycji JMX bardziej szczegółowo

Proste atrybuty

Nazwa obiektu MBean jest po prostu ciągiem znaków, który definiujesz w swojej aplikacji Java. Z kolei nazwa atrybutu może być bardziej złożona. Jeśli atrybut zwraca prymitywny typ danych (liczbę całkowitą, ciąg znaków itp.), nie ma się czym martwić — klucz będzie wyglądał tak:

```
jmx[com.example:Type=Hello,weight]
```

W tym przykładzie nazwa obiektu to „com.example:Type=Hello”, nazwa atrybutu to „weight”, a typ zwracanej wartości powinien prawdopodobnie być ustawiony na „Numeric (float)”.

Atrybuty zwracające dane złożone

Sytuacja staje się bardziej skomplikowana, gdy atrybut zwraca dane złożone. Na przykład: nazwa atrybutu to ”apple” i zwraca on hash reprezentujący jego parametry, takie jak ”weight”, ”color” itp. Klucz może wyglądać następująco:

```
jmx[com.example:Type=Hello,apple.weight]
```

W ten sposób nazwa atrybutu i klucz hash są rozdzielane przy użyciu symbolu kropki. Analogicznie, jeśli atrybut zwraca zagnieżdżone dane złożone, poszczególne części są rozdzielane kropką:

```
jmx[com.example:Type=Hello,fruits.apple.weight]
```

Atrybuty zwracające dane tabelaryczne

Atrybuty danych tabelarycznych składają się z jednego lub wielu atrybutów złożonych. Jeśli taki atrybut zostanie określony w parametrze nazwy atrybutu, wówczas wartość tej pozycji zwróci kompletną strukturę atrybutu w formacie JSON. Wartości poszczególnych elementów wewnątrz atrybutu danych tabelarycznych można pobrać za pomocą preprocessing.

Przykład atrybutu danych tabelarycznych:

```
jmx[com.example:type=Hello,foodinfo]
```

Wartość pozycji:

```
[
  {
    "a": "apple",
    "b": "banana",
    "c": "cherry"
  },
  {
    "a": "potato",
    "b": "lettuce",
    "c": "onion"
  }
]
```

Problem z kropkami

Jak dotąd wszystko dobrze. Ale co, jeśli nazwa atrybutu lub klucz hash zawiera symbol kropki? Oto przykład:

```
jmx[com.example:Type=Hello,all.fruits.apple.weight]
```

To problem. Jak powiedzieć Zabbixowi, że nazwa atrybutu to „all.fruits”, a nie tylko „all”? Jak odróżnić kropkę, która jest częścią nazwy, od kropki, która oddziela nazwę atrybutu od kluczy hash?

Jest to możliwe, wystarczy poprzedzić kropki będące częścią nazwy ukośnikiem odwrotnym:

```
jmx[com.example:Type=Hello,all\.fruits.apple.weight]
```

W ten sam sposób, jeśli klucz hash zawiera kropkę, należy ją poprzedzić znakiem ucieczki:

```
jmx[com.example:Type=Hello,all\.fruits.apple.total\.weight]
```

Inne problemy

Znak ukośnika odwrotnego w nazwie atrybutu powinien być poprzedzony znakiem ucieczki:

```
jmx[com.example:type=Hello,c:\\documents]
```

Aby obsłużyć wszelkie inne znaki specjalne w kluczu pozycji JMX, zobacz sekcję dotyczącą formatu klucza pozycji section.

To właściwie wszystko. Życzymy udanego monitorowania JMX!

Nieprymitywne typy danych

Możliwa jest praca z niestandardowymi MBeanami zwracającymi nieprymitywne typy danych, które przesłaniają metodę **toString()**.

Używanie niestandardowego endpointu z JBoss EAP 6.4

Niestandardowe endpointy umożliwiają pracę z różnymi protokołami transportowymi innymi niż domyślne RMI.

Aby zilustrować tę możliwość, spróbujmy skonfigurować monitorowanie JBoss EAP 6.4 jako przykład. Najpierw przyjmijmy kilka założeń:

- Masz już zainstalowany Zabbix Java gateway. Jeśli nie, możesz to zrobić zgodnie z dokumentacją.
- Zabbix serwer i Java gateway są zainstalowane z prefiksem `/usr/local/`
- JBoss jest już zainstalowany w `/opt/jboss-eap-6.4/` i działa w trybie standalone
- Załóżmy, że wszystkie te komponenty działają na tym samym hoście
- Firewall i SELinux są wyłączone (lub odpowiednio skonfigurowane)

Wprowadźmy kilka prostych ustawień w pliku zabbix_server.conf:

```
JavaGateway=127.0.0.1
StartJavaPollers=5
```

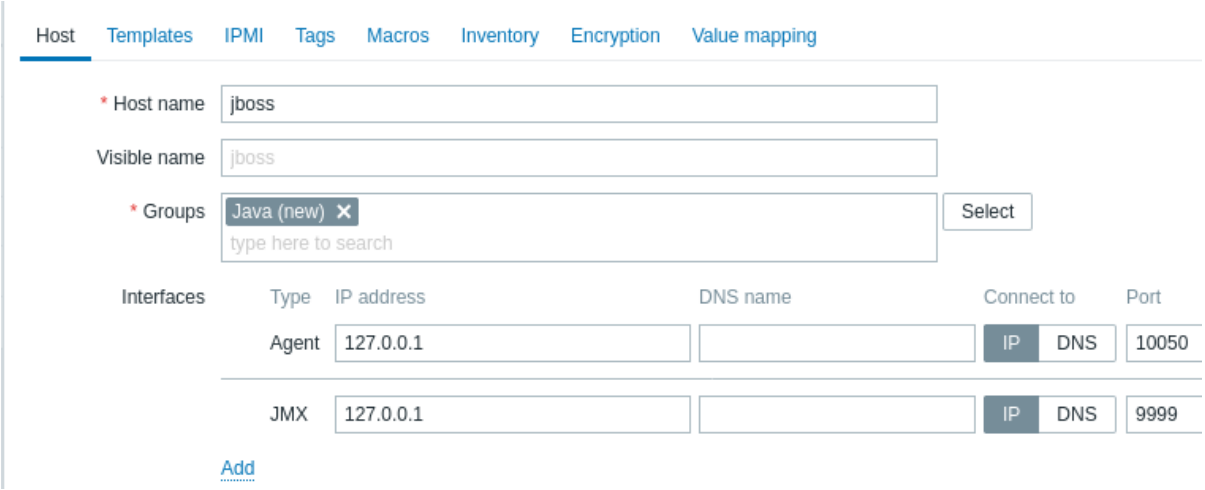
A w pliku konfiguracyjnym `zabbix_java/settings.sh` (lub `zabbix_java_gateway.conf`):

```
START_POLLERS=5
```

Sprawdź, czy JBoss nasłuchuje na swoim standardowym porcie zarządzania:

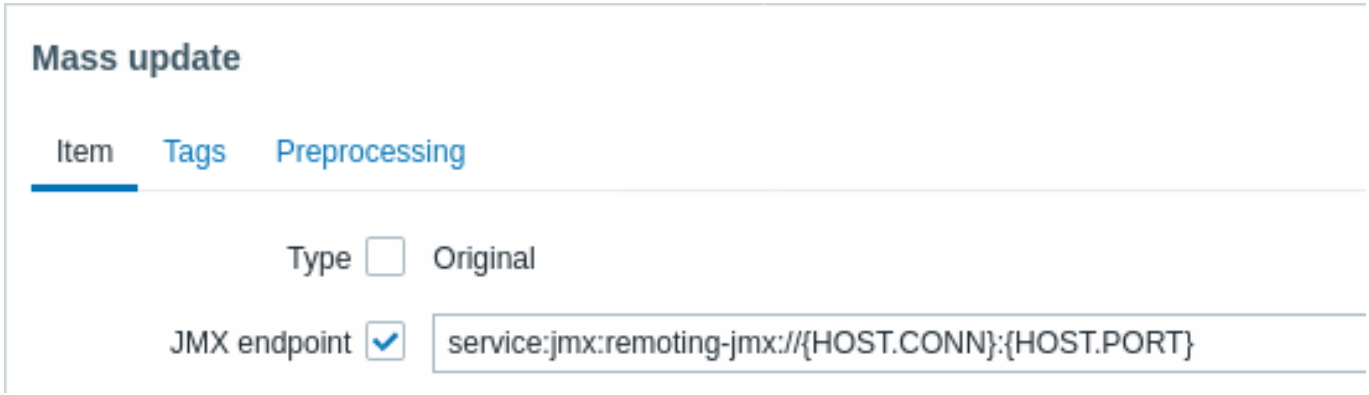
```
$ netstat -natp | grep 9999
tcp        0      0 127.0.0.1:9999          0.0.0.0:*               LISTEN      10148/java
```

Teraz utwórzmy w Zabbix host z interfejsem JMX 127.0.0.1:9999.



Ponieważ wiemy, że ta wersja JBoss używa protokołu JBoss Remoting zamiast RMI, możemy odpowiednio zbiorczo zaktualizować parametr endpointu JMX dla pozycji w naszym szablonie JMX:

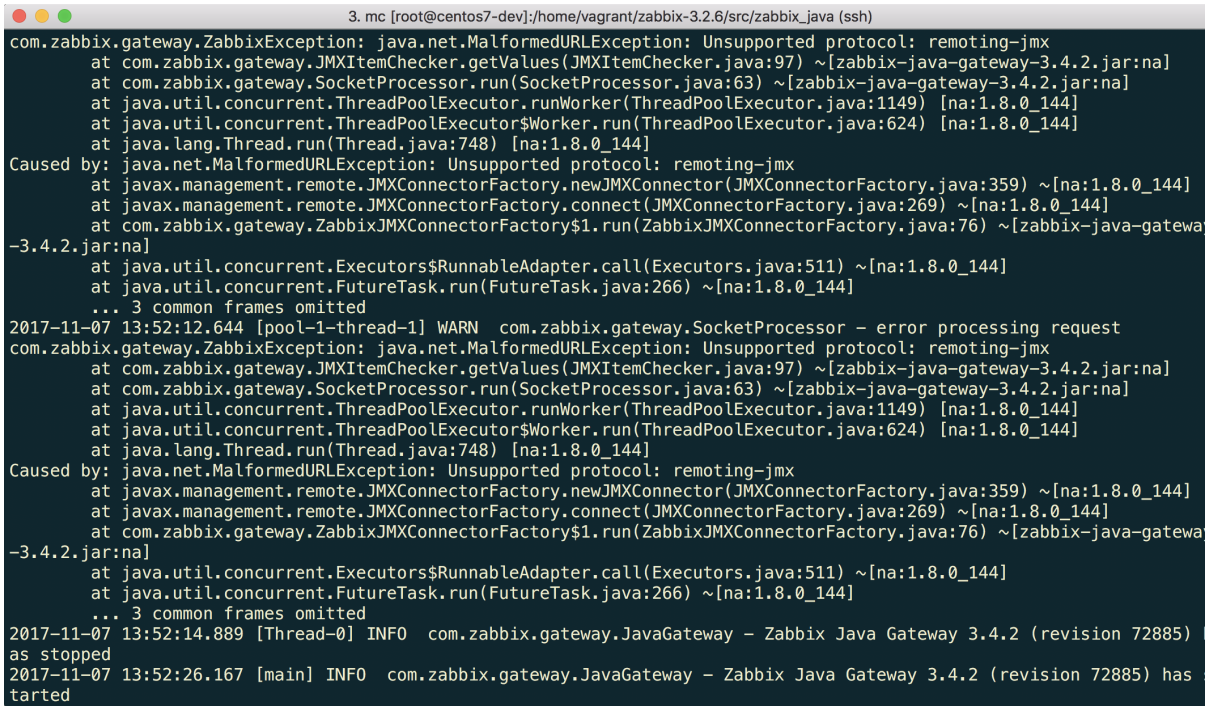
```
service:jmx:remoting-jmx://{HOST.CONN}:{HOST.PORT}
```



Zaktualizujmy pamięć podręczną konfiguracji:

```
/usr/local/sbin/zabbix_server -R config_cache_reload
```

Zwróć uwagę, że na początku możesz napotkać błąd.



„Unsupported protocol: remoting-jmx" oznacza, że Java gateway nie wie, jak pracować z określonym protokołem. Można to naprawić przez utworzenie pliku `~/needed_modules.txt` z następującą zawartością:

```
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
```

a następnie wykonanie polecenia:

```
for i in $(cat ~/needed_modules.txt); do find /opt/jboss-eap-6.4 -iname "${i}*.jar" -exec cp '{}' /usr/loc
```

W ten sposób Java gateway będzie miał wszystkie niezbędne moduły do pracy z jmx-remoting. Pozostaje zrestartować Java gateway, chwilę odczekać i, jeśli wszystko zostało wykonane poprawnie, zobaczyć, że dane monitorowania JMX zaczną napływać do Zabbix (zobacz także: Latest data).

14 Obliczona pozycja

Przegląd

Pozycja obliczana umożliwia utworzenie obliczenia na podstawie wartości niektórych istniejących pozycji. Na przykład możesz chcieć obliczyć średnią godzinową wartości jakiejś pozycji albo obliczyć łączną wartość dla grupy pozycji. Do tego służy pozycja obliczana.

Obliczenia mogą wykorzystywać zarówno:

- pojedyncze wartości poszczególnych pozycji
- złożony filtr do wyboru wielu pozycji do agregacji (szczegóły w sekcji obliczenia agregujące)

Pozycje obliczane są sposobem tworzenia wirtualnych źródeł danych. Wszystkie obliczenia są wykonywane wyłącznie przez serwer Zabbix. Wartości są okresowo obliczane na podstawie użytego wyrażenia arytmetycznego.

Wynikowe dane są przechowywane w bazie danych Zabbix tak samo jak w przypadku każdej innej pozycji; zapisywane są zarówno dane historyczne, jak i trendów, a także można generować wykresy.

Note:
Jeśli wynik obliczenia jest wartością typu float, zostanie on obcięty do liczby całkowitej, jeśli typ informacji pozycji obliczanej to *Numeric (unsigned)*.
Ponadto, jeśli w pamięci podręcznej nie ma świeżych danych i w funkcji nie zdefiniowano okresu zapytania, Zabbix domyślnie sięgnie maksymalnie tydzień wstecz, aby pobrać z bazy danych wartości historyczne.

Pozycje obliczane współdzielą składnię z wyrażeniami wyzwalacza. Porównywanie ze stringami jest dozwolone w pozycjach obliczanych. Pozycje obliczane mogą być odwoływane przez makra lub inne encje tak samo jak każdy inny typ pozycji.

Aby używać pozycji obliczanych, wybierz typ pozycji **Calculated**.

Konfigurowalne pola

key to unikalny identyfikator pozycji (dla hosta). Możesz utworzyć dowolną nazwę key, używając obsługiwanych symboli.

Definicję obliczeń należy wprowadzić w polu **Formula**. Nie ma żadnego powiązania między formułą a key. Parametry key nie są w żaden sposób używane w formule.

Składnia prostej formuły jest następująca:

`function(/host/key,<parameter1>,<parameter2>,...)`

gdzie:

function	Jedna z obsługiwanych funkcji: last, min, max, avg, count itp.
host	Host pozycji używanej do obliczeń. Bieżący host można pominąć (tj. jak w `function(//key,parameter,...)`).
key	Key pozycji używanej do obliczeń. Pozycje zwracające wartości binarnego typu danych nie są obsługiwane.
parameter(s)	Parametry funkcji, jeśli są wymagane. Obsługiwane są sufiksy czasu oraz sufiksy rozmiaru pamięci.

Attention:
Makra użytkownika w formule zostaną rozwinięte, jeśli są używane do odwołania się do parametru funkcji, parametru filtra pozycji lub stałej. Makra użytkownika NIE zostaną rozwinięte, jeśli odwołują się do funkcji, nazwy hosta, key pozycji, parametru key pozycji lub operatora.

Bardziej złożona formuła może używać kombinacji funkcji, operatorów i nawiasów. Możesz używać wszystkich funkcji i operatorów obsługiwanych w wyrażeniach wyzwalaczy. Logika i kolejność operatorów są dokładnie takie same.

W przeciwieństwie do wyrażeń wyzwalaczy, Zabbix przetwarza obliczane pozycje zgodnie z interwałem aktualizacji pozycji, a nie po otrzymaniu nowej wartości.

Wszystkie pozycje, do których odwołują się funkcje historyczne w formule obliczanej pozycji, muszą istnieć i zbierać dane. Ponadto, jeśli zmienisz key odwoływanej pozycji, musisz ręcznie zaktualizować wszystkie formuły używające tego key.

Obliczana pozycja może stać się nieobsługiwana w kilku przypadkach:

- odwoływana pozycja(-e)
  - nie została znaleziona

    - jest wyłączona
    - należy do wyłączonego hosta
    - nie jest obsługiwana (z wyjątkiem funkcji nodata() oraz operatorów z nieznanymi wartościami)
- brak danych do obliczenia funkcji
- dzielenie przez zero
- użyto nieprawidłowej składni

Przykłady użycia

Przykład 1

Obliczanie procentu wolnego miejsca na dysku dla '/'.

Użycie funkcji **last**:

```
100*last(//vfs.fs.size[/,free])/last(//vfs.fs.size[/,total])
```

Zabbix pobierze najnowsze wartości wolnego i całkowitego miejsca na dysku i obliczy procent zgodnie z podanym wzorem.

Przykład 2

Obliczanie 10-minutowej średniej liczby wartości przetworzonych przez Zabbix.

Użycie funkcji **avg**:

```
avg(/Zabbix Server/zabbix[wcache,values],10m)
```

Należy pamiętać, że intensywne używanie obliczanych pozycji z długimi okresami może wpłynąć na wydajność serwera Zabbix.

Przykład 3

Obliczanie całkowitej przepustowości na eth0.

Suma dwóch funkcji:

```
last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes])
```

Przykład 4

Obliczanie procentowego udziału ruchu przychodzącego.

Bardziej złożone wyrażenie:

```
100*last(//net.if.in[eth0,bytes])/(last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes]))
```

Zobacz także: Przykłady obliczeń agregowanych

1 Obliczenia agregatowe

Przegląd

Obliczenia agregujące to typ pozycji obliczanej, który umożliwia zbieranie informacji z kilku pozycji przez serwer Zabbix, a następnie obliczanie wartości agregowanej w zależności od użytej funkcji agregującej.

Obliczenia agregujące nie wymagają uruchomionego agenta na monitorowanym hoście.

Składnia

Aby pobrać agregaty, użyj jednej z obsługiwanych funkcji agregujących: `avg`, `max`, `min`, `sum` itd. Następnie dodaj funkcję **foreach** jako jedyny parametr oraz jej filtr pozycji, aby wybrać wymagane pozycje:

```
aggregate_function(function_foreach(/host/key?[group="host group"],timeperiod))
```

Funkcja **foreach** (np. *avg_foreach*, *count_foreach* itd.) zwraca jedną wartość agregatu dla każdej wybranej pozycji. Pozycje są wybierane za pomocą filtra pozycji (`/host/key?[group="host group"]`), na podstawie historii pozycji. Więcej informacji znajdziesz w sekcji funkcje foreach.

Jeśli niektóre z pozycji nie mają danych dla żądanego okresu, są pomijane w obliczeniach. Jeśli żadna pozycja nie ma danych, funkcja zwróci błąd.

Alternatywnie możesz podać kilka pozycji jako parametry agregacji:

```
aggregate_function(function(/host/key,parameter),function(/host2/key2,parameter),...)
```

Zwróć uwagę, że `function` musi tutaj być funkcją historii/trendu.

> **Note:**
> Jeśli wynik agregacji ma wartość zmiennoprzecinkową, zostanie ona obcięta do liczby całkowitej, jeśli zagregowany typ informacji pozycji to *Numeric (unsigned)*.

Makra użytkownika i makra niskopoziomowego wykrywania są obsługiwane w:

- parametrach klucza pozycji
- parametrach funkcji
- warunkach filtra (nazwa grupy hostów i nazwa tagu)
- stałych wyrażenia

Obliczenie agregatu może stać się nieobsługiwane, jeśli:

- nie zostanie znaleziony żaden z odwołanych elementów (co może się zdarzyć, jeśli klucz pozycji jest nieprawidłowy, żadna z pozycji nie istnieje lub wszystkie uwzględnione grupy są nieprawidłowe)
- brak danych do obliczenia funkcji

Przykłady użycia

Przykłady kluczy dla obliczeń agregowanych.

Przykład 1

Całkowita przestrzeń dyskowa host group 'MySQL Servers'.

```
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"]))
```

Przykład 2

Suma najnowszych wartości wszystkich pozycji pasujących do net.if.in[*] na hoście.

```
sum(last_foreach(/host/net.if.in[*]))
```

Przykład 3

Średnie obciążenie procesora grupy hostów „MySQL Servers".

```
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"]))
```

Przykład 4

5-minutowa średnia liczby zapytań na sekundę dla grupy hostów 'MySQL Servers'.

```
avg(avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m))
```

Przykład 5

Średnie obciążenie CPU na wszystkich hostach w wielu grupach hostów, które mają określone tagi.

```
avg(last_foreach(/*/system.cpu.load?[(group="Servers A" or group="Servers B" or group="Servers C") and (ta
```

Przykład 6

Obliczenie używane dla sum wartości najnowszych pozycji w całej grupie hostów.

```
sum(last_foreach(/*/net.if.out[eth0,bytes]?[group="video"])) / sum(last_foreach(/*/nginx_stat.sh[active]?[
```

Przykład 7

Łączna liczba nieobsługiwanych pozycji w grupie hostów 'Zabbix servers'.

```
sum(last_foreach(/*/zabbix[host,,items_unsupported]?[group="Zabbix servers"]))
```

Przykład 8

Suma najnowszych numerycznych wyników sprawdzeń DNS we wszystkich hostach. Pokazana forma pozycji to `net.dns[192.0.2.1,exampl` jako przykład jednego z możliwych kluczy.

```
sum(last_foreach(/*/net.dns[*,*,*]))
```

Należy pamiętać, że symbole wieloznaczne muszą odpowiadać liczbie parametrów w kluczu (tutaj `net.dns` ma trzy parametry: ip, name, type).

Przykłady poprawnej/niepoprawnej składni

Wyrażenia (w tym wywołania funkcji) nie mogą być używane jako parametry history, trend ani foreach funkcji function. Jednak same te funkcje mogą być używane w innych parametrach funkcji (niehistorycznych).

Expression	Example
Valid	`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)))`

Invalid	`sum(/host/key,10+2)`
	`sum(/host/key, avg(10,2))`
	`sum(/host/key,last(/host/key2))`

Należy pamiętać, że w wyrażeniu takim jak:

```
sum(sum_foreach(//resptime[*],5m))/sum(count_foreach(//resptime[*],5m))
```

nie można zagwarantować, że obie części równania zawsze będą miały ten sam zestaw wartości. Podczas obliczania jednej części wyrażenia może nadejść nowa wartość dla żądanego okresu, a wtedy druga część wyrażenia będzie miała inny zestaw wartości.

15 Pozycja zależna

Przegląd

Istnieją sytuacje, gdy jedna pozycja zbiera jednocześnie wiele metryk lub nawet bardziej sensowne jest jednoczesne zbieranie powiązanych metryk, na przykład:

- wykorzystanie CPU poszczególnych rdzeni
- ruch sieciowy przychodzący/wychodzący/całkowity

Aby umożliwić zbiorcze zbieranie metryk i jednoczesne użycie ich w kilku powiązanych pozycjach, Zabbix obsługuje pozycje zależne. Pozycje zależne zależą od pozycji nadrzędnej, która zbiera ich dane jednocześnie, w jednym zapytaniu. Nowa wartość pozycji nadrzędnej automatycznie wypełnia wartości pozycji zależnych. Pozycje zależne nie mogą mieć innego interwału aktualizacji niż pozycja nadrzędna.

Opcje przetwarzania wstępnego Zabbix można wykorzystać do wyodrębnienia z danych pozycji nadrzędnej części potrzebnej dla pozycji zależnej.

Przetwarzaniem wstępnym zarządza proces `preprocessing manager`, wraz z wątkami roboczymi, które wykonują kroki przetwarzania wstępnego. Wszystkie wartości z przetwarzaniem wstępnym (przed Zabbix 7.4.1, wszystkie wartości), otrzymywane od różnych zbieraczy danych, przechodzą przez `preprocessing manager` przed dodaniem do pamięci podręcznej historii. Do komunikacji IPC opartej na gniazdach używany jest pomiędzy zbieraczami danych (pollers, trappers itp.) a procesem przetwarzania wstępnego.

Serwer Zabbix lub proxy Zabbix (jeśli host jest monitorowany przez proxy) wykonują kroki przetwarzania wstępnego i przetwarzają pozycje zależne.

Pozycje dowolnego typu, nawet pozycje zależne, mogą być ustawione jako pozycje nadrzędne. Dodatkowe poziomy pozycji zależnych można wykorzystać do wyodrębniania mniejszych części z wartości istniejącej pozycji zależnej.

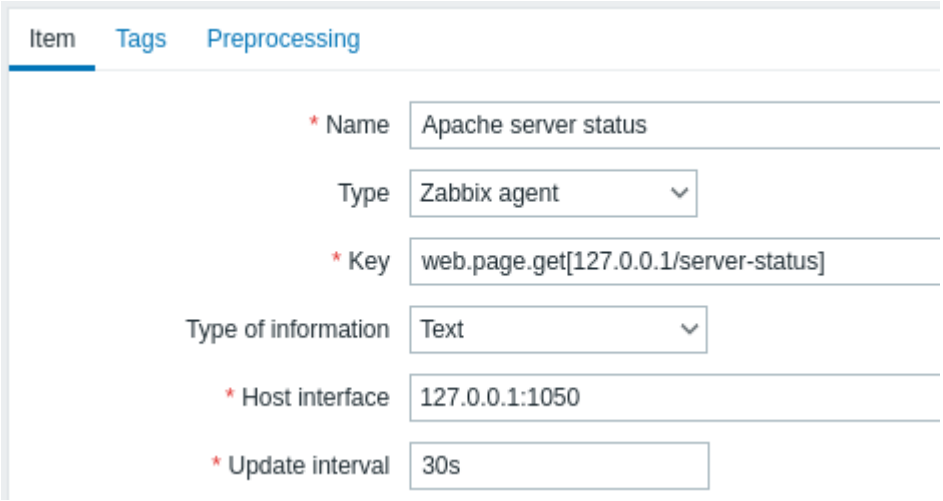
Ograniczenia

- Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu)
- Prototyp pozycji może zależeć od innego prototypu pozycji lub zwykłej pozycji z tego samego hosta
- Zależna pozycja na hoście z główną pozycją z szablonu nie zostanie wyeksportowana do XML

Konfiguracja pozycji

Pozycja zależna zależy od swojej pozycji nadrzędnej pod względem danych. Dlatego najpierw musi zostać skonfigurowana (lub istnieć) **pozycja nadrzędna**:

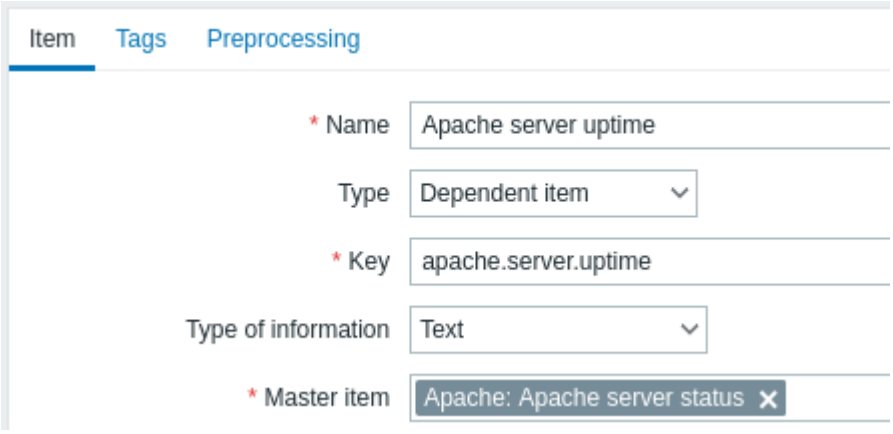
- Przejdź do: *Data collection* → *Hosts*
- Kliknij *Items* w wierszu hosta
- Kliknij *Create item*
- Wprowadź parametry pozycji w formularzu



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Kliknij *Add*, aby zapisać pozycję nadrzędną.

Następnie możesz skonfigurować **pozycję zależną**.

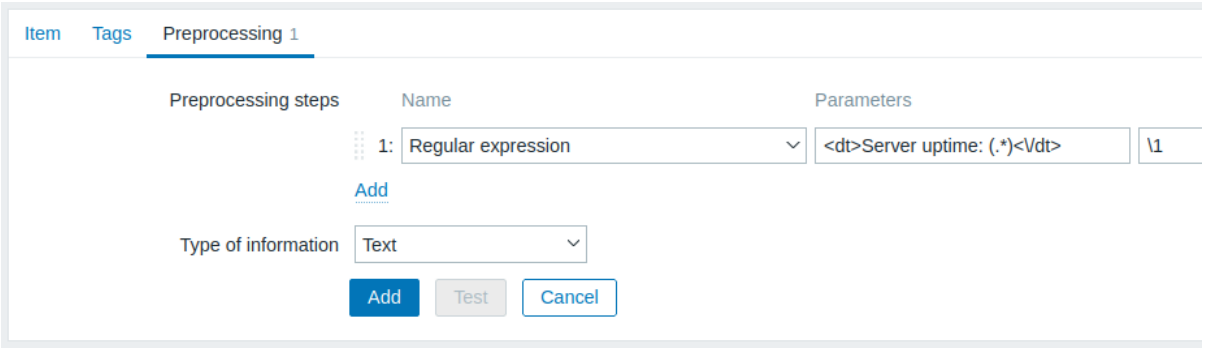


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji zależnych to:

Type	Wybierz tutaj **Dependent item**.
Key	Wprowadź klucz, który będzie używany do rozpoznawania pozycji.
Master item	Wybierz pozycję nadrzędną. Wartość pozycji nadrzędnej zostanie użyta do wypełnienia wartości pozycji zależnej.
Type of information	Wybierz typ informacji, który będzie odpowiadał formatowi danych, które będą przechowywane.

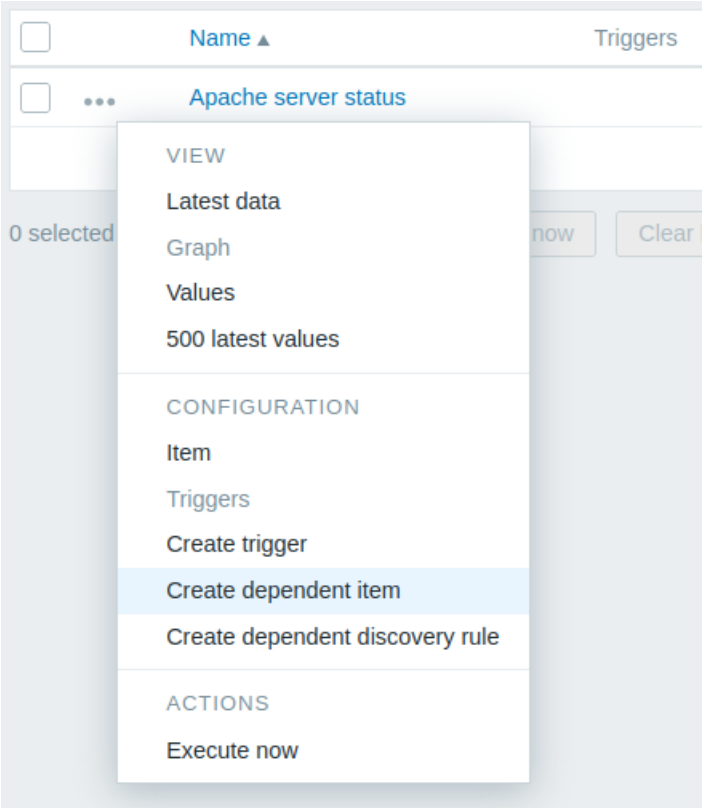
Możesz użyć wstępnego przetwarzania wartości pozycji, aby wyodrębnić wymaganą część wartości pozycji nadrzędnej.



Bez wstępnego przetwarzania wartość pozycji zależnej będzie dokładnie taka sama, jak wartość pozycji nadrzędnej.

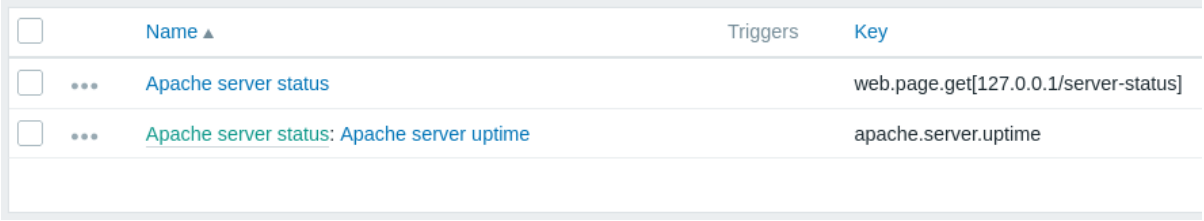
Kliknij *Add*, aby zapisać pozycję zależną.

Skrót umożliwiający szybsze utworzenie pozycji zależnej jest dostępny po kliknięciu przycisku ••• na liście pozycji i wybraniu *Create dependent item*.



Wyświetlanie

Na liście pozycji pozycje zależne są wyświetlane z nazwą nadrzędnej pozycji jako prefiksem.



Jeśli nadrzędna pozycja zostanie usunięta, wszystkie jej pozycje zależne również zostaną usunięte.

16 Pozycja skryptu

Przegląd

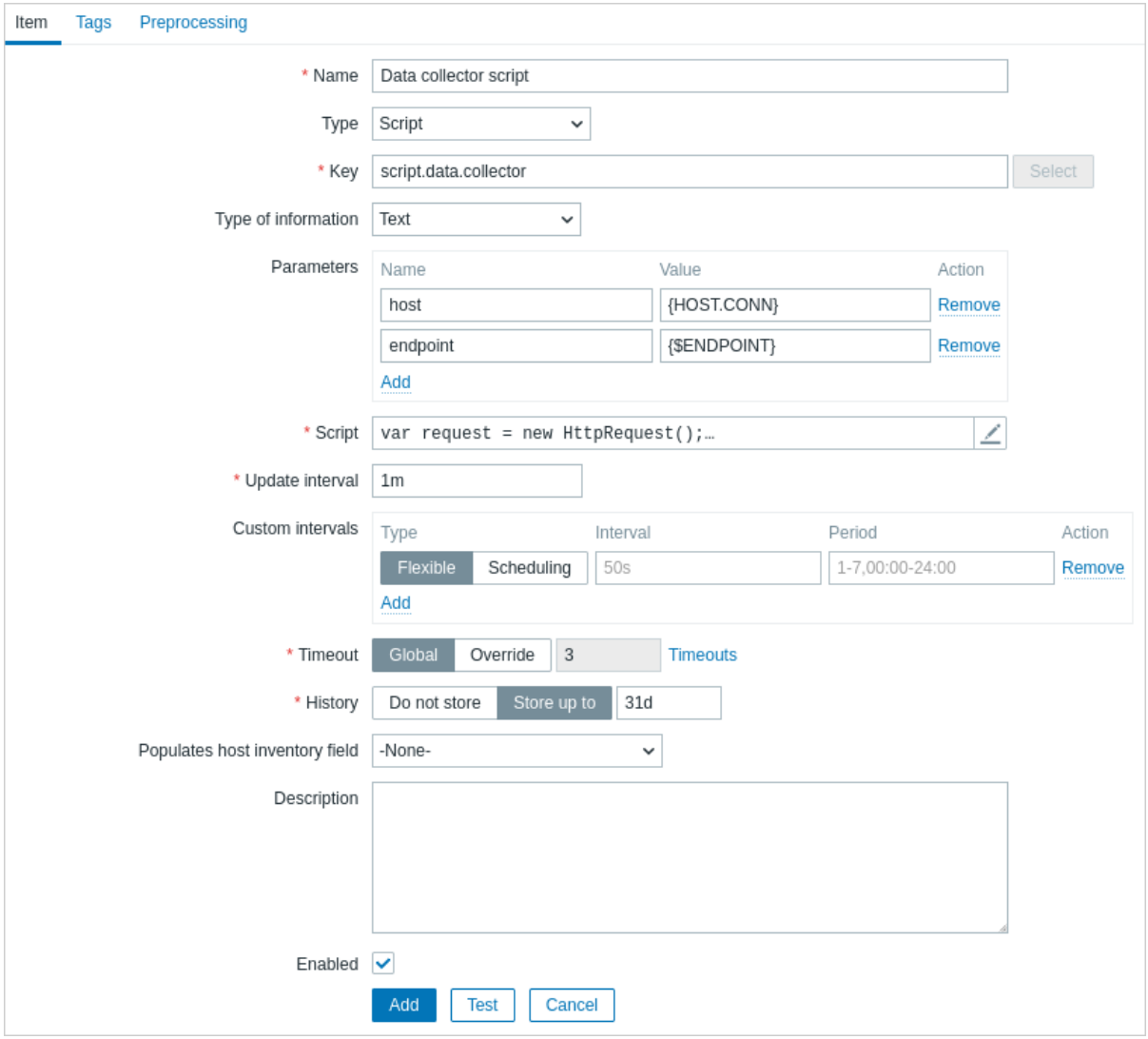
Pozycje skryptowe mogą być używane do zbierania danych przez wykonywanie zdefiniowanego przez użytkownika kodu JavaScript z możliwością pobierania danych przez HTTP/HTTPS. Oprócz skryptu można określić opcjonalną listę parametrów (par nazw i wartości) oraz limit czasu.

Ten typ pozycji może być przydatny w scenariuszach zbierania danych, które wymagają wielu kroków lub złożonej logiki. Na przykład pozycja skryptowa może być skonfigurowana tak, aby wykonać wywołanie HTTP, następnie w pewien sposób przetworzyć dane otrzymane w pierwszym kroku i przekazać przekształconą wartość do drugiego wywołania HTTP.

Pozycje skryptowe są przetwarzane przez pollery serwera lub proxy Zabbix.

Konfiguracja

W polu *Typ* formularza konfiguracji pozycji wybierz Script, a następnie wypełnij wymagane pola.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji typu Script to:

Pole	Opis
Key	Wprowadź unikalny klucz, który będzie używany do identyfikacji pozycji.
Parameters	Określ zmienne przekazywane do skryptu jako pary atrybut-wartość. Obsługiwane są makra użytkownika. Aby sprawdzić, które wbudowane makra są obsługiwane, wyszukaj „Script-type item” w tabeli obsługiwanych makr.
Script	Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Kod ten musi zawierać logikę zwracania wartości metryki. Kod ma dostęp do wszystkich parametrów oraz wszystkich dodatkowych obiektów JavaScript dodanych przez Zabbix. Zobacz także: Przewodnik po JavaScript.
Timeout	Limit czasu wykonywania JavaScript (1-600 s; jego przekroczenie spowoduje zwrócenie błędu). Pamiętaj, że w zależności od skryptu wyzwolenie limitu czasu może potrwać dłużej. Więcej informacji o parametrze *Timeout* znajdziesz w sekcji ogólnych atrybutów pozycji.

Przykłady

Proste zbieranie danych

Pobierz zawartość *https://www.example.com/release_notes*:

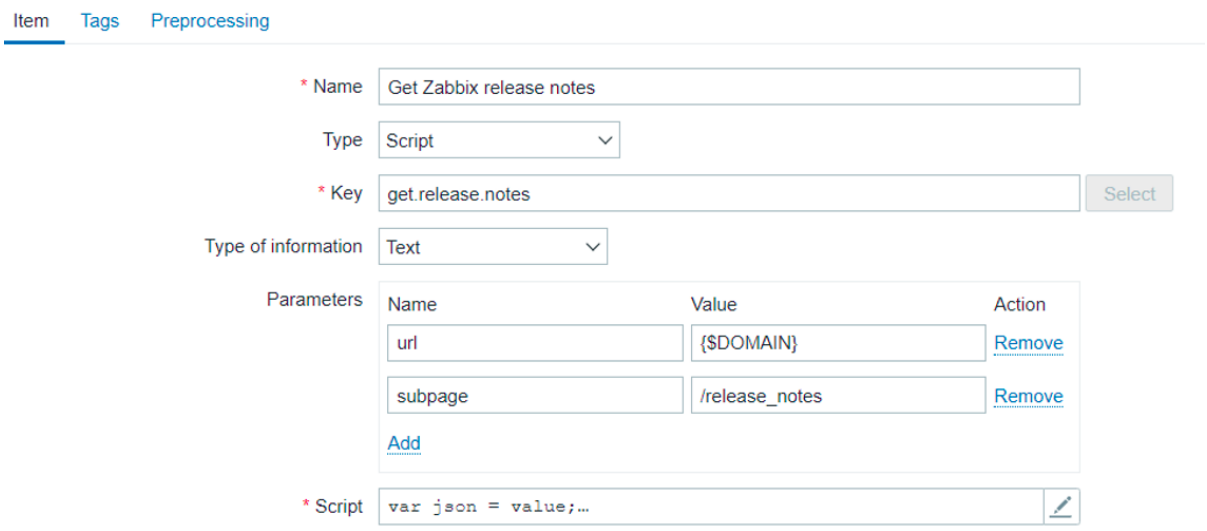
- Utwórz pozycję typu „Script”.
- W polu *Script* wprowadź:

```
var request = new HttpRequest();
return request.get("https://www.example.com/release_notes");
```

Zbieranie danych z użyciem parametrów

Pobierz zawartość określonej strony i wykorzystaj parametry:

- Utwórz pozycję typu ”Script” z dwoma parametrami:
  - **url : {$DOMAIN}** (makro użytkownika {$DOMAIN} powinno być zdefiniowane, najlepiej na poziomie hosta)
  - **subpage : /release_notes**



- W polu *Script* wprowadź:

```
var obj = JSON.parse(value);
var url = obj.url;
var subpage = obj.subpage;
var request = new HttpRequest();
return request.get(url + subpage);
```

Wiele żądań HTTP

Pobierz zawartość zarówno *https://www.example.com*, jak i *https://www.example.com/release_notes*:

- Utwórz pozycję typu „Script”.
- W polu *Script* wprowadź:

```
var request = new HttpRequest();
return request.get("https://www.example.com") + request.get("https://www.example.com/release_notes");
```

Logowanie

Dodaj wpis „Log test” do logu serwera Zabbix i w odpowiedzi odbierz wartość pozycji „1”:

- Utwórz pozycję typu „Script”.
- W polu *Script* wprowadź:

```
Zabbix.log(3, 'Log test');
return 1;
```

17 Pozycja przeglądarki

Przegląd

Pozycje przeglądarkowe umożliwiają monitorowanie złożonych witryn internetowych i aplikacji webowych przy użyciu przeglądarki.

> **Attention:**
> Obsługa pozycji przeglądarkowych jest obecnie eksperymentalna.

Pozycje przeglądarkowe zbierają dane, wykonując zdefiniowany przez użytkownika kod JavaScript i pobierając dane przez HTTP/HTTPS. Ta pozycja może symulować takie działania związane z przeglądarką, jak klikanie, wprowadzanie tekstu, nawigowanie po stronach internetowych oraz inne interakcje użytkownika z witrynami internetowymi lub aplikacjami webowymi.

Oprócz skryptu można określić opcjonalną listę parametrów (pary nazwy i wartości) oraz limit czasu.

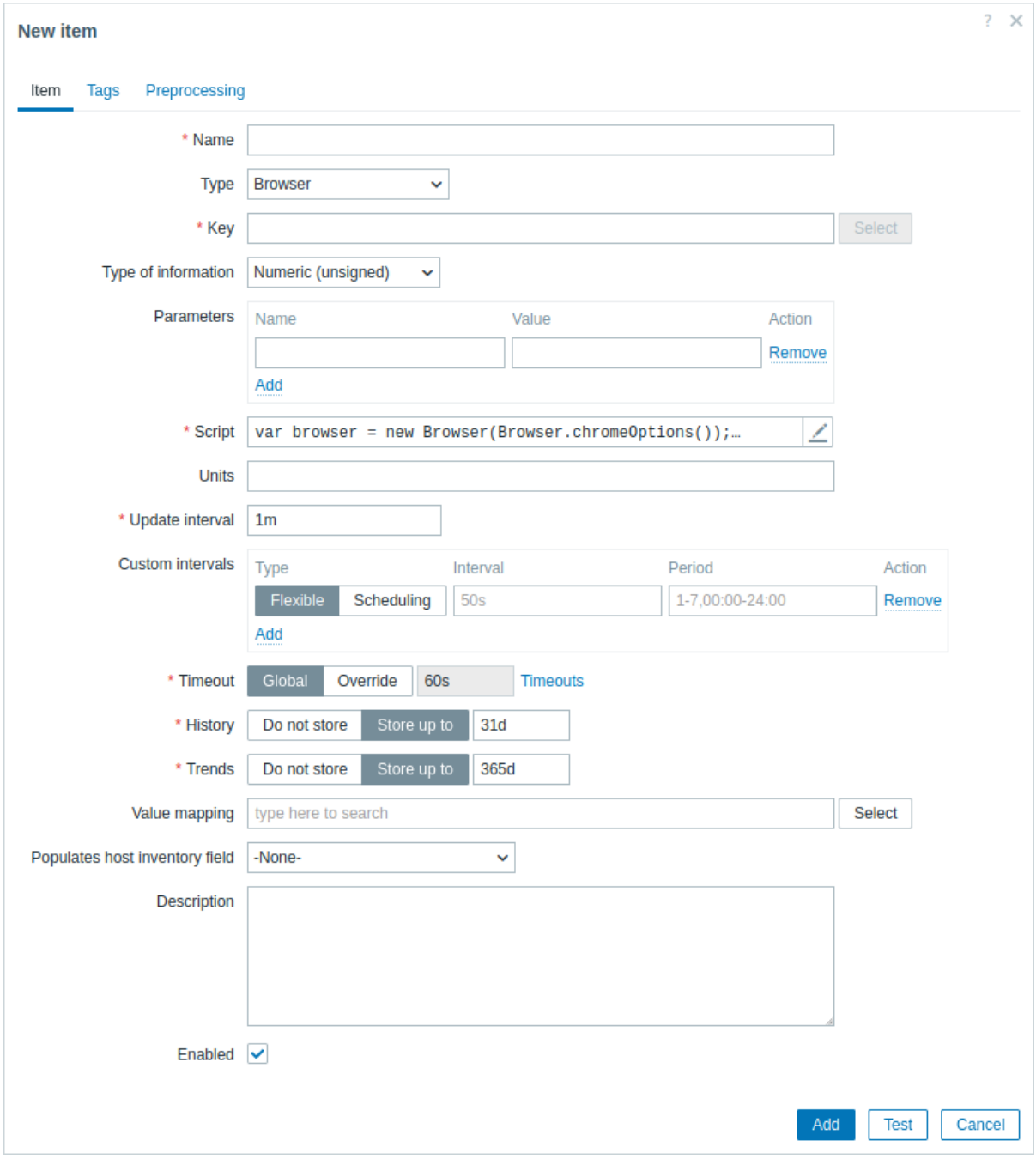
Attention:
Ta pozycja częściowo implementuje standard W3C WebDriver z użyciem Selenium Server lub zwykłego WebDrivera (na przykład ChromeDriver) jako punktu końcowego testów webowych. Aby pozycja działała, ustaw parametr `WebDriverURL` w pliku konfiguracyjnym Zabbix serwer/proxy (jeśli używasz ChromeDriver, zobacz Security Considerations). Aby uzyskać lepszą wydajność, rozważ użycie dedykowanego serwera dla środowiska testów webowych.

Kontrole pozycji przeglądarkowych są wykonywane i przetwarzane przez pollery przeglądarkowe serwera Zabbix lub proxy. W razie potrzeby możesz dostosować liczbę wstępnie uruchomionych instancji pollerów przeglądarkowych w pliku konfiguracyjnym Zabbix serwer/proxy w parametrze `StartBrowserPollers`.

Do monitorowania złożonych witryn internetowych i aplikacji webowych dostępny jest szablon Website by Browser jako szablon dostarczany domyślnie.

Konfiguracja

W polu *Type* formularza konfiguracji pozycji wybierz Browser, a następnie wypełnij wymagane pola.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji Browser to:

Field	Description
Key	Wprowadź unikalny klucz, który będzie używany do identyfikacji pozycji.
Parameters	Określ zmienne, które mają zostać przekazane do skryptu jako pary atrybut-wartość. Obsługiwane są makra użytkownika. Aby sprawdzić, które wbudowane makra są obsługiwane, wyszukaj frazę "Browser-type item" w tabeli obsługiwanych makr.
Script	Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Kod ten musi zawierać logikę zwracania wartości metryki. Kod ma dostęp do wszystkich parametrów, wszystkich dodatkowych obiektów JavaScript oraz obiektów JavaScript pozycji Browser dodanych przez Zabbix. Zobacz także: Przewodnik po JavaScript.
Timeout	Limit czasu wykonania JavaScript (1-600 s; jego przekroczenie spowoduje zwrócenie błędu). Należy pamiętać, że w zależności od skryptu wyzwolenie limitu czasu może potrwać dłużej. Więcej informacji o parametrze *Timeout* można znaleźć w sekcji ogólnych atrybutów pozycji.

Przykłady

Przykład konfiguracji Zabbix do monitorowania stron internetowych przy użyciu szablonu *Website by Browser* można znaleźć w Monitor websites with Browser items.

Domyślny skrypt

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Przechodzi pod określony adres URL.
3. Zbiera wpisy wydajności i statystyki sesji, a następnie zwraca je jako ciąg JSON.

W polu *Skrypt* wprowadź:

```
var browser = new Browser(Browser.chromeOptions());

try {
    browser.navigate("http://example.com");
    browser.collectPerfEntries();
}
finally {
    return JSON.stringify(browser.getResult());
}
```

Inicjalizacja przeglądarki z niestandardowymi capabilities

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki dla dostępnej przeglądarki na podstawie pierwszej pasującej przeglądarki zgodnie z kolejnością określoną w skrypcie.
2. Definiuje capabilities przeglądarki, w tym strategię ładowania strony oraz opcje specyficzne dla każdej przeglądarki, takie jak tryb headless dla przeglądarek Chrome, Firefox i Microsoft Edge.

W polu *Script* wprowadź:

```
var browser = new Browser({
    "capabilities":{
        "firstMatch":[
            {
                "browserName":"chrome",
                "pageLoadStrategy":"normal",
                "goog:chromeOptions":{
                    "args":[
                        "--headless=new"
                    ]
                }
            },
            {
                "browserName":"firefox",
                "pageLoadStrategy":"normal",
                "moz:firefoxOptions":{
                    "args":[
```

```
                    "--headless"
                ]
            }
        },
        {
            "browserName":"MicrosoftEdge",
            "pageLoadStrategy":"normal",
            "ms:edgeOptions":{
                "args":[
                    "--headless=new"
                ]
            }
        },
        {
            "browserName":"safari",
            "pageLoadStrategy":"normal"
        }
    ]
}
});
```

Inicjalizacja przeglądarki z GUI

Domyślnie sesje przeglądarki (z wyjątkiem Safari) są inicjalizowane w trybie headless, co oznacza, że graficzny interfejs użytkownika (GUI) przeglądarki nie jest wyświetlany.

Poniższy skrypt inicjalizuje sesję przeglądarki z włączonym GUI.

Pamiętaj, że jeśli WebDriver nie może zlokalizować pliku binarnego przeglądarki, możesz ręcznie określić ścieżkę.

```
var opts = Browser.chromeOptions();
opts.capabilities.alwaysMatch['goog:chromeOptions'].args = [];

// Aby zainicjalizować sesję Firefox z GUI, odkomentuj poniższe linie:
// var opts = Browser.firefoxOptions();
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].binary = 'usr/bin/firefox';
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].args = [];

// Aby zainicjalizować sesję Microsoft Edge z GUI, odkomentuj poniższe linie:
// var opts = Browser.edgeOptions();
// opts.capabilities.alwaysMatch['ms:edgeOptions'].binary = 'usr/bin/microsoft-edge';
// opts.capabilities.alwaysMatch['ms:edgeOptions'].args = [];

var browser = new Browser(opts);
```

> **Note:**
> Jeśli testy są uruchamiane na zdalnym serwerze lub w kontenerze, możesz użyć klienta Virtual Network Computing (VNC), aby połączyć się z serwerem VNC tej maszyny. Pozwala to zdalnie wyświetlać GUI przeglądarki i wchodzić z nim w interakcję.

Wykonywanie zrzutów ekranu

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Ustawia rozmiar obszaru roboczego przeglądarki w celu określenia rozmiaru zrzutu ekranu (określonego jako parametry, patrz poniżej).
3. Przechodzi do adresu URL (określonego jako parametr, patrz poniżej).
4. Zbiera statystyki sesji, wykonuje zrzut ekranu i dodaje go do zebranych statystyk.
5. Obsługuje błędy, przechwytując komunikaty o błędach i zrzut ekranu.
6. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- webURL - http://example.com
- width - 1920
- height - 1080

W polu *Script* wprowadź:

```
var browser, result;

var browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Parse the JSON string containing parameters passed from Zabbix.

    browser.setScreenSize(Number(params.width), Number(params.height))

    browser.navigate(params.webURL);

    result = browser.getResult();
    result.screenshot = browser.getScreenshot();
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

Sprawdzenie logowania do Zabbix

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Przechodzi do strony (określonej jako parametr, patrz poniżej).
3. Wprowadza nazwę użytkownika i hasło (określone jako parametry, patrz poniżej).
4. Wyszukuje i klika przycisk logowania.
5. Wyszukuje i klika przycisk wylogowania.
6. Zbiera dane wydajności przed logowaniem i po nim, a także po wylogowaniu.
7. Obsługuje błędy przez przechwytywanie komunikatów o błędach i zrzutu ekranu.
8. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- webURL - http://{HOST.CONN}/index.php
- username - {$USERNAME}
- password - {$PASSWORD}

W polu *Script* wprowadź:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Parse the JSON string containing parameters passed from Zabbix.

    browser.navigate(params.webURL);
    browser.collectPerfEntries("open page");

    var el = browser.findElement("xpath", "//input[@id='name']");
    if (el === null) {
        throw Error("cannot find name input field");
    }
    el.sendKeys(params.username);

    el = browser.findElement("xpath", "//input[@id='password']");
    if (el === null) {
        throw Error("cannot find password input field");
```

```
    }
    el.sendKeys(params.password);

    el = browser.findElement("xpath", "//button[@id='enter']");
    if (el === null) {
        throw Error("cannot find login button");
    }
    el.click();

    browser.collectPerfEntries("login");

    el = browser.findElement("link text", "Sign out");
    if (el === null) {
        throw Error("cannot find logout button");
    }
    el.click();

    browser.collectPerfEntries("logout");

    result = browser.getResult();
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

Znajdowanie linków

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Definiuje funkcję usuwania zduplikowanych elementów z tablicy (zobacz krok 5).
3. Przechodzi do strony (określonej przez parametry, patrz poniżej).
4. Znajduje linki na stronie.
5. Usuwa zduplikowane linki, aby zapewnić ich unikalność.
6. Wyodrębnia tylko linki zaczynające się od ”http”.
7. Formatuje wyodrębnione linki do określonej struktury.
8. Obsługuje błędy przez przechwytywanie komunikatów o błędach i zrzutu ekranu.
9. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- scheme - {$WEBSITE.SCHEME}
- domain - {$WEBSITE.DOMAIN}
- path - {$WEBSITE.PATH}

W polu *Script* wprowadź:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value);  // Parse the JSON string containing parameters passed from Zabbix.

    function uniq(a) {
        return a.sort().filter(function (item, pos, ary) {
            return !pos || item != ary[pos - 1];
        });
    }
```

```
    browser.navigate(params.scheme + '://' + params.domain + params.path);

    var el = browser.findElements("link text", "");
    var links = [];
    for (var n = 0; n < el.length; n++) {
        links.push(el[n].getAttribute('href'));
    }

    links = uniq(links);

    result = [];
    for (i = 0; i < links.length; i++) {
        if (links[i].match(/^http.*/)) {
            var row = {};
            row["{#URL}"] = links[i];
            result.push(row);
        }
    }
}
catch (err) {
    if (!(err instanceof BrowserError)) {
        browser.setError(err.message);
    }
    result = browser.getResult();
    result.error.screenshot = browser.getScreenshot();
}
finally {
    return JSON.stringify(result);
}
```

4 Historia i trendy

Przegląd

Historia i trendy to dwa sposoby przechowywania zebranych danych w Zabbix.

Podczas gdy historia przechowuje każdą zebraną wartość, trendy przechowują uśrednione informacje w ujęciu godzinowym i dlatego zużywają mniej zasobów.

Przechowywanie historii

Możesz ustawić, przez ile dni historia będzie przechowywana:

- we formularzu właściwości pozycji
- podczas masowej aktualizacji pozycji
- podczas konfigurowania zadań housekeepera

Wszelkie starsze dane zostaną usunięte przez housekeeper.

Ogólna, zdecydowana rekomendacja jest taka, aby przechowywać historię przez możliwie najmniejszą liczbę dni i w ten sposób nie przeciążać bazy danych dużą liczbą wartości historycznych.

Zamiast przechowywać długą historię, możesz dłużej przechowywać dane trendów. Na przykład możesz przechowywać historię przez 14 dni, a trendy przez 5 lat.

Możesz uzyskać dobre wyobrażenie o tym, ile miejsca wymagają dane historii w porównaniu z danymi trendów, odwołując się do strony dotyczącej rozmiaru bazy danych.

Przy krótszym okresie przechowywania historii nadal będzie można przeglądać starsze dane na wykresach, ponieważ do wyświetlania starszych danych wykresy będą używać wartości trendów.

> **Attention:**
> Jeśli historia jest ustawiona na '0', pozycja będzie aktualizować tylko pozycje zależne i inwentarz. Żadne funkcje wyzwalaczy nie będą obliczane, ponieważ obliczanie wyzwalaczy opiera się wyłącznie na danych historii.

Note:
Jako alternatywny sposób zachowania historii rozważ użycie funkcji eksportu historii modułów ładowalnych.

Przechowywanie trendów

Trendy to wbudowany mechanizm redukcji danych historycznych, który przechowuje minimalną, maksymalną, średnią oraz całkowitą liczbę wartości dla każdej godziny dla numerycznych typów danych.

Możesz ustawić, przez ile dni trendy będą przechowywane:

- we formularzu właściwości pozycji
- podczas masowej aktualizacji pozycji
- podczas konfigurowania zadań Housekeepera

Trendy zwykle mogą być przechowywane znacznie dłużej niż historia. Wszelkie starsze dane zostaną usunięte przez housekeeper.

Serwer Zabbix gromadzi dane trendów w czasie działania w pamięci podręcznej trendów, w miarę napływu danych. Serwer zapisuje trendy z **poprzedniej godziny** każdej pozycji do bazy danych (gdzie frontend może je znaleźć) w następujących sytuacjach:

- serwer otrzymuje pierwszą wartość pozycji z bieżącej godziny
- do końca bieżącej godziny pozostało 5 minut lub mniej, a nadal nie ma żadnych wartości pozycji z bieżącej godziny
- serwer zatrzymuje się

Aby zobaczyć trendy na wykresie, należy poczekać co najmniej do początku następnej godziny (jeśli pozycja jest aktualizowana często) i co najwyżej do końca następnej godziny (jeśli pozycja jest aktualizowana rzadko), czyli maksymalnie 2 godziny.

Gdy serwer zapisuje pamięć podręczną trendów i w bazie danych istnieją już trendy dla tej godziny (na przykład serwer został uruchomiony ponownie w połowie godziny), serwer musi używać instrukcji aktualizacji zamiast prostych instrukcji wstawiania. Dlatego w większych instalacjach, jeśli wymagane jest ponowne uruchomienie, zaleca się zatrzymanie serwera pod koniec jednej godziny i uruchomienie go na początku następnej godziny, aby uniknąć nakładania się danych trendów.

Tabele historii w żaden sposób nie uczestniczą w generowaniu trendów.

Attention:
Jeśli dla trendów ustawiono wartość „0”, serwer Zabbix w ogóle nie oblicza ani nie przechowuje trendów.

Note:
Trendy są obliczane i przechowywane z użyciem tego samego typu danych co oryginalne wartości. W rezultacie obliczenia wartości średniej dla wartości typu unsigned są zaokrąglane, a im mniejszy jest przedział wartości, tym mniej precyzyjny będzie wynik. Na przykład, jeśli pozycja ma wartości 0 i 1, wartość średnia wyniesie 0, a nie 0,5.
Ponadto ponowne uruchomienie serwera może spowodować utratę precyzji obliczeń wartości średniej dla typu unsigned w bieżącej godzinie.

5 Parametry użytkownika

Przegląd

Czasami możesz chcieć uruchomić sprawdzenie agenta, które nie jest wstępnie zdefiniowane w Zabbix. W takich przypadkach pomocne są parametry użytkownika.

Możesz napisać polecenie, które pobiera potrzebne dane, i uwzględnić je w parametrze użytkownika w pliku konfiguracji agenta (parametr konfiguracji 'UserParameter').

Parametr użytkownika ma następującą składnię:

```
UserParameter=<key>,<command>
```

Jak widać, parametr użytkownika zawiera również klucz. Klucz będzie potrzebny podczas konfigurowania pozycji. Wprowadź wybrany przez siebie klucz, do którego łatwo będzie się odwołać (musi być unikalny w obrębie hosta).

Uruchom ponownie agent lub użyj opcji kontroli w czasie wykonywania agenta, aby wczytać nowy parametr, np.:

```
zabbix_agentd -R userparameter_reload
```

Następnie podczas konfigurowania pozycji wprowadź klucz, aby odwołać się do polecenia z parametru użytkownika, które ma zostać wykonane.

Parametry użytkownika to polecenia wykonywane przez agent Zabbix. Należy pamiętać, że przed etapami wstępnego przetwarzania wartości pozycji może zostać zwrócone do 16 MB danych.

W systemach operacyjnych UNIX jako interpreter wiersza poleceń używany jest **/bin/sh**. Parametry użytkownika podlegają limitowi czasu sprawdzenia agenta; jeśli limit czasu zostanie osiągnięty, rozwidlony proces parametru użytkownika zostanie zakończony.

Zobacz także:

- Samouczek krok po kroku dotyczący korzystania z parametrów użytkownika
- Wykonywanie poleceń

Przykłady prostych parametrów użytkownika

Proste polecenie:

```
UserParameter=ping,echo 1
```

agent będzie zawsze zwracał '1' dla pozycji z kluczem 'ping'.

Bardziej złożony przykład:

```
UserParameter=mysql.ping,mysqladmin -uroot ping | grep -c alive
```

agent zwróci '1', jeśli serwer MySQL działa, w przeciwnym razie '0'.

Elastyczne parametry użytkownika

Elastyczne parametry użytkownika akceptują parametry wraz z kluczem. Dzięki temu elastyczny parametr użytkownika może stanowić podstawę do utworzenia kilku pozycji.

Elastyczne parametry użytkownika mają następującą składnię:

```
UserParameter=key[*],command
```

Parametr	Opis
Key	Unikalny klucz pozycji. [*] określa, że ten klucz akceptuje parametry w nawiasach kwadratowych. Parametry są podawane podczas konfigurowania pozycji.
Command	Polecenie, które ma zostać wykonane w celu obliczenia wartości klucza. *Tylko dla elastycznych parametrów użytkownika*: W poleceniu można używać odwołań pozycyjnych $1...$9, aby odwołać się do odpowiedniego parametru w kluczu pozycji. Zabbix analizuje parametry ujęte w [ ] klucza pozycji i odpowiednio podstawia $1,...,$9 w poleceniu. $0 zostanie zastąpione oryginalnym poleceniem (przed rozwinięciem $0,...,$9), które ma zostać uruchomione. Odwołania pozycyjne są interpretowane niezależnie od tego, czy są ujęte w podwójne (") czy pojedyncze (') cudzysłowy. Aby użyć odwołań pozycyjnych bez zmian, należy podać podwójny znak dolara — na przykład awk '{print $$2}'. W takim przypadku $$2 podczas wykonywania polecenia faktycznie zostanie zamienione na $2.

Attention:
Odwołania pozycyjne ze znakiem $ są wyszukiwane i zastępowane przez Zabbix agent tylko dla elastycznych parametrów użytkownika. Dla prostych parametrów użytkownika takie przetwarzanie odwołań jest pomijane i dlatego żadne cytowanie znaku $ nie jest konieczne.

Attention:
Niektóre znaki domyślnie nie są dozwolone w parametrach użytkownika. Pełną listę znaków można znaleźć w UnsafeUserParameters.

Przykład 1

Coś bardzo prostego:

```
UserParameter=ping[*],echo $1
```

Możemy zdefiniować nieograniczoną liczbę pozycji do monitorowania, wszystkie w formacie ping[coś].

- ping[0] - zawsze zwróci '0'
- ping[aaa] - zawsze zwróci 'aaa'

Przykład 2

Dodajmy więcej sensu!

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping | grep -c alive
```

Ten parametr może być używany do monitorowania dostępności bazy danych MySQL. Możemy przekazać nazwę użytkownika i hasło:

```
mysql.ping[zabbix,our_password]
```

Przykład 3

Ile linii w pliku pasuje do wyrażenia regularnego?

```
UserParameter=wc[*],grep -c "$2" $1
```

Ten parametr może być używany do obliczania liczby linii w pliku.

```
wc[/etc/passwd,root]
wc[/etc/services,zabbix]
```

Wynik polecenia

Wartość zwracana przez polecenie to standardowe wyjście wraz ze standardowym wyjściem błędów wygenerowanym przez polecenie.

Attention:
Pozycja, która zwraca tekst (informacje typu character, log lub text), nie stanie się nieobsługiwana w przypadku standardowego wyjścia błędów.

Wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.

Parametry użytkownika, które zwracają tekst (informacje typu character, log lub text), mogą również zwracać biały znak. W przypadku nieprawidłowego wyniku pozycja stanie się nieobsługiwana.

1 Rozszerzanie agentów Zabbix

Ten samouczek zawiera instrukcje krok po kroku dotyczące rozszerzania funkcjonalności agenta Zabbix przy użyciu parametru użytkownika.

Krok 1

Napisz skrypt lub polecenie wiersza poleceń, aby pobrać wymagany parametr.

Na przykład możemy użyć następującego polecenia, aby uzyskać całkowitą liczbę zapytań wykonanych przez serwer MySQL:

```
mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

Po wykonaniu polecenie zwraca całkowitą liczbę zapytań SQL.

Krok 2

Dodaj polecenie do zabbix_agentd.conf:

```
UserParameter=mysql.questions,mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

mysql.questions jest unikalnym identyfikatorem. Może to być dowolny prawidłowy identyfikator klucza, na przykład *queries*.

Przetestuj ten parametr, używając agenta Zabbix z flagą "-t" (jeśli jest uruchamiany jako root, pamiętaj jednak, że agent może mieć inne uprawnienia po uruchomieniu jako demon):

```
zabbix_agentd -t mysql.questions
```

Krok 3

Przeładuj parametry użytkownika z pliku konfiguracyjnego, uruchamiając:

```
zabbix_agentd -R userparameter_reload
```

Możesz także ponownie uruchomić agent zamiast używać polecenia runtime control.

Przetestuj parametr za pomocą narzędzia zabbix_get.

Krok 4

Dodaj nową pozycję z kluczem Key=mysql.questions do monitorowanego hosta. Typ pozycji musi być ustawiony na Zabbix Agent lub Zabbix Agent (active).

Pamiętaj, że typ zwracanych wartości musi być poprawnie ustawiony na serwerze Zabbix. W przeciwnym razie Zabbix ich nie zaakceptuje.

6 Liczniki wydajności systemu Windows

Przegląd

Możesz skutecznie monitorować liczniki wydajności systemu Windows za pomocą klucza *perf_counter[]*.

Na przykład:

```
perf_counter["\Processor(0)\Interrupts/sec"]
```

lub

```
perf_counter["\Processor(0)\Interrupts/sec", 10]
```

Aby uzyskać więcej informacji o użyciu tego klucza lub jego odpowiednika dostępnego wyłącznie w języku angielskim, `perf_counter_en`, zobacz klucze pozycji specyficzne dla systemu Windows.

Aby uzyskać pełną listę liczników wydajności dostępnych do monitorowania, możesz uruchomić:

```
typeperf -qx
```

Możesz również użyć wykrywania niskiego poziomu do wykrywania wielu instancji obiektów liczników wydajności systemu Windows i zautomatyzować tworzenie pozycji perf_counter dla obiektów z wieloma instancjami.

Reprezentacja numeryczna

Windows utrzymuje numeryczne reprezentacje (indeksy) nazw obiektów i liczników wydajności. Zabbix obsługuje te reprezentacje numeryczne jako parametry w kluczach pozycji `perf_counter`, `perf_counter_en` oraz w parametrach konfiguracyjnych `PerfCounter`, `PerfCounterEn`.

Nie zaleca się jednak ich używania, chyba że możesz zagwarantować, że Twoje indeksy numeryczne są mapowane na poprawne ciągi znaków na określonych hostach. Jeśli chcesz utworzyć przenośne pozycje, które działają na różnych hostach z różnymi zlokalizowanymi wersjami Windows, możesz użyć klucza `perf_counter_en` lub parametru konfiguracyjnego `PerfCounterEn`, które pozwalają używać nazw angielskich niezależnie od ustawień regionalnych systemu.

Aby znaleźć odpowiedniki numeryczne, uruchom **regedit**, a następnie znajdź *Counter* w *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009*.

Wpis rejestru zawiera informacje takie jak poniżej:

```
1
1847
2
System
4
Memory
6
% Processor Time
10
File Read Operations/sec
12
File Write Operations/sec
14
File Control Operations/sec
16
File Read Bytes/sec
18
File Write Bytes/sec
....
```

Tutaj możesz znaleźć odpowiadające numery dla każdej części tekstowej licznika wydajności, jak w '\System\% Processor Time':

```
System → 2
% Processor Time → 6
```

Następnie możesz użyć tych numerów, aby przedstawić ścieżkę w postaci liczbowej:

\2\6

Parametry licznika wydajności

Możesz wdrożyć niektóre parametry PerfCounter do monitorowania liczników wydajności systemu Windows.

Na przykład możesz dodać je do pliku konfiguracyjnego agenta Zabbix:

```
PerfCounter=UserPerfCounter1,"\Memory\Page Reads/sec",30
or
PerfCounter=UserPerfCounter2,"\4\24",30
```

Po skonfigurowaniu tych parametrów możesz następnie po prostu użyć *UserPerfCounter1* lub *UserPerfCounter2* jako kluczy do utworzenia odpowiednich pozycji.

Pamiętaj, aby po wprowadzeniu zmian w pliku konfiguracyjnym zrestartować agent Zabbix.

7 Aktualizacja masowa

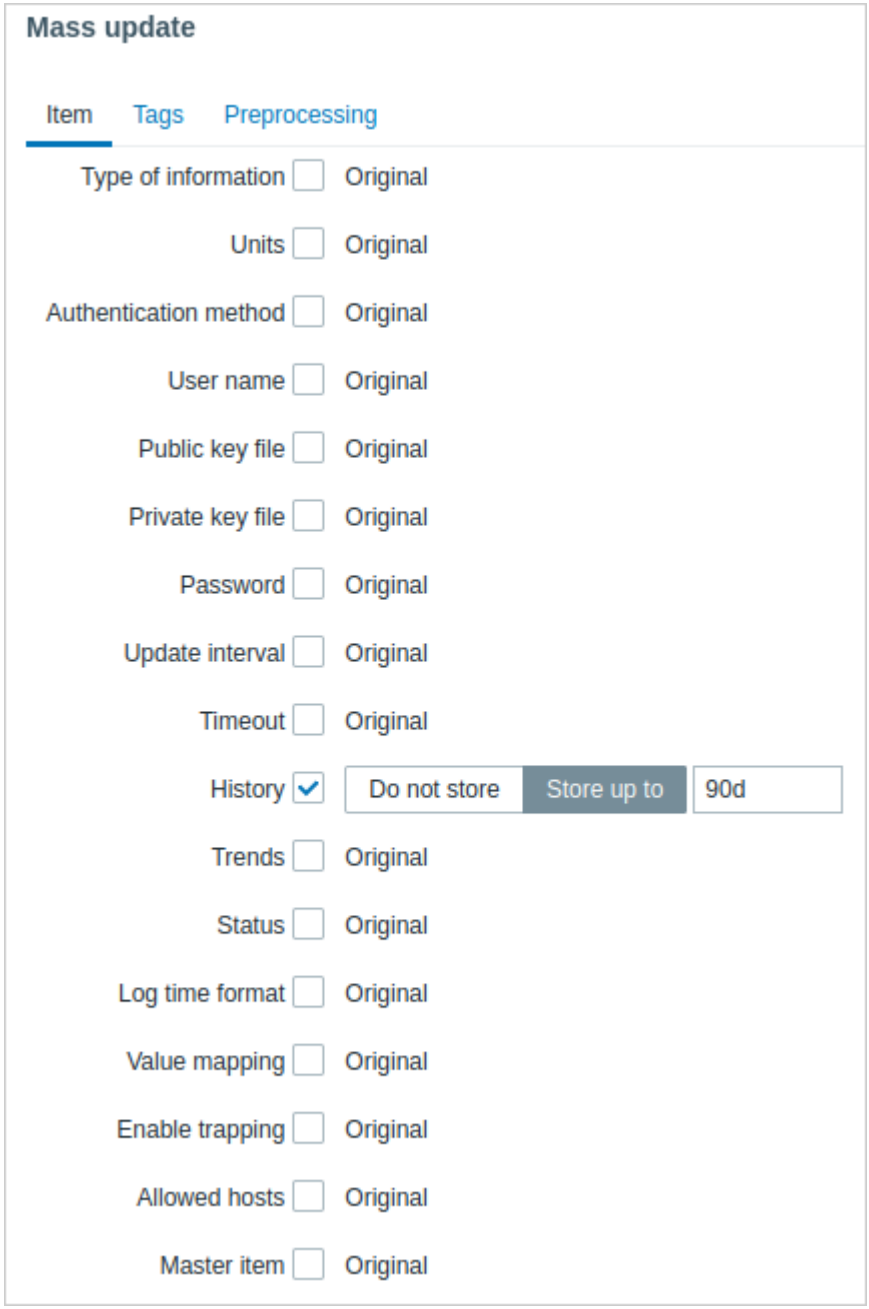
Przegląd

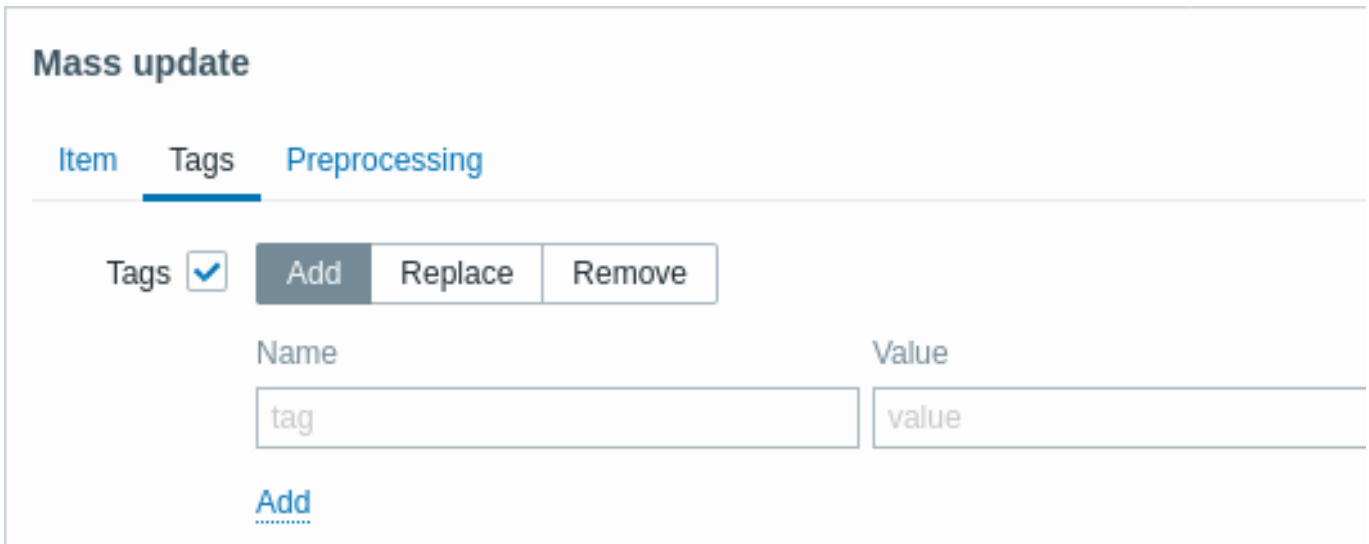
Czasami możesz chcieć zmienić jakiś atrybut dla wielu pozycji jednocześnie. Zamiast otwierać każdą pojedynczą pozycję do edycji, możesz w tym celu użyć funkcji masowej aktualizacji.

Używanie aktualizacji masowej

Aby masowo zaktualizować niektóre pozycje, wykonaj następujące czynności:

- Zaznacz pola wyboru pozycji do zaktualizowania na liście
- Kliknij *Mass update* pod listą
- Przejdź do karty z wymaganymi atrybutami (*Item*, *Tags* lub *Preprocessing*)
- Zaznacz pola wyboru atrybutów do zaktualizowania
- Wprowadź nowe wartości atrybutów



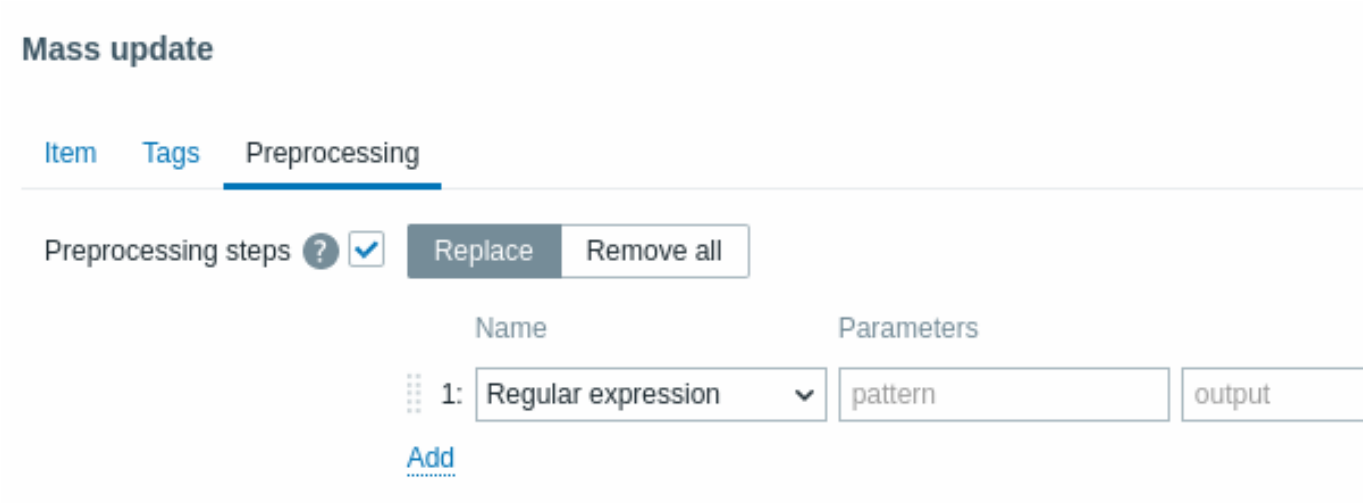


Opcja *Tags* umożliwia:

- *Add* — dodanie określonych tagów do pozycji (tagi o tej samej nazwie, ale różnych wartościach nie są uznawane za „duplikaty" i mogą zostać dodane do tej samej pozycji).
- *Replace* — usunięcie określonych tagów i dodanie tagów z nowymi wartościami

- *Remove* — usunięcie określonych tagów z pozycji

W tagach obsługiwane są makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID}.



Zaznacz pole wyboru obok *Preprocessing steps*, aby włączyć aktualizację masową kroków przetwarzania wstępnego.

Aktualizacja masowa *Preprocessing* umożliwia:

- *Replace* — zastąpienie istniejących kroków przetwarzania wstępnego w pozycjach krokami przetwarzania wstępnego określonymi poniżej
- *Remove all* — usunięcie wszystkich kroków przetwarzania wstępnego z pozycji

Po zakończeniu kliknij *Update*.

8 Mapowanie wartości

Przegląd

Mapowanie wartości umożliwia skonfigurowanie bardziej przyjaznej dla użytkownika reprezentacji otrzymanych wartości przy użyciu mapowań między wartościami numerycznymi/tekstowymi a ich reprezentacjami tekstowymi.

Na przykład, gdy wartość pozycji to „0” lub „1”, mapowania wartości mogą zostać użyte do przedstawienia tych wartości w bardziej przyjazny dla użytkownika sposób:

- 0 → Niedostępne
- 1 → Dostępne

Mapowania wartości dla typów kopii zapasowych danych mogą zostać skonfigurowane następująco:

- F → Pełna
- D → Różnicowa
- I → Przyrostowa

Zakresy wartości dla napięcia mogą zostać skonfigurowane następująco:

- <=209 → Niskie
- 210-230 → OK
- >=231 → Wysokie

Mapowanie wartości jest używane w frontendzie Zabbix oraz w powiadomieniach wysyłanych przez typy mediów.

Attention:
Podstawienie otrzymanej wartości skonfigurowaną reprezentacją jest wykonywane zarówno w frontendzie Zabbix, jak i na serwerze; jednak serwer obsługuje podstawienie tylko w następujących przypadkach:

- podczas wypełniania inwentarza hosta;
- podczas rozwijania obsługiwanych makr {ITEM.VALUE}, {ITEM.LASTVALUE}, {EVENT.OPDATA} oraz {EVENT.CAUSE.OPDATA}.

Mapowania wartości są konfigurowane w szablonach lub hostach. Po skonfigurowaniu są dostępne dla wszystkich pozycji w odpowiednim szablonie lub hoście. Podczas konfigurowania pozycji należy podać nazwę wcześniej skonfigurowanego mapowania wartości w parametrze *Mapowanie wartości*.

Note:
Nie ma dziedziczenia mapowań wartości — hosty i szablony nie dziedziczą mapowań wartości z podłączonych szablonów. Pozycje szablonu na hoście będą nadal używać mapowań wartości skonfigurowanych w szablonie.

Note:
Mapowania wartości mogą być używane z pozycjami mającymi typ informacji *Numeric (unsigned)*, *Numeric (float)* oraz *Character*.

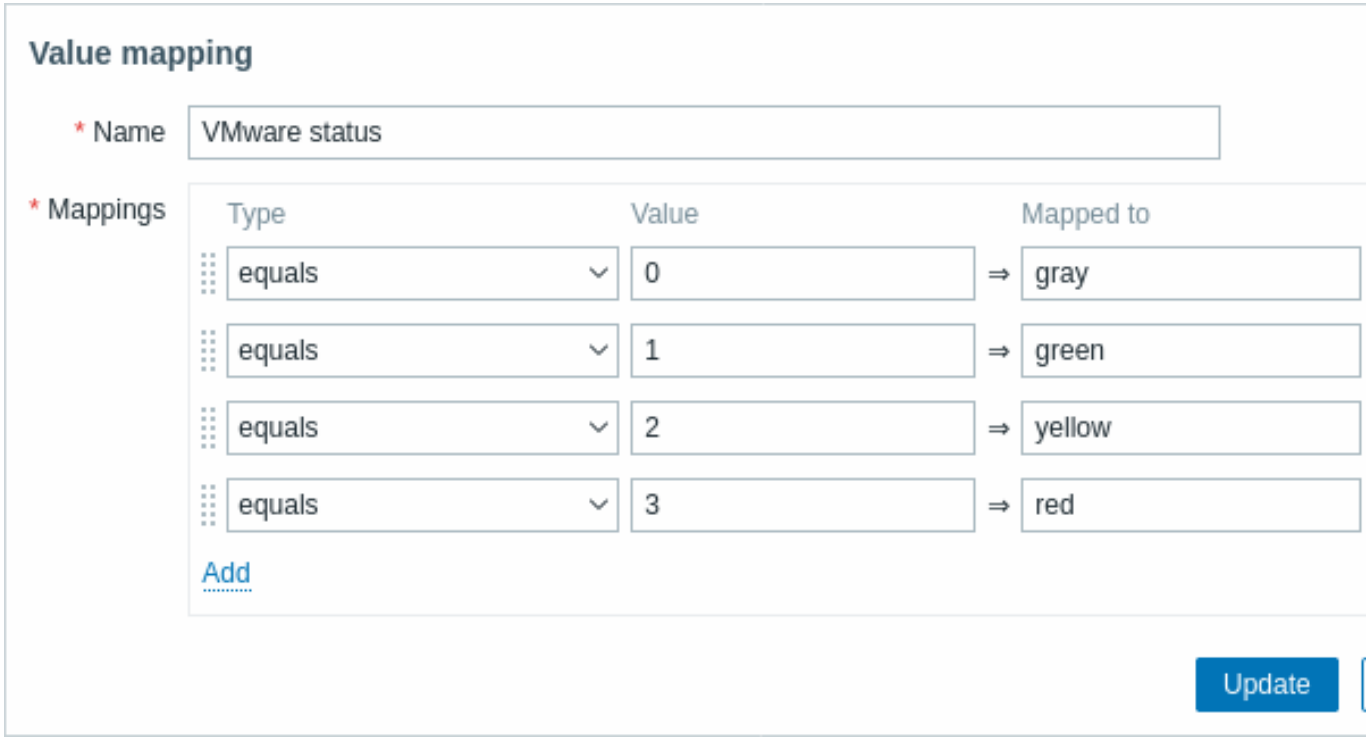
Mapowania wartości są eksportowane/importowane wraz z odpowiednim szablonem lub hostem. Mogą również być masowo aktualizowane przy użyciu formularzy masowej aktualizacji hosta i szablonu.

Konfiguracja

Aby skonfigurować mapowanie wartości, wykonaj następujące kroki:

1. Otwórz formularz konfiguracji hosta lub szablonu.

2. Na karcie *Mapowanie wartości* kliknij *Dodaj*, aby dodać nowe mapowanie wartości, lub kliknij nazwę istniejącego mapowania, aby je edytować.



Parametry mapowania wartości:

Parametr	Opis
Nazwa	Unikalna nazwa zestawu mapowań wartości.
Mapowania	Poszczególne reguły mapowania wartości numerycznych/ciągów znaków na reprezentacje tekstowe. Mapowanie jest stosowane zgodnie z kolejnością reguł, którą można zmieniać przez przeciąganie.

Parametr	Opis
Typ	Typ mapowania: **równa się** - równe wartości zostaną zmapowane; **jest większe lub równe** - równe lub większe wartości zostaną zmapowane; **jest mniejsze lub równe** - równe lub mniejsze wartości zostaną zmapowane; **w zakresie** - wartości z zakresu zostaną zmapowane; zakres jest wyrażany jako <number1>-<number2> lub <number>; obsługiwanych jest wiele zakresów (na przykład 1-10,101-110,201); **regexp** - wartości odpowiadające wyrażeniu regularnemu zostaną zmapowane (globalne wyrażenia regularne nie są obsługiwane); **domyślne** - wszystkie pozostałe wartości zostaną zmapowane, inne niż te ze zdefiniowanymi mapowaniami. W przypadku mapowania zakresów obsługiwane są tylko numeryczne typy wartości (*jest większe lub równe*, *jest mniejsze lub równe*, *w zakresie*).
Wartość	Wartość przychodząca (może zawierać zakres lub wyrażenie regularne, w zależności od typu mapowania).
Mapowane na	Reprezentacja tekstowa (do 64 znaków) dla wartości przychodzącej.

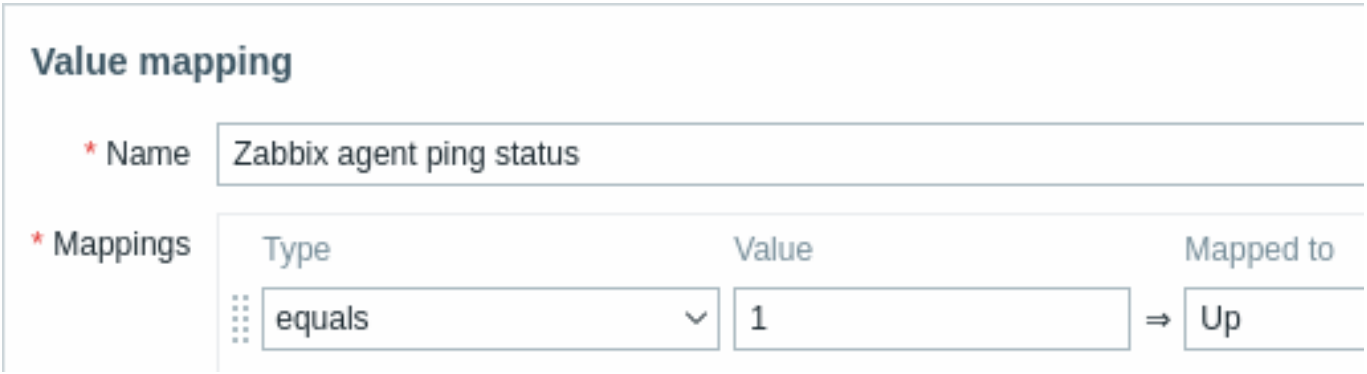
Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Podczas wyświetlania mapowania wartości na liście widoczne są tylko pierwsze trzy mapowania, a trzy kropki wskazują, że istnieją kolejne mapowania.



Przykład mapowania wartości

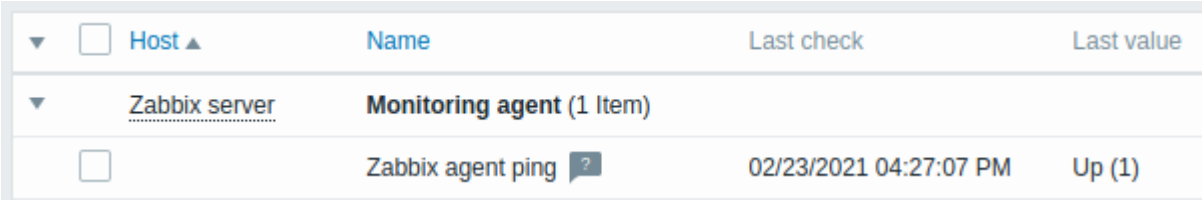
Jedna ze wstępnie zdefiniowanych pozycji agenta, *Zabbix agent ping*, używa mapowania wartości na poziomie szablonu „Zabbix agent ping status” do wyświetlania swoich wartości.



W formularzu konfiguracji pozycji można znaleźć odwołanie do tego mapowania wartości w polu *Mapowanie wartości*:

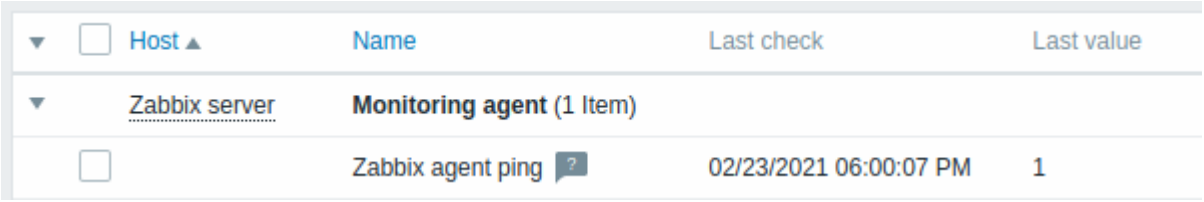


To mapowanie jest używane w sekcji *Monitoring* → *Najnowsze dane* do wyświetlania „Up” (z surową wartością w nawiasach).



Note:
W sekcji *Najnowsze dane* wyświetlane wartości są skracane do 20 znaków. Jeśli używane jest mapowanie wartości, to skrócenie nie jest stosowane do zmapowanej wartości, lecz tylko do surowej wartości (wyświetlanej w nawiasach).

Bez wstępnie zdefiniowanego mapowania wartości widoczne byłoby tylko „1”, co mogłoby być trudne do zrozumienia.



9 Kolejka

Przegląd

Kolejka wyświetla pozycje oczekujące na odświeżenie. Kolejka jest tylko **logiczną** reprezentacją danych. W Zabbix nie ma kolejki IPC ani żadnego innego mechanizmu kolejkowania.

Pozycje monitorowane przez proxy są również uwzględniane w kolejce — będą liczone jako oczekujące w okresie aktualizacji danych historycznych proxy.

W kolejce wyświetlane są tylko pozycje z zaplanowanymi czasami odświeżania. Oznacza to, że następujące typy pozycji są wykluczone z kolejki:

- aktywne pozycje log, logrt i event log agenta Zabbix
- pozycje SNMP trap
- pozycje trapper
- pozycje monitorowania WWW
- pozycje zależne

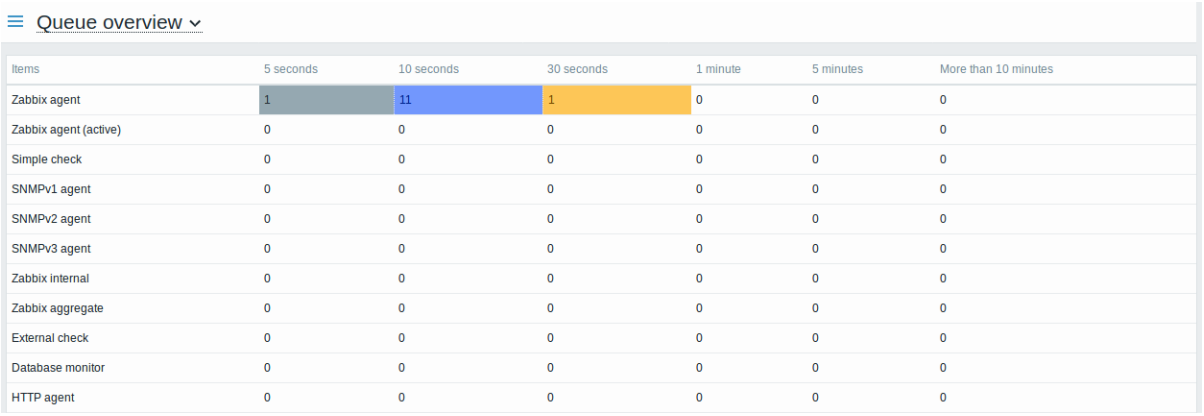
Statystyki pokazywane przez kolejkę są dobrym wskaźnikiem wydajności serwera Zabbix.

Kolejka jest pobierana bezpośrednio z serwera Zabbix przy użyciu protokołu JSON. Informacje są dostępne tylko wtedy, gdy serwer Zabbix jest uruchomiony.

Attention:
Pozycje nie są uwzględniane w kolejce, jeśli interfejs pozycji staje się niedostępny z powodu problemów z połączeniem lub nieprawidłowego działania agenta.

Odczytywanie kolejki

Aby odczytać kolejkę, przejdź do *Administration* > *Queue* > *Queue overview*.



Obraz tutaj jest ogólnie „ok”, więc możemy założyć, że serwer działa prawidłowo.

Kolejka pokazuje niektóre pozycje oczekujące do 30 sekund. Dobrze byłoby wiedzieć, jakie to pozycje.

Aby to sprawdzić, wybierz *Queue details* z rozwijanego menu w tytule. Teraz możesz zobaczyć listę tych opóźnionych pozycji.

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

Mając te szczegóły, może być możliwe ustalenie, dlaczego te pozycje są opóźnione.

W przypadku jednej lub dwóch opóźnionych pozycji prawdopodobnie nie ma powodów do niepokoju. Mogą zostać zaktualizowane za sekundę. Jednak jeśli widzisz wiele pozycji opóźnionych zbyt długo, może to oznaczać poważniejszy problem.

Zobacz także: Wyrównywanie stref czasowych podczas używania scheduling intervals.

Queue overview

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	0	1	1	26	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
IPMI agent	0	0	0	0	0	0
SSH agent	0	0	0	0	0	0
TELNET agent	0	0	0	0	0	0
JMX agent	0	0	0	0	0	0
Calculated	0	0	0	0	0	0

Pozycja kolejki

Specjalna wewnętrzna pozycja **zabbix[queue,<from>,<to>]** może być używana do monitorowania stanu kolejki w Zabbix. Zwraca liczbę pozycji opóźnionych o ustawiony czas. Więcej informacji można znaleźć w sekcji Pozycje wewnętrzne.

10 Cache wartości

Przegląd

Aby znacznie przyspieszyć obliczanie wyrażeń wyzwalaczy, pozycji obliczanych oraz niektórych makr, serwer Zabbix obsługuje opcję pamięci podręcznej wartości.

Ta pamięć podręczna w pamięci operacyjnej może być używana do uzyskiwania dostępu do danych historycznych zamiast wykonywania bezpośrednich wywołań SQL do bazy danych. Jeśli wartości historyczne nie są obecne w pamięci podręcznej, brakujące wartości są pobierane z bazy danych, a pamięć podręczna jest odpowiednio aktualizowana.

Wartości pozycji pozostają w pamięci podręcznej wartości do momentu, gdy:

- pozycja zostanie usunięta (wartości z pamięci podręcznej są usuwane po następnej synchronizacji konfiguracji);
- wartość pozycji znajdzie się poza zakresem czasu lub liczby określonym w wyrażeniu wyzwalacza/obliczanej pozycji (wartość z pamięci podręcznej jest usuwana po odebraniu nowej wartości);
- zakres czasu lub liczby określony w wyrażeniu wyzwalacza/obliczanej pozycji zostanie zmieniony tak, że do obliczeń wymagane będzie mniej danych (niepotrzebne wartości z pamięci podręcznej są usuwane po 24 godzinach).

Note:
Stan pamięci podręcznej wartości można obserwować za pomocą opcji runtime control serwera `diaginfo` (lub `diaginfo=valuecache`) oraz sprawdzając sekcję z informacjami diagnostycznymi pamięci podręcznej wartości. Może to być przydatne przy określaniu błędnie skonfigurowanych wyzwalaczy lub obliczanych pozycji.

Aby włączyć funkcjonalność pamięci podręcznej wartości, w pliku konfiguracji serwera Zabbix obsługiwany jest opcjonalny parametr **ValueCacheSize**.

Do monitorowania pamięci podręcznej wartości obsługiwane są dwie wewnętrzne pozycje: **zabbix[vcache,buffer,<mode>]** oraz **zabbix[vcache,cache,<parameter>]**. Więcej szczegółów można znaleźć w sekcji pozycje wewnętrzne.

11 Wykonaj teraz

Przegląd

Opcja *Wykonaj teraz* umożliwia natychmiastowe wykonywanie kontroli pasywnych.

Pobieranie wartości pozycji w Zabbix odbywa się zgodnie ze skonfigurowanymi interwałami aktualizacji. Niektóre pozycje (np. reguły wykrywania niskopoziomowego) mają długie interwały i może być konieczne natychmiastowe pobranie nowej wartości — na przykład w celu wykrycia zmian w wykrywalnych zasobach.

Opcja *Wykonaj teraz* jest obsługiwana dla następujących typów pozycji:

- agent Zabbix (pasywny)
- Kontrola prosta
- agent SNMP (v1/v2/v3)
- Wewnętrzny Zabbix
- Kontrola zewnętrzna
- Monitor bazy danych
- agent HTTP
- agent IPMI
- agent SSH
- agent TELNET
- agent JMX
- Obliczana
- Pozycja zależna
- Skrypt
- Przeglądarka

Jeśli pozycja jest typu „Pozycja zależna”, jej pozycja nadrzędna również musi być jednym z powyższych typów.

Konfiguracja

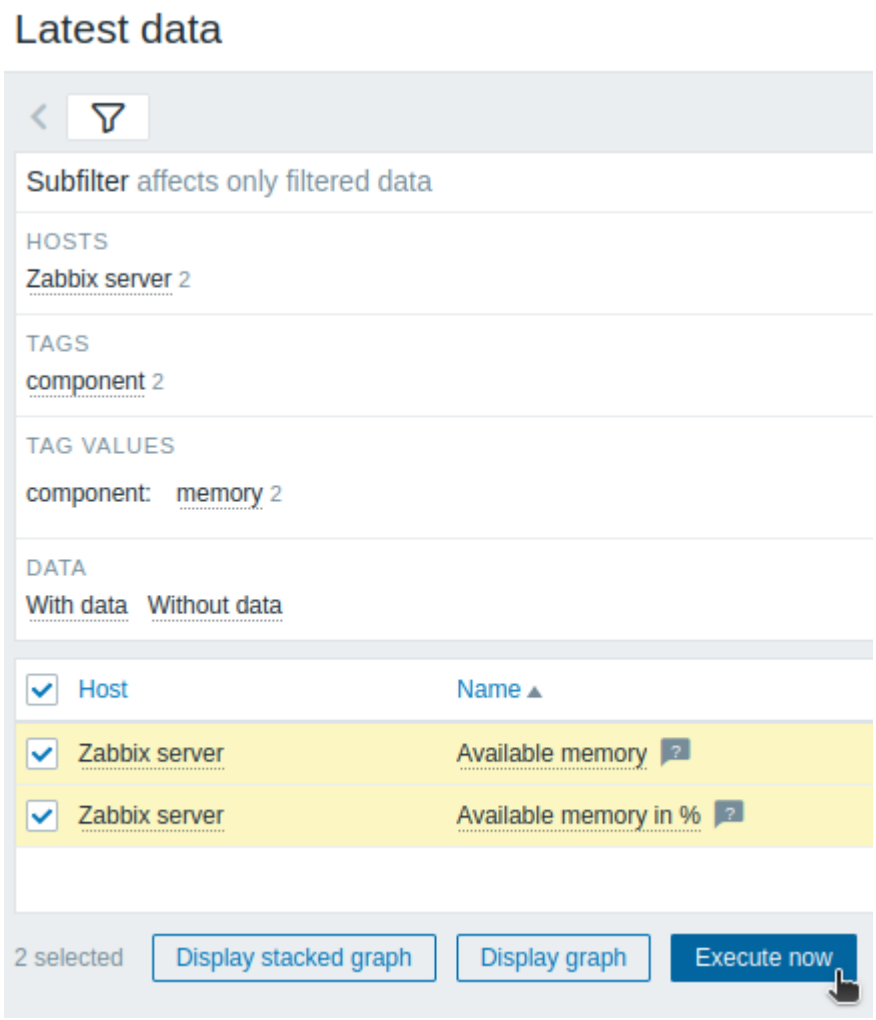
Możesz wykonywać pasywne sprawdzenia z poziomu:

- *Monitoring* > *Najnowsze dane*
- *Zbieranie danych* > *Hosty* > *Pozycje* lub *Odkrywanie*
- menu pozycji
- Podczas edycji pozycji lub reguł wykrywania niskiego poziomu (w ich formularzach konfiguracyjnych)

Attention:
Sprawdzenie musi istnieć w pamięci podręcznej konfiguracji, aby mogło zostać wykonane; zobacz także CacheUpdateFrequency. Pamięć podręczna **nie jest aktualizowana** przed wykonaniem sprawdzenia, więc ostatnie zmiany w konfiguracji pozycji lub reguły wykrywania niskiego poziomu nie zostaną uwzględnione. Aby testować pozycje lub reguły LLD, które są tworzone lub zostały właśnie utworzone, możesz zamiast tego użyć opcji Test.

Na przykład, aby wykonać pasywne sprawdzenie w *Monitoring* > *Najnowsze dane*:

1. Wybierz pozycje; możesz wykonać sprawdzenie dla wielu pozycji jednocześnie.
2. Kliknij *Wykonaj teraz*.



Note:
W sekcji *Najnowsze dane* użytkownicy mogą używać opcji *Wykonaj teraz* dla pozycji na hostach, do których mają uprawnienia odczyt-zapis, albo uprawnienia tylko do odczytu oraz włączoną dla swojej roli akcję Invoke "Execute now" on read-only hosts.

12 Ograniczanie kontroli agenta

Przegląd

Możesz kontrolować, których kluczy pozycji agent Zabbix lub agent 2 może używać albo których nie może używać podczas wykonywania kontroli pozycji, zdalnych poleceń lub skryptów.

Aby to zrobić, użyj tych parametrów konfiguracji agenta do zdefiniowania reguł zezwalania/odrzucania:

- `AllowKey=<pattern>`
- `DenyKey=<pattern>`

`<pattern>` musi zawierać pojedynczy klucz pozycji i obsługuje symbole wieloznaczne (`*`). Symbol wieloznaczny dopasowuje dowolną liczbę dowolnych znaków w swoim położeniu i może być używany do dopasowywania kluczy pozycji lub parametrów (np. `vfs.file.*[*]`).

Attention:
Aby zwiększyć bezpieczeństwo, zaleca się używanie dokładnych kluczy pozycji zamiast symboli wieloznacznych. Szczegóły znajdują się w sekcji Zabezpieczanie reguł zezwalania/odrzucania.

W przeciwieństwie do innych parametrów konfiguracji agenta możesz określić nieograniczoną liczbę parametrów `AllowKey` lub `DenyKey`.

Ważne uwagi

- Wszystkie pozycje `system.run` są domyślnie wyłączone (nawet gdy `DenyKey` jest puste), tak jakby `DenyKey=system.run[*]`

było ustawione jako ostatnia reguła. Z tego powodu możesz zezwolić na określone pozycje system.run bez jawnego zabraniania innych pozycji system.run.

- Pozycja określona w AllowKey musi być również określona w DenyKey (z wyjątkiem pozycji system.run); w przeciwnym razie agent Zabbix **nie uruchomi się**.
- Jeśli to możliwe, używaj AllowKey, aby zezwalać tylko na wymagane pozycje i zabraniać wszystkiego innego. Niektóre klucze mogą zostać wykorzystane do odczytu niezamierzonych plików poprzez przechodzenie po ścieżkach (np. vfs.file.contents["../../../../etc/passwd"]), a nowe wersje agenta Zabbix mogą wprowadzać klucze nieobjęte regułami DenyKey.
- Konfiguracja AllowKey i DenyKey nie wpływa na parametry agenta HostnameItem, HostMetadataItem ani HostInterfaceItem.
- Zabronione pozycje stają się nieobsługiwane bez żadnych wskazówek ani komunikatów o błędach; na przykład:
  - Parametr wiersza poleceń agenta Zabbix --print (-p) nie pokaże kluczy zabronionych pozycji.
  - Parametr wiersza poleceń agenta Zabbix --test (-t) zwróci „Unsupported item key." dla kluczy zabronionych pozycji.
  - Gdy rejestrowanie jest włączone (LogRemoteCommands=1), plik dziennika agenta Zabbix nie będzie rejestrować zabronionych zdalnych poleceń.

Kolejność reguł zezwalania/odmawiania

Możesz określić nieograniczoną liczbę reguł AllowKey lub DenyKey, jednak ich kolejność ma znaczenie.

- Reguły są oceniane jedna po drugiej, od góry do dołu.
- Gdy klucz pozycji pasuje do reguły, zostaje odpowiednio dozwolony lub zabroniony, a ocena reguł zostaje zatrzymana.

Na przykład podczas oceny vfs.file.contents[/etc/passwd] reguły są przetwarzane w następujący sposób:

```
AllowKey=vfs.file.contents[/tmp/app.log]    # Wzorzec klucza pozycji nie pasuje, agent przechodzi do nastę
AllowKey=vfs.file.contents[/etc/passwd]     # Wzorzec klucza pozycji pasuje; agent zezwala na sprawdzenie
DenyKey=vfs.file.*[*]                       # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
```

Poniższa kolejność reguł spowoduje odmowę sprawdzenia pozycji:

```
DenyKey=vfs.file.*[*]                       # Wzorzec klucza pozycji pasuje; agent odmawia sprawdzenia poz
AllowKey=vfs.file.contents[/etc/passwd]     # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
AllowKey=vfs.file.contents[/tmp/app.log]    # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
```

Przykłady

Poniższe przykłady pokazują typowe wzorce konfiguracji dla AllowKey i DenyKey.

Zezwalanie na określone kontrole i polecenia

Zezwól tylko na dwie kontrole pozycji vfs.file i dwa polecenia system.run:

```
AllowKey=vfs.file.contents[/tmp/app.log]
AllowKey=vfs.file.size[/tmp/app.log]
AllowKey=system.run[/usr/bin/uptime]
AllowKey=system.run[/usr/bin/df -h /]
DenyKey=vfs.file.*[*]
```

Note:
Ustawienie DenyKey=system.run[*] nie jest konieczne, ponieważ wszystkie pozostałe polecenia system.run są domyślnie zabronione.

Zezwalanie na skrypty

Zezwól agentowi Zabbix na wykonywanie skryptów na hostach wszystkimi dostępnymi metodami:

- Skrypty globalne, które mogą być wykonywane we frontendzie lub przez API (ta metoda zawsze używa klucza system.run[myscript.sh])
- Zdalne polecenia z operacji akcji (ta metoda zawsze używa klucza system.run[myscript.sh,nowait])
- Pozycje agenta Zabbix system.run ze skryptem, na przykład:
  - system.run[myscript.sh]
  - system.run[myscript.sh,wait]
  - system.run[myscript.sh,nowait]

```
AllowKey=system.run[myscript.sh,*]
```

Aby kontrolować parametr `wait/nowait`, należy ustawić inną regułę. Na przykład można zezwolić tylko na pozycje `system.run[myscript.s` wykluczając w ten sposób inne metody:

```
AllowKey=system.run[myscript.sh,wait]
```

Zabezpieczanie reguł allow/deny

Ten przykład pokazuje, jak zabezpieczyć zbyt liberalne reguły `AllowKey` lub `DenyKey`.

Rozważ następujące reguły:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat*"]
DenyKey=vfs.file.*
DenyKey=system.cpu.load[*]
```

Note:
W systemie Windows spacje w ścieżce należy poprzedzać znakiem daszka (^).

Te reguły zawierają symbol wieloznaczny (*), który może zostać niewłaściwie wykorzystany:

- Skrypt `test.bat` może zostać uruchomiony z dowolnymi argumentami, w tym niezamierzonymi.
- Wzorzec `vfs.file.*` dopasowuje tylko klucze pozycji bez parametrów; jednak wszystkie pozycje `vfs.file` wymagają parametrów.
- Wzorzec `system.cpu.load[*]` dopasowuje tylko klucze pozycji z parametrami; jednak pozycje `system.cpu.load` nie wymagają parametrów.

Aby zabezpieczyć te reguły, jawnie zezwól na wykonywanie `test.bat` tylko z określonymi argumentami oraz zabroń poprawnych wzorców kluczy pozycji; na przykład:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat status"]
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat version"]
DenyKey=vfs.file.*[*]
DenyKey=system.cpu.load
DenyKey=system.cpu.load[*]
```

Możesz przetestować reguły, uruchamiając następujące polecenia, które zwrócą `ZBX_NOTSUPPORTED`.

```
cd "C:\Program Files\Zabbix Agent 2"
zabbix_agent2.exe -t system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat debug"]
zabbix_agent2.exe -t vfs.file.size["C:\ProgramData\MyApp\config.ini"]
zabbix_agent2.exe -t vfs.file.contents["C:\Windows\System32\drivers\etc\hosts"]
zabbix_agent2.exe -t system.cpu.load
zabbix_agent2.exe -t system.cpu.load[all,avg1]
```

Przykłady wzorców

Poniższa tabela pokazuje, jak dopasowywane są wzorce kluczy pozycji:

- Klucz pasuje do wzorca tylko wtedy, gdy spełnia **wszystkie** warunki w kolumnie *Dopasowania*.
- Parametry muszą być w całości ujęte w nawiasy kwadratowe (np. `vfs.file.contents[*` i `vfs.file.contents*utf8]` są nieprawidłowymi wzorcami).

Wzorzec	Dopasowania	Przykłady
*	Dowolny klucz z parametrami lub bez parametrów	

`vfs.file.*`	Klucz zaczyna się od `vfs.file.`
	Brak parametrów
		Pasuje:
		`vfs.file.size`
		`vfs.file.contents`
		
		Nie pasuje:
		`vfs.file.contents[]`
		`vfs.file.size[/var/log/app.log]`

Wzorzec	Dopasowania	Przykłady

`vfs.*.contents`	Klucz zaczyna się od `vfs.`
	Klucz kończy się na `.contents`
	Brak parametrów
		Pasuje:
		`vfs..contents`
		`vfs.mount.point.file.contents`
		
		Nie pasuje:
		`vfs.contents`
		`vfs.file.contents[]`

`vfs.file.*[*]`	Klucz zaczyna się od `vfs.file.`
	Dowolne lub puste parametry
		Pasuje
		`vfs.file.get.custom[]`
		`vfs.file.size[/var/log/app.log, utf8]`
		
		Nie pasuje:
		`vfs.file.get.custom`

`vfs.file.contents`	Klucz to `vfs.file.contents` Brak parametrów	Pasuje: `vfs.file.contents`

`vfs.file.contents[]`	Klucz to `vfs.file.contents[]`
	Puste parametry
		Nie pasuje:
		`vfs.file.contents[/etc/passwd]`
		Pasuje:
		`vfs.file.contents[]`

`vfs.file.contents[*]`	Klucz to `vfs.file.contents`
	Dowolne lub puste parametry
		Nie pasuje:
		`vfs.file.contents`
		Pasuje:
		`vfs.file.contents[/path/to/file]`

`vfs.file.contents[/etc/passwd,*]`	Klucz to `vfs.file.contents`
	Pierwszy parametr to `/etc/passwd`
	Dowolny lub pusty drugi parametr
		Nie pasuje:
		`vfs.file.contents`
		Pasuje:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`

`vfs.file.contents[*passwd*]`	Klucz to `vfs.file.contents`
	Pierwszy parametr zawiera `passwd`
	Brak drugiego parametru
		Nie pasuje:
		`vfs.file.contents[]`
		`vfs.file.contents[/etc/passwd]`
		Pasuje:
		`vfs.file.contents[/etc/passwd]`

`vfs.file.contents[*passwd*,*]`	Klucz to `vfs.file.contents`
	Pierwszy parametr zawiera `passwd`
	Dowolny lub pusty drugi parametr
		Nie pasuje:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`
		Pasuje:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf8]`

`vfs.file.contents[/etc/passwd,utf8]`	Klucz to `vfs.file.contents`
	Pierwszy parametr to `/etc/passwd`
	Drugi parametr to `utf8`
		Nie pasuje:
		`vfs.file.contents[/etc/passwd]`
		`vfs.file.contents[/tmp/test]`
		Pasuje:
		`vfs.file.contents[/etc/passwd,utf8]`
		
		Nie pasuje:
		`vfs.file.contents[/etc/passwd,]`
		`vfs.file.contents[/etc/passwd,utf16]`

3 Wyzwalacze

Przegląd

Wyzwalacze są logicznymi wyrażeniami, które ”oceniają” dane zbierane przez pozycje i reprezentują bieżący stan systemu.

Chociaż pozycje służą do zbierania danych systemowych, śledzenie tych danych przez cały czas w oczekiwaniu na warunek, który jest alarmujący lub wymaga uwagi, jest bardzo niepraktyczne. Zadanie ”oceny” danych można pozostawić wyrażeniom wyzwalaczy.

Wyrażenia wyzwalaczy pozwalają zdefiniować próg określający, jaki stan danych jest ”akceptowalny”. Dlatego jeśli napływające dane przekroczą akceptowalny stan, wyzwalacz zostaje ”uruchomiony” - albo zmienia swój status na PROBLEM.

Wyzwalacz może mieć następujący status:

Status	Opis
OK	To jest normalny status wyzwalacza.
Problem	Coś się wydarzyło. Na przykład obciążenie procesora jest zbyt wysokie.
Unknown	Nie można obliczyć wartości wyzwalacza. Zobacz Unknown status.

W prostym wyzwalaczu możemy chcieć ustawić próg dla pięciominutowej średniej pewnych danych, na przykład obciążenia CPU. Osiąga się to przez zdefiniowanie wyrażenia wyzwalacza, w którym:

- funkcja ‘avg’ jest zastosowana do wartości otrzymanej w kluczu pozycji
- używany jest pięciominutowy okres oceny
- ustawiony jest próg ’2’

```
avg(/host/key,5m)>2
```

Ten wyzwalacz zostanie ”uruchomiony” (stanie się PROBLEM), jeśli pięciominutowa średnia będzie *większa* niż 2.

W bardziej złożonym wyzwalaczu wyrażenie może zawierać **kombinację** wielu funkcji i wielu progów. Zobacz także: Wyrażenie wyzwalacza.

Nie można tworzyć wyzwalaczy dla pozycji zwracających wartości binarne data type.

Note:
Po włączeniu wyzwalacza (zmianie jego statusu konfiguracji z *Disabled* na *Enabled*) wyrażenie wyzwalacza jest oceniane natychmiast po otrzymaniu wartości przez pozycję w nim zawartą lub gdy nadejdzie czas obsługi funkcji date and time i/lub nodata().

Większość funkcji wyzwalacza jest oceniana na podstawie danych wartości pozycji history, podczas gdy niektóre funkcje wyzwalacza do długoterminowej analityki, np. **trendavg()**, **trendcount()**, itp., używają danych trendów.

Czas obliczeń

Wyzwalacz jest przeliczany za każdym razem, gdy serwer Zabbix otrzyma nową wartość, która jest częścią wyrażenia. Gdy zostanie odebrana nowa wartość, każda funkcja uwzględniona w wyrażeniu jest przeliczana (nie tylko ta, która otrzymała nową wartość).

Dodatkowo wyzwalacz jest przeliczany za każdym razem, gdy zostanie odebrana nowa wartość, **oraz** co 30 sekund, jeśli w wyrażeniu są używane funkcje data i czas i/lub nodata().

Funkcje data i czas i/lub nodata() są przeliczane co 30 sekund przez proces synchronizacji historii Zabbix.

Wyzwalacze, które odwołują się wyłącznie do funkcji trendów, są oceniane raz na najmniejszy okres czasu w wyrażeniu. Zobacz także funkcje trendów.

Okres ewaluacji

Okres ewaluacji jest używany w funkcjach odwołujących się do historii pozycja. Pozwala określić interesujący nas przedział czasu. Może być określony jako okres czasu (30s, 10m, 1h) lub jako zakres wartości (#5 - dla pięciu najnowszych wartości).

Okres ewaluacji jest liczony do ”teraz” - gdzie ”teraz” to najnowszy czas ponownego obliczenia wyzwalacza (zobacz Czas obliczeń powyżej); ”teraz” nie oznacza czasu ”teraz” na serwerze.

Okres ewaluacji określa:

- Wszystkie wartości między ”teraz - okres czasu” a ”teraz” (lub, po zastosowaniu przesunięcia czasu, między ”teraz - przesunięcie czasu - okres czasu” a ”teraz - przesunięcie czasu”).
- Określoną liczbę wartości z przeszłości, do ”teraz”.

Jeśli dla określonego okresu ewaluacji nie istnieją żadne dane, wyzwalacz lub obliczana pozycja korzystająca z tej funkcji staje się nieobsługiwana.

Należy pamiętać, że:

- Jeśli w wyzwalaczu używana jest tylko jedna funkcja (odwołująca się do historii danych), ”teraz” jest zawsze najnowszą otrzymaną wartością. Na przykład, jeśli ostatnia wartość została odebrana godzinę temu, okres ewaluacji będzie traktowany jako obejmujący najnowszą wartość sprzed godziny.
- Nowy wyzwalacz jest obliczany natychmiast po otrzymaniu pierwszej wartości (funkcje historii); zostanie obliczony w ciągu 30 sekund dla funkcji date and time oraz nodata(). W związku z tym wyzwalacz zostanie obliczony nawet wtedy, gdy ustawiony okres ewaluacji (na przykład jedna godzina) jeszcze nie upłynął od momentu utworzenia wyzwalacza. Wyzwalacz zostanie również obliczony po otrzymaniu pierwszej wartości, nawet jeśli zakres ewaluacji został ustawiony na przykład na dziesięć najnowszych wartości.

Nieznany status

Możliwe jest, że w wyrażeniu wyzwalacza pojawi się nieznany operand, jeśli:

- użyto nieobsługiwanej pozycji
- ocena funkcji dla obsługiwanej pozycji zakończy się błędem

W takim przypadku wyzwalacz zazwyczaj przyjmuje wartość „unknown” (choć istnieją pewne wyjątki). Więcej informacji można znaleźć w sekcji Wyrażenia z nieznanymi operandami.

Możliwe jest otrzymywanie powiadomień o nieznanych wyzwalaczach.

1 Konfigurowanie wyzwalacza

Przegląd

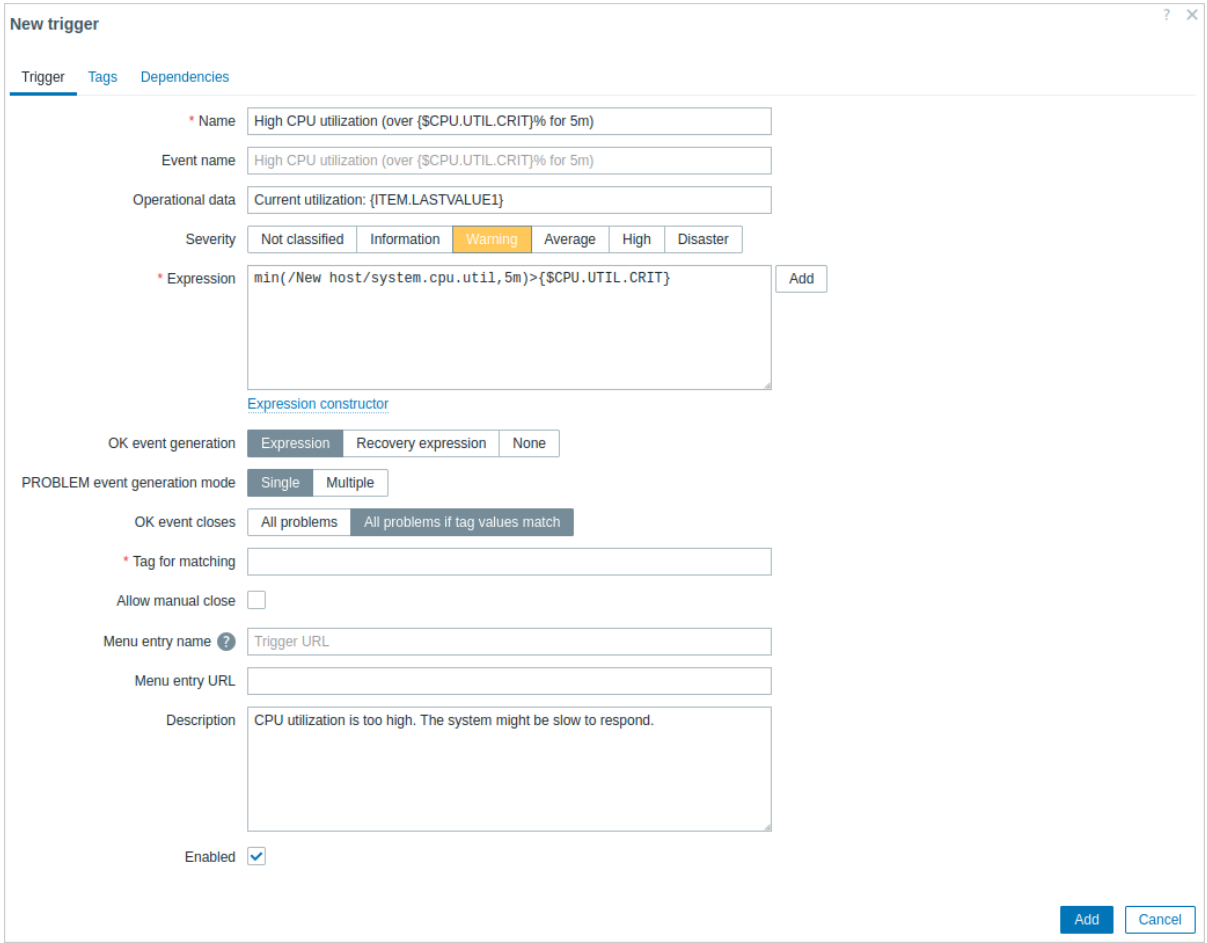
Aby skonfigurować wyzwalacz, wykonaj następujące czynności:

- Przejdź do: *Data collection* > *Hosts*
- Kliknij *Triggers* w wierszu hosta
- Kliknij *Create trigger* po prawej stronie (lub nazwę wyzwalacza, aby edytować istniejący wyzwalacz)
- Wprowadź parametry wyzwalacza w formularzu

Zobacz także ogólne informacje o wyzwalaczach i czasie ich obliczania.

Konfiguracja

Karta **Wyzwalacz** zawiera wszystkie podstawowe atrybuty wyzwalacza.

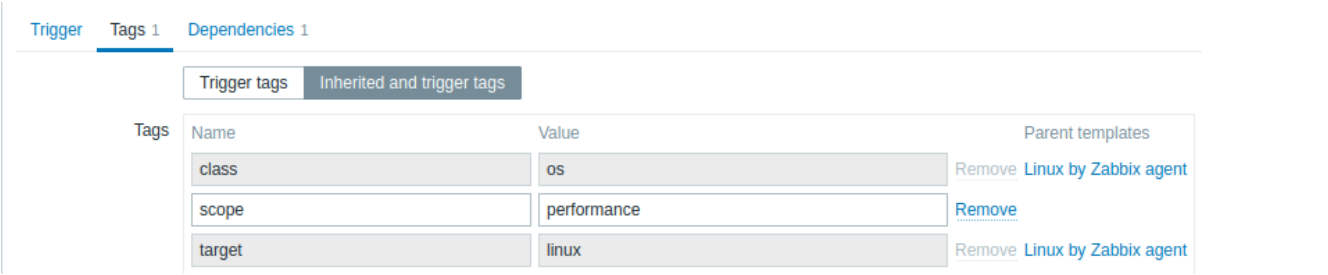


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa wyzwalacza. Obsługiwane makra to: {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {ITEM.VALUE}, {ITEM.VALUE.AGE}, {ITEM.VALUE.DATE}, {ITEM.VALUE.TIME}, {ITEM.VALUE.TIMESTAMP}, {ITEM.LASTVALUE}, {ITEM.LASTVALUE.AGE}, {ITEM.LASTVALUE.DATE}, {ITEM.LASTVALUE.TIME}, {ITEM.LASTVALUE.TIMESTAMP}, {ITEM.LOG.*} oraz makra użytkownika {$MACRO}. Makra **$1, $2...$9** można użyć do odwołania się do pierwszej, drugiej... dziewiątej stałej wyrażenia. *Uwaga*: makra $1-$9 zostaną poprawnie rozwinięte, jeśli odnoszą się do stałych w stosunkowo prostych, jednoznacznych wyrażeniach. Na przykład nazwa ”Processor load above $1 on {HOST.NAME}” zostanie automatycznie zmieniona na ”Processor load above 5 on New host”, jeśli wyrażenie ma postać last(/New host/system.cpu.load[percpu,avg1])>5.
Event name	Jeśli zostanie zdefiniowana, ta nazwa będzie używana do tworzenia nazwy zdarzenia problemu. Domyślnie nazwa zdarzenia jest taka sama jak nazwa wyzwalacza. Nazwa zdarzenia może być używana do tworzenia czytelnych alertów zawierających dane problemu (zobacz przykład). Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a także makra wyrażenia {TIME}, {TIMESTAMP} i {?EXPRESSION}.
Operational data	Dane operacyjne umożliwiają definiowanie dowolnych ciągów znaków wraz z makrami. Makra będą dynamicznie rozwijane do danych w czasie rzeczywistym w *Monitoring* > *Problems*. Podczas gdy makra w nazwie wyzwalacza (zobacz wyżej) zostaną rozwinięte do swoich wartości w momencie wystąpienia problemu i staną się podstawą statycznej nazwy problemu, makra w danych operacyjnych zachowują możliwość wyświetlania najnowszych informacji dynamicznie. Jeśli na poziomie wyzwalacza nie skonfigurowano danych operacyjnych, zostaną wyświetlone najnowsze wartości wszystkich pozycji z wyrażenia. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza.
Severity	Ustaw wymaganą severity wyzwalacza, klikając przyciski.
Expression	Logiczne wyrażenie używane do definiowania warunków problemu. Obsługiwane są sufiksy czasu oraz sufiksy rozmiaru pamięci. Problem jest tworzony po spełnieniu wszystkich warunków zawartych w wyrażeniu, tj. gdy wyrażenie przyjmuje wartość TRUE. Problem zostanie rozwiązany, gdy tylko wyrażenie przyjmie wartość FALSE, chyba że w *Recovery expression* zostaną określone dodatkowe warunki odzyskiwania.
OK event generation	Opcje generowania zdarzeń OK: **Expression** - zdarzenia OK są generowane na podstawie tego samego wyrażenia co zdarzenia problemu; **Recovery expression** - zdarzenia OK są generowane, jeśli wyrażenie problemu przyjmuje wartość FALSE, a wyrażenie odzyskiwania przyjmuje wartość TRUE; **None** - w tym przypadku wyzwalacz nigdy sam nie powróci do stanu OK.
Recovery expression	Logiczne wyrażenie (opcjonalne) definiujące dodatkowe warunki, które muszą zostać spełnione, zanim problem zostanie rozwiązany, po tym jak pierwotne wyrażenie problemu zostało już ocenione jako FALSE. Wyrażenie odzyskiwania jest przydatne dla histerezy wyzwalacza. Nie jest **możliwe** rozwiązanie problemu wyłącznie przez wyrażenie odzyskiwania, jeśli wyrażenie problemu nadal ma wartość TRUE. To pole jest dostępne tylko wtedy, gdy dla *OK event generation* wybrano opcję ’Recovery expression’.
PROBLEM event generation mode	Tryb generowania zdarzeń problemu: **Single** - pojedyncze zdarzenie jest generowane, gdy wyzwalacz po raz pierwszy przejdzie w stan ’Problem’; **Multiple** - zdarzenie jest generowane przy *każdej* ocenie wyzwalacza jako ’Problem’.
OK event closes	Wybierz, czy zdarzenie OK zamyka: **All problems** - wszystkie problemy tego wyzwalacza; **All problems if tag values match** - tylko te problemy wyzwalacza, których wartości tagów zdarzeń są zgodne.
Tag for matching	Wprowadź nazwę tagu zdarzenia, która ma być używana do korelacji zdarzeń. To pole jest wyświetlane, jeśli dla właściwości *OK event closes* wybrano opcję ’All problems if tag values match’ i w takim przypadku jest obowiązkowe.
Allow manual close	Zaznacz, aby zezwolić na ręczne zamykanie zdarzeń problemu generowanych przez ten wyzwalacz. Ręczne zamykanie jest możliwe podczas potwierdzania zdarzeń problemu.

Parameter	Description
Menu entry name	Jeśli pole nie jest puste, wpisana tutaj nazwa (do 64 znaków) jest używana w kilku miejscach frontend jako etykieta dla adresu URL wyzwalacza określonego w parametrze *Menu entry URL*. Jeśli pole jest puste, używana jest domyślna nazwa *Trigger URL*. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a także {EVENT.ID}, {HOST.ID} i {TRIGGER.ID}.
Menu entry URL	Jeśli pole nie jest puste, wpisany tutaj adres URL (do 2048 znaków) jest dostępny jako link w menu zdarzeń w kilku miejscach frontend, na przykład po kliknięciu nazwy problemu w widżecie *Monitoring* > *Problems* lub *Problems*. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a także {EVENT.ID}, {HOST.ID} i {TRIGGER.ID}. Uwaga: makra użytkownika z wartościami tajnymi nie zostaną rozwinięte w adresie URL.
Description	Pole tekstowe służące do podania dodatkowych informacji o tym wyzwalaczu. Może zawierać instrukcje dotyczące naprawy konkretnego problemu, dane kontaktowe odpowiedzialnego personelu itp. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza.
Enabled	Odznaczenie tego pola spowoduje wyłączenie wyzwalacza, jeśli jest to wymagane. Problemy wyłączonego wyzwalacza nie są już wyświetlane we frontend, ale nie są usuwane.

Karta **Tags** umożliwia definiowanie tagów na poziomie wyzwalacza. Wszystkie problemy tego wyzwalacza będą oznaczone wartościami wprowadzonymi tutaj.



Dodatkowo opcja *Inherited and trigger tags* umożliwia wyświetlanie tagów zdefiniowanych na poziomie szablonu, jeśli wyzwalacz pochodzi z tego szablonu. Jeśli istnieje wiele szablonów z tym samym tagiem, tagi te są wyświetlane tylko raz, a nazwy szablonów są oddzielone przecinkami. Wyzwalacz nie ”dziedziczy” i nie wyświetla tagów na poziomie hosta.

Parameter	Description
Name/Value	Ustaw niestandardowe tagi do oznaczania zdarzeń wyzwalacza. Tagi są parą nazwy tagu i wartości. Można użyć tylko nazwy albo połączyć ją z wartością. Wyzwalacz może mieć kilka tagów o tej samej nazwie, ale różnych wartościach. Makra użytkownika, makra użytkownika z kontekstem, makra niskopoziomowego wykrywania oraz funkcje makr functions z `{{ITEM.VALUE}}`, `{{ITEM.LASTVALUE}}` są obsługiwane w tagach zdarzeń. Makra niskopoziomowego wykrywania mogą być używane wewnątrz kontekstu makra. Makro {TRIGGER.ID} jest obsługiwane w wartościach tagów wyzwalacza. Może to być przydatne do identyfikowania wyzwalaczy utworzonych na podstawie prototypów wyzwalaczy i na przykład do wstrzymywania problemów z tych wyzwalaczy podczas konserwacji. Jeśli łączna długość rozwiniętej wartości przekroczy 255, zostanie ona skrócona do 255 znaków. Zobacz wszystkie makra obsługiwane dla tagów zdarzeń. Tagi zdarzeń mogą być używane do korelacji zdarzeń, w warunkach akcji, a także będą widoczne w *Monitoring* > *Problems* lub w widżecie *Problems*.

Karta **Dependencies** zawiera wszystkie zależności wyzwalacza.

Kliknij *Add*, aby dodać nową zależność.

Note:
Możesz także skonfigurować wyzwalacz, otwierając istniejący, klikając przycisk *Clone*, a następnie zapisując go pod inną nazwą.

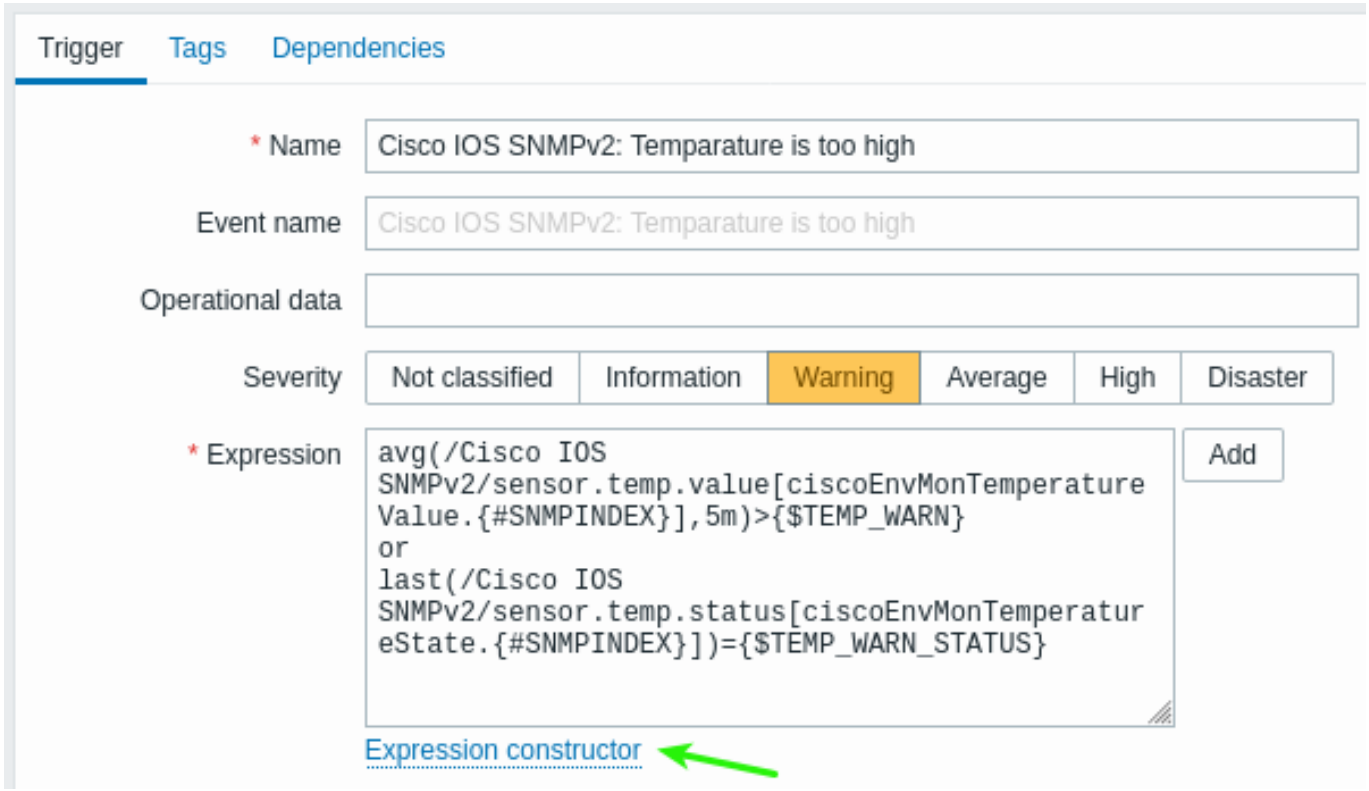
Testowanie wyrażeń

Możliwe jest przetestowanie skonfigurowanego wyrażenia wyzwalacza, aby sprawdzić, jaki byłby wynik wyrażenia w zależności od otrzymanej wartości.

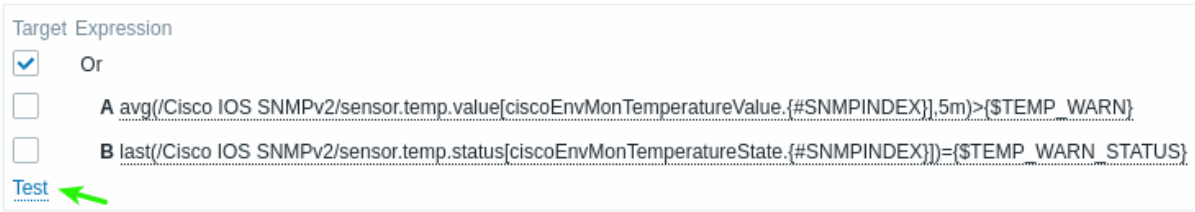
Poniższe wyrażenie z oficjalnego szablonu zostało użyte jako przykład:

```
avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)>{$TEMP_WARN}
or
last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])={$TEMP_WARN_STATUS}
```

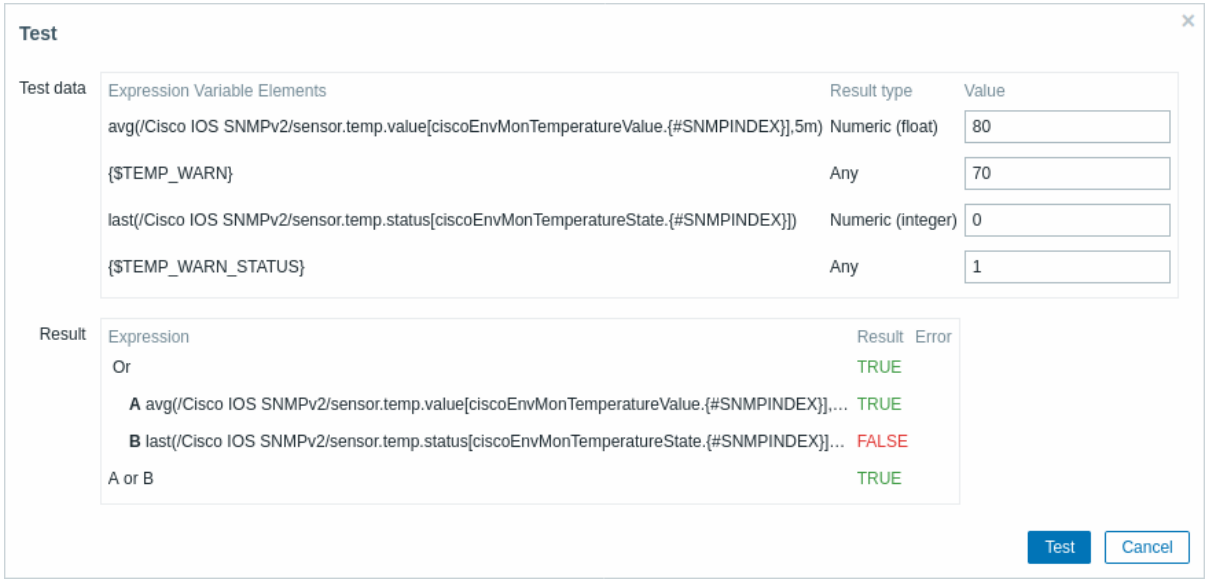
Aby przetestować wyrażenie, kliknij *Expression constructor* pod polem wyrażenia.



W Expression constructor wyświetlane są wszystkie pojedyncze wyrażenia. Aby otworzyć okno testowe, kliknij *Test* pod listą wyrażeń.



W oknie testowym możesz wprowadzić przykładowe wartości (’80’, ’70’, ’0’, ’1’ w tym przykładzie), a następnie zobaczyć wynik wyrażenia, klikając przycisk *Test*.



Widoczny jest wynik poszczególnych wyrażeń, a także całego wyrażenia.

”TRUE” oznacza, że określone wyrażenie jest poprawne. W tym konkretnym przypadku A, ”80” jest większe niż określona wartość {$TEMP_WARN}, czyli ”70” w tym przykładzie. Zgodnie z oczekiwaniami pojawia się wynik ”TRUE”.

”FALSE” oznacza, że określone wyrażenie jest niepoprawne. W tym konkretnym przypadku B, {$TEMP_WARN_STATUS} ”1” musi być równe określonej wartości, czyli ”0” w tym przykładzie. Zgodnie z oczekiwaniami pojawia się wynik ”FALSE”.

Wybrany typ wyrażenia to ”OR”. Jeśli co najmniej jeden z określonych warunków (w tym przypadku A lub B) ma wartość TRUE, wynik końcowy również będzie TRUE. Oznacza to, że bieżąca wartość przekracza wartość ostrzegawczą i wystąpił problem.

2 Wyrażenie wyzwalacza

Przegląd

Wyrażenia używane w wyzwalaczach są bardzo elastyczne. Można ich używać do tworzenia złożonych testów logicznych dotyczących monitorowanych statystyk.

Proste wyrażenie używa **funkcji**, która jest stosowana do pozycji z pewnymi parametrami. Funkcja zwraca wynik, który jest porównywany z progiem za pomocą operatora i stałej.

Składnia prostego, użytecznego wyrażenia to `function(/host/key,parameter)<operator><constant>`.

Na przykład:

```
min(/Zabbix server/net.if.in[eth0,bytes],5m)>100K
```

spowoduje wyzwolenie, jeśli liczba odebranych bajtów w ciągu ostatnich pięciu minut zawsze była większa niż 100 kilobajtów.

Chociaż składnia jest dokładnie taka sama, z funkcjonalnego punktu widzenia istnieją dwa typy wyrażeń wyzwalaczy:

- wyrażenie problemu - definiuje warunki wystąpienia problemu
- wyrażenie odzyskania (opcjonalne) - definiuje dodatkowe warunki rozwiązania problemu

Podczas definiowania wyłącznie wyrażenia problemu, będzie ono używane zarówno jako próg problemu, jak i próg odzyskania. Gdy tylko wyrażenie problemu zostanie ocenione jako TRUE, występuje problem. Gdy tylko wyrażenie problemu zostanie ocenione jako FALSE, problem zostaje rozwiązany.

Podczas definiowania zarówno wyrażenia problemu, jak i dodatkowego wyrażenia odzyskania, rozwiązanie problemu staje się bardziej złożone: nie tylko wyrażenie problemu musi mieć wartość FALSE, ale także wyrażenie odzyskania musi mieć wartość TRUE. Jest to przydatne do tworzenia histerezy i unikania migotania wyzwalacza.

> **Note:**
> Nie ma sensu używać makra {TRIGGER.VALUE} w wyrażeniu odzyskania, ponieważ to wyrażenie jest oceniane tylko wtedy, gdy wyzwalacz znajduje się w stanie ”Problem”. W konsekwencji {TRIGGER.VALUE} zawsze będzie miało wartość ”1” (co oznacza stan ”Problem”) podczas oceny wyrażenia.

Funkcje

Funkcje umożliwiają obliczanie zebranych wartości (średnia, minimum, maksimum, suma), wyszukiwanie ciągów znaków, odwoływanie się do bieżącego czasu oraz innych czynników.

Dostępna jest pełna lista obsługiwanych funkcji.

Zazwyczaj funkcje zwracają wartości liczbowe do porównania. W przypadku zwracania ciągów znaków porównanie jest możliwe za pomocą operatorów **=** i **<>** (zobacz przykład).

Parametry funkcji

Parametry funkcji umożliwiają określenie:

- hosta i klucza pozycji (funkcje odwołujące się wyłącznie do historii pozycji hosta)
- parametrów specyficznych dla danej funkcji
- innych wyrażeń (niedostępne dla funkcji odwołujących się wyłącznie do historii pozycji hosta; zobacz inne wyrażenia, aby poznać przykłady)

Host i klucz pozycji można określić jako `/host/key`.
 Pomijanie nazwy hosta w pierwszym parametrze (tj. w postaci `function(//key,parameter,...)`) jest obsługiwane tylko w niektórych kontekstach:

- W formule obliczanych pozycji
- W makrach wyrażeń, które można stosować w:
  - polu *Nazwa zdarzenia*
  - nazwie wykresu
  - etykiecie elementów mapy "Host" i "Trigger" map elements

W tych kontekstach można również użyć makra `{HOST.HOST}`. `{HOST.HOST<1-9>}` może być użyte w przypadku pola *Nazwa zdarzenia* oraz elementu mapy "Trigger", aby odwołać się do określonej pozycji w wyrażeniu wyzwalacza. Gdy nazwa hosta jest pominięta lub zastąpiona przez `{HOST.HOST}` w tych kontekstach, odwołanie wskazuje na pierwszą pozycję w wyrażeniu wyzwalacza lub na pierwszą pozycję na wykresie. Poza tymi obsługiwanymi kontekstami pominięcie nazwy hosta w wyrażeniach wyzwalacza spowoduje błąd. Zobacz Przykład 18, aby zobaczyć ilustrację użycia podwójnego ukośnika w makrach Nazwy zdarzenia.

Odwoływana pozycja musi znajdować się w obsługiwanym stanie (z wyjątkiem funkcji **nodata()**, która jest obliczana również dla nieobsługiwanych pozycji).

Chociaż inne wyrażenia wyzwalacza jako parametry funkcji są ograniczone do funkcji innych niż historyczne w wyzwalaczach, to ograniczenie nie ma zastosowania w obliczanych pozycjach.

Parametry specyficzne dla funkcji

Parametry specyficzne dla funkcji są umieszczane po kluczu pozycji i oddzielane od klucza pozycji przecinkiem. Pełną listę tych parametrów można znaleźć w sekcji obsługiwane funkcje.

Większość funkcji numerycznych przyjmuje czas jako parametr. Do określenia czasu można użyć sekund lub przyrostków czasu. Jeśli parametr jest poprzedzony znakiem hash, ma on inne znaczenie:

Expression	Description
sum(/host/key,**10m)**	Suma wartości z ostatnich 10 minut.
sum(/host/key,**#10)**	Suma ostatnich dziesięciu wartości.

Parametry ze znakiem hash mają inne znaczenie w przypadku funkcji **last** - oznaczają N-tą poprzednią wartość, więc dla wartości 30, 70, 20, 60, 50 (od najnowszej do najstarszej):

- `last(/host/key,#2)` zwróci '70'
- `last(/host/key,#5)` zwróci '50'

Przesunięcie czasu

Opcjonalne przesunięcie czasu jest obsługiwane przy użyciu czasu lub liczby wartości jako parametru funkcji. Ten parametr umożliwia odwołanie się do danych z okresu w przeszłości.

Przesunięcie czasu zaczyna się od `now` - określenia bieżącego czasu, a następnie występuje `+N<time unit>` lub `-N<time unit>` - aby dodać lub odjąć N jednostek czasu.

Na przykład `avg(/host/key,1h:now-1d)` zwróci średnią wartość dla godziny sprzed jednego dnia.

Attention:
Przesunięcie czasu określone w miesiącach (M) i latach (y) jest obsługiwane tylko przez funkcje trendów. Inne funkcje obsługują sekundy (s), minuty (m), godziny (h), dni (d) i tygodnie (w).

Przesunięcie czasu z bezwzględnymi okresami czasu

Bezwzględne okresy czasu są obsługiwane w parametrze przesunięcia czasu, na przykład od północy do północy dla dnia, od poniedziałku do niedzieli dla tygodnia, od pierwszego do ostatniego dnia miesiąca dla miesiąca.

Przesunięcie czasu dla bezwzględnych okresów czasu zaczyna się od `now` - określenia bieżącego czasu, a następnie występuje dowolna liczba operacji czasu: `/<time unit>` - definiuje początek i koniec jednostki czasu, na przykład od północy do północy dla dnia, `+N<time unit>` lub `-N<time unit>` - aby dodać lub odjąć N jednostek czasu.

Należy pamiętać, że wartość przesunięcia czasu może być większa lub równa 0, podczas gdy minimalna wartość okresu czasu wynosi 1.

Parameter	Description
1d:now/d	Wczoraj
1d:now/d+1d	Dzisiaj
2d:now/d+1d	Ostatnie 2 dni
1w:now/w	Ostatni tydzień
1w:now/w+1w	Ten tydzień

Inne wyrażenia

Parametry funkcji mogą zawierać inne wyrażenia, jak w następującej składni:

```
min(min(/host/key,1h),min(/host2/key2,1h)*10)
```

Należy pamiętać, że innych wyrażeń nie można używać, jeśli funkcja odwołuje się do historii pozycji. Na przykład następująca składnia **nie** jest dozwolona:

~~`min(/host/key,#5*10)`~~

Operatory

Następujące operatory są obsługiwane dla wyzwalaczy **(w kolejności malejącego priorytetu wykonywania)**:

Priorytet	Operator	Definicja	**Uwagi dla nieznanych wartości**	Wymuś rzutowanie operandu na float [1]
1	**-**	Minus unarny	**-**Nieznane → Nieznane	Tak
2	**not**	Logiczne NOT	**not** Nieznane → Nieznane	Tak
3	*	Mnożenie	0 * Nieznane → Nieznane (tak, Nieznane, a nie 0 - aby nie tracić Nieznane w operacjach arytmetycznych) 1.2 * Nieznane → Nieznane	Tak
	/	Dzielenie	Nieznane / 0 → błąd Nieznane / 1.2 → Nieznane 0.0 / Nieznane → Nieznane	Tak
4	**+**	Dodawanie arytmetyczne	1.2 + Nieznane → Nieznane	Tak
	-	Odejmowanie arytmetyczne	1.2 - Nieznane → Nieznane	Tak
5	**<**	Mniejsze niż. Operator jest zdefiniowany jako: A<B ⇔ (A<B-0.000001)	1.2 < Nieznane → Nieznane	Tak

Priorytet	Operator	Definicja	**Uwagi dla nieznanych wartości**	Wymuś rzutowanie operandu na float [1]
	<=	Mniejsze lub równe. Operator jest zdefiniowany jako: A<=B ⇔ (A≤B+0.000001)	Nieznane **<=** Nieznane → Nieznane	Tak
	>	Większe niż. Operator jest zdefiniowany jako: A>B ⇔ (A>B+0.000001)		Tak
	>=	Większe lub równe. Operator jest zdefiniowany jako: A>=B ⇔ (A≥B-0.000001)		Tak
6	**=**	Równe. Operator jest zdefiniowany jako: A=B ⇔ (A≥B-0.000001) and (A≤B+0.000001)		Nie [1]
	<>	Różne. Operator jest zdefiniowany jako: A<>B ⇔ (A<B-0.000001) or (A>B+0.000001)		Nie [1]
7	**and**	Logiczne AND	0 **and** Nieznane → 0 1 **and** Nieznane → Nieznane Nieznane **and** Nieznane → Nieznane	Tak
8	**or**	Logiczne OR	1 **or** Nieznane → 1 0 **or** Nieznane → Nieznane Nieznane **or** Nieznane → Nieznane	Tak

[1] Operand tekstowy nadal jest rzutowany na wartość numeryczną, jeśli:

- drugi operand jest numeryczny
- na operandzie użyto operatora innego niż **=** lub **<>**

W przypadku operatorów relacyjnych (<, <=, >, >=): jeśli rzutowanie się nie powiedzie (na przykład dla ""), wyrażenie zwraca Nieznane (bez alternatywnego porównania jako ciągów znaków).

W przypadku operatorów równości (=, <>): jeśli rzutowanie się nie powiedzie, operandy są porównywane jako surowe ciągi znaków.

Operatory **not**, **and** i **or** są rozróżniane wielkością liter i muszą być zapisane małymi literami. Muszą też być otoczone spacjami lub nawiasami.

Wszystkie operatory, z wyjątkiem unarnego **-** i **not**, mają łączność lewostronną. Unarny **-** i **not** nie są łączne (co oznacza, że należy

używać **-(-1)** i **not (not 1)** zamiast **--1** i **not not 1**).

Wynik ewaluacji:

- operatory **<**, **<=**, **>**, **>=**, **=**, **<>** zwracają w wyrażeniu wyzwalacza wartość '1', jeśli określona relacja jest prawdziwa, oraz '0', jeśli jest fałszywa. Jeśli co najmniej jeden operand jest Nieznane, wynik jest Nieznane;
- **and** dla znanych operandów zwraca '1', jeśli oba operandy są różne od '0'; w przeciwnym razie zwraca '0'; dla nieznanych operandów **and** zwraca '0' tylko wtedy, gdy jeden z operandów jest równy '0'; w przeciwnym razie zwraca 'Nieznane';
- **or** dla znanych operandów zwraca '1', jeśli przynajmniej jeden z operandów jest różny od '0'; w przeciwnym razie zwraca '0'; dla nieznanych operandów **or** zwraca '1' tylko wtedy, gdy jeden z operandów jest różny od '0'; w przeciwnym razie zwraca 'Nieznane';
- Wynik operatora negacji logicznej **not** dla znanego operandu to '0', jeśli wartość operandu jest różna od '0'; '1', jeśli wartość operandu jest równa '0'. Dla nieznanego operandu **not** zwraca 'Nieznane'.

Buforowanie wartości

Wartości wymagane do obliczania wyzwalaczy są buforowane przez serwer Zabbix. Z tego powodu obliczanie wyzwalaczy przez pewien czas po ponownym uruchomieniu serwera powoduje większe obciążenie bazy danych.

Bufor wartości nie jest czyszczony po usunięciu wartości historii pozycja (ręcznie lub przez housekeeper), dlatego serwer będzie używać wartości z pamięci podręcznej, dopóki nie będą one starsze niż okresy czasu zdefiniowane w funkcjach wyzwalaczy lub dopóki serwer nie zostanie ponownie uruchomiony.

> **Note:**
> Jeśli w pamięci podręcznej nie ma aktualnych danych i w funkcji nie zdefiniowano okresu odpytywania, Zabbix domyślnie cofnie się maksymalnie o tydzień, aby odpytać bazę danych o wartości historyczne.

Przykłady wyzwalaczy

Przykład 1

Obciążenie procesora jest zbyt wysokie na serwerze Zabbix.

```
last(/Zabbix server/system.cpu.load[all,avg1])>5
```

Używając funkcji `last()`, odwołujemy się do najnowszej wartości. `/Zabbix server/system.cpu.load[all,avg1]` podaje krótką nazwę monitorowanego parametru. Określa ona, że hostem jest `Zabbix server`, a monitorowanym kluczem jest `system.cpu.load\[all,avg1\]`. Na koniec >5 oznacza, że wyzwalacz znajduje się w stanie PROBLEM zawsze wtedy, gdy najnowszy pomiar obciążenia procesora z serwera Zabbix jest większy niż 5.

Przykład 2

www.example.com jest przeciążony.

```
last(/www.example.com/system.cpu.load[all,avg1])>5 or min(/www.example.com/system.cpu.load[all,avg1],10m)>
```

Wyrażenie jest prawdziwe, gdy bieżące obciążenie procesora jest większe niż 5 lub obciążenie procesora było większe niż 2 w ciągu ostatnich 10 minut.

Przykład 3

/etc/passwd został zmieniony.

```
last(/www.example.com/vfs.file.cksum[/etc/passwd],#1)<>last(/www.example.com/vfs.file.cksum[/etc/passwd],#
```

Wyrażenie jest prawdziwe, gdy poprzednia wartość sumy kontrolnej /etc/passwd różni się od najnowszej.

Podobne wyrażenia mogą być przydatne do monitorowania zmian w ważnych plikach, takich jak /etc/passwd, /etc/inetd.conf, /kernel itd.

Przykład 4

Ktoś pobiera duży plik z Internetu.

Użycie funkcji min:

```
min(/www.example.com/net.if.in[eth0,bytes],5m)>100K
```

Wyrażenie jest prawdziwe, gdy liczba odebranych bajtów na eth0 jest większa niż 100 KB w ciągu ostatnich 5 minut.

Przykład 5

Oba węzły klastrowanego serwera SMTP są niedostępne.

Zwróć uwagę na użycie dwóch różnych hostów w jednym wyrażeniu:

```
last(/smtp1.example.com/net.tcp.service[smtp])=0 and last(/smtp2.example.com/net.tcp.service[smtp])=0
```

Wyrażenie jest prawdziwe, gdy oba serwery SMTP są niedostępne zarówno na smtp1.example.com, jak i smtp2.example.com.

Przykład 6

Agent Zabbix musi zostać zaktualizowany.

Użycie funkcji find():

```
find(/example.example.com/agent.version,,"like","beta8")=1
```

Wyrażenie jest prawdziwe, jeśli agent Zabbix ma wersję beta8.

Przykład 7

Serwer jest nieosiągalny.

```
count(/example.example.com/icmpping,30m,,"0")>5
```

Wyrażenie jest prawdziwe, jeśli host ”example.example.com” jest nieosiągalny więcej niż 5 razy w ciągu ostatnich 30 minut.

Przykład 8

Brak danych przez ostatnie 3 minuty.

Użycie funkcji nodata():

```
nodata(/example.example.com/tick,3m)=1
```

Aby skorzystać z tego wyzwalacza, element ’tick’ musi być zdefiniowany jako pozycja Zabbix trapper. Host powinien okresowo wysyłać dane dla tej pozycji za pomocą zabbix_sender. Jeśli w ciągu 180 sekund nie zostaną odebrane żadne dane, wartość wyzwalacza zmienia się na PROBLEM.

Uwaga Funkcji ’nodata’ można używać dla dowolnego typu pozycji.

Przykład 9

Aktywność CPU w nocy.

Użycie funkcji time():

```
min(/Zabbix server/system.cpu.load[all,avg1],5m)>2 and time()<060000
```

Wyzwalacz może zmienić swój stan na problem tylko w nocy (00:00 - 06:00).

Przykład 10

Aktywność CPU w dowolnym momencie z wyjątkiem.

Użycie funkcji time() i operatora **not**:

```
min(/zabbix/system.cpu.load[all,avg1],5m)>2
and not (dayofweek()=7 and time()>230000)
and not (dayofweek()=1 and time()<010000)
```

Wyzwalacz może zmienić swój stan na problem w dowolnym momencie, z wyjątkiem 2 godzin podczas zmiany tygodnia (niedziela, 23:00 - poniedziałek, 01:00).

Przykład 11

Sprawdź, czy czas lokalny klienta jest zsynchronizowany z czasem serwera Zabbix.

Użycie funkcji fuzzytime():

```
fuzzytime(/MySQL_DB/system.localtime,10s)=0
```

Wyzwalacz przejdzie do stanu problemu w przypadku, gdy czas lokalny na serwerze MySQL_DB i serwerze Zabbix różni się o więcej niż 10 sekund. Należy pamiętać, że `system.localtime` musi być skonfigurowany jako passive check dla agent Zabbix; w agent Zabbix 2 może być skonfigurowany jako check aktywny.

Przykład 12

Porównanie średniego obciążenia dzisiaj ze średnim obciążeniem z tego samego czasu wczoraj (z użyciem przesunięcia czasu jako `now-1d`).

```
avg(/server/system.cpu.load,1h)/avg(/server/system.cpu.load,1h:now-1d)>2
```

Wyzwalacz uruchomi się, jeśli średnie obciążenie z ostatniej godziny przekroczy średnie obciążenie z tej samej godziny wczoraj ponad dwa razy.

Przykład 13

Użycie wartości innej pozycji do uzyskania progu wyzwalacza:

```
last(/Template PfSense/hrStorageFree[{#SNMPVALUE}])<last(/Template PfSense/hrStorageSize[{#SNMPVALUE}])*0.
```

Wyzwalacz zostanie uruchomiony, jeśli wolna przestrzeń dyskowa spadnie poniżej 10 procent.

Przykład 14

Użycie wyniku oceny do uzyskania liczby wyzwalaczy przekraczających próg:

```
(last(/server1/system.cpu.load[all,avg1])>5) + (last(/server2/system.cpu.load[all,avg1])>5) + (last(/serve
```

Wyzwalacz zostanie uruchomiony, jeśli co najmniej dwa z wyzwalaczy w wyrażeniu znajdują się w stanie problemu.

Przykład 15

Porównywanie wartości tekstowych dwóch pozycji - operandami są tutaj funkcje zwracające ciągi znaków.

Problem: utworzyć alert, jeśli wersja Ubuntu jest inna na różnych hostach

```
last(/NY Zabbix server/vfs.file.contents[/etc/os-release])<>last(/LA Zabbix server/vfs.file.contents[/etc/
```

Przykład 16

Porównywanie dwóch wartości ciągów znaków - operandami są:

- funkcja zwracająca ciąg znaków
- kombinacja makr i ciągów znaków

Problem: wykrywanie zmian w zapytaniu DNS

Klucz pozycji to:

```
net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1]
```

z makrami zdefiniowanymi jako

```
{$WEBSITE_NAME} = example.com
{$DNS_RESOURCE_RECORD_TYPE} = MX
```

i zwykle zwraca:

```
example.com             MX       0 mail.example.com
```

Zatem wyrażenie wyzwalacza służące do wykrycia, czy wynik zapytania DNS odbiega od oczekiwanego wyniku, ma postać:

```
last(/Zabbix server/net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1])<>"{$WEBSITE
```

Zwróć uwagę na cudzysłowy wokół drugiego operandu.

Przykład 17

Porównywanie dwóch wartości typu string - operandami są:

- funkcja zwracająca string
- stała string z znakami specjalnymi \ i ”

Problem: wykryć, czy zawartość pliku `/tmp/hello` jest równa:

```
\" //hello ?\"
```

Opcja 1. Zapisz string bezpośrednio:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])="\\\" //hello ?\\\""
```

Zwróć uwagę, jak znaki \ i ” są escape’owane, gdy string jest porównywany bezpośrednio.

Opcja 2. Użyj makra

```
{$HELLO_MACRO} = \" //hello ?\"
```

w wyrażeniu:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])={$HELLO_MACRO}
```

Przykład 18

Porównywanie okresów długoterminowych.

Problem: Obciążenie serwera Exchange wzrosło o ponad 10% w ubiegłym miesiącu

```
trendavg(/Exchange/system.cpu.load,1M:now/M)>1.1*trendavg(/Exchange/system.cpu.load,1M:now/M-1M)
```

Możesz także użyć pola Nazwa zdarzenia w konfiguracji wyzwalacza, aby utworzyć sensowny komunikat alarmowy, na przykład otrzymać coś takiego:

```
"Obciążenie serwera Exchange wzrosło o 24% w lipcu (0.69) w porównaniu z czerwcem (0.56)"
```

nazwa zdarzenia musi być zdefiniowana jako:

```
Obciążenie serwera {HOST.HOST} wzrosło o {{?100*trendavg(//system.cpu.load,1M:now/M)/trendavg(//system.cpu
```

W tego typu problemach przydatne jest również zezwolenie na ręczne zamykanie w konfiguracji wyzwalacza.

Histereza

Czasami potrzebny jest odstęp między stanem problemu a stanem odzyskania, zamiast prostego progu. Na przykład, jeśli chcemy zdefiniować wyzwalacz, który zgłasza problem, gdy temperatura w serwerowni przekroczy 20°C, i chcemy, aby pozostawał on w stanie problemu do momentu, gdy temperatura spadnie poniżej 15°C, sam prosty próg wyzwalacza ustawiony na 20°C nie będzie wystarczający.

Zamiast tego musimy najpierw zdefiniować wyrażenie wyzwalacza dla zdarzenia problemu (temperatura powyżej 20°C). Następnie musimy zdefiniować dodatkowy warunek odzyskania (temperatura poniżej 15°C). Robi się to przez zdefiniowanie dodatkowego parametru *Wyrażenie odzyskania* podczas definiowania wyzwalacza.

W tym przypadku odzyskanie po problemie nastąpi w dwóch krokach:

- Najpierw wyrażenie problemu (temperatura powyżej 20°C) musi zostać ocenione jako FALSE
- Następnie wyrażenie odzyskania (temperatura poniżej 15°C) musi zostać ocenione jako TRUE

Wyrażenie odzyskania będzie oceniane tylko wtedy, gdy zdarzenie problemu zostanie najpierw rozwiązane.

Warning:
Samo to, że wyrażenie odzyskania ma wartość TRUE, nie rozwiązuje problemu, jeśli wyrażenie problemu nadal ma wartość TRUE!

Przykład 1

Temperatura w serwerowni jest zbyt wysoka.

Wyrażenie problemu:

```
last(/server/temp)>20
```

Wyrażenie odzyskania:

```
last(/server/temp)<=15
```

Przykład 2

Wolne miejsce na dysku jest zbyt małe.

Wyrażenie problemu: jest mniejsze niż 10 GB przez ostatnie 5 minut

```
max(/serwer/vfs.fs.size[/,free],5m)<10G
```

Wyrażenie odzyskania: jest większe niż 40 GB przez ostatnie 10 minut

```
min(/serwer/vfs.fs.size[/,free],10m)>40G
```

Wyrażenia z nieznanymi operandami

Ogólnie nieznany operand (na przykład nieobsługiwany item) w wyrażeniu natychmiast spowoduje ustawienie wartości wyzwalacza na `Unknown`.

Jednak w niektórych przypadkach nieznane operandy (nieobsługiwane itemy, błędy funkcji) są dopuszczane do oceny wyrażenia:

- Funkcja `nodata()` jest oceniana niezależnie od tego, czy odwoływany item jest obsługiwany, czy nie.

- Wyrażenia logiczne z OR i AND mogą zostać ocenione do znanych wartości w dwóch przypadkach, niezależnie od nieznanych operandów:
  - **Przypadek 1**: "`1 or some_function(unsupported_item1) or some_function(unsupported_item2) or ...`" może zostać ocenione do znanego wyniku ('1' lub "Problem"),
  - **Przypadek 2**: "`0 and some_function(unsupported_item1) and some_function(unsupported_item2) and ...`" może zostać ocenione do znanego wyniku ('0' lub "OK").
 Zabbix próbuje ocenić takie wyrażenia logiczne, traktując nieobsługiwane itemy jako nieznane operandy. W dwóch powyższych przypadkach zostanie wygenerowana znana wartość ("Problem" lub "OK", odpowiednio); we **wszystkich innych** przypadkach wyzwalacz zostanie oceniony jako `Unknown`.
- Jeśli ocena funkcji dla obsługiwanego itemu zakończy się błędem, wartość funkcji staje się `Unknown` i bierze ona udział jako nieznany operand w dalszej ocenie wyrażenia.

Należy pamiętać, że nieznane operandy mogą "zniknąć" tylko w wyrażeniach logicznych, jak opisano powyżej. W wyrażeniach arytmetycznych nieznane operandy zawsze prowadzą do wyniku `Unknown` (z wyjątkiem dzielenia przez 0).

Attention:
Wyrażenie, które daje wynik `Unknown`, nie zmienia stanu wyzwalacza ("Problem/OK"). Tak więc, jeśli było "Problem" (zobacz Przypadek 1), pozostaje w tym samym stanie problemu nawet wtedy, gdy znana część zostanie rozwiązana ('1' zmienia się na '0'), ponieważ wyrażenie jest teraz oceniane jako `Unknown`, a to nie zmienia stanu wyzwalacza.

Jeśli wyrażenie wyzwalacza z kilkoma nieobsługiwanymi itemami zostanie ocenione jako `Unknown`, komunikat błędu w frontend odnosi się do ostatniego ocenianego nieobsługiwanego itemu.

3 Zależności wyzwalaczy

Omówienie

Czasami dostępność jednego hosta zależy od innego. Serwer znajdujący się za routerem stanie się niedostępny, jeśli router przestanie działać. Przy wyzwalaczach skonfigurowanych dla obu z nich możesz otrzymać powiadomienia o niedostępności dwóch hostów - podczas gdy winny był tylko router.

W takiej sytuacji przydatna może być pewna zależność między hostami. Po ustawieniu zależności można wstrzymać powiadomienia dotyczące zależnych hostów i wysłać tylko powiadomienie o problemie źródłowym.

Chociaż Zabbix nie obsługuje bezpośrednio zależności między hostami, można je zdefiniować za pomocą innej, bardziej elastycznej metody - zależności wyzwalaczy. Wyzwalacz może mieć jeden lub więcej wyzwalaczy, od których zależy.

W naszym prostym przykładzie otwieramy formularz konfiguracji wyzwalacza serwera i ustawiamy, że zależy on od odpowiedniego wyzwalacza routera. Przy takiej zależności wyzwalacz serwera nie zmieni swojego stanu tak długo, jak wyzwalacz, od którego zależy, znajduje się w stanie 'PROBLEM' - a więc nie zostaną podjęte żadne działania zależne i nie zostaną wysłane żadne powiadomienia.

Jeśli zarówno serwer, jak i router są niedostępne, a zależność między wyzwalaczem "server down" i wyzwalaczem "router down" jest ustawiona, Zabbix nie wykona działań dla wyzwalacza zależnego.

Gdy wyzwalacz nadrzędny znajduje się w stanie PROBLEM, jego elementy zależne mogą zwracać wartości, którym nie można ufać. Dlatego wyzwalacze zależne nie będą ponownie oceniane, dopóki wyzwalacz nadrzędny (router w powyższym przykładzie):

- nie wróci ze stanu 'PROBLEM' do stanu 'OK';
- nie zmieni stanu z 'PROBLEM' na 'UNKNOWN';
- nie zostanie zamknięty ręcznie, przez korelację lub z pomocą funkcji date and time i/lub nodata();
- nie zostanie rozwiązany przez wartość pozycja, która nie bierze udziału w wyzwalaczu zależnym;
- nie zostanie wyłączony, nie będzie miał wyłączonej pozycji lub wyłączonego hosta pozycji

We wszystkich przypadkach wymienionych powyżej wyzwalacz zależny (serwer) zostanie ponownie oceniony dopiero po otrzymaniu nowej metryki. Oznacza to, że wyzwalacz zależny może nie zostać zaktualizowany natychmiast.

Ponadto:

- Zależność wyzwalacza można dodać z dowolnego wyzwalacza hosta do dowolnego innego wyzwalacza hosta, o ile nie spowoduje to zależności cyklicznej.
- Zależność wyzwalacza można dodać z jednego szablonu do innego. Jeśli jakiś wyzwalacz z szablonu A zależy od jakiegoś wyzwalacza z szablonu B, szablon A może być połączony z hostem (lub innym szablonem) tylko razem z szablonem B, natomiast szablon B może być połączony z hostem (lub innym szablonem) samodzielnie.
- Zależność wyzwalacza można dodać z wyzwalacza szablonu do wyzwalacza hosta. W takim przypadku połączenie takiego szablonu z hostem utworzy wyzwalacz hosta, który będzie zależał od tego samego wyzwalacza szablonu, od którego zależał wyzwalacz. Pozwala to na przykład mieć szablon, w którym niektóre wyzwalacze zależą od wyzwalaczy routera (hosta). Wszystkie hosty połączone z tym szablonem będą zależeć od tego konkretnego routera.

- Zależności wyzwalacza nie można dodać z wyzwalacza hosta do wyzwalacza szablonu.
- Zależność wyzwalacza można dodać z prototypu wyzwalacza do innego prototypu wyzwalacza (w ramach tej samej reguły niskopoziomowego wykrywania) lub do rzeczywistego wyzwalacza. Prototyp wyzwalacza nie może zależeć od prototypu wyzwalacza z innej reguły LLD ani od wyzwalacza utworzonego z prototypu wyzwalacza. Prototyp wyzwalacza hosta nie może zależeć od wyzwalacza z szablonu.

Konfiguracja

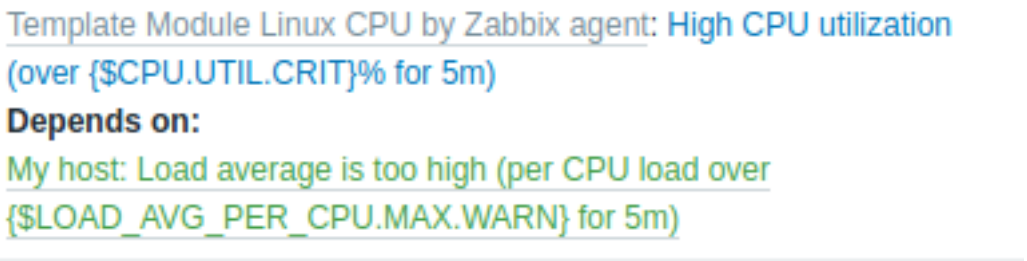
Aby zdefiniować zależność:

1. Otwórz kartę *Dependencies* w formularzu konfiguracji wyzwalacza.
2. Kliknij *Add* w sekcji *Dependencies* i wybierz jeden lub więcej wyzwalaczy, od których będzie zależeć ten wyzwalacz.



3. Kliknij *Update*.

Teraz wyzwalacz ma w liście oznaczenie swojej zależności.



Przykład kilku zależności

Na przykład host znajduje się za Router2, a Router2 znajduje się za Router1.

```
Zabbix - Router1 - Router2 - Host
```

Jeśli Router1 nie działa, to oczywiście host i Router2 są również niedostępne, jednak otrzymywanie trzech powiadomień o tym, że host, Router1 i Router2 nie działają, jest przesadą.

W takim przypadku definiujemy dwie zależności:

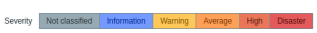
```
wyzwalacz 'Host is down' zależy od wyzwalacza 'Router2 is down'
wyzwalacz 'Router2 is down' zależy od wyzwalacza 'Router1 is down'
```

Przed zmianą stanu wyzwalacza 'Host is down' Zabbix sprawdzi odpowiednie zależności wyzwalaczy. Jeśli takie zależności zostaną znalezione i jeden z tych wyzwalaczy znajduje się w stanie 'Problem', stan wyzwalacza nie zostanie zmieniony, akcje nie zostaną wykonane i nie zostaną wysłane żadne powiadomienia.

Zabbix wykonuje to sprawdzenie rekurencyjnie. Jeśli Router1 lub Router2 są niedostępne, wyzwalacz hosta nie zostanie zaktualizowany.

4 Poziom ważności wyzwalacza

Poziom ważności wyzwalacza reprezentuje poziom istotności wyzwalacza.



Zabbix obsługuje następujące domyślne poziomy ważności wyzwalaczy.

Severity	Color	Description
Not classified	Gray	Może być używany tam, gdzie poziom ważności zdarzenia jest nieznany, nie został określony, nie należy do regularnego zakresu monitorowania itp., na przykład podczas początkowej konfiguracji, jako tymczasowy element do późniejszej oceny lub w ramach procesu integracji.
Information	Light blue	Może być używany dla zdarzeń informacyjnych, które nie wymagają natychmiastowej uwagi, ale nadal mogą dostarczać cennych informacji.
Warning	Yellow	Może być używany do wskazania potencjalnego problemu, który może wymagać analizy lub działania, ale nie jest krytyczny.
Average	Orange	Może być używany do wskazania istotnego problemu, który należy rozwiązać stosunkowo szybko, aby zapobiec dalszym problemom.
High	Light red	Może być używany do wskazania krytycznych problemów, które wymagają natychmiastowej uwagi, aby uniknąć poważnych zakłóceń.
Disaster	Red	Może być używany do wskazania poważnego incydentu, który wymaga natychmiastowego działania, aby zapobiec na przykład awariom systemu lub utracie danych.

Note:
Nazwy i kolory poziomów ważności wyzwalaczy można dostosować.

Poziomy ważności wyzwalaczy są używane do:

- wizualnej reprezentacji wyzwalaczy - różne kolory dla różnych poziomów ważności;
- dźwięku w globalnych alarmach - różne dźwięki dla różnych poziomów ważności;
- mediów użytkownika - różne media (kanał powiadomień) dla różnych poziomów ważności (na przykład SMS dla wyzwalaczy o poziomie ważności *High* i *Disaster* oraz Email dla wyzwalaczy o innych poziomach ważności);
- ograniczania działań na podstawie warunków dotyczących poziomów ważności wyzwalaczy.

5 Dostosowywanie poziomów ważności wyzwalaczy

Nazwy i kolory poziomów ważności wyzwalaczy dla elementów GUI związanych z poziomem ważności można skonfigurować w *Administration → General → Trigger displaying options*. Kolory są współdzielone przez wszystkie motywy GUI.

Tłumaczenie niestandardowych nazw poziomów ważności

Attention:
Jeśli są używane tłumaczenia frontend Zabbix, niestandardowe nazwy poziomów ważności domyślnie zastąpią przetłumaczone nazwy.

Domyślne nazwy poziomów ważności wyzwalaczy są dostępne do tłumaczenia we wszystkich lokalizacjach. Jeśli nazwa poziomu ważności zostanie zmieniona, niestandardowa nazwa jest używana we wszystkich lokalizacjach i wymagane jest dodatkowe ręczne tłumaczenie.

Procedura tłumaczenia niestandardowej nazwy poziomu ważności:

- ustaw wymaganą niestandardową nazwę poziomu ważności, na przykład 'Important'
- edytuj <frontend_dir>/locale/<required_locale>/LC_MESSAGES/frontend.po
- dodaj 2 wiersze:

```
msgid "Important"
msgstr "<translation string>"
```

i zapisz plik.

- utwórz pliki .mo zgodnie z opisem w <frontend_dir>/locale/README

Tutaj **msgid** powinno odpowiadać nowej niestandardowej nazwie poziomu ważności, a **msgstr** powinno być jej tłumaczeniem w danym języku.

Tę procedurę należy wykonać po każdej zmianie nazwy poziomu ważności.

6 Masowa aktualizacja

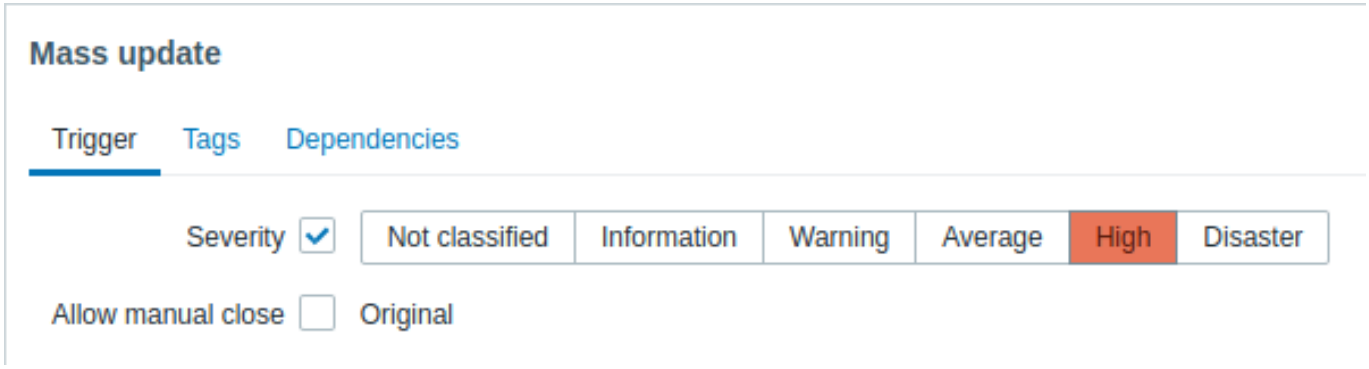
Przegląd

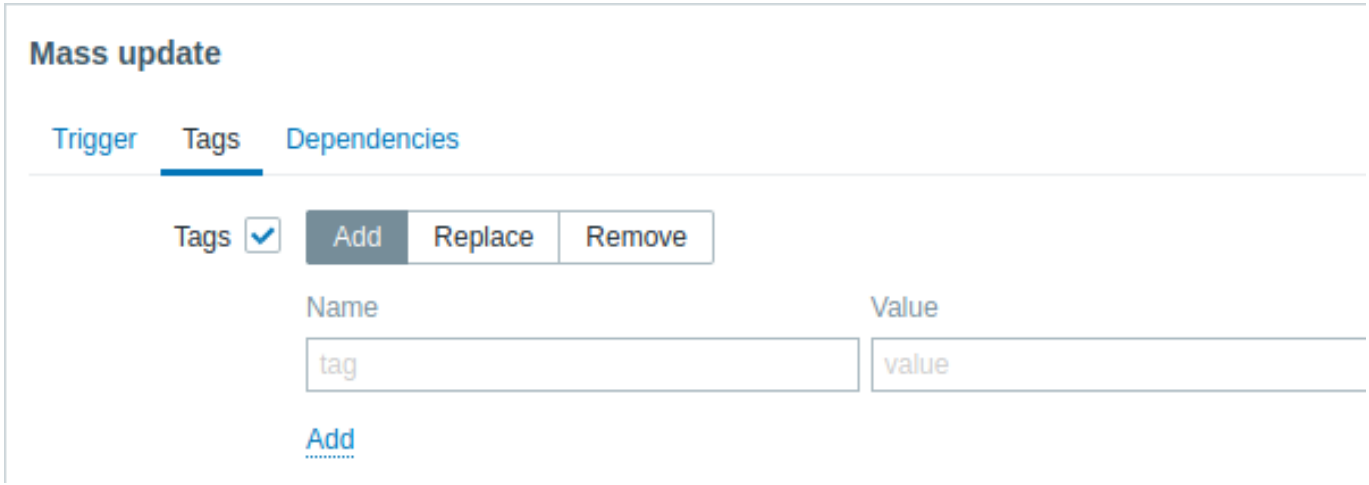
Dzięki masowej aktualizacji możesz zmienić niektóre atrybuty dla wielu wyzwalaczy jednocześnie, oszczędzając sobie konieczności otwierania każdego wyzwalacza osobno w celu edycji.

Using mass update

To mass-update some triggers, do the following:

- Mark the checkboxes of the triggers you want to update in the list
- Click on *Mass update* below the list
- Navigate to the tab with required attributes (*Trigger*, *Tags* or *Dependencies*)
- Mark the checkboxes of any attribute to update

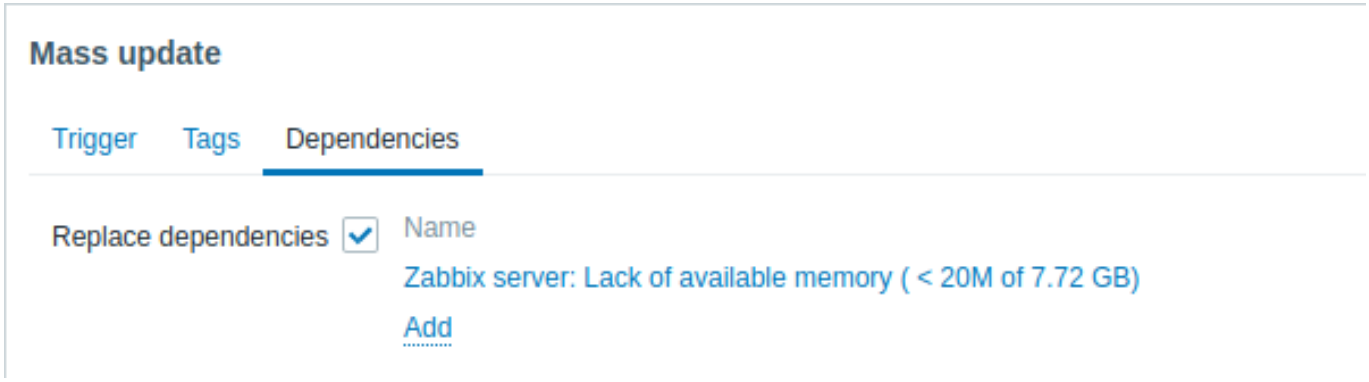




The following options are available when selecting the respective button for tag update:

- *Add* - allows to add new tags for the triggers;
- *Replace* - will remove any existing tags from the trigger and replace them with the one(s) specified below;
- *Remove* - will remove specified tags from triggers.

Note, that tags with the same name, but different values are not considered 'duplicates' and can be added to the same trigger.



Replace dependencies - will remove any existing dependencies from the trigger and replace them with the one(s) specified.

Click on *Update* to apply the changes.

7 Funkcje wyzwalaczy predykcyjnych

Przegląd

Czasami pojawiają się oznaki nadchodzącego problemu. Oznaki te można dostrzec z wyprzedzeniem, aby można było podjąć działania wcześniej i zapobiec problemowi lub przynajmniej zminimalizować jego skutki.

Zabbix oferuje narzędzia do przewidywania przyszłego zachowania monitorowanego systemu na podstawie danych historycznych. Narzędzia te są realizowane za pomocą predykcyjnych funkcji wyzwalacza.

Funkcje

Przed ustawieniem wyzwalacza konieczne jest zdefiniowanie, czym jest stan problemowy i ile czasu potrzeba na podjęcie działania. Następnie istnieją dwa sposoby skonfigurowania wyzwalacza sygnalizującego potencjalnie niepożądaną sytuację. Po pierwsze: wyzwalacz musi zadziałać, gdy oczekuje się, że system będzie w stanie problemowym po upływie „czasu na działanie". Po drugie: wyzwalacz musi zadziałać, gdy system ma osiągnąć stan problemowy za mniej niż „czas na działanie". Odpowiednimi funkcjami wyzwalacza do użycia są **forecast** i **timeleft**. Należy zauważyć, że podstawowa analiza statystyczna jest w obu przypadkach zasadniczo identyczna. Możesz skonfigurować wyzwalacz w dowolny preferowany sposób, uzyskując podobne rezultaty.

Parametry

Obie funkcje używają niemal tego samego zestawu parametrów. Skorzystaj z listy obsługiwanych funkcji jako odniesienia.

Przedział czasu

Przede wszystkim należy określić historyczny okres, który Zabbix ma analizować, aby wygenerować prognozę. Robi się to w znany sposób za pomocą parametru `time period` oraz opcjonalnego przesunięcia czasu, tak jak w przypadku funkcji **avg**, **count**, **delta**, **max**, **min** i **sum**.

Horyzont prognozowania

(tylko `forecast`)
 Parametr `time` określa, jak daleko w przyszłość Zabbix ma ekstrapolować zależności znalezione w danych historycznych. Niezależnie od tego, czy używasz `time_shift`, czy nie, `time` jest zawsze liczony od bieżącego momentu.

Próg do osiągnięcia

(tylko **timeleft**)
 Parametr `threshold` określa wartość, którą analizowana pozycja musi osiągnąć, niezależnie od tego, czy od góry, czy od dołu. Gdy już wyznaczymy f(t) (patrz poniżej), należy rozwiązać równanie f(t) = `threshold` i zwrócić pierwiastek, który jest bliżej chwili obecnej i znajduje się na prawo od niej, albo 1.7976931348623158E+308, jeśli taki pierwiastek nie istnieje.

Note:
Gdy wartości pozycji zbliżają się do progu, a następnie go przekraczają, **timeleft** zakłada, że przecięcie jest już w przeszłości, i dlatego przełącza się na następne przecięcie z poziomem `threshold`, jeśli istnieje. Najlepszą praktyką jest używanie prognoz jako uzupełnienia zwykłej diagnostyki problemów, a nie jako jej zastępstwa.[1]

Funkcje dopasowania

Domyślną funkcją `fit` jest funkcja *liniowa*. Jeśli jednak monitorowany system jest bardziej złożony, masz do wyboru więcej opcji.

`fit`	x = f(t)
linear	$x = a + b*t$
polynomialN[2]	$x = a_0 + a_1*t + a_2*t^2 + ... + a_n*t^n$

`fit`	x = f(t)
exponential	x = a*exp(b*t)
logarithmic	x = a + b*log(t)
power	$x = a*t^b$

Tryby

(Tylko dla **forecast**)
 Za każdym razem, gdy funkcja wyzwalacza jest oceniana, pobiera dane z określonego okresu historii i dopasowuje do tych danych określoną funkcję. Jeśli więc dane są nieco inne, dopasowana funkcja również będzie nieco inna. Jeśli po prostu obliczymy wartość dopasowanej funkcji dla określonego momentu w przyszłości, nie będziesz wiedzieć nic o tym, jak analizowana pozycja ma się zachowywać między teraz a tym momentem w przyszłości. W przypadku niektórych opcji `fit` (na przykład *polynomial*) zwykła wartość z przyszłości może być myląca.

`mode`	wynik **forecast**
value	f(now + `time`)
max	$\max_{\text{now <= t <= now + time}}$ f(t)
min	$\min_{\text{now <= t <= now + time}}$ f(t)
delta	*max - min*
avg	średnia f(t) (now <= t <= now + `time`) zgodnie z definicją

Szczegóły

Aby uniknąć obliczeń z ogromnymi liczbami, za nowy punkt zerowy czasu przyjmujemy znacznik czasu pierwszej wartości w określonym okresie powiększony o 1 ns (bieżący czas epoki ma rząd 10^9, kwadrat czasu epoki to 10^{18}, a precyzja podwójna wynosi około 10^{-16}). Dodanie 1 ns ma na celu zapewnienie wszystkich dodatnich wartości czasu dla dopasowań *logarytmicznych* i *potęgowych*, które obejmują obliczanie log(t). Przesunięcie czasu nie wpływa na dopasowania *liniowe*, *wielomianowe* ani *wykładnicze* (poza łatwiejszymi i dokładniejszymi obliczeniami), ale zmienia kształt funkcji *logarytmicznych* i *potęgowych*.

Potencjalne błędy

Funkcje zwracają -1 w następujących sytuacjach:

- określony okres oceny nie zawiera żadnych danych;
- wynik operacji matematycznej nie jest zdefiniowany[3];
- komplikacje numeryczne (niestety dla niektórych zestawów danych wejściowych zakres i precyzja formatu zmiennoprzecinkowego podwójnej precyzji stają się niewystarczające)[4].

Note:
Nie są zgłaszane żadne ostrzeżenia ani błędy, jeśli wybrana funkcja dopasowania słabo opisuje dostarczone dane lub jeśli danych jest po prostu zbyt mało, aby uzyskać dokładną prognozę.

Przykłady i postępowanie z błędami

Aby otrzymać ostrzeżenie, gdy na Twoim host zaczyna brakować wolnego miejsca na dysku, możesz użyć wyrażenia wyzwalacza takiego jak to:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h
```

Jednak kod błędu -1 może wejść w grę i spowodować, że Twój wyzwalacz przejdzie w stan problemu. Ogólnie jest to dobre, ponieważ otrzymujesz ostrzeżenie, że Twoje prognozy nie działają poprawnie i powinieneś przyjrzeć się im dokładniej, aby ustalić dlaczego. Czasami jednak jest to złe, ponieważ -1 może po prostu oznaczać, że w ciągu ostatniej godziny nie uzyskano żadnych danych o wolnym miejscu na dysku host. Jeśli otrzymujesz zbyt wiele fałszywie dodatnich alertów, rozważ użycie bardziej złożonego wyrażenia wyzwalacza [5]:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1
```

Sytuacja jest nieco trudniejsza w przypadku **forecast**. Przede wszystkim, -1 może, ale nie musi, spowodować przejście wyzwalacza w stan problemu, zależnie od tego, czy masz wyrażenie takie jak `forecast(/host/item,(...))<...` czy takie jak `forecast(/host/item,(...))>...`

Ponadto -1 może być prawidłową prognozą, jeśli ujemna wartość pozycji jest czymś normalnym. Jednak prawdopodobieństwo takiej sytuacji w rzeczywistych warunkach jest znikome (zobacz, jak działa operator =). Jeśli więc chcesz traktować -1 jako problem lub nie chcesz tego robić, dodaj odpowiednio `... or forecast(/host/item,(...))=-1` albo `... and forecast(/host/item,(...))<>-1`.

Przypisy

[1] Na przykład prosty wyzwalacz taki jak `timeleft(/host/item,1h,X) < 1h` może przejść w stan problemu, gdy wartość pozycji zbliża się do X, a następnie nagle wrócić do normalnego stanu, gdy wartość X zostanie osiągnięta. Jeśli problemem jest to, że wartość pozycji jest poniżej X, użyj: `last(/host/item) < X or timeleft(/host/item,1h,X) < 1h` Jeśli problemem jest to, że wartość pozycji jest powyżej X, użyj: `last(/host/item) > X or timeleft(/host/item,1h,X) < 1h`

[2] Stopień wielomianu może wynosić od 1 do 6, *polynomial1* jest równoważny *linear*. Jednak wyższych stopni wielomianów należy używać ostrożnie. Jeśli okres oceny zawiera mniej punktów niż potrzeba do wyznaczenia współczynników wielomianu, stopień wielomianu zostanie obniżony (np. wymagany jest *polynomial5*, ale dostępne są tylko 4 punkty, więc zostanie dopasowany *polynomial3*).

[3] Na przykład dopasowywanie funkcji *exponential* lub *power* obejmuje obliczanie log() wartości pozycji. Jeśli dane zawierają zera lub liczby ujemne, wystąpi błąd, ponieważ log() jest zdefiniowane tylko dla wartości dodatnich.

[4] Dla dopasowań *linear*, *exponential*, *logarithmic* i *power* wszystkie niezbędne obliczenia można zapisać jawnie. Dla *polynomial* bez dodatkowych kroków można obliczyć tylko *value*. Obliczanie *avg* obejmuje wyznaczenie antypochodnej wielomianu (analitycznie). Obliczanie *max*, *min* i *delta* obejmuje wyznaczenie pochodnej wielomianu (analitycznie) i znalezienie jej miejsc zerowych (numerycznie). Rozwiązanie f(t) = 0 polega na znalezieniu miejsc zerowych wielomianu (numerycznie).

[5] Jednak w tym przypadku -1 może spowodować, że Twój wyzwalacz wróci ze stanu problemu do normalnego. Aby być w pełni zabezpieczonym, użyj: `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)`

4 Zdarzenia

Przegląd

W Zabbix generowanych jest kilka typów zdarzeń:

- zdarzenia wyzwalaczy — za każdym razem, gdy wyzwalacz zmienia swój stan (*OK→PROBLEM→OK*)
- zdarzenia usług — za każdym razem, gdy usługa zmienia swój stan (*OK→PROBLEM→OK*)
- zdarzenia wykrywania — gdy wykrywane są hosty lub usługi
- zdarzenia autorejestracji — gdy aktywne agenty są automatycznie rejestrowane przez serwer
- zdarzenia wewnętrzne — gdy pozycja/reguła wykrywania niskiego poziomu staje się nieobsługiwana lub wyzwalacz przechodzi w nieznany stan

Zdarzenia są oznaczane znacznikiem czasu i mogą stanowić podstawę działań, takich jak wysyłanie wiadomości e-mail z powiadomieniami itp.

Aby wyświetlić szczegóły zdarzeń we frontendzie, przejdź do *Monitoring → Problems*. Tam możesz kliknąć datę i godzinę zdarzenia, aby wyświetlić jego szczegóły.

Więcej informacji jest dostępnych na stronach:

- zdarzenia wyzwalaczy
- inne źródła zdarzeń

1 Generowanie zdarzeń wyzwalacza

Przegląd

Zmiana statusu wyzwalacza jest najczęstszym i najważniejszym źródłem zdarzeń. Za każdym razem, gdy wyzwalacz zmienia swój stan, generowane jest zdarzenie. Zdarzenie zawiera szczegóły zmiany stanu wyzwalacza - kiedy nastąpiła i jaki jest nowy stan.

Wyzwalacze tworzą dwa typy zdarzeń - Problem i OK.

Zdarzenia problemów

Zdarzenie problemu jest tworzone:

- gdy wyrażenie wyzwalacza zostanie obliczone jako TRUE, jeśli wyzwalacz jest w stanie OK;
- za każdym razem, gdy wyrażenie wyzwalacza zostanie obliczone jako TRUE, jeśli dla wyzwalacza włączone jest generowanie wielu zdarzeń problemów.

Zdarzenia OK

Zdarzenie OK zamyka powiązane zdarzenie(a) problemu i może zostać utworzone przez 3 komponenty:

- wyzwalacze — na podstawie ustawień „Generowanie zdarzenia OK” i „Zdarzenie OK zamyka”;
- korelację zdarzeń

- menedżera zadań — gdy zdarzenie jest zamykane ręcznie

Wyzwalacze

Wyzwalacze mają ustawienie „Generowanie zdarzenia OK", które określa, w jaki sposób generowane są zdarzenia OK:

- *Wyrażenie* — zdarzenie OK jest generowane dla wyzwalacza w stanie problemu, gdy jego wyrażenie przyjmuje wartość FALSE. Jest to najprostsze ustawienie, domyślnie włączone.
- *Wyrażenie odzyskiwania* — zdarzenie OK jest generowane dla wyzwalacza w stanie problemu, gdy jego wyrażenie przyjmuje wartość FALSE, a wyrażenie odzyskiwania przyjmuje wartość TRUE. Można tego użyć, jeśli kryteria odzyskiwania wyzwalacza różnią się od kryteriów problemu.
- *Brak* — zdarzenie OK nigdy nie jest generowane. Można tego użyć w połączeniu z generowaniem wielu zdarzeń problemu, aby po prostu wysłać powiadomienie, gdy coś się wydarzy.

Dodatkowo wyzwalacze mają ustawienie „Zdarzenie OK zamyka", które określa, które zdarzenia problemu są zamykane:

- *Wszystkie problemy* — zdarzenie OK zamknie wszystkie otwarte problemy utworzone przez wyzwalacz
- *Wszystkie problemy, jeśli wartości tagów są zgodne* — zdarzenie OK zamknie otwarte problemy utworzone przez wyzwalacz i mające co najmniej jedną pasującą wartość tagu. Tag jest definiowany przez ustawienie wyzwalacza „Tag do dopasowania". Jeśli nie ma żadnych zdarzeń problemu do zamknięcia, zdarzenie OK nie jest generowane. Jest to często nazywane korelacją zdarzeń na poziomie wyzwalacza.

Korelacja zdarzeń

Korelacja zdarzeń (nazywana również globalną korelacją zdarzeń) to sposób konfiguracji niestandardowych reguł zamykania zdarzeń (skutkujących generowaniem zdarzenia OK).

Reguły określają, w jaki sposób nowe zdarzenia problemowe są parowane z istniejącymi zdarzeniami problemowymi, i umożliwiają zamknięcie nowego zdarzenia lub dopasowanych zdarzeń przez wygenerowanie odpowiednich zdarzeń OK.

Jednak korelację zdarzeń należy konfigurować bardzo ostrożnie, ponieważ może ona negatywnie wpływać na wydajność przetwarzania zdarzeń lub, w przypadku błędnej konfiguracji, zamknąć więcej zdarzeń, niż zamierzano (w najgorszym przypadku mogą zostać zamknięte nawet wszystkie zdarzenia problemowe). Kilka wskazówek dotyczących konfiguracji:

1. zawsze ograniczaj zakres korelacji przez ustawienie unikalnego tagu dla zdarzenia kontrolnego (zdarzenia, które jest parowane ze starymi zdarzeniami) i użycie warunku korelacji „tag nowego zdarzenia"
2. nie zapomnij dodać warunku opartego na starym zdarzeniu podczas używania operacji „zamknij stare zdarzenie", w przeciwnym razie wszystkie istniejące problemy mogą zostać zamknięte
3. unikaj używania popularnych nazw tagów stosowanych przez różne konfiguracje korelacji

Menedżer zadań

Jeśli dla wyzwalacza włączone jest ustawienie „Allow manual close", możliwe jest ręczne zamykanie zdarzeń problemów wygenerowanych przez ten wyzwalacz. Odbywa się to we frontend podczas aktualizacji problemu. Zdarzenie nie jest zamykane bezpośrednio — zamiast tego tworzona jest pozycja zadania „close event", która wkrótce zostanie obsłużona przez menedżera zadań. Menedżer zadań wygeneruje odpowiednie zdarzenie OK, a zdarzenie problemu zostanie zamknięte.

2 Inne źródła zdarzeń

Zdarzenia usługi

Zdarzenia usługi są generowane tylko wtedy, gdy dla tych zdarzeń włączone są akcje usługi. W takim przypadku każda zmiana stanu usługi tworzy nowe zdarzenie:

- Zdarzenie problemu - gdy stan usługi zmienia się z OK na PROBLEM
- Zdarzenie OK - gdy stan usługi zmienia się z PROBLEM na OK

Zdarzenie zawiera szczegóły zmiany stanu usługi - kiedy nastąpiła i jaki jest nowy stan.

Zdarzenia wykrywania

Zabbix okresowo skanuje zakresy adresów IP zdefiniowane w regułach wykrywania sieci. Częstotliwość sprawdzania można skonfigurować indywidualnie dla każdej reguły. Po wykryciu hosta lub usługi generowane jest zdarzenie wykrywania (lub kilka zdarzeń).

Zabbix generuje następujące zdarzenia:

Zdarzenie	Kiedy jest generowane
Usługa dostępna	Za każdym razem, gdy Zabbix wykryje aktywną usługę.
Usługa niedostępna	Za każdym razem, gdy Zabbix nie może wykryć usługi.
Host dostępny	Jeśli dla adresu IP co najmniej jedna z usług jest w stanie UP.
Host niedostępny	Jeśli wszystkie usługi nie odpowiadają.

Zdarzenie	Kiedy jest generowane
Usługa wykryta	Jeśli usługa powróci po okresie niedostępności lub zostanie wykryta po raz pierwszy.
Usługa utracona	Jeśli usługa zostanie utracona po wcześniejszym stanie dostępności.
Host wykryty	Jeśli host powróci po okresie niedostępności lub zostanie wykryty po raz pierwszy.
Host utracony	Jeśli host zostanie utracony po wcześniejszym stanie dostępności.

Zdarzenia aktywnej autorejestracji agenta

Aktywna autorejestracja agenta tworzy zdarzenia w Zabbixie.

Jeśli jest skonfigurowana, zdarzenie aktywnej autorejestracji agenta jest tworzone, gdy wcześniej nieznany aktywny agent żąda testów lub gdy metadane hosta uległy zmianie. Serwer dodaje nowy automatycznie zarejestrowany host, używając otrzymanego adresu IP i portu agenta.

Więcej informacji można znaleźć na stronie aktywna autorejestracja agenta.

Zdarzenia wewnętrzne

Zdarzenia wewnętrzne występują, gdy:

- pozycja zmienia stan z „normalny” na „nieobsługiwany”
- pozycja zmienia stan z „nieobsługiwany” na „normalny”
- reguła wykrywania niskopoziomowego zmienia stan z „normalny” na „nieobsługiwany”
- reguła wykrywania niskopoziomowego zmienia stan z „nieobsługiwany” na „normalny”
- wyzwalacz zmienia stan z „normalny” na „nieznany”
- wyzwalacz zmienia stan z „nieznany” na „normalny”

Celem wprowadzenia zdarzeń wewnętrznych jest umożliwienie użytkownikom otrzymywania powiadomień, gdy wystąpi dowolne zdarzenie wewnętrzne, na przykład gdy pozycja stanie się nieobsługiwana i przestanie zbierać dane.

Zdarzenia wewnętrzne są tworzone tylko wtedy, gdy włączone są działania wewnętrzne dla tych zdarzeń. Aby zatrzymać generowanie zdarzeń wewnętrznych (na przykład dla pozycji stających się nieobsługiwane), wyłącz wszystkie działania dla zdarzeń wewnętrznych w Alerts → Actions → Internal actions.

Note:
Jeśli działania wewnętrzne są wyłączone, to gdy obiekt znajduje się w stanie „nieobsługiwany”, zdarzenie odzyskania dla tego obiektu nadal zostanie utworzone.
Jeśli działania wewnętrzne są włączone, to gdy obiekt znajduje się w stanie „nieobsługiwany”, zdarzenie odzyskania dla tego obiektu zostanie utworzone, mimo że dla tego obiektu nie zostało utworzone „zdarzenie problemu”.

Zobacz także: Otrzymywanie powiadomień o nieobsługiwanych pozycjach

3 Ręczne zamykanie problemów

Omówienie

Chociaż zazwyczaj zdarzenia problemowe są rozwiązywane automatycznie, gdy status wyzwalacza zmienia się z „Problem” na „OK”, mogą wystąpić przypadki, w których trudno jest określić, czy problem został rozwiązany na podstawie wyrażenia wyzwalacza. W takich przypadkach problem musi zostać rozwiązany ręcznie.

Na przykład *syslog* może zgłaszać, że niektóre parametry jądra wymagają dostrojenia w celu uzyskania optymalnej wydajności. W takim przypadku problem jest zgłaszany administratorom systemu Linux, którzy go naprawiają, a następnie zamykają problem ręcznie.

Problemy można zamykać ręcznie tylko dla wyzwalaczy z włączoną opcją *Allow manual close*.

Gdy problem zostanie „zamknięty ręcznie”, Zabbix generuje nowe wewnętrzne zadanie dla serwera Zabbix. Następnie proces *task manager* wykonuje to zadanie i generuje zdarzenie OK, a tym samym zamyka zdarzenie problemowe.

Ręczne zamknięcie problemu nie oznacza, że powiązany wyzwalacz nigdy więcej nie przejdzie w stan „Problem”. Wyrażenie wyzwalacza jest ponownie oceniane i może ponownie spowodować wystąpienie problemu:

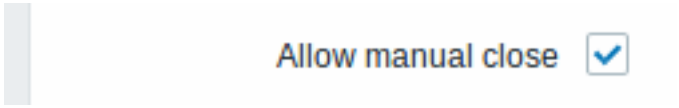
- Gdy pojawią się nowe dane dla dowolnej pozycji uwzględnionej w wyrażeniu wyzwalacza (należy pamiętać, że wartości odrzucone przez krok preprocessingu ograniczającego częstotliwość nie są uznawane za odebrane i nie spowodują ponownej oceny wyrażenia wyzwalacza);
- Gdy w wyrażeniu są używane funkcje date and time i/lub nodata().

Konfiguracja

Do ręcznego zamknięcia problemu wymagane są dwa kroki.

Konfiguracja wyzwalacza

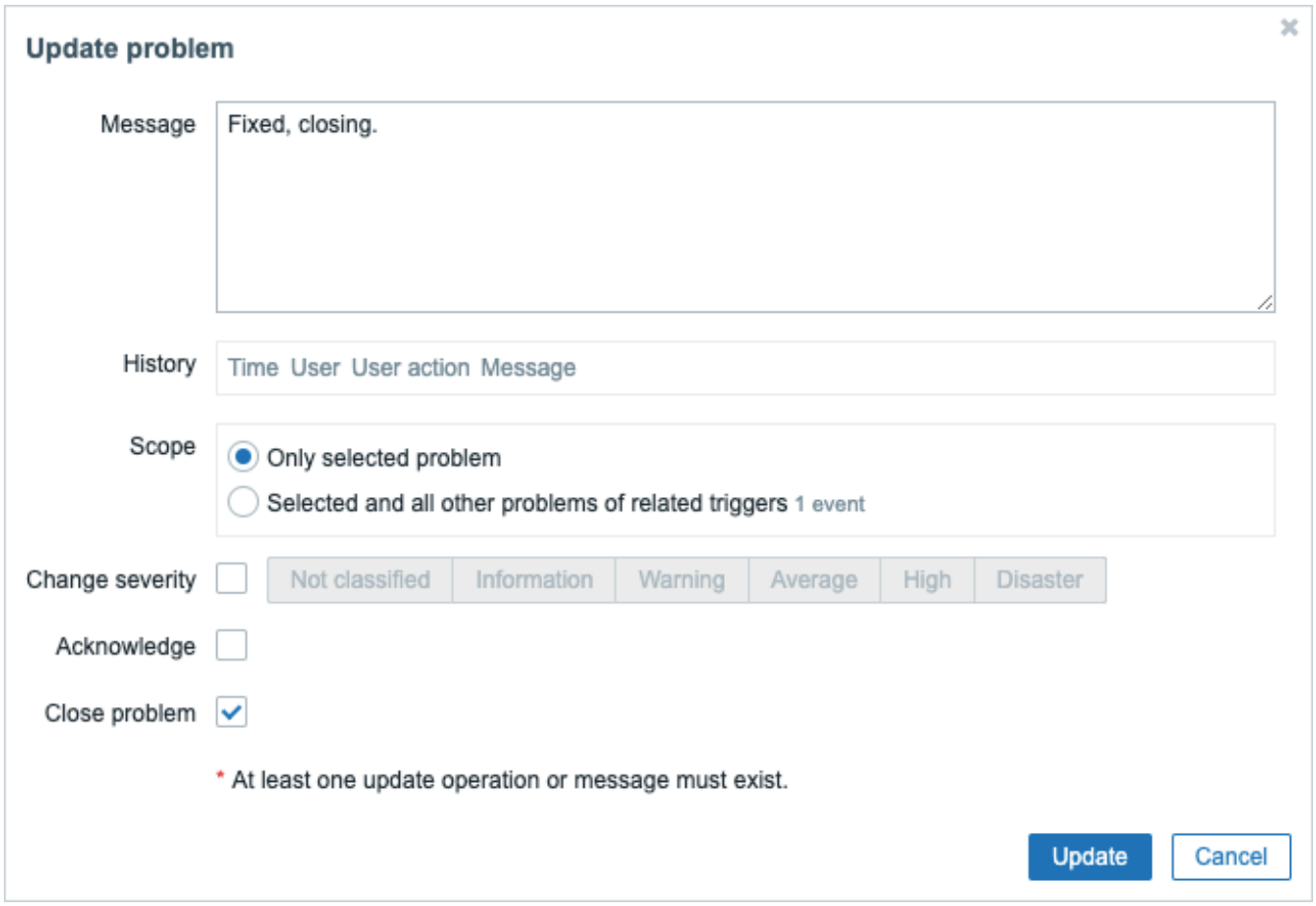
W konfiguracji wyzwalacza włącz opcję *Allow manual close*.



Okno aktualizacji problemu

Jeśli wystąpi problem dla wyzwalacza z flagą *Manual close*, możesz otworzyć okno podręczne aktualizacji problemu tego problemu i zamknąć problem ręcznie.

Aby zamknąć problem, zaznacz opcję *Close problem* w formularzu i kliknij *Update*.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Żądanie jest przetwarzane przez serwer Zabbix. Zwykle zamknięcie problemu zajmuje kilka sekund. W trakcie tego procesu w *Monitoring* → *Problems* jako status problemu wyświetlany jest komunikat *CLOSING*.

Weryfikacja

Można zweryfikować, że problem został zamknięty ręcznie:

- w szczegółach zdarzenia, dostępnych przez *Monitoring* → *Problemy*;
- za pomocą makra {EVENT.UPDATE.HISTORY} w wiadomościach powiadomień, które dostarczy tych informacji.

5 Korelacja zdarzeń

Przegląd

Korelacja zdarzeń umożliwia powiązanie zdarzeń problemowych z ich rozwiązaniem w sposób bardzo precyzyjny i elastyczny.

Korelację zdarzeń można zdefiniować:

- na poziomie wyzwalacza - jeden wyzwalacz może być użyty do powiązania oddzielnych problemów z ich rozwiązaniem
- globalnie - problemy mogą być powiązane z ich rozwiązaniem z innego wyzwalacza/metody odpytywania przy użyciu globalnych reguł korelacji

1 Korelacja zdarzeń oparta na wyzwalaczach

Przegląd

Korelacja zdarzeń oparta na wyzwalaczach pozwala korelować oddzielne problemy zgłaszane przez jeden wyzwalacz.

Chociaż zasadniczo zdarzenie OK może zamknąć wszystkie zdarzenia problemów utworzone przez jeden wyzwalacz, istnieją przypadki, w których potrzebne jest bardziej szczegółowe podejście. Na przykład podczas monitorowania plików dziennika możesz chcieć wykrywać określone problemy w pliku dziennika i zamykać je indywidualnie, a nie wszystkie jednocześnie.

Dotyczy to wyzwalaczy, dla których parametr *PROBLEM event generation mode* jest ustawiony na *Multiple*. Takie wyzwalacze są zwykle używane do monitorowania logów, przetwarzania trapów itp.

W Zabbix możliwe jest powiązanie zdarzeń problemów na podstawie tagów. Tagi są używane do wyodrębniania wartości i tworzenia identyfikacji dla zdarzeń problemów. Dzięki temu problemy mogą być również zamykane indywidualnie na podstawie pasującego tagu.

Innymi słowy, ten sam wyzwalacz może tworzyć oddzielne zdarzenia identyfikowane przez tag zdarzenia. Dlatego zdarzenia problemów mogą być identyfikowane pojedynczo i zamykane oddzielnie na podstawie identyfikacji przez tag zdarzenia.

Jak to działa

Podczas monitorowania logów możesz napotkać wiersze podobne do tych:

```
Wiersz1: Usługa 1 została zatrzymana
Wiersz2: Usługa 2 została zatrzymana
Wiersz3: Usługa 1 została uruchomiona ponownie
Wiersz4: Usługa 2 została uruchomiona ponownie
```

Ideą korelacji zdarzeń jest możliwość dopasowania zdarzenia problemowego z Wiersza1 do rozwiązania z Wiersza3 oraz zdarzenia problemowego z Wiersza2 do rozwiązania z Wiersza4 i zamykania tych problemów jeden po drugim:

```
Wiersz1: Usługa 1 została zatrzymana
Wiersz3: Usługa 1 została uruchomiona ponownie #problem z Wiersza 1 zamknięty

Wiersz2: Usługa 2 została zatrzymana
Wiersz4: Usługa 2 została uruchomiona ponownie #problem z Wiersza 2 zamknięty
```

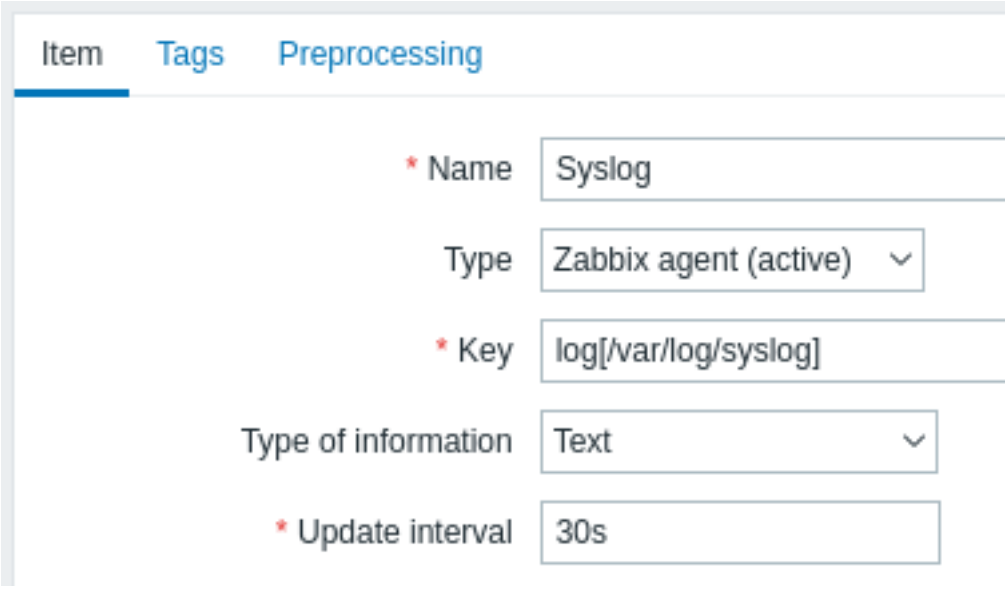
Aby to zrobić, musisz oznaczyć te powiązane zdarzenia tagami, na przykład „Usługa 1” i „Usługa 2”. Można to zrobić, stosując wyrażenie regularne do wiersza logu w celu wyodrębnienia wartości tagu. Następnie, gdy zdarzenia są tworzone, otrzymują odpowiednio tagi „Usługa 1” i „Usługa 2”, a problem może zostać dopasowany do rozwiązania.

Konfiguracja

Pozycja

Na początek możesz skonfigurować pozycję monitorującą plik dziennika, na przykład:

```
log[/var/log/syslog]
```



Po skonfigurowaniu pozycji odczekaj minutę, aż zmiany konfiguracji zostaną zastosowane, a następnie przejdź do Najnowsze dane, aby upewnić się, że pozycja rozpoczęła zbieranie danych.

Wyzwalacz

Gdy pozycja działa, należy skonfigurować wyzwalacz. Ważne jest, aby zdecydować, które wpisy w pliku dziennika są warte uwagi. Na przykład poniższe wyrażenie wyzwalacza będzie wyszukiwać ciąg znaków 'Stopping', aby sygnalizować potencjalne problemy:

```
find(/My host/log[/var/log/syslog],,"regexp","Stopping")=1
```

Attention:
Aby upewnić się, że każda linia zawierająca ciąg "Stopping" jest traktowana jako problem, ustaw również opcję *Tryb generowania zdarzeń problemów* w konfiguracji wyzwalacza na 'Multiple'.

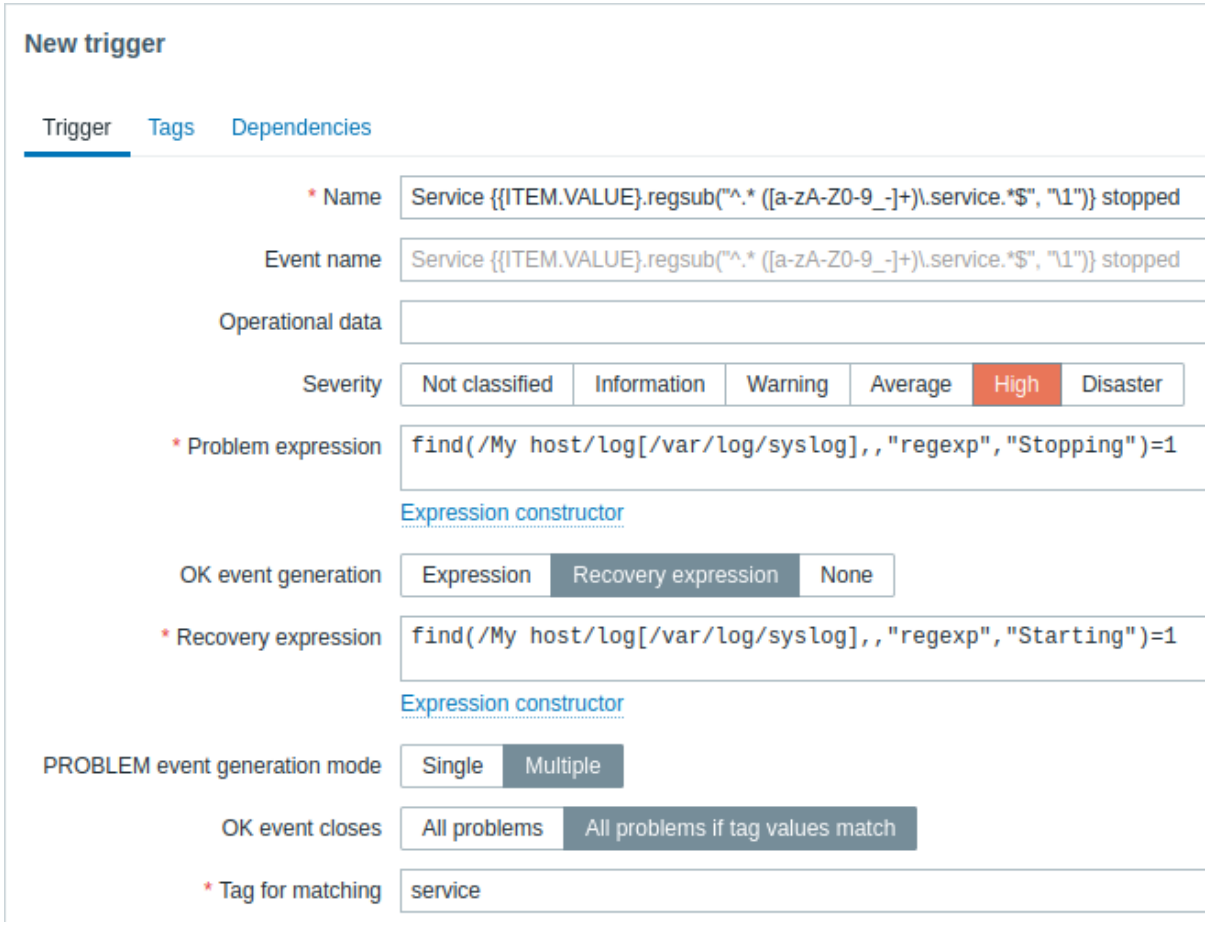
Następnie zdefiniuj wyrażenie odzyskiwania. Poniższe wyrażenie odzyskiwania spowoduje rozwiązanie wszystkich problemów, jeśli zostanie znaleziona linia dziennika zawierająca ciąg "Starting":

```
find(/My host/log[/var/log/syslog],,"regexp","Starting")=1
```

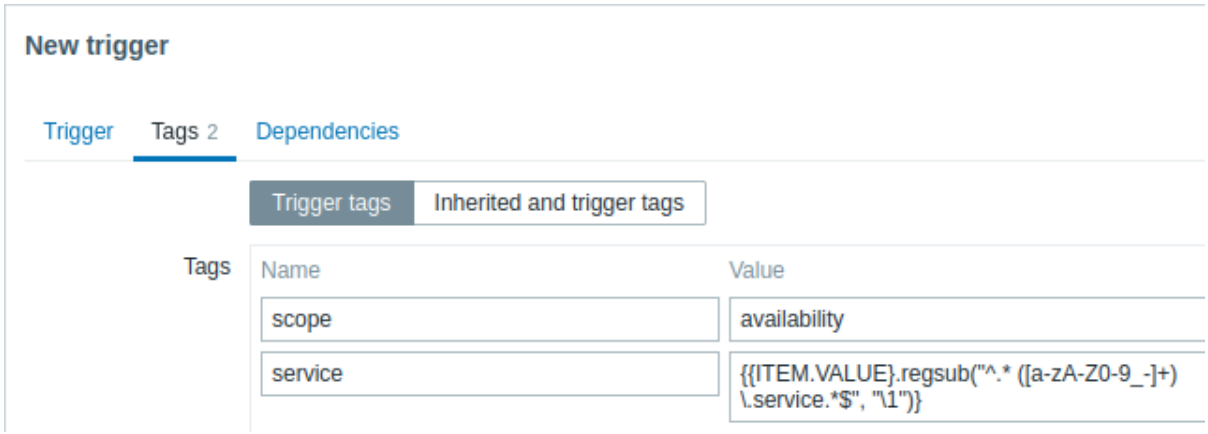
Ponieważ nie tego chcemy, ważne jest, aby w jakiś sposób upewnić się, że zamykane są odpowiadające im główne problemy, a nie po prostu wszystkie problemy. W tym miejscu pomocne może być tagowanie.

Problemy i rozwiązania można dopasować, określając tag w konfiguracji wyzwalacza. Należy wprowadzić następujące ustawienia:

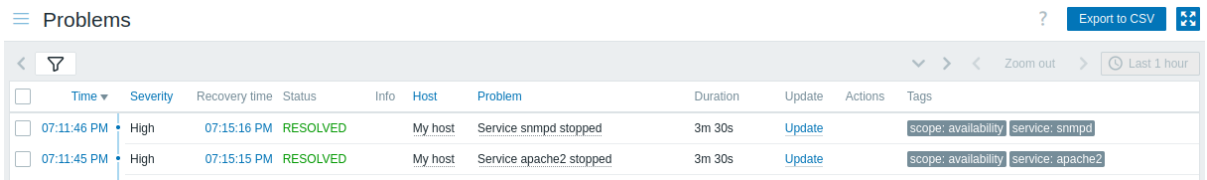
- *Tryb generowania zdarzeń problemów*: Multiple
- *Zdarzenie OK zamyka*: Wszystkie problemy, jeśli wartości tagów są zgodne
- Wprowadź nazwę tagu do dopasowywania zdarzeń



- skonfiguruj tagi, aby wyodrębniać wartości tagów z linii dziennika



Jeśli konfiguracja zakończy się powodzeniem, będzie można zobaczyć zdarzenia problemów otagowane według aplikacji i dopasowane do ich rozwiązania w *Monitoring* → *Problems*.



Warning:
Ponieważ możliwa jest błędna konfiguracja, gdy podobne tagi zdarzeń mogą zostać utworzone dla **niepowiązanych** problemów, zapoznaj się z przypadkami opisanymi poniżej!

- Makra indeksowane zawsze odnoszą się do pola *Expression* w konfiguracji wyzwalacza, a nie do *Recovery expression*. Na przykład w zdarzeniu odzyskiwania **{ITEM.VALUE1}** zostanie rozwinięte do najnowszej wartości pierwszej pozycji **w wyrażeniu problemu** w momencie odzyskiwania. Jeśli wyrażenie odzyskiwania opiera się na innej pozycji, a wartość pozycji z wyrażenia problemu zmieni się do czasu odzyskiwania, zdarzenia otrzymają różne tagi i nie zostaną skorelowane.

- Gdy dwie aplikacje zapisują komunikaty o błędach i odzyskiwaniu do tego samego pliku dziennika, użytkownik może zdecydować się na użycie dwóch tagów *service* w tym samym wyzwalaczu z różnymi wartościami tagów, stosując osobne wyrażenia regularne w wartościach tagów do wyodrębnienia nazw, powiedzmy, usługi A i usługi B z makra {ITEM.VALUE} (np. gdy formaty komunikatów się różnią). Jednak może to nie działać zgodnie z oczekiwaniami, jeśli nie będzie dopasowania do wyrażeń regularnych. Niedopasowane regexpy dadzą puste wartości tagów, a pojedyncza pusta wartość tagu zarówno w zdarzeniach problemów, jak i OK wystarczy do ich skorelowania. Tak więc komunikat odzyskiwania z usługi A może przypadkowo zamknąć komunikat o błędzie z usługi B.
- Rzeczywiste tagi i wartości tagów stają się widoczne dopiero po zadziałaniu wyzwalacza. Jeśli użyte wyrażenie regularne jest nieprawidłowe, zostaje ono po cichu zastąpione ciągiem *UNKNOWN*. Jeśli początkowe zdarzenie problemu z wartością tagu *UNKNOWN* zostanie przeoczone, mogą pojawić się kolejne zdarzenia OK z tą samą wartością tagu *UNKNOWN*, które mogą zamknąć zdarzenia problemów, których nie powinny były zamknąć.
- Jeśli użytkownik używa makra {ITEM.VALUE} bez funkcji makr jako wartości tagu, obowiązuje ograniczenie do 255 znaków. Gdy komunikaty dziennika są długie, a pierwsze 255 znaków nie jest charakterystyczne, może to również skutkować podobnymi tagami zdarzeń dla niepowiązanych problemów.

2 Globalna korelacja zdarzeń

Przegląd

Globalna korelacja zdarzeń pozwala objąć wszystkie metryki monitorowane przez Zabbix i tworzyć korelacje.

Możliwe jest korelowanie zdarzeń utworzonych przez całkowicie różne wyzwalacze i stosowanie do nich wszystkich tych samych operacji. Tworząc inteligentne reguły korelacji, można w praktyce oszczędzić sobie tysięcy powtarzających się powiadomień i skupić się na rzeczywistych przyczynach problemu!

Globalna korelacja zdarzeń to potężny mechanizm, który pozwala uniezależnić się od logiki problemów i ich rozwiązywania opartej na pojedynczym wyzwalaczu. Dotychczas pojedyncze zdarzenie problemu było tworzone przez jeden wyzwalacz i od tego samego wyzwalacza zależało rozwiązanie problemu. Nie mogliśmy rozwiązać problemu utworzonego przez jeden wyzwalacz za pomocą innego wyzwalacza. Jednak dzięki korelacji zdarzeń opartej na tagowaniu zdarzeń jest to możliwe.

Na przykład wyzwalacz logu może zgłaszać problemy aplikacji, podczas gdy wyzwalacz odpytywania może zgłaszać, że aplikacja działa i jest dostępna. Korzystając z tagów zdarzeń, można oznaczyć wyzwalacz logu tagiem *status:down*, a wyzwalacz odpytywania tagiem *status:up*. Następnie w globalnej regule korelacji można powiązać te wyzwalacze i przypisać do tej korelacji odpowiednią operację, taką jak zamykanie starych zdarzeń.

W innym zastosowaniu globalna korelacja może identyfikować podobne wyzwalacze i stosować do nich tę samą operację. Co jeśli moglibyśmy otrzymywać tylko jedno zgłoszenie problemu dla problemu z portem sieciowym? Nie ma potrzeby zgłaszać ich wszystkich. To również jest możliwe dzięki globalnej korelacji zdarzeń.

Globalna korelacja zdarzeń jest konfigurowana w **regułach korelacji**. Reguła korelacji określa, w jaki sposób nowe zdarzenia problemów są łączone z istniejącymi zdarzeniami problemów oraz co zrobić w przypadku dopasowania (zamknąć nowe zdarzenie, zamknąć dopasowane stare zdarzenia przez wygenerowanie odpowiednich zdarzeń OK). Jeśli problem zostanie zamknięty przez globalną korelację, zostanie to zgłoszone w kolumnie *Info* w *Monitoring* > *Problems*.

Konfigurowanie globalnych reguł korelacji jest dostępne wyłącznie dla użytkowników na poziomie Super Admin.

> **Attention:**
> Korelację zdarzeń należy konfigurować bardzo ostrożnie, ponieważ może ona negatywnie wpływać na wydajność przetwarzania zdarzeń lub, w przypadku błędnej konfiguracji, zamknąć więcej zdarzeń, niż zamierzano (w najgorszym przypadku mogą zostać zamknięte nawet wszystkie zdarzenia problemów).

Aby **bezpiecznie** skonfigurować globalną korelację, należy przestrzegać następujących ważnych wskazówek:

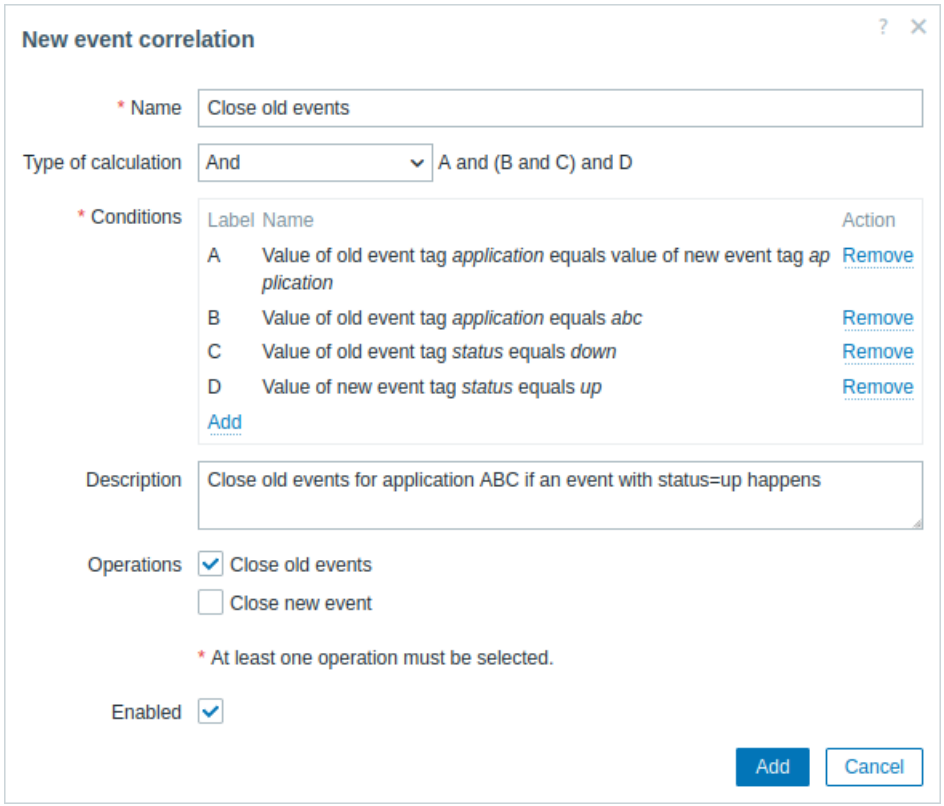
- Ogranicz zakres korelacji. Zawsze ustawiaj unikalny tag dla nowego zdarzenia, które jest łączone ze starymi zdarzeniami, i używaj warunku korelacji *New event tag name*.
- Dodaj jawny warunek starego zdarzenia podczas używania **Close old events**. Zawsze dodawaj co najmniej jeden warunek *Old event* (na przykład *Old event tag name*, *Old event tag value* lub *Event tag pair*) przy wyborze **Close old events** — w przeciwnym razie reguła może dopasować i zamknąć niepowiązane istniejące zdarzenia problemów (w najgorszym przypadku wszystkie problemy). Preferuj **Event tag pair** do dopasowywania wartości środowiska uruchomieniowego (host:port, session id itp.) i w miarę możliwości dodatkowo zawężaj dopasowanie według hosta lub grupy hostów.
- Unikaj używania popularnych nazw tagów, które mogą zostać użyte przez różne konfiguracje korelacji.
- Ogranicz liczbę reguł korelacji do tych, których naprawdę potrzebujesz.

Zobacz także: znane problemy.

Konfiguracja

Aby skonfigurować globalne reguły korelacji zdarzeń:

- Przejdź do *Data collection* > *Event correlation*
- Kliknij *Create event correlation* po prawej stronie (lub nazwę korelacji, aby edytować istniejącą regułę)
- Wprowadź parametry reguły korelacji w formularzu

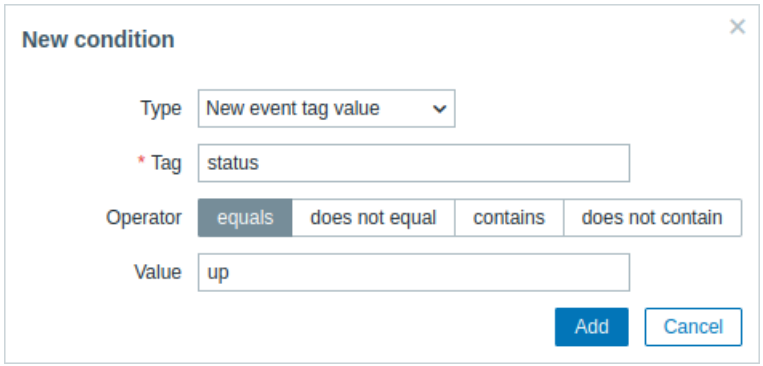


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Unikalna nazwa reguły korelacji.
Type of calculation	Dostępne są następujące opcje obliczania warunków: **And** - wszystkie warunki muszą być spełnione **Or** - wystarczy, że jeden warunek jest spełniony **And/Or** - AND dla różnych typów warunków oraz OR dla tego samego typu warunku **Custom expression** - zdefiniowana przez użytkownika formuła obliczeniowa do oceny warunków akcji. Musi zawierać wszystkie warunki (reprezentowane jako wielkie litery A, B, C, ...) i może zawierać spacje, tabulatory, nawiasy ( ), **and** (z rozróżnianiem wielkości liter), **or** (z rozróżnianiem wielkości liter), **not** (z rozróżnianiem wielkości liter).
Conditions	Lista warunków. Szczegóły dotyczące konfiguracji warunku znajdują się poniżej.
Description	Opis reguły korelacji.
Operations	Zaznacz pole wyboru operacji, która ma zostać wykonana po skorelowaniu zdarzenia. Dostępne są następujące operacje: **Close old events** - zamknij stare zdarzenia, gdy wystąpi nowe zdarzenie. Zawsze dodawaj warunek oparty na starym zdarzeniu podczas używania operacji *Close old events*, w przeciwnym razie wszystkie istniejące problemy mogą zostać zamknięte. **Close new event** - zamknij nowe zdarzenie, gdy wystąpi. **Warning!** Nie pozostawiaj pustych warunków starego/nowego zdarzenia podczas używania *Close old events*/*Close new event*. Jeśli wybierzesz operację *Close old events* bez dodania warunku pasującego do starego zdarzenia, Zabbix może dopasować wszystkie istniejące stare zdarzenia i je zamknąć. Zawsze dodawaj jawny warunek starego zdarzenia (na przykład *Old event tag name* lub *Event tag pair*) podczas używania *Close old events*. Na przykład reguła, która używa tylko *New event condition* oraz operacji *Close old events*, dopasuje wszystkie stare zdarzenia spełniające kryteria starego zdarzenia (których brak) — skutecznie zamykając stare problemy.

Parameter	Description
Enabled	Jeśli zaznaczysz to pole wyboru, reguła korelacji będzie włączona.

Aby skonfigurować szczegóły nowego warunku, kliknij Add w bloku Conditions. Otworzy się wyskakujące okno, w którym można edytować szczegóły warunku.



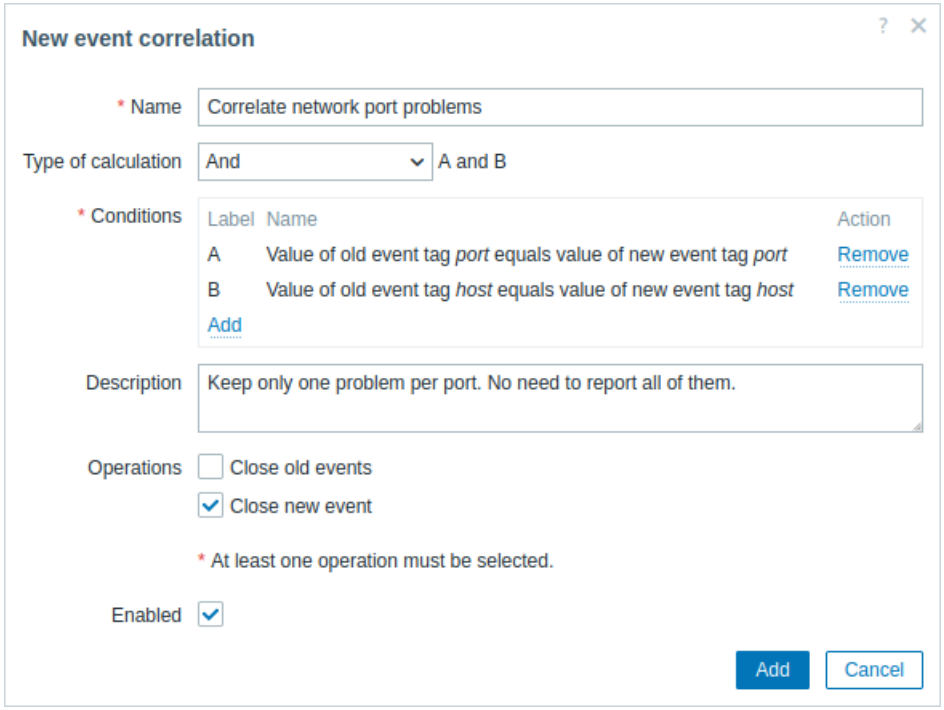
Parameter	Description
New condition	Wybierz warunek do korelacji zdarzeń. *Uwaga:* jeśli nie określono warunku starego zdarzenia, wszystkie stare zdarzenia mogą zostać dopasowane i zamknięte. Podobnie, jeśli nie określono warunku nowego zdarzenia, wszystkie nowe zdarzenia mogą zostać dopasowane i zamknięte. Dostępne są następujące warunki: **Old event tag name** - określ nazwę tagu starego zdarzenia do dopasowania. **New event tag name** - określ nazwę tagu nowego zdarzenia do dopasowania. **New event host group** - określ grupę hostów nowego zdarzenia do dopasowania. **Event tag pair** - określ nazwę tagu nowego zdarzenia i nazwę tagu starego zdarzenia do dopasowania. W takim przypadku dopasowanie nastąpi, jeśli **wartości** tagów w obu zdarzeniach są zgodne. *Nazwy* tagów nie muszą być zgodne. Ta opcja jest przydatna do dopasowywania wartości czasu działania, które mogą nie być znane w momencie konfiguracji (zobacz także Przykład). **Old event tag value** - określ nazwę i wartość tagu starego zdarzenia do dopasowania, używając następujących operatorów: *equals* - ma wartość tagu starego zdarzenia *does not equal* - nie ma wartości tagu starego zdarzenia *contains* - zawiera ciąg w wartości tagu starego zdarzenia *does not contain* - nie zawiera ciągu w wartości tagu starego zdarzenia **New event tag value** - określ nazwę i wartość tagu nowego zdarzenia do dopasowania, używając następujących operatorów: *equals* - ma wartość tagu nowego zdarzenia *does not equal* - nie ma wartości tagu nowego zdarzenia *contains* - zawiera ciąg w wartości tagu nowego zdarzenia *does not contain* - nie zawiera ciągu w wartości tagu nowego zdarzenia

Warning:
Ponieważ możliwa jest błędna konfiguracja, gdy podobne tagi zdarzeń mogą być tworzone dla **niepowiązanych** problemów, zapoznaj się z przypadkami opisanymi poniżej!

- Rzeczywiste tagi i wartości tagów stają się widoczne dopiero po uruchomieniu wyzwalacza. Jeśli użyte wyrażenie regularne jest nieprawidłowe, jest ono po cichu zastępowane ciągiem *UNKNOWN*. Jeśli początkowe zdarzenie problemu z wartością tagu *UNKNOWN* zostanie pominięte, mogą pojawić się kolejne zdarzenia OK z tą samą wartością tagu *UNKNOWN*, które mogą zamknąć zdarzenia problemów, których nie powinny były zamknąć.
- Jeśli użytkownik używa makra {ITEM.VALUE} bez funkcji makr jako wartości tagu, obowiązuje ograniczenie do 255 znaków. Gdy komunikaty logów są długie, a pierwsze 255 znaków jest niespecyficzne, może to również skutkować podobnymi tagami zdarzeń dla niepowiązanych problemów.

Przykład

Zatrzymaj powtarzające się zdarzenia problemów z tego samego portu sieciowego.



Ta globalna reguła korelacji będzie korelować problemy, jeśli wartości tagów *host* i *port* istnieją w wyzwalaczu i są takie same w oryginalnym oraz nowym zdarzeniu.

Operacja zamknie nowe zdarzenia problemów na tym samym porcie sieciowym, pozostawiając otwarty tylko oryginalny problem.

6 Tagowanie

Przegląd

Tagi składają się z nazwy tagu i wartości tagu. Podczas tagowania encji można użyć tylko nazwy albo połączyć ją z wartością (na przykład `mysql`, `jira`, `target:mysql`, `service:jira` itd.).

Tagi można definiować dla różnych encji:

- Szablony
- Hosty
- Pozycje
- Scenariusze web
- Wyzwalacze
- Usługi
- Pozycje i wyzwalacze szablonu
- Prototypy hosta, pozycji i wyzwalacza

Note:
Zapoznaj się z oficjalnymi wytycznymi Zabbix dotyczącymi ogólnych zaleceń dotyczących definiowania tagów, a także ze szczegółowymi wskazówkami dla szablonów, pozycji, wyzwalaczy oraz reguł wykrywania niskiego poziomu.

Tagi mają wiele zastosowań, przede wszystkim do oznaczania zdarzeń. Gdy encje są otagowane, każde nowe zdarzenie związane z otagowaną encją odziedziczy jej tagi. Na przykład:

- w przypadku otagowanych szablonów - każdy problem hosta (utworzony przez wyzwalacze z szablonu) odziedziczy tagi szablonu;
- w przypadku otagowanych hostów - każdy problem hosta odziedziczy tagi hosta;
- w przypadku otagowanych pozycji/scenariuszy web - każdy problem pozycji/scenariusza web odziedziczy tagi pozycji/scenariusza web;
- w przypadku otagowanych wyzwalaczy - każdy problem utworzony przez wyzwalacz odziedziczy tagi wyzwalacza.

Zdarzenie problemu dziedziczy wszystkie tagi z całego łańcucha encji - szablonów, hostów, pozycji/scenariuszy web, wyzwalaczy. Identyczne kombinacje `tag:value` (po rozwiązaniu makr) są scalane w jedną, co pozwala uniknąć duplikacji.

Niestandardowe tagi zdarzeń oferują większą elastyczność. Na przykład:

- korelację zdarzeń można skonfigurować na podstawie tagów zdarzeń;
- warunki akcji można skonfigurować na podstawie tagów zdarzeń;
- problemy pozycji można grupować na podstawie tagów zdarzeń;
- tagi problemów można wykorzystać do mapowania problemów na usługi.

Encje mogą być otagowane tą samą nazwą tagu, ale z różnymi wartościami tagu (na przykład `component:memory` i `component:storage`). Podobnie encja może mieć tag bez wartości oraz ten sam tag z wartością (na przykład `database` i `database:postgresql`). Takie tagi nie są uznawane za duplikaty.

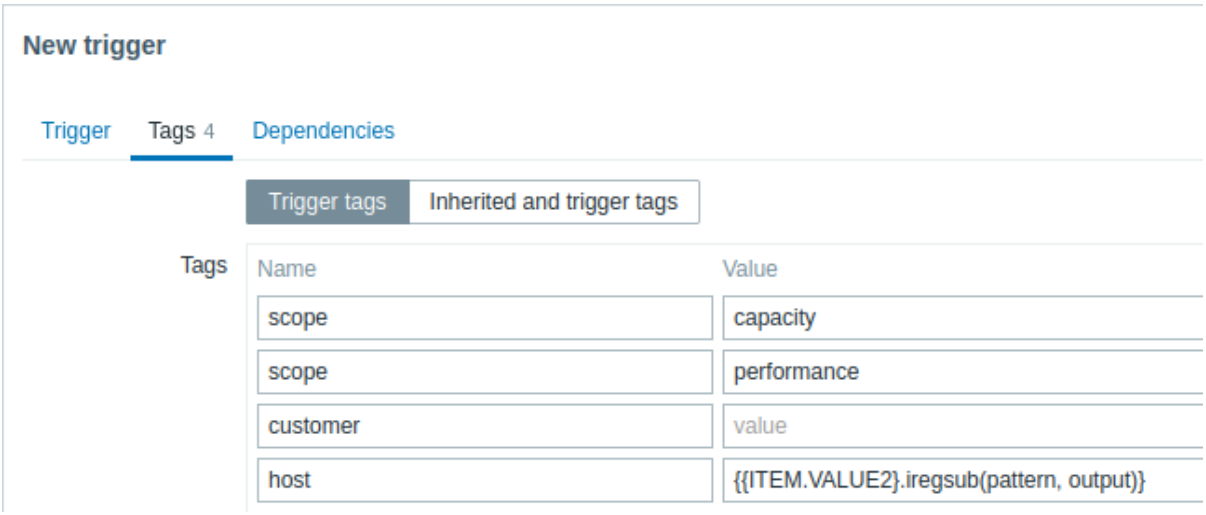
Przypadki użycia

Oto kilka typowych przypadków użycia tagowania:

1. Oznaczanie zdarzeń wyzwalacza:
   - Zdefiniuj tag wyzwalacza (na przykład `scope:performance`).
   - Problemy utworzone przez ten wyzwalacz będą miały tag wyzwalacza.
2. Oznaczanie problemów odziedziczonych z szablonu:
   - Zdefiniuj tag szablonu (na przykład `target:mysql`).
   - Problemy utworzone przez wyzwalacze z tego szablonu będą miały tag szablonu.
3. Oznaczanie problemów hosta:
   - Zdefiniuj tag hosta (na przykład `service:jira`).
   - Problemy utworzone przez wyzwalacze z tego hosta będą miały tag hosta.
4. Filtrowanie powiązanych pozycji:
   - Zdefiniuj tag pozycji (na przykład `component:cpu`).
   - W *Monitoring* → *Latest data* pozycje można filtrować według tagu `component:cpu`.
5. Używanie informacji wyodrębnionych z wartości pozycji jako wartości tagu:
   - Zdefiniuj tag z makrem jako wartością tagu (na przykład `tag-name:{{ITEM.VALUE<N>}.regsub()}`).
   - W *Monitoring* → *Problems* problemy będą miały wartość tagu rozwiązaną do danych wyodrębnionych z wartości pozycji.
6. Identyfikowanie problemów w pliku dziennika i zamykanie ich osobno:
   - Zdefiniuj tag wyzwalacza dla wyzwalacza pozycji monitorującej logi, który wyodrębni wartości z wartości pozycji za pomocą makra (na przykład `service:{{ITEM.VALUE<N>}.regsub()}`).
   - W konfiguracji wyzwalacza skonfiguruj korelację zdarzeń:
     - ustaw *PROBLEM event generation mode* na "Multiple";
     - ustaw *OK event closes* na "All problems if tag values match";
     - ustaw tag do dopasowania.
   - Problemy utworzone przez wyzwalacz pozycji logów będą miały tag wyzwalacza i będą zamykane indywidualnie.
7. Filtrowanie powiadomień:
   - Zdefiniuj tagi wyzwalacza (na przykład `scope:security` dla trigger1 i `scope:availability` dla trigger2).
   - Użyj filtrowania tagów w warunkach akcji, aby otrzymywać powiadomienia tylko o zdarzeniach zgodnych z danymi tagów.
8. Identyfikowanie problemów w powiadomieniach:
   - Zdefiniuj tagi wyzwalacza.
   - Użyj makra {EVENT.TAGS} w powiadomieniu o problemie.
   - Powiadomienie o problemie będzie zawierać tagi wyzwalacza, co ułatwi identyfikację, do której aplikacji/usługi należy powiadomienie.
9. Uproszczenie zadań konfiguracyjnych przez użycie tagów szablonu:
   - Zdefiniuj tag wyzwalacza szablonu.
   - Wyzwalacze utworzone z tego wyzwalacza szablonu będą miały jego tag.
10. Tworzenie wyzwalaczy z tagami z niskopoziomowego wykrywania (LLD):
    - Zdefiniuj tag prototypu wyzwalacza z makrem LLD w nazwie lub wartości tagu (na przykład `scope:{#FSNAME}`).
    - Wyzwalacze utworzone z prototypu wyzwalacza będą miały jego tag.
11. Dopasowywanie usług za pomocą tagów usług:
    - Zdefiniuj tagi usług.
    - Skonfiguruj akcje usług dla usług z pasującymi tagami.
    - Dodatkowo użyj tagów usług do powiązania usługi z SLA na potrzeby obliczeń SLA.
12. Łączenie usług z problemami za pomocą tagów problemów usług:
    - Zdefiniuj tag problemu w konfiguracji usługi (na przykład `target:mysql`).
    - Problemy z pasującym tagiem zostaną automatycznie skorelowane z usługą, a stan usługi zmieni się zgodnie ze skonfigurowanymi regułami obliczania stanu usługi.
13. Wstrzymywanie problemów, gdy host jest w trybie konserwacji:
    - Zdefiniuj tagi w konfiguracji okresu konserwacji.
    - Problemy z zdefiniowanymi tagami zostaną wstrzymane.
14. Przyznawanie dostępu grupom użytkowników:
    - Zdefiniuj tagi w konfiguracji grupy użytkowników.
    - Użytkownicy w grupie użytkowników będą mogli wyświetlać tylko problemy z zdefiniowanymi tagami.

Konfiguracja

Tagi można zdefiniować w dedykowanej karcie, na przykład w konfiguracji wyzwalacza:



Obsługa makr

Wbudowane i makra użytkownika w tagach są rozwijane w momencie wystąpienia zdarzenia. Do czasu wystąpienia zdarzenia makra te będą wyświetlane w Zabbix frontend jako nierozwinięte.

Makra niskopoziomowego wykrywania są rozwijane podczas procesu wykrywania.

Następujące makra mogą być używane w nazwach i wartościach tagów wyzwalacza:

- wbudowane makra {ITEM.VALUE}, {ITEM.VALUE.AGE}, {ITEM.VALUE.DATE}, {ITEM.VALUE.TIME}, {ITEM.VALUE.TIMESTAMP}, {ITEM.LASTVALUE}, {ITEM.LASTVALUE.AGE}, {ITEM.LASTVALUE.DATE}, {ITEM.LASTVALUE.TIME}, {ITEM.LASTVALUE.TIMESTAMP}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID}
- wbudowane makra {INVENTORY.*} (do odwoływania się do wartości inwentarza hosta z jednego lub kilku hostów w wyrażeniu wyzwalacza)
- makra użytkownika oraz makra użytkownika z kontekstem (kontekst może zawierać makra niskopoziomowego wykrywania)
- makra niskopoziomowego wykrywania (tylko w tagach prototypu wyzwalacza)

Następujące makra mogą być używane w nazwach i wartościach tagów szablonu, hosta oraz pozycji/scenariusza WWW:

- wbudowane makra {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID}
- wbudowane makra {INVENTORY.*}
- makra użytkownika
- makra niskopoziomowego wykrywania (tylko w tagach prototypu hosta i pozycji)

Następujące makra mogą być używane w powiadomieniach opartych na wyzwalaczach:

- wbudowane makra {EVENT.TAGS} oraz {EVENT.RECOVERY.TAGS} (te makra zostaną rozwinięte do listy tagów zdarzenia lub tagów zdarzenia odzyskania rozdzielonej przecinkami)
- wbudowane makra {EVENT.TAGSJSON} oraz {EVENT.RECOVERY.TAGSJSON} (te makra zostaną rozwinięte do tablicy JSON zawierającej obiekty tagów zdarzenia lub obiekty tagów zdarzenia odzyskania)

Wyodrębnianie podciągu w tagach wyzwalacza

Wyodrębnianie podciągu jest obsługiwane podczas wypełniania nazwy tagu lub wartości tagu przy użyciu makra funkcji. Funkcja stosuje wyrażenie regularne do wartości uzyskanej przez obsługiwane makro. Na przykład:

```
{{ITEM.VALUE}.regsub(pattern, output)}
{{ITEM.VALUE}.iregsub(pattern, output)}

{{#LLDMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.iregsub(pattern, output)}
```

Jeśli nazwa tagu lub jego wartość przekroczy 255 znaków po rozwinięciu makr, zostanie skrócona do 255 znaków.

Zobacz także: Używanie funkcji makr w makrach niskopoziomowego wykrywania do tagowania zdarzeń.

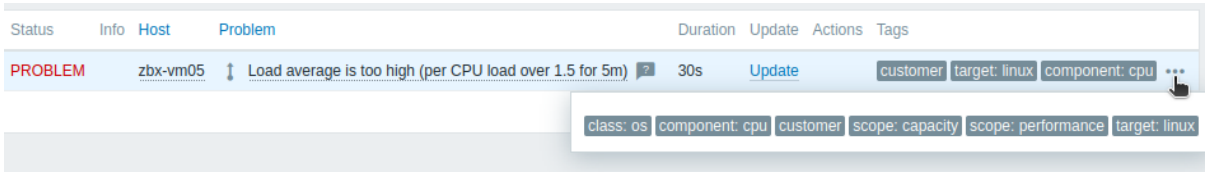
Wyświetlanie tagów zdarzeń

Tagi, jeśli są zdefiniowane, można wyświetlać wraz z nowymi zdarzeniami w:

- *Monitoring* → *Problemy*

- *Monitoring → Problemy → Szczegóły zdarzenia*
- *Pulpity → Widżet Problemy*

Kolejność i liczba wyświetlanych tagów są określane przez opcje filtrowania *Priorytet wyświetlania tagów* oraz *Pokaż tagi* w *Monitoring → Problemy* lub w widżecie pulpitu *Problemy*. Należy pamiętać, że można wyświetlić maksymalnie trzy tagi; jeśli tagów jest więcej, najechanie kursorem na trzy kropki spowoduje wyświetlenie wszystkich tagów w oknie podręcznym.



7 Wizualizacja

Przegląd

Wizualizacja zapewnia różne sposoby wyświetlania danych monitoringu zbieranych przez Zabbix. Wykresy pokazują trendy i metryki, mapy sieci wyświetlają topologię infrastruktury, a pulpity łączą konfigurowalne widgety, aby zapewnić kompleksowe widoki monitoringu.

Narzędzia wizualizacji

- Wykresy — wyświetlaj dane pozycji jako proste, automatycznie generowane wykresy, twórz własne wykresy lub porównuj wiele pozycji na wykresach tworzonych ad hoc.
- Mapy sieci — twórz wizualne reprezentacje infrastruktury sieciowej z konfigurowalnymi elementami i wskaźnikami.
- Pulpity — dostosowuj układy, łącząc różne widżety pulpitu, które wizualizują dane w różnych formatach.

1 Wykresy

Przegląd

Przy dużej ilości danych napływających do Zabbix użytkownikom znacznie łatwiej jest korzystać z wizualnej reprezentacji tego, co się dzieje, zamiast opierać się wyłącznie na liczbach.

W tym miejscu pojawiają się wykresy. Wykresy pozwalają na pierwszy rzut oka zrozumieć przepływ danych, powiązać problemy, ustalić, kiedy coś się zaczęło, lub przedstawić, kiedy coś może przerodzić się w problem.

Zabbix zapewnia użytkownikom:

- wbudowane proste wykresy dla danych jednej pozycji
- możliwość tworzenia bardziej złożonych niestandardowych wykresów
- szybki dostęp do porównania kilku pozycji w wykresach ad hoc
- nowoczesne, konfigurowalne wektorowe wykresy i wykresy kołowe

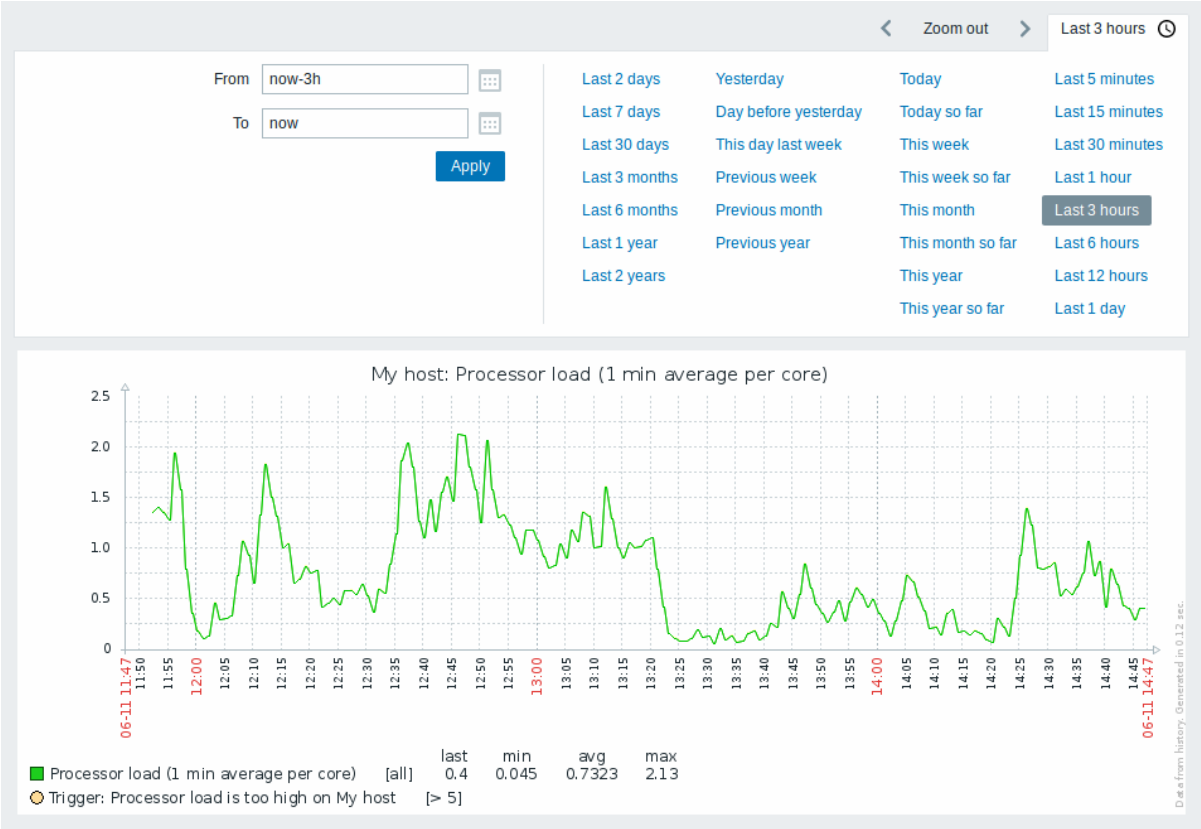
1 Proste wykresy

Przegląd

Proste wykresy służą do wizualizacji danych zbieranych przez pozycje.

Wyświetlanie prostych wykresów nie wymaga od użytkownika żadnej konfiguracji. Są one udostępniane przez Zabbix automatycznie.

Wystarczy przejść do *Monitoring* > *Latest data* i kliknąć link Graph dla odpowiedniej pozycji, a wykres zostanie wyświetlony.



Note:
Proste wykresy są dostępne dla wszystkich pozycji numerycznych. Dla pozycji tekstowych w *Monitoring* > *Latest data* dostępny jest link do historii.

Selektor okresu czasu

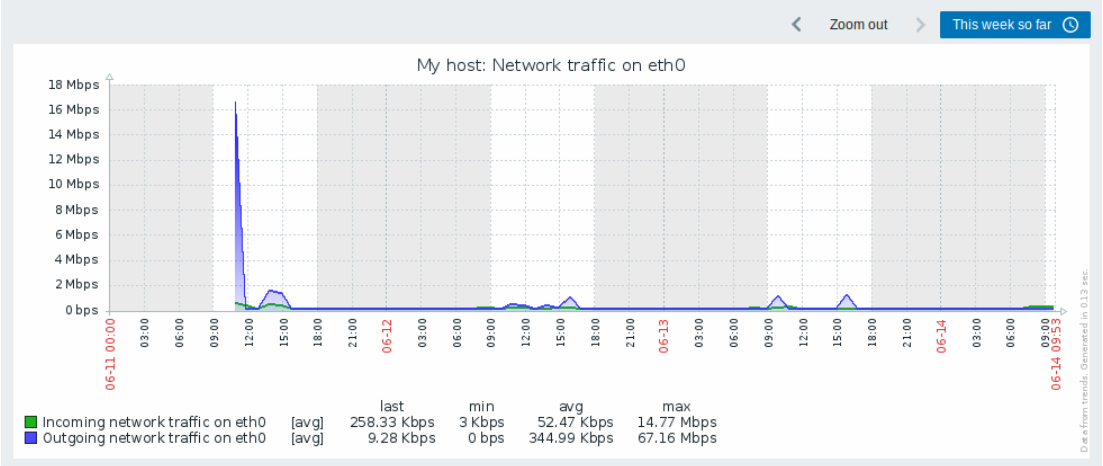
Selektor *Okres czasu* nad wykresem umożliwia wybranie często używanych okresów jednym kliknięciem myszy. Więcej informacji można znaleźć w sekcji *Selektor okresu czasu*.

Ostatnie dane a dłuższe okresy

Dla bardzo ostatnich danych rysowana jest **pojedyncza** linia łącząca każdą otrzymaną wartość. Pojedyncza linia jest rysowana tak długo, jak długo dostępny jest co najmniej jeden poziomy piksel dla jednej wartości.

Dla danych pokazujących dłuższy okres rysowane są **trzy linie** — ciemnozielona pokazuje średnią, natomiast jasnoróżowa i jasnozielona linia pokazują wartości maksymalne i minimalne w danym momencie. Przestrzeń między wartościami wysokimi i niskimi jest wypełniona żółtym tłem.

Czas pracy (dni robocze) jest wyświetlany na wykresach jako białe tło, natomiast czas wolny od pracy jest wyświetlany na szaro (w domyślnym motywie frontend *Original blue*).



Czas pracy jest zawsze wyświetlany na prostych wykresach, natomiast jego wyświetlanie na wykresach niestandardowych zależy od preferencji użytkownika.

Czas pracy nie jest wyświetlany, jeśli wykres pokazuje okres dłuższy niż 3 miesiące.

Linie wyzwalaczy

Proste wyzwalacze są wyświetlane jako linie z czarnymi kreskami na kolorze ważności wyzwalacza — zwróć uwagę na niebieską linię na wykresie oraz informacje o wyzwalaczu wyświetlane w legendzie. Na wykresie mogą być wyświetlane maksymalnie 3 linie wyzwalaczy; jeśli wyzwalaczy jest więcej, priorytet mają wyzwalacze o niższej ważności. Wyzwalacze są zawsze wyświetlane na prostych wykresach, natomiast ich wyświetlanie na wykresach niestandardowych zależy od preferencji użytkownika.



Generowanie z historii/trendów

Wykresy mogą być rysowane na podstawie historii lub trendów pozycji.

Dla użytkowników, którzy mają włączony tryb debugowania frontendu, w prawym dolnym rogu wykresu wyświetlany jest szary, pionowy podpis wskazujący, skąd pochodzą dane.

Na to, czy używana jest historia, czy trendy, wpływa kilka czynników:

- okres przechowywania historii pozycji. Na przykład historia pozycji może być przechowywana przez 14 dni. W takim przypadku wszelkie dane starsze niż czternaście dni będą pochodzić z trendów.
- zagęszczenie danych na wykresie. Jeśli liczba sekund przypadających na poziomy piksel wykresu przekracza 3600/16, wyświetlane są dane trendów (nawet jeśli historia pozycji jest nadal dostępna dla tego samego okresu).
- jeśli trendy są wyłączone, do budowania wykresu używana jest historia pozycji — jeśli jest dostępna dla tego okresu.

Brak danych

W przypadku pozycji z regularnym interwałem aktualizacji na wykresie nic nie jest wyświetlane, jeśli dane pozycji nie są zbierane.

Jednak w przypadku pozycji trapper oraz pozycji z harmonogramowanym interwałem aktualizacji (i regularnym interwałem aktualizacji ustawionym na 0) rysowana jest prosta linia prowadząca do pierwszej zebranej wartości oraz od ostatniej zebranej wartości do końca wykresu; linia znajduje się odpowiednio na poziomie pierwszej/ostatniej wartości.

Przełączanie na surowe wartości

Lista rozwijana w prawym górnym rogu umożliwia przełączenie z prostego wykresu na listy *Wartości/500 najnowszych wartości*. Może to być przydatne do wyświetlania wartości liczbowych składających się na wykres.

Przedstawione tutaj wartości są surowe, tj. nie są używane żadne jednostki ani przetwarzanie końcowe wartości. Mapowanie wartości jest jednak stosowane.

Znane problemy

Zobacz znane problemy dotyczące wykresów.

2 Własne wykresy

Przegląd

Niestandardowe wykresy, jak sama nazwa wskazuje, oferują możliwości dostosowywania.

Podczas gdy proste wykresy dobrze nadają się do wyświetlania danych pojedynczej pozycji, nie oferują możliwości konfiguracji.

Dlatego jeśli chcesz zmienić styl wykresu lub sposób wyświetlania linii albo porównać kilka pozycji, na przykład ruch przychodzący i wychodzący na jednym wykresie, potrzebujesz niestandardowego wykresu.

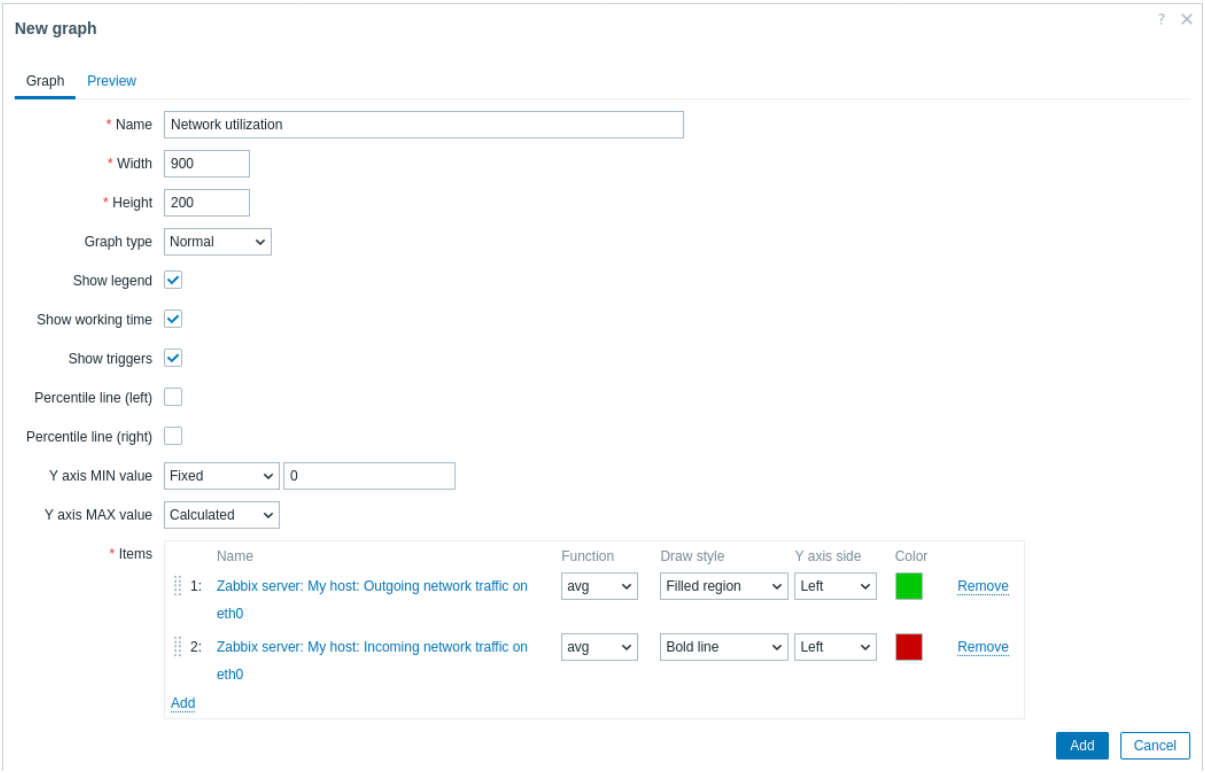
Niestandardowe wykresy są konfigurowane ręcznie.

Można je tworzyć dla jednego hosta lub kilku hostów albo dla pojedynczego szablonu.

Konfigurowanie własnych wykresów

Aby utworzyć własny wykres, wykonaj następujące czynności:

- Przejdź do *Data collection* > *Hosts (lub Templates)*
- Kliknij *Graphs* w wierszu obok wybranego hosta lub szablonu
- Na ekranie Graphs kliknij *Create graph*
- Edytuj atrybuty wykresu



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Atrybuty wykresu:

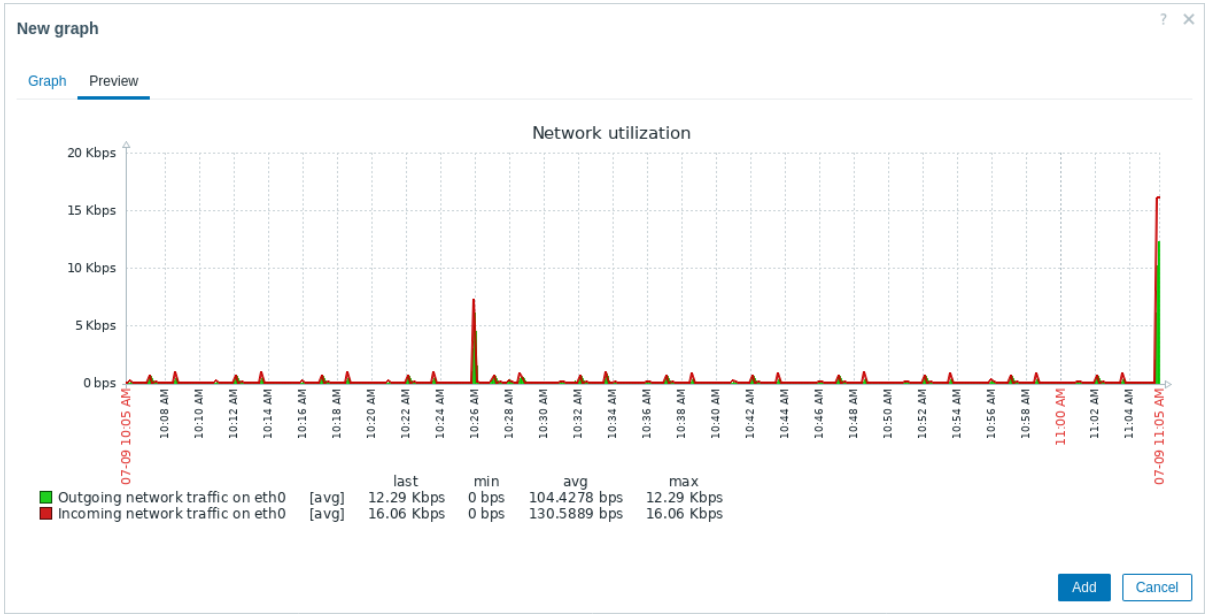
Parametr	Opis
Name	Unikalna nazwa wykresu. W tym polu obsługiwane są wyrażenia makr, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`). Makra {HOST.HOST<1-9>} są obsługiwane do użycia wewnątrz tego makra, odwołując się do pierwszego, drugiego, trzeciego itd. hosta na wykresie, na przykład `{?avg(/{HOST.HOST2}/key,1h)}`. Zauważ, że odwoływanie się do pierwszego hosta za pomocą tego makra jest zbędne, ponieważ do pierwszego hosta można odwołać się niejawnie, na przykład `{?avg(//key,1h)}`.
Width	Szerokość wykresu w pikselach (tylko dla podglądu oraz wykresów kołowych/rozsuniętych).
Height	Wysokość wykresu w pikselach.

Parametr	Opis
Graph type	Typ wykresu: **Normal** — zwykły wykres, wartości wyświetlane jako linie **Stacked** — wykres skumulowany, wyświetlane są wypełnione obszary **Pie** — wykres kołowy **Exploded** — „rozsunięty” wykres kołowy, segmenty są wyświetlane jako „wycięte” z wykresu
Show legend	Zaznaczenie tego pola spowoduje wyświetlenie legendy wykresu.
Show working time	Jeśli wybrane, godziny poza czasem pracy będą pokazane z szarym tłem. Ten parametr nie jest dostępny dla wykresów kołowych i rozsuniętych wykresów kołowych.
Show triggers	Jeśli wybrane, proste wyzwalacze będą wyświetlane jako linie z czarnymi kreskami na kolorze ważności wyzwalacza. Ten parametr nie jest dostępny dla wykresów kołowych i rozsuniętych wykresów kołowych.
Percentile line (left)	Wyświetl percentyl dla lewej osi Y. Jeśli na przykład ustawiony jest 95. percentyl, linia percentyla będzie na poziomie, poniżej którego znajduje się 95 procent wartości. Wyświetlana jako jasnozielona linia. Dostępne tylko dla zwykłych wykresów.
Percentile line (right)	Wyświetl percentyl dla prawej osi Y. Jeśli na przykład ustawiony jest 95. percentyl, linia percentyla będzie na poziomie, poniżej którego znajduje się 95 procent wartości. Wyświetlana jako jasnoczerwona linia. Dostępne tylko dla zwykłych wykresów.
Y axis MIN value	Minimalna wartość osi Y: **Calculated** — minimalna wartość osi Y zostanie obliczona automatycznie. **Fixed** — stała minimalna wartość osi Y. **Item** — ostatnia wartość wybranej pozycji będzie wartością minimalną. Ten parametr nie jest dostępny dla wykresów kołowych i rozsuniętych wykresów kołowych.
Y axis MAX value	Maksymalna wartość osi Y: **Calculated** — maksymalna wartość osi Y zostanie obliczona automatycznie. **Fixed** — stała maksymalna wartość osi Y. **Item** — ostatnia wartość wybranej pozycji będzie wartością maksymalną Ten parametr nie jest dostępny dla wykresów kołowych i rozsuniętych wykresów kołowych.
3D view	Włącz styl 3D. Tylko dla wykresów kołowych i rozsuniętych wykresów kołowych.
Items	Pozycje, których dane mają być wyświetlane na tym wykresie. Kliknij *Add*, aby wybrać pozycje. Możesz także wybrać różne opcje wyświetlania (funkcja, styl rysowania, wyświetlanie lewej/prawej osi, kolor).
Sort order (0→100)	Kolejność rysowania. 0 będzie przetwarzane jako pierwsze. Może być użyte do rysowania linii lub obszarów za (lub przed) innym elementem. Możesz przeciągać i upuszczać pozycje za pomocą ikony na początku wiersza, aby ustawić kolejność sortowania lub określić, która pozycja jest wyświetlana przed inną.
Name	Nazwa wybranej pozycji jest wyświetlana jako link. Kliknięcie linku otwiera listę innych dostępnych pozycji.
Type	Typ (dostępny tylko dla wykresów kołowych i rozsuniętych wykresów kołowych): **Simple** — wartość pozycji jest przedstawiana proporcjonalnie na wykresie kołowym **Graph sum** — wartość pozycji reprezentuje cały wykres kołowy Zauważ, że kolorowanie pozycji „graph sum” będzie widoczne tylko w takim zakresie, w jakim nie jest zajęte przez pozycje „proportional”.
Function	Wybierz, jakie wartości będą wyświetlane, gdy dla pozycji istnieje więcej niż jedna wartość na pionowy piksel wykresu: **all** — wyświetl wszystkie możliwe wartości (minimalną, maksymalną, średnią) na wykresie. Zauważ, że dla krótszych okresów to ustawienie nie ma wpływu; dopiero dla dłuższych okresów, gdy zagęszczenie danych w pionowym pikselu wykresu wzrasta, opcja „all” zaczyna wyświetlać wartości minimalne, maksymalne i średnie. Ta funkcja jest dostępna tylko dla typu wykresu *Normal*. Zobacz także: Generowanie wykresów z historii/trendów. **avg** — wyświetl wartości średnie **last** — wyświetl najnowsze wartości. Ta funkcja jest dostępna tylko wtedy, gdy jako typ wykresu wybrano *Pie/Exploded pie*. **max** — wyświetl wartości maksymalne **min** — wyświetl wartości minimalne

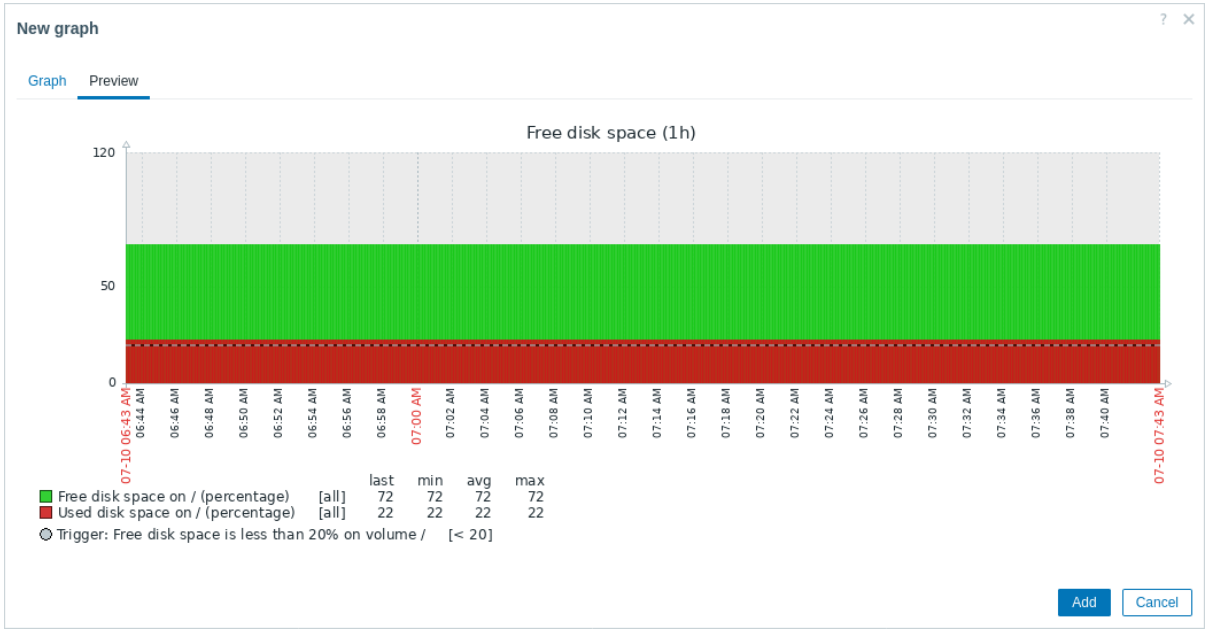
Parametr	Opis
Draw style	Wybierz styl rysowania (dostępny tylko dla zwykłych wykresów; dla wykresów skumulowanych zawsze używany jest wypełniony obszar), który ma zostać zastosowany do danych pozycji — *Line*, *Bold line*, *Filled region*, *Dot*, *Dashed line*, *Gradient line*.
Y axis side	Wybierz stronę osi Y, po której mają być pokazywane dane pozycji — *Left*, *Right*.
Color	Wybierz kolor, który ma zostać zastosowany do danych pozycji.

Podgląd wykresu

Na karcie *Podgląd* wyświetlany jest podgląd wykresu, dzięki czemu możesz od razu zobaczyć, co tworzysz.



Pamiętaj, że podgląd nie będzie wyświetlał żadnych danych dla pozycji szablonu.



W tym przykładzie zwróć uwagę na pogrubioną linię przerywaną przedstawiającą poziom wyzwalacza oraz informacje o wyzwalaczu wyświetlane w legendzie.

Note:
Można wyświetlić nie więcej niż 3 linie wyzwalaczy. Jeśli wyzwalaczy jest więcej, do wyświetlenia priorytetowo wybierane są wyzwalacze o niższym poziomie ważności.

 Jeśli wysokość wykresu jest ustawiona na mniej niż 120 pikseli, w legendzie nie będzie wyświetlany żaden wyzwalacz.

3 Wykresy ad hoc

Przegląd

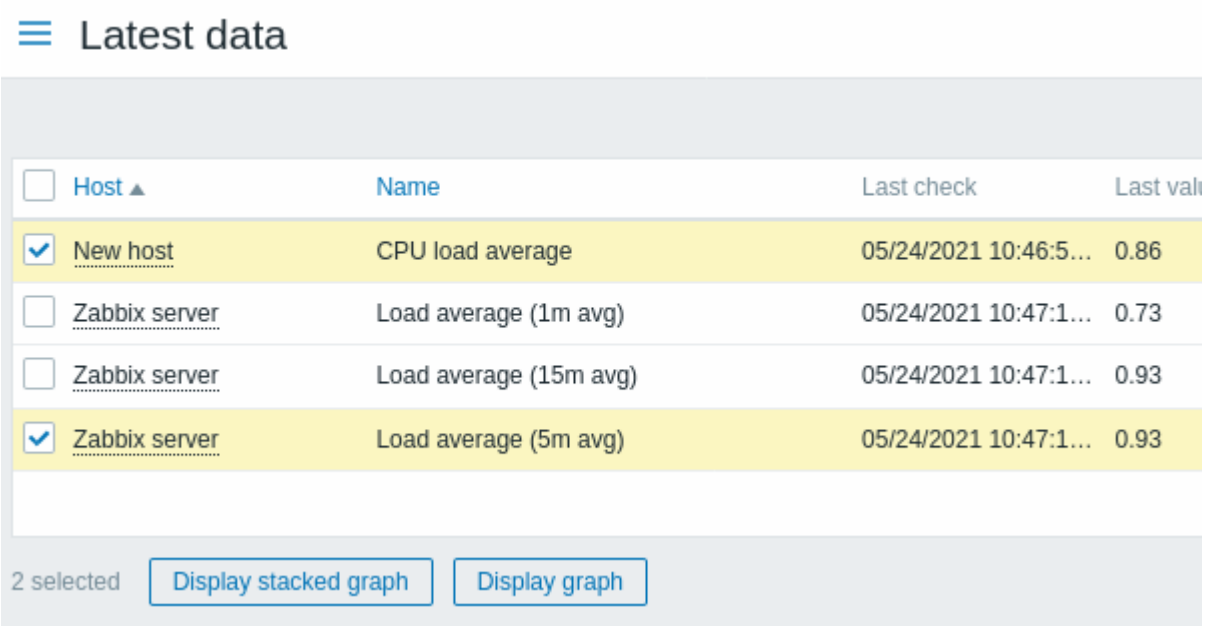
Podczas gdy prosty wykres świetnie nadaje się do uzyskiwania dostępu do danych jednej pozycji, a wykresy niestandardowe oferują opcje dostosowywania, żaden z tych dwóch nie pozwala szybko utworzyć wykresu porównawczego dla wielu pozycji przy niewielkim nakładzie pracy i bez konieczności utrzymania.

Aby rozwiązać ten problem, można w bardzo szybki sposób tworzyć doraźne wykresy dla kilku pozycji.

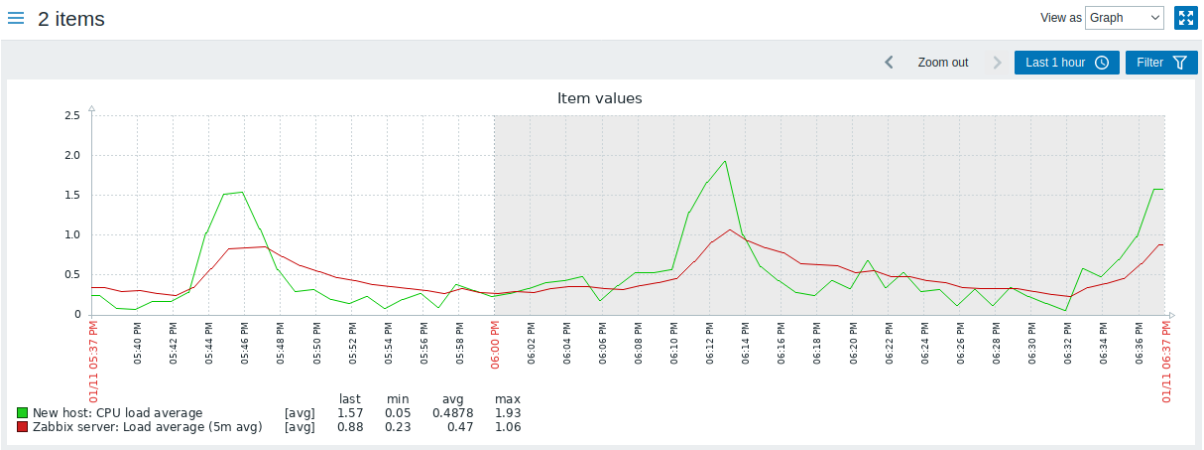
Konfiguracja

Aby utworzyć wykres ad hoc, wykonaj następujące czynności:

- Przejdź do *Monitoring* → *Latest data*
- Użyj filtra, aby wyświetlić interesujące Cię pozycje
- Zaznacz pola wyboru przy pozycjach, które chcesz przedstawić na wykresie
- Kliknij przyciski *Display stacked graph* lub *Display graph*

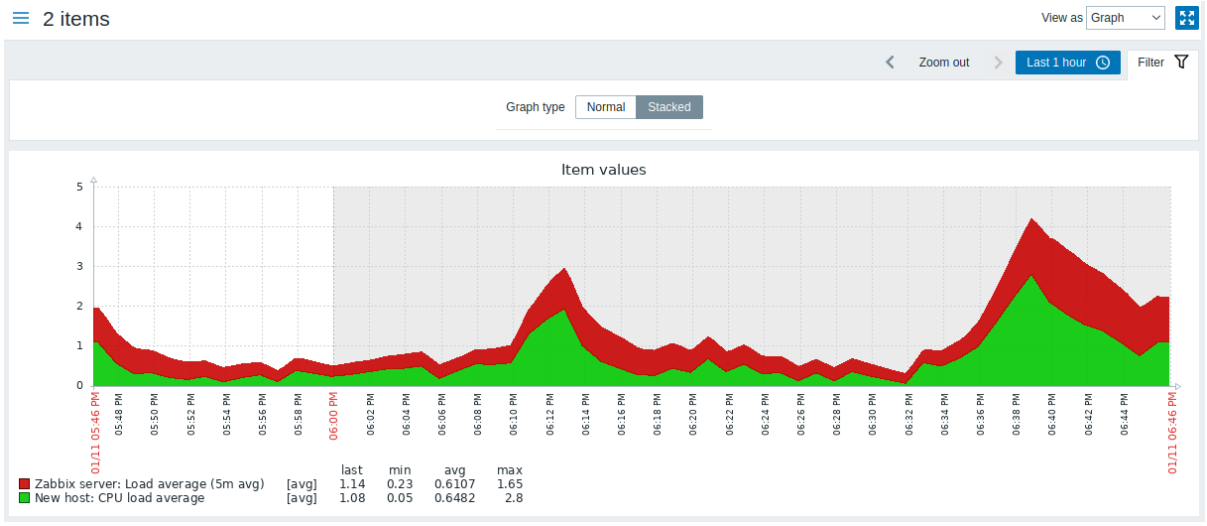


Wykres zostanie utworzony natychmiast:



Należy pamiętać, że aby uniknąć wyświetlania zbyt wielu linii na wykresie, dla każdej pozycji wyświetlana jest tylko wartość średnia (linie wartości min./maks. nie są wyświetlane). Wyzwalacze i informacje o wyzwalaczach nie są wyświetlane na wykresie.

W oknie utworzonego wykresu dostępny jest selektor *Time period* oraz możliwość przełączenia z „normalnego” wykresu liniowego na wykres skumulowany i z powrotem.



2 Mapy sieciowe

Przegląd

Jeśli zarządzasz siecią, możesz chcieć mieć gdzieś ogólny widok swojej infrastruktury. W tym celu możesz tworzyć mapy w Zabbix — sieci oraz czegokolwiek zechcesz.

Wszyscy użytkownicy mogą tworzyć mapy sieci. Mapy mogą być publiczne (dostępne dla wszystkich użytkowników) lub prywatne (dostępne dla wybranych użytkowników).

Przejdź do konfigurowania mapy sieci.

1 Konfigurowanie mapy sieciowej

Przegląd

Konfigurowanie mapy w Zabbix wymaga najpierw utworzenia mapy przez zdefiniowanie jej ogólnych parametrów, a następnie rozpoczęcia wypełniania właściwej mapy elementami i ich połączeniami.

Mapę można wypełnić elementami, takimi jak host, grupa hostów, wyzwalacz, obraz lub inna mapa.

Ikony są używane do reprezentowania elementów mapy. Można zdefiniować informacje, które będą wyświetlane wraz z ikonami, a także ustawić specjalny sposób wyświetlania ostatnich problemów. Można łączyć ikony i definiować informacje wyświetlane na połączeniach.

Do ikon można dodać niestandardowe adresy URL dostępne po kliknięciu. Dzięki temu można powiązać ikonę hosta z właściwościami hosta lub ikonę mapy z inną mapą.

Liczba problemów na mapach jest wyświetlana tylko dla problemów przyczynowych.

Mapami zarządza się w *Monitoring* > *Maps*, gdzie można je konfigurować, zarządzać nimi i je przeglądać. W widoku monitorowania można klikać ikony i korzystać z odnośników do niektórych skryptów i adresów URL.

Mapy sieciowe są oparte na grafice wektorowej (SVG).

Mapy publiczne i prywatne

Wszyscy użytkownicy Zabbix (w tym użytkownicy niebędący administratorami) mogą tworzyć mapy sieci. Mapy mają właściciela — użytkownika, który je utworzył. Mapy mogą być publiczne lub prywatne.

- Mapy *publiczne* są widoczne dla wszystkich użytkowników, jednak aby je zobaczyć, użytkownik musi mieć uprawnienia do odczytu co najmniej jednego elementu mapy. Mapy publiczne mogą być edytowane, jeśli użytkownik/grupa użytkowników ma uprawnienia do odczytu i zapisu do tej mapy oraz co najmniej uprawnienia do odczytu do wszystkich elementów odpowiedniej mapy, w tym wyzwalaczy w połączeniach.
- Mapy *prywatne* są widoczne tylko dla ich właściciela oraz użytkowników/grup użytkowników, którym mapa została udostępniona przez właściciela. Zwykli użytkownicy (niebędący Super adminami) mogą udostępniać mapy tylko grupom, do których należą, oraz użytkownikom należącym do tych grup. Użytkownicy na poziomie Admin mogą widzieć mapy prywatne niezależnie od tego, czy są ich właścicielami, czy należą do listy użytkowników, którym mapa została udostępniona. Mapy prywatne mogą być edytowane przez właściciela mapy oraz wtedy, gdy użytkownik/grupa użytkowników ma uprawnienia do

odczytu i zapisu do tej mapy oraz co najmniej uprawnienia do odczytu do wszystkich elementów odpowiedniej mapy, w tym wyzwalaczy w połączeniach.

Elementy mapy, do których użytkownik nie ma uprawnień odczytu, są wyświetlane z wyszarzoną ikoną, a wszystkie informacje tekstowe o elemencie są ukryte. Jednak etykieta wyzwalacza jest widoczna, nawet jeśli użytkownik nie ma uprawnień do wyzwalacza.

Aby dodać element do mapy, użytkownik musi również mieć co najmniej uprawnienia do odczytu tego elementu.

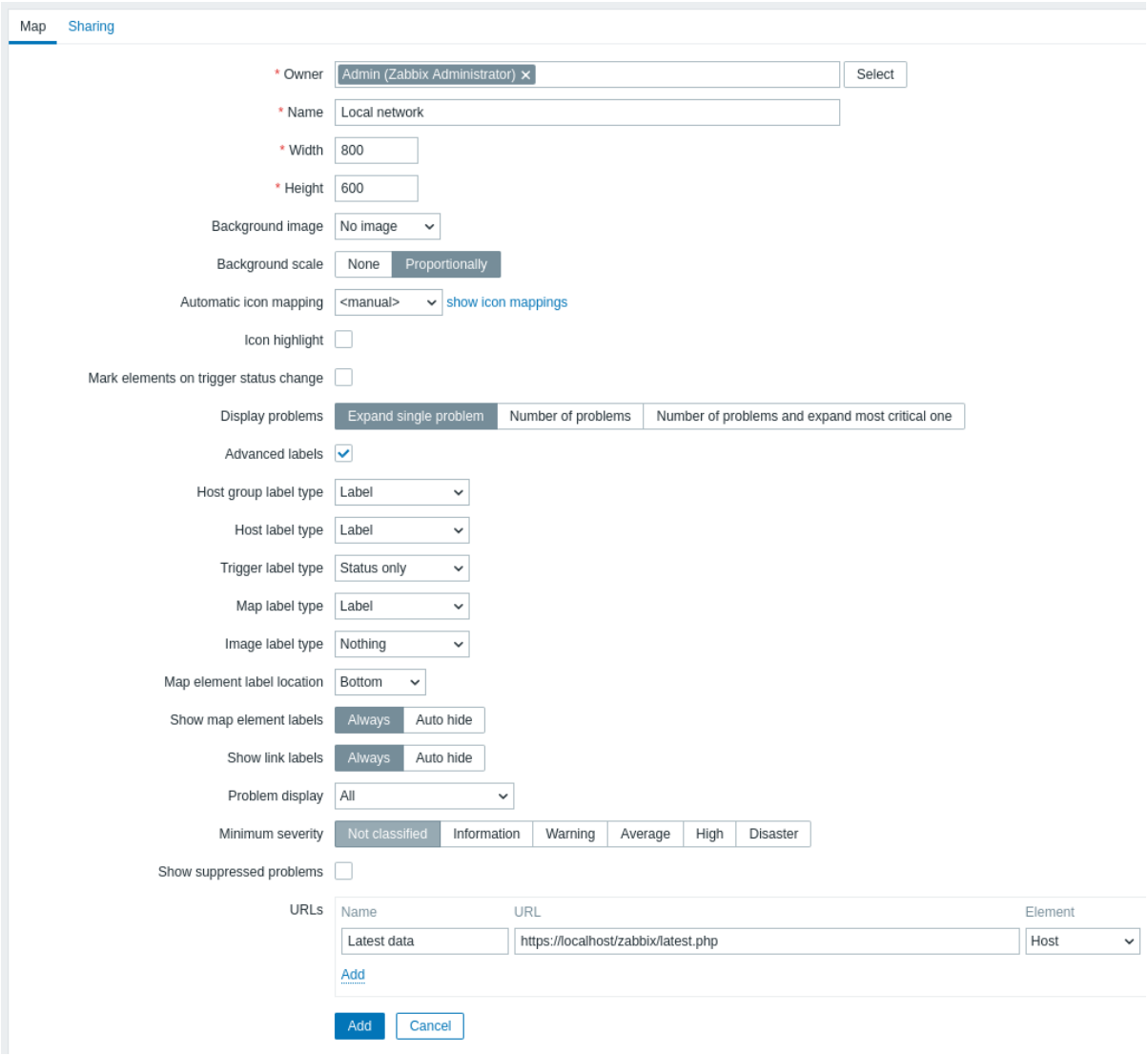
Tworzenie mapy

Aby utworzyć mapę, wykonaj następujące czynności:

1. Przejdź do *Monitoring* > *Maps*.
2. Przejdź do widoku ze wszystkimi mapami.
3. Kliknij *Create map*.

Możesz również użyć przycisku *Clone* w formularzu konfiguracji istniejącej mapy, aby utworzyć nową mapę. Ta mapa będzie miała wszystkie właściwości istniejącej mapy, w tym ogólne atrybuty układu, a także elementy istniejącej mapy.

Karta **Map** zawiera ogólne atrybuty mapy:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Ogólne atrybuty mapy:

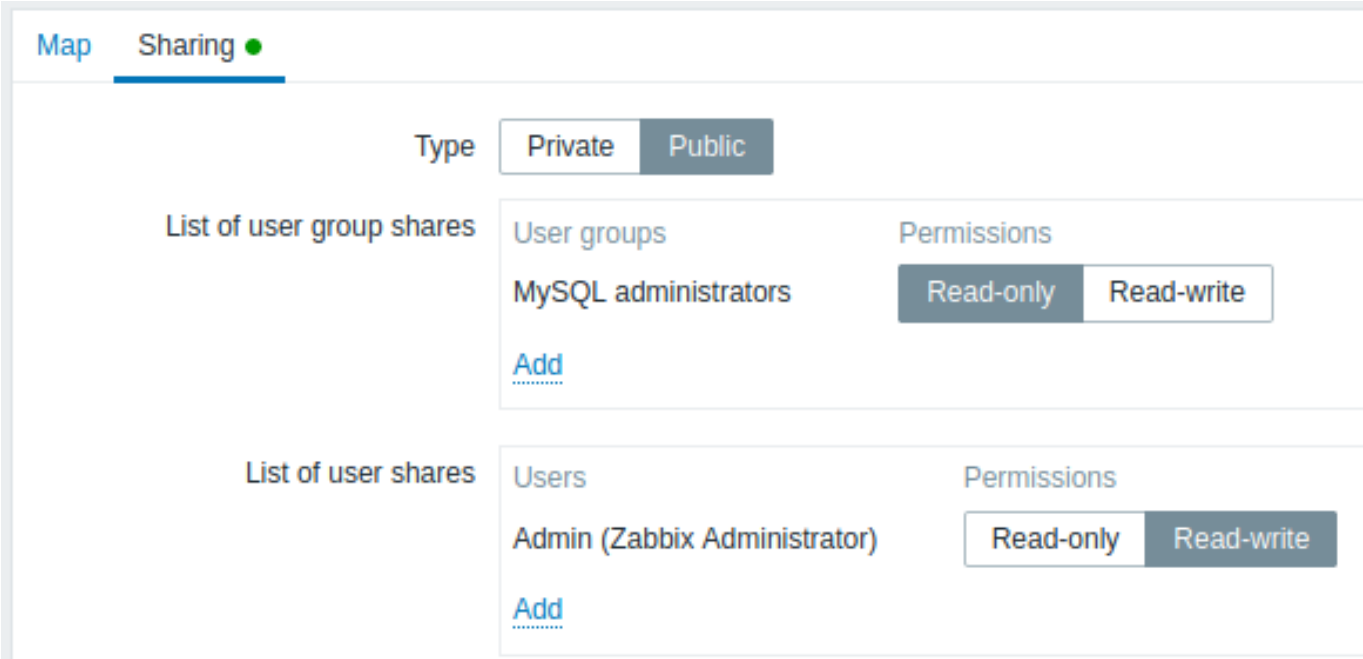
Parameter	Description
Owner	Nazwa właściciela mapy.
Name	Unikalna nazwa mapy.
Width	Szerokość mapy w pikselach.
Height	Wysokość mapy w pikselach.

Parameter	Description
Background image	Użyj obrazu tła: **No image** - brak obrazu tła (białe tło) **Image** - wybrany obraz używany jako obraz tła. Możesz użyć mapy geograficznej lub dowolnego innego obrazu, aby wzbogacić swoją mapę.
Background scale	Użyj skalowania tła: **None** - bez skalowania; **Proportionally** - wypełnia całe tło mapy, zachowując proporcje obrazu (domyślnie).
Automatic icon mapping	Możesz włączyć automatyczne mapowanie ikon, skonfigurowane w *Administration* > *General* > *Icon mapping*. Mapowanie ikon umożliwia przypisywanie określonych ikon do określonych pól inwentarza hosta.
Icon highlight	Jeśli zaznaczysz to pole wyboru, elementy mapy zostaną wyróżnione. Elementy z aktywnym wyzwalaczem otrzymają okrągłe tło w tym samym kolorze co wyzwalacz o najwyższym poziomie ważności. Ponadto wokół okręgu zostanie wyświetlona gruba zielona linia, jeśli wszystkie problemy zostały potwierdzone. Elementy o statusie ”disabled” lub ”in maintenance” otrzymają odpowiednio szare i pomarańczowe kwadratowe tło. Zobacz też: Viewing maps
Mark elements on trigger status change	Niedawna zmiana statusu wyzwalacza (niedawny problem lub rozwiązanie) zostanie wyróżniona znacznikami (czerwonymi trójkątami skierowanymi do wewnątrz) na trzech bokach ikony elementu, które są wolne od etykiety. Znaczniki są wyświetlane przez 30 minut.
Display problems	Wybierz sposób wyświetlania problemów dla elementu mapy: **Expand single problem** - jeśli występuje tylko jeden problem, wyświetlana jest nazwa problemu. W przeciwnym razie wyświetlana jest łączna liczba problemów. **Number of problems** - wyświetlana jest łączna liczba problemów **Number of problems and expand most critical one** - wyświetlana jest nazwa najbardziej krytycznego problemu oraz łączna liczba problemów. ”Najbardziej krytyczny” jest określany na podstawie ważności problemu, a jeśli jest taka sama, na podstawie ID zdarzenia problemu (najpierw wyświetlany jest wyższy ID lub późniejszy problem). W przypadku *trigger map element* jest to określane na podstawie ważności problemu, a jeśli jest taka sama, na podstawie pozycji wyzwalacza na liście wyzwalaczy. W przypadku wielu problemów tego samego wyzwalacza wyświetlany będzie najnowszy z nich.
Advanced labels	Jeśli zaznaczysz to pole wyboru, będziesz mógł zdefiniować oddzielne typy etykiet dla poszczególnych typów elementów.
Host group label type	Typ etykiety używany dla elementu mapy: **Label** - etykieta elementu mapy **IP address** - adres IP **Element name** - nazwa elementu (na przykład nazwa hosta) **Status only** - tylko status (OK lub PROBLEM) **Nothing** - etykiety nie są wyświetlane
Host label type *Trigger label type* *Map label type* *Image label type*	
Map element label location	Położenie etykiety względem elementu mapy: **Bottom** - pod elementem mapy **Left** - po lewej stronie **Right** - po prawej stronie **Top** - nad elementem mapy
Show map element labels	Wybierz sposób wyświetlania etykiet elementów mapy: **Always** - zawsze wyświetlane (domyślnie); **Auto hide** - ukryj etykietę, jeśli nie jest wskazywana kursorem ani zaznaczona.
Show link labels	Wybierz sposób wyświetlania etykiet łączy: **Always** - zawsze wyświetlane (domyślnie); **Auto hide** - ukryj etykietę, jeśli nie jest wskazywana kursorem ani zaznaczona.
Problem display	Wyświetlaj liczbę problemów jako: **All** - wyświetlana będzie pełna liczba problemów **Separated** - wyświetlana będzie liczba niepotwierdzonych problemów oddzielnie od łącznej liczby problemów **Unacknowledged only** - wyświetlana będzie tylko liczba niepotwierdzonych problemów
Minimum severity	Problemy poniżej wybranego minimalnego poziomu ważności nie będą wyświetlane na mapie. Na przykład przy wybranym poziomie *Warning* zmiany związane z wyzwalaczami o poziomie *Information* i *Not classified* nie będą odzwierciedlane na mapie.

Parameter	Description
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta.
URLs	Tutaj można zdefiniować adresy URL dla każdego typu elementu, gdzie: **Name** - etykieta adresu URL; **URL** - adres URL (do 2048 znaków); **Element** - typ elementu (np. *Host*, *Host group* itd.). Będą one wyświetlane jako łącza, gdy użytkownik kliknie element w trybie przeglądania mapy. W nazwach i wartościach adresów URL mapy można używać makr. Pełną listę znajdziesz w sekcji supported macros i wyszukaj ”map URL names and values”.

Udostępnianie

Zakładka **Udostępnianie** zawiera typ mapy oraz opcje udostępniania (grupy użytkowników, użytkownicy) dla map prywatnych:



Parameter	Description
Type	Wybierz typ mapy: **Prywatna** - mapa jest widoczna tylko dla wybranych grup użytkowników i użytkowników **Publiczna** - mapa jest widoczna dla wszystkich
List of user group shares	Wybierz grupy użytkowników, którym mapa ma być udostępniona. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.
List of user shares	Wybierz użytkowników, którym mapa ma być udostępniona. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.

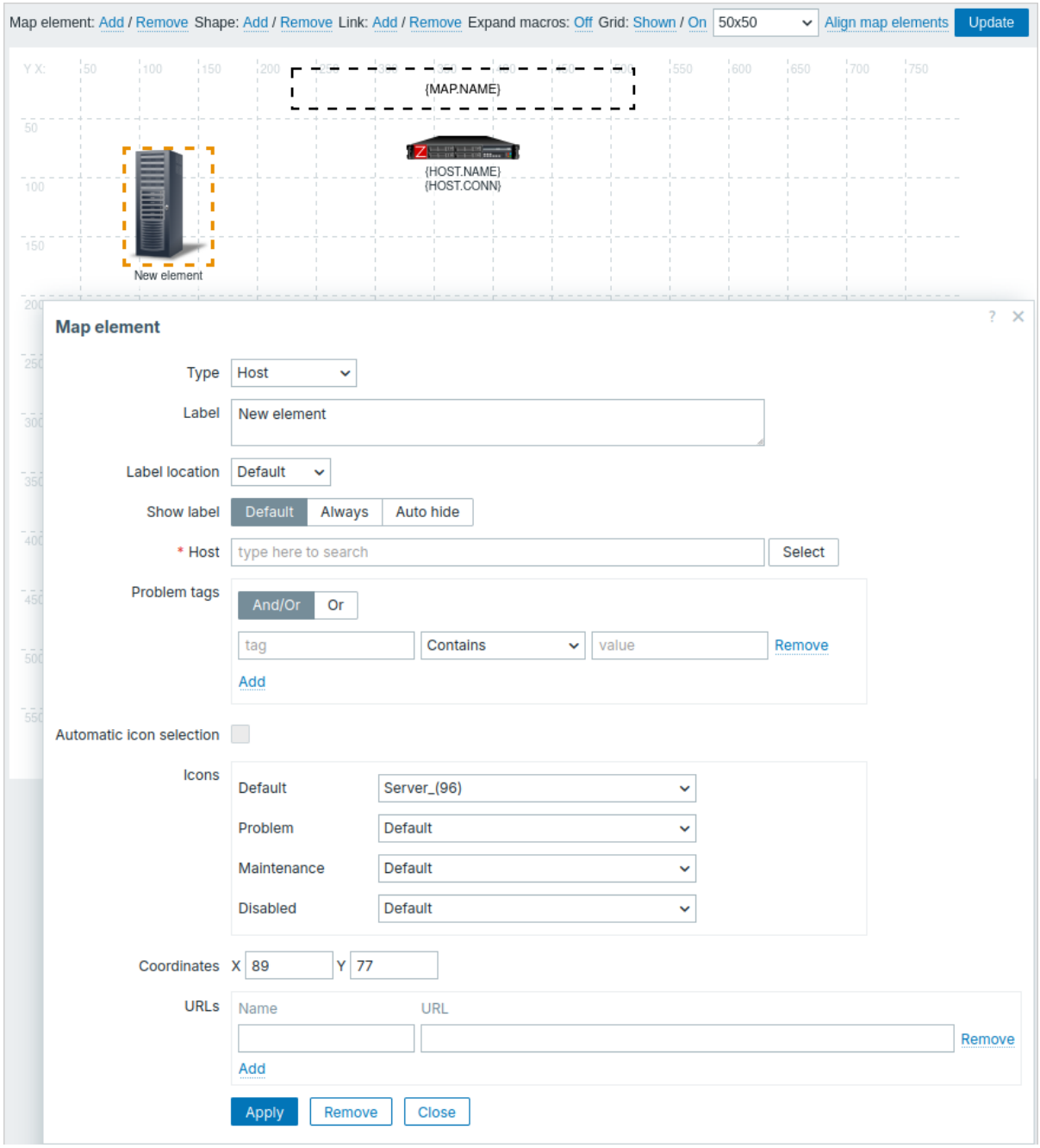
Po kliknięciu *Dodaj*, aby zapisać tę mapę, utworzysz pustą mapę z nazwą, wymiarami i określonymi preferencjami. Teraz musisz dodać kilka elementów. W tym celu kliknij *Edytuj* na liście map, aby otworzyć obszar edycji.

Dodawanie elementów

Aby dodać element, kliknij *Dodaj* obok `Map element`. Nowy element pojawi się w lewym górnym rogu mapy. Przeciągnij go i upuść w dowolnym miejscu.

Pamiętaj, że przy włączonej opcji siatki (”On”) elementy będą zawsze wyrównywane do siatki (możesz wybrać różne rozmiary siatki z listy rozwijanej, a także ukryć/pokazać siatkę). Jeśli chcesz umieszczać elementy w dowolnym miejscu bez wyrównania, ustaw tę opcję na ”Off”. (Losowo rozmieszczone elementy możesz później wyrównać do siatki, klikając *Align map elements*.)

Gdy masz już kilka elementów na mapie, możesz chcieć zacząć je rozróżniać, nadając im nazwy itp. Po kliknięciu elementu wyświetlany jest formularz, w którym można ustawić typ elementu, nadać nazwę, wybrać inną ikonę itd.



Atrybuty elementu mapy:

Parameter	Description
Type	Typ elementu: **Host** - ikona reprezentująca stan wszystkich wyzwalaczy wybranego hosta **Map** - ikona reprezentująca stan wszystkich elementów mapy **Trigger** - ikona reprezentująca stan jednego lub większej liczby wyzwalaczy **Host group** - ikona reprezentująca stan wszystkich wyzwalaczy wszystkich hostów należących do wybranej grupy **Image** - ikona niepowiązana z żadnym zasobem
Label	Etykieta ikony, dowolny ciąg znaków. Można używać makr i ciągów wielowierszowych. W tym polu obsługiwane są makra wyrażeń macros, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`). Pełną listę obsługiwanych makr można znaleźć w sekcji supported macros, wyszukując 'map element labels'.

Parameter	Description
Label location	Położenie etykiety względem ikony: **Default** - domyślne położenie etykiety mapy; **Bottom** - pod ikoną; **Left** - po lewej stronie; **Right** - po prawej stronie; **Top** - nad ikoną.
Show label	Wybierz sposób wyświetlania etykiety elementu: **Default** - użyj ustawień *Show map element labels* z konfiguracji mapy (domyślnie); **Always** - zawsze wyświetlana (domyślnie); **Auto hide** - ukryj etykietę, jeśli nie jest wskazana kursorem ani zaznaczona.
Host	Wybierz hosta; alternatywnie wpisz nazwę hosta. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana z pasującymi hostami. Przewiń w dół, aby wybrać jeden. Kliknij ikonę usunięcia (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Host*.
Map	Wybierz mapę; alternatywnie wpisz nazwę mapy. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy zostanie wyświetlona lista rozwijana z pasującymi mapami. Przewiń w dół, aby wybrać jedną. Kliknij ikonę usunięcia (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Map*.
Triggers	Wybierz jeden lub więcej wyzwalaczy w polu *New triggers* poniżej i kliknij *Add*. Alternatywnie zacznij wpisywać nazwę wyzwalacza i wybierz go z listy rozwijanej pasujących wyzwalaczy. Kliknij ikonę usunięcia (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Trigger*. Kolejność wybranych wyzwalaczy można zmieniać, ale tylko w obrębie tego samego poziomu ważności wyzwalaczy. Wybór wielu wyzwalaczy wpływa również na rozwiązywanie makr {HOST.*} zarówno w trybie edycji, jak i widoku. W trybie edycji pierwsze wyświetlane makra {HOST.*} będą rozwiązywane zależnie od pierwszego wyzwalacza na liście (na podstawie ważności wyzwalacza). Tryb widoku zależy od parametru Display problems w ogólnych atrybutach mapy: - Jeśli wybrano tryb *Expand single problem*, pierwsze wyświetlane makra {HOST.*} będą rozwiązywane zależnie od ostatniego wykrytego wyzwalacza problemu (niezależnie od ważności) lub od pierwszego wyzwalacza na liście (jeśli nie wykryto problemu). - Jeśli wybrano tryb *Number of problems and expand most critical one*, pierwsze wyświetlane makra {HOST.*} będą rozwiązywane zależnie od ważności wyzwalacza.
Host group	Wybierz grupę hostów; alternatywnie wpisz grupę hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy hostów zostanie wyświetlona lista rozwijana z pasującymi grupami hostów. Przewiń w dół, aby wybrać jedną. Kliknij ikonę usunięcia (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Host group*.
Problem tags	Określ tagi, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or **Or** - wystarczy spełnienie jednego warunku To pole jest dostępne tylko dla elementów typu *Host* i *Host group*.
Automatic icon selection	Po zaznaczeniu pola wyboru mapowanie ikon zostanie użyte do określenia, która ikona ma być wyświetlana.
Icons	Możesz wybrać wyświetlanie różnych ikon dla elementu w następujących przypadkach: *Default*, *Problem*, *Maintenance* i *Disabled*. To pole jest dostępne tylko dla elementów typu *Host*.

Parameter	Description
Coordinates	Współrzędne **X** i **Y** elementu mapy.
URLs	Tutaj można ustawić adresy URL specyficzne dla elementu, gdzie: **Name** - etykieta dla adresu URL; **URL** - adres URL (do 2048 znaków). Będą one wyświetlane jako linki, gdy użytkownik kliknie element w trybie przeglądania mapy. Jeśli element ma własne adresy URL i zdefiniowano adresy URL na poziomie mapy dla jego typu, zostaną one połączone w tym samym menu. W nazwach i wartościach elementów mapy można używać makr. Pełną listę można znaleźć w sekcji supported macros, wyszukując 'map URL names and values'.

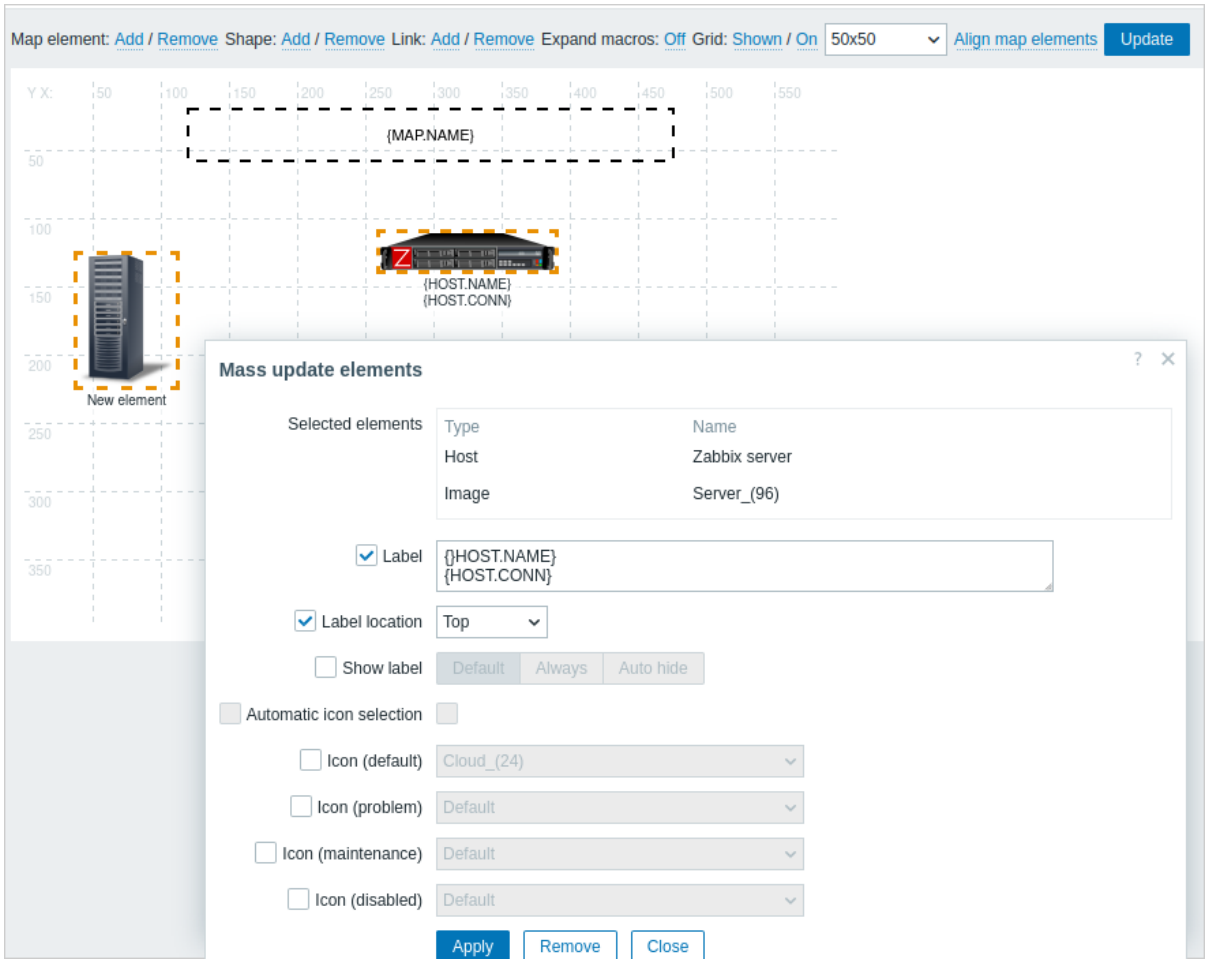
Attention:
Dodane elementy nie są zapisywane automatycznie. Jeśli opuścisz tę stronę, wszystkie zmiany mogą zostać utracone. Dlatego warto kliknąć przycisk **Update** w prawym górnym rogu. Po jego kliknięciu zmiany zostaną zapisane niezależnie od tego, co wybierzesz w następnym wyskakującym oknie.
Wybrane opcje siatki są również zapisywane dla każdej mapy.

Wybieranie elementów

Aby wybrać elementy, wybierz jeden z nich, a następnie przytrzymaj *Ctrl*, aby wybrać pozostałe.

Możesz również wybrać wiele elementów, przeciągając prostokąt w edytowalnym obszarze i zaznaczając wszystkie elementy, które się w nim znajdują.

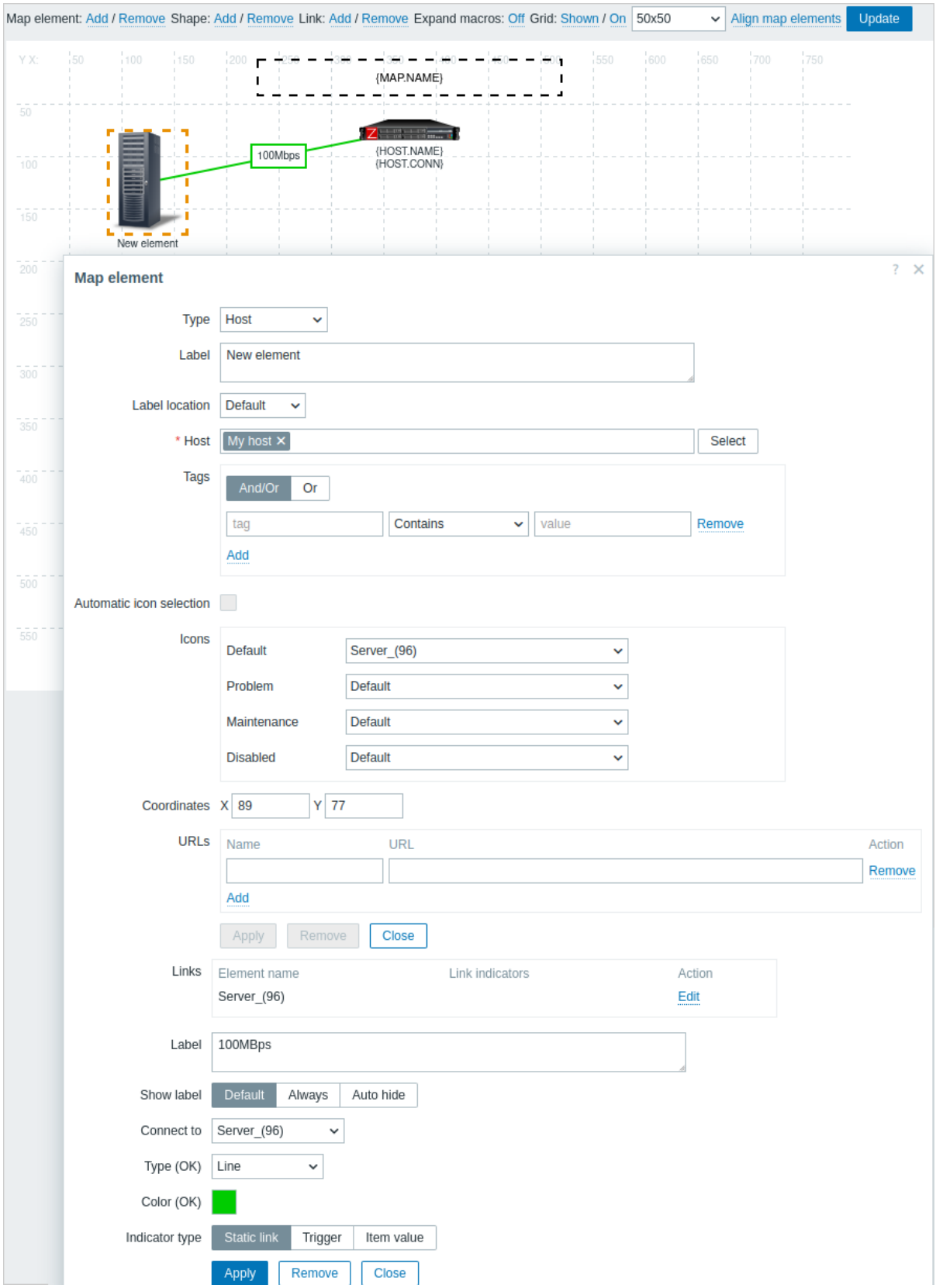
Po wybraniu więcej niż jednego elementu formularz właściwości elementu przełącza się w tryb masowej aktualizacji, dzięki czemu możesz jednocześnie zmienić atrybuty wybranych elementów. Aby to zrobić, zaznacz atrybut za pomocą pola wyboru i wprowadź dla niego nową wartość. Możesz tutaj używać makr (na przykład {HOST.NAME} jako etykiety elementu).



Łączenie elementów

Po umieszczeniu kilku elementów na mapie można rozpocząć ich łączenie. Aby połączyć dwa elementy, należy najpierw je zaznaczyć. Gdy elementy są zaznaczone, kliknij *Dodaj* obok pola Link.

Po utworzeniu połączenia formularz pojedynczego elementu zawiera teraz dodatkową sekcję *Links*. Kliknij *Edytuj*, aby edytować atrybuty połączenia.



Atrybuty połączenia:

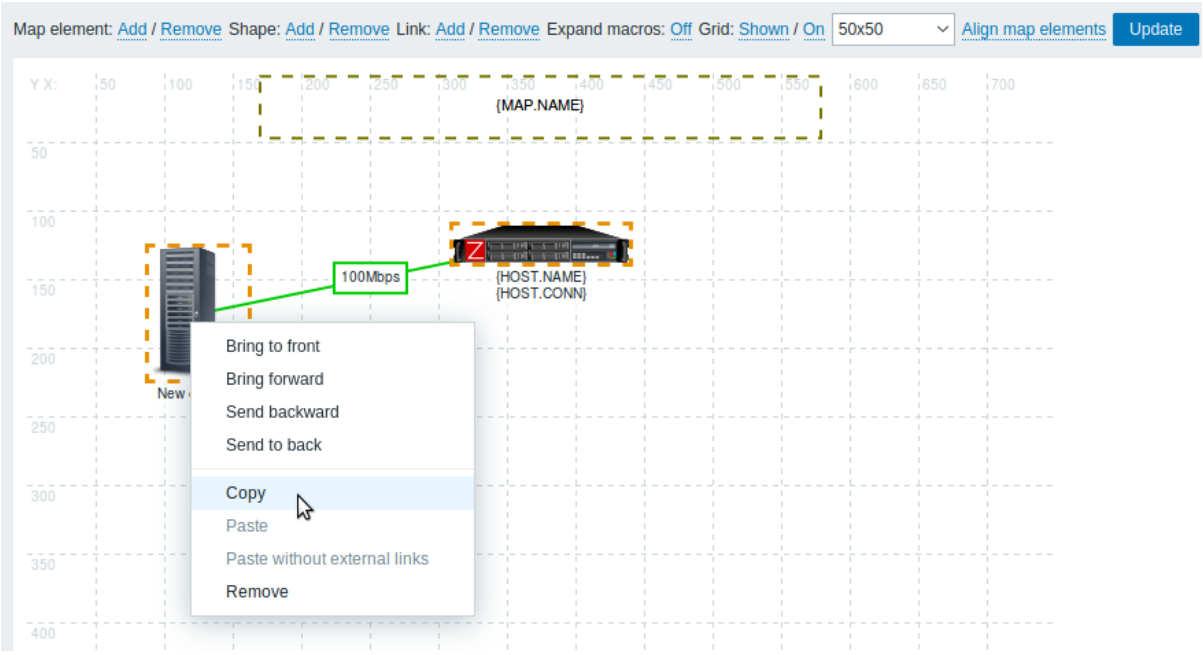
Parametr	Opis
Etykieta	Etykieta, która będzie wyświetlana nad połączeniem. W tym polu obsługiwane są wyrażenia makr, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`).

Parametr	Opis
Pokaż etykietę	Wybierz sposób wyświetlania etykiety połączenia: **Domyślnie** - użyj ustawień *Show map element labels* z konfiguracji mapy (domyślnie); **Zawsze** - zawsze wyświetlana (domyślnie); **Ukryj automatycznie** - ukryj etykietę, jeśli nie jest wskazana kursorem ani zaznaczona.
Połącz z	Element, z którym połączenie jest zestawiane.
Typ (OK)	Domyślny styl połączenia. Wybierz styl: *Linia*, *Pogrubiona linia*, *Kropkowana* lub *Linia przerywana*.
Kolor (OK)	Domyślny kolor połączenia. Kliknij próbkę koloru, aby wybrać inny kolor.
Typ wskaźnika	Wybierz typ wskaźnika połączenia: **Połączenie statyczne** - bez wskaźników; **Wyzwalacz** - zezwala na wskaźniki połączenia oparte na wyzwalaczach; **Wartość pozycji** - zezwala na wskaźniki połączenia oparte na wartościach pozycji.
Pozycja	Wybierz pozycję. Wartość tej pozycji będzie wpływać na styl połączenia. To pole jest dostępne, gdy dla *Typ wskaźnika* wybrano *Wartość pozycji*.
Wskaźniki	Lista wskaźników połączenia. To pole jest dostępne, gdy *Typ wskaźnika* jest ustawiony na *Wyzwalacz* lub *Wartość pozycji*. Jeśli ustawiono *Wyzwalacz*, wyzwalacze mogą zostać przypisane jako wskaźniki połączenia. Gdy wyzwalacz jest w stanie *Problem*, jego styl (wybrany kolor i typ linii) zostanie zastosowany do połączenia. Jeśli ustawiono *Wartość pozycji* i w polu *Pozycja* wybrano pozycję, wartości pozycji mogą zostać przypisane jako wskaźniki połączenia. Gdy wartość pozycji osiągnie określony próg (dla pozycji numerycznej) lub będzie zgodna ze wzorcem (dla pozycji tekstowej), jej styl (wybrany kolor i typ linii) zostanie zastosowany do połączenia.

Przenoszenie i kopiowanie-wklejanie elementów

Kilka wybranych elementów można **przenieść** w inne miejsce na mapie, klikając jeden z zaznaczonych elementów, przytrzymując przycisk myszy i przesuwając kursor do żądanej lokalizacji.

Jeden lub więcej elementów można **skopiować**, zaznaczając elementy, następnie klikając zaznaczony element prawym przyciskiem myszy i wybierając z menu opcję *Kopiuj*.

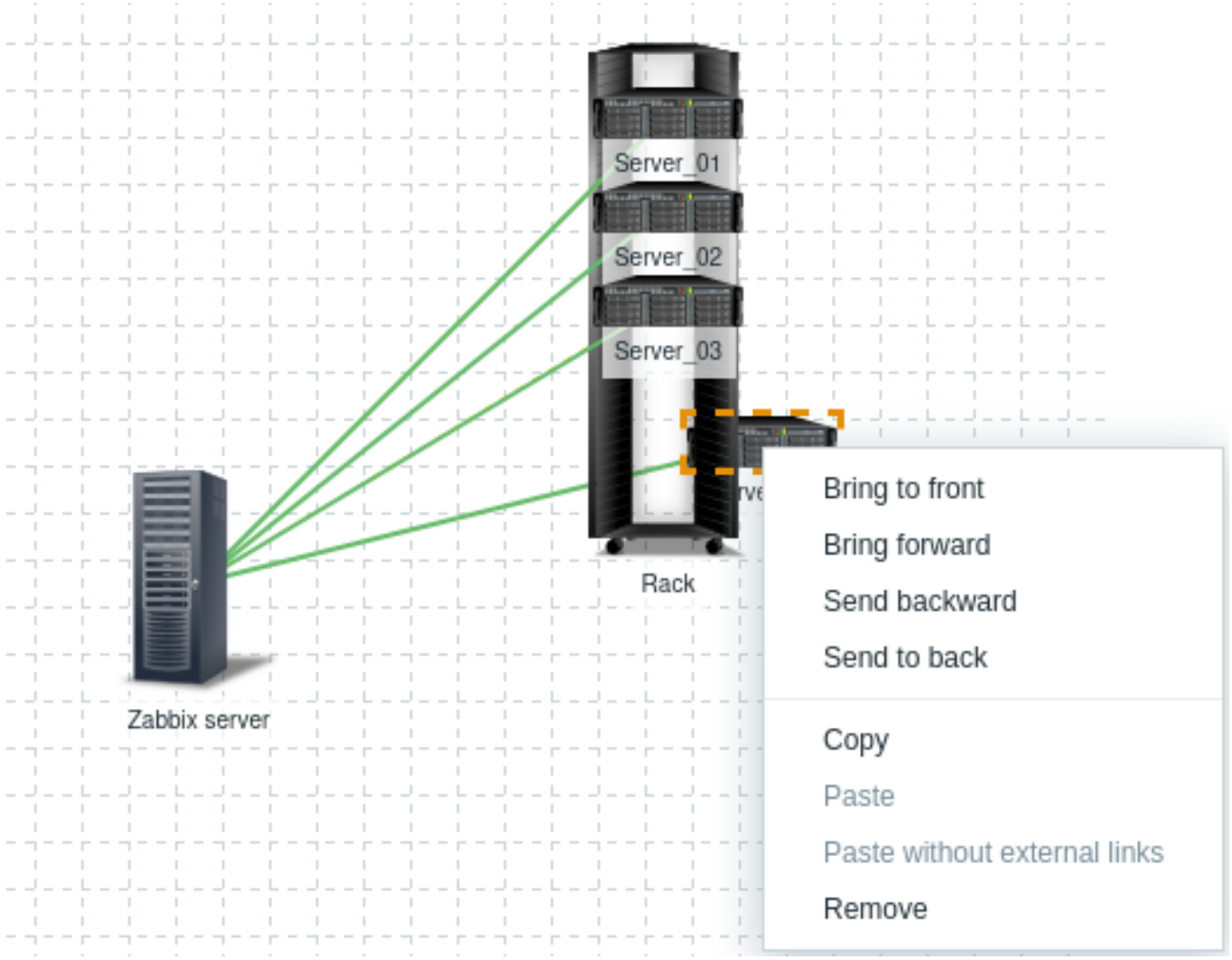


Aby wkleić elementy, kliknij obszar mapy prawym przyciskiem myszy i wybierz z menu opcję *Wklej*. Opcja *Wklej bez zewnętrznych połączeń* wklei elementy, zachowując tylko połączenia między zaznaczonymi elementami.

Kopiowanie i wklejanie działa w obrębie tego samego okna przeglądarki. Skróty klawiaturowe nie są obsługiwane.

Porządkowanie elementów

Aby przenieść jeden element przed drugi (lub odwrotnie), kliknij element prawym przyciskiem myszy i wybierz *Przesuń do przodu/Przenieś na wierzch* lub *Przesuń do tyłu/Przenieś na spód*.

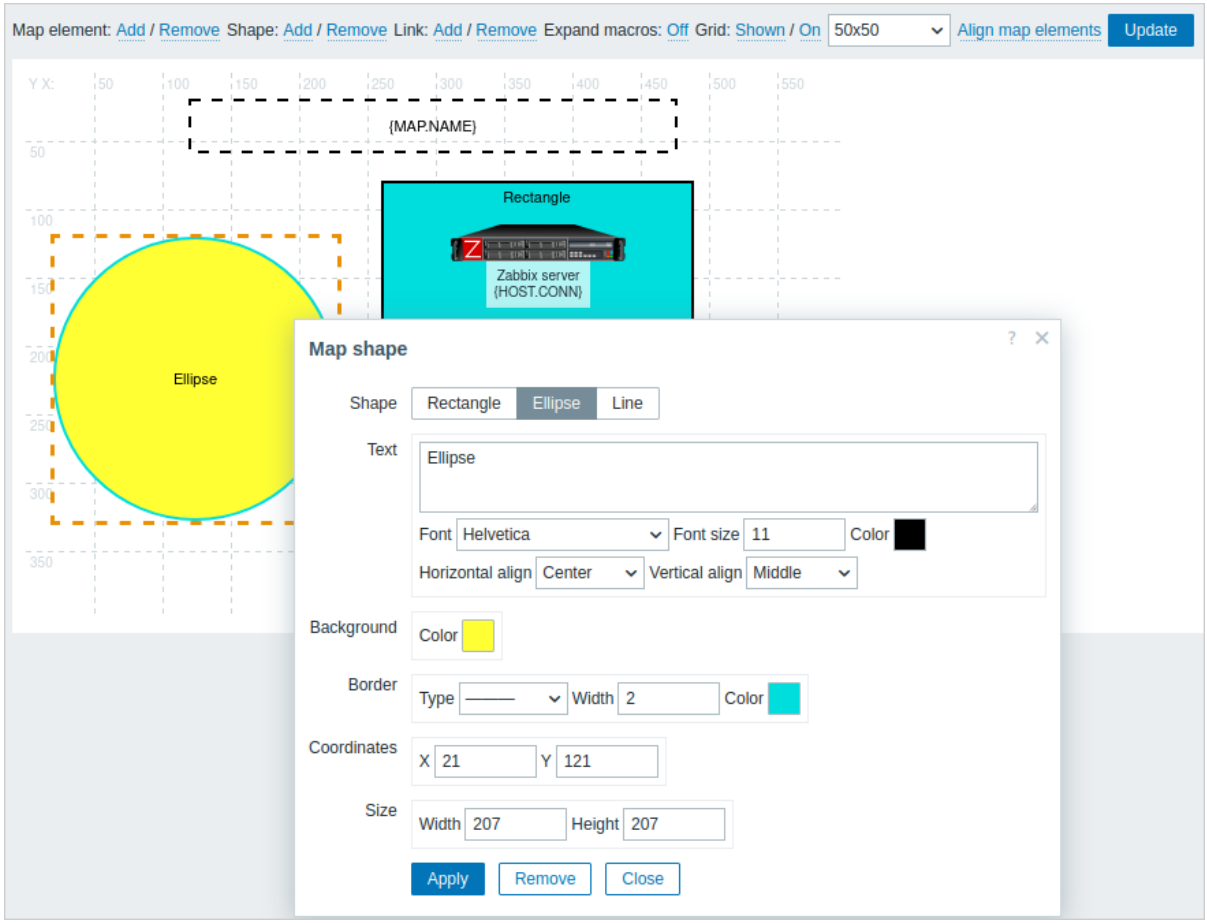


Dodawanie kształtów

Oprócz elementów mapy można również dodawać kształty. Kształty nie są elementami mapy; stanowią jedynie reprezentację wizualną. Na przykład kształt prostokąta może być użyty jako tło do grupowania kilku hostów. Można dodawać kształty prostokąta i elipsy.

Aby dodać kształt, kliknij *Dodaj* obok pola Kształt. Nowy kształt pojawi się w lewym górnym rogu mapy. Przeciągnij go i upuść w dowolnym miejscu.

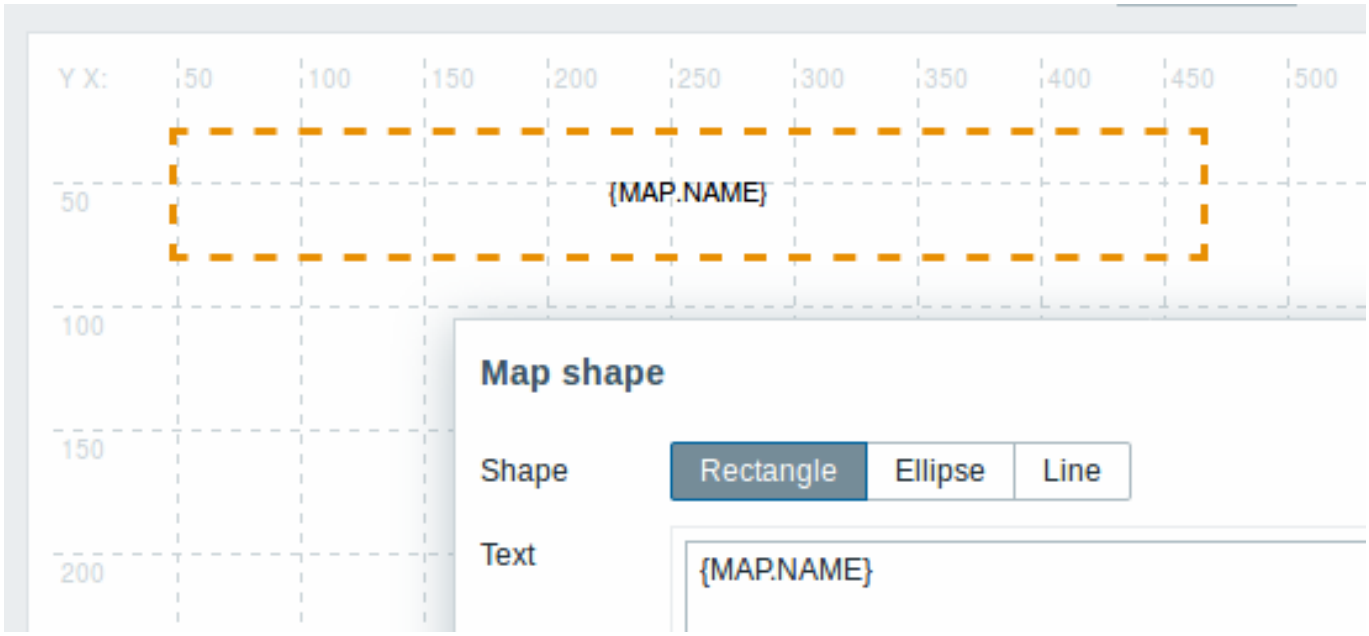
Nowy kształt jest dodawany z domyślnymi kolorami. Po kliknięciu kształtu zostanie wyświetlony formularz, w którym można dostosować wygląd kształtu, dodać tekst itd.



Aby zaznaczyć kształty, wybierz jeden z nich, a następnie przytrzymaj *Ctrl*, aby zaznaczyć pozostałe. Gdy zaznaczonych jest kilka kształtów, wspólne właściwości można zbiorczo zaktualizować, podobnie jak w przypadku elementów.

Do kształtów można dodawać tekst. Wyrażenia makr są obsługiwane w tekście, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`).

Aby wyświetlać tylko tekst, można uczynić kształt niewidocznym przez usunięcie obramowania kształtu (wybierz „Brak” w polu *Obramowanie*). Na przykład zwróć uwagę, że makro {MAP.NAME}, widoczne na powyższym zrzucie ekranu, jest w rzeczywistości kształtem prostokąta z tekstem, co można zobaczyć po kliknięciu makra:



{MAP.NAME} jest rozwijane do skonfigurowanej nazwy mapy podczas wyświetlania mapy.

Jeśli w tekście używane są hiperłącza, stają się klikalne podczas wyświetlania mapy.

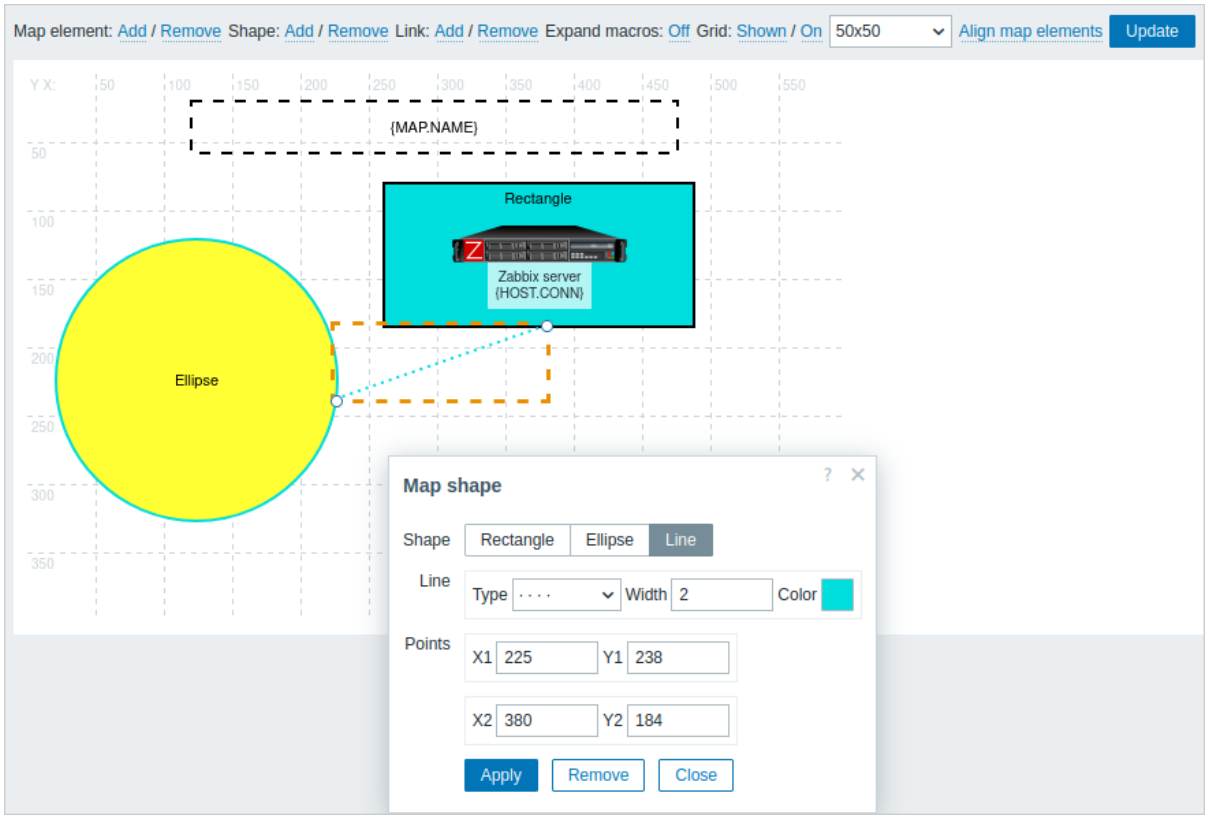
Zawijanie wierszy dla tekstu wewnątrz kształtów jest zawsze włączone. Jednak w obrębie elipsy wiersze są zawijane tak, jakby elipsa była prostokątem. Zawijanie wyrazów nie jest zaimplementowane, więc długie wyrazy (wyrazy, które nie mieszczą się w

kształcie) nie są zawijane, lecz maskowane (na stronie edycji mapy) albo przycinane (na innych stronach z mapami).

Dodawanie linii

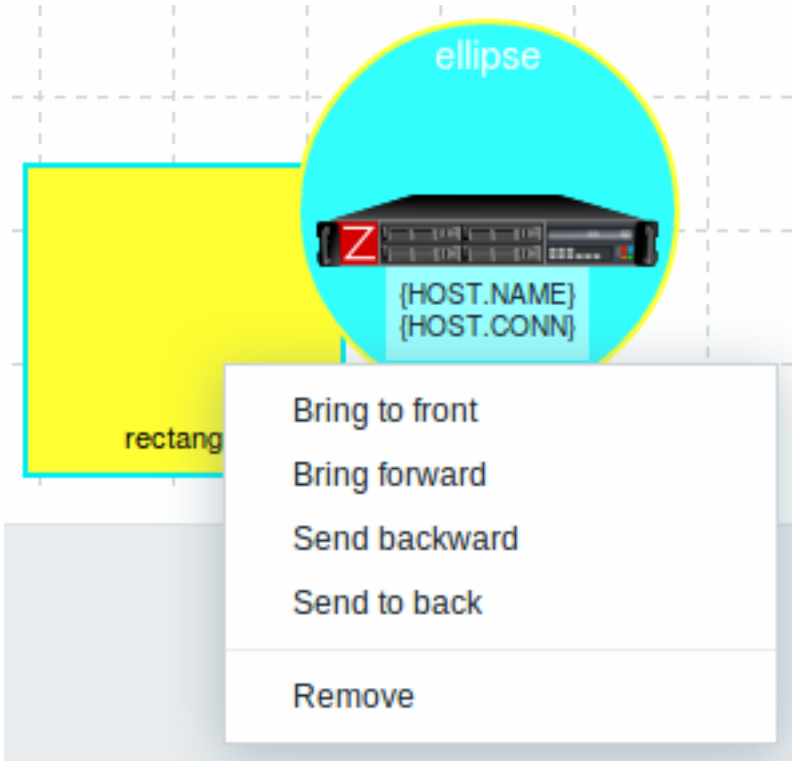
Oprócz kształtów można również dodawać linie. Linie mogą być używane do łączenia elementów lub kształtów na mapie.

Aby dodać linię, kliknij *Dodaj* obok opcji Kształt. Nowy kształt pojawi się w lewym górnym rogu mapy. Zaznacz go i kliknij *Linia* w formularzu edycji, aby zmienić kształt na linię. Następnie dostosuj właściwości linii, takie jak typ linii, szerokość, kolor itp.



Ustalanie kolejności kształtów i linii

Aby przenieść jeden kształt przed drugi (lub odwrotnie), kliknij kształt prawym przyciskiem myszy, aby wyświetlić menu kształtu mapy.

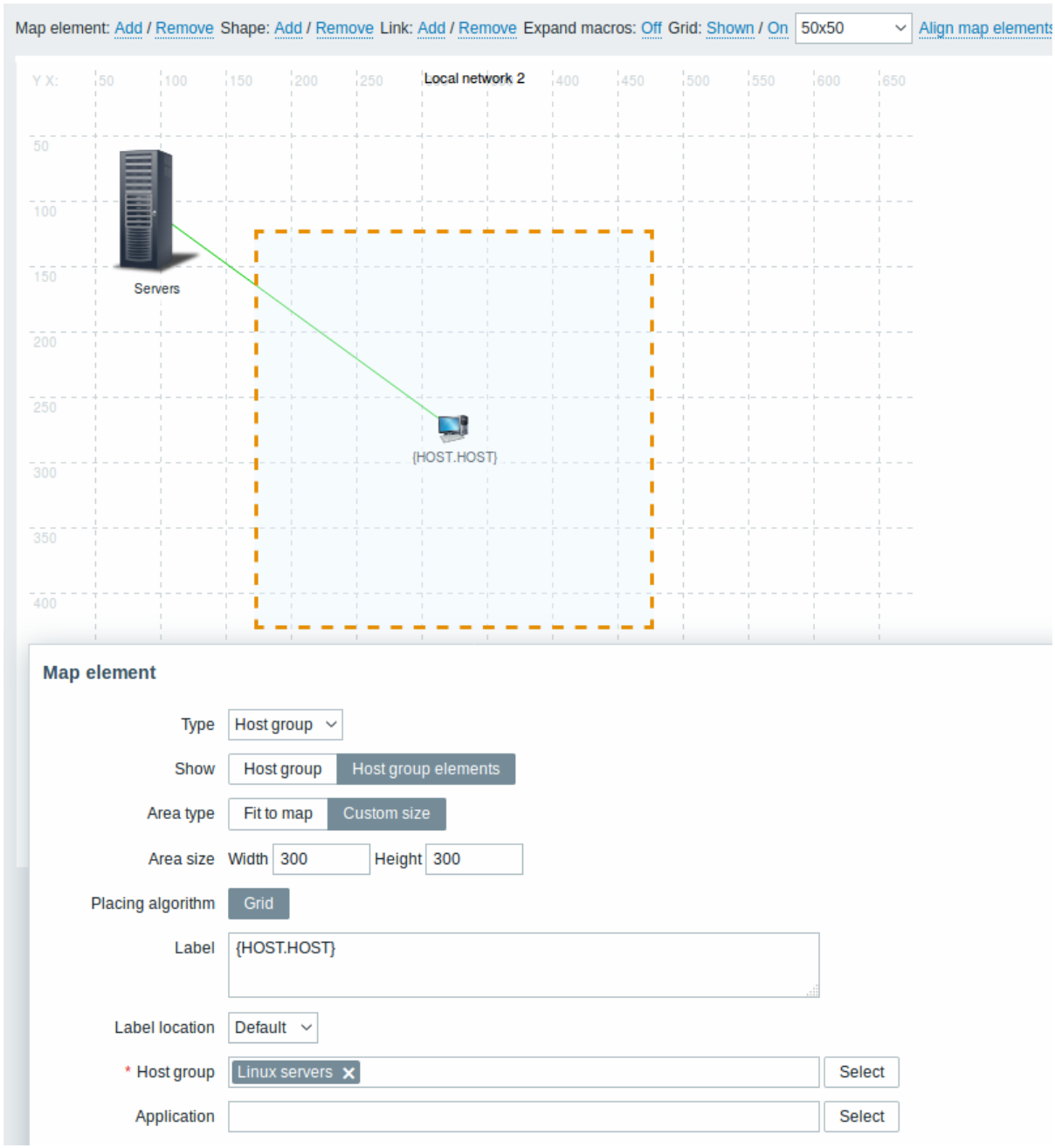


2 Elementy grup hostów

Przegląd

Ta sekcja wyjaśnia, jak dodać element typu *Grupa hostów* podczas konfigurowania mapy sieci.

Konfiguracja



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Ta tabela zawiera parametry typowe dla typu elementu *Host group*:

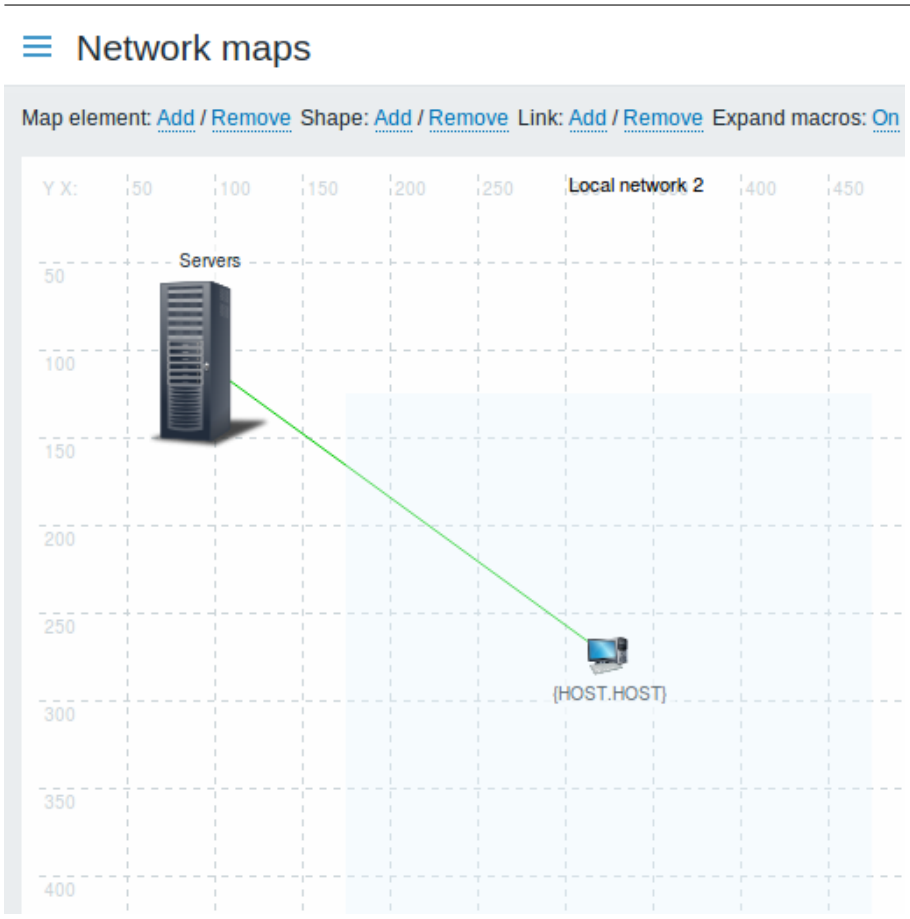
Parametr	Opis
Type	Wybierz typ elementu: **Host group** - Ikona reprezentująca stan wszystkich wyzwalaczy wszystkich hostów należących do wybranej grupy.
Show	Opcje wyświetlania: **Host group** - Wybranie tej opcji spowoduje wyświetlenie jednej ikony prezentującej odpowiednie informacje o określonej grupie hostów. **Host group elements** - Wybranie tej opcji spowoduje wyświetlenie wielu ikon prezentujących odpowiednie informacje o każdym pojedynczym elemencie (hoście) określonej grupy hostów.

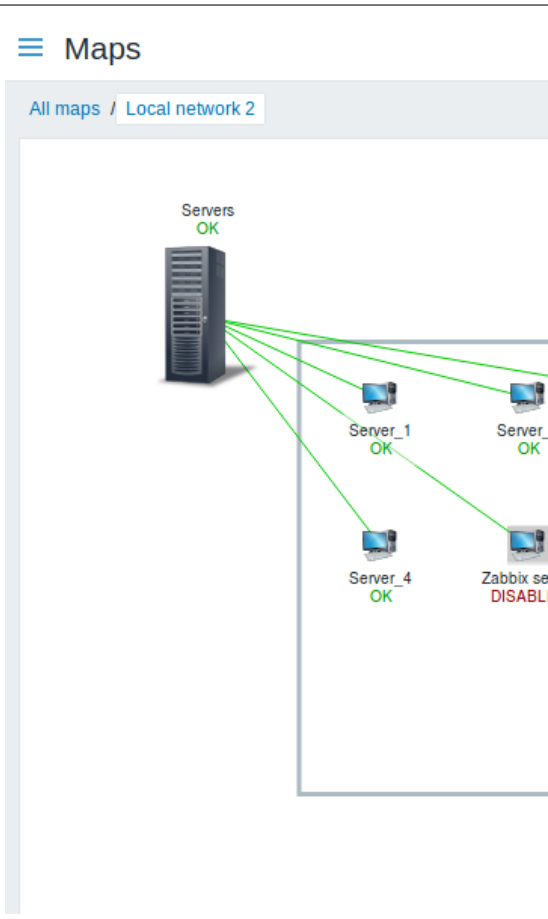
Parametr	Opis
Area type	To ustawienie jest dostępne, jeśli *Show* jest ustawione na *Host group elements*: **Fit to map** - Wszystkie elementy grupy hostów są rozmieszczane równomiernie na mapie. **Custom size** - Ręczne ustawienie obszaru mapy, w którym mają być wyświetlane wszystkie elementy grupy hostów.
Area size	To ustawienie jest dostępne, jeśli *Area type* jest ustawione na *Custom size*: **Width** - Wartość w pikselach określająca szerokość obszaru mapy. **Height** - Wartość w pikselach określająca wysokość obszaru mapy.
Placing algorithm	**Grid** - Jedyna dostępna opcja wyświetlania wszystkich elementów grupy hostów.
Label	Etykieta ikony, dowolny ciąg znaków. W etykietach można używać makr oraz ciągów wielowierszowych. Jeśli typ elementu mapy to *Host group*, określenie niektórych makr wpływa na widok mapy, wyświetlając odpowiednie informacje o każdym pojedynczym hoście. Na przykład, jeśli użyte jest makro {HOST.IP}, widok edycji mapy będzie wyświetlał tylko samo makro {HOST.IP}, natomiast widok mapy będzie zawierał i wyświetlał unikalny adres IP każdego hosta.

Wyświetlanie elementów grupy hostów

Ta opcja jest dostępna, jeśli wybrano opcję wyświetlania *Elementy grupy hostów*. Po wybraniu *Elementy grupy hostów* jako opcji *pokaż* początkowo zobaczysz tylko jedną ikonę dla grupy hostów. Jednak po zapisaniu mapy i przejściu do widoku mapy zobaczysz, że mapa zawiera wszystkie elementy (hosty) określonej grupy hostów:

Widok edycji mapy	Widok mapy





Zwróć uwagę, jak używane jest makro {HOST.HOST}. Podczas edycji mapy nazwa makra nie jest rozwinięta, natomiast w widoku mapy wyświetlane są wszystkie unikalne nazwy hostów.

3 Wskaźniki łączy

Przegląd

Możesz przypisywać wskaźniki do linku między elementami na mapie sieci.

Wskaźniki mogą być oparte na wyzwalaczach lub wartościach pozycji. Możliwe jest wyświetlanie różnych stylów i kolorów linku:

- gdy wyzwalacze przechodzą w stan problemu;
- gdy wartość pozycji:
  - osiąga próg (dla pozycji numerycznych);
  - pasuje do wyrażenia regularnego (dla pozycji tekstowych).

Podczas konfigurowania linku ustawiasz domyślny typ i kolor linku. Przypisując wskaźniki do linku, można sprawić, że styl i kolor linku będą zależeć od stanu wyzwalacza lub wartości pozycji.

Na przykład, jeśli którykolwiek z przypisanych wyzwalaczy przejdzie w stan problemu, styl i kolor linku zmienią się tak, aby to odzwierciedlić. Załóżmy więc, że domyślny link był zieloną linią. Teraz, gdy wyzwalacz znajduje się w stanie problemu, link może stać się pogrubioną czerwoną linią (jeśli tak to zdefiniowano).

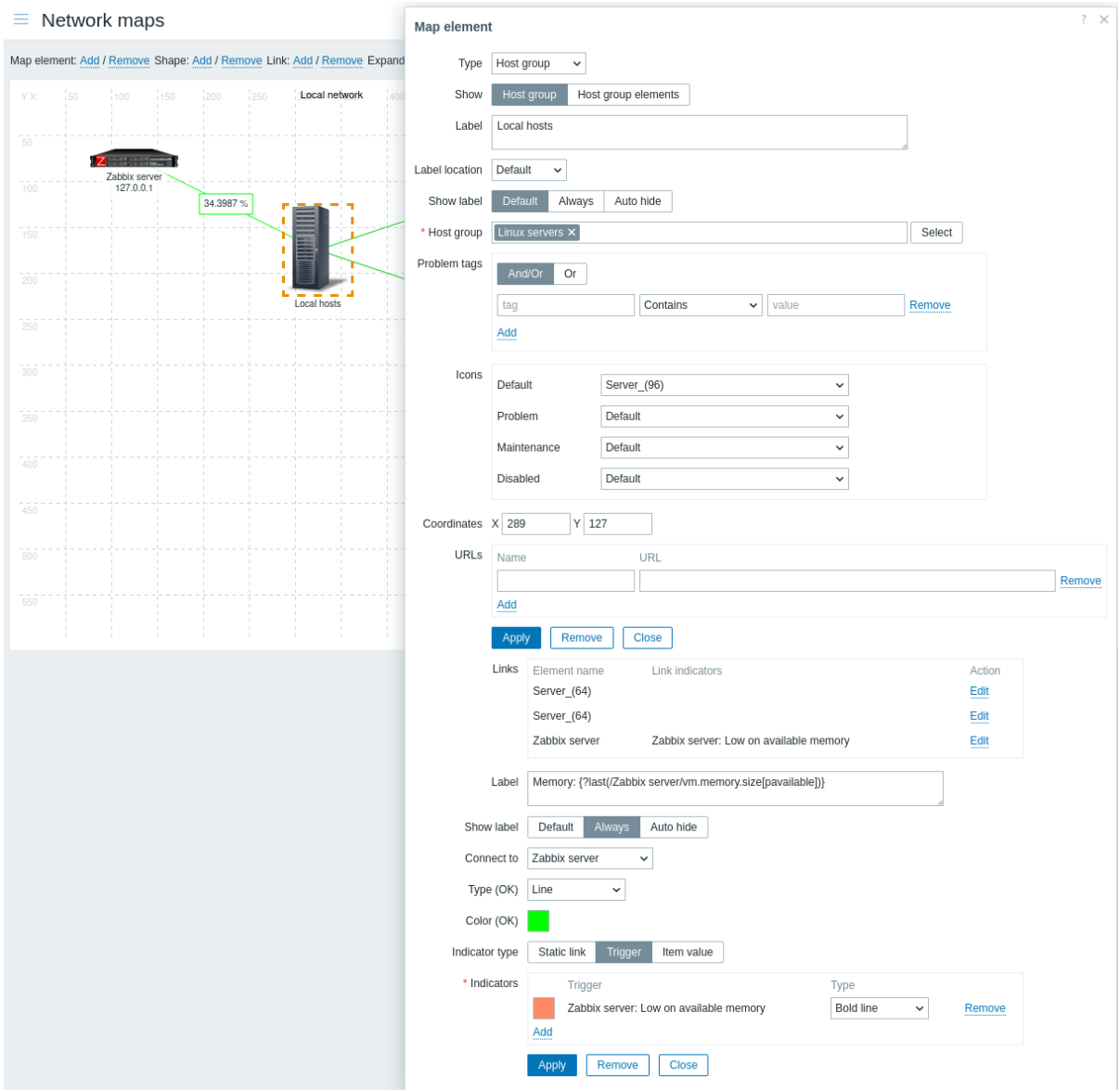
Podobnie, jeśli wartość pozycji osiągnie określony próg lub dopasuje się do określonego wyrażenia regularnego, styl linku może to odzwierciedlać.

Konfiguracja

Wyzwalacze

Aby przypisać wyzwalacze jako wskaźniki łączy, wykonaj następujące czynności:

- wybierz element mapy
- kliknij *Edytuj* w sekcji *Łącza* dla odpowiedniego łącza
- wybierz *Wyzwalacz* jako typ wskaźnika
- kliknij *Dodaj* w bloku *Wskaźniki* i wybierz jeden lub więcej wyzwalaczy

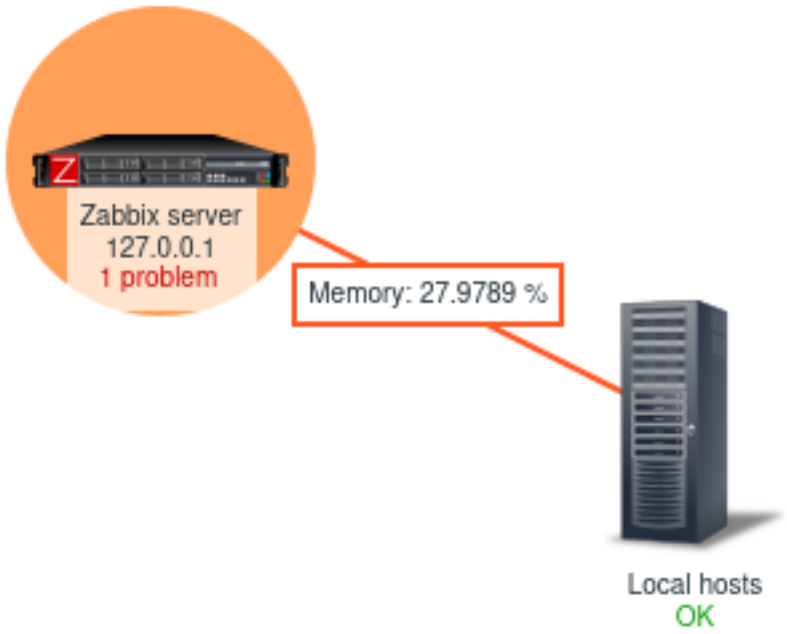


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Dodane wyzwalacze są widoczne na liście *Wskaźniki*.

Możesz ustawić typ łącza i kolor dla każdego wskaźnika bezpośrednio z listy. Po zakończeniu kliknij *Zastosuj*, zamknij formularz i kliknij *Aktualizuj*, aby zapisać zmiany mapy.

W *Monitoring* → *Mapy* odpowiedni typ i kolor łącza będą wyświetlane, jeśli **wyzwalacz** przejdzie w stan problemu.



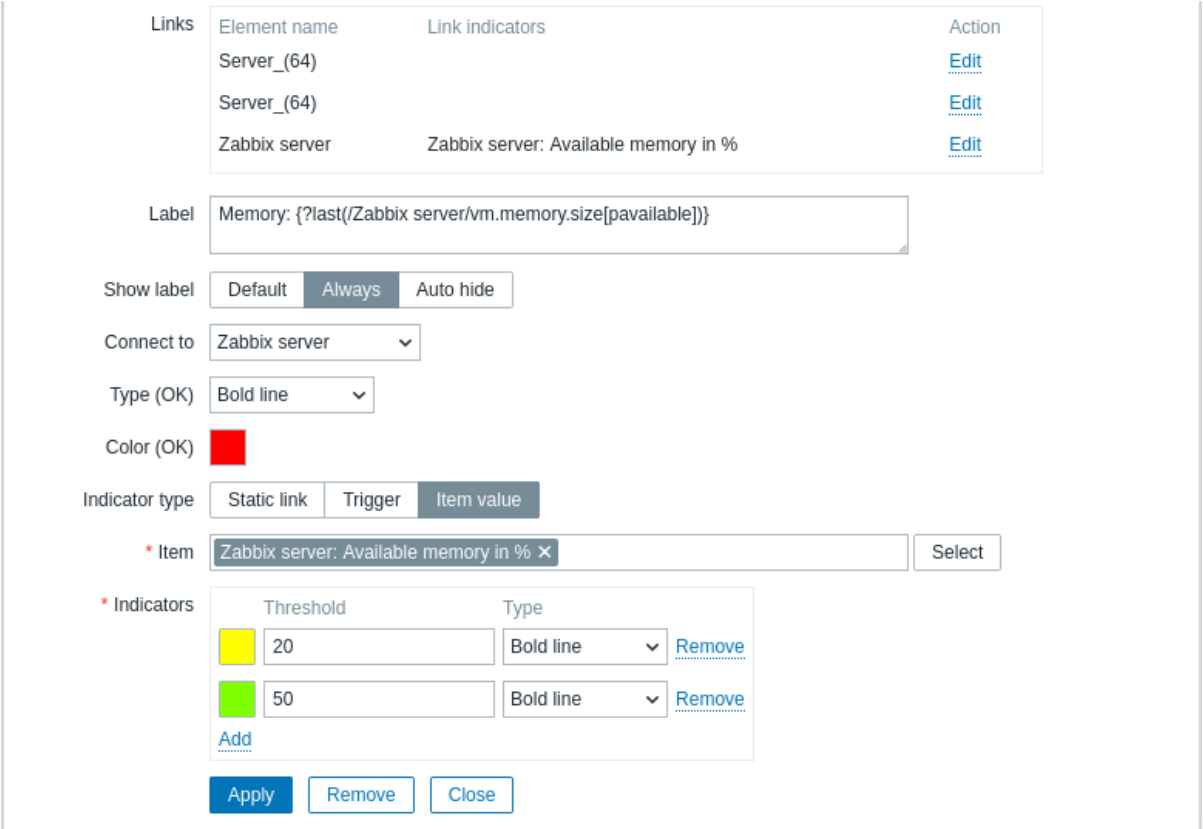
Note:
Jeśli wiele wyzwalaczy przejdzie w stan problemu, styl i kolor łącza zostaną określone przez problem o najwyższym poziomie ważności. Jeśli do tego samego łącza mapy przypisano wiele wyzwalaczy o tym samym poziomie ważności, pierwszeństwo ma ten o najniższym ID. Zwróć także uwagę, że:

1. Ustawienia *Minimalny poziom ważności wyzwalacza* i *Pokaż ukryty problem* w konfiguracji mapy wpływają na to, które problemy są brane pod uwagę.
2. W przypadku wyzwalaczy z wieloma problemami (wielokrotne generowanie problemów) każdy problem może mieć poziom ważności inny niż poziom ważności wyzwalacza (zmieniony ręcznie), może mieć różne tagi (z powodu makr) i może być ukryty.

Wartości pozycji

Aby przypisać wartości pozycji jako wskaźniki łączy, wykonaj następujące czynności:

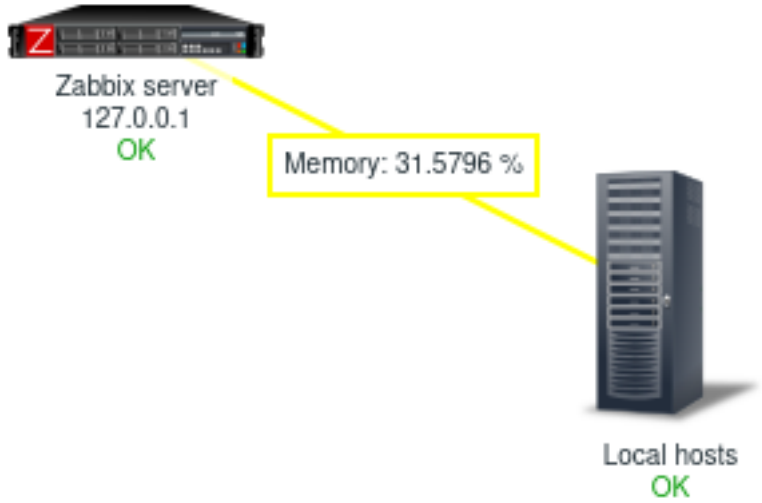
- wybierz element mapy
- kliknij *Edytuj* w sekcji *Łącza* dla odpowiedniego łącza
- wybierz *Wartość pozycji* jako typ wskaźnika
- wybierz pozycja
- dodaj jeden lub więcej progów wartości pozycji lub wzorców w bloku *Wskaźniki*



Dodane progi/wzorce pozycji są widoczne na liście *Wskaźniki*.

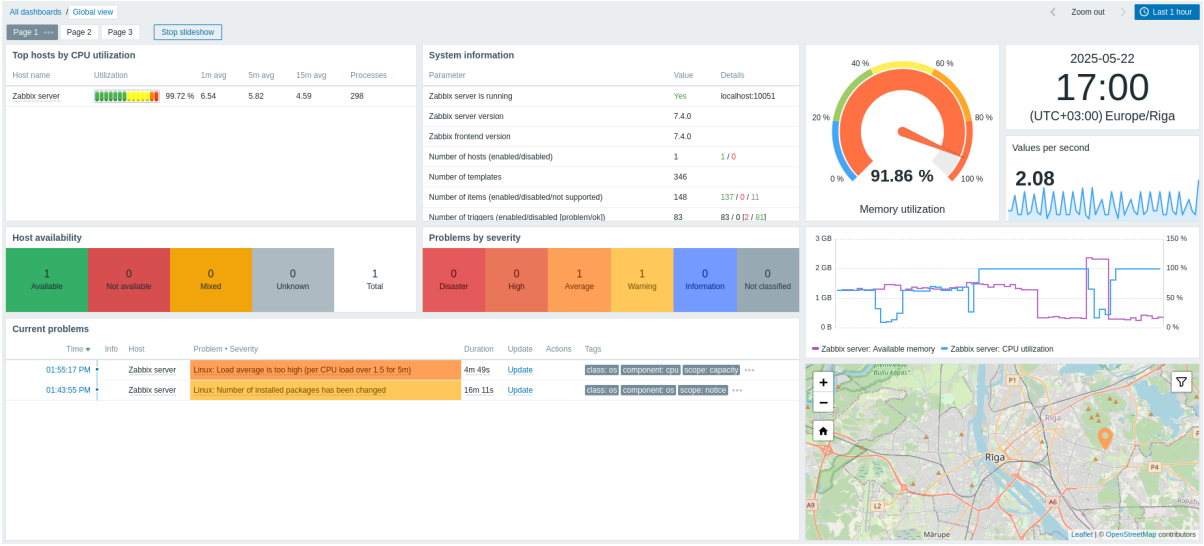
Możesz ustawić typ łącza i kolor dla każdego wskaźnika bezpośrednio z listy. Po zakończeniu kliknij *Zastosuj*, zamknij formularz i kliknij *Aktualizuj*, aby zapisać zmiany mapy.

W *Monitoring* → *Mapy* odpowiedni typ i kolor łącza będą wyświetlane, jeśli **wartość pozycji** osiągnie ustawiony próg (dla typów danych numerycznych) lub będzie zgodna ze wzorcem wyrażenia regularnego (dla typów danych tekstowych).



3 Pulpity

Pulpity — zarówno pulpity globalne, jak i pulpity hostów — zapewniają rozbudowaną platformę wizualizacji z takimi widgetami i narzędziami, jak nowoczesne wykresy, mapy, pokazy slajdów i inne.



8 Szablony i grupy szablonów

Przegląd

Użycie szablonów to doskonały sposób na zmniejszenie nakładu pracy i usprawnienie konfiguracji Zabbix. Szablon to zestaw encji, który można wygodnie zastosować do wielu hostów.

Encje te mogą obejmować:

- pozycje
- wyzwalacze
- wykresy
- pulpity
- reguły wykrywania niskiego poziomu
- scenariusze WWW

Ponieważ wiele hostów w rzeczywistości jest identycznych lub dość podobnych, naturalne jest, że zestaw encji (pozycji, wyzwalaczy, wykresów, ...) utworzony dla jednego hosta może być przydatny dla wielu innych. Oczywiście można skopiować je do każdego nowego hosta, ale wymagałoby to dużo ręcznej pracy. Zamiast tego, korzystając z szablonów, można skopiować je do jednego szablonu, a następnie zastosować ten szablon do dowolnej liczby hostów.

Gdy szablon jest podłączony do hosta, wszystkie encje (pozycje, wyzwalacze, wykresy, ...) ze szablonu są dodawane do hosta. Szablony są przypisywane bezpośrednio do każdego pojedynczego hosta (a nie do grupy hostów).

Szablony są często używane do grupowania encji dla określonych usług lub aplikacji (takich jak Apache, MySQL, PostgreSQL, Postfix...) i następnie stosowane do hostów, na których działają te usługi.

Kolejną korzyścią z używania szablonów jest możliwość wprowadzania zmian dla wszystkich hostów jednocześnie. Jednorazowa zmiana na poziomie szablonu zostanie propagowana do wszystkich powiązanych hostów.

Szablony są zorganizowane w grupy szablonów.

Przejdź do tworzenia i konfigurowania szablonu.

9 Szablony dostępne od razu

Przegląd

Zabbix udostępnia stale rosnący zestaw wstępnie skonfigurowanych szablonów, aby uprościć i przyspieszyć konfigurację monitorowanych obiektów.

Wszystkie gotowe szablony są dostępne w *Data collection* > *Templates*.

Podczas aktualizacji Zabbix istniejące szablony nie są aktualizowane automatycznie, aby uniknąć nadpisania niestandardowych modyfikacji. Aby zaktualizować istniejące szablony lub dodać nowe, zobacz Aktualizacja szablonów.

Skorzystaj z panelu bocznego, aby uzyskać informacje o określonych typach szablonów i wymaganiach operacyjnych.

Zobacz także:

- Podłączanie szablonu
- Znane problemy dotyczące szablonów

Aktualizacja szablonu

Aby zaktualizować pojedynczy szablon lub dodać nowy po aktualizacji Zabbix:

1. Przejdź do repozytorium Git Zabbix.
2. W selektorze gałęzi (pod etykietą *Source*) wybierz gałąź zgodną z Twoją wersją Zabbix (np. *release/7.4* dla Zabbix 7.4).
3. Otwórz szablon, który chcesz dodać lub zaktualizować (np. MySQL by Zabbix agent).
4. Skopiuj zawartość pliku szablonu (np. `template_db_mysql_agent.yaml`) i zapisz ją do lokalnego pliku.
5. W frontend Zabbix przejdź do *Data collection* > *Templates* i zaimportuj lokalny plik.

Możesz też zaktualizować wszystkie szablony i dodać nowe jednocześnie:

1. Pobierz pakiet szablonów odpowiedni dla Twojego serwera WWW: `templates_1M_7.4.7z` (Nginx) lub `templates_2M_7.4.7z` (Apache).
2. Rozpakuj pakiet. Zawiera on pliki YAML (każdy z wieloma szablonami) oraz indeks (`index-1M_74.md`) z listą szablonów zawartych w każdym pliku YAML.
3. W frontend Zabbix przejdź do *Data collection* > *Templates* i zaimportuj pliki YAML jeden po drugim.

Alternatywnie możesz zaimportować wszystkie pliki YAML naraz, używając skryptu `import_templates.sh`. Skrypt wymaga narzędzi `curl` i `jq` oraz korzysta z metody API Zabbix `configuration.import`. Podczas uruchamiania skryptu podaj katalog zawierający pliki YAML oraz adres URL API Zabbix i token uwierzytelniający.

Warning:
Skrypt `import_templates.sh` nadpisuje wszystkie istniejące szablony, bez możliwości przeglądu lub cofnięcia zmian w szablonach.

Przykład:

```
./import_templates.sh /tmp/templates_2M_7.4 https://example.com/zabbix/api_jsonrpc.php d8d6b5c78ee2a8333db
```

Zgodność szablonu z Host Wizard

Pełną listę gotowych szablonów, które są obecnie zgodne z Host Wizard, znajdziesz w pliku `hw_compatible_templates.txt`.

Możesz również dostosować własne szablony tak, aby były zgodne z Host Wizard:

1. W frontend Zabbix przejdź do *Data collection* > *Templates* i wyeksportuj własny szablon.
2. Edytuj wyeksportowany plik:
   - Dodaj element szablonu `wizard_ready` template element, aby wskazać zgodność z Host Wizard.
   - Opcjonalnie dodaj element szablonu `readme` lub `config` template element, jeśli jest to potrzebne.
3. W frontend Zabbix przejdź do *Data collection* > *Templates* i zaimportuj zaktualizowany plik szablonu.

1 Operacja szablonu Zabbix agent

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą Zabbix agent:

1. Upewnij się, że Zabbix agent jest zainstalowany na hoście. W przypadku kontroli aktywnych upewnij się również, że adres serwera/proxy Zabbix został dodany do parametru 'ServerActive' w pliku konfiguracyjnym agenta.
2. Połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała współdzielenie danych z Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- Apache by Zabbix agent
- Apache by Zabbix agent active
- HAProxy by Zabbix agent
- IIS by Zabbix agent
- IIS by Zabbix agent active

- Microsoft Exchange Server 2016 by Zabbix agent
- Microsoft Exchange Server 2016 by Zabbix agent active
- MySQL by Zabbix agent
- MySQL by Zabbix agent active
- Nginx by Zabbix agent
- Nginx by Zabbix agent active
- PHP-FPM by Zabbix agent
- PHP-FPM by Zabbix agent active
- PostgreSQL by Zabbix agent
- PostgreSQL by Zabbix agent active
- RabbitMQ cluster by Zabbix agent

2 Operacja szablonu Zabbix agent 2

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą Zabbix agent 2:

1. Upewnij się, że agent 2 jest zainstalowany na hoście i że zainstalowana wersja zawiera wymagany plugin. W niektórych przypadkach może być konieczne wcześniejsze uaktualnienie agenta 2.
2. Powiąż szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu. Zwróć uwagę, że makra użytkownika mogą służyć do nadpisywania parametrów konfiguracji.
4. Skonfiguruj monitorowaną instancję tak, aby zezwalała na współdzielenie danych z Zabbix.

Attention:
Szablony Zabbix agent 2 działają w połączeniu z pluginami. Podstawową konfigurację można wykonać, po prostu dostosowując makra użytkownika, natomiast głębszą personalizację można osiągnąć przez samą konfigurację pluginu. Na przykład, jeśli plugin obsługuje nazwane sesje, można monitorować kilka encji tego samego typu (np. MySQL1 i MySQL2), określając w pliku konfiguracyjnym dla każdej encji nazwaną sesję z własnym URI, nazwą użytkownika i hasłem.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- Ceph by Zabbix agent 2
- Docker
- Memcached
- MongoDB cluster by Zabbix agent 2
- MongoDB node by Zabbix agent 2
- MySQL by Zabbix agent 2
- MySQL by Zabbix agent 2 active
- Oracle by Zabbix agent 2
- PostgreSQL by Zabbix agent 2
- PostgreSQL by Zabbix agent 2 active
- Redis
- SMART by Zabbix agent 2
- SMART by Zabbix agent 2 active
- Systemd by Zabbix agent 2
- Website certificate by Zabbix agent 2

3 Operacja szablonu HTTP

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą HTTP agent:

1. Utwórz host w Zabbix i jako główny interfejs wskaż adres IP lub nazwę DNS monitorowanego obiektu. Jest to potrzebne, aby makro {HOST.CONN} mogło zostać poprawnie rozwiązane w pozycjach szablonu.
2. Połącz szablon z hostem utworzonym w kroku 1 (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała udostępnianie danych z Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- Acronis Cyber Protect Cloud by HTTP
- Apache by HTTP
- Asterisk by HTTP
- AWS by HTTP
- AWS Cost Explorer by HTTP
- AWS EC2 by HTTP
- AWS ECS Cluster by HTTP
- AWS ECS Serverless Cluster by HTTP
- AWS ELB Application Load Balancer by HTTP
- AWS ELB Network Load Balancer by HTTP
- AWS Lambda by HTTP
- AWS RDS instance by HTTP
- AWS S3 bucket by HTTP
- Azure by HTTP
- Cisco Meraki organization by HTTP
- Cisco SD-WAN by HTTP
- Cisco Secure Firewall Threat Defense by HTTP
- ClickHouse by HTTP
- Cloudflare by HTTP
- CockroachDB by HTTP
- Control-M enterprise manager by HTTP
- Control-M server by HTTP
- DELL PowerEdge R720 by HTTP
- DELL PowerEdge R740 by HTTP
- DELL PowerEdge R820 by HTTP
- DELL PowerEdge R840 by HTTP
- Elasticsearch Cluster by HTTP
- Envoy Proxy by HTTP
- Etcd by HTTP
- FortiGate by HTTP
- GitHub repository by HTTP
- GitLab by HTTP
- Google Cloud Platform (GCP) by HTTP
- Hadoop by HTTP
- HAProxy by HTTP
- HashiCorp Consul Cluster by HTTP
- HashiCorp Consul Node by HTTP
- HashiCorp Nomad by HTTP
- HashiCorp Vault by HTTP
- Hikvision camera by HTTP
- HPE iLO by HTTP
- HPE MSA 2040 Storage by HTTP
- HPE MSA 2060 Storage by HTTP
- HPE Primera by HTTP
- HPE Synergy by HTTP
- InfluxDB by HTTP
- Jenkins by HTTP
- Kubernetes API server by HTTP
- Kubernetes cluster state by HTTP
- Kubernetes Controller manager by HTTP
- Kubernetes kubelet by HTTP
- Kubernetes nodes by HTTP
- Kubernetes Scheduler by HTTP
- MantisBT by HTTP
- Microsoft 365 reports by HTTP
- Microsoft SharePoint by HTTP
- NetApp AFF A700 by HTTP
- Nextcloud by HTTP

- NGINX by HTTP
- NGINX Plus by HTTP
- Nutanix Prism Element by HTTP
- OpenStack by HTTP
- OpenWeatherMap by HTTP
- Oracle Cloud by HTTP
- Palo Alto PA-440 by HTTP
- PHP-FPM by HTTP
- Proxmox VE by HTTP
- Pure Storage FlashArray v1 and v2 by HTTP
- RabbitMQ cluster by HTTP
- Ribbon SBC Edge by HTTP
- Ribbon SBC SWe CE by HTTP
- Ribbon SBC SWe Core by HTTP
- TiDB by HTTP
- TiDB PD by HTTP
- TiDB TiKV by HTTP
- Travis CI by HTTP
- Veeam Backup Enterprise Manager by HTTP
- Veeam Backup and Replication by HTTP
- VeloCloud SD-WAN by HTTP
- VeloCloud SD-WAN Edge by HTTP
- YugabyteDB by HTTP
- ZooKeeper by HTTP

4 Operacja szablonu IPMI

Szablony IPMI nie wymagają żadnej specjalnej konfiguracji. Aby rozpocząć monitorowanie, połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępny szablon:

- Chassis by IPMI

5 Operacja szablonu JMX

Kroki zapewniające prawidłowe działanie szablonów, które zbierają metryki przez JMX:

1. Upewnij się, że Zabbix Java gateway jest zainstalowany i poprawnie skonfigurowany.
2. Połącz szablon z docelowym hostem. Host powinien mieć skonfigurowany interfejs JMX.
 Jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu.
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała udostępnianie danych Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- Apache ActiveMQ by JMX
- Apache Cassandra by JMX
- Apache Kafka by JMX
- Apache Tomcat by JMX
- GridGain by JMX
- Ignite by JMX
- Jira Data Center by JMX
- WildFly Domain by JMX
- WildFly Server by JMX

6 Operacja szablonu ODBC

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą monitorowania ODBC:

1. Upewnij się, że wymagany sterownik ODBC jest zainstalowany na serwerze Zabbix lub proxy.
2. Połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała udostępnianie danych Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- MSSQL by ODBC
- MySQL by ODBC
- Oracle by ODBC
- PostgreSQL by ODBC

7 Standaryzowane szablony dla urządzeń sieciowych

Przegląd

Aby zapewnić monitorowanie urządzeń sieciowych, takich jak przełączniki i routery, utworzyliśmy dwa tak zwane modele: dla samego urządzenia sieciowego (zasadniczo jego obudowy) oraz dla interfejsu sieciowego.

Dostępne są szablony dla wielu rodzin urządzeń sieciowych. Wszystkie szablony obejmują (tam, gdzie możliwe jest pobranie tych pozycji z urządzenia):

- Monitorowanie usterek obudowy (zasilacze, wentylatory i temperatura, ogólny stan)
- Monitorowanie wydajności obudowy (pozycje CPU i pamięci)
- Zbieranie danych inwentaryzacyjnych obudowy (numery seryjne, nazwa modelu, wersja firmware)
- Monitorowanie interfejsów sieciowych za pomocą IF-MIB i EtherLike-MIB (stan interfejsu, obciążenie ruchem interfejsu, stan dupleksu dla Ethernet)

Jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne najpierw jego zaimportowanie.

Jeśli importujesz nowe gotowe szablony, możesz również chcieć zaktualizować globalne wyrażenie regularne `@Network interfaces for discovery` do postaci:

```
Result is FALSE: ^Software Loopback Interface
Result is FALSE: ^(In)?[lL]oop[bB]ack[0-9._]*$
Result is FALSE: ^NULL[0-9.]*$
Result is FALSE: ^[lL]o[0-9.]*$
Result is FALSE: ^[sS]ystem$
Result is FALSE: ^Nu[0-9.]*$
```

aby odfiltrować interfejsy loopback i null w większości systemów.

Urządzenia

Lista rodzin urządzeń, dla których dostępne są szablony:

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Alcatel Timetra TiMOS SNMP	Alcatel	Alcatel Timetra	ALCATEL SR 7750	TiMOS	TIMETRA-SYSTEM-MIB,TIMETRA-CHASSIS-MIB	Certified
Aruba CX 8300s by SNMP	HPE Aruba Networking	Seria przełączników Aruba CX	Aruba JL636A 8325, Aruba JL717A 8360	Aruba AOS-CX	ARUBAWIRED-FAN-MIB, ARUBAWIRED-POWERSUPPLY-MIB, ARUBAWIRED-SYSTEMINFO-MIB, ARUBAWIRED-TEMPSENSOR-MIB, OSPF-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Brocade FC SNMP	Brocade	Przełączniki Brocade FC	Brocade 300 SAN Switch-	-	SW-MIB,ENTITY-MIB	Performance Fault
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 Nonstackable SNMP	Brocade, Foundry	Brocade MLX, Foundry	Brocade MLXe, Foundry FLS648, Foundry FWSX424		FOUNDRY-SN-AGENT-MIB	Performance Fault
Check Point Next Generation Firewall by SNMP	Check Point	Zapora nowej generacji	-	Gaia	HOST-RESOURCES-MIB, CHECKPOINT-MIB, UCD-SNMP-MIB, SNMPv2-MIB, IF-MIB	Certified
Ciena 3906 by SNMP	Ciena	Ciena CPE	Ciena 3906	SAOS	WWP-LEOS-BLADE-MIB, WWP-LEOS-CHASSIS-MIB, WWP-LEOS-SYSTEM-CONFIG-MIB	Certified
Cisco Catalyst 3750<device model> 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 versions 12.0_3_T-12.2_3.5 SNMP	Cisco	Cisco IOS > 12.0 3 T i 12.2 3.5	-	IOS	CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB	Certified
Cisco IOS prior to 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	Przełączniki DES/DGX	D-Link DES-xxxx/DGS-xxxx,DLINK DGS-3420-26SC	-	DLINK-AGENT-MIB,EQUIPMENT-MIB,ENTITY-MIB	Certified
D-Link DES 7200 SNMP	D-Link	DES-7xxx	D-Link DES 7206	-	ENTITY-MIB,MY-SYSTEM-MIB,MY-PROCESS-MIB,MY-MEMORY-MIB	Performance Fault Interfaces
Dell Force S-Series SNMP	Dell	Dell Force S-Series	S4810		F10-S-SERIES-CHASSIS-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Extreme Exos SNMP	Extreme	Extreme EXOS	X670V-48x	EXOS	EXTREME-SYSTEM-MIB,EXTREME-SOFTWARE-MONITOR-MIB	Certified
FortiGate by SNMP	Fortinet	FortiGate (NGFW)	-	FortiOS	HOST-RESOURCES-MIB FORTINET-FORTIGATE-MIB FORTINET-CORE-MIB SNMPv2-MIB IF-MIB ENTITY-MIB	Performance Inventory
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	Przełącznik HP Enterprise	HP ProCurve J4900B Switch 2626, HP J9728A 2920-48G Switch		STATISTICS-MIB,NETSWITCH-MIB,HP-ICF-CHASSIS,ENTITY-MIB,SEMI-MIB	Certified
Huawei AR600 by SNMP	Huawei	Routery serii Huawei NetEngine AR600	Huawei AR600/AR6100/AR6200/AR6300 series routers with VRP software (V300R019/V300R021/V300R023 or later)		HUAWEI-CBQOS-MIB, HUAWEI-NQA-MIB, ENTITY-MIB, IF-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certified
Huawei OceanStor 5300 V5 by SNMP	Huawei	Huawei OceanStor Dorado	Huawei OceanStor Dorado with V5 software		HUAWEI-STORAGE-HARDWARE-MIB, HUAWEI-STORAGE-SPACE-MIB, ISM-PERFORMANCE-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certified
Huawei OceanStor Dorado by SNMP	Huawei	Huawei OceanStor, Huawei OceanStor Dorado	Huawei OceanStor Dorado 3000/5000/6000/8000/18000/5300V5/5500/5600/6800/18500/18800 with V6/V700 software		HUAWEI-STORAGE-HARDWARE-MIB, HUAWEI-STORAGE-SPACE-MIB, ISM-PERFORMANCE-MIB, HOST-RESOURCES-MIB, SNMPv2-MIB	Certified
Huawei VRP by SNMP	Huawei	Huawei VRP	S2352P-EI		ENTITY-MIB,HUAWEI-ENTITY-EXTENT-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Intel_Qlogic Infiniband SNMP	Intel/QLogic	Urządzenia Intel/QLogic Infiniband	Infiniband 12300		ICS-CHASSIS-MIB	Fault Inventory
Juniper SNMP	Juniper	Modele MX,SRX,EX	Juniper MX240, Juniper EX4200-24F	JunOS	JUNIPER-MIB	Certified
Juniper MX NETCONF	Juniper	Modele MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	*Nie dotyczy (używa NETCONF przez SSH)*	Certified
Juniper MX SNMP	Juniper	Modele MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	OSPF-MIB, JUNIPER-DOM-MIB, JUNIPER-MIB, BGP4-V2-MIB-JUNIPER, OSPFV3-MIB-JUNIPER	Certified
Mellanox SNMP	Mellanox	Urządzenia Mellanox Infiniband	SX1036	MLNX-OS	HOST-RESOURCES-MIB,ENTITY-MIB,ENTITY-SENSOR-MIB,MELLANOX-MIB	Certified
MikroTik CCR<device model> SNMP	MikroTik	Routery MikroTik Cloud Core (seria CCR)	Dostępne są oddzielne dedykowane szablony dla 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

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System opera-cyjny	Używane MIB-y	**Tagi**
MikroTik CRS<device model> SNMP	MikroTik	Przełączniki MikroTik Cloud Router (seria CRS)	Dostępne są oddzielne dedykowane szablony dla 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	Certified
MikroTik CSS<device model> SNMP	MikroTik	Przełączniki MikroTik Cloud Smart (seria CSS)	Dostępne są oddzielne dedykowane szablony dla MikroTik CSS326-24G-2S+RM, MikroTik CSS610-8G-2S+IN	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik FiberBox SNMP	MikroTik	MikroTik FiberBox	MikroTik FiberBox	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik hEX <device model> SNMP	MikroTik	MikroTik hEX	Dostępne są oddzielne dedykowane szablony dla MikroTik hEX, MikroTik hEX lite, MikroTik hEX PoE, MikroTik hEX PoE lite, MikroTik hEX S	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System opera-cyjny	Używane MIB-y	**Tagi**
MikroTik netPower <device model> SNMP	MikroTik	MikroTik net-Power	Dostępne są oddzielne dedykowane szablony dla MikroTik netPower 15FR, MikroTik netPower 16P SNMP, MikroTik netPower Lite 7R	RouterOS/SwitchOS, SwitchOS Lite	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik PowerBox <device model> SNMP	MikroTik	MikroTik Power-Box	Dostępne są oddzielne dedykowane szablony dla MikroTik PowerBox, MikroTik PowerBox Pro	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik RB<device model> SNMP	MikroTik	Routery serii MikroTik RB	Dostępne są oddzielne dedykowane szablony dla MikroTik RB1100AHx4, MikroTik RB1100AHx4 Dude Edition, MikroTik RB2011iL-IN, MikroTik RB2011iL-RM, MikroTik RB2011iLS-IN, MikroTik RB2011UiAS-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	Urządzenia MikroTik RouterOS	MikroTik CCR1016-12G, MikroTik RB2011UAS-2HnD, MikroTik 912UAG-5HPnD, MikroTik 941-2nD, MikroTik 951G-2HnD, MikroTik 1100AHx2	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Netgear Fastpath SNMP	Netgear	Netgear Fastpath	M5300-28G		FASTPATH-SWITCHING-MIB,FASTPATH-BOXSERVICES-PRIVATE-MIB	Fault Inventory
QTech QSW SNMP	QTech	Urządzenia Qtech	Qtech QSW-2800-28T	-	QTECH-MIB,ENTITY-MIB	Performance Inventory
Stormshield SNS by SNMP	Stormshield	Zapory Stormshield Network Security (SNS)	SN3100	Stormshield SNS	HOST-RESOURCES-MIB, UCD-SNMP-MIB, STORMSHIELD-ASQ-STATS-MIB, STORMSHIELD-AUTOUPDATE-MIB, STORMSHIELD-HA-MIB, STORMSHIELD-PROPERTY-MIB, STORMSHIELD-HEALTH-MONITOR-MIB, STORMSHIELD-IF-MIB, STORMSHIELD-SYSTEM-MONITOR-MIB, STORMSHIELD-IPSEC-STATS-MIB	Certified
TP-LINK SNMP	TP-LINK	TP-LINK	T2600G-28TS v2.0		TPLINK-SYSMONITOR-MIB,TPLINK-SYSINFO-MIB	Performance Inventory
Ubiquiti AirOS SNMP	Ubiquiti	Bezprzewodowe urządzenia Ubiquiti AirOS	NanoBridge,NanoStation,Unifi	AirOS	FROGFOOT-RESOURCES-MIB,IEEE802dot11-MIB	Performance
Vyatta Virtual Router by SNMP	Ciena	Vyatta	Vyatta Virtual Router 1908e	Vyatta 1908e	SNMPv2-MIB, HOST-RESOURCES-MIB, UCD-SNMP-MIB, IF-MIB, DISMAN-EVENT-MIB	Performance Inventory

Projekt szablonów

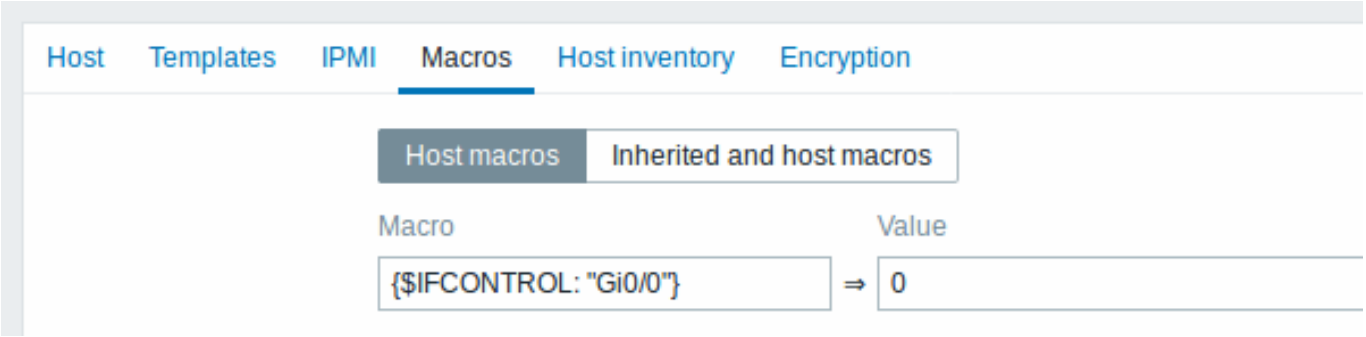
Szablony zostały zaprojektowane z uwzględnieniem następujących założeń:

- Makra użytkownika są używane w możliwie największym zakresie, aby wyzwalacze mogły być dostrajane przez użytkownika;
- Mechanizm wykrywania niskiego poziomu jest używany w możliwie największym zakresie, aby zminimalizować liczbę nieobsługiwanych pozycji;
- Wszystkie szablony zależą od Template ICMP Ping, dzięki czemu wszystkie urządzenia są również sprawdzane przez ICMP;
- Pozycje nie używają żadnych MIB-ów — w pozycjach i regułach wykrywania niskiego poziomu używane są OID-y SNMP. Dzięki temu nie ma potrzeby ładowania żadnych MIB-ów do Zabbix, aby szablony działały;
- Interfejsy sieciowe loopback są filtrowane podczas wykrywania, podobnie jak interfejsy z ifAdminStatus = down(2)
- Tam, gdzie to możliwe, używane są liczniki 64-bitowe z IF-MIB::ifXTable. Jeśli nie są obsługiwane, zamiast nich używane są domyślne liczniki 32-bitowe.

Wszystkie wykryte interfejsy sieciowe mają wyzwalacz monitorujący ich stan operacyjny (link), na przykład:

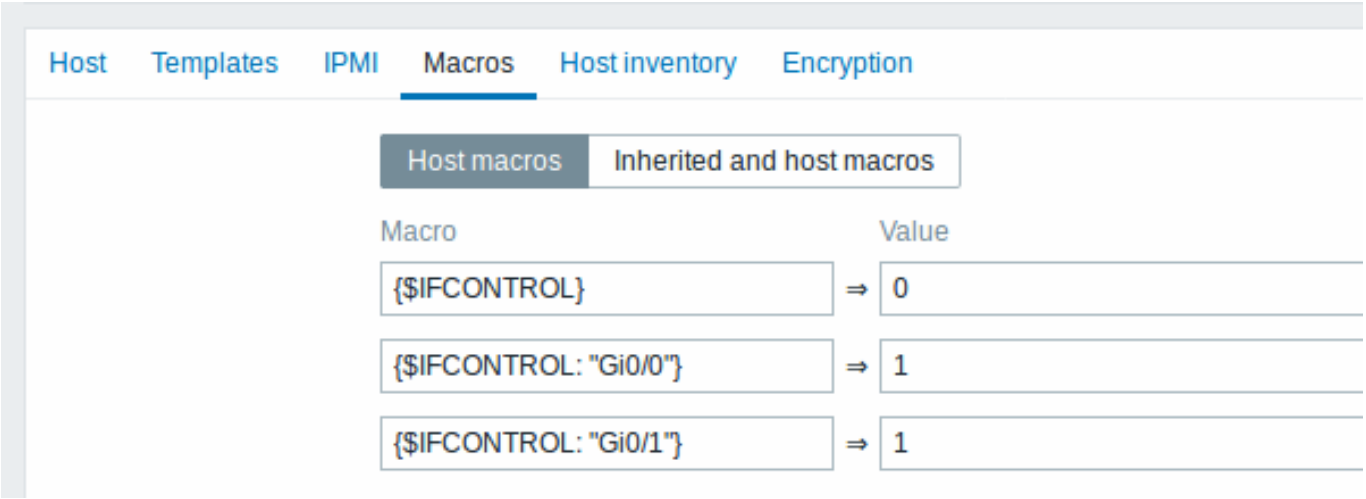
```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/Alcatel Timetra TiMOS SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])
```

- Jeśli nie chcesz monitorować tego warunku dla określonego interfejsu, utwórz makro użytkownika z kontekstem o wartości 0. Na przykład:



gdzie Gi0/0 to {#IFNAME}. W ten sposób wyzwalacz nie będzie już używany dla tego konkretnego interfejsu.

- Możesz również zmienić domyślne zachowanie wszystkich wyzwalaczy tak, aby się nie uruchamiały, i aktywować ten wyzwalacz tylko dla ograniczonej liczby interfejsów, takich jak uplinki:



Tagi

- Wydajność - MIB-y rodziny urządzeń zapewniają sposób monitorowania pozycji procesora i pamięci;
- Błędy - MIB-y rodziny urządzeń zapewniają sposób monitorowania co najmniej jednego czujnika temperatury;
- Inwentaryzacja - MIB-y rodziny urządzeń zapewniają sposób zbierania co najmniej numeru seryjnego urządzenia i nazwy modelu;
- Certyfikowane - wszystkie trzy główne kategorie powyżej są objęte.

8 Działanie szablonu VMware

Przegląd

Zabbix zawiera zestaw gotowych do użycia szablonów do monitorowania maszyn wirtualnych, przeznaczonych dla hipernadzorców VMware vCenter i ESXi. Dostępne szablony są zawarte w dwóch oddzielnych zestawach szablonów:

- VMware - używa danych UUID dla odpowiednich makr
- VMware FQDN - używa danych FQDN dla odpowiednich makr

Szablony przeznaczone do monitorowania hipernadzorców VMware vCenter lub ESXi zawierają wstępnie skonfigurowane reguły wykrywania niskiego poziomu oraz różne wbudowane kontrole do monitorowania instalacji wirtualnych.

Note:
Aby szablon *VMware FQDN* działał poprawnie, każda monitorowana maszyna wirtualna powinna mieć unikalną nazwę systemu operacyjnego zgodną z zasadami FQDN. Dodatkowo na każdej maszynie muszą być zainstalowane VMware Tools/Open Virtual Machine tools. Jeśli te wymagania są spełnione, zaleca się użycie szablonu *VMware FQDN*. Szablon *VMware FQDN* jest dostępny od Zabbix 5.2, wraz z wprowadzeniem możliwości tworzenia hostów z niestandardowymi interfejsami.

 Dostępny jest również klasyczny szablon *VMware*, którego można użyć, jeśli wymagania FQDN nie są spełnione. Szablon *VMware* ma jednak znany problem. Hosty dla wykrytych maszyn wirtualnych są tworzone z nazwami zapisanymi w vCenter (na przykład „VM1", „VM2" itd.). Jeśli na tych hostach jest zainstalowany agent Zabbix, a aktywna autorejestracja agenta Zabbix jest włączona, proces autorejestracji odczyta nazwy hostów tak, jak zostały zarejestrowane podczas uruchomienia (na przykład „vm1.example.com", „vm2.example.com" itd.). Może to prowadzić do tworzenia nowych hostów dla istniejących maszyn wirtualnych (ponieważ nie znaleziono dopasowania nazw), co skutkuje duplikatami hostów o różnych nazwach.

Proszę pamiętać:

- Szablon *VMware Hypervisor* można ręcznie powiązać z hostem, a także używać go w wykrywaniu;
- Szablon *VMware Guest* nie powinien być ręcznie powiązany z hostem i może być używany wyłącznie w wykrywaniu.

Konfiguracja makr hosta

Aby używać prostych kontroli VMware, host musi mieć zdefiniowane następujące makra użytkownika:

- {$VMWARE.URL} - adres URL SDK usługi VMware (vCenter lub hiperwizora ESXi) (https://servername/sdk)
- {$VMWARE.USERNAME} - nazwa użytkownika usługi VMware
- {$VMWARE.PASSWORD} - hasło użytkownika {$VMWARE.USERNAME} usługi VMware

10 Powiadomienia o zdarzeniach

Przegląd

Zakładając, że skonfigurowaliśmy już pewne pozycje i wyzwalacze, a teraz otrzymujemy zdarzenia występujące w wyniku zmiany stanu wyzwalaczy, nadszedł czas, aby rozważyć działania.

Na początek nie chcielibyśmy cały czas wpatrywać się w listę wyzwalaczy lub zdarzeń. Znacznie lepiej byłoby otrzymywać powiadomienie, jeśli wydarzyło się coś istotnego (na przykład problem). Ponadto, gdy wystąpią problemy, chcielibyśmy mieć pewność, że wszystkie zainteresowane osoby zostały poinformowane.

Dlatego wysyłanie powiadomień jest jednym z podstawowych działań oferowanych przez Zabbix. Można określić, kto i kiedy powinien zostać powiadomiony o określonym zdarzeniu.

Aby móc wysyłać i odbierać powiadomienia z Zabbix, należy:

- zdefiniować media
- skonfigurować działanie, które wysyła wiadomość do jednego ze zdefiniowanych mediów

Działania składają się z *warunków* i *operacji*. Zasadniczo, gdy warunki są spełnione, wykonywane są operacje. Dwie główne operacje to wysłanie wiadomości (powiadomienia) oraz wykonanie zdalnego polecenia.

W przypadku zdarzeń utworzonych przez wykrywanie i autorejestrację dostępne są dodatkowe operacje. Obejmują one dodawanie lub usuwanie hosta, podłączanie szablonu itp.

1 Typy mediów

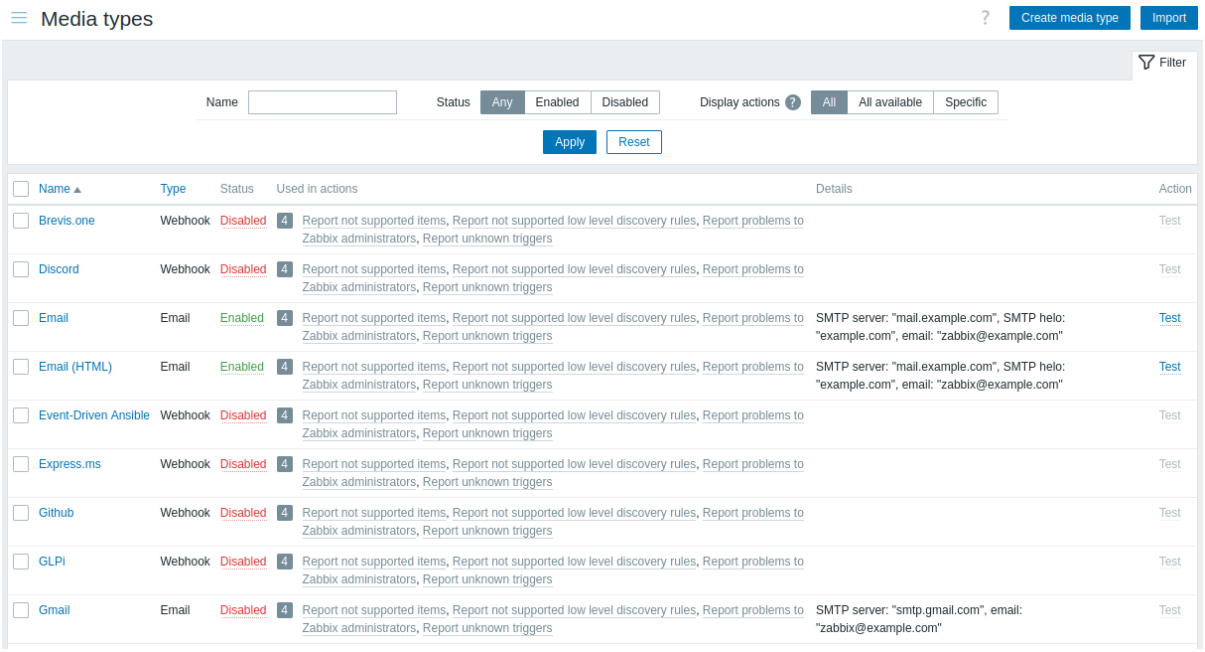
Przegląd

Typy mediów to kanały dostarczania używane do wysyłania powiadomień i alertów z Zabbixa.

Typy mediów obsługują następujące metody dostarczania:

- Email
- SMS
- Custom script
- Webhook

Typy mediów są zarządzane w *Alerts* > *Media types*. Niektóre typy mediów są wstępnie zdefiniowane w domyślnym zestawie danych. Wystarczy dostosować ich parametry, aby zaczęły działać.



Aby zobaczyć, jak typy mediów wpisują się w proces alertowania, przyjrzyjmy się trzem wymaganiom potrzebnym do dostarczania powiadomień z Zabbixa do użytkowników końcowych:

1. Musi być zdefiniowana operacja action, która wysyła powiadomienia
2. Musi być zdefiniowany działający media type (na przykład *Email*, który wysyła alerty za pomocą SMTP)
3. Szczegóły dostarczania na poziomie użytkownika (takie jak adresy e-mail, numery telefonów itp.) muszą być zdefiniowane w user media

Testowanie typów mediów

Aby sprawdzić, czy skonfigurowany typ mediów działa, kliknij *Test* na liście typów mediów.

Żądanie testowe zostanie wysłane do serwera Zabbix. Serwer Zabbix spróbuje wysłać alert przy użyciu określonego typu mediów i zwróci wynik do frontend. Frontend będzie oczekiwać na zwrócenie wyników przez serwer. Testowanie typów mediów ma domyślnie limit czasu 65 sekund (można go skonfigurować w *Administracja* > *Ogólne* > *Timeouts*).

Zobacz także szczegóły testowania dla:

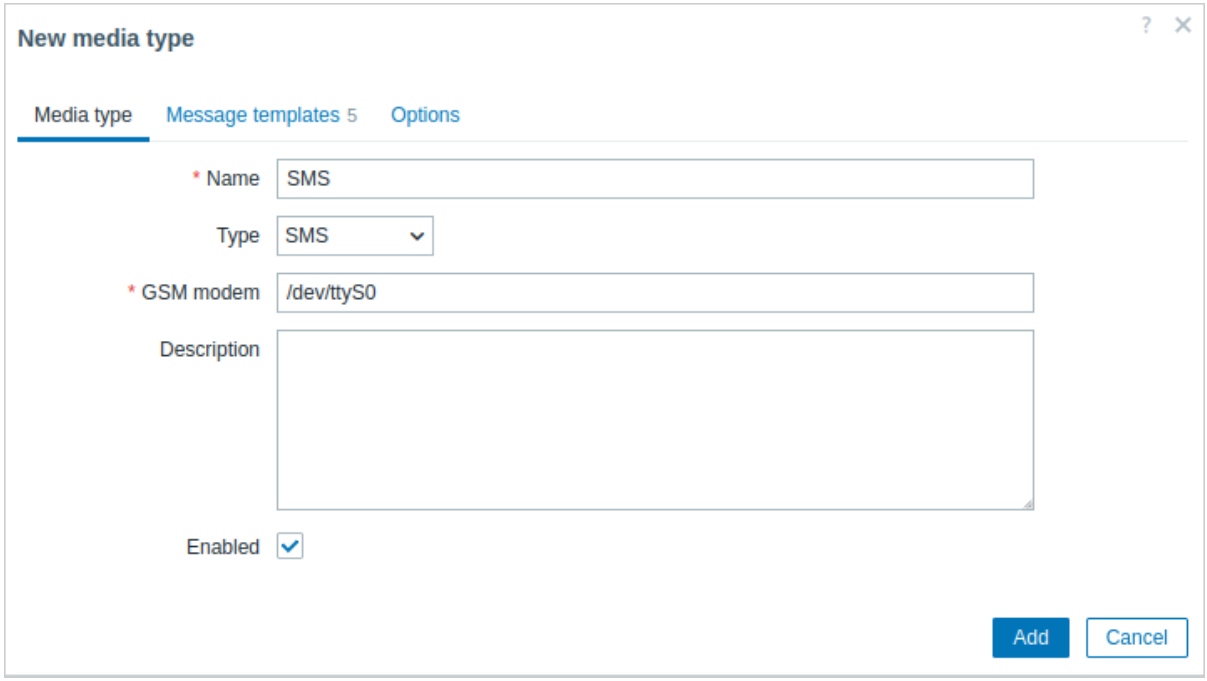
- Email
- Webhook
- Script

Konfiguracja

Aby utworzyć typ mediów w frontend Zabbix:

- Przejdź do: *Alerts* > *Media types*
- Kliknij *Create media type*
- Wprowadź parametry typu mediów w formularzu

Niektóre parametry są **wspólne** dla wszystkich metod dostarczania.

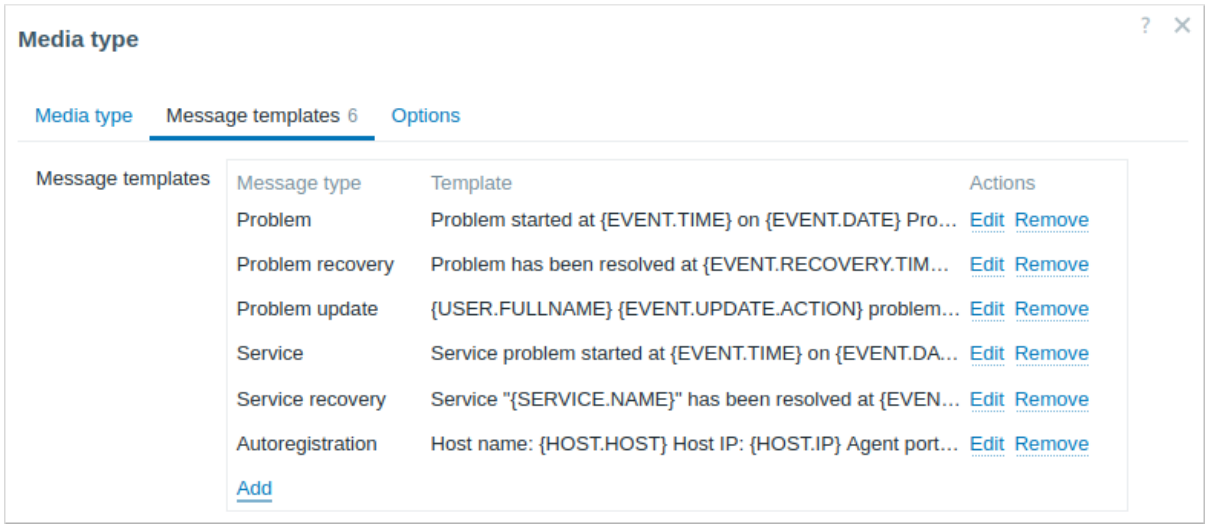


Parameter	Description
Name	Nazwa typu mediów.
Type	Wybierz metodę dostarczania dla typu mediów.
Description	Wprowadź opis typu mediów.
Enabled	Zaznacz pole wyboru, aby włączyć typ mediów.

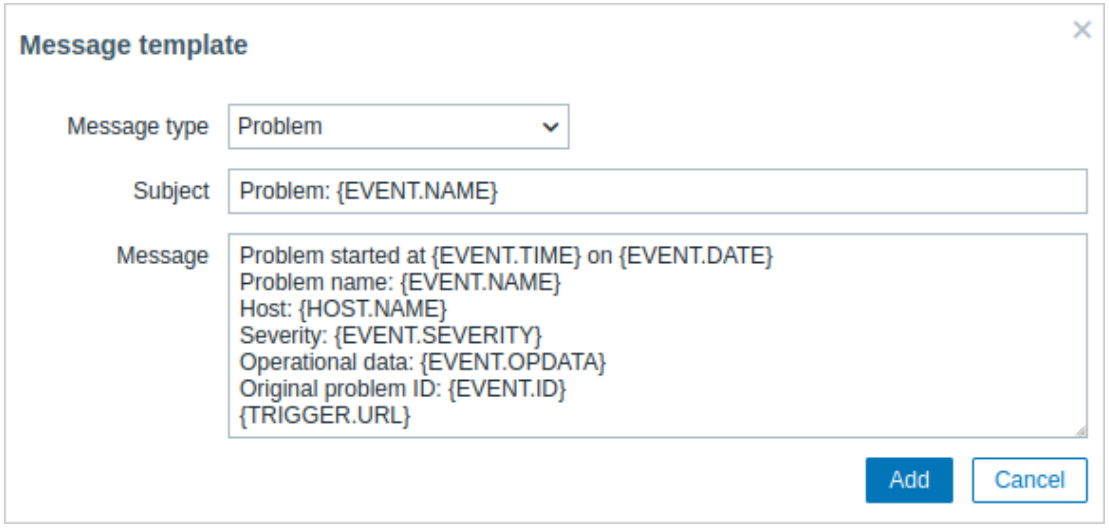
Informacje o parametrach specyficznych dla danej metody znajdują się na stronach email, SMS, custom alertscript lub webhook.

Szablony wiadomości

Karta **Szablony wiadomości** zawiera domyślne wiadomości dla poszczególnych typów zdarzeń (problem, odzyskanie po problemie, wykrywanie itp.).



Kliknij **Dodaj**, aby zdefiniować domyślną wiadomość (lub **Edytuj**, aby zaktualizować istniejącą wiadomość):



Parameter	Description
Message type	Typ zdarzenia, dla którego ma być używana domyślna wiadomość. Dla każdego typu zdarzenia można zdefiniować tylko jedną domyślną wiadomość.
Subject	Temat domyślnej wiadomości. Temat może zawierać makra. Jego długość jest ograniczona do 255 znaków. Temat nie jest dostępny dla typu nośnika SMS.
Message	Domyślna wiadomość. Jej długość jest ograniczona do określonej liczby znaków w zależności od typu bazy danych (więcej informacji można znaleźć w sekcji Wysyłanie wiadomości). Wiadomość może zawierać obsługiwane makra. W wiadomościach o problemie i aktualizacji problemu obsługiwane są makra wyrażeń (na przykład `{?avg(/host/key,1h)}`).

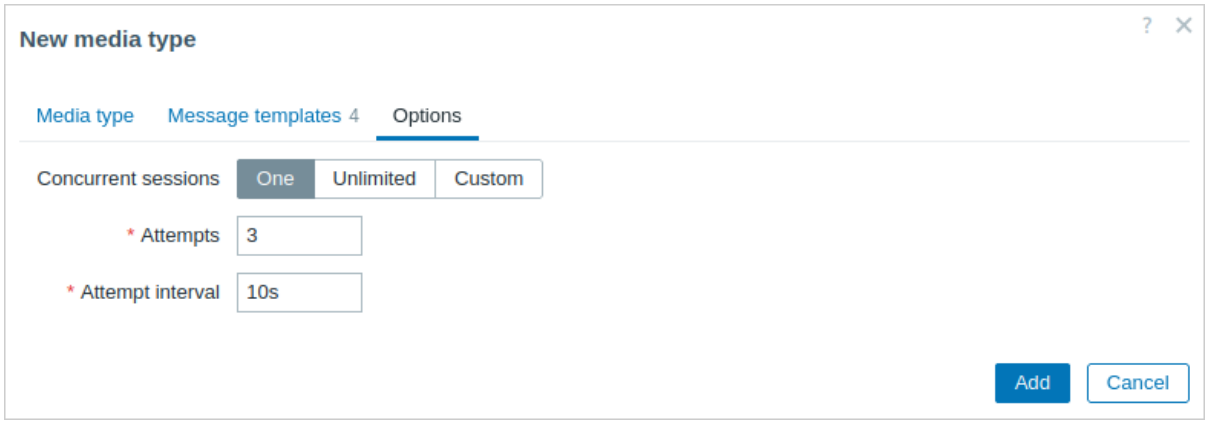
Należy pamiętać, że domyślne wiadomości są zastępowane przez niestandardowe wiadomości, jeśli zostały zdefiniowane w operacjach akcji.

Warning:
Definiowanie szablonów wiadomości jest obowiązkowe dla wszystkich metod dostarczania, w tym webhooków lub niestandardowych skryptów alertów, które nie używają domyślnych wiadomości do powiadomień. Na przykład akcja „Wyślij wiadomość do webhooka Pushover" nie wyśle powiadomień o problemie, jeśli wiadomość o problemie dla webhooka Pushover nie została zdefiniowana.

Opcje

Zakładka **Opcje** zawiera ustawienia przetwarzania alertów. Ten sam zestaw opcji można skonfigurować dla każdego typu mediów.

Wszystkie typy mediów są przetwarzane równolegle. Maksymalna liczba równoczesnych sesji jest konfigurowalna dla każdego typu mediów, jednak łączna liczba procesów alertera na serwerze może być ograniczona wyłącznie przez parametr StartAlerters. Alerty wygenerowane przez jeden wyzwalacz są przetwarzane sekwencyjnie. Dlatego wiele powiadomień może być przetwarzanych jednocześnie tylko wtedy, gdy zostały wygenerowane przez wiele wyzwalaczy.



Parametr	Opis
Równoczesne sesje	Wybierz liczbę równoległych sesji alertera dla typu mediów: **Jedna** - jedna sesja **Nieograniczona** - nieograniczona liczba sesji **Niestandardowa** - wybierz niestandardową liczbę sesji Nieograniczone/wysokie wartości oznaczają więcej równoległych sesji i zwiększoną wydajność wysyłania powiadomień. Nieograniczone/wysokie wartości powinny być używane w dużych środowiskach, w których może być konieczne jednoczesne wysyłanie wielu powiadomień. Jeśli do wysłania będzie więcej powiadomień niż dostępnych równoczesnych sesji, pozostałe powiadomienia zostaną umieszczone w kolejce; nie zostaną utracone.
Próby	Liczba prób wysłania powiadomienia. Można określić maksymalnie 100 prób; wartość domyślna to „3”. Jeśli określono „1”, Zabbix wyśle powiadomienie tylko raz i nie będzie ponawiał próby, jeśli wysyłanie się nie powiedzie.
Interwał prób	Częstotliwość ponawiania próby wysłania powiadomienia w przypadku niepowodzenia wysyłania, w sekundach (0-3600). Jeśli określono „0”, Zabbix ponowi próbę natychmiast. Obsługiwane są sufiksy czasu, np. 5s, 3m, 1h.

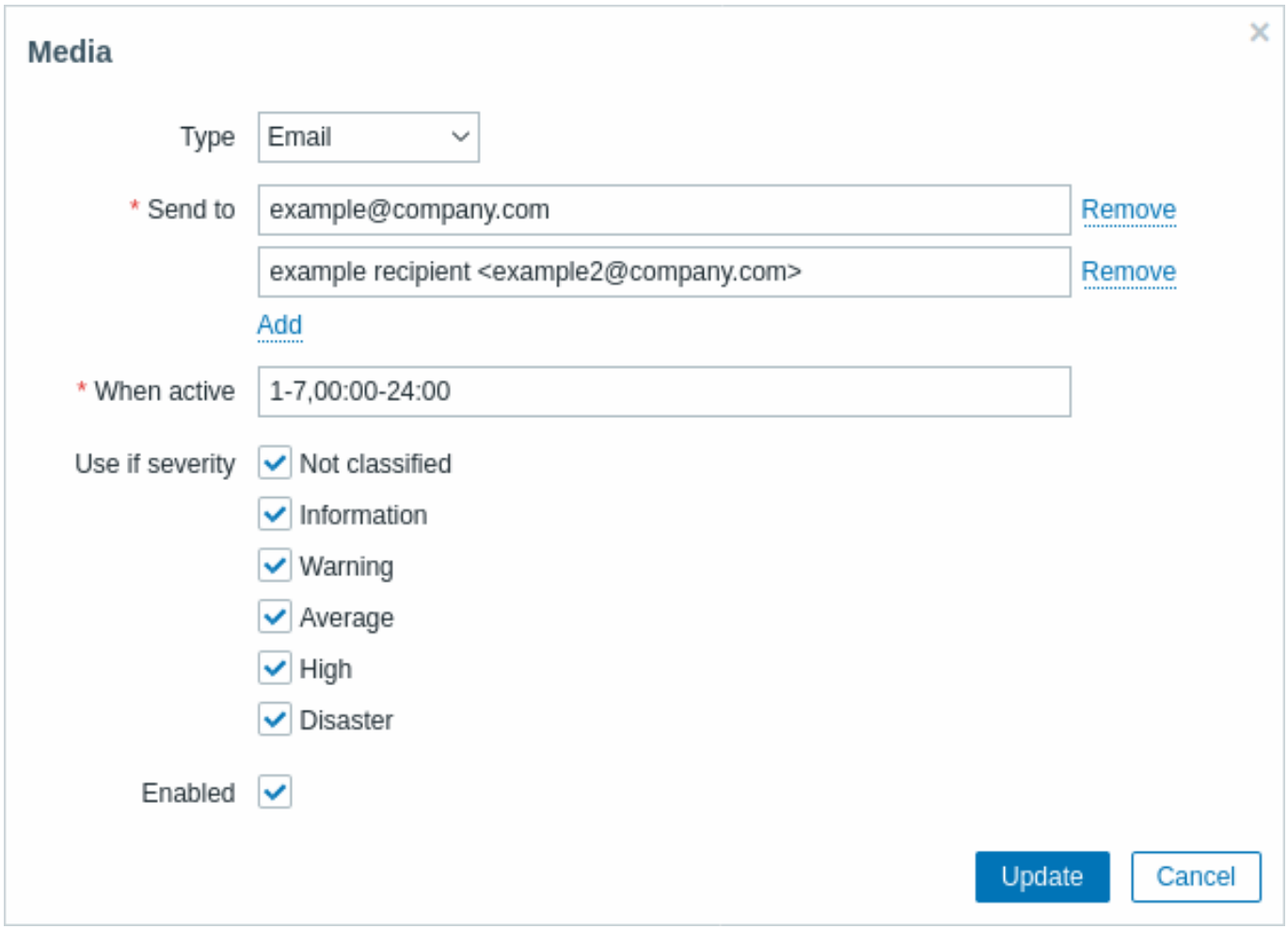
Media użytkownika

Podczas gdy typy mediów określają **jak** zostanie wysłane powiadomienie, media użytkownika określają **dokąd** powiadomienie musi zostać wysłane.

Media użytkownika (np. adres e-mail, identyfikator użytkownika webhooka itp.) muszą być zdefiniowane w profilu użytkownika niezależnie od metody dostarczenia. Akcja wysyłająca wiadomości do użytkownika *Admin* za pomocą webhooka *X* zakończy się niepowodzeniem, jeśli szczegóły dostarczenia webhooka *X* nie są zdefiniowane w profilu użytkownika Admin.

Aby zdefiniować media użytkownika:

- Przejdź do *Users* > *Users* i otwórz formularz właściwości użytkownika (lub przejdź do *User settings* > *Notifications* własnego profilu użytkownika)
- Kliknij *Add* na karcie Media



Parameter	Description
Type	Lista rozwijana zawiera nazwy włączonych typów mediów. Zwróć uwagę, że podczas edycji medium wyłączonego typu medium, typ będzie wyświetlany na czerwono.
Send to	Wprowadź dane kontaktowe, na które mają być wysyłane wiadomości. W przypadku typu medium e-mail można dodać wiele adresów, klikając przycisk Add poniżej pola adresu. W takim przypadku powiadomienia zostaną wysłane na wszystkie wymienione adresy. Przykłady adresów znajdziesz w opisie parametru *Email* dla typu medium email.
When active	Możesz ograniczyć czas wysyłania wiadomości, na przykład ustawić tylko dni robocze (1-5,09:00-18:00). Zwróć uwagę, że to ograniczenie jest oparte na użytkowniczej strefie czasowej. Jeśli strefa czasowa użytkownika zostanie zmieniona i będzie inna niż systemowa strefa czasowa, to ograniczenie może wymagać odpowiedniego dostosowania, aby nie przegapić ważnych wiadomości. Opis formatu znajduje się na stronie Specyfikacja okresu czasu. Obsługiwane są makra użytkownika.
Use if severity	Zaznacz pola wyboru dla poziomów ważności wyzwalaczy, dla których chcesz otrzymywać powiadomienia. *Uwaga* domyślny poziom ważności ('Not classified') **musi być** zaznaczony, jeśli chcesz otrzymywać powiadomienia dla zdarzeń events innych niż wyzwalacze. Po zapisaniu wybrane poziomy ważności wyzwalaczy będą wyświetlane w odpowiadających im kolorach, a niewybrane zostaną wyszarzone.
Status	Status mediów użytkownika. **Enabled** - jest używane. **Disabled** - nie jest używane.

1 Email

Przegląd

Aby skonfigurować e-mail jako kanał dostarczania wiadomości, należy skonfigurować e-mail jako typ mediów i przypisać użytkownikom określone adresy.

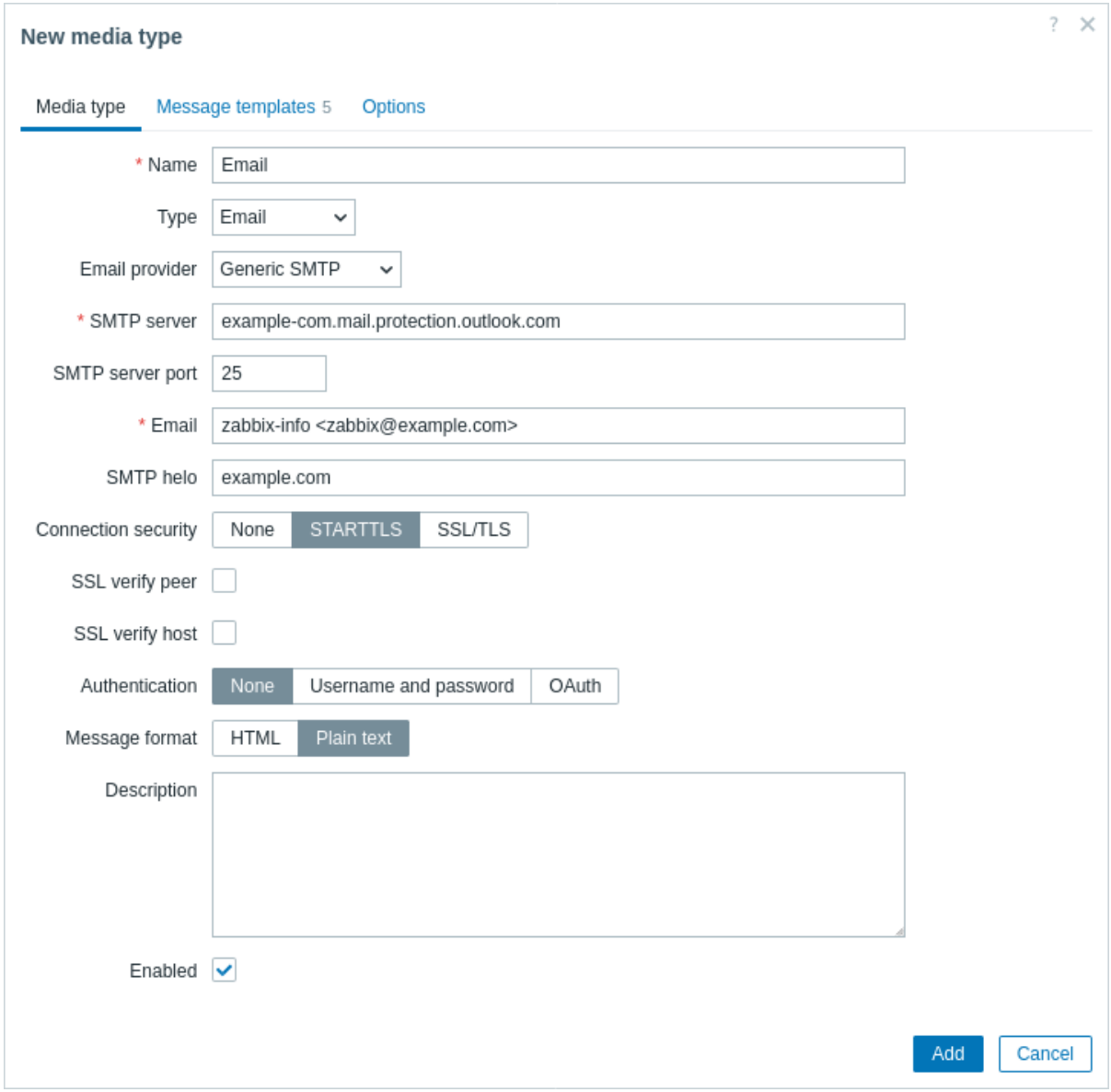
> **Note:**
> Wiele powiadomień dla pojedynczego zdarzenia zostanie zgrupowanych w tym samym wątku e-mail.

Konfiguracja

Aby skonfigurować email jako typ nośnika:

1. Przejdź do *Alerts* > *Media types*.
2. Kliknij *Create media type* (lub kliknij *Email* na liście predefiniowanych typów nośnika).

Karta **Media type** zawiera ogólne atrybuty typu nośnika:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Warning:
Hasło jest eksportowane jako zwykły tekst podczas eksportowania konfiguracji typu nośnika email.

Poniższe parametry są specyficzne dla typu nośnika email:

Parameter	Description
Email provider	Wybierz dostawcę poczty email: *Generic SMTP*, *Gmail*, *Gmail relay*, *Office365* lub *Office365 relay*. Jeśli wybierzesz opcje związane z Gmail/Office365, wystarczy podać adres email nadawcy i hasło; takie opcje jak *SMTP server*, *SMTP server port*, *SMTP helo* oraz *Connection security* zostaną automatycznie uzupełnione przez Zabbix. Zobacz także: Automated Gmail/Office365 media types.
SMTP server	Ustaw serwer SMTP do obsługi wiadomości wychodzących. To pole jest dostępne, jeśli jako dostawca poczty email wybrano *Generic SMTP*.
SMTP server port	Ustaw port serwera SMTP do obsługi wiadomości wychodzących. To pole jest dostępne, jeśli jako dostawca poczty email wybrano *Generic SMTP*.

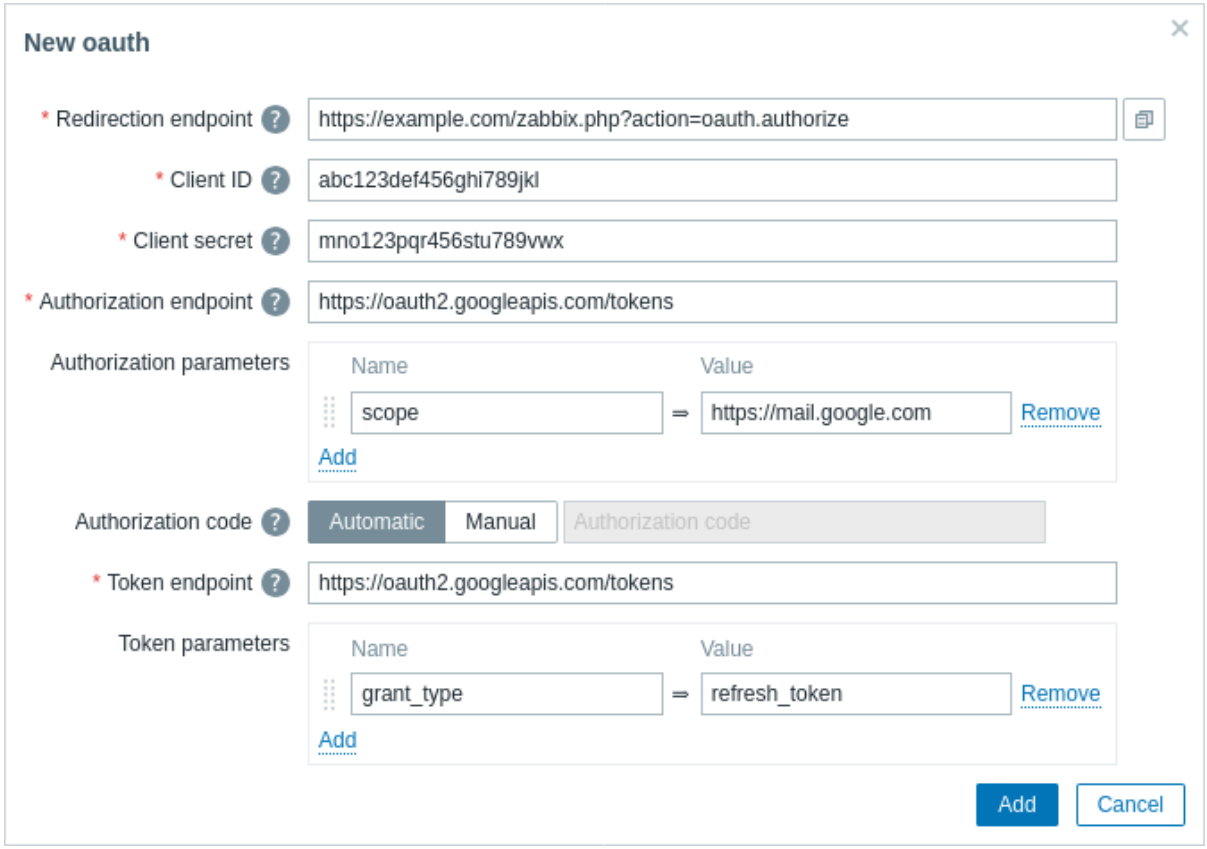
Parameter	Description
Email	Adres wpisany tutaj będzie używany jako adres **From** dla wysyłanych wiadomości. Obsługiwane jest dodanie wyświetlanej nazwy nadawcy (takiej jak ”Zabbix_info” w *Zabbix_info <zabbix@company.com>* na powyższym zrzucie ekranu) wraz z rzeczywistym adresem email. W Zabbix obowiązują pewne ograniczenia dotyczące nazw wyświetlanych w wiadomościach email w porównaniu z tym, co dopuszcza RFC 5322, co ilustrują przykłady: Poprawne przykłady: *zabbix@company.com* (sam adres email, bez nawiasów ostrych) *Zabbix_info <zabbix@company.com>* (nazwa wyświetlana i adres email w nawiasach ostrych) *∑Ω-monitoring <zabbix@company.com>* (znaki UTF-8 w nazwie wyświetlanej) Niepoprawne przykłady: *Zabbix HQ zabbix@company.com* (nazwa wyświetlana obecna, ale bez nawiasów ostrych wokół adresu email) *”Zabbix\@\<H(comment)Q\>” <zabbix@company.com>* (choć poprawne według RFC 5322, pary cytowane i komentarze nie są obsługiwane w wiadomościach email Zabbix)
SMTP helo	Ustaw poprawną wartość SMTP helo, zwykle nazwę domeny. Jeśli pole jest puste, zostanie wysłana nazwa domeny adresu email (czyli to, co znajduje się po @ w polu *Email*). Jeśli nie uda się pobrać nazwy domeny, zostanie zapisane ostrzeżenie na poziomie debug, a jako domena dla polecenia HELO zostanie wysłana nazwa hosta serwera. To pole jest dostępne, jeśli jako dostawca poczty email wybrano *Generic SMTP*.
Connection security	Wybierz poziom bezpieczeństwa połączenia: **None** - nie używaj opcji CURLOPT_USE_SSL **STARTTLS** - użyj opcji CURLOPT_USE_SSL z wartością CURLUSESSL_ALL **SSL/TLS** - użycie CURLOPT_USE_SSL jest opcjonalne
SSL verify peer	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera SMTP. Wartość dyrektywy konfiguracji serwera ”SSLCALocation” powinna zostać przekazana do CURLOPT_CAPATH w celu walidacji certyfikatu. Ustawia to opcję cURL CURLOPT_SSL_VERIFYPEER.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub pole *Subject Alternate Name* certyfikatu serwera SMTP jest zgodne. Ustawia to opcję cURL CURLOPT_SSL_VERIFYHOST.
Authentication	Wybierz poziom uwierzytelniania: **None** - nie są ustawiane żadne opcje cURL **Username and password** - oznacza ”AUTH=*”, pozostawiając wybór mechanizmu uwierzytelniania cURL **OAuth** - uwierzytelnianie OAuth Uwierzytelnianie OAuth nie jest obsługiwane dla dostawcy poczty email *Office365 relay*.
Username	Nazwa użytkownika używana do uwierzytelniania. Ustawia wartość CURLOPT_USERNAME. Obsługiwane makra użytkownika.
Password	Hasło używane do uwierzytelniania. Ustawia wartość CURLOPT_PASSWORD. Obsługiwane makra użytkownika. Wartość wpisana tutaj jest eksportowana bez zmian podczas eksportowania konfiguracji typu nośnika. Aby uniknąć ujawnienia poufnych danych uwierzytelniających w eksportowanych plikach, użyj makra użytkownika (na przykład `{$EMAIL_SMTP_PASSWORD}`) zamiast jawnego hasła — odwołanie do makra zostanie wyeksportowane, a wartość tajna powinna zostać skonfigurowana ponownie w systemie docelowym.
OAuth tokens	Kliknij *Configure*, aby w nowym oknie ustawić parametry pobierania OAuth tokens. To pole jest dostępne tylko wtedy, gdy w polu *Authentication* wybrano opcję ”OAuth”.
Message format	Wybierz format wiadomości: **HTML** - wysyłaj jako HTML **Plain text** - wysyłaj jako zwykły tekst

Attention:
Aby włączyć uwierzytelnianie SMTP, serwer Zabbix musi zostać skompilowany z opcją kompilacji `--with-libcurl`, która wymaga libcurl (szczegóły wersji znajdują się w wymaganiach dla serwer).

Zobacz także common media type parameters, aby uzyskać szczegółowe informacje o konfigurowaniu domyślnych wiadomości i opcji przetwarzania alertów.

Tokeny OAuth

Do pobrania tokenów OAuth wymagane są następujące parametry:



Parametr	Opis
Punkt końcowy przekierowania	Wprowadź adres URL frontend Zabbix, do którego usługa OAuth przekieruje z powrotem autoryzację OAuth (w formacie `https://<zabbix-frontend-url>/zabbix.php?action=oauth.authorize`). Dla nowego typu nośnika e-mail jest on ustawiany automatycznie, gdy zdefiniowano adres URL frontend.
ID klienta	Wprowadź unikalny identyfikator aplikacji klienckiej zarejestrowanej na serwerze autoryzacji OAuth.
Tajny klucz klienta	Wprowadź prywatny tajny klucz aplikacji klienckiej zarejestrowanej na serwerze autoryzacji OAuth.
Punkt końcowy autoryzacji	Wprowadź adres URL serwera autoryzacji OAuth służący do żądania autoryzacji użytkownika.
Parametry autoryzacji	Wprowadź parametry punktu końcowego autoryzacji.
Kod autoryzacji	Wprowadź kod autoryzacji: **Automatycznie** - kod zostanie pobrany automatycznie za pośrednictwem strony przekierowania **Ręcznie** - wprowadź kod ręcznie, jeśli automatyczne pobranie się nie powiedzie
Punkt końcowy tokenu	Wprowadź adres URL serwera autoryzacji OAuth służący do wymiany kodu autoryzacji na tokeny dostępu i odświeżania.
Parametry tokenu	Wprowadź parametry tokenu dostępu.

Note:
Pobieranie tokenów OAuth ma zautomatyzowane funkcje dla dostawców poczty e-mail **Gmail**, **Gmail relay** i **Office365**. Wymagane jest jedynie podanie wartości parametrów *Punkt końcowy przekierowania*, *ID klienta* i *Tajny klucz klienta*. Zabbix automatycznie uzupełni pozostałe wymagane wartości (zobacz Domyślne adresy URL OAuth według dostawcy).

Pobieranie tokenów dostępu i odświeżania OAuth

Po przesłaniu parametrów OAuth:

1. Otwierane jest wyskakujące okno przeglądarki, które kieruje użytkownika do *punktu końcowego autoryzacji*.
2. Użytkownik autoryzuje Zabbix w usłudze OAuth.

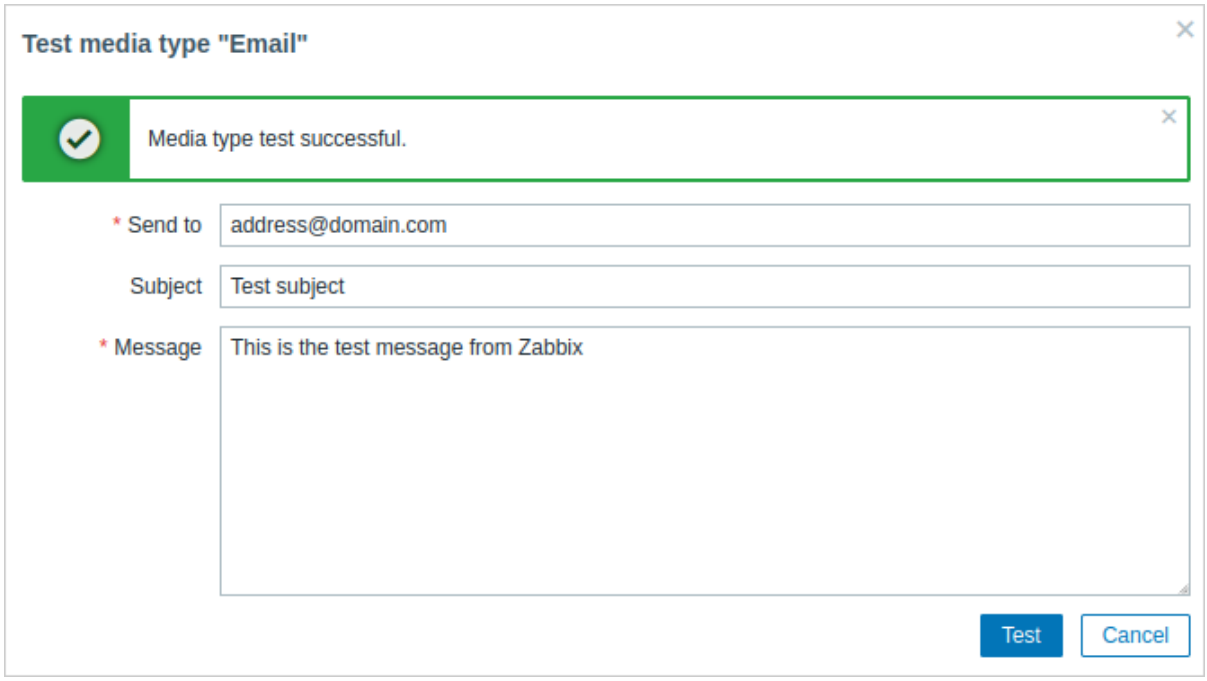
3. Usługa OAuth przekierowuje użytkownika do akcji frontendu Zabbix `oauth.authorize` z kodem autoryzacyjnym i wartością zakresu.
4. W odpowiedzi Zabbix wyśle żądanie do *punktu końcowego tokenu*, aby wymienić kod autoryzacyjny na token dostępu i token odświeżania.

Testowanie

Aby sprawdzić, czy skonfigurowany typ nośnika e-mail działa poprawnie:

1. Znajdź odpowiednią wiadomość e-mail na liście typów nośników.
2. Kliknij *Test* w ostatniej kolumnie listy (otworzy się okno testowania).
3. Wprowadź adres odbiorcy w polu *Send to*, treść wiadomości oraz opcjonalnie temat.
4. Kliknij *Test*, aby wysłać wiadomość testową.

Komunikat o powodzeniu lub niepowodzeniu testu zostanie wyświetlony w tym samym oknie:



Media użytkownika

Po skonfigurowaniu typu mediów e-mail przejdź do sekcji *Użytkownicy* > *Użytkownicy* i edytuj profil użytkownika, aby przypisać użytkownikowi media e-mail. Kroki konfiguracji mediów użytkownika, wspólne dla wszystkich typów mediów, są opisane na stronie Typy mediów.

1 Zautomatyzowane typy mediów Gmail/Office365

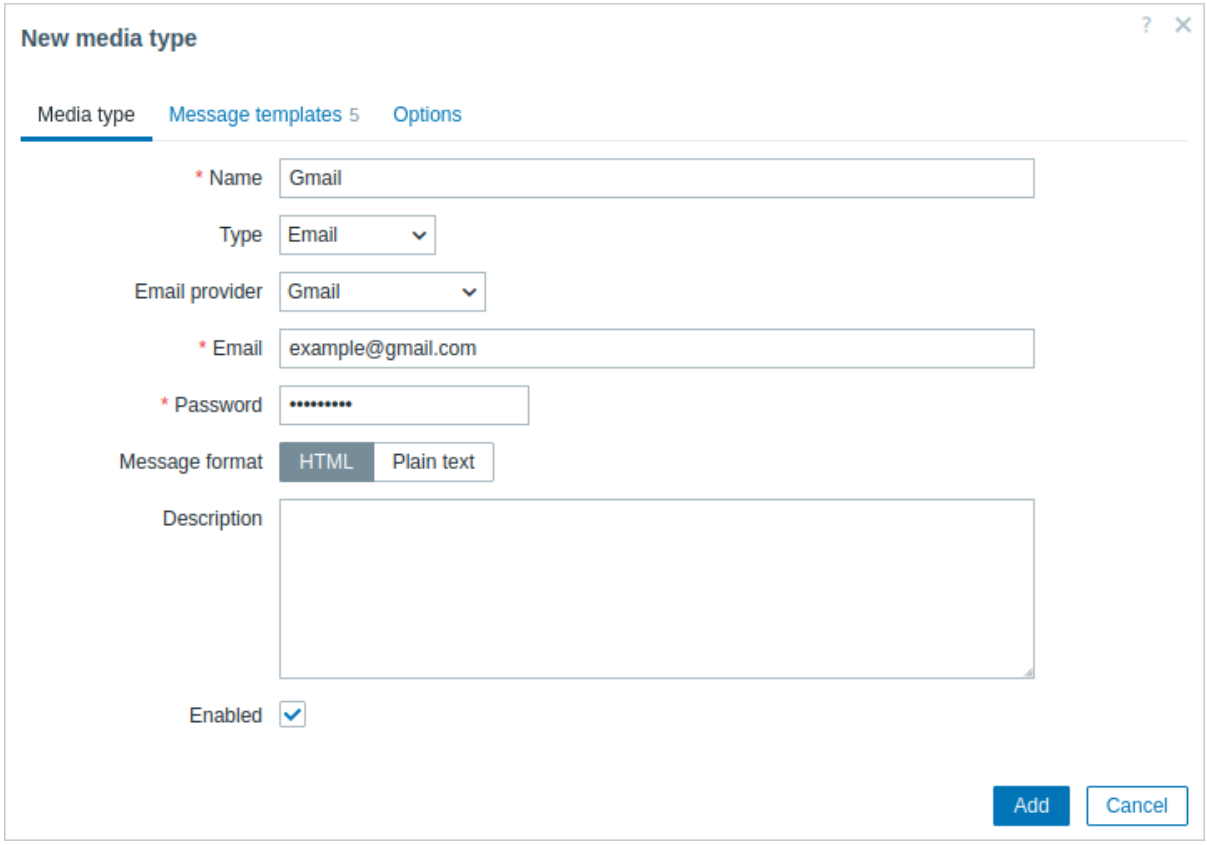
Przegląd

Użytkownicy Gmaila lub Office365 mogą skorzystać z automatycznych funkcji w konfiguracji typu mediów.

Konfiguracja

Pole *Dostawca poczty e-mail* w konfiguracji typu nośnika e-mail umożliwia wybór wstępnie skonfigurowanych opcji dla Gmail i Office 365.

Po wybraniu opcji związanych z Gmail/Office365, aby utworzyć działający typ nośnika, wystarczy podać adres e-mail nadawcy oraz hasło.



Po podaniu adresu e-mail/hasła Zabbix będzie mógł automatycznie uzupełnić wszystkie wymagane ustawienia dla typów nośników Gmail/Office365 rzeczywistymi/zalecanymi wartościami, tj. *Serwer SMTP*, *Port serwera SMTP*, *SMTP helo* oraz *Bezpieczeństwo połączenia*. Ze względu na tę automatyzację pola te nie są nawet wyświetlane, jednak możliwe jest sprawdzenie serwera SMTP i szczegółów wiadomości e-mail na liście typów nośników (zobacz kolumnę *Szczegóły*).

Należy również pamiętać, że:

- Hasło nie jest wymagane dla opcji relay.
- W przypadku relay Office365 nazwa domeny podanego adresu e-mail zostanie użyta do dynamicznego uzupełnienia serwera SMTP (tj. zastąpienia „example.com" w `example-com.mail.protection.outlook.com` rzeczywistą wartością).

Tokeny OAuth

Pobieranie tokenów OAuth ma zautomatyzowane funkcje dla dostawców poczty e-mail *Gmail*, *Gmail relay* i *Office365*.



Konieczne jest podanie jedynie ograniczonego zestawu parametrów do uzyskania dostępu OAuth oraz pobrania tokenu odświeżania — *Redirection endpoint*, *Client ID* i *Client secret*. Zabbix automatycznie uzupełni pozostałe wymagane wartości (zobacz Domyślne adresy URL OAuth według dostawcy).

Należy pamiętać, że możliwe jest również użycie formularza Generic SMTP do skonfigurowania autoryzacji OAuth dla tych dostawców.

Note:
SmtpClientAuthentication musi być włączone albo dla użytkownika/współdzielonej skrzynki pocztowej, albo dla całej dzierżawy w Office365 (domyślnie nie jest włączone).

Domyślne adresy URL OAuth według dostawcy

Poniższa tabela zawiera domyślne wartości adresów URL OAuth i parametrów dla poszczególnych dostawców używane przez Zabbix.

Parametr	Gmail	Office365	Generic SMTP
Punkt końcowy autoryzacji	`https://accounts.google.com/o/oauth2/auth`	`https://login.microsoftonline.com/common/oauth2/v2.0/authorize`	Brak wartości domyślnej
scope	`https://mail.google.com`	`https://outlook.office.com/SMTP.Send offline_access`	Brak wartości domyślnej
access_type	`offline`	Nieużywane	Brak wartości domyślnej
prompt	`consent`	Nieużywane	Brak wartości domyślnej
redirect_uri	Wartość pola *Redirection endpoint* jest dodawana automatycznie przez kod. Nie jest wymieniona w parametrach *Authorization endpoint*.		
state	Unikalny hash identyfikujący nośnik aktualizowany tokenem OAuth jest dodawany automatycznie przez kod. Nie jest wymieniony w parametrach *Authorization endpoint*.		
Punkt końcowy tokenu	`https://oauth2.googleapis.com/token`	`https://login.microsoftonline.com/common/oauth2/v2.0/token`	Brak wartości domyślnej
grant_type	`authorization_code`	`authorization_code`	Brak wartości domyślnej
redirect_uri	Wartość pola *Redirection endpoint* jest dodawana automatycznie przez kod. Nie jest wymieniona w parametrach *Token endpoint*.		
state	Unikalny hash identyfikujący nośnik aktualizowany tokenem OAuth jest dodawany automatycznie przez kod. Nie jest wymieniony w parametrach *Token endpoint*.		

Obejście dla kont osobistych Office365

Konta osobiste Office365 nie obsługują ciągów zapytań w adresie URL przekierowania.

Aby obejść ten problem, administrator serwera WWW Apache może dodać następującą regułę przepisywania do pliku `.htaccess`:

```
RewriteEngine On
RewriteRule ^/oauth\.authorize$ /zabbix.php?action=oauth.authorize [QSA,L,PT]
```

Umożliwia to użycie uproszczonego adresu URL przekierowania, takiego jak:

```
http://server-name/zabbix/oauth.authorize
```

zamiast adresu URL z parametrami zapytania, co zapewnia zgodność z kontami osobistymi Office365.

2 SMS

Przegląd

Zabbix obsługuje wysyłanie wiadomości SMS przy użyciu szeregowego modemu GSM podłączonego do portu szeregowego serwera Zabbix.

Upewnij się, że:

- Prędkość urządzenia szeregowego (zwykle /dev/ttyS0 w systemie Linux) odpowiada prędkości modemu GSM. Zabbix nie ustawia prędkości połączenia szeregowego. Używa ustawień domyślnych.
- Użytkownik 'zabbix' ma uprawnienia do odczytu/zapisu urządzenia szeregowego. Uruchom polecenie ls -l /dev/ttyS0, aby sprawdzić bieżące uprawnienia urządzenia szeregowego.
- W modemie GSM wprowadzono kod PIN i jest on zachowywany po ponownym włączeniu zasilania. Alternatywnie można wyłączyć kod PIN na karcie SIM. Kod PIN można wprowadzić, wydając polecenie AT+CPIN="NNNN" (NNNN to Twój numer PIN, cudzysłowy muszą być obecne) w programie terminalowym, takim jak Unix minicom lub Windows HyperTerminal.

Zabbix został przetestowany z następującymi modemami GSM:

- Siemens MC35
- Teltonika ModemCOM/G10

Aby skonfigurować SMS jako kanał dostarczania wiadomości, należy również skonfigurować SMS jako typ nośnika oraz wprowadzić odpowiednie numery telefonów dla użytkowników.

Konfiguracja

Aby skonfigurować SMS jako typ nośnika:

- Przejdź do *Alerty* → *Typy nośników*
- Kliknij *Utwórz typ nośnika* (lub kliknij *SMS* na liście predefiniowanych typów nośników).

Następujące parametry są specyficzne dla typu nośnika SMS:

Parametr	Opis
Modem GSM	Ustaw nazwę urządzenia szeregowego modemu GSM. Ścieżka wprowadzona tutaj zostanie zweryfikowana względem parametru serwera SMSDevices (jeśli został określony).

Zobacz wspólne parametry typu nośnika, aby uzyskać szczegółowe informacje o konfigurowaniu domyślnych wiadomości i opcji przetwarzania alertów. Należy pamiętać, że równoległe przetwarzanie wysyłania powiadomień SMS nie jest możliwe.

Media użytkownika

Po skonfigurowaniu typu mediów SMS przejdź do sekcji *Użytkownicy* → *Użytkownicy* i edytuj profil użytkownika, aby przypisać użytkownikowi media SMS. Kroki konfiguracji mediów użytkownika, wspólne dla wszystkich typów mediów, są opisane na stronie Typy mediów.

3 Niestandardowe skrypty alertów

Przegląd

Jeśli nie jesteś zadowolony z istniejących typów mediów do wysyłania alertów, istnieje alternatywny sposób wykonania tego zadania. Możesz utworzyć skrypt, który będzie obsługiwał powiadomienie w wybrany przez Ciebie sposób.

Niestandardowe skrypty alertów są wykonywane na serwerze Zabbix. Skrypty te muszą znajdować się w katalogu określonym w parametrze AlertScriptsPath w pliku konfiguracyjnym serwera.

Oto przykład niestandardowego skryptu alertu:

```
#####!/bin/bash

to=$1
subject=$2
body=$3
host=$4
value=$5

cat <<EOF | mail -s "$subject" "$to"
$body

Host: $host
```

```
Value: $value
EOF
```

Attention:
Zabbix sprawdza kod zakończenia wykonywanych poleceń i skryptów. Każdy kod zakończenia inny niż **0** jest uznawany za błąd wykonania polecenia. W takich przypadkach Zabbix spróbuje ponownie wykonać nieudaną operację.

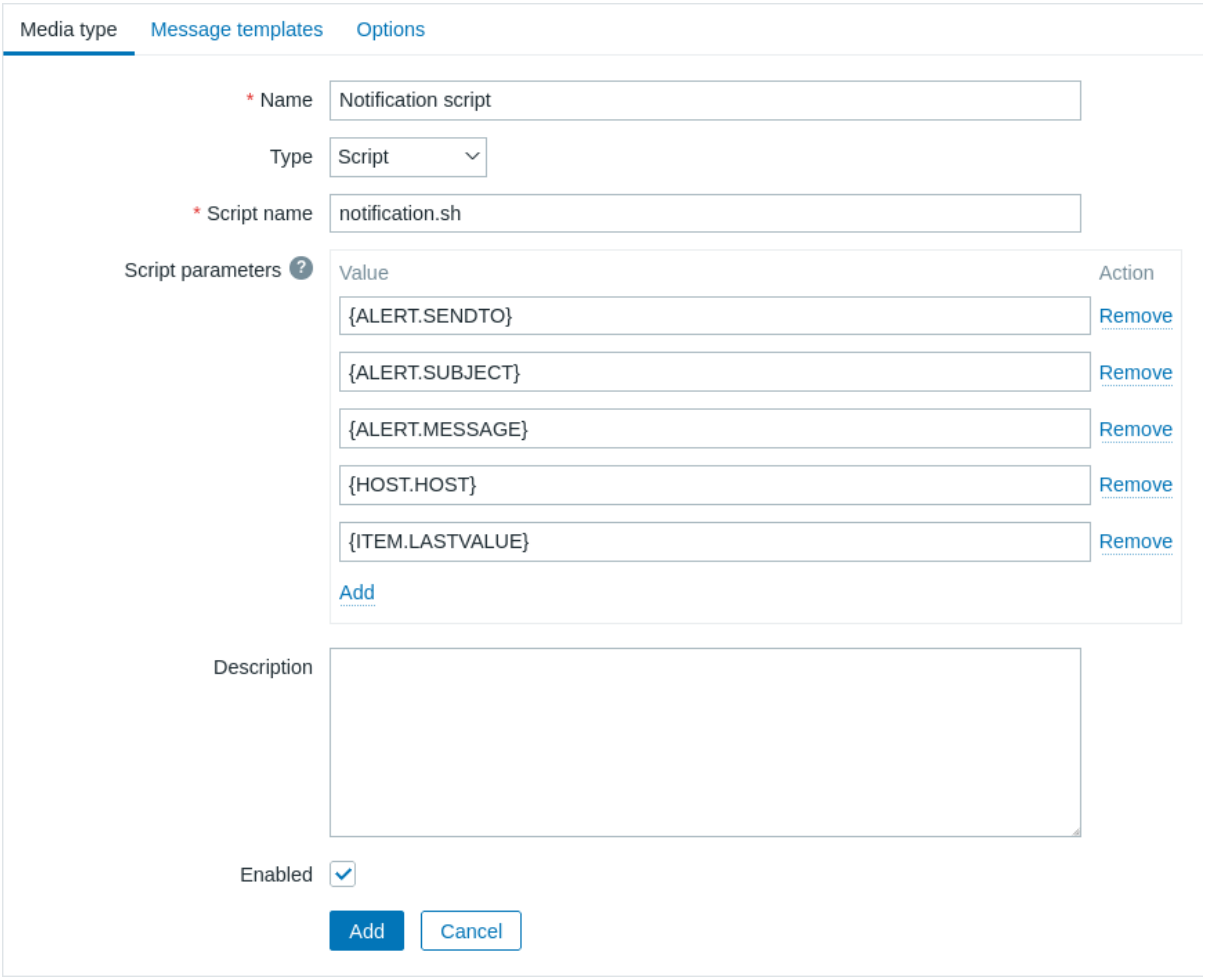
Zmienne środowiskowe nie są zachowywane ani tworzone dla skryptu, więc należy obsłużyć je jawnie.

Konfiguracja

Aby skonfigurować niestandardowe skrypty alertów jako typ mediów:

1. Przejdź do *Alerts* → *Media types*.
2. Kliknij *Create media type*.

Karta **Media type** zawiera ogólne atrybuty typu mediów:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Następujące parametry są specyficzne dla typu mediów skryptu:

Parameter	Description
Script name	Wprowadź nazwę pliku skryptu (np. notification.sh), który znajduje się w katalogu określonym w parametrze konfiguracji serwera AlertScriptsPath.
Script parameters	Dodaj opcjonalne parametry skryptu, które zostaną przekazane do skryptu jako argumenty wiersza poleceń w kolejności, w jakiej zostały zdefiniowane. Parametry skryptu obsługują makra {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} oraz wszystkie makra obsługiwane w powiadomieniach, a także makra użytkownika.

Szczegóły dotyczące konfiguracji domyślnych wiadomości i opcji przetwarzania alertów znajdują się w sekcji common media type parameters.

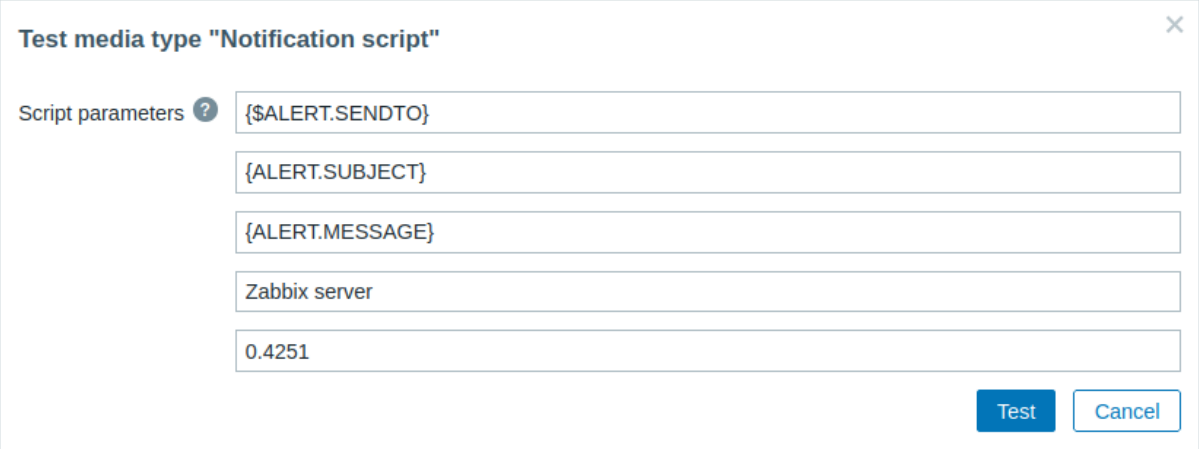
Warning:
Nawet jeśli skrypt alertu nie używa domyślnych wiadomości, szablony wiadomości dla typów operacji używanych przez ten typ mediów nadal muszą być zdefiniowane. W przeciwnym razie powiadomienie nie zostanie wysłane.

Attention:
Jeśli skonfigurowano więcej niż jeden typ mediów skryptu, skrypty te mogą być przetwarzane równolegle przez procesy alertera. Łączna liczba procesów alertera jest ograniczona przez parametr `StartAlerters` w pliku konfiguracji serwera.

Testowanie

Aby przetestować skonfigurowany typ mediów skryptu:

1. Znajdź odpowiedni skrypt na liście typów mediów.
2. Kliknij *Test* w ostatniej kolumnie listy; formularz testowy zostanie otwarty w oknie podręcznym. Formularz testowy będzie zawierał taką samą liczbę parametrów, jaka została skonfigurowana dla typu mediów skryptu.
3. W razie potrzeby edytuj wartości parametrów skryptu. Edycja wpływa wyłącznie na procedurę testową; rzeczywiste wartości nie zostaną zmienione.
4. Kliknij *Test*.



Note:
Podczas testowania skonfigurowanego typu mediów skryptu {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} oraz makra użytkownika zostaną rozwinięte do swoich wartości, ale makra powiązane z zdarzeniami (np. {HOST.HOST}, {ITEM.LASTVALUE} itd.) nie zostaną rozwinięte, ponieważ podczas testowania nie ma powiązanego zdarzenia, z którego można pobrać szczegóły. Należy pamiętać, że makra wewnątrz makr {ALERT.SUBJECT} i {ALERT.MESSAGE} również nie zostaną rozwinięte. Na przykład, jeśli wartość {ALERT.SUBJECT} jest złożona z "Problem: {EVENT.NAME}", to makro {EVENT.NAME} nie zostanie rozwinięte.

Media użytkownika

Po skonfigurowaniu typu mediów przejdź do sekcji *Users* → *Users* i edytuj profil użytkownika, przypisując ten typ mediów do użytkownika. Kroki konfiguracji mediów użytkownika, wspólne dla wszystkich typów mediów, są opisane na stronie Media types.

Należy pamiętać, że podczas definiowania mediów użytkownika pole *Send to* nie może być puste. Jeśli to pole nie jest używane w skrypcie alertu, wprowadź dowolną kombinację obsługiwanych znaków, aby ominąć wymagania walidacji.

4 Webhook

Przegląd

Typ nośnika webhook jest przydatny do wykonywania wywołań HTTP przy użyciu niestandardowego kodu JavaScript w celu łatwej integracji z zewnętrznym oprogramowaniem, takim jak systemy helpdesk, czaty lub komunikatory. Możesz wybrać import integracji dostarczonej przez Zabbix lub utworzyć własną integrację od podstaw.

Integracje

Dostępne są następujące integracje, umożliwiające korzystanie z predefiniowanych typów mediów webhook do przesyłania powiadomień Zabbix do:

- brevis.one
- Discord
- Event-Driven Ansible
- Express.ms messenger
- GitHub
- GLPI
- IBM Maximo Service Request
- iLert
- iTop
- Jira
- Jira Service Management
- ManageEngine ServiceDesk
- Mantis Bug Tracker
- Mattermost
- MS Teams
- MS Teams Workflows
- LINE
- Opsgenie
- OTRS CE
- Pagerduty
- Pushover
- Redmine
- Rocket.Chat
- ServiceNow
- SIGNL4
- Slack
- SolarWinds
- SysAid
- Telegram
- TOPdesk
- VictorOps
- Zammad
- Zendesk

Note:
Oprócz usług wymienionych tutaj, Zabbix można zintegrować z **Spiceworks** (nie jest wymagany webhook). Aby przekształcać powiadomienia Zabbix w zgłoszenia Spiceworks, utwórz typ mediów e-mail i wpisz adres e-mail helpdesku Spiceworks (np. help@zabbix.on.spiceworks.com) w ustawieniach profilu wybranego użytkownika Zabbix.

Konfiguracja

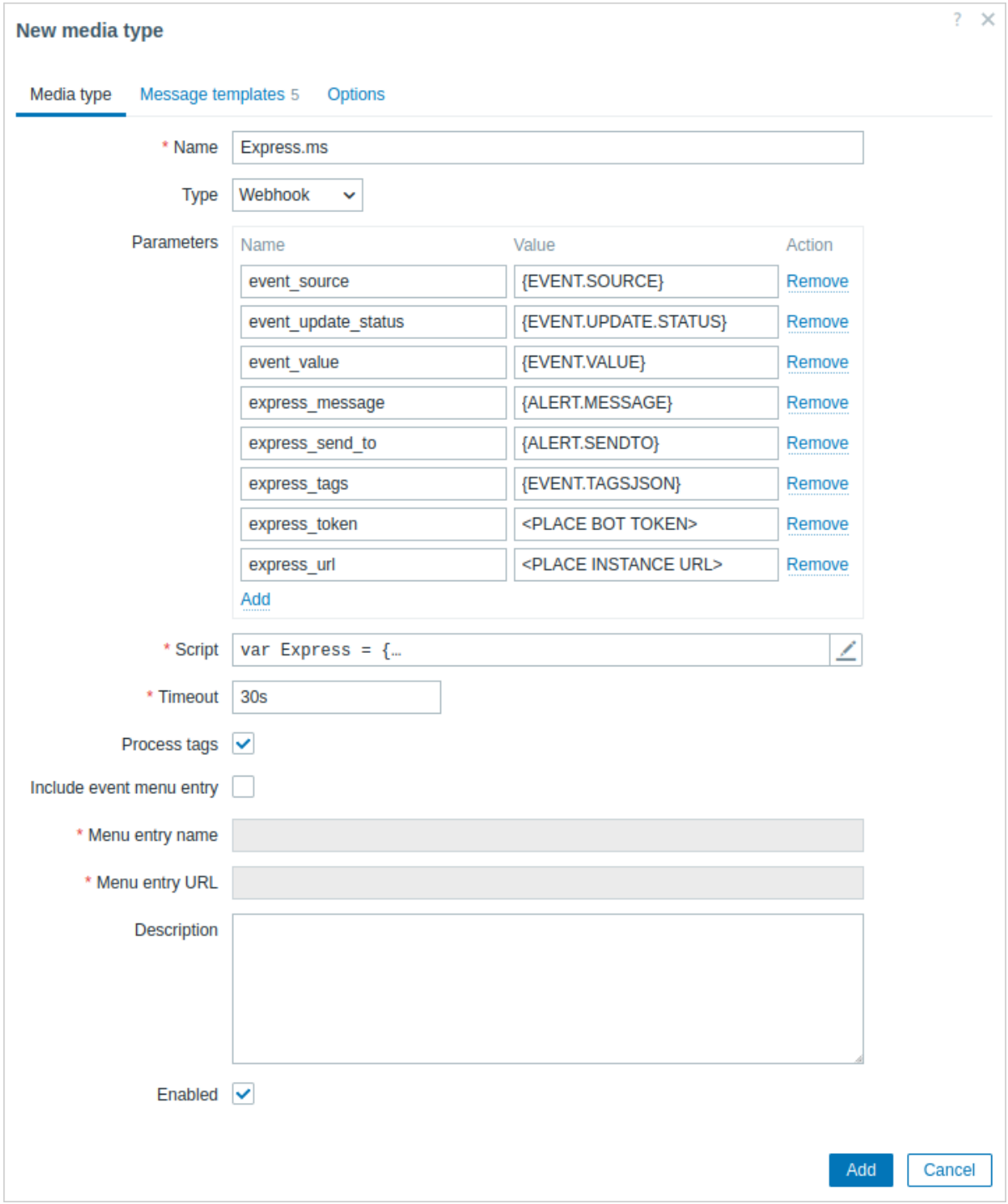
Aby rozpocząć korzystanie z integracji webhook:

1. Znajdź wymagany plik .yaml w katalogu `templates/media` pobranej wersji Zabbixa lub pobierz go z repozytorium Zabbixa git repository.
2. Zaimportuj plik do swojej instalacji Zabbixa. Webhook pojawi się na liście typów mediów.
3. Skonfiguruj webhook zgodnie z instrukcjami w pliku *Readme.md* (możesz kliknąć nazwę webhooka powyżej, aby szybko przejść do *Readme.md*).

Aby utworzyć własny webhook od podstaw:

1. Przejdź do *Alerts* > *Media types*.
2. Kliknij *Create media type*.

Karta **Media type** zawiera różne atrybuty specyficzne dla tego typu mediów:



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Następujące parametry są specyficzne dla typu mediów webhook:

Parameter	Description
Parameters	Określ zmienne webhooka jako pary atrybut-wartość. W przypadku wstępnie skonfigurowanych webhooków lista parametrów różni się w zależności od usługi. Opis parametrów znajduje się w pliku *Readme.md* webhooka. W przypadku nowych webhooków domyślnie uwzględniane są niektóre wspólne zmienne (URL:<empty>, HTTPProxy:<empty>, To:{ALERT.SENDTO}, Subject:{ALERT.SUBJECT}, Message:{ALERT.MESSAGE}); możesz je pozostawić lub usunąć. Parametry webhooka obsługują makra użytkownika, wszystkie makra obsługiwane w powiadomieniach o problemach oraz dodatkowo makra {ALERT.SENDTO}, {ALERT.SUBJECT} i {ALERT.MESSAGE}. Jeśli określisz serwer proxy HTTP, pole obsługuje tę samą funkcjonalność co pole HTTP proxy w konfiguracji pozycji. Ciąg proxy może być poprzedzony `[scheme]://`, aby określić, jaki rodzaj proxy jest używany (np. https, socks4, socks5; zobacz dokumentację).
Script	Wprowadź kod JavaScript w edytorze modalnym, który otwiera się po kliknięciu pola parametru lub ikony ołówka obok niego. Ten kod wykona operację webhooka. Skrypt jest kodem funkcji, który przyjmuje pary parametr - wartość. Wartości należy konwertować na obiekty JSON za pomocą metody JSON.parse(), na przykład: `var params = JSON.parse(value);`. Kod ma dostęp do wszystkich parametrów, może wykonywać żądania HTTP GET, POST, PUT i DELETE, obsługuje dodatkowe metody, takie jak CONNECT, PATCH, HEAD, OPTIONS i TRACE, oraz ma kontrolę nad nagłówkami HTTP i treścią żądania. Skrypt musi zawierać operator return, w przeciwnym razie nie będzie poprawny. Może zwracać status OK wraz z opcjonalną listą tagów i wartości tagów (zobacz opcję *Process tags*) albo ciąg błędu. Pamiętaj, że skrypt jest wykonywany dopiero po utworzeniu alertu. Jeśli skrypt jest skonfigurowany tak, aby zwracał i przetwarzał tagi, tagi te nie zostaną rozwiązane w makrach {EVENT.TAGS} i {EVENT.RECOVERY.TAGS} w początkowej wiadomości o problemie oraz w wiadomościach odzyskiwania, ponieważ skrypt nie zdążył jeszcze zostać uruchomiony. *Uwaga*: Zaleca się używanie zmiennych lokalnych (np. `var local = 1`) zamiast globalnych (np. `global = 1`), aby zapewnić, że każdy skrypt działa na własnych danych i uniknąć kolizji między równoczesnymi wywołaniami (zobacz znane problemy). Zobacz także: Webhook development guidelines, Webhook script examples, Additional JavaScript objects.
Timeout	Limit czasu wykonania JavaScript (1-60 s, domyślnie 30 s). Obsługiwane są sufiksy czasu, np. 30s, 1m.
Process tags	Zaznacz pole wyboru, aby przetwarzać zwrócone wartości właściwości JSON jako tagi. Tagi te są dodawane do wszystkich istniejących tagów problemu. Pamiętaj, że podczas używania webhook tags webhook musi zwrócić obiekt JSON zawierający co najmniej pusty obiekt tags: `var result = {tags: {}};` Przykłady tagów, które mogą zostać zwrócone: *jira-id:prod-1234*, *responsible:John Smith*, *processed:<no value>*
Include event menu entry	Zaznacz pole wyboru, aby dodać wpis w menu zdarzenia prowadzący do utworzonego zewnętrznego zgłoszenia. Wpis zostanie dodany dla każdego włączonego webhooka, dla którego zaznaczono to pole. Pamiętaj, że jeśli parametry *Menu entry name* i *Menu entry URL* zawierają jakiekolwiek makra {EVENT.TAGS.<tag name>}, wpis zostanie dodany tylko wtedy, gdy makra te mogą zostać rozwiązane (to znaczy, gdy zdarzenie ma zdefiniowane te tagi). Jeśli pole jest zaznaczone, webhook nie powinien być używany do wysyłania powiadomień do różnych użytkowników (rozważ utworzenie dedykowanego użytkownika) i nie powinien być używany w wielu akcjach alertów dla pojedynczego zdarzenia problemu.
Menu entry name	Określ nazwę wpisu menu. Obsługiwane jest makro {EVENT.TAGS.<tag name>}. To pole jest obowiązkowe tylko wtedy, gdy zaznaczono *Include event menu entry*.
Menu entry URL	Określ docelowy adres URL wpisu menu. Obsługiwane jest makro {EVENT.TAGS.<tag name>}. To pole jest obowiązkowe tylko wtedy, gdy zaznaczono *Include event menu entry*.

Szczegóły dotyczące konfiguracji domyślnych wiadomości i opcji przetwarzania alertów znajdziesz w sekcji common media type parameters.

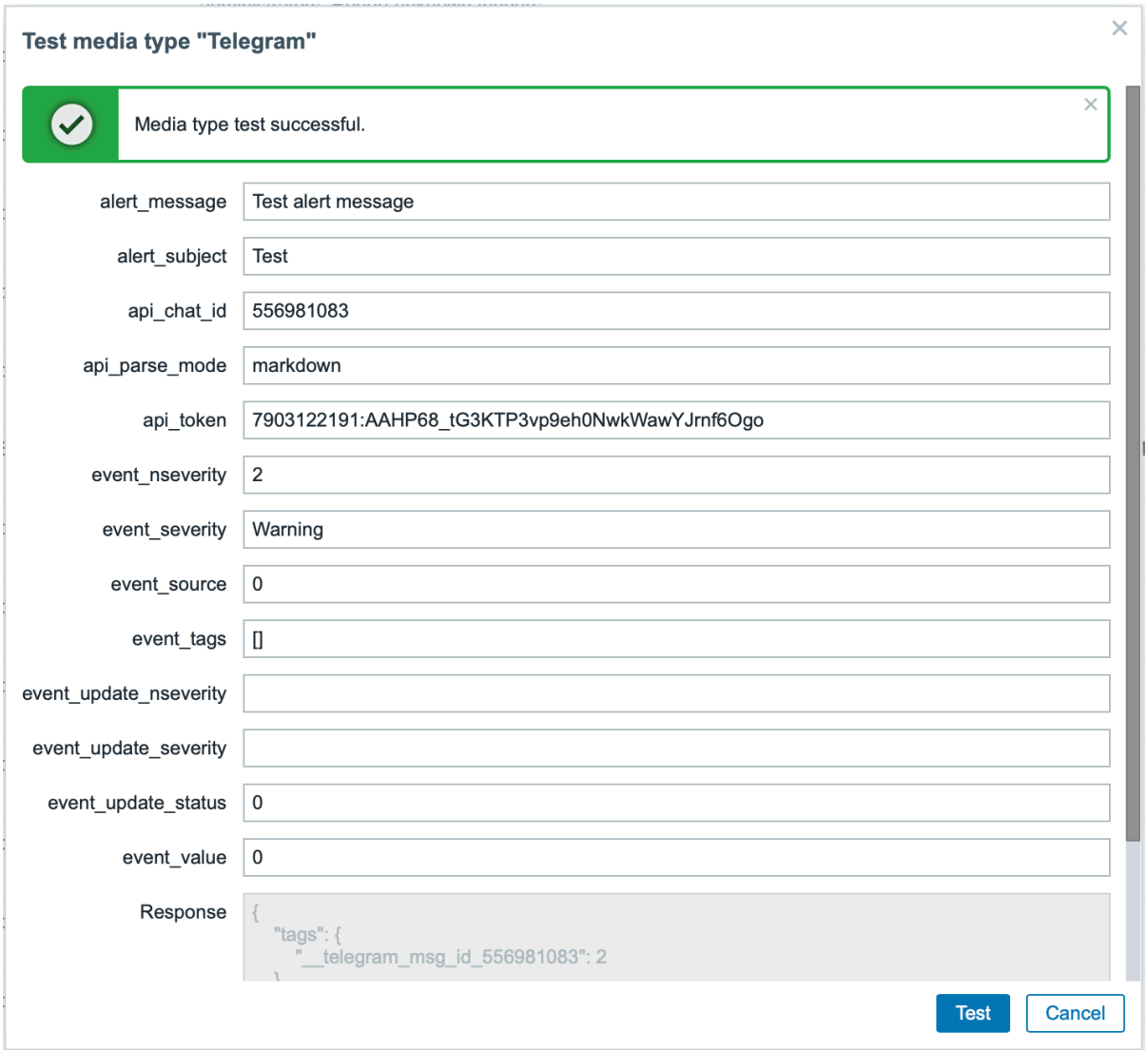
Warning:
Nawet jeśli webhook nie używa domyślnych wiadomości, szablony wiadomości dla typów operacji używanych przez ten webhook nadal muszą być zdefiniowane.

Testowanie

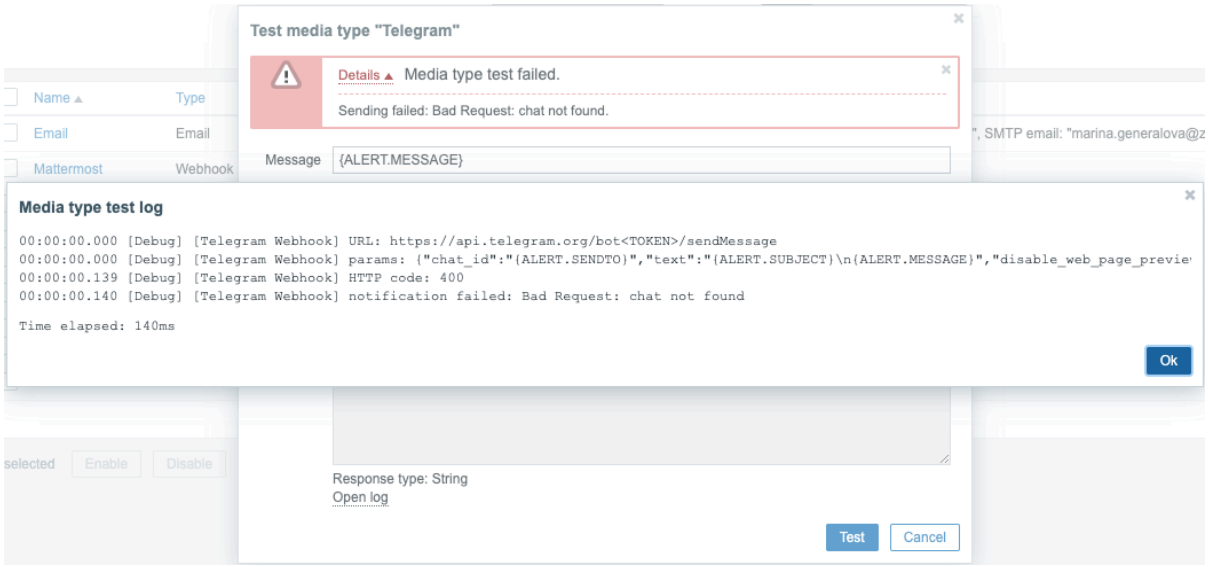
Aby przetestować skonfigurowany typ nośnika webhook:

1. Znajdź odpowiedni webhook na liście typów nośników.
2. Kliknij *Test* w ostatniej kolumnie listy (otworzy się okno testowania).
3. W razie potrzeby edytuj wartości parametrów webhooka. Zastąp makra przykładowymi wartościami; w przeciwnym razie makra nie zostaną rozwinięte, a test zakończy się niepowodzeniem.
4. Kliknij *Test*.

Zastąpienie lub usunięcie wartości w oknie testowania wpływa tylko na procedurę testową; rzeczywiste wartości atrybutów webhooka pozostaną bez zmian.



Aby wyświetlić wpisy dziennika testu typu nośnika bez opuszczania okna testowego, kliknij *Open log* (otworzy się nowe wyskakujące okno).



Jeśli test webhooka zakończy się powodzeniem:

- Wyświetlany jest komunikat *„Media type test successful.”*.
- Odpowiedź serwera pojawia się w szarym polu *Response*.
- Typ odpowiedzi (JSON lub String) jest określony poniżej pola *Response*.

Jeśli test webhooka zakończy się niepowodzeniem:

- Wyświetlany jest komunikat *„Test typu nośnika nie powiódł się.”*, po którym podawane są dodatkowe szczegóły dotyczące błędu.

Media użytkownika

Po skonfigurowaniu typu mediów przejdź do sekcji *Users* > *Users* i przypisz media webhook do istniejącego użytkownika lub utwórz nowego użytkownika reprezentującego webhook. Kroki konfiguracji mediów użytkownika dla istniejącego użytkownika, wspólne dla wszystkich typów mediów, opisano na stronie Typy mediów.

Jeśli webhook używa tagów do przechowywania ID zgłoszenia\wiadomości, unikaj przypisywania tego samego webhooka jako mediów różnym użytkownikom, ponieważ może to powodować błędy webhooka (dotyczy to większości webhooków korzystających z opcji *Include event menu entry*). W takim przypadku zalecaną praktyką jest utworzenie dedykowanego użytkownika reprezentującego webhook:

1. Po skonfigurowaniu typu mediów webhook przejdź do sekcji *Users* > *Users* i utwórz dedykowanego użytkownika Zabbix reprezentującego webhook — na przykład o nazwie użytkownika *Slack* dla webhooka Slack. Wszystkie ustawienia, z wyjątkiem mediów, można pozostawić domyślne, ponieważ ten użytkownik nie będzie logował się do Zabbix.
2. W profilu użytkownika przejdź do zakładki *Media* i dodaj webhook, podając wymagane informacje kontaktowe. Jeśli webhook nie używa pola *Send to*, wprowadź dowolną kombinację obsługiwanych znaków, aby obejść wymagania walidacji.
3. Przyznaj temu użytkownikowi co najmniej uprawnienia do odczytu permissions do wszystkich hostów, dla których powinien wysyłać alerty.

Podczas konfigurowania akcji alertów dodaj tego użytkownika w polu *Send to users* w szczegółach operacji — spowoduje to, że Zabbix będzie używać webhooka do powiadomień z tej akcji.

Konfigurowanie akcji alertów

Akcje określają, które powiadomienia mają być wysyłane za pośrednictwem webhooka. Kroki konfigurowania akcji z użyciem webhooków są takie same jak dla wszystkich innych typów mediów, z następującymi wyjątkami:

- Jeśli webhook używa tagów webhooka do przechowywania ID zgłoszenia\wiadomości oraz obsługi operacji aktualizacji\rozwiązania, należy unikać używania tego samego webhooka w wielu akcjach alertów dla pojedynczego zdarzenia problemu. Jeśli {EVENT.TAGS.<tag name>} istnieje i zostanie zaktualizowane w webhooku, jego wynikowa wartość będzie niezdefiniowana. Aby tego uniknąć, użyj w webhooku nowej nazwy tagu do przechowywania zaktualizowanych wartości. Dotyczy to webhooków Jira, Jira Service Desk, Mattermost, Opsgenie, OTRS, Redmine, ServiceNow, Slack, Zammad i Zendesk dostarczanych przez Zabbix oraz większości webhooków wykorzystujących opcję *Include event menu entry*. Należy jednak pamiętać, że pojedynczy webhook może być używany w wielu operacjach lub krokach eskalacji tej samej akcji, a także w różnych akcjach, które nie zostaną wyzwolone przez to samo zdarzenie problemu z powodu różnych warunków.
- W przypadku używania webhooka w akcjach dla zdarzeń wewnętrznych należy zaznaczyć pole wyboru *Custom message* i zdefiniować własną wiadomość w konfiguracji operacji akcji. W przeciwnym razie powiadomienie nie zostanie wysłane.

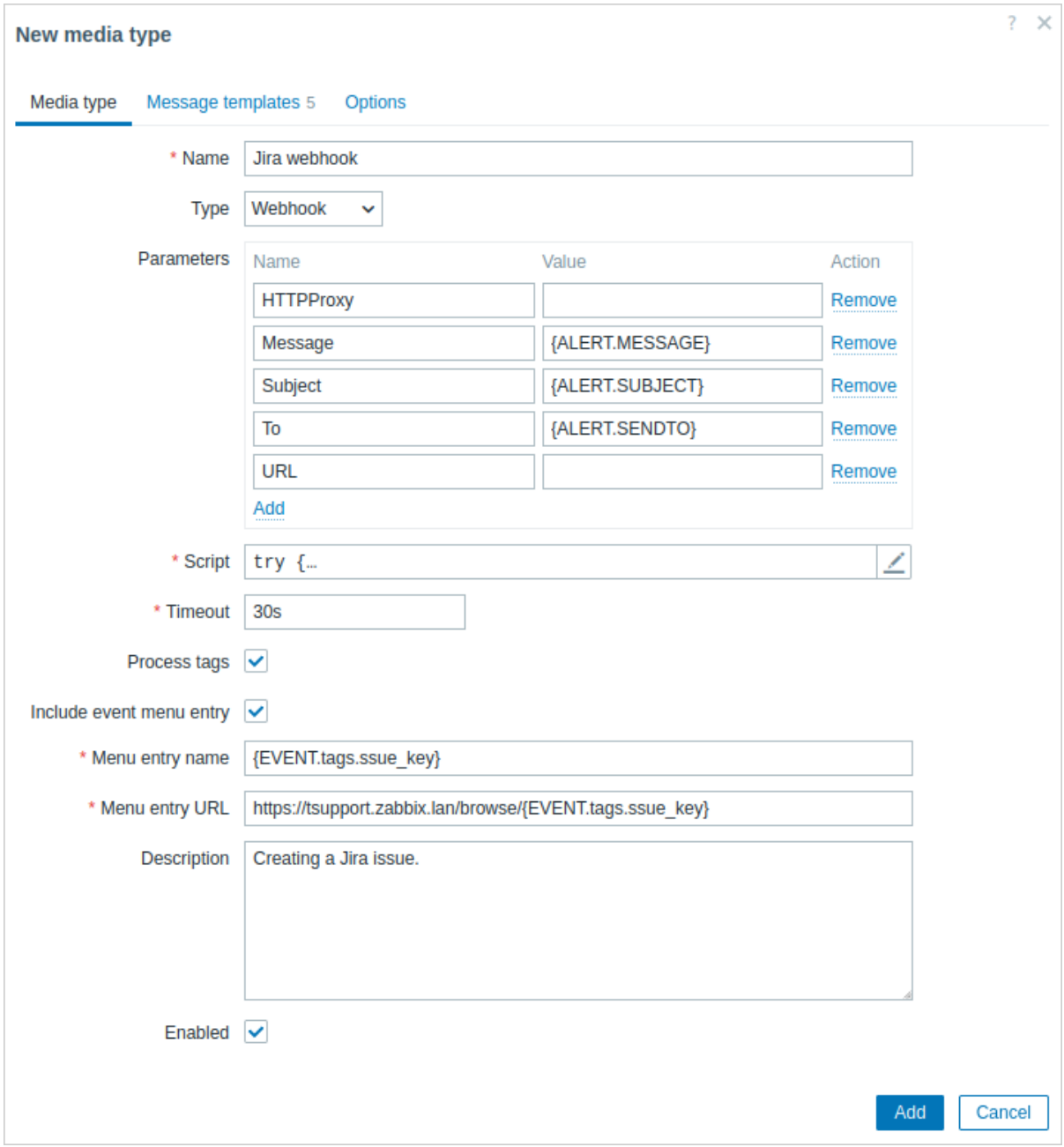
1 Przykłady skryptów webhooka

Przegląd

Chociaż Zabbix oferuje dużą liczbę integracji webhook dostępnych od razu po instalacji, możesz zamiast tego chcieć utworzyć własne webhooki. Ta sekcja zawiera przykłady niestandardowych skryptów webhook (używanych w parametrze *Script*). Opis innych parametrów webhook znajduje się w sekcji webhook.

Attention:
Nie używaj niezadeklarowanych przypisań w JavaScript przetwarzania wstępnego. Do deklarowania zmiennych lokalnych używaj `var`.

Jira webhook (niestandardowy)



Ten skrypt utworzy zgłoszenie JIRA i zwróci pewne informacje o utworzonym zgłoszeniu.

```
try {
    Zabbix.log(4, '[ Jira webhook ] Started with 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://jira.example.com/rest/api/2/issue/',
        JSON.stringify({"fields": fields})
    );

    if (req.getStatus() != 201) {
        throw 'Response code: ' + req.getStatus();
    }

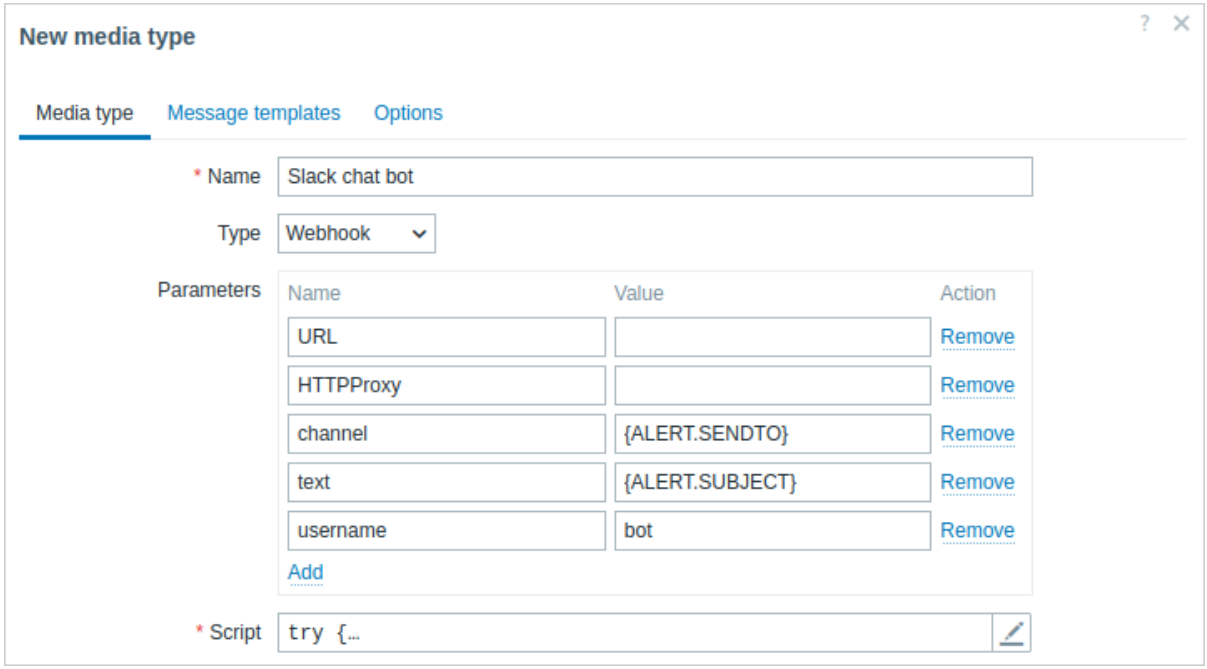
    resp = JSON.parse(resp);
    result.tags.issue_id = resp.id;
    result.tags.issue_key = resp.key;

    return JSON.stringify(result);
}
catch (error) {
    Zabbix.log(4, '[ Jira webhook ] Issue creation failed json : ' + JSON.stringify({"fields": fields}));
    Zabbix.log(3, '[ Jira webhook ] issue creation failed : ' + error);

    throw 'Failed with error: ' + error;
}
```

Slack webhook (niestandardowy)

Ten webhook będzie przekazywać powiadomienia z Zabbix do kanału 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 ] Webhook request with value=' + value);

    response = req.post(params.hook_url, 'payload=' + encodeURIComponent(value));
    Zabbix.log(4, '[ Slack webhook ] Responded with code: ' + req.getStatus() + '. Response: ' + response)

    try {
        response = JSON.parse(response);
    }
    catch (error) {
        if (req.getStatus() < 200 || req.getStatus() >= 300) {
            throw 'Request failed with status code ' + req.getStatus();
        }
        else {
            throw 'Request success, but response parsing failed.';
        }
    }

    if (req.getStatus() !== 200 || !response.ok || response.ok === 'false') {
        throw response.error;
    }

    return 'OK';
}
catch (error) {
    Zabbix.log(3, '[ Slack webhook ] Sending failed. Error: ' + error);

    throw 'Failed with error: ' + error;
}
```

2 Akcje

Przegląd

Jeśli chcesz, aby w wyniku zdarzeń wykonywały się pewne operacje (na przykład wysyłane były powiadomienia), musisz skonfigurować akcje.

Akcje można definiować w odpowiedzi na zdarzenia wszystkich obsługiwanych typów:

- Akcje wyzwalaczy - dla zdarzeń, gdy status wyzwalacza zmienia się z *OK* na *PROBLEM* i z powrotem
- Akcje usług - dla zdarzeń, gdy status usługi zmienia się z *OK* na *PROBLEM* i z powrotem
- Akcje wykrywania - dla zdarzeń, gdy odbywa się wykrywanie sieci
- Akcje autorejestracji - dla zdarzeń, gdy nowe aktywne agenty automatycznie się rejestrują (lub zmieniają się metadane hosta dla już zarejestrowanych)
- Akcje wewnętrzne - dla zdarzeń, gdy pozycje stają się nieobsługiwane lub wyzwalacze przechodzą w stan nieznany

Główne różnice dotyczące akcji usług to:

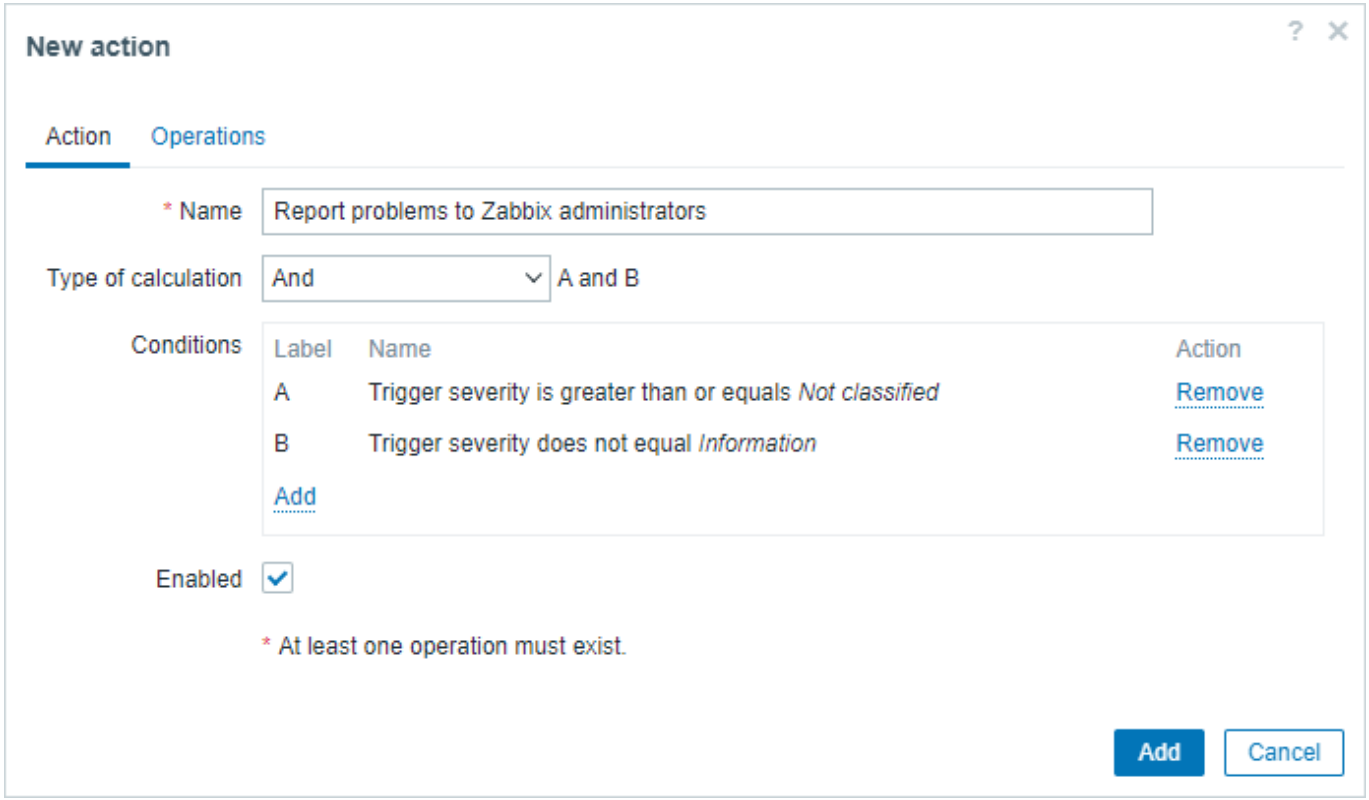
- Dostęp użytkownika do akcji usług zależy od uprawnień do usług przyznanych przez rolę użytkownika
- Akcje usług obsługują inny zestaw warunków

Konfigurowanie akcji

Aby skonfigurować akcję, wykonaj następujące czynności:

- Przejdź do *Alerty* > *Akcje* i wybierz wymagany typ akcji z podmenu (później możesz przełączyć się na inny typ, używając rozwijanej listy w tytule).
- Kliknij *Utwórz akcję*.
- Nadaj akcji nazwę.
- Wybierz warunki, po spełnieniu których wykonywane są operacje.
- Wybierz operacje, które mają zostać wykonane.

Ogólne atrybuty akcji:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Unikalna nazwa akcji.

Parametr	Opis
Typ obliczania	Wybierz opcję oceny dla warunków akcji (gdy jest więcej niż jeden warunek): **I** - wszystkie warunki muszą być spełnione. **Lub** - wystarczy, że jeden warunek zostanie spełniony. **I/Lub** - połączenie obu: AND dla różnych typów warunków oraz OR dla tego samego typu warunku. **Wyrażenie niestandardowe** - zdefiniowana przez użytkownika formuła obliczeniowa do oceny warunków akcji.
Warunki	Lista warunków akcji. Kliknij *Dodaj*, aby dodać nowy warunek. Jeśli nie skonfigurowano żadnych warunków, akcja zostanie uruchomiona dla każdego zdarzenia odpowiadającego konfigurowanemu typowi akcji.
Włączone	Zaznacz pole wyboru, aby włączyć akcję. W przeciwnym razie będzie ona wyłączona.

1 Warunki

Przegląd

Można zdefiniować, że akcja zostanie wykonana tylko wtedy, gdy zdarzenie spełnia określony zestaw warunków. Warunki ustawia się podczas konfigurowania action.

Dopasowanie warunków uwzględnia wielkość liter.

Akcje wyzwalaczy

W akcjach opartych na wyzwalaczach można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Grupa hostów	równa się nie równa się	Określ grupy hostów lub grupy hostów do wykluczenia. **równa się** - zdarzenie należy do tej grupy hostów. **nie równa się** - zdarzenie nie należy do tej grupy hostów. Określenie nadrzędnej grupy hostów niejawnie wybiera wszystkie zagnieżdżone grupy hostów. Aby wskazać tylko grupę nadrzędną, wszystkie zagnieżdżone grupy muszą zostać dodatkowo ustawione z operatorem **nie równa się**.
Szablon	równa się nie równa się	Określ szablony lub szablony do wykluczenia. **równa się** - zdarzenie należy do wyzwalacza dziedziczonego z tego szablonu. **nie równa się** - zdarzenie nie należy do wyzwalacza dziedziczonego z tego szablonu.
Host	równa się nie równa się	Określ hosty lub hosty do wykluczenia. **równa się** - zdarzenie należy do tego hosta. **nie równa się** - zdarzenie nie należy do tego hosta.
Nazwa tagu	równa się nie równa się zawiera nie zawiera	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. **równa się** - zdarzenie ma ten tag. **nie równa się** - zdarzenie nie ma tego tagu. **zawiera** - zdarzenie ma tag zawierający ten ciąg znaków. **nie zawiera** - zdarzenie nie ma tagu zawierającego ten ciąg znaków.
Wartość tagu	równa się nie równa się zawiera nie zawiera	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. **równa się** - zdarzenie ma ten tag i wartość. **nie równa się** - zdarzenie nie ma tego tagu i wartości. **zawiera** - zdarzenie ma tag i wartość zawierające te ciągi znaków. **nie zawiera** - zdarzenie nie ma tagu i wartości zawierających te ciągi znaków.
Wyzwalacz	równa się nie równa się	Określ wyzwalacze lub wyzwalacze do wykluczenia. **równa się** - zdarzenie jest generowane przez ten wyzwalacz. **nie równa się** - zdarzenie jest generowane przez dowolny inny wyzwalacz, z wyjątkiem tego.

Typ warunku	Obsługiwane operatory	Opis
Nazwa zdarzenia	zawiera nie zawiera	Określ ciąg znaków w nazwie zdarzenia wygenerowanego przez wyzwalacz lub ciąg znaków do wykluczenia. Domyślnie nazwa zdarzenia odpowiada nazwie wyzwalacza, chyba że w konfiguracji wyzwalacza określono niestandardową nazwę zdarzenia. **zawiera** - nazwa zdarzenia zawiera ten ciąg znaków. **nie zawiera** - ten ciąg znaków jest wykluczony z nazwy zdarzenia. Uwaga: Wprowadzona wartość będzie porównywana z nazwą zdarzenia po rozwinięciu wszystkich makr.
Poziom ważności wyzwalacza	równa się nie równa się jest większy lub równy jest mniejszy lub równy	Określ poziom ważności wyzwalacza. **równa się** - równy poziomowi ważności wyzwalacza. **nie równa się** - nie równy poziomowi ważności wyzwalacza. **jest większy lub równy** - większy lub równy poziomowi ważności wyzwalacza. **jest mniejszy lub równy** - mniejszy lub równy poziomowi ważności wyzwalacza.
Okres czasu	w nie w	Określ okres czasu lub okres czasu do wykluczenia. **w** - czas zdarzenia mieści się w okresie czasu. **nie w** - czas zdarzenia nie mieści się w okresie czasu. Opis formatu znajduje się na stronie specyfikacji okresu czasu. Obsługiwane są makra użytkownika.
Problem jest wyciszony	nie tak	Określ, czy problem jest wyciszony (niewyświetlany) z powodu konserwacji hosta. **nie** - problem nie jest wyciszony. **tak** - problem jest wyciszony.

Akcje usług

W akcjach usług można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Usługa	równa się nie równa się	Określ usługę lub usługę do wykluczenia. **równa się** - zdarzenie należy do tej usługi. **nie równa się** - zdarzenie nie należy do tej usługi. Określenie usługi nadrzędnej niejawnie wybiera wszystkie usługi podrzędne. Aby wskazać tylko usługę nadrzędną, wszystkie zagnieżdżone usługi muszą zostać dodatkowo ustawione z operatorem **nie równa się**.
Nazwa usługi	zawiera nie zawiera	Określ ciąg znaków w nazwie usługi lub ciąg znaków do wykluczenia. **zawiera** - zdarzenie jest generowane przez usługę, której nazwa zawiera ten ciąg znaków. **nie zawiera** - tego ciągu znaków nie można znaleźć w nazwie usługi.
Nazwa tagu usługi	równa się nie równa się zawiera nie zawiera	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. Tagi zdarzeń usługi można zdefiniować w sekcji konfiguracji usługi *Tagi*. **równa się** - zdarzenie ma ten tag. **nie równa się** - zdarzenie nie ma tego tagu. **zawiera** - zdarzenie ma tag zawierający ten ciąg znaków. **nie zawiera** - zdarzenie nie ma tagu zawierającego ten ciąg znaków.
Wartość tagu usługi	równa się nie równa się zawiera nie zawiera	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. Tagi zdarzeń usługi można zdefiniować w sekcji konfiguracji usługi *Tagi*. **równa się** - zdarzenie ma ten tag i wartość. **nie równa się** - zdarzenie nie ma tego tagu i wartości. **zawiera** - zdarzenie ma tag i wartość zawierające te ciągi znaków. **nie zawiera** - zdarzenie nie ma tagu i wartości zawierających te ciągi znaków.

Attention:
Upewnij się, że w menu *Alerty* → *Typy mediów* zdefiniowano szablony wiadomości dla akcji usług. W przeciwnym razie powiadomienia nie będą wysyłane.

Akcje wykrywania

W zdarzeniach opartych na wykrywaniu można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
IP hosta	równa się nie równa się	Określ zakres adresów IP lub zakres do wykluczenia dla wykrytego hosta. **równa się** - IP hosta znajduje się w zakresie. **nie równa się** - IP hosta nie znajduje się w zakresie. Może mieć następujące formaty: Pojedynczy adres IP: 192.168.1.33 Zakres adresów IP: 192.168.1-10.1-254 Maska IP: 192.168.4.0/24 Lista: 192.168.1.1-254, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 Spacje w formacie listy są obsługiwane.
Typ usługi	równa się nie równa się	Określ typ usługi wykrytej usługi lub typ usługi do wykluczenia. **równa się** - pasuje do wykrytej usługi. **nie równa się** - nie pasuje do wykrytej usługi. Dostępne typy usług: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping, telnet.
Port usługi	równa się nie równa się	Określ zakres portów TCP wykrytej usługi lub zakres do wykluczenia. **równa się** - port usługi znajduje się w zakresie. **nie równa się** - port usługi nie znajduje się w zakresie.
Reguła wykrywania	równa się nie równa się	Określ regułę wykrywania lub regułę wykrywania do wykluczenia. **równa się** - używa tej reguły wykrywania. **nie równa się** - używa dowolnej innej reguły wykrywania, z wyjątkiem tej.
Sprawdzenie wykrywania	równa się nie równa się	Określ sprawdzenie wykrywania lub sprawdzenie wykrywania do wykluczenia. **równa się** - używa tego sprawdzenia wykrywania. **nie równa się** - używa dowolnego innego sprawdzenia wykrywania, z wyjątkiem tego.
Obiekt wykrywania	równa się	Określ wykryty obiekt. **równa się** - równe wykrytemu obiektowi (urządzenie lub usługa).
Status wykrywania	równa się	**Up** - pasuje do zdarzeń „Host Up” i „Service Up”. **Down** - pasuje do zdarzeń „Host Down” i „Service Down”. **Discovered** - pasuje do zdarzeń „Host Discovered” i „Service Discovered”. **Lost** - pasuje do zdarzeń „Host Lost” i „Service Lost”.
Czas działania/przestoju	jest większy lub równy jest mniejszy lub równy	Czas działania dla zdarzeń „Host Up” i „Service Up”. Czas przestoju dla zdarzeń „Host Down” i „Service Down”. **jest większy lub równy** - jest większy lub równy. Parametr jest podawany w sekundach. **jest mniejszy lub równy** - jest mniejszy lub równy. Parametr jest podawany w sekundach.

Typ warunku	Obsługiwane operatory	Opis
Otrzymana wartość	równa się nie równa się jest większy lub równy jest mniejszy lub równy zawiera nie zawiera	Określ wartość otrzymaną ze sprawdzenia agent (Zabbix, SNMP) w regule wykrywania. Porównanie ciągów znaków. Jeśli dla reguły skonfigurowano kilka sprawdzeń Zabbix agent lub SNMP, sprawdzane są otrzymane wartości dla każdego z nich (każde sprawdzenie generuje nowe zdarzenie, które jest dopasowywane do wszystkich warunków). **równa się** - równe wartości. **nie równa się** - nie równe wartości. **jest większy lub równy** - większe lub równe wartości. **jest mniejszy lub równy** - mniejsze lub równe wartości. **zawiera** - zawiera podciąg. Parametr jest podawany jako ciąg znaków. **nie zawiera** - nie zawiera podciągu. Parametr jest podawany jako ciąg znaków.
Proxy	równa się nie równa się	Określ proxy lub proxy do wykluczenia. **równa się** - używa tego proxy. **nie równa się** - używa dowolnego innego proxy z wyjątkiem tego.

Note:
Sprawdzenia usług w regule wykrywania, które skutkują zdarzeniami wykrywania, nie są wykonywane jednocześnie. Dlatego jeśli w akcji skonfigurowano **wiele** wartości dla warunków `Service type`, `Service port` lub `Received value`, będą one porównywane z jednym zdarzeniem wykrywania naraz, ale **nie** z kilkoma zdarzeniami jednocześnie. W rezultacie akcje z wieloma wartościami dla tych samych typów sprawdzeń mogą nie być wykonywane poprawnie.

Akcje autorejestracji

W akcjach opartych na aktywnej autorejestracji agent można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Metadane hosta	zawiera nie zawiera pasuje nie pasuje	Określ metadane hosta lub metadane hosta do wykluczenia. **zawiera** - metadane hosta zawierają ciąg znaków. **nie zawiera** - metadane hosta nie zawierają ciągu znaków. Metadane hosta można określić w pliku konfiguracyjnym agent. **pasuje** - metadane hosta pasują do wyrażenia regularnego. **nie pasuje** - metadane hosta nie pasują do wyrażenia regularnego.
Nazwa hosta	zawiera nie zawiera pasuje nie pasuje	Określ nazwę hosta lub nazwę hosta do wykluczenia. **zawiera** - nazwa hosta zawiera ciąg znaków. **nie zawiera** - nazwa hosta nie zawiera ciągu znaków. **pasuje** - nazwa hosta pasuje do wyrażenia regularnego. **nie pasuje** - nazwa hosta nie pasuje do wyrażenia regularnego.
Proxy	równa się nie równa się	Określ proxy lub proxy do wykluczenia. **równa się** - używane jest to proxy. **nie równa się** - używane jest dowolne inne proxy poza tym.

Akcje zdarzeń wewnętrznych

Dla akcji opartych na zdarzeniach wewnętrznych można ustawić następujące warunki:

Typ warunku	Obsługiwane operatory	Opis
Typ zdarzenia	equals	**Pozycja w stanie „nieobsługiwane”** - pasuje do zdarzeń, w których pozycja przechodzi ze stanu „normalny” do stanu „nieobsługiwane”. **Reguła wykrywania niskiego poziomu w stanie „nieobsługiwane”** - pasuje do zdarzeń, w których reguła wykrywania niskiego poziomu przechodzi ze stanu „normalny” do stanu „nieobsługiwane”. **Wyzwalacz w stanie „nieznany”** - pasuje do zdarzeń, w których wyzwalacz przechodzi ze stanu „normalny” do stanu „nieznany”.
Grupa hostów	equals does not equal	Określ grupy hostów lub grupy hostów do wykluczenia. **equals** - zdarzenie należy do tej grupy hostów. **does not equal** - zdarzenie nie należy do tej grupy hostów.

Typ warunku	Obsługiwane operatory	Opis
Nazwa tagu	equals does not equal contains does not contain	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. **equals** - zdarzenie ma ten tag. **does not equal** - zdarzenie nie ma tego tagu. **contains** - zdarzenie ma tag zawierający ten ciąg znaków. **does not contain** - zdarzenie nie ma tagu zawierającego tego ciągu znaków.
Wartość tagu	equals does not equal contains does not contain	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. **equals** - zdarzenie ma ten tag i wartość. **does not equal** - zdarzenie nie ma tego tagu i wartości. **contains** - zdarzenie ma tag i wartość zawierające te ciągi znaków. **does not contain** - zdarzenie nie ma tagu i wartości zawierających tych ciągów znaków.
Szablon	equals does not equal	Określ szablony lub szablony do wykluczenia. **equals** - zdarzenie należy do pozycji/wyzwalacza/reguły wykrywania niskiego poziomu dziedziczonych z tego szablonu. **does not equal** - zdarzenie nie należy do pozycji/wyzwalacza/reguły wykrywania niskiego poziomu dziedziczonych z tego szablonu.
Host	equals does not equal	Określ hosty lub hosty do wykluczenia. **equals** - zdarzenie należy do tego hosta. **does not equal** - zdarzenie nie należy do tego hosta.

Typ obliczania

Dostępne są następujące opcje obliczania warunków:

- **And** - wszystkie warunki muszą być spełnione

Zwróć uwagę, że użycie obliczania „And" jest niedozwolone między kilkoma wyzwalaczami, gdy są one wybrane jako warunek `Trigger=`. Akcje mogą być wykonywane wyłącznie na podstawie zdarzenia jednego wyzwalacza.

- **Or** - wystarczy, że jeden warunek jest spełniony
- **And/Or** - połączenie obu: AND dla różnych typów warunków oraz OR dla tego samego typu warunku, na przykład:

Grupa hostów równa się Oracle servers
 Grupa hostów równa się MySQL servers
 Nazwa zdarzenia zawiera 'Database is down'
 Nazwa zdarzenia zawiera 'Database is unavailable'

jest oceniane jako

(Grupa hostów równa się Oracle servers **or** Grupa hostów równa się MySQL servers**) and (**Nazwa zdarzenia zawiera 'Database is down' **or** Nazwa zdarzenia zawiera 'Database is unavailable'**)**

- **Custom expression** - formuła obliczania zdefiniowana przez użytkownika do oceny warunków akcji. Musi zawierać wszystkie warunki (reprezentowane jako wielkie litery A, B, C, ...) i może zawierać spacje, tabulatory, nawiasy ( ), **and** (z rozróżnianiem wielkości liter), **or** (z rozróżnianiem wielkości liter), **not** (z rozróżnianiem wielkości liter).

Podczas gdy poprzedni przykład z `And/Or` byłby przedstawiony jako (A or B) and (C or D), w wyrażeniu niestandardowym możesz mieć również wiele innych sposobów obliczania:

(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
 itd.

Usuwanie obiektów używanych w akcjach

Podczas próby usunięcia obiektów, które są używane w warunku/operacji akcji, użytkownik otrzyma odpowiedni komunikat o błędzie.

Akcje nie są wyłączane, jeśli warunki lub operacje akcji:

- mają usunięte obiekty;
- mają obiekty, które nie są już wykrywane przez wykrywanie niskopoziomowe i zostały usunięte (po upływie okresu *Delete lost resources*).

Warunki/operacje akcji nie są usuwane, jeśli mają brakujące obiekty. Zamiast tego usunięte obiekty są wyświetlane jako *Deleted host group*, *Deleted host*, *Deleted trigger* itd. Ikona ostrzeżenia z komunikatem „This action has conditions or operations referencing deleted object(s)." jest wyświetlana w kolumnie „Info" na liście akcji.

2 Operacje

Przegląd

Dla wszystkich zdarzeń można zdefiniować następujące operacje:

- Wyślij wiadomość
- Wykonaj zdalne polecenie

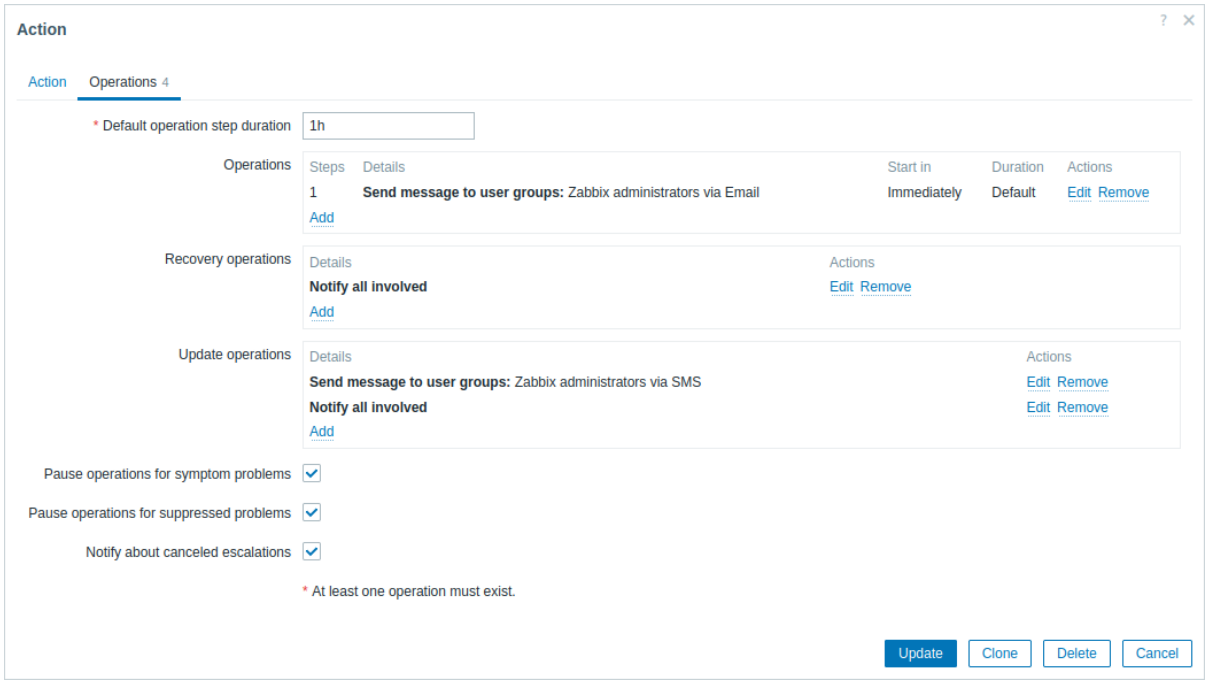
Attention:
Serwer Zabbix nie tworzy alertów, jeśli dostęp do hosta jest jawnie „zabroniony” dla użytkownika zdefiniowanego jako odbiorca operacji akcji lub jeśli użytkownik w ogóle nie ma zdefiniowanych praw do hosta.

Dla zdarzeń wykrywania i autorejestracji dostępne są dodatkowe operacje:

- Dodaj host
- Usuń host
- Włącz host
- Wyłącz host
- Dodaj do grupy hostów
- Usuń z grupy hostów
- Dodaj tagi hosta
- Usuń tagi hosta
- Podłącz szablon
- Odłącz szablon
- Ustaw tryb inwentaryzacji hosta

Konfigurowanie operacji

Aby skonfigurować operację, przejdź do zakładki *Operacje* w konfiguracji action.



Ogólne atrybuty operacji:

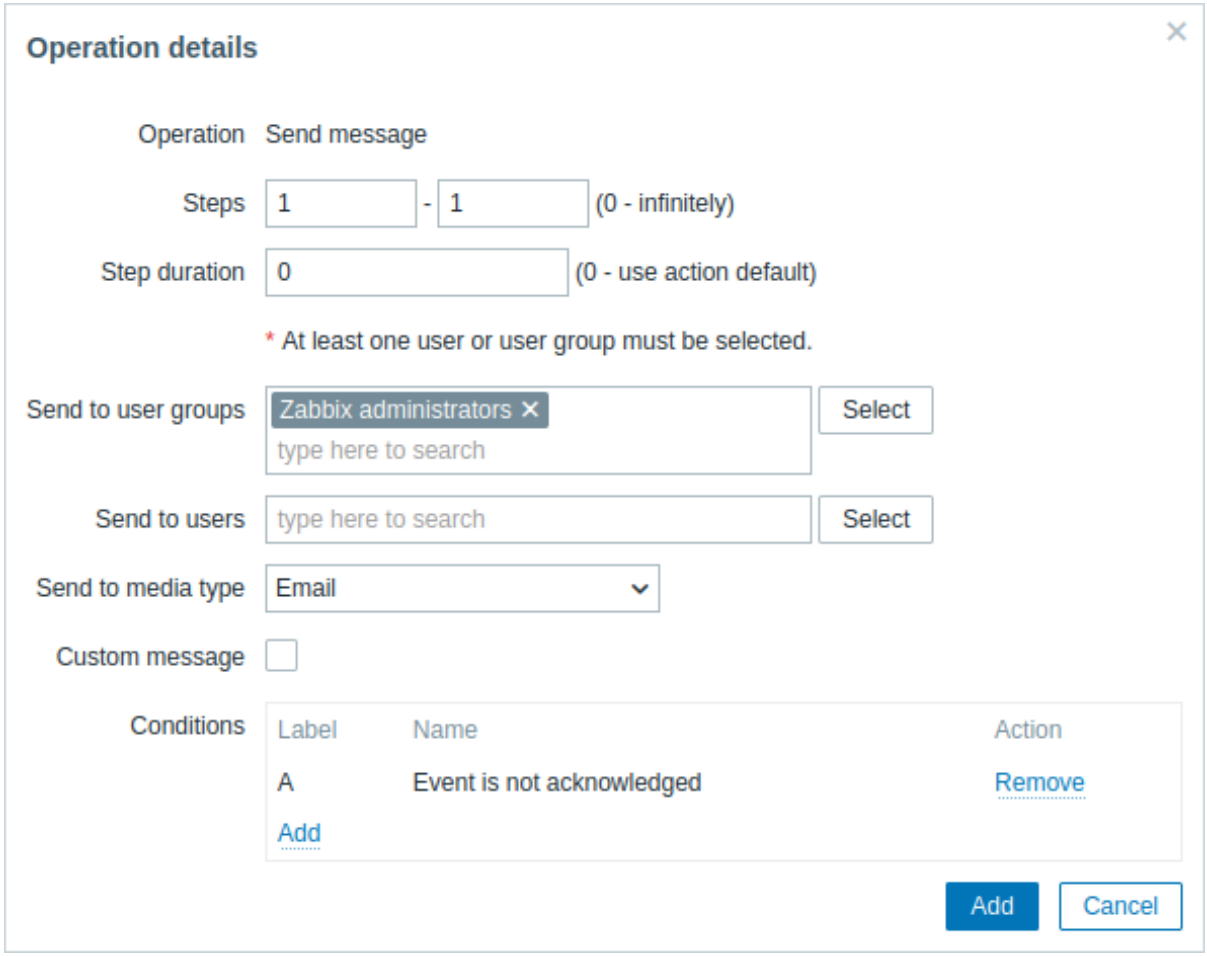
Parametr	Opis
Domyślny czas trwania kroku operacji	Domyślny czas trwania jednego kroku operacji (od 60 sekund do 1 tygodnia). Na przykład godzinny czas trwania kroku oznacza, że jeśli operacja zostanie wykonana, przed następnym krokiem upłynie godzina. Obsługiwane są sufiksy czasu, np. 60s, 1m, 2h, 1d. Obsługiwane są makra użytkownika.

Parametr	Opis
Operacje	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Kroki** — krok(i) eskalacji, do których przypisana jest operacja. **Szczegóły** — typ operacji oraz jej odbiorca/cel. Na liście operacji wyświetlany jest również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasie po nazwie użytkownika) odbiorcy powiadomienia. **Rozpocznij za** — po jakim czasie od zdarzenia operacja zostanie wykonana. **Czas trwania (sek.)** — wyświetlany jest czas trwania kroku. Jeśli krok używa czasu domyślnego, wyświetlane jest *Domyślny*, a jeśli używany jest czas niestandardowy, wyświetlany jest czas. **Działania** — wyświetlane są odnośniki do edycji i usunięcia operacji.
Operacje odzyskiwania	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Szczegóły** — typ operacji oraz jej odbiorca/cel. Na liście operacji wyświetlany jest również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasie po nazwie użytkownika) odbiorcy powiadomienia. **Działania** — wyświetlane są odnośniki do edycji i usunięcia operacji.
Operacje aktualizacji	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Szczegóły** — typ operacji oraz jej odbiorca/cel. Na liście operacji wyświetlany jest również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasie po nazwie użytkownika) odbiorcy powiadomienia. **Działania** — wyświetlane są odnośniki do edycji i usunięcia operacji.
Wstrzymaj operacje dla problemów symptomów	Zaznacz to pole wyboru, aby wstrzymać operacje (po pierwszej operacji) dla problemów symptomów. Pamiętaj, że to ustawienie wpływa tylko na eskalacje problemów; operacje odzyskiwania i aktualizacji nie będą nim objęte. Ta opcja jest dostępna tylko dla *Akcji wyzwalaczy*.
Wstrzymaj operacje dla wyciszonych problemów	Zaznacz to pole wyboru, aby opóźnić rozpoczęcie operacji na czas trwania okresu konserwacji. Po uruchomieniu operacji, po zakończeniu konserwacji, zostaną wykonane wszystkie operacje, w tym także te dotyczące zdarzeń z okresu konserwacji. Pamiętaj, że to ustawienie wpływa tylko na eskalacje problemów; operacje odzyskiwania i aktualizacji nie będą nim objęte. Jeśli odznaczysz to pole wyboru, operacje będą wykonywane bez opóźnienia nawet w trakcie okresu konserwacji. Ta opcja nie jest dostępna dla *Akcji usług*.
Powiadamiaj o anulowanych eskalacjach	Odznacz to pole wyboru, aby wyłączyć powiadomienia o anulowanych eskalacjach (gdy host, pozycja, wyzwalacz lub akcja są wyłączone).

Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Aby skonfigurować szczegóły nowej operacji, kliknij Add w bloku *Operacje*. Aby edytować istniejącą operację, kliknij Edit obok operacji. Otworzy się wyskakujące okno, w którym można edytować szczegóły kroków operacji.

Szczegóły operacji



Parametr	Opis
Operacja	Wybierz operację: **Wyślij wiadomość** - wyślij wiadomość do użytkownika. **<nazwa zdalnego polecenia>** - wykonaj zdalne polecenie. Polecenia są dostępne do wykonania, jeśli zostały wcześniej zdefiniowane w skryptach globalnych z wybranym zakresem *Operacja akcji*. Dla zdarzeń opartych na wykrywaniu i autorejestracji dostępnych jest więcej operacji (zobacz wyżej).
Kroki	Wybierz krok(i), do których ma zostać przypisana operacja w harmonogramie eskalacji: **Od** - wykonuj od tego kroku. **Do** - wykonuj do tego kroku (0=nieskończoność, wykonanie nie będzie ograniczone).
Czas tr-wa-nia kroku	Niestandardowy czas trwania dla tych kroków (0=użyj domyślnego czasu trwania kroku). Obsługiwane są sufiksy czasu, np. 60s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Do tego samego kroku można przypisać kilka operacji. Jeśli dla tych operacji zdefiniowano różny czas trwania kroku, brany jest pod uwagę najkrótszy i stosowany do kroku.
Typ op-er-acji: wyślij wiado-mość	
Wyślij do grup użytkown-ików	Wybierz grupy użytkowników, do których ma zostać wysłana wiadomość. Grupa użytkowników musi mieć co najmniej uprawnienia „odczyt” do hosta, aby otrzymać powiadomienie.

Parametr	Opis
Wyślij do użytkowników	Wybierz użytkowników, do których ma zostać wysłana wiadomość. Użytkownik musi mieć co najmniej uprawnienia „odczyt” do hosta, aby otrzymać powiadomienie.
Wyślij do typu mediów	Wyślij wiadomość do wszystkich dostępnych (skonfigurowanych i włączonych) typów mediów lub tylko do jednego określonego typu.
Własna wiadomość	Jeśli zaznaczone, można skonfigurować własną wiadomość. W przypadku powiadomień o zdarzeniach wewnętrznych przez webhooki własna wiadomość jest obowiązkowa.
Temat	Temat własnej wiadomości. Temat może zawierać makra. Jest ograniczony do 255 znaków.
Wiadomość	Własna wiadomość. Wiadomość może zawierać makra. Jest ograniczona do określonej liczby znaków w zależności od typu bazy danych (zobacz Wysyłanie wiadomości, aby uzyskać więcej informacji).
Typ operacji: zdalne polecenie	
Lista celów	Wybierz cele, na których ma zostać wykonane polecenie: **Bieżący host** - polecenie jest wykonywane na hoście wyzwalacza, który spowodował zdarzenie problemu. Ta opcja nie będzie działać, jeśli w wyzwalaczu występuje wiele hostów. **Host** - wybierz host(y), na których ma zostać wykonane polecenie. **Grupa hostów** - wybierz grupę(y) hostów, na których ma zostać wykonane polecenie. Określenie nadrzędnej grupy hostów niejawnie wybiera wszystkie zagnieżdżone grupy hostów. W ten sposób zdalne polecenie zostanie również wykonane na hostach z grup zagnieżdżonych. Polecenie na hoście jest wykonywane tylko raz, nawet jeśli host pasuje więcej niż raz (np. z kilku grup hostów; indywidualnie i z grupy hostów). Lista celów nie ma znaczenia, jeśli na serwerze Zabbix wykonywany jest skrypt niestandardowy. Wybranie większej liczby celów w takim przypadku powoduje jedynie wielokrotne wykonanie skryptu na serwerze. Zwróć uwagę, że w przypadku skryptów globalnych wybór celu zależy również od ustawienia *Grupa hostów* w konfiguracji skryptu globalnego. Opcja *Lista celów* nie jest dostępna dla *Akcji usługi*, ponieważ w tym przypadku zdalne polecenia są zawsze wykonywane na serwerze Zabbix.
Warunki	Warunek wykonania operacji: **Zdarzenie nie jest potwierdzone** - tylko gdy zdarzenie nie jest potwierdzone. **Zdarzenie jest potwierdzone** - tylko gdy zdarzenie jest potwierdzone. Opcja *Warunki* jest dostępna tylko dla *Akcji wyzwalacza*.

Po zakończeniu kliknij *Dodaj*, aby dodać operację do listy *Operacje*.

1 Wysyłanie wiadomości

Przegląd

Wysłanie wiadomości jest jednym z najlepszych sposobów powiadamiania ludzi o problemie. Dlatego jest to jedna z podstawowych akcji oferowanych przez Zabbix.

Konfiguracja

Aby móc wysyłać i odbierać powiadomienia z Zabbix, należy:

- zdefiniować media, na które mają być wysyłane wiadomości

Jeśli operacja ma miejsce poza okresem When active, zdefiniowanym dla wybranego medium w konfiguracji użytkownika, wiadomość nie zostanie wysłana.

Domyślny poziom ważności wyzwalacza („Not classified") **musi być** zaznaczony w konfiguracji mediów użytkownika, jeśli chcesz otrzymywać powiadomienia o zdarzeniach niezwiązanych z wyzwalaczami, takich jak wykrywanie, aktywna autorejestracja agent lub zdarzenia wewnętrzne.

- skonfigurować operację akcji, która wysyła wiadomość do jednego ze zdefiniowanych mediów

Attention:
Zabbix wysyła powiadomienia tylko do tych użytkowników, którzy mają co najmniej uprawnienia „odczyt" do host, który wygenerował zdarzenie. Co najmniej jeden host z wyrażenia wyzwalacza musi być dostępny.

Możesz skonfigurować niestandardowe scenariusze wysyłania wiadomości przy użyciu eskalacji.

Aby pomyślnie odbierać i odczytywać wiadomości e-mail z Zabbix, serwery/klienci poczty e-mail muszą obsługiwać standardowy format „SMTP/MIME email", ponieważ Zabbix wysyła dane UTF-8 (jeśli temat zawiera wyłącznie znaki ASCII, nie jest kodowany w UTF-8). Temat i treść wiadomości są kodowane w base64, aby zachować zgodność ze standardem formatu „SMTP/MIME email".

Limit wiadomości po rozwinięciu wszystkich makr jest taki sam jak limit wiadomości dla zdalnych poleceń.

Śledzenie wiadomości

Możesz wyświetlić status wiadomości wysłanych w *Monitoring* → *Problemy*.

W kolumnie *Akcje* możesz zobaczyć podsumowane informacje o wykonanych działaniach. Zielone liczby oznaczają tam wysłane wiadomości, czerwone — wiadomości, których nie udało się wysłać. *W toku* wskazuje, że działanie zostało zainicjowane. *Niepowodzenie* informuje, że żadne działanie nie zostało wykonane pomyślnie.

Jeśli klikniesz czas zdarzenia, aby wyświetlić szczegóły zdarzenia, będziesz mógł zobaczyć szczegóły wiadomości wysłanych (lub niewysłanych) z powodu tego zdarzenia w bloku *Akcje*.

W *Raporty* → *Dziennik akcji* zobaczysz szczegóły wszystkich wykonanych działań dla tych zdarzeń, dla których skonfigurowano akcję.

2 Zdalne polecenia

Przegląd

Za pomocą zdalnych poleceń można zdefiniować, że określone, wcześniej zdefiniowane polecenie zostanie automatycznie wykonane na monitorowanym hoście po spełnieniu określonego warunku.

Dzięki temu zdalne polecenia są potężnym mechanizmem inteligentnego, proaktywnego monitorowania.

W najbardziej oczywistych zastosowaniach tej funkcji można spróbować:

- Automatycznie restartować aplikację (serwer WWW, middleware, CRM), jeśli nie odpowiada
- Użyć polecenia IPMI „reboot", aby ponownie uruchomić zdalny serwer, jeśli nie odpowiada na żądania
- Automatycznie zwalniać miejsce na dysku (usuwanie starszych plików, czyszczenie /tmp), jeśli kończy się wolne miejsce na dysku
- Migrować maszynę wirtualną z jednego fizycznego serwera na inny w zależności od obciążenia CPU
- Dodawać nowe węzły do środowiska chmurowego przy niewystarczających zasobach CPU (dysku, pamięci lub innych)

Konfigurowanie akcji dla zdalnych poleceń jest podobne do konfigurowania akcji wysyłania wiadomości; jedyna różnica polega na tym, że Zabbix wykona polecenie zamiast wysyłać wiadomość.

Zdalne polecenia mogą być wykonywane przez serwer Zabbix, proxy lub agent. Zdalne polecenia na Zabbix agent mogą być wykonywane bezpośrednio przez serwer Zabbix lub za pośrednictwem Zabbix proxy. Zarówno na Zabbix agent, jak i Zabbix proxy zdalne polecenia są domyślnie wyłączone. Można je włączyć przez:

- dodanie parametru `AllowKey=system.run[*]` w konfiguracji agenta;
- ustawienie parametru EnableRemoteCommands na „1" w konfiguracji proxy.

Zdalne polecenia wykonywane przez serwer Zabbix są uruchamiane zgodnie z opisem w Command execution, łącznie ze sprawdzaniem kodu wyjścia.

Zdalne polecenia są wykonywane nawet wtedy, gdy docelowy host jest w trybie konserwacji.

Limit polecenia zdalnego

Limit polecenia zdalnego po rozwiązaniu wszystkich makr zależy od typu bazy danych i zestawu znaków (znaki inne niż ASCII wymagają więcej niż jednego bajta do zapisania):

Database	Limit in characters	Limit in bytes
MySQL	65535	65535
PostgreSQL	65535	not limited
SQLite (only Zabbix proxy)	65535	not limited

Dane wyjściowe wykonania polecenia zdalnego (wartość zwracana) są ograniczone do 16 MB (w tym końcowe białe znaki, które są obcinane). Limit zdalnego polecenia IPMI zależy od zainstalowanej biblioteki IPMI. Należy pamiętać, że limity bazy danych mają zastosowanie do wszystkich poleceń zdalnych.

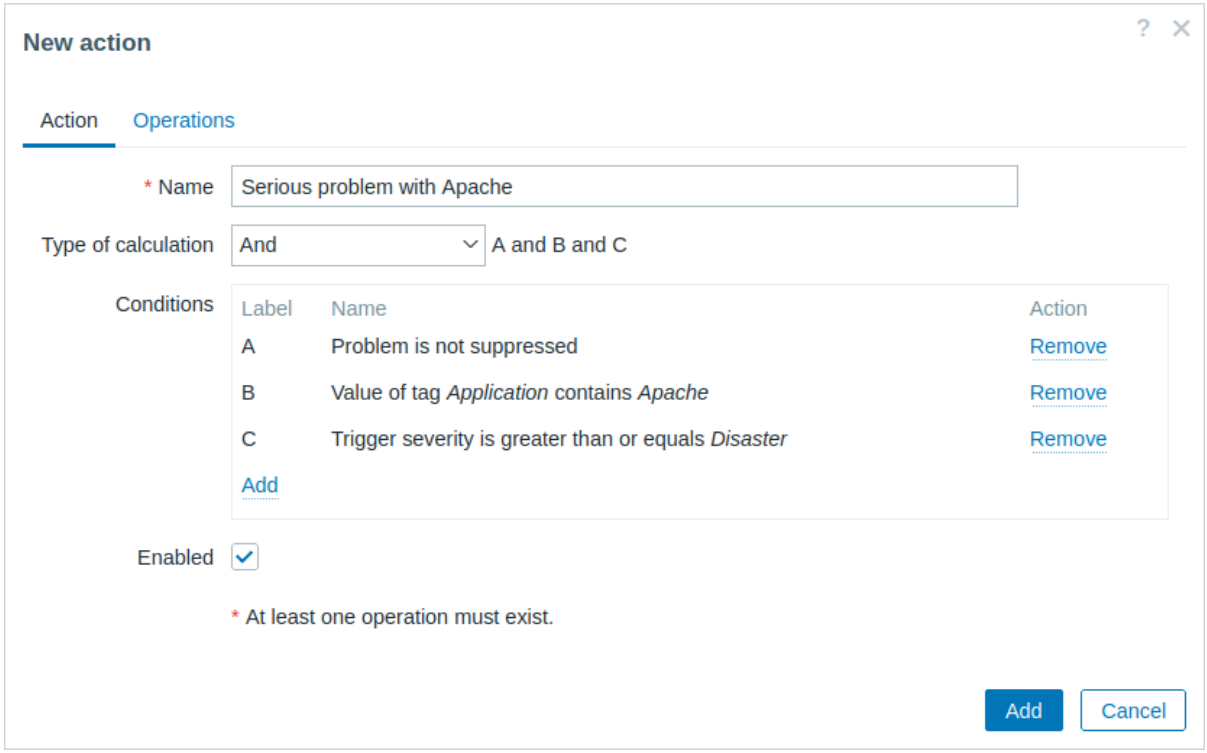
Konfiguracja

Te zdalne polecenia, które są wykonywane na Zabbix agent (niestandardowe skrypty), muszą być najpierw włączone w konfiguracji agenta.

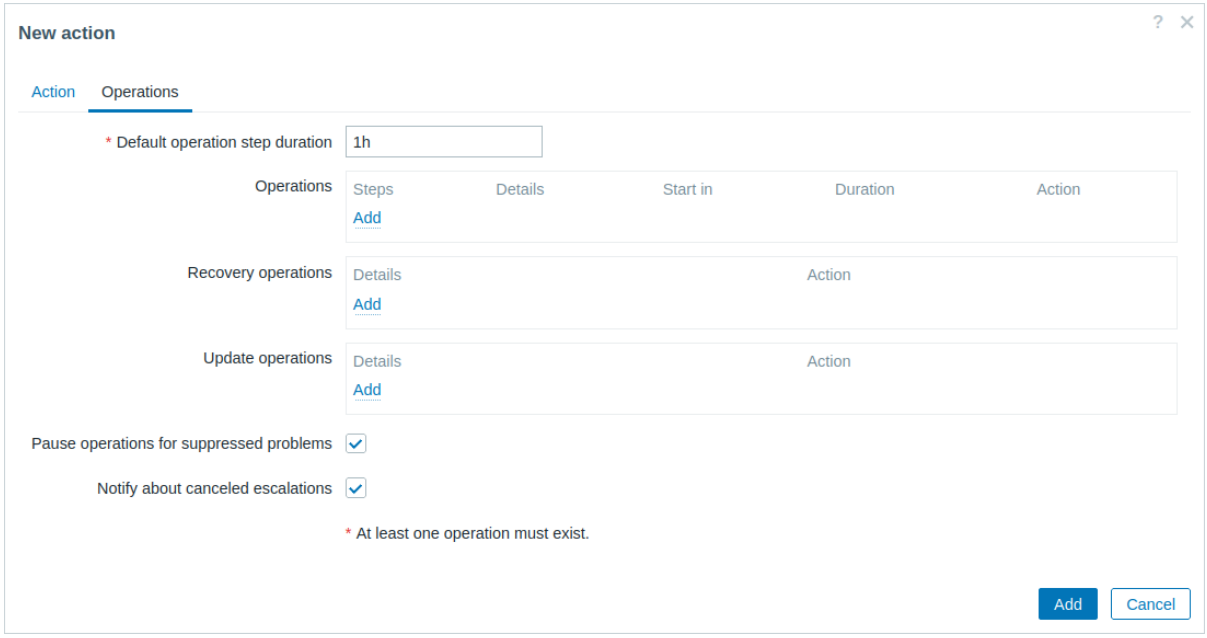
Upewnij się, że parametr AllowKey=system.run[<command>,*] jest dodany dla każdego dozwolonego polecenia w konfiguracji agenta, aby zezwolić na określone polecenie w trybie nowait. Jeśli zmieniasz ten parametr, uruchom ponownie demon agenta.

Następnie, podczas konfigurowania nowej akcji w *Alerty* → *Akcje* → *Akcje wyzwalaczy*:

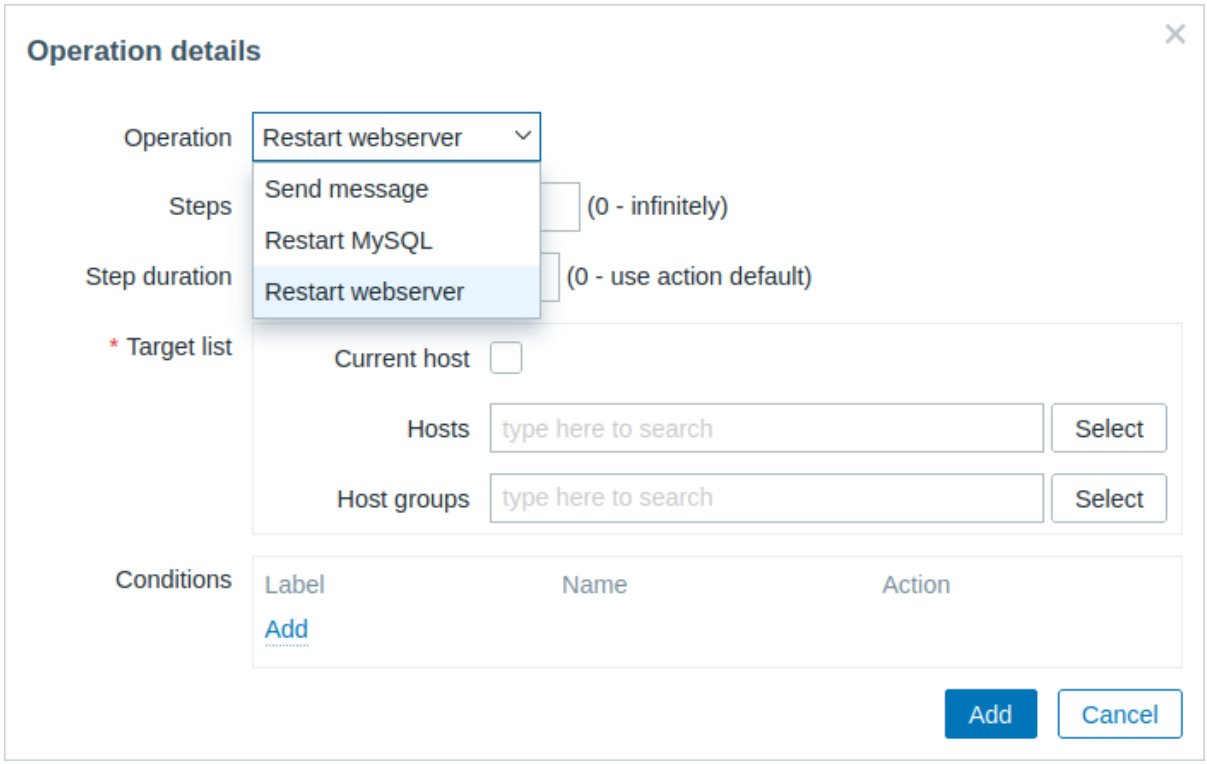
1. Zdefiniuj odpowiednie warunki, na przykład ustaw, aby akcja była aktywowana przy dowolnych problemach o poziomie katastrofa dla jednej z aplikacji Apache.



2. Na karcie *Operacje* kliknij *Dodaj* w bloku *Operacje*, *Operacje odzyskiwania* lub *Operacje aktualizacji*.



3. Wybierz jeden ze wstępnie zdefiniowanych skryptów z listy rozwijanej *Operacja* i ustaw dla skryptu *Listę celów*.



Predefiniowane skrypty

Skrypty dostępne dla operacji akcji (webhook, skrypt, SSH, Telnet, IPMI) są definiowane w skryptach globalnych.

Na przykład:

```
sudo /etc/init.d/apache restart
```

W tym przypadku Zabbix spróbuje ponownie uruchomić proces Apache. W przypadku tego polecenia należy upewnić się, że polecenie jest wykonywane na Zabbix agent (kliknij przycisk *Zabbix agent* przy polu *Execute on*).

Attention:
Zwróć uwagę na użycie **sudo** — użytkownik Zabbix domyślnie nie ma uprawnień do ponownego uruchamiania usług systemowych. Poniżej znajdują się wskazówki dotyczące konfiguracji **sudo**.

Note:
Począwszy od Zabbix agent 7.0, polecenia zdalne mogą być również wykonywane na agencie działającym w trybie aktywnym. Zabbix agent — aktywny lub pasywny — powinien działać na zdalnym hoście i wykonuje polecenia w tle.

Polecenia zdalne na Zabbix agent są wykonywane bez limitu czasu przez klucz system.run[,nowait] i nie są sprawdzane pod kątem wyników wykonania. Na Zabbix serwer i Zabbix proxy polecenia zdalne są wykonywane z limitem czasu ustawionym w parametrze TrapperTimeout pliku zabbix_server.conf lub zabbix_proxy.conf i są sprawdzane pod kątem wyników wykonania. Aby uzyskać dodatkowe informacje, zobacz *Limit czasu skryptu*.

Uprawnienia dostępu

Upewnij się, że użytkownik 'zabbix' ma uprawnienia do wykonywania skonfigurowanych poleceń. Można rozważyć użycie **sudo**, aby przyznać dostęp do uprzywilejowanych poleceń. Aby skonfigurować dostęp, wykonaj jako root:

```
visudo
```

Przykładowe linie, które można umieścić w pliku *sudoers*:

```
# pozwala użytkownikowi 'zabbix' uruchamiać wszystkie polecenia bez hasła.
zabbix ALL=NOPASSWD: ALL

# pozwala użytkownikowi 'zabbix' restartować apache bez hasła.
zabbix ALL=NOPASSWD: /etc/init.d/apache restart
```

Note:
W niektórych systemach plik *sudoers* uniemożliwia użytkownikom nielokalnym wykonywanie poleceń. Aby to zmienić, zakomentuj opcję **requiretty** w */etc/sudoers*.

Zdalne polecenia przy wielu interfejsach

Jeśli system docelowy ma wiele interfejsów wybranego typu (agent Zabbix lub IPMI), zdalne polecenia będą wykonywane na domyślnym interfejsie.

Możliwe jest wykonywanie zdalnych poleceń przez SSH i Telnet przy użyciu interfejsu innego niż interfejs agenta Zabbix. Dostępny interfejs do użycia jest wybierany w następującej kolejności:

- Domyślny interfejs agenta Zabbix
- Domyślny interfejs SNMP
- Domyślny interfejs JMX
- Domyślny interfejs IPMI

Zdalne polecenia IPMI

Dla zdalnych poleceń IPMI należy używać następującej składni:

```
<command> [<value>]
```

gdzie

- <command> - jedno z poleceń IPMI bez spacji
- <value> - 'on', 'off' lub dowolna nieujemna liczba całkowita. <value> jest parametrem opcjonalnym.

Przykłady

Przykłady globalnych skryptów, które mogą być używane jako zdalne polecenia w operacjach akcji.

Przykład 1

Restart systemu Windows przy określonym warunku.

Aby automatycznie zrestartować system Windows po wykryciu problemu przez Zabbix, zdefiniuj następujący skrypt:

Parametr skryptu	Wartość
Zakres	'Operacja akcji'
Typ	'Skrypt'
Polecenie	c:\windows\system32\shutdown.exe -r -f

Przykład 2

Uruchom ponownie host przy użyciu sterowania IPMI.

Parametr skryptu	Wartość
Zakres	'Operacja akcji'
Typ	'IPMI'
Polecenie	reset

Przykład 3

Wyłącz zasilanie hosta za pomocą sterowania IPMI.

Parametr skryptu	Wartość
Zakres	'Operacja akcji'
Typ	'IPMI'
Polecenie	power off

3 Dodatkowe operacje

Przegląd

W tej sekcji możesz znaleźć szczegóły dodatkowych operacji dla zdarzeń wykrywania/autorejestracji.

Dodawanie hosta

Hosty są dodawane podczas procesu wykrywania, natychmiast po wykryciu hosta, a nie dopiero na końcu procesu wykrywania.

Note:
Ponieważ wykrywanie sieci może zająć trochę czasu z powodu wielu niedostępnych hostów/usług, zaleca się cierpliwość i używanie rozsądnych zakresów IP.

Podczas dodawania hosta jego nazwa jest ustalana przez standardową funkcję **gethostbyname**. Jeśli host może zostać rozwiązany, używana jest rozwiązana nazwa. Jeśli nie, używany jest adres IP. Ponadto, jeśli jako nazwa hosta musi zostać użyty adres IPv6, wszystkie ":" (dwukropki) są zastępowane przez "_" (podkreślenia), ponieważ dwukropki nie są dozwolone w nazwach hostów.

Attention:
Jeśli wykrywanie jest wykonywane przez proxy, obecnie wyszukiwanie nazwy hosta nadal odbywa się na serwerze Zabbix.

Attention:
Jeśli host o tej samej nazwie co nowo wykryty już istnieje w konfiguracji Zabbix, Zabbix doda **_N** do nazwy hosta, gdzie **N** jest rosnącą liczbą, zaczynając od 2.

4 Używanie makr w wiadomościach

Przegląd

W tematach wiadomości i treści wiadomości można używać makr w celu bardziej efektywnego raportowania problemów.

Oprócz wielu wbudowanych makr obsługiwane są również makra użytkownika oraz makra wyrażeń. Dostępna jest pełna lista makr obsługiwanych przez Zabbix.

Przykłady

Przykłady tutaj ilustrują, jak można używać makr w wiadomościach.

Przykład 1

Temat wiadomości:

```
Problem: {TRIGGER.NAME}
```

Po otrzymaniu wiadomości temat wiadomości zostanie zastąpiony czymś takim jak:

```
Problem: Obciążenie procesora jest zbyt wysokie na serwerze Zabbix
```

Przykład 2

Wiadomość:

```
Obciążenie procesora wynosi: {?last(/zabbix.zabbix.com/system.cpu.load[,avg1])}
```

Po otrzymaniu wiadomości treść wiadomości zostanie zastąpiona czymś takim jak:

```
Obciążenie procesora wynosi: 1.45
```

Przykład 3

Wiadomość:

```
Ostatnia wartość: {?last(/{HOST.HOST}/{ITEM.KEY})}
MAX z 15 minut: {?max(/{HOST.HOST}/{ITEM.KEY},15m)}
MIN z 15 minut: {?min(/{HOST.HOST}/{ITEM.KEY},15m)}
```

Po otrzymaniu wiadomości jej treść zostanie zastąpiona czymś takim jak:

```
Ostatnia wartość: 1.45
MAX z 15 minut: 2.33
MIN z 15 minut: 1.01
```

Przykład 4

Wiadomość:

```
http://<server_ip_or_name>/zabbix/tr_events.php?triggerid={TRIGGER.ID}&eventid={EVENT.ID}
```

Po otrzymaniu wiadomości będzie ona zawierać link do strony *Szczegóły zdarzenia*, która zawiera informacje o zdarzeniu, jego wyzwalaczu oraz listę najnowszych zdarzeń wygenerowanych przez ten sam wyzwalacz.

Przykład 5

Informowanie o wartościach z kilku hostów w wyrażeniu wyzwalacza.

Wiadomość:

```
Nazwa problemu: {TRIGGER.NAME}
Wyrażenie wyzwalacza: {TRIGGER.EXPRESSION}

1. Wartość pozycji na hoście {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
2. Wartość pozycji na hoście {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})
```

Po otrzymaniu wiadomości zostanie ona zastąpiona czymś takim:

```
Nazwa problemu: Obciążenie procesora jest zbyt wysokie na lokalnym hoście
Wyrażenie wyzwalacza: last(/Myhost/system.cpu.load[percpu,avg1])>5 or last(/Myotherhost/system.cpu.load[pe

1. Wartość pozycji na hoście Myhost: 0.83 (Obciążenie procesora (średnia 1-minutowa na rdzeń))
2. Wartość pozycji na hoście Myotherhost: 5.125 (Obciążenie procesora (średnia 1-minutowa na rdzeń))
```

Przykład 6

Otrzymywanie szczegółów zarówno zdarzenia problemu, jak i zdarzenia odzyskania w wiadomości odzyskania:

Wiadomość:

```
Problem:

ID zdarzenia: {EVENT.ID}
Wartość zdarzenia: {EVENT.VALUE}
Status zdarzenia: {EVENT.STATUS}
Czas zdarzenia: {EVENT.TIME}
Data zdarzenia: {EVENT.DATE}
Wiek zdarzenia: {EVENT.AGE}
Potwierdzenie zdarzenia: {EVENT.ACK.STATUS}
Historia aktualizacji zdarzenia: {EVENT.UPDATE.HISTORY}

Odzyskanie:

ID zdarzenia: {EVENT.RECOVERY.ID}
```

```
Wartość zdarzenia: {EVENT.RECOVERY.VALUE}
Status zdarzenia: {EVENT.RECOVERY.STATUS}
Czas zdarzenia: {EVENT.RECOVERY.TIME}
Data zdarzenia: {EVENT.RECOVERY.DATE}
Dane operacyjne: {EVENT.OPDATA}
```

Po otrzymaniu wiadomości makra zostaną zastąpione czymś takim:

```
Problem:

ID zdarzenia: 21874
Wartość zdarzenia: 1
Status zdarzenia: PROBLEM
Czas zdarzenia: 13:04:30
Data zdarzenia: 2018.01.02
Wiek zdarzenia: 5m 0s
Potwierdzenie zdarzenia: Tak
Historia aktualizacji zdarzenia: 2018.01.02 13:05:51 "John Smith (Admin)"
Działania: potwierdzono.

Odzyskanie:

ID zdarzenia: 21896
Wartość zdarzenia: 0
Status zdarzenia: OK
Czas zdarzenia: 13:10:07
Data zdarzenia: 2018.01.02
Dane operacyjne: Bieżąca wartość to 0.83
```

3 Operacje odzyskiwania

Przegląd

Operacje odzyskiwania umożliwiają otrzymywanie powiadomień, gdy problemy zostaną rozwiązane.

Zarówno wiadomości, jak i zdalne polecenia są obsługiwane w operacjach odzyskiwania. Chociaż można dodać kilka operacji, eskalacja nie jest obsługiwana — wszystkie operacje są przypisane do jednego kroku i dlatego zostaną wykonane jednocześnie.

Przypadki użycia

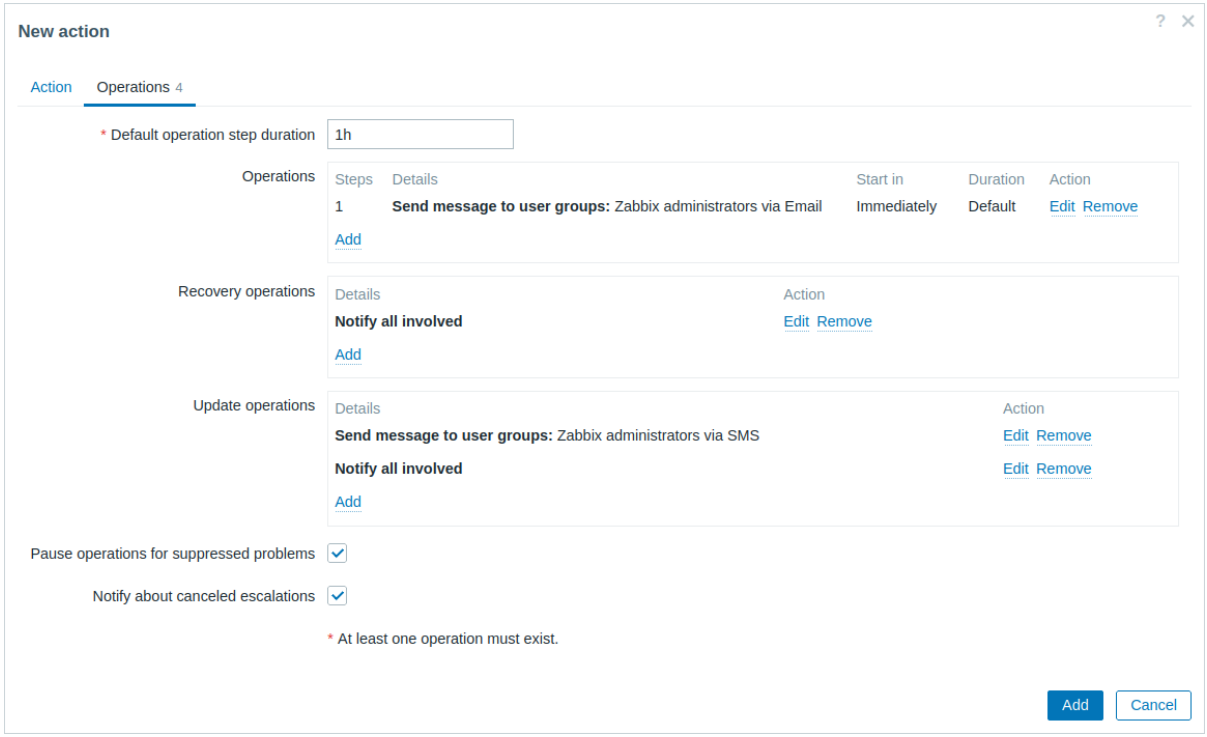
Niektóre przypadki użycia operacji odzyskiwania są następujące:

1. Powiadom przy odzyskaniu wszystkich użytkowników, którzy zostali powiadomieni o problemie:
   - Wybierz *Powiadom wszystkich zaangażowanych* jako typ operacji.
2. Miej wiele operacji przy odzyskaniu: wyślij powiadomienie i wykonaj zdalne polecenie:
   - Dodaj typy operacji do wysyłania wiadomości i wykonywania polecenia.
3. Otwórz zgłoszenie w zewnętrznym systemie helpdesk/ticketing i zamknij je po rozwiązaniu problemu:
   - Utwórz zewnętrzny skrypt, który komunikuje się z systemem helpdesk.
   - Utwórz akcję z operacją, która wykonuje ten skrypt i w ten sposób otwiera zgłoszenie.
   - Skonfiguruj operację odzyskiwania, która wykonuje ten skrypt z innymi parametrami i zamyka zgłoszenie.
   - Użyj makra {EVENT.ID}, aby odwołać się do pierwotnego problemu.

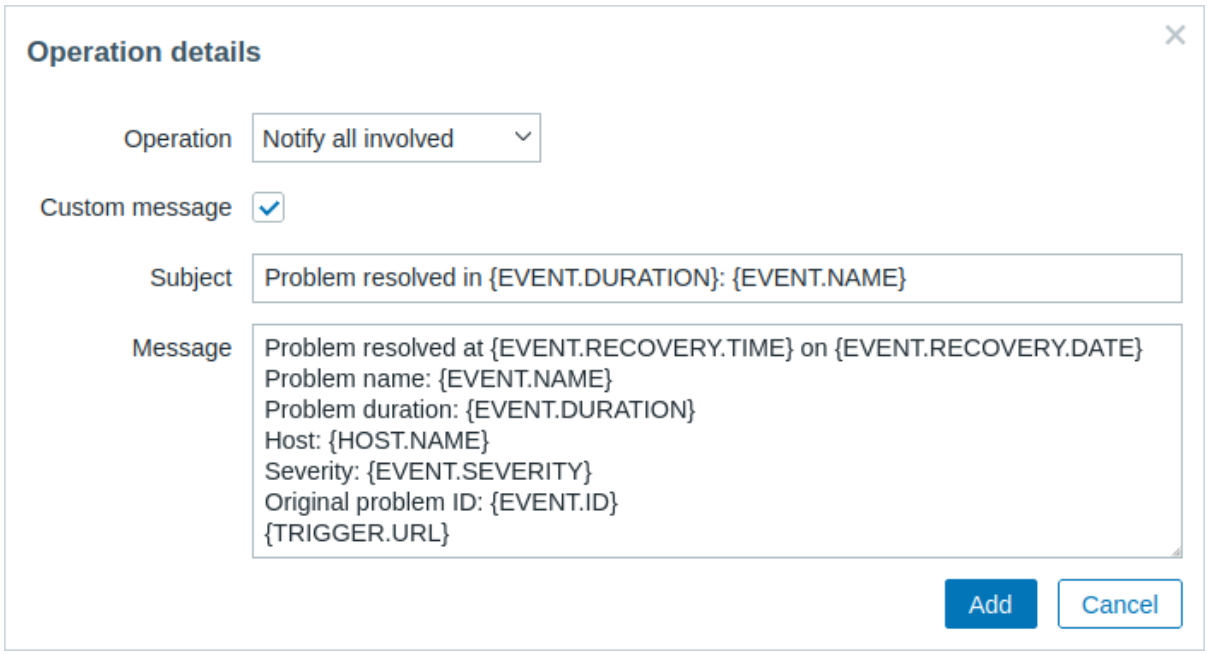
Konfigurowanie operacji odzyskiwania

Aby skonfigurować operację odzyskiwania, wykonaj następujące kroki:

1. Przejdź do zakładki *Operations* w formularzu konfiguracji akcji.
2. Aby skonfigurować nową operację odzyskiwania, kliknij *Add* w sekcji *Recovery operations*. Aby edytować istniejącą operację, kliknij *Edit* obok operacji.
3. Skonfiguruj szczegóły operacji.



Szczegóły operacji odzyskiwania



Dla zdarzeń odzyskiwania dostępne są trzy typy operacji:

- **Wyślij wiadomość** - wyślij wiadomość odzyskiwania do określonego użytkownika;
- **Powiadom wszystkich zaangażowanych** - wyślij wiadomość odzyskiwania do wszystkich użytkowników, którzy zostali powiadomieni o zdarzeniu problemu;
- **<remote command name>** - wykonaj zdalne polecenie. Polecenia są dostępne do wykonania, jeśli zostały wcześniej zdefiniowane w global scripts z wybranym zakresem *Action operation*.

Parametry dla każdego typu operacji są opisane poniżej. Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką. Po zakończeniu kliknij *Add*, aby dodać operację do listy *Recovery operations*.

Note:
Należy pamiętać, że jeśli ten sam odbiorca jest zdefiniowany w kilku typach operacji bez określonej *Custom message*, zduplikowane powiadomienia nie są wysyłane.

Typ operacji: wyślij wiadomość

Parametr	Opis
Wyślij do grup użytkowników	Wybierz grupy użytkowników, do których ma zostać wysłana wiadomość o odzyskaniu. Aby otrzymać powiadomienie, grupa użytkowników musi mieć co najmniej uprawnienia „odczyt” do hosta permissions.
Wyślij do użytkowników	Wybierz użytkowników, do których ma zostać wysłana wiadomość o odzyskaniu. Aby otrzymać powiadomienie, użytkownik musi mieć co najmniej uprawnienia „odczyt” do hosta permissions.
Wyślij do typu nośnika	Wyślij domyślną wiadomość o odzyskaniu do wszystkich dostępnych (skonfigurowanych i włączonych) typów nośników lub tylko do jednego określonego typu.
Własna wiadomość	Jeśli ta opcja jest zaznaczona, można zdefiniować własną wiadomość.
Temat	Temat własnej wiadomości. Temat może zawierać makra.
Wiadomość	Własna wiadomość. Wiadomość może zawierać makra.

Typ operacji: zdalne polecenie

Parametr	Opis
Lista celów	Wybierz cele, na których ma zostać wykonane polecenie: **Bieżący host** - polecenie jest wykonywane na hoście wyzwalacza, który spowodował zdarzenie problemu. Ta opcja nie będzie działać, jeśli w wyzwalaczu występuje wiele hostów. **Host** - wybierz host(y), na których ma zostać wykonane polecenie. **Grupa hostów** - wybierz grupę(y) hostów, na których ma zostać wykonane polecenie. Określenie nadrzędnej grupy hostów domyślnie wybiera wszystkie zagnieżdżone grupy hostów. W związku z tym zdalne polecenie zostanie również wykonane na hostach z grup zagnieżdżonych. Polecenie na hoście jest wykonywane tylko raz, nawet jeśli host pasuje więcej niż jeden raz (np. z kilku grup hostów; indywidualnie i z grupy hostów). Lista celów nie ma znaczenia, jeśli polecenie jest wykonywane na serwerze Zabbix. Wybranie większej liczby celów w takim przypadku powoduje jedynie, że polecenie zostanie wykonane na serwerze więcej razy. Zwróć uwagę, że w przypadku skryptów globalnych wybór celu zależy również od ustawienia *Grupa hostów* w konfiguracji skryptu globalnego.

Typ operacji: powiadom wszystkich zaangażowanych

Parametr		Opis
Własna wiadomość		Jeśli zaznaczone, można zdefiniować własną wiadomość.
	Temat	Temat własnej wiadomości. Temat może zawierać makra.
	Wiadomość	Własna wiadomość. Wiadomość może zawierać makra.

4 Operacje aktualizacji

Przegląd

Operacje aktualizacji są dostępne w akcjach z następującymi źródłami zdarzeń:

- *Wyzwalacze* - gdy problemy są aktualizowane przez innych użytkowników, tj. dodano komentarz, potwierdzono, zmieniono ważność, zamknięto (ręcznie);
- *Usługi* - gdy ważność usługi uległa zmianie, ale usługa nadal nie została przywrócona.

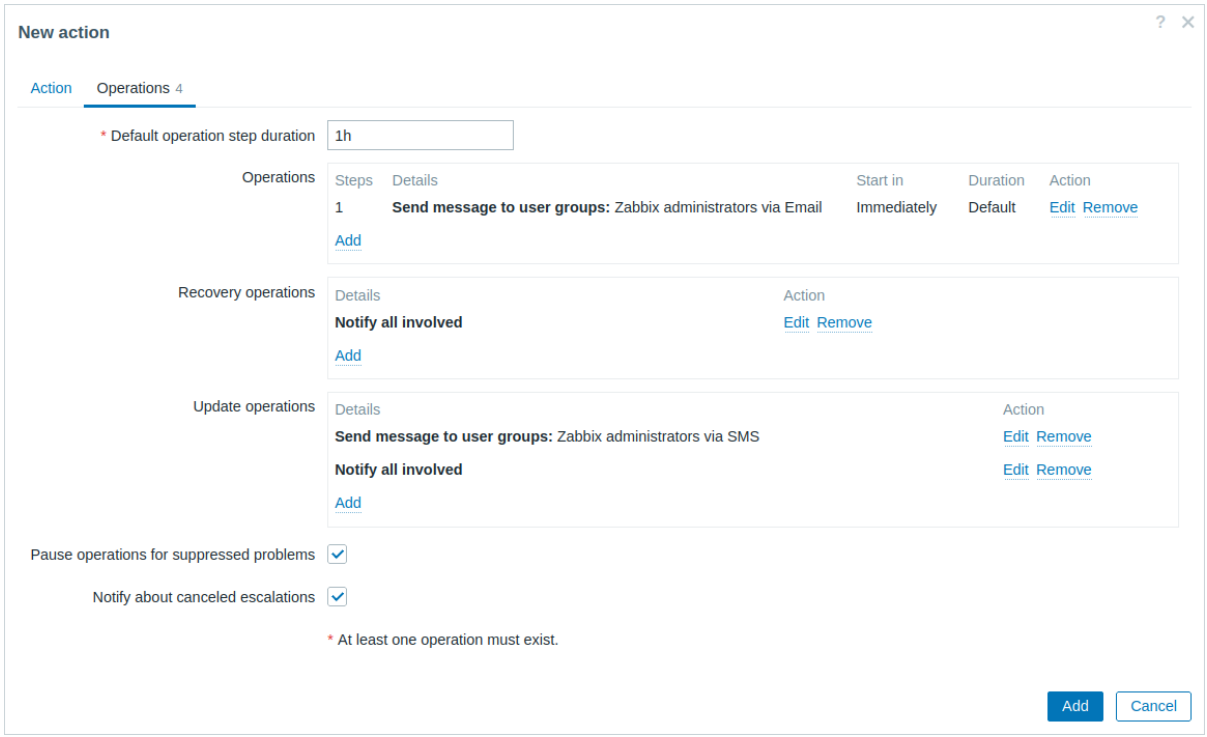
Należy pamiętać, że użytkownicy nie otrzymują powiadomień o własnych aktualizacjach.

W operacjach aktualizacji obsługiwane są zarówno wiadomości, jak i zdalne polecenia. Chociaż można dodać kilka operacji, eskalacja nie jest obsługiwana — wszystkie operacje są przypisywane do jednego kroku i dlatego będą wykonywane jednocześnie.

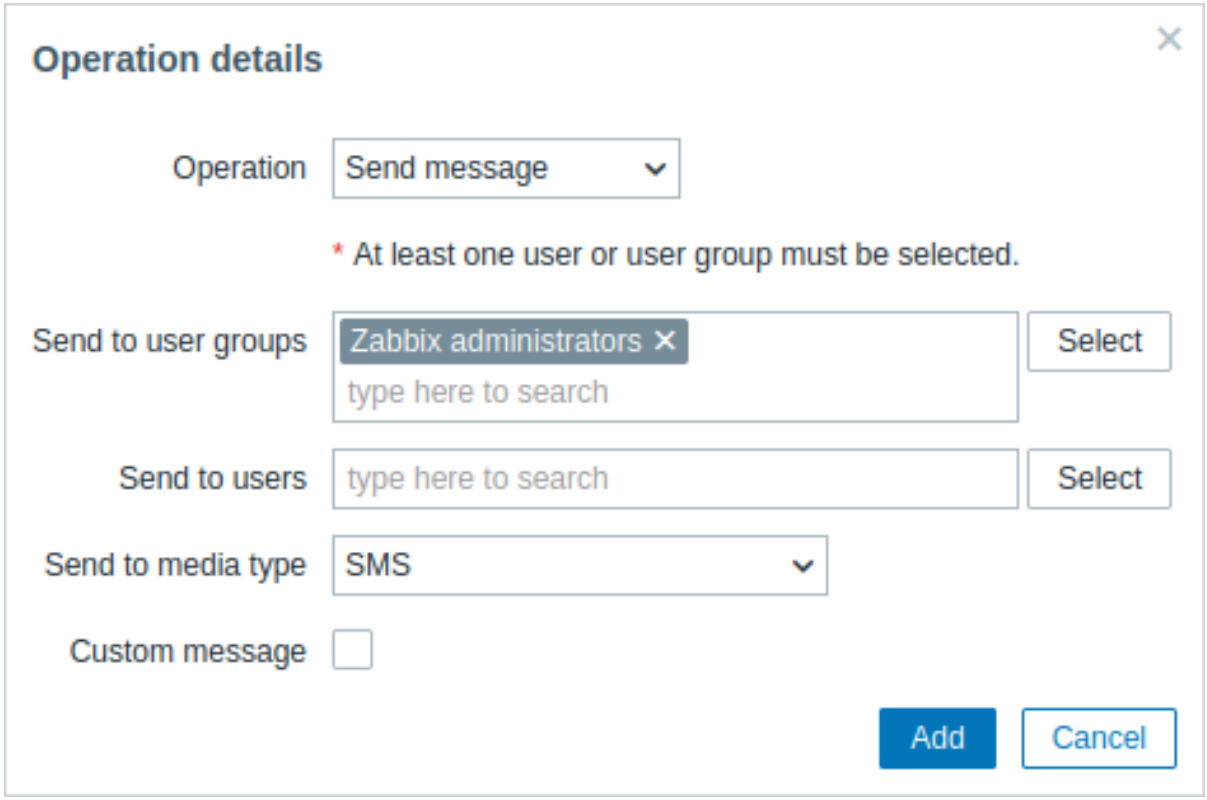
Konfigurowanie operacji aktualizacji

Aby skonfigurować operację aktualizacji, wykonaj następujące kroki:

1. Przejdź do zakładki *Operacje* w formularzu konfiguracji akcji.
2. Aby skonfigurować nową operację aktualizacji, kliknij *Dodaj* w sekcji *Operacje aktualizacji*. Aby edytować istniejącą operację, kliknij *Edytuj* obok operacji.
3. Skonfiguruj szczegóły operacji.



Szczegóły operacji aktualizacji



Operacje aktualizacji oferują ten sam zestaw parametrów co Operacje odzyskiwania.

5 Eskalacje

Przegląd

Za pomocą eskalacji można tworzyć niestandardowe scenariusze wysyłania powiadomień lub wykonywania zdalnych poleceń.

W praktyce oznacza to, że:

- Użytkownicy mogą być natychmiast informowani o nowych problemach.
- Powiadomienia mogą być powtarzane do momentu rozwiązania problemu.
- Wysłanie powiadomienia może zostać opóźnione.
- Powiadomienia mogą być eskalowane do innej, „wyższej" grupy użytkowników.
- Zdalne polecenia mogą być wykonywane natychmiast lub wtedy, gdy problem nie zostanie rozwiązany przez dłuższy czas.

Akcje są eskalowane na podstawie **kroku eskalacji**. Każdy krok ma określony czas trwania.

Można zdefiniować zarówno domyślny czas trwania, jak i niestandardowy czas trwania poszczególnego kroku. Minimalny czas trwania jednego kroku eskalacji wynosi 60 sekund.

Można uruchamiać akcje, takie jak wysyłanie powiadomień lub wykonywanie poleceń, od dowolnego kroku. Krok pierwszy służy do działań natychmiastowych. Jeśli chcesz opóźnić akcję, możesz przypisać ją do późniejszego kroku. Dla każdego kroku można zdefiniować kilka akcji.

Liczba kroków eskalacji nie jest ograniczona.

Eskalacje są definiowane podczas konfigurowania operacji. Eskalacje są obsługiwane tylko dla operacji problemowych, a nie dla odzyskiwania.

Różne aspekty zachowania eskalacji

Rozważmy, co dzieje się w różnych okolicznościach, jeśli akcja zawiera kilka kroków eskalacji.

Sytuacja	Zachowanie
Dany host przechodzi w tryb konserwacji po wysłaniu początkowego powiadomienia o problemie	W zależności od ustawienia *Pause operations for suppressed problems* w konfiguracji akcji, wszystkie pozostałe kroki eskalacji są wykonywane albo z opóźnieniem spowodowanym okresem konserwacji, albo bez opóźnienia. Okres konserwacji nie anuluje operacji.
*Okres czasu zdefiniowany w warunku akcji **Time period** kończy się po wysłaniu początkowego powiadomienia*	Wszystkie pozostałe kroki eskalacji są wykonywane. Warunek *Time period* nie może zatrzymać operacji; ma wpływ na to, kiedy akcje są uruchamiane/nieuruchamiane, a nie na operacje.
Problem rozpoczyna się podczas konserwacji i trwa dalej (nie zostaje rozwiązany) po jej zakończeniu	W zależności od ustawienia *Pause operations for suppressed problems* w konfiguracji akcji, wszystkie kroki eskalacji są wykonywane albo od momentu zakończenia konserwacji, albo natychmiast.
Problem rozpoczyna się podczas konserwacji typu no-data i trwa dalej (nie zostaje rozwiązany) po jej zakończeniu	Musi zaczekać, aż wyzwalacz się uruchomi, zanim zostaną wykonane wszystkie kroki eskalacji.
Różne eskalacje następują po sobie w krótkich odstępach czasu i nakładają się	Wykonanie każdej nowej eskalacji zastępuje poprzednią eskalację, ale co najmniej jeden krok eskalacji z poprzedniej eskalacji jest zawsze wykonywany. To zachowanie ma znaczenie w akcjach dla zdarzeń tworzonych przy KAŻDEJ ocenie problemu przez wyzwalacz.
Podczas trwającej eskalacji (na przykład podczas wysyłania wiadomości), na podstawie dowolnego typu zdarzenia: - akcja zostaje wyłączona Na podstawie zdarzenia wyzwalacza: - wyzwalacz zostaje wyłączony - host lub pozycja zostaje wyłączona Na podstawie zdarzenia wewnętrznego dotyczącego wyzwalaczy: - wyzwalacz zostaje wyłączony Na podstawie zdarzenia wewnętrznego dotyczącego pozycji/reguł wykrywania niskopoziomowego: - pozycja zostaje wyłączona - host zostaje wyłączony	Wiadomość będąca w trakcie wysyłania zostaje wysłana, a następnie wysyłana jest jeszcze jedna wiadomość w ramach eskalacji. Ta kolejna wiadomość będzie miała na początku treści komunikat anulowania (*NOTE: Escalation canceled*) z podaniem przyczyny (na przykład *NOTE: Escalation canceled: action '<Action name>' disabled*). W ten sposób odbiorca jest informowany, że eskalacja została anulowana i żadne dalsze kroki nie zostaną wykonane. Ta wiadomość jest wysyłana do wszystkich, którzy wcześniej otrzymali powiadomienia. Przyczyna anulowania jest również zapisywana w pliku dziennika serwera (od Debug Level 3=Warning). Zwróć uwagę, że wiadomość *Escalation canceled* jest wysyłana również wtedy, gdy operacje są zakończone, ale operacje odzyskiwania są skonfigurowane i nie zostały jeszcze wykonane.

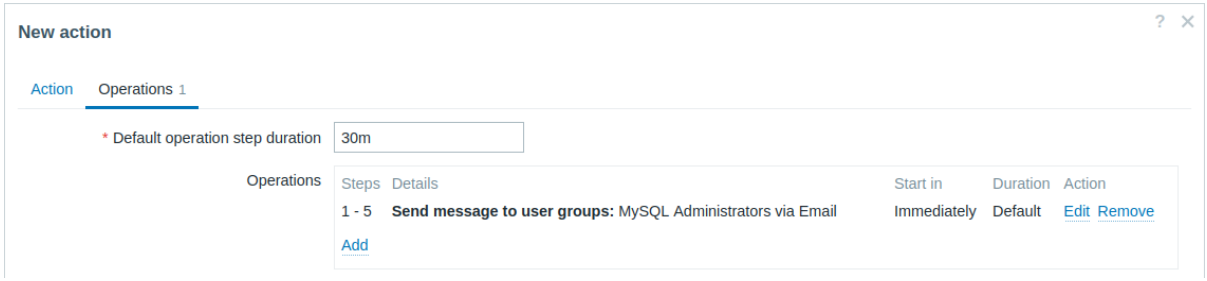
Sytuacja	Zachowanie
Podczas trwającej eskalacji (na przykład podczas wysyłania wiadomości) akcja zostaje usunięta	Nie są już wysyłane żadne kolejne wiadomości. Informacja ta jest zapisywana w pliku dziennika serwera (od Debug Level 3=Warning), na przykład: `escalation canceled: action id:334 deleted`

Przykłady eskalacji

Przykład 1

Wysyłanie powtarzanego powiadomienia co 30 minut (łącznie 5 razy) do grupy „MySQL Administrators". Aby to skonfigurować:

- Na karcie *Operations* ustaw *Default operation step duration* na „30m" (30 minut).
- Ustaw *Steps* eskalacji od „1" do „5".
- Wybierz grupę „MySQL Administrators" jako odbiorców wiadomości.



Powiadomienia będą wysyłane po 0:00, 0:30, 1:00, 1:30 i 2:00 godziny od wystąpienia problemu (chyba że problem zostanie oczywiście rozwiązany wcześniej).

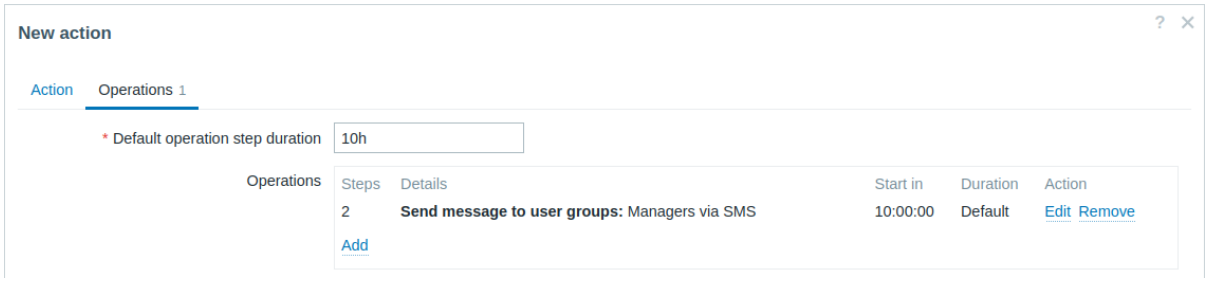
Jeśli problem zostanie rozwiązany i skonfigurowano wiadomość o odzyskaniu, zostanie ona wysłana do tych, którzy otrzymali co najmniej jedną wiadomość o problemie w ramach tego scenariusza eskalacji.

Note:
Jeśli wyzwalacz, który wygenerował aktywną eskalację, zostanie wyłączony, Zabbix wyśle wszystkim osobom, które otrzymały już powiadomienia, wiadomość informacyjną na ten temat.

Przykład 2

Wysyłanie opóźnionego powiadomienia o długotrwałym problemie. Aby to skonfigurować:

- Na karcie *Operations* ustaw *Default operation step duration* na „10h" (10 godzin).
- Ustaw *Steps* eskalacji od „2" do „2".



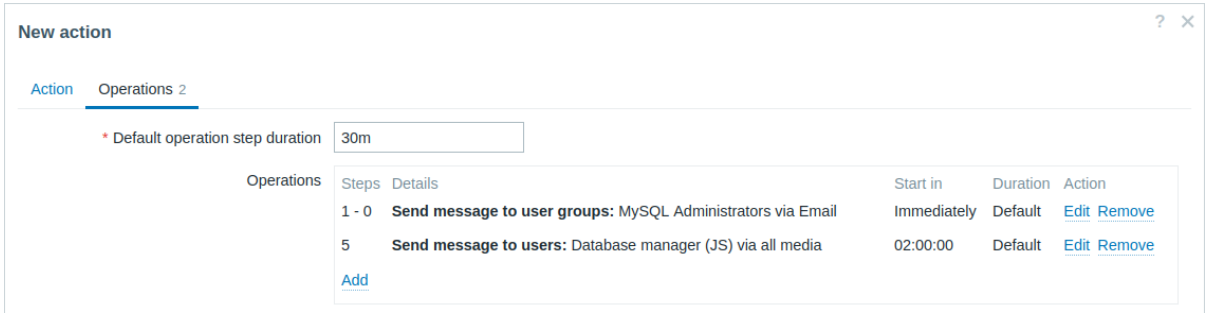
Powiadomienie zostanie wysłane dopiero w kroku 2 scenariusza eskalacji, czyli 10 godzin po wystąpieniu problemu.

Możesz dostosować treść wiadomości do czegoś w rodzaju „Problem trwa dłużej niż 10 godzin".

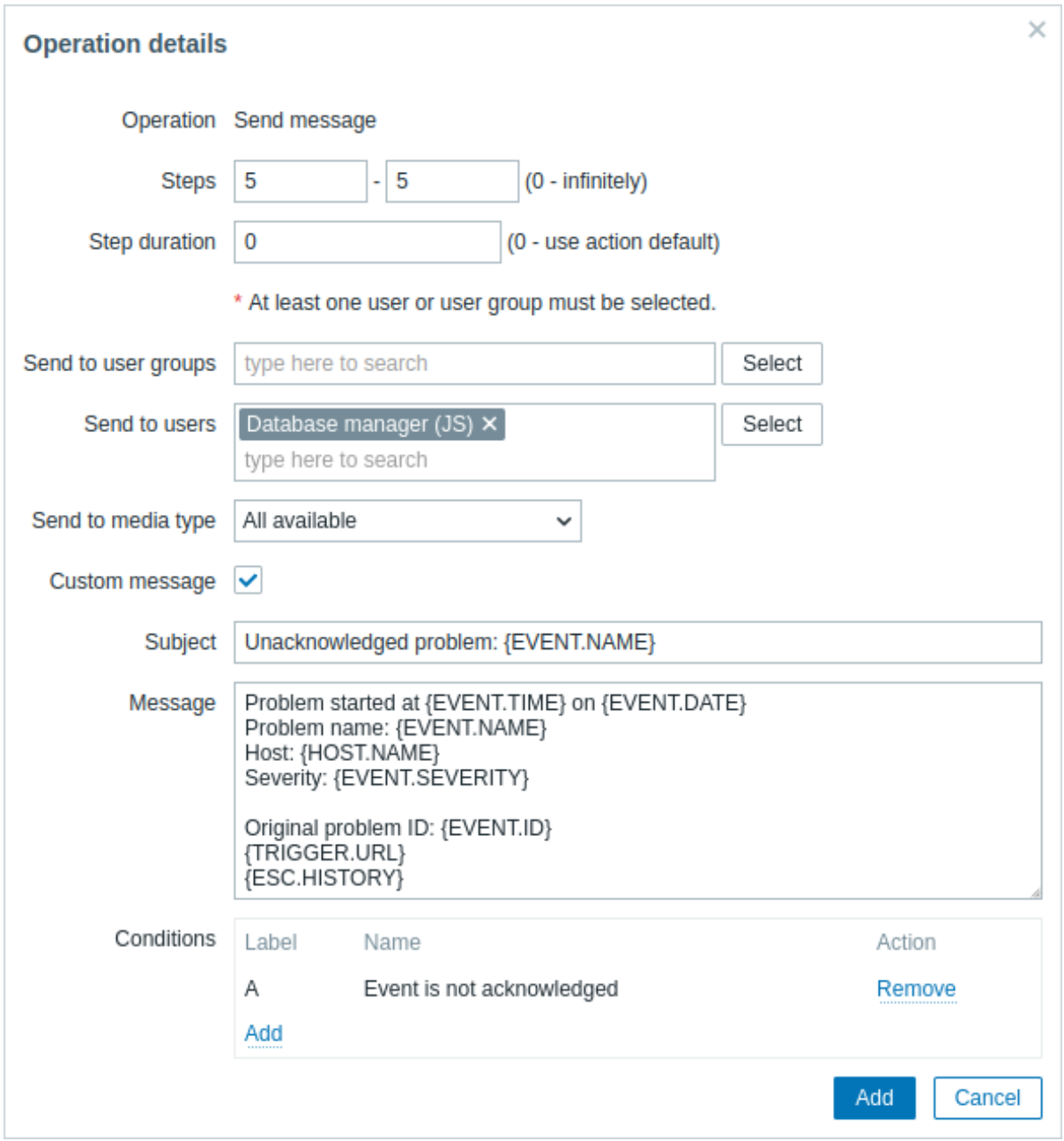
Przykład 3

Eskalacja problemu do Szefa.

W pierwszym przykładzie powyżej skonfigurowaliśmy okresowe wysyłanie wiadomości do administratorów MySQL. W tym przypadku administratorzy otrzymają cztery wiadomości, zanim problem zostanie eskalowany do menedżera bazy danych. Należy pamiętać, że menedżer otrzyma wiadomość tylko wtedy, gdy problem nie został jeszcze potwierdzony, co oznacza, że prawdopodobnie nikt jeszcze nad nim nie pracuje.



Szczegóły operacji 2:



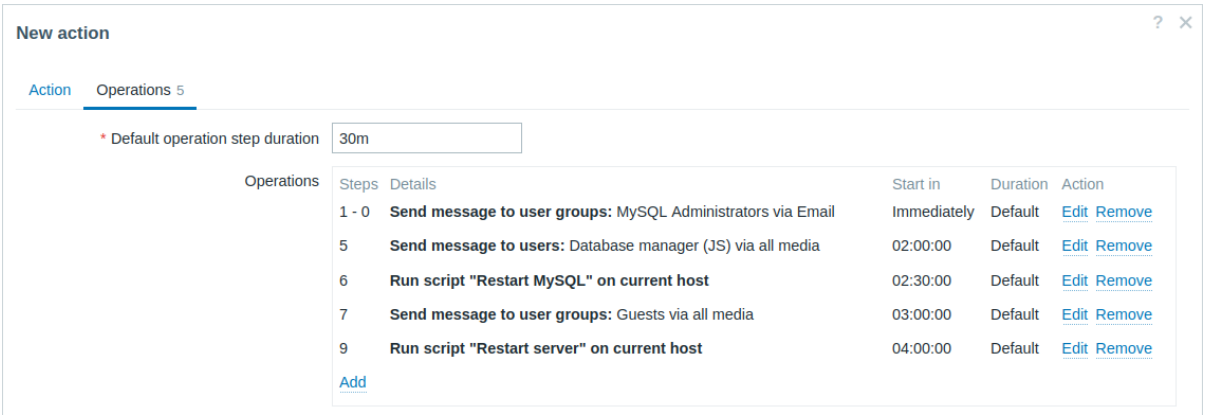
Zwróć uwagę na użycie makra {ESC.HISTORY} w dostosowanej wiadomości. Makro to będzie zawierać informacje o wszystkich wcześniej wykonanych krokach tej eskalacji, takich jak wysłane powiadomienia i wykonane polecenia.

Przykład 4

Bardziej złożony scenariusz. Po wysłaniu wielu wiadomości do administratorów MySQL i eskalacji do menedżera, Zabbix spróbuje ponownie uruchomić bazę danych MySQL. Stanie się to, jeśli problem będzie występował przez 2:30 godziny i nie zostanie potwierdzony.

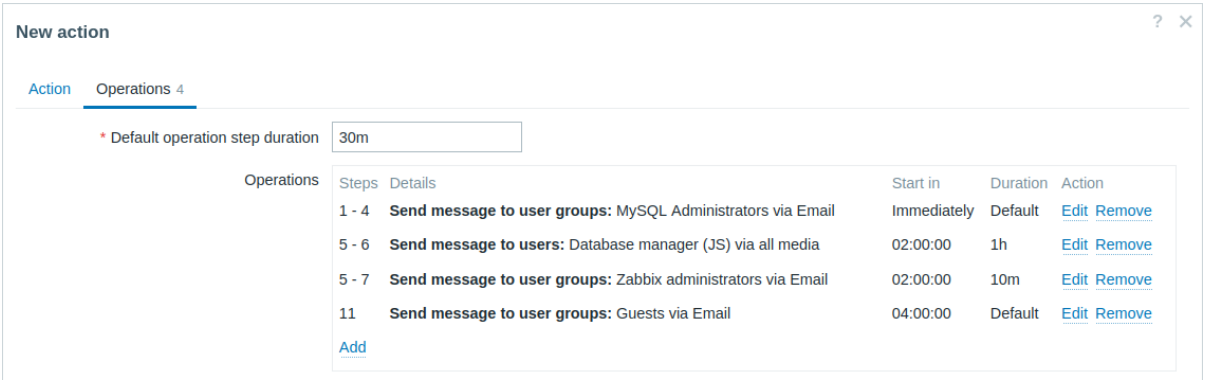
Jeśli problem nadal będzie występował, po kolejnych 30 minutach Zabbix wyśle wiadomość do wszystkich użytkowników gościnnych.

Jeśli to nie pomoże, po kolejnej godzinie Zabbix zrestartuje serwer z bazą danych MySQL (drugie zdalne polecenie) przy użyciu poleceń IPMI.



Przykład 5

Eskalacja z kilkoma operacjami, które mają nakładające się zakresy kroków i niestandardowe interwały. Domyślny czas trwania kroku operacji wynosi 30 minut.



Powiadomienia będą wysyłane w następujący sposób:

- Do administratorów MySQL o 0:00, 0:30, 1:00 i 1:30 od momentu wystąpienia problemu.
- Do menedżera bazy danych o 2:00 i 2:10 (krótszy niestandardowy czas trwania kroku wynoszący 10 minut, zdefiniowany w kolejnej operacji, zastępuje dłuższy czas trwania kroku wynoszący 1 godzinę skonfigurowany tutaj, zgodnie z opisem w Szczegóły operacji dla *Czas trwania kroku*, gdy kroki nakładają się).
- Do administratorów Zabbix o 2:00, 2:10 i 2:20 od momentu wystąpienia problemu (zastosowany jest niestandardowy czas trwania kroku wynoszący 10 minut).
- Do użytkowników guest o 4:00 od momentu wystąpienia problemu (domyślny czas trwania kroku wynoszący 30 minut obowiązuje między krokami 8 i 11).

3 Otrzymywanie powiadomień o nieobsługiwanych pozycjach

Przegląd

Możliwe jest otrzymywanie powiadomień o nieobsługiwanych pozycjach w Zabbix.

Jest to część koncepcji zdarzeń wewnętrznych w Zabbix, umożliwiającej użytkownikom otrzymywanie powiadomień w takich sytuacjach. Zdarzenia wewnętrzne odzwierciedlają zmianę stanu:

- gdy pozycje przechodzą ze stanu „normalny” do „nieobsługiwany” (i z powrotem);
- gdy wyzwalacze przechodzą ze stanu „normalny” do „nieznany” (i z powrotem);
- gdy reguły wykrywania niskiego poziomu przechodzą ze stanu „normalny” do „nieobsługiwany” (i z powrotem).

Ta sekcja przedstawia instrukcję, jak **otrzymywać powiadomienia**, gdy pozycja stanie się nieobsługiwana.

Konfiguracja

Ogólnie proces konfigurowania powiadomienia powinien być znajomy dla osób, które wcześniej konfigurowały alerty w Zabbix.

Krok 1

Skonfiguruj wybrane medium, takie jak e-mail, SMS lub skrypt, które będzie używane do powiadomień. Aby wykonać to zadanie, zapoznaj się z odpowiednimi sekcjami instrukcji.

Attention:
Do powiadamiania o zdarzeniach wewnętrznych używana jest domyślna ważność („Niesklasyfikowane”), więc pozostaw ją zaznaczoną podczas konfigurowania mediów użytkownika, jeśli chcesz otrzymywać powiadomienia o zdarzeniach wewnętrznych.

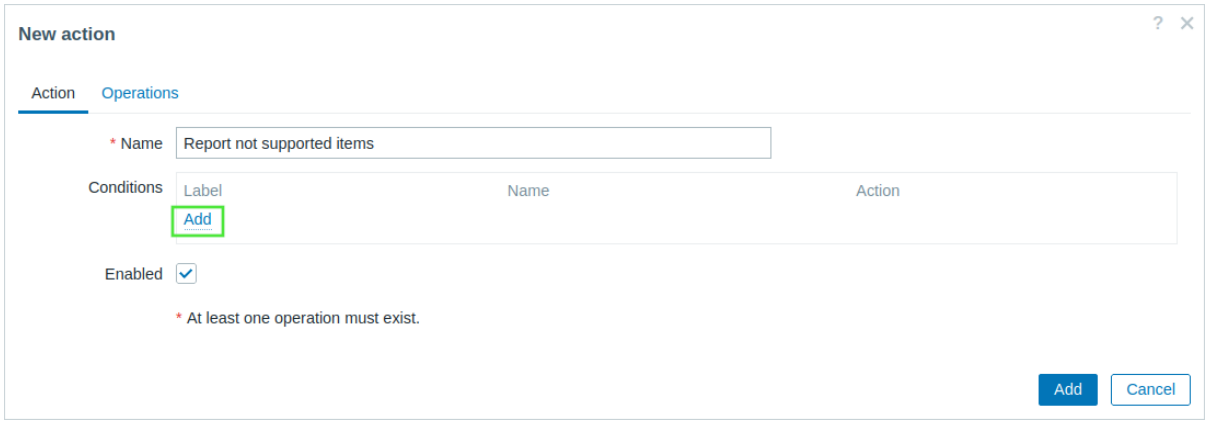
Krok 2

Przejdź do *Alerty* → *Akcje* → *Akcje wewnętrzne*.

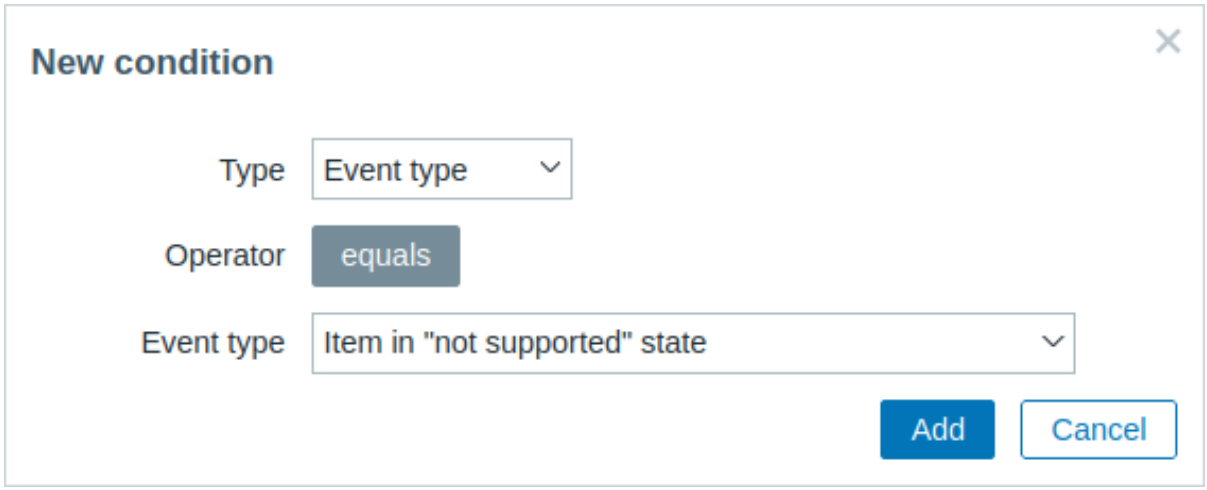
Kliknij *Utwórz akcję* w prawym górnym rogu strony, aby otworzyć formularz konfiguracji akcji.

Krok 3

Na karcie *Action* wprowadź nazwę akcji. Następnie kliknij *Add* w bloku *Conditions*, aby dodać nowy warunek.



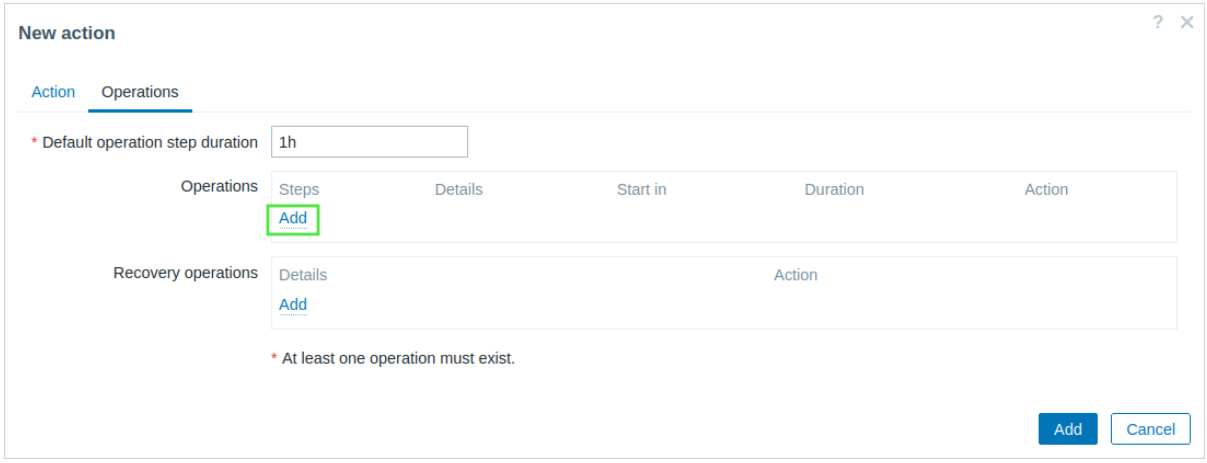
W wyskakującym oknie *New condition* wybierz „Event type” jako typ warunku, a następnie wybierz „Item in ’not supported’ state” jako typ zdarzenia.



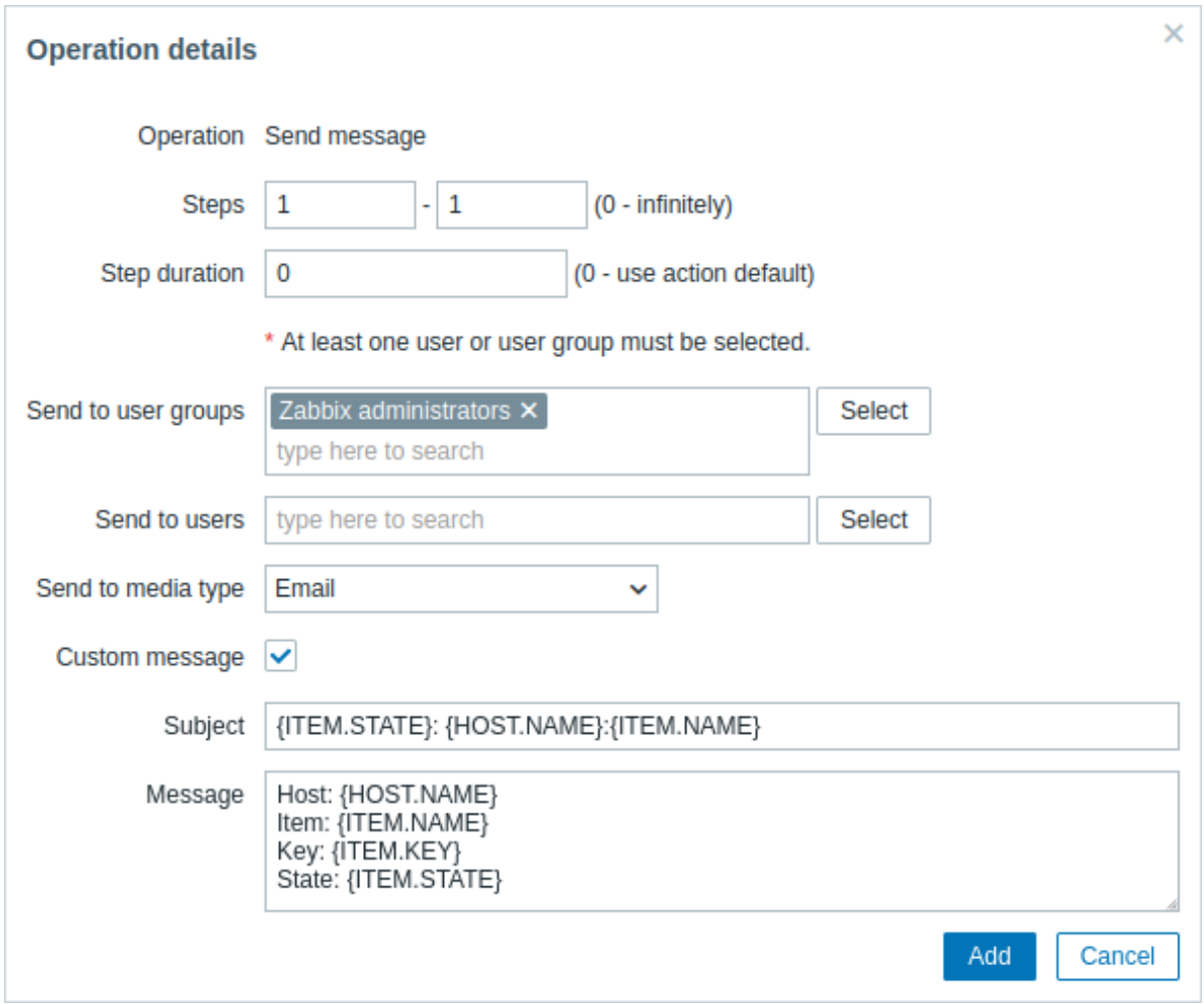
Nie zapomnij kliknąć *Add*, aby warunek został faktycznie dodany do listy w bloku *Conditions*.

Krok 4

W zakładce *Operacje* kliknij *Dodaj* w bloku *Operacje*, aby dodać nową operację.



Wybierz odbiorców wiadomości (grupy użytkowników/użytkowników) oraz typ mediów (lub „Wszystkie”), który ma zostać użyty do dostarczenia wiadomości. Zaznacz pole wyboru *Własna wiadomość*, jeśli chcesz wprowadzić własny temat/treść wiadomości o problemie.

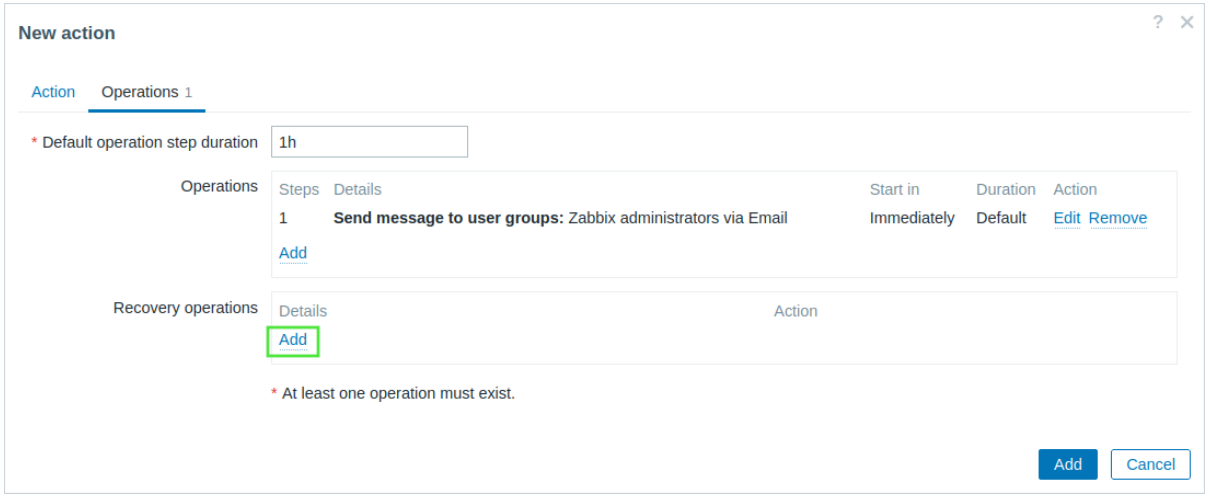


Kliknij *Dodaj*, aby faktycznie wyświetlić operację w bloku *Operacje*.

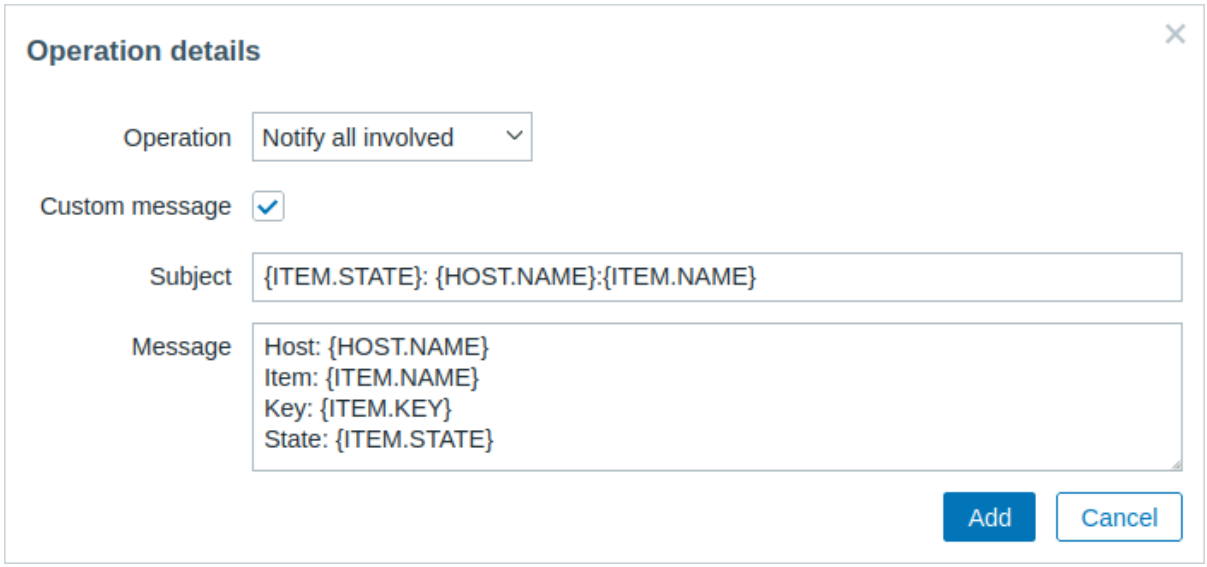
Jeśli chcesz otrzymać więcej niż jedno powiadomienie, ustaw czas trwania kroku operacji (odstęp między wysyłanymi wiadomościami) i dodaj kolejny krok.

Krok 5

Blok *Operacje odzyskiwania* umożliwia skonfigurowanie powiadomienia o odzyskaniu, gdy pozycja wróci do normalnego stanu. Kliknij *Dodaj* w bloku *Operacje odzyskiwania*, aby dodać nową operację odzyskiwania.



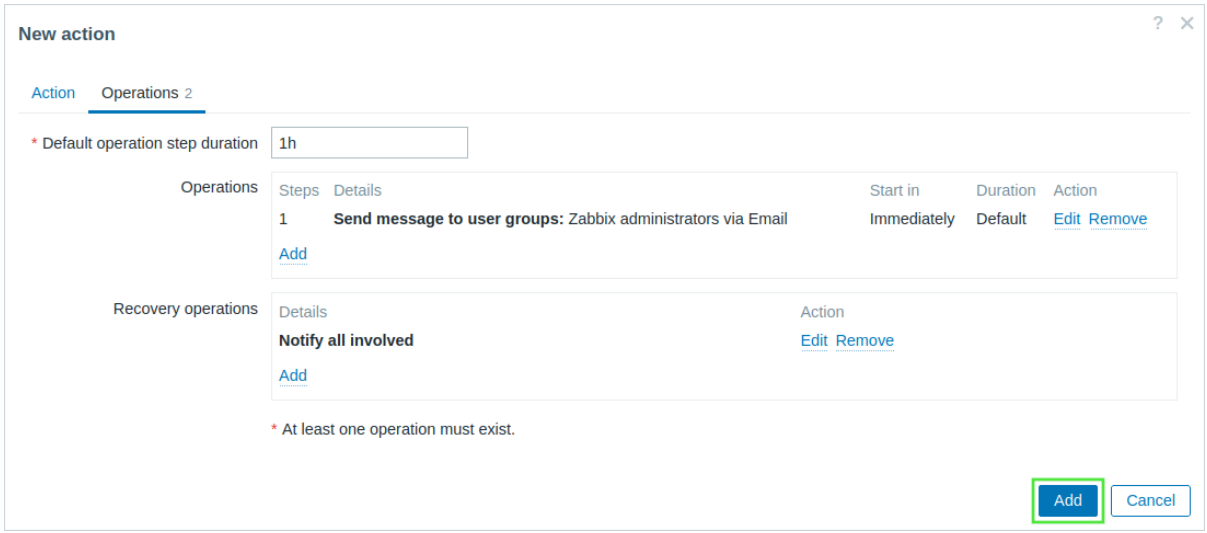
Wybierz typ operacji „Powiadom wszystkich zaangażowanych". Zaznacz pole wyboru *Własna wiadomość*, jeśli chcesz wprowadzić niestandardowy temat/treść wiadomości odzyskiwania.



Kliknij *Dodaj* w wyskakującym oknie *Szczegóły operacji*, aby faktycznie dodać operację do listy w bloku *Operacje odzyskiwania*.

Krok 6

Po zakończeniu kliknij przycisk *Dodaj* u dołu formularza.



I to wszystko, gotowe! Teraz możesz oczekiwać pierwszego powiadomienia z Zabbix, jeśli jakaś pozycja stanie się nieobsługiwana.

11 Makra

Przegląd

Zabbix obsługuje szereg wbudowanych makr, które mogą być używane w różnych sytuacjach. Makra te są zmiennymi identyfikowanymi przez określoną składnię:

`{MACRO}`

Makra są rozwijane do określonej wartości w zależności od kontekstu.

Efektywne użycie makr pozwala zaoszczędzić czas i uczynić konfigurację Zabbix bardziej przejrzystą.

W jednym z typowych zastosowań makro może być użyte w szablonie. W ten sposób wyzwalacz w szablonie może mieć nazwę „Obciążenie procesora jest zbyt wysokie na {HOST.NAME}". Gdy szablon zostanie zastosowany do hosta, na przykład Zabbix server, nazwa zostanie rozwinięta do „Obciążenie procesora jest zbyt wysokie na Zabbix server", gdy wyzwalacz będzie wyświetlany w sekcji Monitorowanie.

Makra mogą być używane w parametrach klucza pozycji. Makro może być użyte tylko dla części parametru, na przykład `item.key[server_{HOST.HOST}_local]`. Ujęcie parametru w podwójny cudzysłów nie jest konieczne, ponieważ Zabbix zadba o wszelkie niejednoznaczne znaki specjalne, jeśli wystąpią w rozwiniętym makrze.

W Zabbix istnieją także inne typy makr.

Zabbix obsługuje następujące makra:

- `{MACRO}` - wbudowane makro (zobacz pełną listę)
- `{<macro>.<func>(<params>)}` - funkcje makr
- `{$MACRO}` - makro zdefiniowane przez użytkownika, opcjonalnie z kontekstem
- `{#MACRO}` - makro dla odkrywania niskiego poziomu
- `{?EXPRESSION}` - makro wyrażenia

1 Funkcje makr

Przegląd

Funkcje makr umożliwiają dostosowanie wartości makr (na przykład skrócenie ich lub wyodrębnienie określonych podciągów), dzięki czemu łatwiej się z nimi pracuje.

Składnia funkcji makra jest następująca:

```
{macro.func(params)}
```

gdzie

- **macro** - makro do dostosowania;
- **func** - funkcja do zastosowania (zobacz obsługiwane funkcje);
- **params** - lista parametrów funkcji rozdzielonych przecinkami, które muszą być ujęte w **podwójne cudzysłowy**, jeśli:
  - zaczynają się od spacji lub podwójnego cudzysłowu;
  - zawierają zamykający nawias lub przecinek.

Na przykład:

```
{{TIME}.fmttime(format,time_shift)}
{{ITEM.VALUE}.regsub(pattern, output)}
{{$USERMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.regsub(pattern, output)}
```

Funkcje makr są obsługiwane dla

- Makr wbudowanych
- Makr użytkownika
- Makr wykrywania niskiego poziomu
- Makr wyrażeń

Funkcje makr mogą być używane we wszystkich miejscach obsługujących wymienione makra. Dotyczy to wszystkich przypadków, chyba że wyraźnie zaznaczono, że oczekiwane jest tylko makro (na przykład podczas konfigurowania makr hosta lub filtrów reguły wykrywania niskiego poziomu).

Obsługiwana jest pojedyncza funkcja na makro; wiele funkcji makr połączonych w łańcuch nie jest obsługiwanych.

Note:
Zobacz przykłady znaków ucieczki dla przypadków, gdy funkcje makr są używane w innych kontekstach (funkcja, klucz pozycji, inne makro itd.).

Obsługiwane funkcje

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
btoa	Kodowanie wartości makra do formatu Base64.
fmtnum	Formatowanie liczby w celu kontrolowania liczby cyfr wyświetlanych po przecinku.
fmttime	Formatowanie czasu.
htmldecode	Dekodowanie wartości makra z kodowania HTML.
htmlencode	Kodowanie wartości makra do kodowania HTML.
iregsub	Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (bez rozróżniania wielkości liter).
lowercase	Przekształcanie znaków wartości makra na małe litery.
regrepl	Zastępowanie znaku/podciągu w wartości makra.
regsub	Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (z rozróżnianiem wielkości liter).
tr	Transliteracja znaków wartości makra.
uppercase	Przekształcanie znaków wartości makra na wielkie litery.
urldecode	Dekodowanie wartości makra z kodowania URL.
urlencode	Kodowanie wartości makra do kodowania URL.

Szczegóły funkcji

Opcjonalne parametry funkcji są oznaczone przez < >.

btoa

Kodowanie wartości makra do kodowania Base64. Kodowanie Base64 to metoda przedstawiania danych binarnych jako tekstu, przydatna do przechowywania i bezpiecznego przesyłania zawartości binarnej za pośrednictwem protokołów tekstowych.

Przykład:

```
{{ITEM.VALUE}.btoa()} - zakoduje wartość taką jak "zabbix" do postaci Base64 "emFiYml4"
```

fmtnum(digits)

Formatowanie liczb umożliwiające kontrolę liczby cyfr wyświetlanych po przecinku.

Parametry:

- **digits** — liczba cyfr po przecinku. Prawidłowy zakres: 0-20. Końcowe zera nie będą dodawane.

Przykłady:

```
{{ITEM.VALUE}.fmtnum(2)} - zwróci "24.35" dla otrzymanej wartości "24.3483523"
{{ITEM.VALUE}.fmtnum(0)} - zwróci "24" dla otrzymanej wartości "24.3483523"
```

fmttime(format,<time_shift>)

Formatowanie czasu.
 Należy pamiętać, że tej funkcji można używać z makrami, które są rozwijane do wartości w jednym z następujących formatów czasu:

- `hh:mm:ss`
- `yyyy-mm-ddThh:mm:ss[tz]` (standard ISO8601)
- znacznik czasu UNIX

Parametry:

- **format** - obowiązkowy ciąg formatu, zgodny z formatowaniem funkcji `strftime`;
- **time_shift** (opcjonalnie) - przesunięcie czasu stosowane do czasu przed formatowaniem; powinno zaczynać się od `-<N><time_unit>` lub `+<N><time_unit>`, gdzie:
  - `N` - liczba jednostek czasu do dodania lub odjęcia;
  - `time_unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok).

Uwagi:

- Parametr `time_shift` obsługuje wieloetapowe operacje na czasie i może zawierać `/<time_unit>` do przesunięcia na początek jednostki czasu (`/d` - północ, `/w` - 1. dzień tygodnia (poniedziałek), `/M` - 1. dzień miesiąca itd.). Przykłady: `-1w` - dokładnie 7 dni wstecz; `-1w/w` - poniedziałek poprzedniego tygodnia; `-1w/w+1d` - wtorek poprzedniego tygodnia.
- Operacje na czasie są obliczane od lewej do prawej bez priorytetów. Na przykład `-1M/d+1h/w` zostanie zinterpretowane jako `((-1M/d)+1h)/w`.

Przykłady:

```
{{TIMESTAMP}.fmttime(%B)} - zwróci "October" dla otrzymanej wartości "1633098961"
{{TIMESTAMP}.fmttime(%d %B,-1M/M)} - zwróci "1 September" dla otrzymanej wartości "1633098961"
```

htmldecode

Dekodowanie wartości makra z kodowania HTML.

Obsługiwane są następujące znaki:

Wartość	Zdekodowana wartość
&	&
<	<
>	>
"	"
'	'
'	'

Przykład:

```
{{ITEM.VALUE}.htmldecode()} - zdekoduje HTML wartość taką jak "&lt;" do "<"
```

htmlencode

Kodowanie wartości makra do kodowania HTML.

Obsługiwane są następujące znaki:

Wartość	Zakodowana wartość
&	&
<	<
>	>
"	"
'	'

Przykład:

```
{{ITEM.VALUE}.htmlencode()} - zakoduje znak HTML, taki jak "<", do postaci "&lt;"
```

iregsub(pattern,output)

Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (bez rozróżniania wielkości liter).

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **output** - opcje wyjściowe. Symbole zastępcze **\1 - \9** są obsługiwane dla grup przechwytywania. **\0** zwraca dopasowany tekst.

Uwagi:

- Jeśli nie ma dopasowania dla wyrażenia regularnego, funkcja zwraca pusty ciąg.
- Jeśli wzorzec funkcji jest niepoprawnym wyrażeniem regularnym, makro jest obliczane jako 'UNKNOWN' (z wyjątkiem makr wykrywania niskiego poziomu, w których przypadku funkcja zostanie zignorowana, a makro pozostanie nierozwiązane).
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępczym są zastępowane pustym ciągiem.

Przykład:

```
{{ITEM.VALUE}.iregsub("fail|error|fault|problem","ERROR")} - zostanie rozwinięte do "ERROR", jeśli zostaną
```

małe litery

Przekształcenie wszystkich znaków wartości makra na małe litery.
Działa z jednobajtowymi zestawami znaków (takimi jak ASCII) i nie obsługuje UTF-8.

Przykład:

```
{{ITEM.VALUE}.lowercase()} - przekształci wartość taką jak "Zabbix SERVER" na "zabbix server" (małe litery
```

regrepl(pattern,replacement,<pattern2>,<replacement2>,...)

Zastępowanie znaku/podciągu w wartości makra.

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **replacement** - ciąg zastępujący. Symbole zastępcze **\1 - \9** są obsługiwane w ciągach zastępujących dla grup przechwytywania.

Uwagi:

- Wzorce i ciągi zastępujące są przetwarzane sekwencyjnie, przy czym każda kolejna para jest stosowana zgodnie z wynikiem poprzedniego zastąpienia;
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępującym są zastępowane pustym ciągiem.

Przykłady:

```
{{ITEM.VALUE}.regrepl("oldParam", "newParam")} - zastąpi "oldParam" przez "newParam"
{{ITEM.VALUE}.regrepl("([^a-z])","\\\1")} - wszystkie znaki niebędące literami zostaną poprzedzone ukośnik
{$THRESHOLD:"{{#FSNAME}.regrepl(\"\\$\",\"\")}"} - usunie końcowy ukośnik odwrotny (na przykład, aby zastą
{{ITEM.VALUE}.regrepl("_v1\.0", "_v2.0", "\(final\)", "")} - zastąpi wiele fragmentów w wartości pozycji
```

regsub(pattern,output)

Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (z uwzględnieniem wielkości liter).

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **output** - opcje danych wyjściowych. Symbole zastępcze **\1 - \9** są obsługiwane dla grup przechwytywania. **\0** zwraca dopasowany tekst.

Uwagi:

- Jeśli nie ma dopasowania do wyrażenia regularnego, funkcja zwraca pusty ciąg.
- Jeśli wzorzec funkcji jest niepoprawnym wyrażeniem regularnym, makro przyjmuje wartość 'UNKNOWN' (z wyjątkiem makr wykrywania niskiego poziomu, w których przypadku funkcja zostanie zignorowana, a makro pozostanie nierozwiązane).
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępczym są zastępowane pustym ciągiem.

Przykłady:

```
{{ITEM.VALUE}.regsub("^([0-9]+)", Problem ID: \1)} - zostanie rozwinięte do "Problem ID: 123", jeśli zosta
{{ITEM.VALUE}.regsub("fail|error|fault|problem","ERROR")} - zostanie rozwinięte do "ERROR", jeśli zostaną
```

Zobacz więcej przykładów.

tr(characters,replacement)

Transliteracja znaków wartości makra.

- **characters** - zestaw znaków do zastąpienia;
- **replacement** - zestaw odpowiadających im pozycyjnie znaków zastępczych.

Przykłady:

```
{{ITEM.VALUE}.tr(abc, xyz)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z"
{{ITEM.VALUE}.tr(abc, xyzq)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z" (
{{ITEM.VALUE}.tr(abcde, xyz)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z",
{{ITEM.VALUE}.tr("\\\'", "\/\"")} - zastąpi wszystkie wystąpienia ukośnika odwrotnego ukośnikiem prostym,
{{ITEM.VALUE}.tr(A-Z,a-z)} - przekonwertuje wszystkie litery na małe
{{ITEM.VALUE}.tr(0-9a-z,*)} - zastąpi wszystkie cyfry i małe litery znakiem "*"
{{ITEM.VALUE}.tr(0-9,ab)} - zastąpi wszystkie wystąpienia 0 przez "a", a wszystkie wystąpienia 1, 2, 3, 4,
{{ITEM.VALUE}.tr(0-9abcA-L,*)} - zastąpi wszystkie cyfry, znaki "abc" oraz zakres A-L znakiem "*"
{{ITEM.VALUE}.tr("\n","*")} - zastąpi wystąpienia końca linii znakiem *
{{ITEM.VALUE}.tr("e", "\n")} - zastąpi wszystkie "e" znakiem końca linii
```

Aby uwzględnić znaki literałowe:

```
ukośnik odwrotny - musi być zapisany jako \\
pojedynczy cudzysłów - musi być zapisany jako \'
podwójny cudzysłów - musi być zapisany jako \"
```

Obsługiwane sekwencje ucieczki z ukośnikiem odwrotnym:

```
\\\\ => \\ - podwójny ukośnik odwrotny na pojedynczy ukośnik odwrotny
\\a  => \a - alert
\\b  => \b - backspace
\\f  => \f - wysunięcie strony
\\n  => \n - nowa linia
\\r  => \r - powrót karetki
\\t  => \t - tabulator poziomy
\\v  => \v - tabulator pionowy
```

wielkie litery

Przekształcenie wszystkich znaków wartości makra na wielkie litery. Działa z jednobajtowymi zestawami znaków (takimi jak ASCII) i nie obsługuje UTF-8.

Przykład:

```
{{ITEM.VALUE}.uppercase()} - przekształci wartość taką jak "Zabbix Server" na "ZABBIX SERVER" (wielkie lit
```

urldecode

Dekodowanie wartości makra z kodowania URL.

Przykład:

```
{{ITEM.VALUE}.urldecode()} - zdekoduje URL wartość taką jak "%2F" do "/"
```

urlencode

Kodowanie wartości makra do formatu URL encoding.

Przykład:

```
{{ITEM.VALUE}.urlencode()} - zakoduje znak taki jak "/" do postaci "%2F"
```

Dodatkowe przykłady

Poniższa tabela pokazuje więcej przykładów użycia funkcji makr.

{#IFALIAS} jest makrem LLD i jest zdefiniowane tylko w kontekstach low-level discovery (reguły wykrywania, prototypy oraz pozycje/wyzwalacze utworzone na ich podstawie). Użycie go poza LLD spowoduje pozostawienie tokenu bez rozwinięcia.

Funkcja makra	Otrzymana wartość	Wynik
{{ITEM.VALUE}.regsub(^[0-9]+, Problem)}	123Log line	Problem
{{ITEM.VALUE}.regsub("^([0-9]+)", "Problem")}	123 Log line	Problem
{{ITEM.VALUE}.regsub(".*", "Problem ID: \1")}	Log line	Problem ID:
{{ITEM.VALUE}.regsub("^(\w+).*?([0-9]+)", " Problem ID: \1_\2 ")}	MySQL crashed errno 123	Problem ID: MySQL_123
{{ITEM.VALUE}.regsub("([1-9]+", "Problem ID: \1")}	123 Log line	*UNKNOWN* (nieprawidłowe wyrażenie regularne)
{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}	customername_1	customername
{{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)}	customername_11	
{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)}	customername_1	{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)} (nieprawidłowe wyrażenie regularne)
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}"}	customername_1	{$MACRO:"customername"}

Funkcja makra	Otrzymana wartość	Wynik
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \2)}"}	customername_1	{$MACRO:"1"}
{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \1)}"}	customername_1	{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+\", \1)}"} (nieprawidłowe wyrażenie regularne)
"{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}\"}"	customername_1	"{$MACRO:"\customername\"}"
"{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)}\"}"	customername_1	"{$MACRO:"\1\"}"
"{$MACRO:\"{{#IFALIAS}.regsub(\\"(.*)_([0-9]+\\", \1)}\"}"	customername_1	"{$MACRO:\"{{#IFALIAS}.regsub(\\"(.*)_([0-9]+\\", \1)}\"}" (nieprawidłowe wyrażenie regularne)

Wyświetlanie pełnych wartości pozycji

Długie wartości rozwiniętych makr {ITEM.VALUE} i {ITEM.LASTVALUE} dla pozycji tekstowych/logów są w niektórych miejscach frontend skracane do 20 znaków. Aby zobaczyć pełne wartości tych makr, można użyć funkcji makr, np.:

```
{{ITEM.VALUE}.regsub("(.*)", \1)}
{{ITEM.LASTVALUE}.regsub("(.*)", \1)}
```

Zobacz także: szczegóły makr {ITEM.VALUE} i {ITEM.LASTVALUE}.

2 Makra użytkownika

Przegląd

Makra użytkownika są obsługiwane w Zabbix w celu zapewnienia większej elastyczności, oprócz makr obsługiwanych domyślnie.

Makra użytkownika można definiować na poziomie globalnym, szablonu i hosta. Makra te mają specjalną składnię:

{$MACRO}

Zabbix rozwiązuje makra zgodnie z następującym priorytetem:

1. makra na poziomie hosta (sprawdzane jako pierwsze)
2. makra zdefiniowane dla szablonów pierwszego poziomu hosta (tj. szablonów połączonych bezpośrednio z hostem), posortowane według ID szablonu
3. makra zdefiniowane dla szablonów drugiego poziomu hosta, posortowane według ID szablonu
4. makra zdefiniowane dla szablonów trzeciego poziomu hosta, posortowane według ID szablonu itd.
5. makra globalne (sprawdzane jako ostatnie)

Innymi słowy, jeśli makro nie istnieje dla hosta, Zabbix spróbuje znaleźć je w szablonach hosta o rosnącej głębokości. Jeśli nadal nie zostanie znalezione, zostanie użyte makro globalne, jeśli istnieje.

Warning:
Jeśli makro o **tej samej nazwie** istnieje w wielu podłączonych szablonach na tym samym poziomie, zostanie użyte makro z szablonu o najniższym ID. Oznacza to, że posiadanie makr o tej samej nazwie w wielu szablonach stanowi ryzyko konfiguracyjne.

Jeśli Zabbix nie będzie w stanie znaleźć makra, makro nie zostanie rozwiązane.

Attention:
Makra (w tym makra użytkownika) celowo pozostają nierozwiązane w sekcji Konfiguracja (na przykład na liście wyzwalaczy), aby zwiększyć przejrzystość złożonej konfiguracji.

Makra użytkownika mogą być używane w:

- nazwie pozycji
- parametrze klucza pozycji
- interwałach aktualizacji pozycji i interwałach elastycznych
- nazwie i opisie wyzwalacza
- parametrach wyrażenia wyzwalacza i stałych (zobacz przykłady)
- wielu innych miejscach — zobacz pełną listę

Typowe przypadki użycia makr globalnych i makr hosta

- użycie makra globalnego w kilku miejscach; następnie zmiana wartości makra i zastosowanie zmian konfiguracji we wszystkich miejscach jednym kliknięciem
- wykorzystanie szablonów z atrybutami specyficznymi dla hosta: hasła, numery portów, nazwy plików, wyrażenia regularne itp.

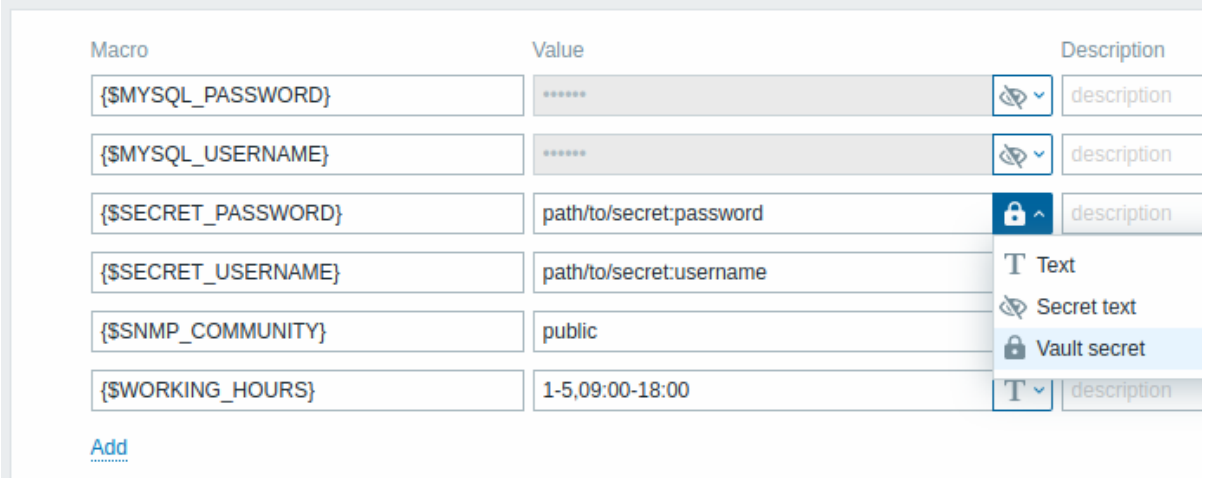
Note:
Zaleca się używanie makr hosta zamiast makr globalnych, ponieważ dodawanie, aktualizowanie lub usuwanie makr globalnych wymusza przyrostową aktualizację konfiguracji dla wszystkich hostów. Więcej informacji można znaleźć w sekcji Passive and active agent checks.

Konfiguracja

Aby zdefiniować makra użytkownika, przejdź do odpowiedniego miejsca w frontend:

- dla makr globalnych przejdź do *Administration* → *Macros*
- dla makr na poziomie hosta i szablonu otwórz właściwości hosta lub szablonu i odszukaj kartę *Macros*

Makro użytkownika ma następujące atrybuty:



Parametr	Opis
Macro	Nazwa makra. Nazwa musi być ujęta w nawiasy klamrowe i zaczynać się od znaku dolara. Przykład: {$FRONTEND_URL}. W nazwach makr dozwolone są następujące znaki: **A-Z** (tylko wielkie litery), **0-9**, _, **.**
Value	Wartość makra. Obsługiwane są trzy typy wartości: **Text** (domyślnie) — wartość w postaci zwykłego tekstu **Secret text** — wartość jest maskowana gwiazdkami **Vault secret** — wartość zawiera ścieżkę/zapytanie do vault secret. Aby zmienić typ wartości, kliknij przycisk na końcu pola wprowadzania wartości. Maksymalna długość wartości makra użytkownika wynosi 2048 znaków.
Description	Pole tekstowe używane do podania dodatkowych informacji o tym makrze.

Podczas konfigurowania makr użytkownika **zwróć uwagę** na następujące zachowania zależne od kontekstu:

- jeśli makra użytkownika są używane w pozycjach lub wyzwalaczach szablonu, rozważ dodanie tych makr również do szablonu (nawet jeśli są zdefiniowane globalnie); dzięki temu makra typu *Text* będą działać zgodnie z oczekiwaniami po wyeksportowaniu szablonu do XML i zaimportowaniu go do innego systemu (wartości tajnych makr nie są eksportowane)
- jeśli makra użytkownika są używane w wyrażeniach wyzwalaczy, te makra zostaną rozwinięte tylko wtedy, gdy odwołują się do parametru lub stałej; NIE zostaną rozwinięte, jeśli odwołują się do hosta, klucza pozycji, funkcji, operatora lub innego wyrażenia wyzwalacza (tajne makra nie mogą być używane w wyrażeniach wyzwalaczy)
- jeśli makra użytkownika są używane na hoście, który ma regułę wykrywania niskopoziomowego z prototypami hostów, wykryte hosty odziedziczą wszystkie makra użytkownika zdefiniowane na tym hoście

Przykłady

Przykład 1

Użycie makra na poziomie hosta w kluczu pozycji „Status demona SSH”:

`net.tcp.service[ssh,,{$SSH_PORT}]`

Ta pozycja może zostać przypisana do wielu hostów, pod warunkiem że wartość **{$SSH_PORT}** jest zdefiniowana na tych hostach.

Przykład 2

Użycie makra na poziomie hosta w wyzwalaczu „Obciążenie CPU jest zbyt wysokie”:

`last(/ca_001/system.cpu.load[,avg1])>{$MAX_CPULOAD}`

Taki wyzwalacz zostałby utworzony w szablonie, a nie edytowany na poszczególnych hostach.

Note:
Jeśli chcesz użyć liczby wartości jako parametru funkcji (na przykład **max(/host/key,#3)**), uwzględnij znak hash w definicji makra w ten sposób: SOME_PERIOD => #3

Przykład 3

Użycie dwóch makr w wyzwalaczu „Obciążenie CPU jest zbyt wysokie”:

`min(/ca_001/system.cpu.load[,avg1],{$CPULOAD_PERIOD})>{$MAX_CPULOAD}`

Zwróć uwagę, że makro może być użyte jako parametr funkcji wyzwalacza, w tym przykładzie funkcji **min()**.

Przykład 4

Zsynchronizuj warunek niedostępności agenta z interwałem aktualizacji pozycji:

- zdefiniuj makro {$INTERVAL} i użyj go w interwale aktualizacji pozycji;
- użyj {$INTERVAL} jako parametru wyzwalacza niedostępności agenta:

`nodata(/ca_001/agent.ping,{$INTERVAL})=1`

Przykład 5

Scentralizuj konfigurację godzin pracy:

- utwórz globalne makro {$WORKING_HOURS} o wartości 1-5,09:00-18:00;
- użyj go w polu *Working time* w *Administration → General → GUI*;
- użyj go w polu *When active* w *Users → Users*, karta *Media* użytkownika;
- użyj go do skonfigurowania częstszego odpytywania pozycji w godzinach pracy:

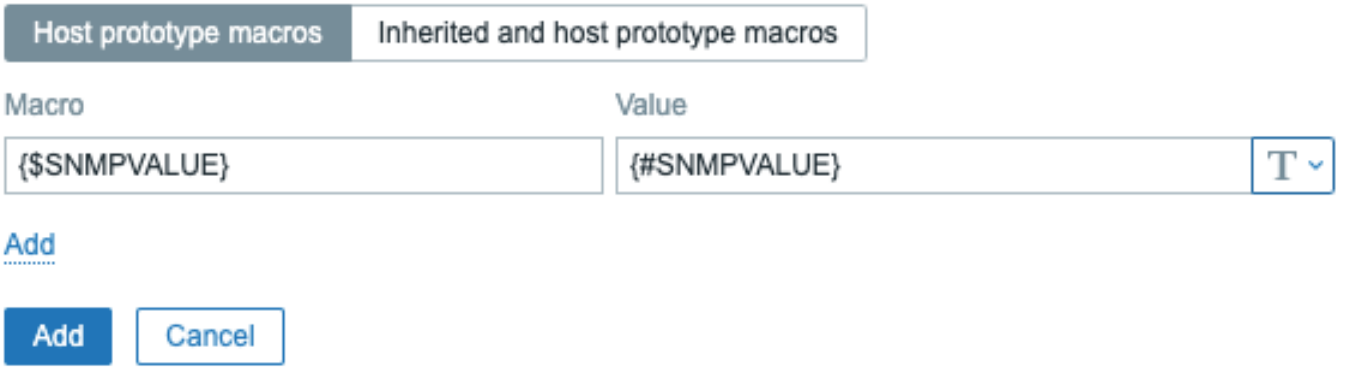


- użyj go w warunku akcji *Time period*;
- w razie potrzeby dostosuj godziny pracy w *Administration → Macros*.

Przykład 6

Użyj makra prototypu hosta do skonfigurowania pozycji dla wykrytych hostów:

- w prototypie hosta zdefiniuj makro użytkownika {$SNMPVALUE} z makrem low-level discovery {#SNMPVALUE} jako wartością:



- przypisz szablon *Generic SNMPv2* do prototypu hosta;
- użyj {$SNMPVALUE} w polu *SNMP OID* pozycji szablonu *Generic SNMPv2*.

Kontekst makra użytkownika

Zobacz makra użytkownika z kontekstem.

3 Makra użytkownika z kontekstem

Przegląd

Opcjonalny kontekst może być używany w makrach użytkownika, umożliwiając zastąpienie wartości domyślnej wartością specyficzną dla danego kontekstu.

Kontekst jest dołączany do nazwy makra; składnia zależy od tego, czy kontekst jest statyczną wartością tekstową:

```
{$MACRO:"tekst statyczny"}
```

czy wyrażeniem regularnym:

```
{$MACRO:regex:"wyrażenie regularne"}
```

Należy pamiętać, że makro z kontekstem wyrażenia regularnego może być zdefiniowane tylko w konfiguracji makr użytkownika. Jeśli prefiks `regex:` zostanie użyty gdzie indziej jako kontekst makra użytkownika, na przykład w wyrażeniu wyzwalacza, zostanie potraktowany jako kontekst statyczny.

Ujmowanie kontekstu w cudzysłów jest opcjonalne (zobacz także ważne uwagi).

Przykłady kontekstu makra:

Przykład	Opis
{$LOW_SPACE_LIMIT}	Makro użytkownika bez kontekstu.
{$LOW_SPACE_LIMIT:/tmp}	Makro użytkownika z kontekstem (ciąg statyczny).
{$LOW_SPACE_LIMIT:regex:"^/tmp$"}	Makro użytkownika z kontekstem (wyrażenie regularne). To samo co {$LOW_SPACE_LIMIT:/tmp}.
{$LOW_SPACE_LIMIT:regex:"^/var/log/.*$"}	Makro użytkownika z kontekstem (wyrażenie regularne). Dopasowuje wszystkie ciągi rozpoczynające się od /var/log/.

Przykłady użycia

Makra użytkownika z kontekstem można definiować w celu uzyskania bardziej elastycznych progów w wyrażeniach wyzwalaczy (na podstawie wartości pobranych przez wykrywanie niskiego poziomu). Na przykład można zdefiniować następujące makra:

- {$LOW_SPACE_LIMIT} = 10
- {$LOW_SPACE_LIMIT:/home} = 20
- {$LOW_SPACE_LIMIT:regex:”^/[a-z]+$”} = 30

Następnie makro wykrywania niskiego poziomu może zostać użyte jako kontekst makra w prototypie wyzwalacza dla wykrywania zamontowanych systemów plików:

```
last(/host/vfs.fs.size[{#FSNAME},pfree])<{$LOW_SPACE_LIMIT:"{#FSNAME}"}
```

Po wykryciu w wyzwalaczach będą stosowane różne progi niskiej ilości wolnego miejsca w zależności od wykrytych punktów montowania lub typów systemów plików. Zdarzenia problemów będą generowane, jeśli:

- katalog /home ma mniej niż 20% wolnego miejsca na dysku
- katalogi pasujące do wzorca regexp (takie jak /etc, /tmp lub /var) mają mniej niż 30% wolnego miejsca na dysku
- katalogi, które nie pasują do wzorca regexp i nie są /home, mają mniej niż 10% wolnego miejsca na dysku

Ważne uwagi

- Jeśli istnieje więcej niż jedna makrodefinicja użytkownika z kontekstem, Zabbix najpierw spróbuje dopasować makra z prostym kontekstem, a następnie makra kontekstowe z wyrażeniami regularnymi w niezdefiniowanej kolejności.

Warning:
Nie twórz różnych makr kontekstowych pasujących do tego samego ciągu, aby uniknąć niezdefiniowanego zachowania.

- Jeśli makro wraz z jego kontekstem nie zostanie znalezione na hoście, w podłączonych szablonach ani globalnie, wyszukiwane jest makro bez kontekstu.

- W kontekście obsługiwane są tylko makra wykrywania niskiego poziomu. Wszelkie inne makra są ignorowane i traktowane jako zwykły tekst.

Technicznie rzecz biorąc, kontekst makra jest określany przy użyciu reguł podobnych do parametrów klucza pozycji, z wyjątkiem tego, że kontekst makra nie jest analizowany jako kilka parametrów, jeśli zawiera znak ,:

- Kontekst makra musi być ujęty w " , jeśli kontekst zawiera znak } lub zaczyna się od znaku ". Cudzysłowy wewnątrz ujętego w cudzysłów kontekstu muszą być poprzedzone znakiem ucieczki \.
- Sam znak \ nie jest znakowany znakiem ucieczki, co oznacza, że niemożliwe jest użycie ujętego w cudzysłów kontekstu kończącego się znakiem \ — makro {$MACRO:"a:\b\c\"} jest nieprawidłowe.
- Początkowe spacje w kontekście są ignorowane, końcowe nie:
  - Na przykład {$MACRO:A} jest takie samo jak {$MACRO: A}, ale nie jak {$MACRO:A }.
- Wszystkie spacje przed początkowym cudzysłowem i po końcowym cudzysłowie są ignorowane, ale wszystkie spacje wewnątrz cudzysłowów nie:
  - Makra {$MACRO:"A"}, {$MACRO: "A"}, {$MACRO:"A" } oraz {$MACRO: "A" } są takie same, ale makra {$MACRO:"A"} i {$MACRO:" A "} nie są.

Wszystkie poniższe makra są równoważne, ponieważ mają ten sam kontekst: {$MACRO:A}, {$MACRO: A} oraz {$MACRO:"A"}. Stanowi to kontrast wobec kluczy pozycji, gdzie 'key[a]', 'key[a]' oraz 'key["a"]' są semantycznie takie same, ale różnią się pod względem unikalności.

4 Tajne makra użytkownika

Przegląd

Zabbix udostępnia dwie opcje ochrony poufnych informacji w wartościach makr użytkownika:

- Tajny tekst
- Sekret z vault

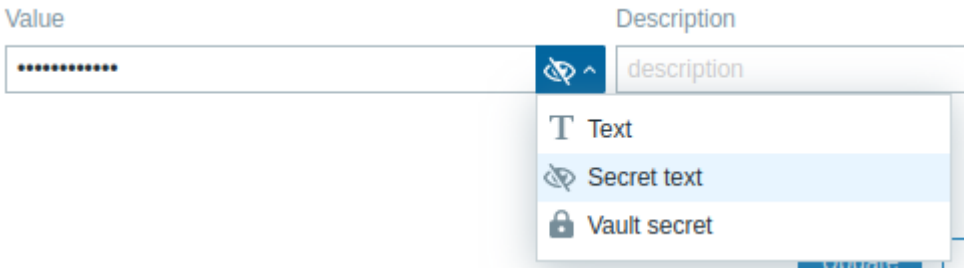
Note:
Chociaż wartość tajnego makra jest ukryta, może zostać ujawniona poprzez użycie w pozycjach. Na przykład w skrypcie zewnętrznym instrukcja echo odwołująca się do tajnego makra może zostać użyta do ujawnienia wartości makra we front-tend, ponieważ serwer Zabbix ma dostęp do rzeczywistej wartości makra. Zobacz lokalizacje, w których wartości tajnych makr są odsłaniane.

Tajnych makr nie można używać w wyrażeniach wyzwalaczy.

Tajny tekst

W przypadku makr typu Tajny tekst wartość makra jest maskowana gwiazdkami.

Aby ustawić wartość makra jako tajną, kliknij przycisk na końcu pola *Value* i wybierz opcję *Secret text*:



Po zapisaniu konfiguracji wyświetlenie wartości nie będzie już możliwe.

Aby zmienić wartość makra, najedź kursorem na pole *Value* i kliknij przycisk *Set new value* (pojawia się po najechaniu kursorem):



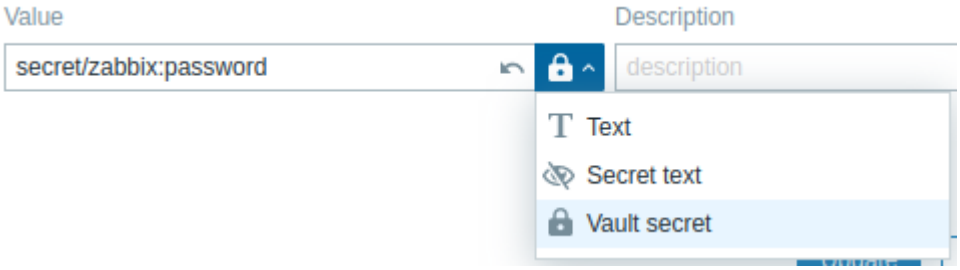
Po kliknięciu przycisku *Set new value* (lub zmianie typu wartości makra) bieżąca wartość zostanie usunięta. Możesz przywrócić oryginalną wartość, klikając strzałkę na końcu pola *Value* (dostępne tylko przed zapisaniem nowej konfiguracji). Pamiętaj, że przywrócenie oryginalnej wartości nie spowoduje jej ujawnienia.

Note:
Adresy URL zawierające tajne makro nie będą działać, ponieważ makro w nich zostanie rozwinięte jako "******".

Sekret Vault

W przypadku makr sekretów Vault wartość makra jest przechowywana w zewnętrznym oprogramowaniu do zarządzania sekretami (vault).

Aby skonfigurować makro sekretu Vault, kliknij przycisk na końcu pola *Value* i wybierz opcję *Vault secret*:



Wartość makra musi wskazywać na sekret w vault. Format wejściowy zależy od dostawcy vault. Przykłady konfiguracji dla poszczególnych dostawców znajdują się tutaj:

- HashiCorp
- CyberArk

Wartości makr sekretów Vault są pobierane z sejfu przez serwer Zabbix (oraz proxy Zabbix, jeśli opcja *Resolve secret vault macros by* jest ustawiona na *Zabbix server and proxy*) przy każdym odświeżeniu danych konfiguracyjnych, a następnie przechowywane w pamięci podręcznej konfiguracji. Serwer Zabbix i proxy Zabbix mogą używać różnych sejfów.

Jeśli opcja *Resolve secret vault macros by* jest ustawiona na *Zabbix server*, wówczas sekrety sejfu są pobierane tylko przez serwer, a proxy Zabbix otrzymuje wartości makr sekretów Vault z serwera Zabbix podczas każdej synchronizacji konfiguracji i przechowuje je we własnej pamięci podręcznej konfiguracji. Oznacza to, że proxy Zabbix nie może rozpocząć zbierania danych po ponownym uruchomieniu, dopóki nie otrzyma aktualizacji konfiguracji z serwera Zabbix.

Aby ręcznie odświeżyć wartości sekretów z sejfu, użyj opcji kontroli wykonania `secrets_reload` (tylko serwer).

Szyfrowanie musi być włączone między serwerem Zabbix a proxy; w przeciwnym razie rejestrowany jest komunikat ostrzegawczy serwera.

Warning:
Jeśli wartości makra nie można pomyślnie pobrać, odpowiednia pozycja używająca tej wartości stanie się nieobsługiwana.

Niezamaskowane lokalizacje

Ta lista zawiera lokalizacje parametrów, w których wartości makr tajnych są niezamaskowane.

Note:
Wartości makr tajnych pozostaną zamaskowane w poniższych lokalizacjach, jeśli będą odwoływane pośrednio. Na przykład {ITEM.KEY}, {ITEM.KEY<1-9>}, {LLDRULE.KEY} wbudowane makra używane w typach mediów (parametry Script lub Webhook) zostaną rozwiązane do kluczy pozycji zawierających zamaskowane tajne makra, takie jak `net.tcp.port[******,******]` zamiast `net.tcp.port[192.0.2.1,80]`.

Context		Parameter
Pozycje, prototypy pozycji, reguły LLD		
	Pozycja	*Parametry klucza pozycji*
	Prototyp pozycji	*Parametry klucza prototypu pozycji*
	Reguła wykrywania niskiego poziomu	*Parametry klucza pozycji wykrywania*
	agent SNMP	*Społeczność SNMP*
		Nazwa kontekstu (SNMPv3)
		Nazwa bezpieczeństwa (SNMPv3)
		Hasło uwierzytelniania (SNMPv3)
		Hasło prywatności (SNMPv3)
	agent HTTP	*URL*
		Pola zapytania
		Treść żądania
		Nagłówki
		Nazwa użytkownika
		Hasło
		Hasło klucza SSL

Context		Parameter
	Script	*Parametry* *Skrypt*
	Browser	*Parametry* *Skrypt*
	Monitor bazy danych	*Zapytanie SQL*
	agent TELNET	*Skrypt* *Nazwa użytkownika* *Hasło*
	agent SSH	*Skrypt* *Nazwa użytkownika* *Hasło*
	Simple check	*Nazwa użytkownika* *Hasło*
	agent JMX	*Nazwa użytkownika* *Hasło*
Wstępne przetwarzanie wartości pozycji		
	Krok wstępnego przetwarzania JavaScript	*Skrypt*
Scenariusze webowe		
	Scenariusz webowy	*Wartość zmiennej* *Wartość nagłówka* *URL* *Wartość pola zapytania* *Wartość pola POST* *Surowy POST*
	Uwierzytelnianie scenariusza webowego	*Użytkownik* *Hasło* *Hasło klucza SSL*
Łączniki		
	Łącznik	*URL* *Nazwa użytkownika* *Hasło* *Token* *Proxy HTTP* *Plik certyfikatu SSL* *Plik klucza SSL* *Hasło klucza SSL*
Wykrywanie sieci		
	SNMP	*Społeczność SNMP* *Nazwa kontekstu* (SNMPv3) *Nazwa bezpieczeństwa* (SNMPv3) *Hasło uwierzytelniania* (SNMPv3) *Hasło prywatności* (SNMPv3)
Skrypty globalne		
	Webhook	*Skrypt JavaScript* *Wartość parametru skryptu JavaScript*
	Telnet	*Nazwa użytkownika* *Hasło*
	SSH	*Nazwa użytkownika* *Hasło*
	Script	*Skrypt*
Typy mediów		
	Script	*Parametry skryptu*
	Webhook	*Parametry*
Zarządzanie IPMI		
	Host	*Nazwa użytkownika* *Hasło*

5 Makra wykrywania niskiego poziomu

Przegląd

Istnieje typ makra używanego w funkcji low-level discovery (LLD):

```
{#MACRO}
```

Jest to makro używane w regule LLD, które zwraca rzeczywiste wartości nazwy systemu plików, interfejsu sieciowego, SNMP OID itp.

Te makra mogą być używane do tworzenia prototypów encji (pozycja, wyzwalacz, wykres, reguła LLD, prototyp hosta i grupy hostów). Podczas wykrywania rzeczywistych systemów plików, interfejsów sieciowych, maszyn wirtualnych itp. makra te są zastępowane rzeczywistymi wartościami i stanowią podstawę do tworzenia rzeczywistych encji.

Niektóre makra low-level discovery są dostarczane „w pakiecie" z funkcją LLD w Zabbix — {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. Jednak podczas tworzenia niestandardowej reguły low-level discovery nie ma obowiązku stosowania tych nazw. Możesz wtedy użyć dowolnej innej nazwy makra LLD i odwoływać się do tej nazwy.

Obsługiwane typy danych

Podczas definiowania niestandardowych reguł wykrywania wartości właściwości zwracane w obiektach JSON dla makr LLD muszą być jednym z następujących typów prostych:

- string
- number
- boolean

Tablice, obiekty i wartości null nie są obsługiwane. Każde makro LLD odwołujące się do takiej wartości pozostanie nierozwinięte i będzie wyświetlane dosłownie (np. `'{#MY_MACRO}'`) podczas wstępnego przetwarzania i tworzenia pozycji.

Obsługiwane lokalizacje

Makra LLD mogą być używane:

- w filtrze reguły wykrywania niskiego poziomu
- dla prototypów pozycji i prototypów wykrywania w
  - nazwie
  - parametrach klucza
  - jednostce
  - interwale aktualizacji[1]
  - czasie limitu[1]
  - okresie przechowywania historii[1]
  - okresie przechowywania trendów[1]
  - krokach przetwarzania wstępnego wartości pozycji
  - SNMP OID
  - polu czujnika IPMI
  - wyrażeniu pozycji obliczanej/agregowanej, w:
    - stałych wyrażenia i parametrach funkcji
    - parametrach klucza pozycji
  - warunkach filtra pozycji agregowanej (nazwa grupy hostów i nazwa tagu)
  - skrypcie SSH i skrypcie Telnet
  - zapytaniu SQL monitorowania bazy danych
  - polu punktu końcowego pozycji JMX
  - opisie
  - polu URL agenta HTTP
  - polu pól zapytania HTTP agenta HTTP
  - polu treści żądania agenta HTTP
  - polu wymaganych kodów statusu agenta HTTP
  - kluczu i wartości pól nagłówków agenta HTTP
  - polu nazwy użytkownika uwierzytelniania HTTP agenta HTTP
  - polu hasła uwierzytelniania HTTP agenta HTTP
  - polu proxy HTTP agenta HTTP
  - polu pliku certyfikatu SSL HTTP agenta HTTP
  - polu pliku klucza SSL HTTP agenta HTTP
  - polu hasła klucza SSL HTTP agenta HTTP
  - tagach
- dla prototypów wyzwalaczy w
  - nazwie
  - danych operacyjnych
  - wyrażeniu (tylko w stałych i parametrach funkcji)
  - URL

    - opisie
    - tagach
- dla prototypów wykresów w
    - nazwie
- dla prototypów hostów w
    - nazwie
    - nazwie widocznej
    - niestandardowych polach interfejsu: IP, DNS, port, SNMP v1/v2 community, SNMP v3 context name, SNMP v3 security name, SNMP v3 authentication passphrase, SNMP v3 privacy passphrase
    - nazwie prototypu grupy hostów
    - wartości tagu hosta
    - wartości makra hosta
    - (zobacz pełną listę)

We wszystkich tych miejscach, z wyjątkiem filtra reguły wykrywania niskiego poziomu, makra LLD mogą być używane wewnątrz statycznego kontekstu makra użytkownika.

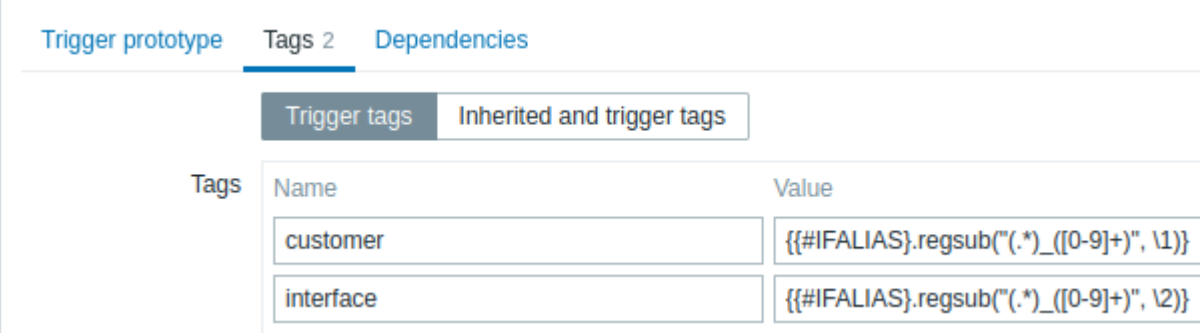
Używanie funkcji makr

Funkcje makr są obsługiwane w makrach niskopoziomowego wykrywania, z wyjątkiem filtra reguły niskopoziomowego wykrywania, i umożliwiają wyodrębnienie określonej części wartości makra za pomocą wyrażenia regularnego.

Na przykład możesz chcieć wyodrębnić nazwę klienta i numer interfejsu z następującego makra LLD na potrzeby tagowania zdarzeń:

```
{#IFALIAS}=customername_1
```

Aby to zrobić, można użyć funkcji makra `regsub` z makrem w polu wartości tagu zdarzenia w prototypie wyzwalacza:



Zwróć uwagę, że przecinki nie są dozwolone w nieujętych w cudzysłów parametrach klucza pozycji, dlatego parametr zawierający funkcję makra musi być ujęty w cudzysłów. Do poprzedzenia podwójnych cudzysłowów wewnątrz parametru znakiem ucieczki należy użyć znaku ukośnika odwrotnego (\). Przykład:

```
net.if.in["{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}",bytes]
```

Więcej informacji o składni funkcji makr można znaleźć tutaj: Funkcje makr

Funkcje makr są obsługiwane w makrach niskopoziomowego wykrywania od wersji Zabbix 4.0.

Prototypy pozycji bez parametrów klucza

Gdy parametry klucza pozycji nie są używane, umieść makro LLD w nawiasach parametru klucza pozycji `[...]`, aby było traktowane jako parametr i podstawiane podczas wykrywania:

```
v_[{#MACRO}]
```

Przypisy

1 W polach oznaczonych [1] pojedyncze makro musi wypełniać całe pole. Wiele makr w polu lub makra zmieszane z tekstem nie są obsługiwane.

6 Makra wyrażeń

Przegląd

Makra wyrażeń umożliwiają wykonywanie obliczeń w polach.

Ich wartość jest obliczana przez najpierw rozwiązanie wszystkich zagnieżdżonych makr, a następnie ocenę wynikowego wyrażenia.

Składnia:

```
{?EXPRESSION}
```

EXPRESSION używa tej samej składni i obsługuje te same funkcje co wyrażenia wyzwalaczy.

Przykład:

```
{?trendavg(/host/item1,1M:now/M)/trendavg(/host/item1,1M:now/M-1y)*100}
```

W miejscach, które obsługują makra wyrażeń, wyszukaj "{?EXPRESSION}" w tabeli obsługiwanych makr.

Uwagi dotyczące użycia:

- Używaj makr {FUNCTION.*}, aby odwoływać się do wartości funkcji wyrażeń wyzwalaczy/wyrażeń odzyskiwania.
- Używaj makr {HOST.HOST<1-9>} i {ITEM.KEY<1-9>}, aby odwoływać się do hostów i pozycji.
- W szablonach używaj makr {HOST.HOST<1-9>} lub pomiń host całkowicie dla pierwszego hosta — na przykład {?avg(//item1,1h)} — zamiast nazw szablonów, ponieważ nazwy szablonów nie są zastępowane nazwami hostów podczas łączenia szablonów.

```
{?{FUNCTION.VALUE2} - {FUNCTION.VALUE3}}
{?max(/{HOST.HOST}/{ITEM.KEY},3h)}
```

Zobacz też Przykłady wyrażeń wyzwalaczy, aby zobaczyć przykład użycia makra wyrażenia w nazwie zdarzenia.

12 Użytkownicy i grupy użytkowników

Przegląd

Wszyscy użytkownicy w Zabbix uzyskują dostęp do aplikacji Zabbix za pośrednictwem internetowego frontend. Każdemu użytkownikowi przypisywana jest unikalna nazwa logowania i hasło.

Wszystkie hasła użytkowników są szyfrowane i przechowywane w bazie danych Zabbix. Użytkownicy nie mogą używać swojego identyfikatora użytkownika i hasła do bezpośredniego logowania się do serwera UNIX, chyba że zostali również odpowiednio skonfigurowani w systemie UNIX. Komunikacja między serwerem WWW a przeglądarką użytkownika może być chroniona przy użyciu SSL.

Dzięki elastycznemu schematowi uprawnień użytkowników można ograniczać i rozróżniać uprawnienia do:

- dostępu do administracyjnych funkcji frontend Zabbix
- wykonywania określonych działań we frontend
- dostępu do monitorowanych hostów w grupach hostów
- korzystania z określonych metod API

1 Konfigurowanie użytkownika

Przegląd

Początkowa instalacja Zabbix zawiera dwóch predefiniowanych użytkowników:

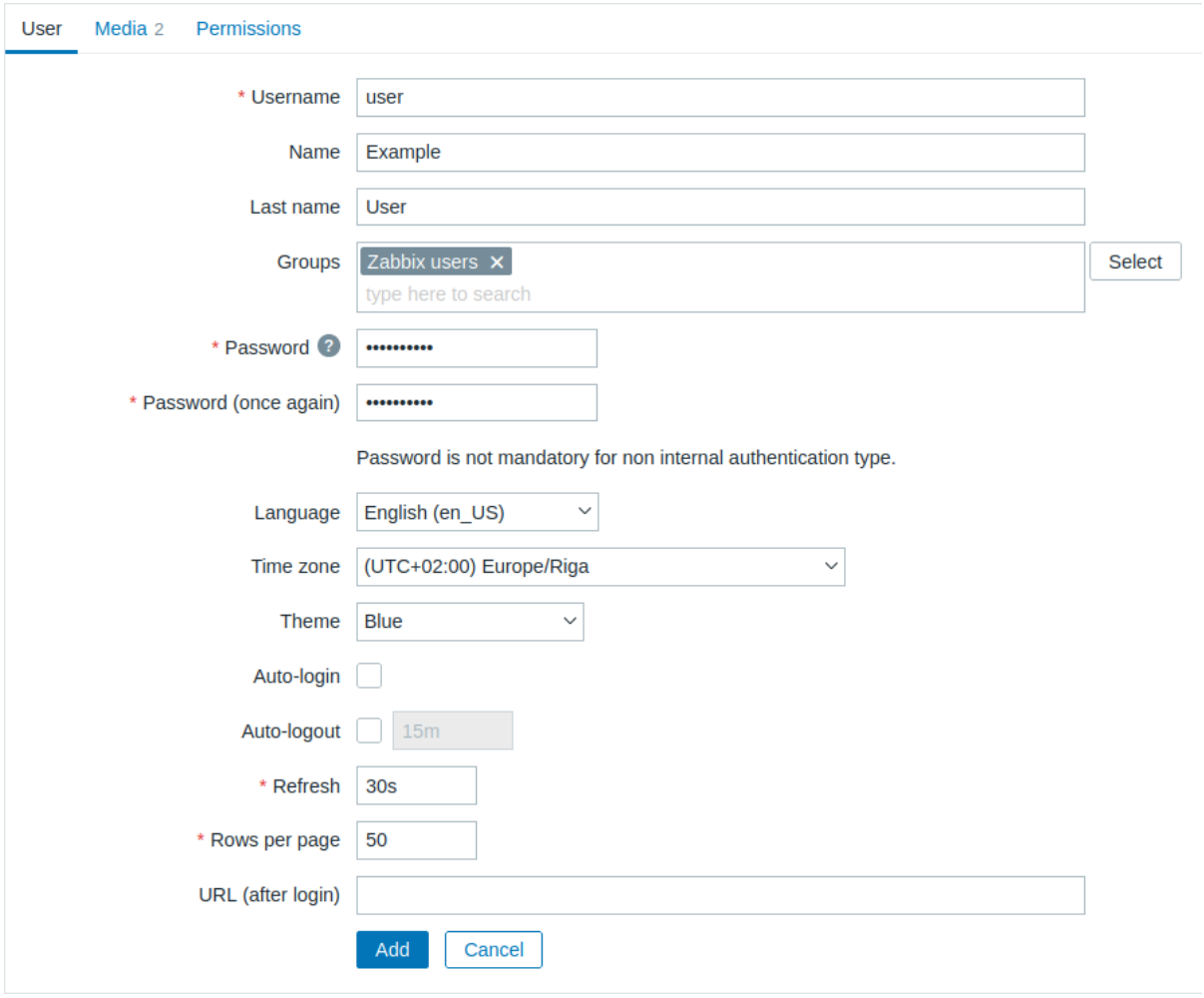
- *Admin* - superużytkownik Zabbix superuser z pełnymi uprawnieniami.
- *guest* - specjalny użytkownik Zabbix user. Użytkownik „guest" jest domyślnie wyłączony. Jeśli dodasz go do grupy użytkowników Guests, możesz zalogować się przy użyciu tego użytkownika i uzyskać dostęp do stron monitorowania w Zabbix. Pamiętaj, że domyślnie użytkownik „guest" nie ma żadnych uprawnień do obiektów Zabbix.

Aby skonfigurować użytkownika:

- Przejdź do *Users* → *Users*.
- Kliknij *Create user* (lub nazwę użytkownika, aby edytować istniejącego użytkownika).
- Edytuj atrybuty użytkownika w formularzu.

Atrybuty ogólne

Karta *User* zawiera ogólne atrybuty użytkownika:



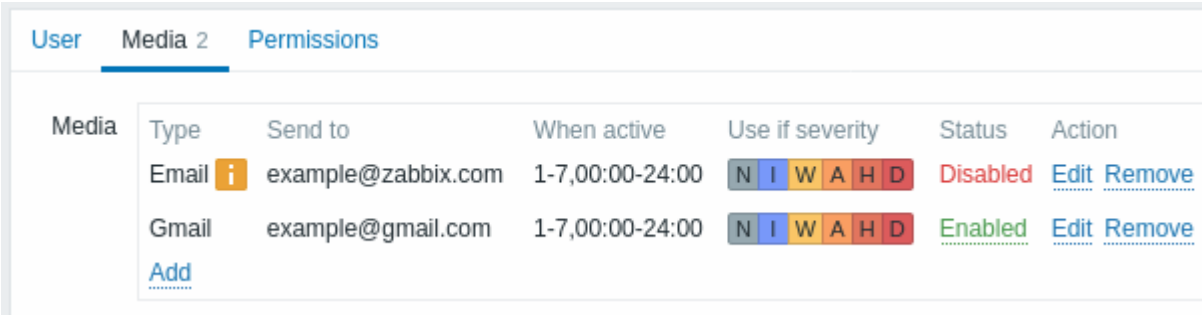
Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Username	Unikalna nazwa użytkownika, używana jako nazwa logowania.
Name	Imię użytkownika (opcjonalne). Jeśli nie jest puste, będzie widoczne w informacjach o potwierdzeniu oraz w informacjach o odbiorcy powiadomienia.
Last name	Nazwisko użytkownika (opcjonalne). Jeśli nie jest puste, będzie widoczne w informacjach o potwierdzeniu oraz w informacjach o odbiorcy powiadomienia.
Groups	Wybierz grupy użytkowników, do których należy użytkownik. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy użytkowników spowoduje wyświetlenie listy rozwijanej pasujących grup. Przewiń w dół, aby wybrać. Alternatywnie kliknij *Select*, aby dodać grupy. Kliknij 'x', aby usunąć zaznaczoną pozycję. Przynależność do grup użytkowników określa, do jakich grup hostów i hostów użytkownik będzie miał dostęp.
Password	Dwa pola do wprowadzenia hasła użytkownika albo przycisk *Change password*, jeśli użytkownik już istnieje. Kliknięcie przycisku *Change password* otwiera dwa pola do wprowadzenia nowego hasła. W przypadku użytkownika z rolą *Super admin role* zmieniającego własne hasło kliknięcie przycisku *Change password* otwiera dodatkowe pole do wprowadzenia bieżącego (starego) hasła. Po pomyślnej zmianie hasła użytkownik, którego hasło zostało zmienione, zostanie wylogowany ze wszystkich aktywnych sesji. Należy pamiętać, że hasło można zmienić tylko dla użytkowników korzystających z wewnętrznego uwierzytelniania Zabbix internal authentication.
Language	Język frontend Zabbix. Do działania tłumaczeń wymagana jest rozszerzenie php gettext.
Time zone	Wybierz strefę czasową, aby zastąpić globalną strefę czasową na poziomie użytkownika, lub wybierz **System default**, aby użyć globalnych ustawień strefy czasowej.

Parameter	Description
Theme	Określa wygląd frontend: **System default** - użyj domyślnych ustawień systemowych **Blue** - standardowy niebieski motyw **Dark** - alternatywny ciemny motyw **High-contrast light** - jasny motyw o wysokim kontraście **High-contrast dark** - ciemny motyw o wysokim kontraście
Auto-login	Zaznacz to pole wyboru, aby Zabbix zapamiętał użytkownika i automatycznie logował go przez 30 dni. Podczas logowania z opcją *Remember for 30 days*: - Auto-logout jest resetowane (sesja trwa 30 dni). - Auto-login jest włączone, aby umożliwić płynne ponowne uwierzytelnienie. Podczas logowania bez opcji *Remember for 30 days*: - Auto-login jest włączone, aby umożliwić płynne ponowne uwierzytelnienie. - Auto-logout pozostaje kontrolowane przez standardowe ustawienie limitu czasu. W tym celu używane są pliki cookie przeglądarki.
Auto-logout	Po zaznaczeniu tego pola użytkownik zostanie automatycznie wylogowany po upływie ustawionej liczby sekund (minimum 90 sekund, maksimum 1 dzień). Należy pamiętać, że to ustawienie jest nadpisywane, jeśli włączona jest opcja *Remember for 30 days*, ponieważ sesja jest przedłużana na cały okres. Obsługiwane są sufiksy czasu, np. 90s, 5m, 2h, 1d. Należy pamiętać, że ta opcja nie będzie działać: * Jeśli globalna opcja konfiguracji "Show warning if Zabbix server is down" jest włączona i frontend Zabbix pozostaje otwarty. * Gdy strony menu Monitoring wykonują odświeżanie informacji w tle. * Jeśli podczas logowania zaznaczono opcję *Remember me for 30 days*.
Refresh	Ustawia częstotliwość odświeżania używaną dla wykresów, danych tekstowych itp. Można ustawić 0, aby wyłączyć. Obsługiwane są sufiksy czasu, np. 90s, 5m, 1h.
Rows per page	Możesz określić, ile wierszy na stronę będzie wyświetlanych na listach.
URL (after login)	Możesz sprawić, aby Zabbix przekierował użytkownika do określonego adresu URL po pomyślnym zalogowaniu, na przykład na stronę *Problems*.

Media użytkownika

Zakładka *Media* zawiera listę wszystkich mediów zdefiniowanych dla użytkownika. Media są używane do wysyłania powiadomień.



Kliknij *Add*, aby przypisać media do użytkownika.

Jeśli typ mediów został wyłączony:

- Po nazwie wyświetlana jest żółta ikona informacyjna.
- W kolumnie Status wyświetlane jest *Disabled*.

Note:
Uprawnienia użytkownika do zmiany szczegółów własnych mediów mogą być przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit own media*). Uprawnienia użytkownika Super admin do zmiany szczegółów mediów innych użytkowników również mogą być przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit user media*)

Szczegółowe informacje dotyczące konfiguracji mediów użytkownika znajdują się w sekcji Typy mediów.

Uprawnienia

Karta *Uprawnienia* zawiera informacje o następujących elementach:

- Rola użytkownika (wymagana dla każdego nowo utworzonego użytkownika), którą może zmienić wyłącznie użytkownik typu *Super admin*.

Attention:
Użytkownicy nie mogą być tworzeni bez roli użytkownika (z wyjątkiem Zabbix User API). Wcześniej utworzeni użytkownicy, którzy nie mają przypisanej roli, mogą być nadal edytowani bez przypisywania im roli. Jednak po przypisaniu roli można ją tylko zmienić, a nie usunąć.

 Należy pamiętać, że użytkownicy bez roli mogą logować się do Zabbix tylko przy użyciu uwierzytelniania LDAP lub SAML, pod warunkiem że ich informacje LDAP/SAML odpowiadają mapowaniom grup użytkowników skonfigurowanym w Zabbix.

- Typ użytkownika (*User*, *Admin*, *Super admin*), który jest zdefiniowany w konfiguracji roli użytkownika.
- Grupy hostów i szablonów, do których użytkownik ma dostęp.
  - Użytkownicy typu *User* i *Admin* domyślnie nie mają dostępu do żadnych grup, szablonów ani hostów. Aby przyznać taki dostęp, użytkownicy muszą być uwzględnieni w grupach użytkowników skonfigurowanych z uprawnieniami do odpowiednich obiektów.
- Prawa dostępu do sekcji i elementów frontend Zabbix, modułów oraz metod API.
  - Elementy z dozwolonym dostępem są wyświetlane na zielono, natomiast te z odmową dostępu - na jasnoszaro.
- Uprawnienia do wykonywania określonych działań.
  - Działania, które użytkownik może wykonywać, są wyświetlane na zielono, natomiast te, które są zabronione - na jasnoszaro.

Szczegóły można znaleźć na stronie Uprawnienia.

2 Uprawnienia

Przegląd

Uprawnienia w Zabbix zależą od typu użytkownika, dostosowanych ról użytkowników oraz dostępu do hostów, który jest określany na podstawie grupy użytkowników.

Typy użytkowników

Uprawnienia w Zabbix zależą przede wszystkim od typu użytkownika:

- *Użytkownik* - ma ograniczone prawa dostępu do sekcji menu (zob. poniżej) i domyślnie nie ma dostępu do żadnych zasobów. Wszelkie uprawnienia do grup hostów lub grup szablonów muszą zostać przypisane jawnie.
- *Administrator* - ma niepełne prawa dostępu do sekcji menu (zob. poniżej). Użytkownik domyślnie nie ma dostępu do żadnych grup hostów. Wszelkie uprawnienia do grup hostów lub grup szablonów muszą zostać przyznane jawnie.
- *Super administrator* - ma dostęp do wszystkich sekcji menu. Użytkownik ma dostęp do odczytu i zapisu do wszystkich grup hostów i grup szablonów. Uprawnień nie można cofnąć przez odmowę dostępu do określonych grup.

Dostęp do menu

Note:
Ograniczony dostęp do niektórych elementów interfejsu użytkownika zapobiega jedynie otwarciu tej strony — nie usuwa możliwości dostępu do danych bazowych w innych częściach interfejsu.

Poniższa tabela przedstawia dostęp do sekcji menu Zabbix według typu użytkownika:

Sekcja menu		Użytkownik	Administrator	Super administrator
Pulpity		+	+	+
Monitorowanie		+	+	+
	Problemy	+	+	+
	Hosty	+	+	+
	Najnowsze dane	+	+	+
	Mapy	+	+	+
	Wykrywanie		+	+
Usługi		+	+	+
	Usługi	+	+	+
	SLA		+	+
	Raport SLA	+	+	+
Inwentaryzacja		+	+	+
	Przegląd	+	+	+

Sekcja menu		Użytkownik	Administrator	Super administrator
	Hosty	+	+	+
Raporty		+	+	+
	Informacje o systemie			+
	Raporty cykliczne		+	+
	Raport dostępności	+	+	+
	Top 100 wyzwalaczy	+	+	+
	Dziennik audytu			+
	Dziennik działań			+
	Powiadomienia		+	+
Zbieranie danych			+	+
	Grupy szablonów		+	+
	Grupy hostów		+	+
	Szablony		+	+
	Hosty		+	+
	Konserwacja		+	+
	Korelacja zdarzeń			+
	Wykrywanie		+	+
Alerty			+	+
	Akcje wyzwalaczy		+	+
	Akcje usług		+	+
	Akcje wykrywania		+	+
	Akcje autorejestracji		+	+
	Akcje wewnętrzne		+	+
	Typy mediów			+
	Skrypty			+
Użytkownicy				+
	Grupy użytkowników			+
	Role użytkowników			+
	Użytkownicy			+
	Tokeny API			+
	Uwierzytelnianie			+
Administracja				+
	Ogólne			+
	Dziennik audytu			+
	Housekeeping			+
	Grupy proxy			+
	Proxy			+
	Makra			+
	Kolejka			+

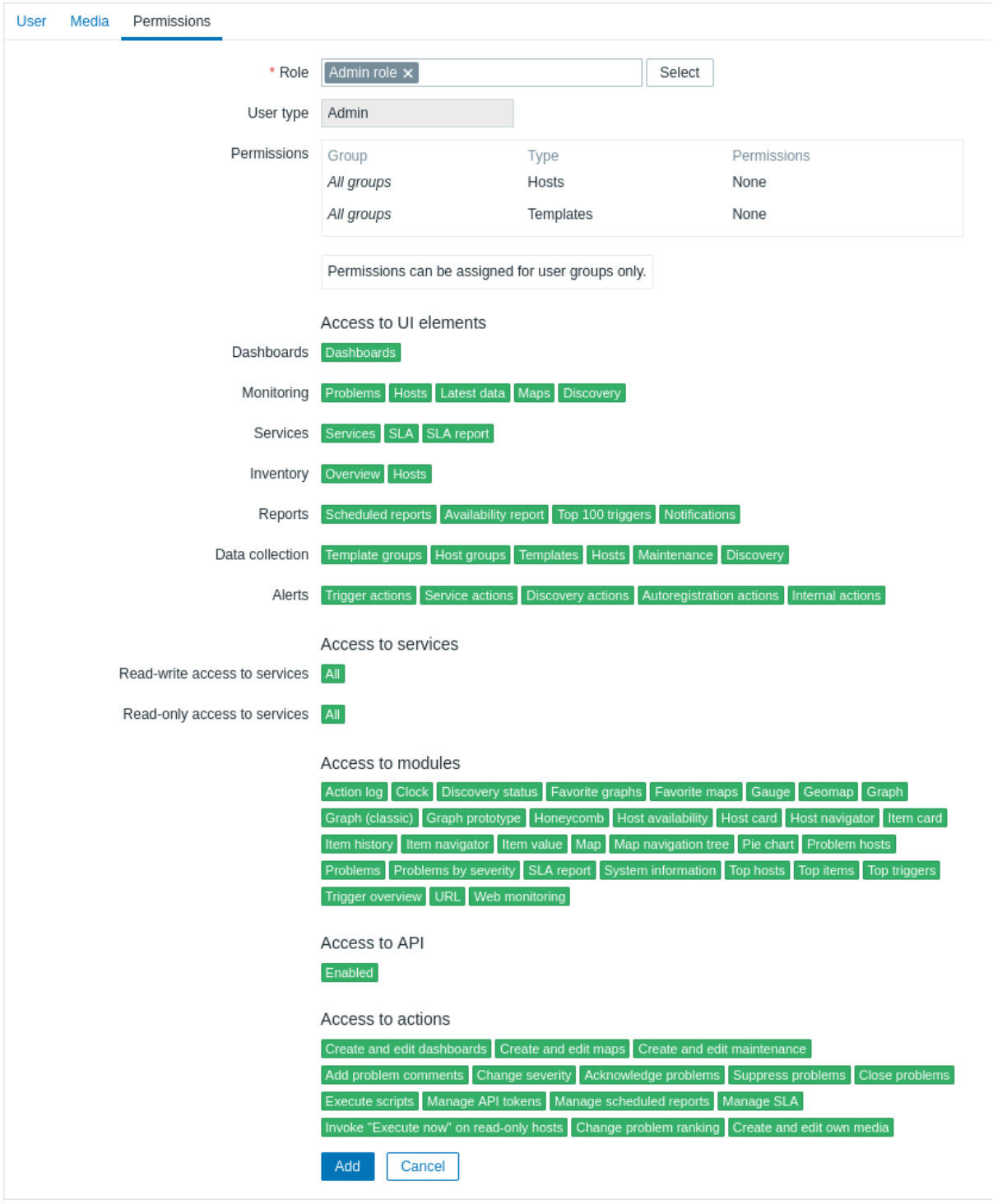
Role użytkowników

Role użytkowników umożliwiają wprowadzanie niestandardowych zmian w uprawnieniach zdefiniowanych przez typ użytkownika. Chociaż nie można dodawać uprawnień (ponieważ wykraczałoby to poza uprawnienia typu użytkownika), niektóre uprawnienia można odebrać.

Ponadto rola użytkownika określa dostęp nie tylko do sekcji menu, ale także do usług, modułów, metod API oraz różnych działań we frontend.

Role użytkowników są konfigurowane w sekcji *Użytkownicy* > *Role użytkowników* przez użytkowników typu Super admin.

Role użytkowników są przypisywane użytkownikom w formularzu konfiguracji użytkownika, na karcie *Uprawnienia*, przez użytkowników typu Super admin.



Dostęp do hostów

Dostęp do danych dowolnego hosta i szablonu w Zabbix jest przyznawany grupom użytkowników wyłącznie na poziomie grup hostów/szablonów.

Oznacza to, że pojedynczemu użytkownikowi nie można bezpośrednio przyznać dostępu do hosta (ani grupy hostów). Dostęp do hosta można przyznać tylko poprzez członkostwo w grupie użytkowników, której przyznano dostęp do grupy hostów zawierającej tego hosta.

Podobnie, użytkownik może uzyskać dostęp do szablonu wyłącznie poprzez członkostwo w grupie użytkowników, której przyznano dostęp do grupy szablonów zawierającej dany szablon.

3 Grupy użytkowników

Przegląd

Grupy użytkowników umożliwiają grupowanie użytkowników zarówno w celach organizacyjnych, jak i do przypisywania uprawnień do danych. Uprawnienia do przeglądania i konfigurowania danych grup hostów i grup szablonów są przypisywane do grup użytkown-

ików, a nie do poszczególnych użytkowników.

Często ma sens rozdzielenie informacji dostępnych dla jednej grupy użytkowników i dla innej. Można to osiągnąć przez grupowanie użytkowników, a następnie przypisywanie różnych uprawnień do grup hostów i szablonów.

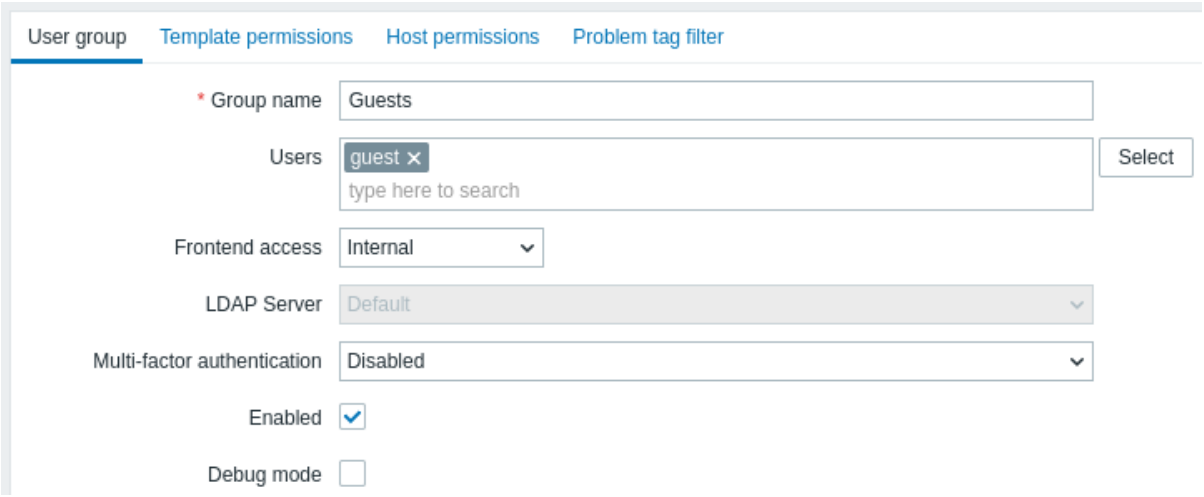
Użytkownik może należeć do dowolnej liczby grup.

Konfiguracja

Aby skonfigurować grupę użytkowników:

- Przejdź do *Users* → *User groups*
- Kliknij *Create user group* (lub nazwę grupy, aby edytować istniejącą grupę)
- Edytuj atrybuty grupy w formularzu

Karta **User group** zawiera ogólne atrybuty grupy:

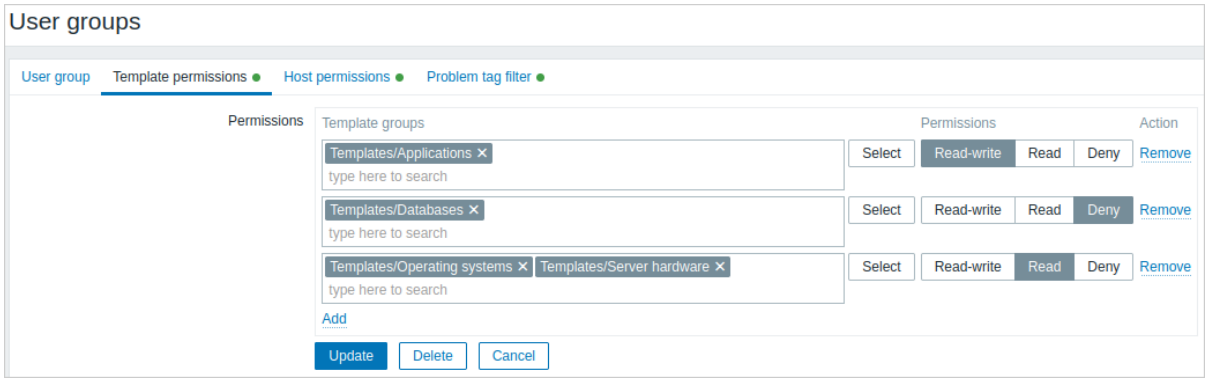


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

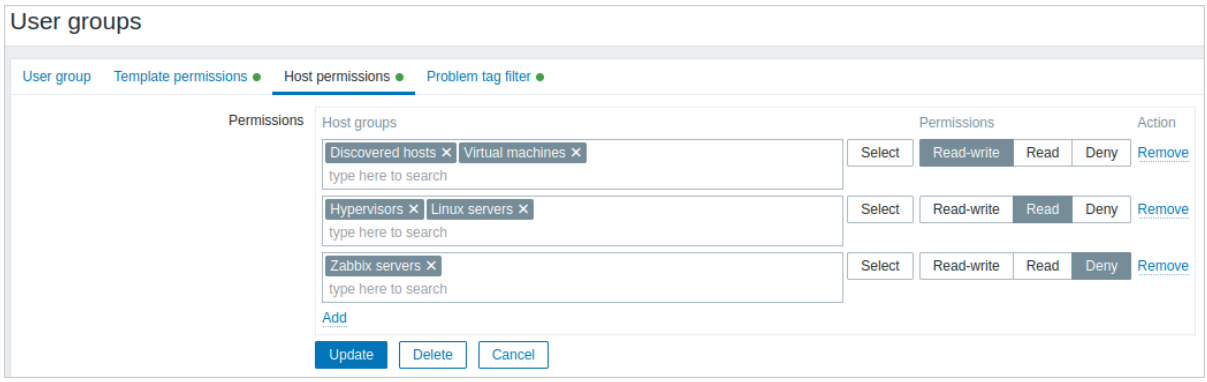
Parameter	Description
Group name	Unikalna nazwa grupy.
Users	Aby dodać użytkowników do grupy, zacznij wpisywać nazwę istniejącego użytkownika. Gdy pojawi się lista rozwijana z pasującymi nazwami użytkowników, przewiń w dół, aby wybrać. Alternatywnie możesz kliknąć przycisk *Select*, aby wybrać użytkowników w oknie podręcznym.
Frontend access	Sposób uwierzytelniania użytkowników grupy. **System default** - użyj domyślnej metody uwierzytelniania (ustawionej globalnie) **Internal** - użyj wewnętrznego uwierzytelniania Zabbix (nawet jeśli globalnie używane jest uwierzytelnianie LDAP). Ignorowane, jeśli globalnym domyślnym ustawieniem jest uwierzytelnianie HTTP. **LDAP** - użyj uwierzytelniania LDAP (nawet jeśli globalnie używane jest uwierzytelnianie wewnętrzne). Ignorowane, jeśli globalnym domyślnym ustawieniem jest uwierzytelnianie HTTP. **Disabled** - dostęp do frontend Zabbix jest zabroniony dla tej grupy
LDAP server	Wybierz, którego serwera LDAP użyć do uwierzytelnienia użytkownika. To pole jest dostępne tylko wtedy, gdy *Frontend access* ma ustawioną wartość LDAP lub System default.
Multi-factor authentication	Wybierz, której metody uwierzytelniania wieloskładnikowego użyć do uwierzytelnienia użytkownika: **Default** - użyj metody ustawionej jako domyślna w konfiguracji MFA; ta opcja jest domyślnie wybrana dla nowych grup użytkowników, jeśli MFA jest włączone; **<Method name>** - użyj wybranej metody (na przykład ”Zabbix TOTP”); **Disabled** - MFA jest wyłączone dla tej grupy; ta opcja jest domyślnie wybrana dla nowych grup użytkowników, jeśli MFA jest wyłączone. Pamiętaj, że jeśli użytkownik należy do wielu grup użytkowników z włączonym MFA (lub co najmniej jedna grupa ma włączone MFA), obowiązują następujące zasady uwierzytelniania: jeśli dowolna grupa używa metody MFA ”Default”, zostanie ona użyta do uwierzytelnienia użytkownika; w przeciwnym razie zostanie użyta pierwsza metoda (w kolejności alfabetycznej).
Enabled	Status grupy użytkowników i jej członków. *Checked* - grupa użytkowników i użytkownicy są włączeni *Unchecked* - grupa użytkowników i użytkownicy są wyłączeni

Parameter	Description
Debug mode	Zaznacz to pole wyboru, aby włączyć tryb debugowania dla użytkowników.

Karta **Template permissions** umożliwia określenie dostępu grupy użytkowników do danych grup szablonów (a tym samym szablonów):



Karta **Host permissions** umożliwia określenie dostępu grupy użytkowników do danych grup hostów (a tym samym hostów):



Kliknij Add, aby wybrać grupy szablonów/hostów (czy to grupę nadrzędną, czy zagnieżdżoną) i przypisać do nich uprawnienia. Zacznij wpisywać nazwę grupy (pojawi się lista rozwijana pasujących grup) lub kliknij *Select*, aby otworzyć okno podręczne z listą wszystkich grup.

Następnie użyj przycisków opcji, aby przypisać uprawnienia do wybranych grup. Dostępne uprawnienia to:

- **Read-write** - dostęp do grupy z możliwością odczytu i zapisu;
- **Read** - dostęp tylko do odczytu do grupy;
- **Deny** - dostęp do grupy zabroniony.

Jeśli ta sama grupa szablonów/hostów zostanie dodana w kilku wierszach z różnymi ustawionymi uprawnieniami, zostanie zastosowane najbardziej restrykcyjne uprawnienie.

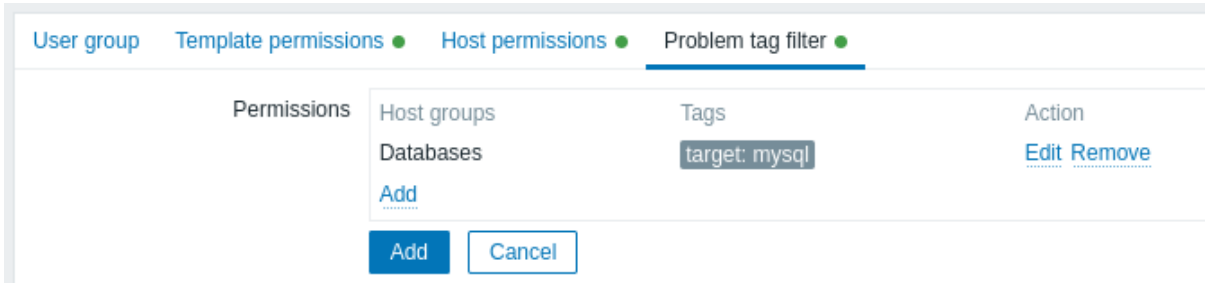
Pamiętaj, że użytkownik *Super admin* może wymusić, aby grupy zagnieżdżone miały ten sam poziom uprawnień co grupa nadrzędna; można to zrobić w formularzu konfiguracji grupy host/template.

Karty **Template permissions** i **Host permissions** obsługują ten sam zestaw parametrów.

Bieżące uprawnienia do grup są wyświetlane w bloku *Permissions* i można je modyfikować lub usuwać.

Note:
Jeśli grupa użytkowników ma uprawnienia **Read-write** do hosta oraz **Deny** lub brak uprawnień do szablonu powiązanego z tym hostem, użytkownicy tej grupy nie będą mogli edytować elementów dziedziczonych po szablonie na hoście, a nazwa szablonu będzie wyświetlana jako *Inaccessible template*.

Karta **Problem tag filter** umożliwia ustawienie uprawnień opartych na tagach dla grup użytkowników, aby widziały problemy filtrowane według nazwy i wartości tagu:

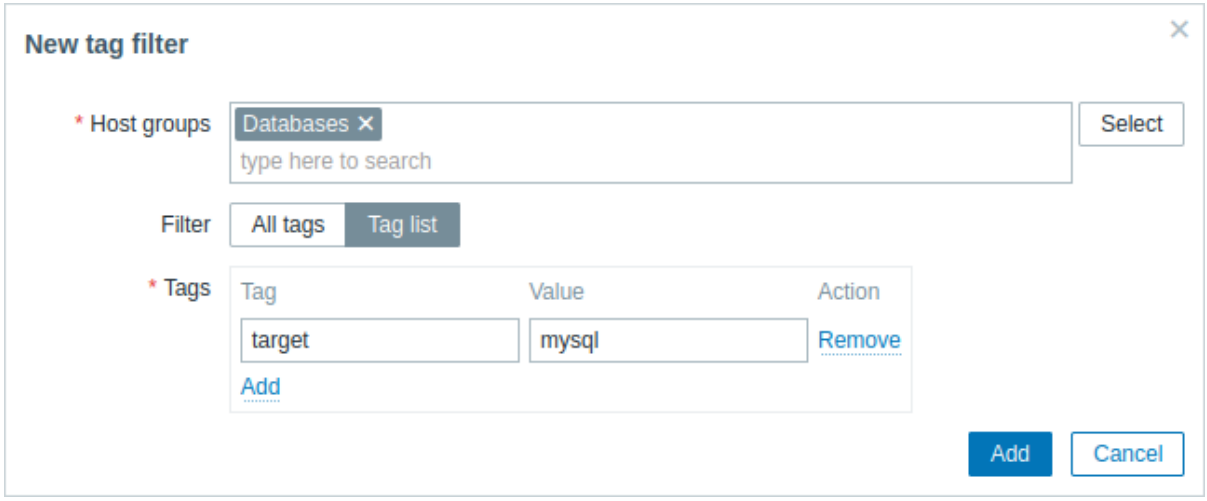


Kliknij Add, aby wybrać grupy hostów. Aby wybrać grupę hostów, dla której ma zostać zastosowany filtr tagów, kliknij *Select*, aby uzyskać pełną listę istniejących grup hostów, lub zacznij wpisywać nazwę grupy hostów, aby wyświetlić listę rozwijaną pasujących grup. Zostaną wyświetlone tylko grupy hostów, ponieważ filtr tagów problemów nie może być zastosowany do grup szablonów.

Następnie można przełączyć z *All tags* na *Tag list*, aby ustawić konkretne tagi i ich wartości do filtrowania. Można dodać nazwy tagów bez wartości, ale nie można dodać wartości bez nazw. Tylko pierwsze trzy tagi (z wartościami, jeśli występują) są wyświetlane w bloku *Permissions*; jeśli jest ich więcej, można je zobaczyć, klikając lub najeżdżając kursorem na ikonę ••• .

Filtr tagów pozwala oddzielić dostęp do grupy hostów od możliwości przeglądania problemów.

Na przykład, jeśli administrator bazy danych ma widzieć tylko problemy bazy danych ”MySQL”, należy najpierw utworzyć grupę użytkowników dla administratorów baz danych, a następnie określić nazwę tagu ”target” i wartość ”mysql”.



Jeśli zostanie określona nazwa tagu ”target”, a pole wartości pozostanie puste, grupa użytkowników będzie widzieć wszystkie problemy z tagiem o nazwie ”target” dla wybranej grupy hostów. Jeśli wybrano *All tags*, grupa użytkowników będzie widzieć wszystkie problemy dla określonej grupy hostów.

Upewnij się, że nazwa tagu i wartość tagu są poprawnie określone, w przeciwnym razie grupa użytkowników nie będzie widzieć żadnych problemów.

Przeanalizujmy przykład, w którym użytkownik jest członkiem kilku wybranych grup użytkowników. Filtrowanie w tym przypadku będzie używać warunku OR dla tagów.

User group A			**User group B**			**Visible result for a user (member) of both groups**
Tag filter						
Host group	*Tag name*	*Tag value*	*Host group*	*Tag name*	*Tag value*	
Databases	target	mysql	Databases	target	oracle	target:mysql or target:oracle problems visible
Databases	set to: *All tags*		Databases	target	oracle	All problems visible
Not configured in the **Problem tag filter**			Databases	target	oracle	target:oracle problems visible

Attention:
Dodanie filtra (na przykład wszystkich tagów w określonej grupie hostów "Databases") powoduje, że nie będzie można widzieć problemów innych grup hostów.

Dostęp z kilku grup użytkowników

Użytkownik może należeć do dowolnej liczby grup użytkowników. Grupy te mogą mieć różne uprawnienia dostępu do hostów lub szablonów.

Dlatego ważne jest, aby wiedzieć, do jakich obiektów użytkownik bez uprawnień będzie mógł uzyskać dostęp w rezultacie. W poniższym przykładzie rozważ, jak dostęp do hosta **X** (w grupie hostów 1) będzie wpływany w różnych sytuacjach dla użytkownika, który należy do grup użytkowników A i B.

- Jeśli grupa A ma tylko dostęp *Read* do grupy hostów 1, a grupa B dostęp *Read-write* do grupy hostów 1, użytkownik otrzyma dostęp **Read-write** do 'X'.

Attention:
Uprawnienia "Read-write" mają pierwszeństwo przed uprawnieniami "Read".

- W tym samym scenariuszu co powyżej, jeśli 'X' jednocześnie znajduje się także w grupie hostów 2, która jest **odmówiona** grupie A lub B, dostęp do 'X' będzie **niedostępny**, mimo dostępu *Read-write* do grupy hostów 1.
- Jeśli grupa A nie ma zdefiniowanych uprawnień, a grupa B ma dostęp *Read-write* do grupy hostów 1, użytkownik otrzyma dostęp **Read-write** do 'X'.
- Jeśli grupa A ma dostęp *Deny* do grupy hostów 1, a grupa B ma dostęp *Read-write* do grupy hostów 1, użytkownik otrzyma dostęp do 'X' **odmówiony**.

Inne szczegóły

- Użytkownik na poziomie Admin z dostępem *Read-write* do hosta nie będzie mógł łączyć/odłączać szablonów, jeśli nie ma dostępu do grupy szablonów, do której one należą. Przy dostępie *Read* do grupy szablonów będzie mógł łączyć/odłączać szablony z hostem, jednak nie zobaczy żadnych szablonów na liście szablonów i nie będzie mógł wykonywać operacji na szablonach w innych miejscach.
- Użytkownik na poziomie Admin z dostępem *Read* do hosta nie zobaczy hosta na liście hostów w sekcji konfiguracji; jednak wyzwalacze hosta będą dostępne w konfiguracji usługi IT.
- Każdy użytkownik inny niż Super Admin (w tym 'guest') może widzieć mapy sieciowe, o ile mapa jest pusta lub zawiera tylko obrazy. Gdy do mapy zostaną dodane hosty, grupy hostów lub wyzwalacze, uprawnienia są respektowane.
- serwer Zabbix nie będzie wysyłał powiadomień do użytkowników zdefiniowanych jako odbiorcy operacji akcji, jeśli dostęp do danego hosta jest jawnie "denied".

13 Przechowywanie sekretów

Przegląd

Zabbix może być skonfigurowany do pobierania poufnych informacji z bezpiecznego sejfu. Obsługiwane są następujące usługi zarządzania sekretami: HashiCorp Vault KV Secrets Engine - Version 2, CyberArk Vault CV12.

Sekrety mogą być używane do pobierania:

- wartości makr użytkownika
- poświadczeń dostępu do bazy danych

Zabbix zapewnia dostęp tylko do odczytu do sekretów w sejfie, zakładając, że sekretami zarządza ktoś inny.

Aby uzyskać informacje o konfiguracji konkretnego dostawcy sejfu, zobacz:

- Konfiguracja HashiCorp
- Konfiguracja CyberArk

Buforowanie wartości sekretów

Domyślnie wartości makr sekretów z vault są pobierane przez serwer Zabbix przy każdym odświeżeniu danych konfiguracyjnych, a następnie przechowywane w pamięci podręcznej konfiguracji. Zabbix proxy otrzymuje wartości makr sekretów z vault od serwera Zabbix przy każdej synchronizacji konfiguracji i przechowuje je we własnej pamięci podręcznej konfiguracji.

Attention:
Szyfrowanie musi być włączone między serwerem Zabbix a proxy; w przeciwnym razie rejestrowany jest komunikat ostrzegawczy serwera.

Można również skonfigurować, aby wartości makr były pobierane niezależnie przez serwer Zabbix i Zabbix proxy.

Aby ręcznie wywołać odświeżenie buforowanych wartości sekretów z vault, użyj opcji wiersza poleceń 'secrets_reload' option.

W przypadku poświadczeń bazy danych frontend buforowanie jest domyślnie wyłączone, ale można je włączyć, ustawiając opcję `$DB['VAULT_CACHE'] = true` w pliku zabbix.conf.php. Poświadczenia będą przechowywane w lokalnej pamięci podręcznej z użyciem katalogu tymczasowych plików systemu plików. Serwer WWW musi zezwalać na zapis w prywatnym folderze tymczasowym (na przykład dla Apache musi być ustawiona opcja konfiguracyjna `PrivateTmp=True`). Aby kontrolować, jak często pamięć podręczna danych jest odświeżana/unieważniana, użyj stałej ZBX_DATA_CACHE_TTL constant .

Konfiguracja TLS

Aby skonfigurować TLS do komunikacji między komponentami Zabbix a skarbcem, dodaj certyfikat podpisany przez urząd certyfikacji (CA) do domyślnego magazynu CA dla całego systemu. Aby użyć innej lokalizacji, określ katalog w parametrze konfiguracyjnym SSLCALocation Zabbix serwer/proxy, umieść plik certyfikatu w tym katalogu, a następnie uruchom polecenie CLI:

```
c_rehash .
```

1 Konfiguracja CyberArk

W tej sekcji wyjaśniono, jak skonfigurować Zabbix do pobierania sekretów z CyberArk Vault CV12.

Vault powinien być zainstalowany i skonfigurowany zgodnie z opisem w oficjalnej dokumentacji CyberArk.

Aby dowiedzieć się więcej o konfiguracji TLS w Zabbix, zobacz *Przechowywanie sekretów*.

Dane uwierzytelniające bazy danych

Dostęp do sekretu zawierającego dane uwierzytelniające bazy danych jest konfigurowany oddzielnie dla każdego komponentu Zabbix.

Serwer i proxy

Aby pobrać dane uwierzytelniające bazy danych z sejfu dla Zabbix serwer lub proxy, należy określić następujące parametry konfiguracyjne w pliku konfiguracyjnym:

- `Vault` - który dostawca sejfu ma być używany;
- `VaultURL` - adres URL HTTP[S] serwera sejfu;
- `VaultDBPath` - zapytanie do sekretu sejfu zawierającego dane uwierzytelniające bazy danych, które zostaną pobrane przy użyciu kluczy "Content" i "UserName" (tej opcji można użyć tylko wtedy, gdy nie określono DBUser i DBPassword);
- `VaultTLSCertFile`, `VaultTLSKeyFile` - nazwy plików certyfikatu SSL i klucza; skonfigurowanie tych opcji nie jest obowiązkowe, ale jest zdecydowanie zalecane;
- `VaultPrefix` - niestandardowy prefiks ścieżki lub zapytania sejfu, w zależności od sejfu; jeśli nie zostanie określony, użyta zostanie najbardziej odpowiednia wartość domyślna.

Attention:
Parametry konfiguracyjne `Vault`, `VaultURL`, `VaultTLSCertFile`, `VaultTLSKeyFile` oraz `VaultPrefix` są również używane do uwierzytelniania w sejfie podczas przetwarzania makr sekretów sejfu przez Zabbix serwer (oraz Zabbix proxy, jeśli jest skonfigurowany). Zabbix serwer i proxy nie będą otwierać makr sekretów sejfu zawierających dane uwierzytelniające bazy danych z VaultDBPath.

Zabbix serwer i Zabbix proxy odczytują parametry konfiguracyjne związane z sejfem z plików *zabbix_server.conf* i *zabbix_proxy.conf* podczas uruchamiania.

Przykład

1. W *zabbix_server.conf* określ następujące parametry:

```
Vault=CyberArk
VaultURL=https://127.0.0.1:1858
VaultDBPath=AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
VaultTLSCertFile=cert.pem
VaultTLSKeyFile=key.pem
VaultPrefix=/AIMWebService/api/Accounts?
```

2. Zabbix wyśle do sejfu następujące żądanie API:

```
curl \
--header "Content-Type: application/json" \
--cert cert.pem \
--key key.pem \
https://127.0.0.1:1858/AIMWebService/api/Accounts?AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbi
```

3. Odpowiedź sejfu będzie zawierać klucze „Content” i „UserName”:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database": <Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

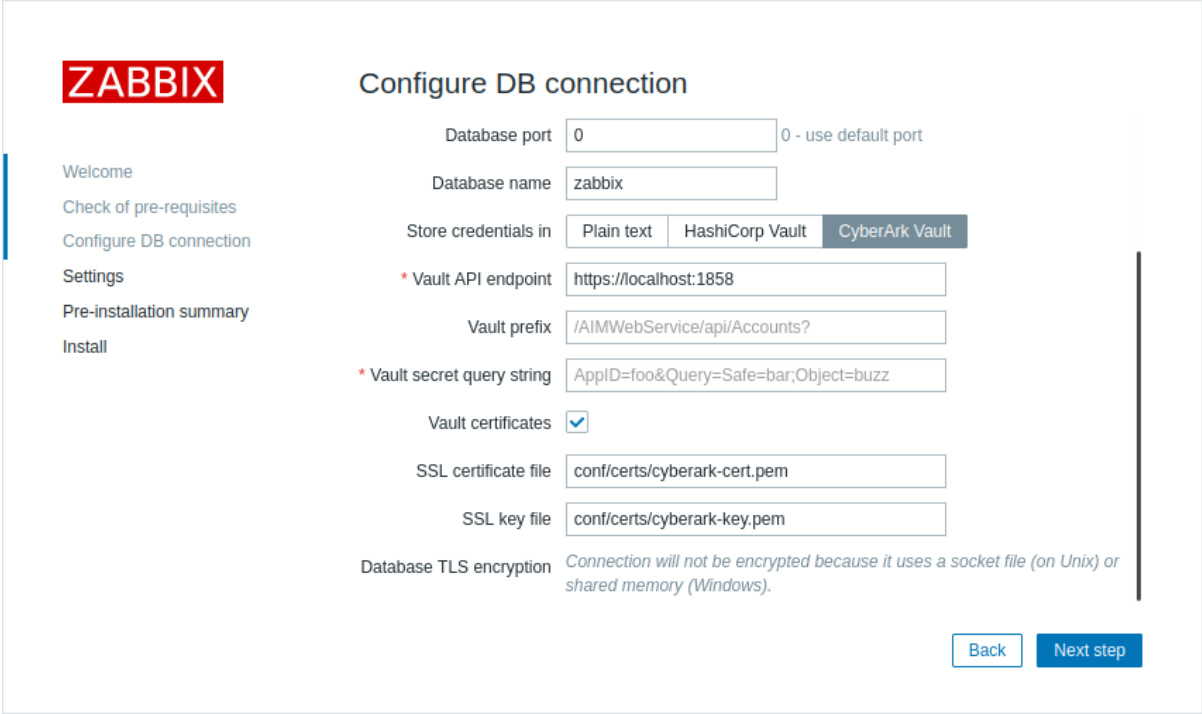
4. W rezultacie Zabbix użyje następujących poświadczeń do uwierzytelniania w bazie danych:

- Nazwa użytkownika: <username>
- Hasło: <password>

Frontend

Aby uzyskać poświadczenia bazy danych z sejfu dla frontend Zabbix, podczas instalacji frontend należy określić następujące parametry.

1. W kroku *Configure DB Connection* ustaw parametr *Store credentials in* na „CyberArk Vault”.



2. Następnie uzupełnij dodatkowe parametry:

Parametr	Wymagany	Wartość domyślna	Opis
Vault API endpoint	tak	https://localhost:1858	Określ adres URL połączenia z sejfem w formacie `scheme://host:port`
Vault prefix	nie	/AIMWebService/api/Accounts?	Podaj niestandardowy prefiks dla ścieżki lub zapytania sejfu. Jeśli nie zostanie podany, użyta zostanie wartość domyślna.
Vault secret query string	tak		Zapytanie określające, skąd mają zostać pobrane poświadczenia bazy danych. Przykład: `AppID=foo&Query=Safe=bar;Object=buzz`

Parametr	Wymagany	Wartość domyślna	Opis
Vault certificates	nie		Po zaznaczeniu tego pola wyboru pojawią się dodatkowe parametry umożliwiające skonfigurowanie uwierzytelniania klienta. Chociaż ten parametr jest opcjonalny, zdecydowanie zaleca się jego włączenie do komunikacji z CyberArk Vault.
SSL certificate file	nie	conf/certs/cyberark-cert.pem	Ścieżka do pliku certyfikatu SSL. Plik musi być w formacie PEM. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw parametr SSL key file pusty.
SSL key file	nie	conf/certs/cyberark-key.pem	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik musi być w formacie PEM.

Wartości makr użytkownika

Aby używać CyberArk Vault do przechowywania wartości makr użytkownika typu *Vault secret*, upewnij się, że:

- serwer Zabbix jest skonfigurowany do pracy z CyberArk Vault;
- parametr *Vault provider* w *Administracja → Ogólne → Inne* jest ustawiony na „CyberArk Vault”.



Note:
Serwer Zabbix (oraz proxy Zabbix, jeśli jest skonfigurowany) wymaga dostępu do wartości makr typu *Vault secret* z sejfu. Frontend Zabbix nie potrzebuje takiego dostępu.

Wartość makra powinna zawierać zapytanie (w postaci `query:key`).

Szczegółowe informacje o przetwarzaniu wartości makr przez Zabbix można znaleźć w sekcji *Makra typu Vault secret*.

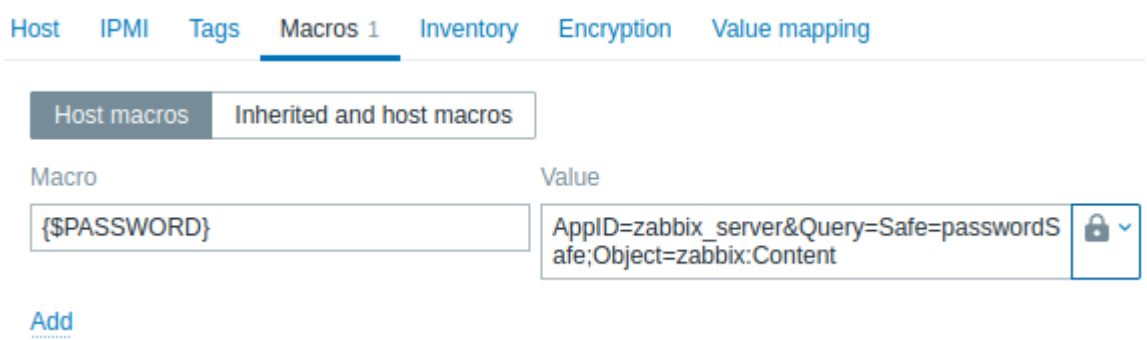
Składnia zapytania

Dwukropek (”:”) jest zarezerwowany do oddzielania zapytania od klucza.

Jeśli samo zapytanie zawiera ukośnik lub dwukropek, symbole te powinny być zakodowane w formacie URL (”/” jest kodowany jako ”%2F”, ”:” jest kodowany jako ”%3A”).

Przykład

1. W Zabbix dodaj makro użytkownika {$PASSWORD} typu *Vault secret* o wartości `AppID=zabbix_server&Query=Safe=passwordSaf`



2. Zabbix wyśle do sejfu następujące żądanie API:

```
curl \
--header "Content-Type: application/json" \
--cert cert.pem \
--key key.pem \
https://127.0.0.1:1858/AIMWebService/api/Accounts?AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbi
```

3. Odpowiedź sejfu będzie zawierać klucz ”Content”:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database" :<Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

4. W rezultacie Zabbix rozwiąże makro {$PASSWORD} do wartości - <password>

Aktualizacja istniejącej konfiguracji

Aby zaktualizować istniejącą konfigurację pobierania sekretów z CyberArk Vault:

1. Zaktualizuj parametry pliku konfiguracyjnego serwera Zabbix lub proxy zgodnie z opisem w sekcji *Dane logowania do bazy danych*.
2. Zaktualizuj ustawienia połączenia z bazą danych, rekonfigurując frontend Zabbix i podając wymagane parametry zgodnie z opisem w sekcji *Frontend*. Aby ponownie skonfigurować frontend Zabbix, otwórz w przeglądarce URL konfiguracji frontendu:

- dla Apache: http://<server_ip_or_name>/zabbix/setup.php
- dla Nginx: http://<server_ip_or_name>/setup.php

Alternatywnie parametry te można ustawić w pliku konfiguracyjnym frontendu (*zabbix.conf.php*):

```
$DB['VAULT']                    = 'CyberArk';
$DB['VAULT_URL']                = 'https://127.0.0.1:1858';
$DB['VAULT_DB_PATH']            = 'AppID=foo&Query=Safe=bar;Object=buzz';
$DB['VAULT_TOKEN']              = '';
$DB['VAULT_CERT_FILE']          = 'conf/certs/cyberark-cert.pem';
$DB['VAULT_KEY_FILE']           = 'conf/certs/cyberark-key.pem';
$DB['VAULT_PREFIX']             = '';
```

3. W razie potrzeby skonfiguruj makra użytkownika zgodnie z opisem w sekcji *Wartości makr użytkownika*.

Aby zaktualizować istniejącą konfigurację pobierania sekretów z HashiCorp Vault, zobacz *Konfiguracja HashiCorp*.

2 Konfiguracja HashiCorp

Przegląd

Ta sekcja wyjaśnia, jak skonfigurować Zabbix do pobierania sekretów z silnika sekretów HashiCorp Vault KV - wersja 2.

Vault powinien zostać wdrożony i skonfigurowany zgodnie z opisem w oficjalnej dokumentacji HashiCorp.

Aby dowiedzieć się więcej o konfigurowaniu TLS w Zabbix, zobacz *Przechowywanie sekretów*.

Pobieranie poświadczeń bazy danych

Aby pomyślnie pobrać sekret zawierający poświadczenia bazy danych, należy skonfigurować oba następujące elementy:

- serwer/proxy Zabbix
- frontend Zabbix

Serwer/proxy

Aby skonfigurować serwer lub proxy Zabbix, należy określić następujące parametry konfiguracyjne w pliku konfiguracyjnym:

- `Vault` — który dostawca vault ma być używany;
- `VaultToken` — token uwierzytelniania vault (szczegóły znajdują się w pliku konfiguracyjnym serwera/proxy Zabbix);
- `VaultURL` — adres URL HTTP[S] serwera vault;
- `VaultDBPath` — ścieżka do sekretu vault zawierającego poświadczenia bazy danych (tej opcji można użyć tylko wtedy, gdy nie określono DBUser i DBPassword); serwer lub proxy Zabbix pobierze poświadczenia przy użyciu kluczy „password” i „username”;
- `VaultPrefix` — niestandardowy prefiks ścieżki lub zapytania vault, zależnie od vault; jeśli nie zostanie określony, użyta zostanie najbardziej odpowiednia wartość domyślna.

Attention:
Parametry konfiguracyjne `Vault`, `VaultToken`, `VaultURL` i `VaultPrefix` są również używane do uwierzytelniania w vault podczas przetwarzania makr sekretów vault przez serwer Zabbix (oraz proxy Zabbix, jeśli jest skonfigurowany). Serwer Zabbix i proxy Zabbix nie będą otwierać makr sekretów vault zawierających poświadczenia DB z VaultDBPath.

Zdecydowanie zaleca się używanie różnych tokenów dla różnych proxy.

Serwer Zabbix i proxy Zabbix odczytują parametry konfiguracyjne związane z vault z plików *zabbix_server.conf* i *zabbix_proxy.conf* podczas uruchamiania. Dodatkowo serwer Zabbix i proxy Zabbix odczytają zmienną środowiskową `VAULT_TOKEN` jednokrotnie podczas uruchamiania, a następnie ją usuną, aby nie była dostępna dla skryptów potomnych; wystąpi błąd, jeśli zarówno parametr `VaultToken`, jak i `VAULT_TOKEN` zawierają wartość.

Przykład

1. W pliku *zabbix_server.conf* określ następujące parametry:

```
Vault=HashiCorp
VaultToken=hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTYONFNSaUp3ZnpWbDF1RUNjUkNTZEg
VaultURL=https://127.0.0.1:8200
VaultDBPath=database
VaultPrefix=/v1/secret/data/zabbix/
```

2. Uruchom następujące polecenia CLI, aby utworzyć wymagany sekret w vault:

```
#### Włącz punkt montowania "secret/", jeśli nie jest jeszcze włączony; zwróć uwagę, że musi być użyte "ku
vault secrets enable -path=secret/ kv-v2

#### Dodaj nowe sekrety z kluczami username i password w punkcie montowania "secret/" oraz pod ścieżką "za
vault kv put -mount=secret zabbix/database username=zabbix password=<password>

#### Sprawdź, czy sekret został pomyślnie dodany.
vault kv get secret/zabbix/database

#### Na koniec przetestuj za pomocą Curl; zwróć uwagę, że "data" należy dodać ręcznie po punkcie montowani
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/database
```

3. W rezultacie serwer Zabbix pobierze następujące dane uwierzytelniające do uwierzytelniania bazy danych:

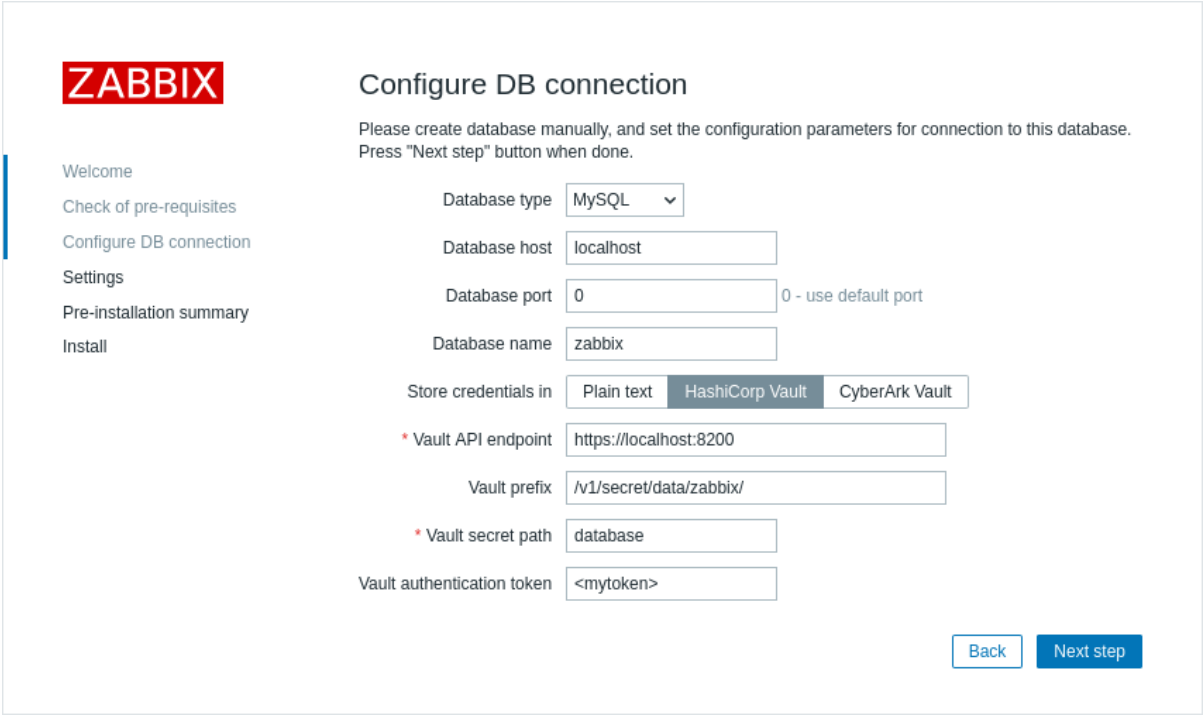
- Nazwa użytkownika: zabbix
- Hasło: <password>

Frontend

Frontend Zabbix może być skonfigurowany do pobierania poświadczeń bazy danych z vault podczas instalacji frontendu lub przez aktualizację pliku konfiguracyjnego frontendu (`zabbix.conf.php`).

Attention:
Jeśli poświadczenia vault zostały zmienione od czasu poprzedniej instalacji frontendu, uruchom ponownie instalację frontendu lub zaktualizuj `zabbix.conf.php`. Zobacz także: Aktualizacja istniejącej konfiguracji.

Podczas **instalacji frontendu** parametry konfiguracyjne muszą zostać określone na etapie *Configure DB Connection*:



- Ustaw parametr *Store credentials in* na „HashiCorp Vault”.
- Określ parametry połączenia:

Parametr	Obowiązkowy	Wartość domyślna	Opis
Vault API endpoint	tak	https://localhost:8200	Określ URL połączenia z vault w formacie `scheme://host:port`
Vault prefix	nie	/v1/secret/data/	Podaj niestandardowy prefiks dla ścieżki lub zapytania vault. Jeśli nie zostanie podany, użyta zostanie wartość domyślna. Przykład: `/v1/secret/data/zabbix/`
Vault secret path	nie		Ścieżka do sekretu, z którego poświadczenia bazy danych będą pobierane przy użyciu kluczy „password” i „username”. Przykład: `database`
Vault authentication token	nie		Podaj token uwierzytelniający do dostępu tylko do odczytu do ścieżki sekretu. Informacje o tworzeniu tokenów i polityk vault można znaleźć w dokumentacji HashiCorp.

Pobieranie wartości makr użytkownika

Aby używać HashiCorp Vault do przechowywania wartości makr użytkownika typu *Vault secret*, upewnij się, że:

- serwer/proxy Zabbix jest skonfigurowany do pracy z HashiCorp Vault;
- parametr *Vault provider* w *Administracja → Ogólne → Inne* jest ustawiony na „HashiCorp Vault” (domyślnie);



Note:
Serwer Zabbix (oraz proxy Zabbix, jeśli jest skonfigurowany) wymaga dostępu do wartości makr typu *Vault secret* z sejfu. Frontend Zabbix nie wymaga takiego dostępu.

Wartość makra powinna zawierać ścieżkę referencyjną (w postaci `path:key`, na przykład `macros:password`). Token uwierzytelniający określony podczas konfiguracji serwera/proxy Zabbix (parametrem `VaultToken`) musi zapewniać dostęp tylko do odczytu do tej ścieżki.

Szczegółowe informacje o przetwarzaniu wartości makr przez Zabbix można znaleźć w sekcji *Makra typu Vault secret*.

Składnia ścieżki

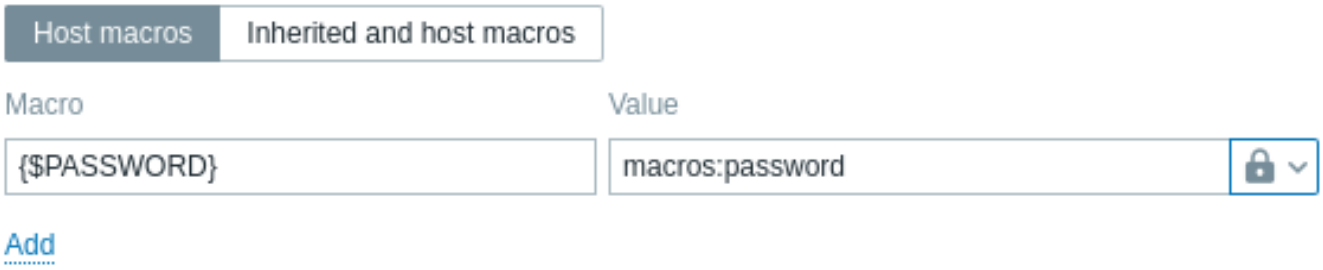
Symbole ukośnika (”/”) i dwukropka (”:”) są zastrzeżone.

Ukośnik może być używany wyłącznie do oddzielania punktu montowania od ścieżki (np. *secret/zabbix*, gdzie punkt montowania to ”secret”, a ścieżka to ”zabbix”). W przypadku makr Vault dwukropek może być używany wyłącznie do oddzielania ścieżki/zapytania od klucza.

Możliwe jest zakodowanie symboli ukośnika i dwukropka w formacie URL, jeśli istnieje potrzeba utworzenia punktu montowania o nazwie rozdzielonej ukośnikiem (np. *foo/bar/zabbix*, gdzie punkt montowania to ”foo/bar”, a ścieżka to ”zabbix”, można zakodować jako ”foo%2Fbar/zabbix”), a także jeśli nazwa punktu montowania lub ścieżka muszą zawierać dwukropek.

Przykład

1. W Zabbix dodaj makro użytkownika {$PASSWORD} typu ”Vault secret” o wartości `macros:password`



2. Uruchom następujące polecenia CLI, aby utworzyć wymagany sekret w vault:

```
#### Włącz punkt montowania "secret/", jeśli nie jest jeszcze włączony; zwróć uwagę, że musi zostać użyte
vault secrets enable -path=secret/ kv-v2

#### Umieść nowy sekret z kluczem "password" w punkcie montowania "secret/" i pod ścieżką "zabbix/macros".
vault kv put -mount=secret zabbix/macros password=<password>

#### Sprawdź, czy sekret został pomyślnie dodany.
vault kv get secret/zabbix/macros

#### Na koniec przetestuj za pomocą Curl; zwróć uwagę, że "data" należy dodać ręcznie po punkcie montowani
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/macros
```

3. W rezultacie Zabbix rozwiąże makro {$PASSWORD} do wartości: <password>

Aktualizacja istniejącej konfiguracji

Aby zaktualizować istniejącą konfigurację pobierania sekretów z HashiCorp Vault:

1. Zaktualizuj parametry pliku konfiguracyjnego serwera Zabbix lub proxy zgodnie z opisem w sekcji *Dane uwierzytelniające bazy danych*.
2. Zaktualizuj ustawienia połączenia z DB, rekonfigurując frontend Zabbix i podając wymagane parametry zgodnie z opisem w sekcji *Frontend*. Aby ponownie skonfigurować frontend Zabbix, otwórz w przeglądarce URL konfiguracji frontend:

- dla Apache: http://<server_ip_or_name>/zabbix/setup.php
- dla Nginx: http://<server_ip_or_name>/setup.php

Alternatywnie parametry te można ustawić w pliku konfiguracyjnym frontend (*zabbix.conf.php*):

```
$DB['VAULT']                    = 'HashiCorp';
$DB['VAULT_URL']                = 'https://localhost:8200';
$DB['VAULT_DB_PATH']            = 'database';
$DB['VAULT_TOKEN']              = '<mytoken>';
$DB['VAULT_CERT_FILE']          = '';
$DB['VAULT_KEY_FILE']           = '';
$DB['VAULT_PREFIX']             = '/v1/secret/data/zabbix/';
```

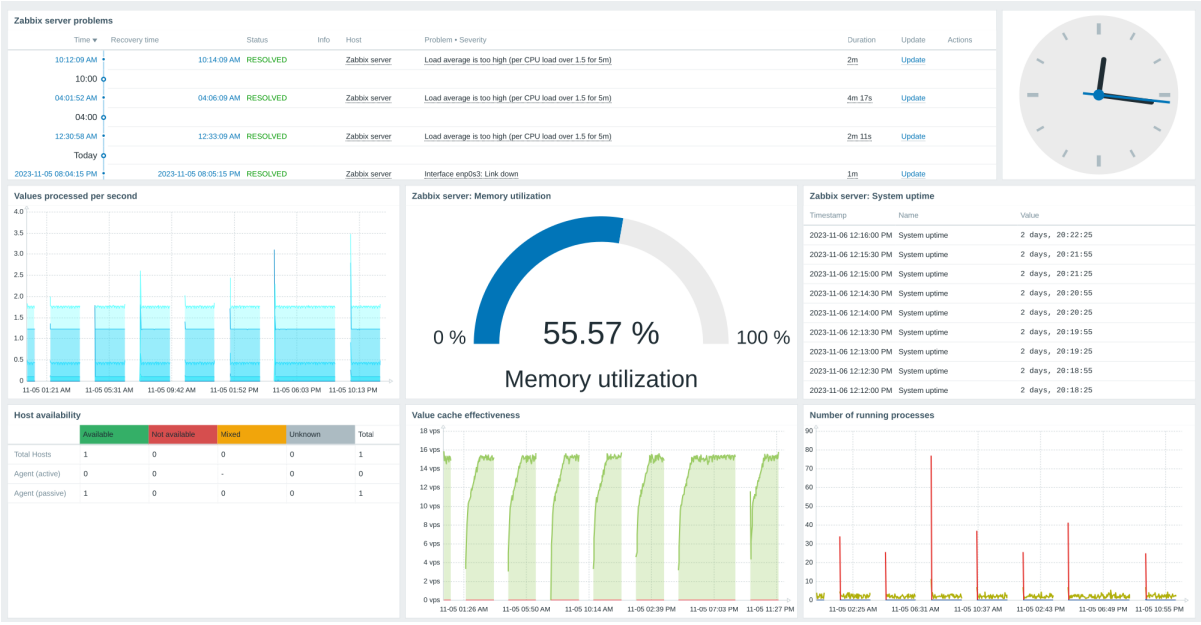
3. W razie potrzeby skonfiguruj makra użytkownika zgodnie z opisem w sekcji *Wartości makr użytkownika*.

Aby zaktualizować istniejącą konfigurację pobierania sekretów z CyberArk Vault, zobacz *Konfiguracja CyberArk*.

14 Zaplanowane raporty

Przegląd

Funkcja *Raporty cykliczne* umożliwia skonfigurowanie wysyłania wersji PDF wybranego dashboardu do określonych odbiorców w regularnych odstępach czasu.



Wymagania wstępne:

- Usługa internetowa Zabbix musi być zainstalowana i poprawnie skonfigurowana, aby umożliwić generowanie raportów cyklicznych — instrukcje znajdują się w Konfigurowanie raportów cyklicznych.
- Użytkownik musi mieć rolę użytkownika typu *Admin* lub *Super admin* z następującymi uprawnieniami:
  - *Raporty cykliczne* w sekcji *Dostęp do elementów interfejsu użytkownika* (aby wyświetlać ustawienia raportów)
  - *Zarządzanie raportami cyklicznymi* w sekcji *Dostęp do działań* (aby tworzyć/edytować raporty)

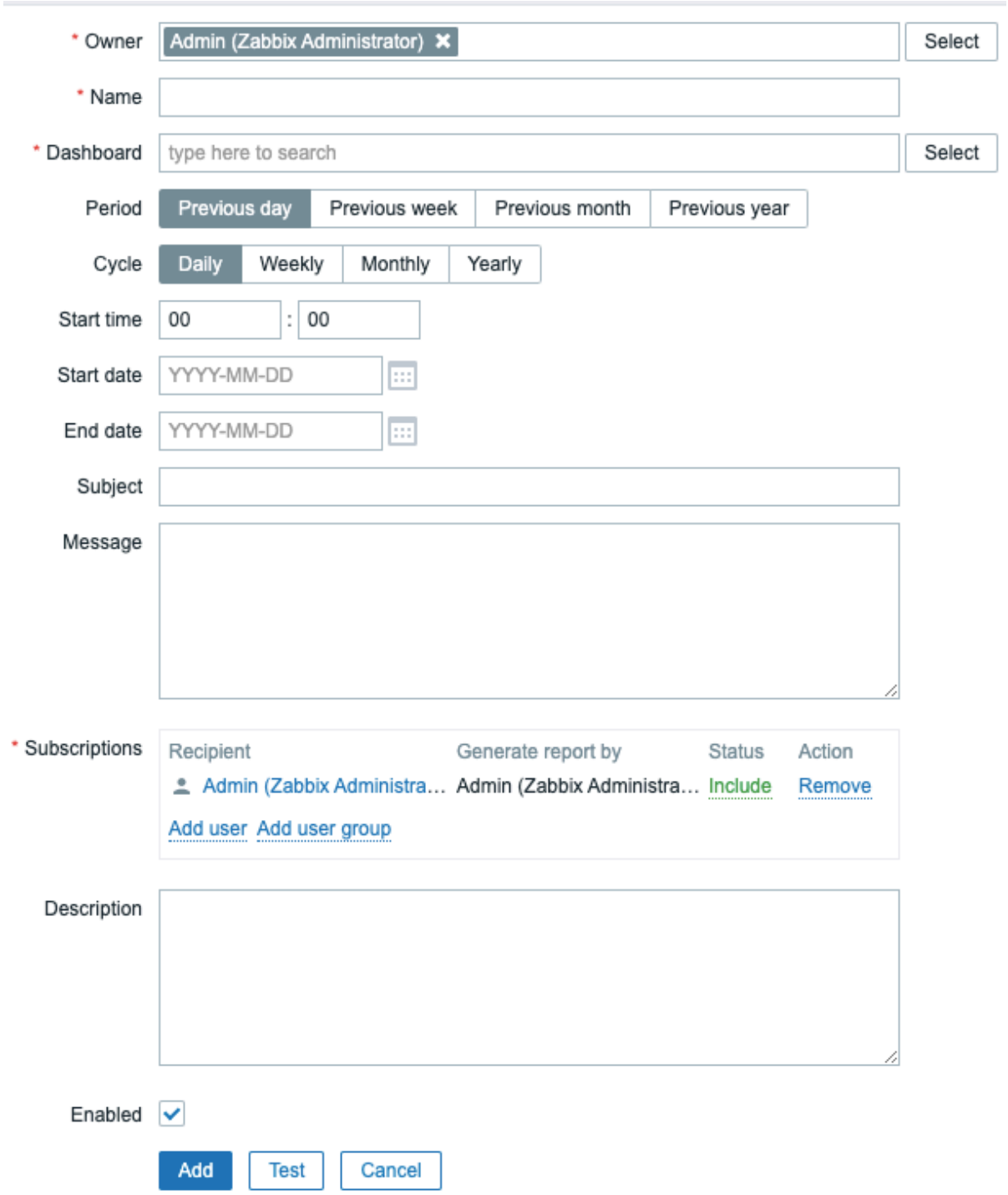
Aby utworzyć raport cykliczny w frontendzie Zabbix, wykonaj następujące czynności:

- Przejdź do: *Raporty* > *Raporty cykliczne*.
- Kliknij *Utwórz raport* w prawym górnym rogu ekranu.
- Wprowadź parametry raportu w formularzu.

Raport można również utworzyć, otwierając istniejący raport, klikając przycisk *Klonuj*, a następnie zapisując go pod inną nazwą.

Konfiguracja

Zakładka *Scheduled reports* zawiera ogólne atrybuty raportu.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Owner	Użytkownik tworzący raport. Użytkownicy z poziomem *Super admin* mogą zmieniać właściciela. Dla użytkowników z poziomem *Admin* to pole jest tylko do odczytu.
Name	Nazwa raportu; musi być unikalna.
Dashboard	Pulpit, na podstawie którego tworzony jest raport; jednocześnie można wybrać tylko jeden pulpit. Aby wybrać pulpit, zacznij wpisywać nazwę — pojawi się lista pasujących pulpitów; przewiń w dół, aby wybrać. Alternatywnie możesz kliknąć *Select* obok pola i wybrać pulpit z wyświetlonej listy.
Period	Okres, dla którego raport zostanie przygotowany. Wybierz poprzedni dzień, tydzień, miesiąc lub rok.
Cycle	Częstotliwość generowania raportu. Raporty mogą być wysyłane codziennie, co tydzień, co miesiąc lub co rok. Tryb „Weekly” umożliwia wybór dni tygodnia, w których raport będzie wysyłany.
Start time	Pora dnia w formacie hh:mm, o której raport zostanie przygotowany. Zwróć uwagę, że zostanie użyta strefa czasowa serwera Zabbix.

Parametr	Opis
Repeat on	Dni tygodnia, w których raport będzie wysyłany. To pole jest dostępne tylko wtedy, gdy *Cycle* jest ustawione na „Weekly”.
Start date	Data rozpoczęcia regularnego generowania raportu.
End date	Data zakończenia regularnego generowania raportu.
Subject	Temat wiadomości e-mail z raportem. Obsługiwane makra: {TIME}, {TIMESTAMP}.
Message	Treść wiadomości e-mail z raportem. Obsługiwane makra: {TIME}, {TIMESTAMP}.
Subscriptions	Lista odbiorców raportu. Domyślnie zawiera tylko właściciela raportu. Jako odbiorcę raportu można wskazać dowolnego użytkownika Zabbix ze skonfigurowanym medium e-mail. Kliknij *Add user* lub *Add user group*, aby dodać kolejnych odbiorców. Kliknij nazwę użytkownika, aby edytować ustawienia: *Generate report by* — określa, czy dane raportu mają być generowane na podstawie uprawnień do pulpitu bieżącego użytkownika czy odbiorcy. *Status* — wybierz „Include”, aby wysłać raport do użytkownika, lub „Exclude”, aby zapobiec wysłaniu raportu do tego użytkownika. Co najmniej jeden użytkownik musi mieć status „Include”. Status „Exclude” może być użyty do wykluczenia określonych użytkowników z uwzględnionej grupy użytkowników. Zwróć uwagę, że użytkownicy z niewystarczającymi uprawnieniami (to znaczy użytkownicy z rolą opartą na typie użytkownika *Admin*, którzy nie są członkami tej samej grupy użytkowników co odbiorca lub właściciel raportu) zobaczą „Inaccessible user” lub „Inaccessible user group” zamiast rzeczywistych nazw w polach *Recipient* i *Generate report by*; pola *Status* i *Action* będą wyświetlane jako tylko do odczytu.
Enabled	Status raportu. Odznaczenie tego pola wyboru wyłączy raport.
Description	Opcjonalny opis raportu. Ten opis jest przeznaczony do użytku wewnętrznego i nie będzie wysyłany do odbiorców raportu.

Przyciski formularza

Przyciski na dole formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj raport. Ten przycisk jest dostępny tylko dla nowych raportów.
Update	Zaktualizuj właściwości raportu.
Clone	Utwórz kolejny raport na podstawie właściwości bieżącego raportu.
Test	Sprawdź, czy konfiguracja raportu jest poprawna, wysyłając raport do bieżącego użytkownika.
Delete	Usuń raport.
Cancel	Anuluj edycję właściwości raportu.

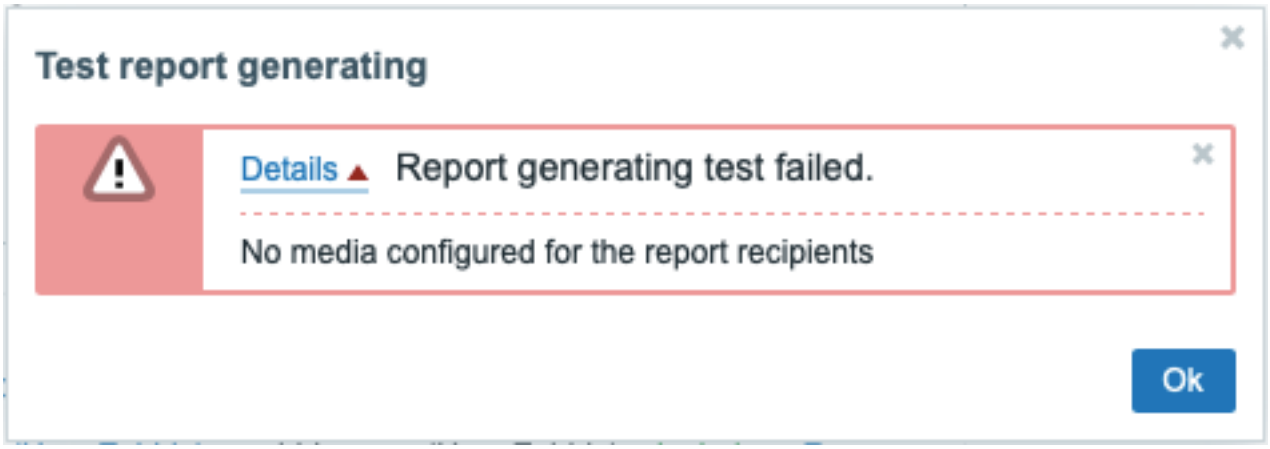
Testowanie

Aby przetestować raport, kliknij przycisk *Test* na dole formularza konfiguracji raportu.

Note:
Przycisk *Test* nie jest dostępny, jeśli formularz konfiguracji raportu został otwarty z menu akcji pulpitu.

Jeśli konfiguracja jest poprawna, raport testowy jest natychmiast wysyłany do bieżącego użytkownika. W przypadku raportów testowych subskrybenci oraz ustawienia użytkownika *Generate report by* są ignorowane.

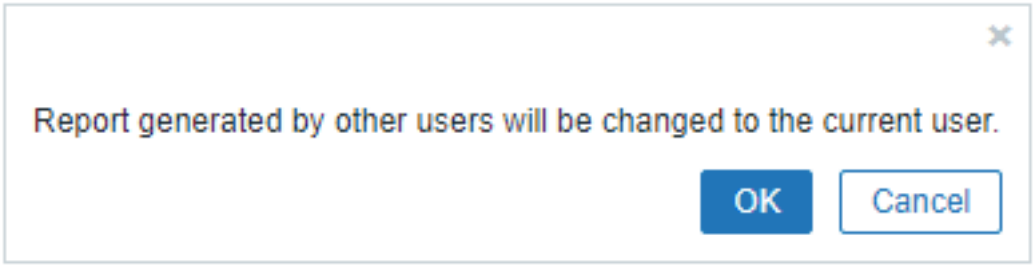
Jeśli konfiguracja jest niepoprawna, wyświetlany jest komunikat o błędzie opisujący możliwą przyczynę.



Aktualizacja raportu

Aby zaktualizować istniejący raport, kliknij nazwę raportu, wprowadź wymagane zmiany konfiguracyjne, a następnie kliknij przycisk *Update*.

Jeśli istniejący raport zostanie zaktualizowany przez innego użytkownika i ten użytkownik zmieni Dashboard, po kliknięciu przycisku *Update* zostanie wyświetlony komunikat ostrzegawczy „Raport generowany przez innych użytkowników zostanie zmieniony na bieżącego użytkownika".



Kliknięcie *OK* na tym etapie spowoduje następujące zmiany:

- Ustawienia *Generate report by* zostaną zaktualizowane tak, aby wyświetlać użytkownika, który edytował raport jako ostatni (chyba że *Generate report by* jest ustawione na odbiorcę).
- Użytkownicy wyświetlani jako „Inaccessible user" lub „Inaccessible user group" zostaną usunięci z listy subskrybentów raportu.

Kliknięcie *Cancel* spowoduje zamknięcie formularza konfiguracji i anulowanie aktualizacji raportu.

Klonowanie raportu

Aby szybko sklonować istniejący raport, kliknij przycisk *Clone* u dołu formularza konfiguracji istniejącego raportu. W przypadku klonowania raportu utworzonego przez innego użytkownika bieżący użytkownik staje się właścicielem nowego raportu.

Ustawienia raportu zostaną skopiowane do nowego formularza konfiguracji raportu z uwzględnieniem uprawnień użytkownika:

- Jeśli użytkownik klonujący raport nie ma uprawnień do pulpitu, pole *Dashboard* zostanie wyczyszczone.
- Jeśli użytkownik klonujący raport nie ma uprawnień do niektórych użytkowników lub grup użytkowników na liście *Subscriptions*, niedostępni odbiorcy nie zostaną sklonowani.
- Ustawienia *Generate report by* zostaną zaktualizowane tak, aby wyświetlać bieżącego użytkownika (chyba że w polu *Generate report by* ustawiono odbiorcę).

Zmień wymagane ustawienia i nazwę raportu, a następnie kliknij *Add*.

15 Eksport danych

Przegląd

Zabbix obsługuje eksport danych w czasie rzeczywistym na dwa sposoby:

- eksport do plików
- przesyłanie strumieniowe do systemów zewnętrznych

Można eksportować następujące encje:

- zdarzenia wyzwalaczy
- wartości pozycji
- trendy (eksport tylko do plików)

1 Eksport do plików

Omówienie

Możliwe jest skonfigurowanie eksportu w czasie rzeczywistym zdarzeń wyzwalaczy, wartości pozycji i trendów w formacie JSON z rozdzielaniem po nowych liniach.

Eksport jest wykonywany do plików, gdzie każda linia pliku eksportu jest obiektem JSON. Mapowania wartości nie są stosowane.

W przypadku błędów (danych nie można zapisać do pliku eksportu albo pliku eksportu nie można zmienić nazwy lub nie można utworzyć nowego po zmianie nazwy), element danych jest odrzucany i nigdy nie jest zapisywany do pliku eksportu. Jest zapisywany wyłącznie w bazie danych Zabbix. Zapisywanie danych do pliku eksportu zostaje wznowione po rozwiązaniu problemu z zapisem.

Od Zabbix 7.4.6 plik eksportu jest tworzony z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są zabronione.

Szczegółowe informacje o tym, jakie dane są eksportowane, można znaleźć na stronie protokół eksportu.

Należy pamiętać, że host/pozycja może nie mieć metadanych (grupy hostów, nazwy hosta, nazwy pozycji), jeśli host/pozycja została usunięta po odebraniu danych, ale przed wyeksportowaniem danych przez serwer.

Konfiguracja

Eksport w czasie rzeczywistym zdarzeń wyzwalaczy, wartości pozycji i trendów jest konfigurowany przez określenie katalogu dla plików eksportu — zobacz parametr ExportDir w konfiguracji serwera.

Dostępne są jeszcze dwa inne parametry:

- `ExportFileSize` może być użyty do ustawienia maksymalnego dopuszczalnego rozmiaru pojedynczego pliku eksportu. Gdy proces musi zapisać dane do pliku, najpierw sprawdza jego rozmiar. Jeśli przekracza on skonfigurowany limit rozmiaru, plik jest zmieniany przez dodanie do jego nazwy rozszerzenia .old, a następnie tworzony jest nowy plik z oryginalną nazwą.

> **Attention:**
> Dla każdego procesu zapisującego dane zostanie utworzony osobny plik (tj. około 4–30 plików). Ponieważ domyślny rozmiar pojedynczego pliku eksportu wynosi 1G, przechowywanie dużych plików eksportu może szybko doprowadzić do wyczerpania miejsca na dysku.

- `ExportType` pozwala określić, które typy encji (zdarzenia, historia, trendy) będą eksportowane.

2 Strumieniowanie do systemów zewnętrznych

Przegląd

Możliwe jest strumieniowanie wartości pozycji i zdarzeń z Zabbix do zewnętrznych systemów przez HTTP (zobacz szczegóły protokołu).

Filtr tagów może być używany do strumieniowania podzbiorów wartości pozycji lub zdarzeń.

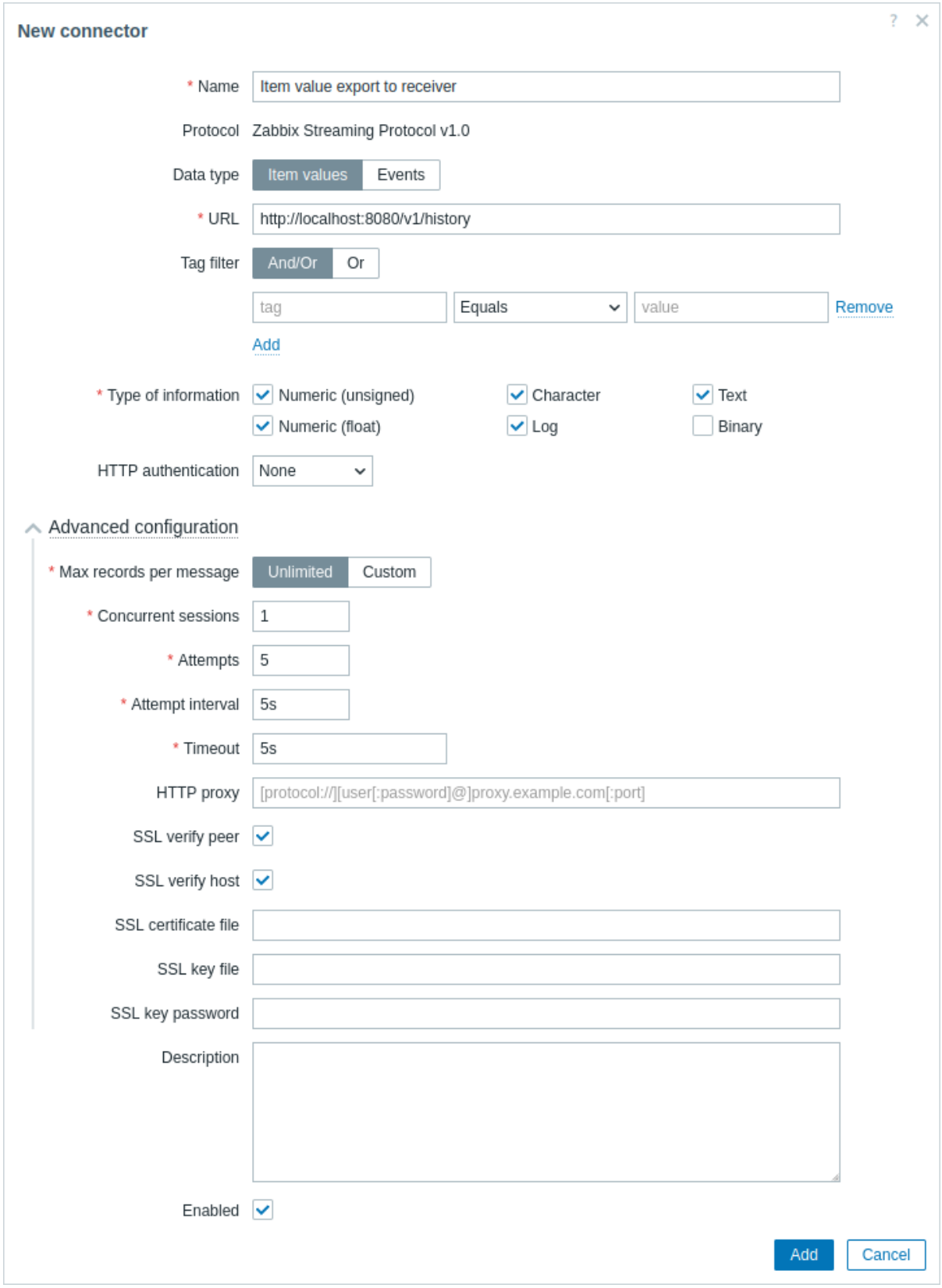
Za strumieniowanie danych odpowiadają dwa typy procesów serwera Zabbix: `connector manager` i `connector worker`. Wewnętrzna pozycja Zabbix `zabbix[connector_queue]` umożliwia monitorowanie liczby wartości umieszczonych w kolejce connector.

Konfiguracja

Aby skonfigurować strumieniowanie danych do systemu zewnętrznego, wymagane są następujące kroki:

1. Przygotuj zdalny system do odbierania danych z Zabbix. W tym celu dostępne są następujące narzędzia:
   - Przykład prostego odbiornika, który zapisuje odebrane informacje w plikach `events.ndjson` i `history.ndjson`.
   - Kafka connector for Zabbix server — lekki serwer napisany w Go, przeznaczony do przekazywania wartości pozycji i zdarzeń z serwera Zabbix do brokera Kafka.
2. Ustaw wymaganą liczbę procesów roboczych connectora w Zabbix, dostosowując parametr `StartConnectors` w pliku `zabbix_server.conf`. Liczba procesów roboczych connectora powinna odpowiadać (lub być większa, jeśli liczba współbieżnych sesji jest większa niż 1) skonfigurowanej liczbie connectorów w Zabbix frontend. Następnie uruchom ponownie serwer Zabbix.

3. Skonfiguruj nowy connector w Zabbix frontend (*Administracja* > *Ogólne* > *Connectors*) i przeładuj pamięć podręczną konfiguracji serwera poleceniem `zabbix_server -R config_cache_reload`.



Pola obowiązkowe są oznaczone gwiazdką.

Parametr	Opis
Name	Wprowadź nazwę connectora.
Data type	Wybierz typ danych do strumieniowania: **Item values** — strumieniowanie wartości pozycji z Zabbix do systemów zewnętrznych; **Events** — strumieniowanie zdarzeń z Zabbix do systemów zewnętrznych.
URL	Wprowadź adres URL odbiornika. Obsługiwane są makra użytkownika.

Parametr	Opis
Tag filter	Eksportuj tylko wartości pozycji lub zdarzenia pasujące do filtra tagów. Jeśli nie ustawiono, eksportowane będzie wszystko. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** — uwzględnij określone nazwy tagów; **Equals** — uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Contains** — uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** — wyklucz określone nazwy tagów; **Does not equal** — wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Does not contain** — wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** — wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** — wystarczy spełnienie jednego warunku.
Type of information	Wybierz typ informacji (numeric (unsigned), numeric (float), character itp.), według którego mają być filtrowane wartości pozycji strumieniowane przez connector. To pole jest dostępne, jeśli *Data type* jest ustawione na ”Item values”.
HTTP authentication	Wybierz opcję uwierzytelniania: **None** — bez uwierzytelniania; **Basic** — używane jest uwierzytelnianie podstawowe; **NTLM** — używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** — używane jest uwierzytelnianie Kerberos (zobacz także: Configuring Kerberos with Zabbix); **Digest** — używane jest uwierzytelnianie Digest; **Bearer** — używane jest uwierzytelnianie Bearer.
Username	Wprowadź nazwę użytkownika (do 255 znaków). Obsługiwane są makra użytkownika. To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na ”Basic”, ”NTLM”, ”Kerberos” lub ”Digest”.
Password	Wprowadź hasło użytkownika (do 255 znaków). Obsługiwane są makra użytkownika. To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na ”Basic”, ”NTLM”, ”Kerberos” lub ”Digest”.
Bearer token	Wprowadź token Bearer. Obsługiwane są makra użytkownika. To pole jest dostępne i wymagane, jeśli *HTTP authentication* jest ustawione na ”Bearer”.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji (patrz poniżej).
Max records per message	Określ maksymalną liczbę wartości lub zdarzeń, które mogą być strumieniowane w jednej wiadomości.
Concurrent sessions	Wybierz liczbę procesów nadawczych uruchamianych dla tego connectora. Można określić do 100 sesji; wartość domyślna to ”1”.
Attempts	Liczba prób strumieniowania danych. Można określić do 5 prób; wartość domyślna to ”1”.
Attempt interval	Określ, jak długo connector powinien czekać po nieudanej próbie strumieniowania danych. Można określić do 10s; wartość domyślna to ”5s”. To pole jest dostępne, jeśli *Attempts* jest ustawione na ”2” lub więcej. Nieudane próby to takie, w których nie udało się nawiązać połączenia lub kod odpowiedzi HTTP nie jest równy 200, 201, 202, 203, 204. Ponowne próby są **wyzwalane** w przypadku błędów komunikacji lub gdy kod odpowiedzi HTTP nie jest równy 200, 201, 202, 203, 204, 400, 401, 403, 404, 405, 415, 422. Przekierowania są obsługiwane, więc 302 -> 200 jest odpowiedzią pozytywną, natomiast 302 -> 503 spowoduje ponowną próbę.
Timeout	Określ limit czasu wiadomości (1-60 sekund, domyślnie — 5 sekund). Obsługiwane są sufiksy czasu, np. 30s, 1m. Obsługiwane są makra użytkownika.

Parametr	Opis
HTTP proxy	Możesz określić HTTP proxy do użycia w następującym formacie: `[protocol://][username[:password]@]proxy.example.com[:port]` Obsługiwane są makra użytkownika. Opcjonalny prefiks `protocol://` może być użyty do określenia alternatywnych protokołów proxy (obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli nie określono protokołu, proxy będzie traktowane jako HTTP proxy. Domyślnie użyty zostanie port 1080. Jeśli określono *HTTP proxy*, proxy nadpisze zmienne środowiskowe związane z proxy, takie jak `http_proxy`, `HTTPS_PROXY`. Jeśli nie określono, proxy nie nadpisze zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana bez zmian, bez sprawdzania poprawności. Możesz również wprowadzić adres proxy SOCKS. Jeśli określisz niewłaściwy protokół, connector nie będzie w stanie strumieniować wartości pozycji lub zdarzeń z Zabbix. Należy pamiętać, że w przypadku HTTP proxy obsługiwane jest tylko proste uwierzytelnianie.
SSL verify peer	Zaznacz pole wyboru, aby weryfikować certyfikat SSL serwera WWW. Certyfikat serwera zostanie automatycznie pobrany z systemowej lokalizacji urzędu certyfikacji (CA). Możesz nadpisać lokalizację plików CA za pomocą parametru konfiguracyjnego serwera Zabbix lub proxy `SSLCALocation`.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub pole *Subject Alternate Name* certyfikatu serwera WWW jest zgodne. Ustawia to opcję cURL `CURLOPT_SSL_VERIFYHOST`.
SSL certificate file	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Obsługiwane są makra użytkownika. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole *SSL key file* puste. Jeśli klucz jest zaszyfrowany, podaj hasło w polu *SSL key password*. Katalog zawierający ten plik jest określany przez parametr konfiguracyjny serwera Zabbix lub proxy `SSLCertLocation`.
SSL key file	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Obsługiwane są makra użytkownika. Katalog zawierający ten plik jest określany przez parametr konfiguracyjny serwera Zabbix lub proxy `SSLKeyLocation`.
SSL key password	Hasło do pliku klucza prywatnego SSL. Obsługiwane są makra użytkownika.
Description	Wprowadź opis connectora.
Enabled	Zaznacz pole wyboru, aby włączyć connector.

Gdy connector Kafka jest skonfigurowany z rozdzieloną przecinkami listą adresów brokerów bootstrap (na przykład `Kafka.URL=kafka1.exam`
klient Kafka łączy się z brokerem (brokerami), które odpowiedzą jako pierwsze, i używa ich metadanych klastra. Jeśli lista zawiera adresy z różnych klastrów Kafka, użyty zostanie tylko klaster odpowiadający najszybciej, a pozostałe adresy zostaną zapisane w logu jako niedostępne; w rezultacie podczas uruchamiania mogą pojawić się ostrzeżenia, takie jak poniższe, mimo że connector jest połączony:

```
kafka cluster connected, but broker(s) "kafka1.example.com:9093, kafka2.example.com:9093" unavailable; wil
```

W niektórych środowiskach (sieci prywatne, sieci kontenerowe lub niestandardowe konfiguracje DNS/hosts) nazwy hostów lub adresy IP mogą być rozwiązywane do adresów pętli zwrotnej (na przykład `127.0.0.1/localhost`) albo normalizowane przez klienta, co może sprawiać, że takie ostrzeżenia będą mylące. Aby ograniczyć nieporozumienia, upewnij się, że wszystkie adresy `Kafka.URL` należą do tego samego klastra Kafka, zweryfikuj rozwiązywanie DNS z hosta connectora oraz ustawienie `advertised.listeners` brokerów, a także preferuj adresy, które są rozwiązywane do adresu rozgłaszanego przez brokera.

Protokół

Komunikacja między serwerem a odbiorcą odbywa się przez HTTP z użyciem REST API, NDJSON, „Content-Type: application/x-ndjson".

Więcej informacji można znaleźć w protokole eksportu JSON rozdzielanego znakami nowej linii.

Żądanie serwera

Przykład strumieniowania wartości pozycji:

```
POST /v1/history HTTP/1.1
Host: localhost:8080
Accept: */*
```

```
Accept-Encoding: deflate, gzip, br, zstd
Content-Length: 628
Content-Type: application/x-ndjson

{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"f
{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"f
{"host":{"host":"Zabbix server","name":"Zabbix server"},"groups":["Zabbix servers"],"item_tags":[{"tag":"b
```

Przykład strumieniowania zdarzeń:

```
POST /v1/events HTTP/1.1
Host: localhost:8080
Accept: */*
Accept-Encoding: deflate, gzip, br, zstd
Content-Length: 333
Content-Type: application/x-ndjson

{"clock":1673454303,"ns":800155804,"value":1,"eventid":5,"name":"trigger for foo being 0","severity":0,"ho
{"clock":1673454303,"ns":832290669,"value":0,"eventid":6,"p_eventid":5}
```

Odpowiedź odbiorcy

Odpowiedź składa się z kodu statusu odpowiedzi HTTP oraz ładunku JSON. Kod statusu odpowiedzi HTTP musi mieć wartość ”200”, ”201”, ”202”, ”203” lub ”204” dla żądań, które zostały obsłużone pomyślnie, inny dla żądań zakończonych niepowodzeniem.

Przykład powodzenia:

```
HTTP/1.1 200 OK
Content-Type: application/json
X-Content-Type-Options: nosniff
Date: Tue, 21 Apr 2026 10:13:04 GMT
Content-Length: 23

{"response":"success"}
```

Przykład z błędami:

```
HTTP/1.1 422 Unprocessable Entity
Content-Type: application/json
X-Content-Type-Options: nosniff
Date: Tue, 21 Apr 2026 12:15:01 GMT
Content-Length: 55

{"error":"invalid character '{' after top-level value"}
```

3 Brama SNMP

Przegląd

Zabbix SNMP gateway to rozszerzenie AgentX dla snmpd, obsługujące zarówno odpytywanie SNMP, jak i odbieranie trapów.

Dzięki Zabbix SNMP gateway można używać protokołu SNMP do pobierania:

- danych wyzwalacza;
- danych wyzwalacza problemu;
- stanu grup hostów (liczby wyzwalaczy według stanu wyzwalacza w grupie)

Dane są pobierane za pomocą OID, który jest kombinacją wspólnej podstawy i określonego sufiksu. Wspólna **podstawa** jest ustawiana w pliku konfiguracyjnym SNMP gateway, na przykład:

- `BaseOID=1.3.6.1.4.1.3043.7.55` - dla dowolnych danych wyzwalacza;
- `ProblemBaseOID=1.3.6.1.4.1.3047.7.55` - dla danych wyzwalacza problemu;
- `BaseOID=1.3.6.1.4.1.3046.7.55` - dla stanu grup hostów.

Sufiks OID jest ustawiany w konfiguracji wyzwalaczy hosta jako **tag** (na przykład `OIDSuffix:3`) w frontend.

W takim przypadku wszystkie informacje dotyczące wyzwalacza będą dostępne pod `OID=1.3.6.1.4.1.3043.7.55.X.3`. „X” oznacza tutaj liczbę pól danych wyzwalacza (tj. 1 - sufiks, 2 - ID, 3 - wyrażenie, 4 - opis itd.).

Bardziej szczegółowy opis oraz przykład pliku konfiguracyjnego można znaleźć w pliku readme SNMP gateway.

Instalacja i konfiguracja

Instrukcje dotyczące:

- instalacji i konfiguracji snmpd;
- włączania obsługi AgentX;
- konfiguracji bramy SNMP Zabbix;
- konfiguracji pułapek SNMP dla zmian stanu wyzwalacza

znajdują się w pliku readme bramy SNMP.

Pobieranie danych

Po poprawnym skonfigurowaniu wszystkiego możesz użyć poleceń `snmpwalk` i `snmpget`, aby pobrać dane:

```
[user@localhost ~]# snmpget -v2c -c public 127.0.0.1 1.3.6.1.4.1.3043.7.55.2.3
SNMPv2-SMI::enterprises.3043.7.55.2.3 = INTEGER: 15247
```

```
[user@localhost ~]# snmpwalk -v2c -c public 127.0.0.1 1.3.6.1.4.1.3043.7.55
SNMPv2-SMI::enterprises.3043.7.55.1.1 = INTEGER: 1
SNMPv2-SMI::enterprises.3043.7.55.1.3 = INTEGER: 3
SNMPv2-SMI::enterprises.3043.7.55.1.4 = INTEGER: 4
SNMPv2-SMI::enterprises.3043.7.55.1.5 = INTEGER: 5
SNMPv2-SMI::enterprises.3043.7.55.1.6 = INTEGER: 6
SNMPv2-SMI::enterprises.3043.7.55.1.10 = INTEGER: 10
SNMPv2-SMI::enterprises.3043.7.55.2.1 = INTEGER: 15367
SNMPv2-SMI::enterprises.3043.7.55.2.3 = INTEGER: 15247
SNMPv2-SMI::enterprises.3043.7.55.2.4 = INTEGER: 15365
SNMPv2-SMI::enterprises.3043.7.55.2.5 = INTEGER: 15366
SNMPv2-SMI::enterprises.3043.7.55.2.6 = INTEGER: 13493
SNMPv2-SMI::enterprises.3043.7.55.2.10 = INTEGER: 13503
...
```

Opcje filtrowania

Możesz ograniczyć informacje o problemach wyzwalaczy w konfiguracji bramy SNMP:

- według ważności (domyślnie `ProblemMinSeverity=-1`)
- przez ukrywanie potwierdzonych problemów (domyślnie `ProblemHideAck=false`)

Możesz ograniczyć liczbę problemów na grupę hostów w konfiguracji bramy SNMP:

- według wyzwalaczy w stanie nieznanym (domyślnie `CountUnknown=false`)
- według wyzwalaczy z potwierdzonymi/niepotwierdzonymi/wszystkimi problemami (domyślnie `CountAcknowledgeStatus=all`)

8 Monitorowanie usług

Przegląd Monitorowanie usług to monitorowanie na poziomie biznesowym, które można wykorzystać do uzyskania przeglądu całego drzewa usług infrastruktury IT, identyfikowania słabych punktów infrastruktury, obliczania SLA dla różnych usług IT oraz sprawdzania innych informacji na wyższym poziomie. Monitorowanie usług koncentruje się na ogólnej dostępności usługi zamiast na szczegółach niskiego poziomu, takich jak brak miejsca na dysku, wysokie obciążenie procesora itp. Monitorowanie usług zapewnia również funkcjonalność pozwalającą znaleźć przyczynę źródłową problemu, jeśli usługa nie działa zgodnie z oczekiwaniami.

Monitorowanie usług umożliwia tworzenie hierarchicznej reprezentacji monitorowanych danych.

Bardzo prosta struktura usług może wyglądać następująco:

```
Service
|
|-Workstations
| |
| |-Workstation1
| |
| |-Workstation2
|
|-Servers
```

Każdy węzeł struktury ma atrybut status. Status jest obliczany i propagowany do wyższych poziomów zgodnie z wybranym algorytmem. Na status poszczególnych węzłów wpływa status powiązanych problemów. Mapowanie problemów odbywa się za pomocą tagowania.

Zabbix może wysyłać powiadomienia lub automatycznie wykonywać skrypt na serwerze Zabbix w przypadku wykrycia zmiany statusu usługi. Możliwe jest zdefiniowanie elastycznych reguł określających, czy usługa nadrzędna powinna przejść w stan 'Problem state' na podstawie statusów usług podrzędnych. Dane o problemach usług mogą być następnie używane do obliczania SLA i wysyłania raportów SLA na podstawie elastycznego zestawu warunków.

Monitorowanie usług jest konfigurowane w menu Services, które składa się z następujących sekcji:

- Services

Sekcja Services umożliwia budowanie hierarchii monitorowanej infrastruktury poprzez dodawanie usług nadrzędnych, a następnie - usług podrzędnych do usług nadrzędnych.

Oprócz konfiguracji drzewa usług, ta sekcja zapewnia przegląd całej infrastruktury i umożliwia szybkie zidentyfikowanie problemów, które doprowadziły do zmiany statusu usługi.

- SLA

W tej sekcji można definiować umowy SLA oraz ustawiać cele poziomu usług dla określonych usług.

- SLA report

W tej sekcji można wyświetlać raporty SLA.

Akcje usług

Można również skonfigurować akcje usług.

Akcje usług są opcjonalne i umożliwiają:

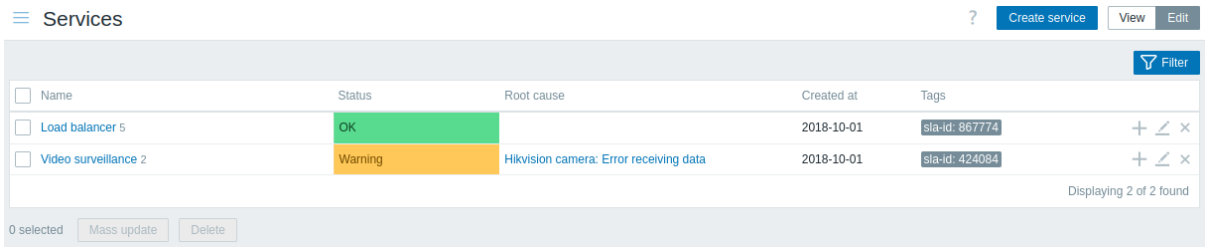
- wysłanie powiadomienia, że usługa jest niedostępna
- wykonanie zdalnego polecenia na serwerze Zabbix po zmianie statusu usługi
- wysłanie powiadomienia o odzyskaniu, gdy usługa ponownie zacznie działać.

Zobacz także:

- Przykład konfiguracji monitorowania SLA example
- Uwagi dotyczące aktualizacji usług z wersji Zabbix poniżej 6.0

1 Drzewo usług

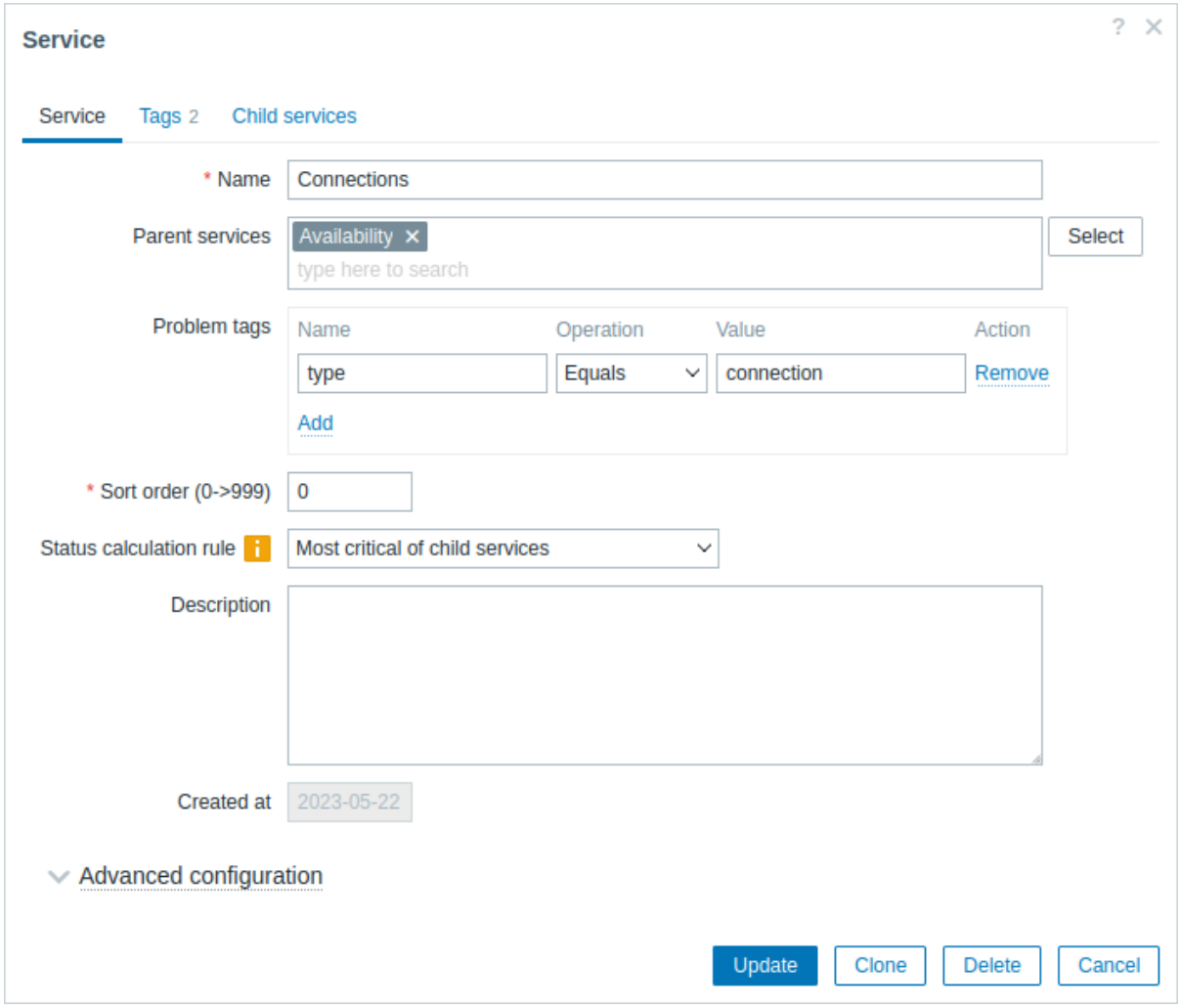
Drzewo usług jest konfigurowane w sekcji menu *Services* -> *Services*. W prawym górnym rogu przełącz z trybu View do trybu edycji.



Aby skonfigurować nową usługę, kliknij przycisk *Create service* w prawym górnym rogu.

Aby szybko dodać usługę podrzędną, możesz też nacisnąć ikonę plusa obok usługi nadrzędnej. Spowoduje to otwarcie tego samego formularza konfiguracji usługi, ale parametr *Parent services* będzie wstępnie wypełniony.

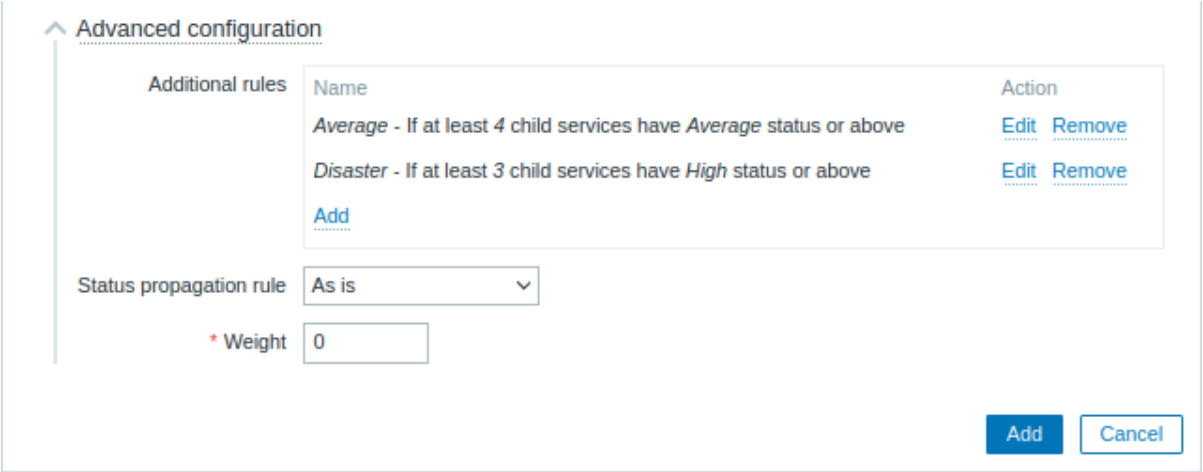
Konfiguracja usługi Na karcie **Usługa** określ wymagane parametry usługi:



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Nazwa usługi.
Usługi nadrzędne	Usługi nadrzędne, do których należy usługa. Pozostaw to pole puste, jeśli dodajesz usługę najwyższego poziomu. Jedna usługa może mieć wiele usług nadrzędnych. W takim przypadku będzie wyświetlana w drzewie usług pod każdą z usług nadrzędnych.
Tagi problemów	Określ tagi, aby mapować dane problemów do usługi. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są dwa operatory: **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, z uwzględnieniem wielkości liter)
Kolejność sortowania	Kolejność sortowania na potrzeby wyświetlania, od najniższej do najwyższej.
Reguła obliczania statusu	Reguła obliczania statusu usługi: **Najbardziej krytyczny, jeśli wszystkie usługi podrzędne mają problemy** - najbardziej krytyczny problem w usługach podrzędnych jest używany do określenia koloru statusu usługi, jeśli wszystkie usługi podrzędne mają problemy **Najbardziej krytyczny spośród usług podrzędnych** - najbardziej krytyczny problem w usługach podrzędnych jest używany do określenia koloru statusu usługi **Ustaw status na OK** - nie obliczaj statusu usługi Dodatkowe reguły obliczania statusu można skonfigurować w opcjach konfiguracji zaawansowanej.
Opis	Opis usługi.
Utworzono	Czas utworzenia usługi; wyświetlany podczas edycji istniejącej usługi.
Konfiguracja zaawansowana	Kliknij nagłówek *Konfiguracja zaawansowana*, aby wyświetlić opcje konfiguracji zaawansowanej.

Konfiguracja zaawansowana



Parametr	Opis
Dodatkowe reguły	Kliknij *Dodaj*, aby skonfigurować dodatkowe reguły obliczania statusu.
Ustaw status na	Ustaw status usługi na *OK* (domyślnie), *Niesklasyfikowany*, *Informacja*, *Ostrzeżenie*, *Przeciętny*, *Wysoki* lub *Katastrofa* w przypadku spełnienia warunku.
Warunek	Wybierz warunek dla bezpośrednich usług podrzędnych: **jeśli co najmniej (N) usług podrzędnych ma status (Status) lub wyższy** **jeśli co najmniej (N%) usług podrzędnych ma status (Status) lub wyższy** **jeśli mniej niż (N) usług podrzędnych ma status (Status) lub niższy** **jeśli mniej niż (N%) usług podrzędnych ma status (Status) lub niższy** **jeśli waga usług podrzędnych ze statusem (Status) lub wyższym wynosi co najmniej (W)** **jeśli waga usług podrzędnych ze statusem (Status) lub wyższym wynosi co najmniej (N%)** **jeśli waga usług podrzędnych ze statusem (Status) lub niższym jest mniejsza niż (W)** **jeśli waga usług podrzędnych ze statusem (Status) lub niższym jest mniejsza niż (N%)** Jeśli określono kilka warunków i sytuacja spełnia więcej niż jeden warunek, zostanie ustawiony najwyższy poziom ważności.
N (W)	Ustaw wartość N lub W (1-100000) albo N% (1-100) w warunku.
Status	Wybierz wartość *Status* w warunku: *OK* (domyślnie), *Niesklasyfikowany*, *Informacja*, *Ostrzeżenie*, *Przeciętny*, *Wysoki* lub *Katastrofa*.
Reguła propagacji statusu	Reguła propagowania statusu usługi do usługi nadrzędnej: **Bez zmian** - status jest propagowany bez zmian **Zwiększ o** - możesz zwiększyć propagowany status o 1 do 5 poziomów ważności **Zmniejsz o** - możesz zmniejszyć propagowany status o 1 do 5 poziomów ważności **Ignoruj tę usługę** - status nie jest w ogóle propagowany do usługi nadrzędnej **Stały status** - status jest propagowany statycznie, tj. zawsze jako taki sam
Waga	Waga usługi (liczba całkowita z zakresu od 0 (domyślnie) do 1000000).

Note:
Dodatkowe reguły obliczania statusu mogą być używane wyłącznie do zwiększania poziomu ważności ponad poziom obliczony zgodnie z głównym parametrem *Reguła obliczania statusu*. Jeśli zgodnie z dodatkowymi regułami status powinien być *Ostrzeżenie*, ale zgodnie z *Reguła obliczania statusu* status to *Katastrofa* — usługa będzie miała status *Katastrofa*.

Zakładka **Tagi** zawiera tagi na poziomie usługi. Tagi na poziomie usługi są używane do identyfikacji usługi. Tagi tego typu nie są używane do mapowania problemów do usługi (w tym celu użyj *Tagi problemów* z pierwszej zakładki).

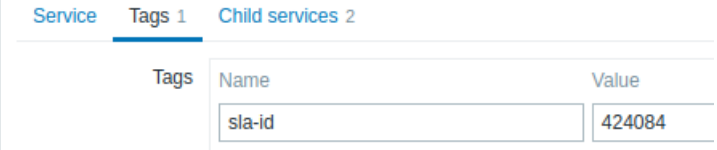
Zakładka **Usługi podrzędne** umożliwia określenie usług zależnych. Kliknij *Dodaj*, aby dodać usługę z listy istniejących usług. Jeśli chcesz dodać nową usługę podrzędną, najpierw zapisz tę usługę, a następnie kliknij ikonę plusa obok usługi, którą właśnie utworzyłeś.

Tagi W usługach istnieją dwa różne typy tagów:

- Tagi usługi
- Tagi problemu

Tagi usługi

Tagi usługi są używane do dopasowywania usług do akcji usług i SLA. Tagi te są określane na karcie konfiguracji usługi *Tagi*. Do mapowania SLA używana jest logika *OR*: usługa zostanie przypisana do SLA, jeśli ma co najmniej jeden pasujący tag. W akcjach usług reguły mapowania można konfigurować i mogą one używać logiki *AND*, *OR* lub *AND/OR*.

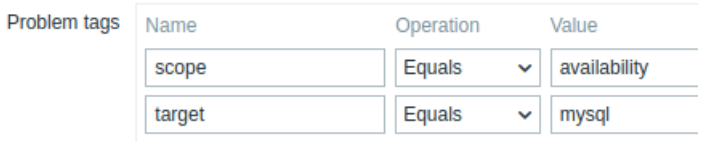


Tagi problemów

Tagi problemów są używane do dopasowywania problemów i usług. Tagi te są określane na karcie konfiguracji usługi podstawowej.

Tylko usługi podrzędne na najniższym poziomie hierarchii mogą mieć zdefiniowane tagi problemów i być bezpośrednio skorelowane z problemami. Jeśli tagi problemów są zgodne, status usługi zmieni się na taki sam status, jaki ma problem. W przypadku kilku problemów usługa będzie miała status problemu o najwyższym poziomie ważności. Status usługi nadrzędnej jest następnie obliczany na podstawie statusów usług podrzędnych zgodnie z regułami obliczania statusu.

Jeśli określono kilka tagów, stosowana jest logika *AND*: problem musi mieć wszystkie tagi określone w konfiguracji usługi, aby został przypisany do usługi.



Note:
Problem w Zabbix dziedziczy tagi z całego łańcucha szablonów, hostów, pozycji, scenariuszy WWW i wyzwalaczy. Każdy z tych tagów może zostać użyty do dopasowywania problemów do usług.

Przykład:

Problem *Kamera internetowa 3 jest niedostępna* ma tagi `type:video-surveillance`, `floor:1` i `name:webcam-3` oraz status *Ostrzeżenie*

Usługa **Kamera internetowa 3** ma określony tylko jeden tag problemu: `name:webcam-3`



Status usługi zmieni się z *OK* na *Ostrzeżenie*, gdy ten problem zostanie wykryty.

Gdyby usługa **Kamera internetowa 3** miała tagi problemu `name:webcam-3` i `floor:2`, jej status nie zostałby zmieniony po wykryciu problemu, ponieważ warunki są spełnione tylko częściowo.

Modyfikowanie istniejących usług

Note:
Przyciski opisane poniżej są widoczne tylko wtedy, gdy sekcja *Services* jest w trybie edycji.

Aby edytować istniejącą usługę, kliknij ikonę ołówka obok usługi.

Aby sklonować istniejącą usługę, kliknij ikonę ołówka, aby otworzyć jej konfigurację, a następnie kliknij przycisk Clone. Po sklonowaniu usługi jej linki nadrzędne są zachowywane, natomiast linki podrzędne nie.

Aby usunąć usługę, kliknij ikonę x obok niej. Gdy usuniesz usługę nadrzędną, jej usługi podrzędne nie zostaną usunięte i zostaną przeniesione o jeden poziom wyżej w drzewie usług (usługi podrzędne 1. poziomu otrzymają ten sam poziom co usunięta usługa nadrzędna).

Dwa przyciski poniżej listy usług oferują opcje edycji zbiorczej:

- *Mass update* - zbiorcza aktualizacja właściwości usług
- *Delete* - usuwanie usług

Aby skorzystać z tych opcji, zaznacz pola wyboru przed odpowiednimi usługami, a następnie kliknij wymagany przycisk.

2 SLA

Przegląd Po utworzeniu usług można rozpocząć monitorowanie, czy ich wydajność jest zgodna z umową o gwarantowanym poziomie usług (SLA).

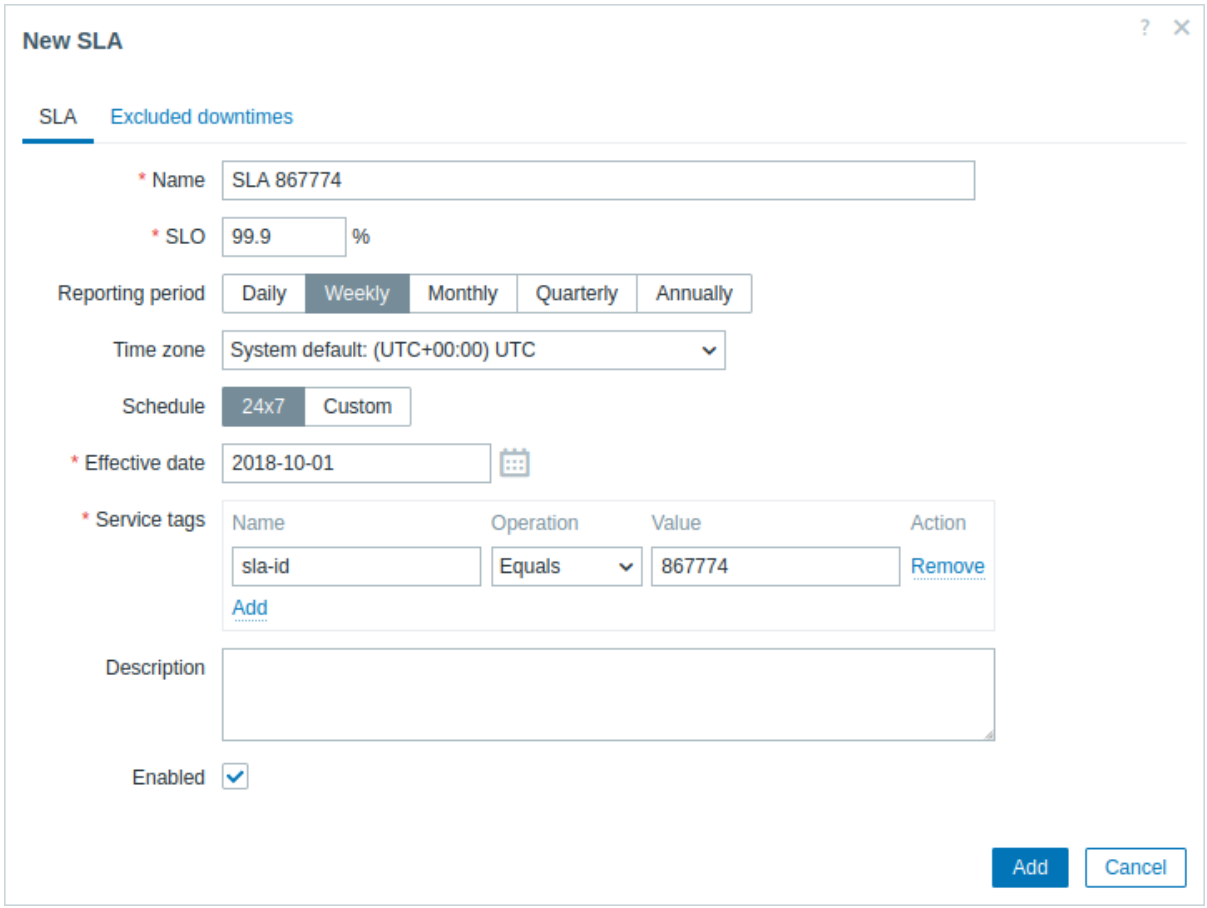
Sekcja menu *Usługi->SLA* umożliwia konfigurowanie umów SLA dla różnych usług. Umowa SLA w Zabbix definiuje cel poziomu usług (SLO), oczekiwany harmonogram dostępności oraz planowane przestoje.

Umowy SLA i usługi są dopasowywane za pomocą tagów usług. Ta sama umowa SLA może być zastosowana do wielu usług — wydajność będzie mierzona oddzielnie dla każdej pasującej usługi. Do jednej usługi może być przypisanych wiele umów SLA — dane dla każdej z nich będą wyświetlane osobno.

W raportach SLA Zabbix udostępnia dane wskaźnika poziomu usług (SLI), który mierzy rzeczywistą dostępność usługi. To, czy usługa spełnia cele SLA, jest określane przez porównanie SLO (oczekiwana dostępność w %) ze SLI (rzeczywista dostępność w %).

Konfiguracja Aby utworzyć nowe SLA, kliknij przycisk *Utwórz SLA*.

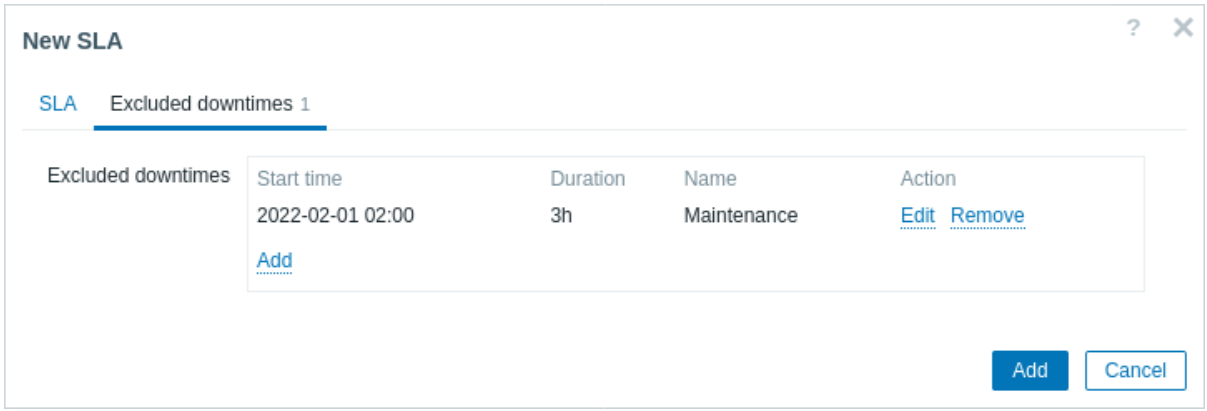
Zakładka **SLA** umożliwia określenie ogólnych parametrów SLA.



Parametr	Opis
Nazwa	Wprowadź nazwę SLA.
SLO	Wprowadź cel poziomu usługi (SLO) jako wartość procentową.
Okres raportowania	Wybierz okres raportowania SLI używany do obliczania danych SLI w raporcie SLA: **Dzienny** - każdy dzień, od 00:00:00 do 23:59:59. **Tygodniowy** - każdy tydzień, od niedzieli 00:00:00 do soboty 23:59:59. **Miesięczny** - każdy miesiąc, od pierwszego dnia 00:00:00 do ostatniego dnia 23:59:59. **Kwartalny** - każdy kwartał kalendarzowy: Q1 (sty-mar), Q2 (kwi-cze), Q3 (lip-wrz), Q4 (paź-gru). **Roczny** - każdy rok kalendarzowy, od 1 stycznia 00:00:00 do 31 grudnia 23:59:59.
Strefa czasowa	Wybierz strefę czasową SLA.
Harmonogram	Wybierz harmonogram SLA - 24x7 lub niestandardowy.
Data obowiązywania	Wybierz datę rozpoczęcia obliczania SLA.

Parametr	Opis
Tagi usługi	Dodaj tagi usługi, aby zidentyfikować usługi, do których to SLA ma być stosowane. **Nazwa** - nazwa tagu usługi; musi być dokładnie zgodna, z rozróżnieniem wielkości liter. **Operacja** - wybierz *Równa się*, jeśli wartość tagu musi być dokładnie zgodna (z rozróżnieniem wielkości liter), lub *Zawiera*, jeśli ma pasować część wartości tagu (bez rozróżniania wielkości liter). **Wartość** - wartość tagu usługi do wyszukania zgodnie z wybraną operacją. SLA jest stosowane do usługi, jeśli pasuje co najmniej jeden tag usługi.
Opis	Dodaj opis SLA.
Włączone	Zaznacz pole wyboru, aby włączyć obliczanie SLA.

Zakładka **Wykluczone przestoje** umożliwia określenie przestojów, które są wyłączone z obliczania SLA.



Kliknij *Dodaj*, aby skonfigurować wykluczone przestoje, a następnie wprowadź nazwę okresu, datę rozpoczęcia i czas trwania.

Raporty SLA To, jak usługa działa w porównaniu do SLA, jest widoczne w raporcie SLA. Raporty SLA można wyświetlać:

- z sekcji *SLA*, klikając hiperłącze raportu SLA;
- z sekcji *Usługi*, klikając nazwę SLA na karcie informacji;
- w widżecie pulpitu *Raport SLA*.

Po skonfigurowaniu SLA karta *Informacje* w sekcji usług będzie również wyświetlać pewne informacje o wydajności usługi.

3 Przykład konfiguracji

Przegląd Ta sekcja opisuje prostą konfigurację monitorowania klastra wysokiej dostępności Zabbix jako usługi.

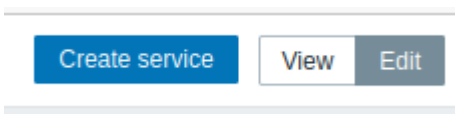
Wymagania wstępne Przed skonfigurowaniem monitorowania usług należy mieć skonfigurowane hosty:

- *Węzeł HA 1* z co najmniej jednym wyzwalaczem i tagiem (najlepiej ustawionym na poziomie wyzwalacza) `component:ha-node-1`
- *Węzeł HA 2* z co najmniej jednym wyzwalaczem i tagiem (najlepiej ustawionym na poziomie wyzwalacza) `component:ha-node-2`

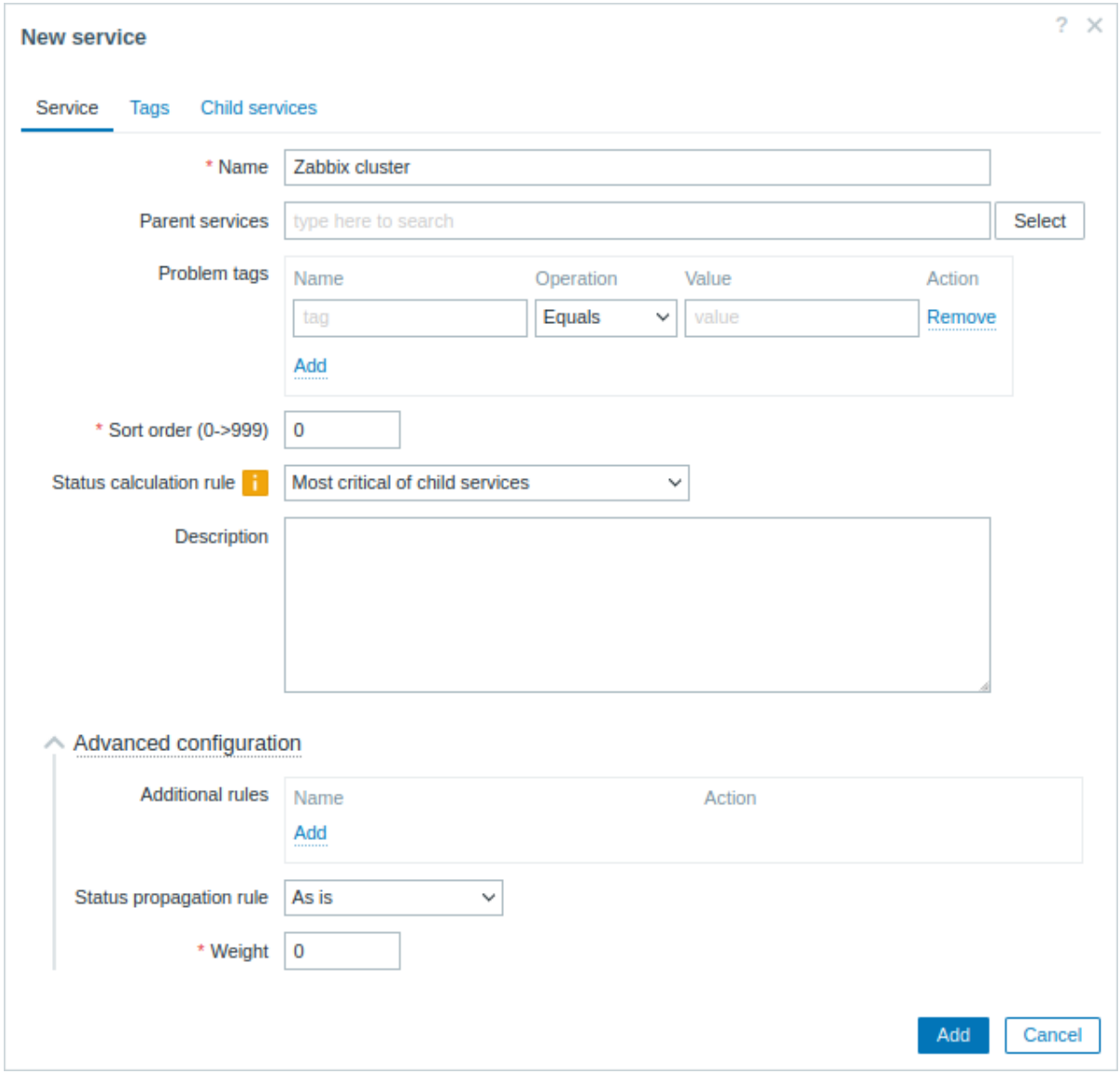
Drzewo usług Następnym krokiem jest zbudowanie drzewa usług. W tym przykładzie infrastruktura jest bardzo prosta i składa się z trzech usług: *Zabbix cluster* (nadrzędna) oraz dwóch usług podrzędnych *Zabbix server node 1* i *Zabbix server node 2*.

```
Zabbix cluster
|
|- Zabbix server node 1
|- Zabbix server node 2
```

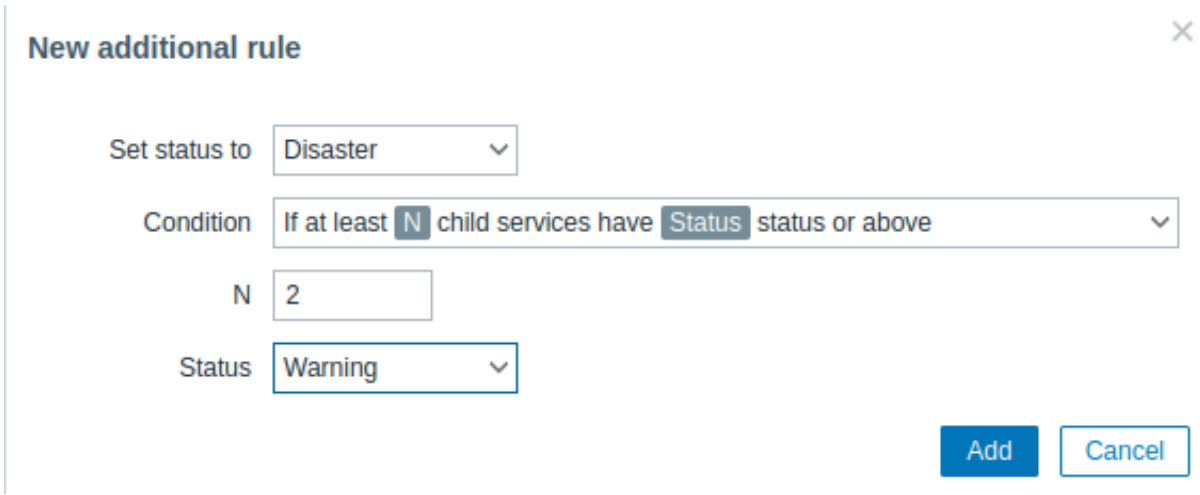
Na stronie Usługi włącz tryb *Edit* i naciśnij Create service:



W oknie konfiguracji usługi wprowadź nazwę *Zabbix cluster* i kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji.



Skonfiguruj dodatkową regułę:

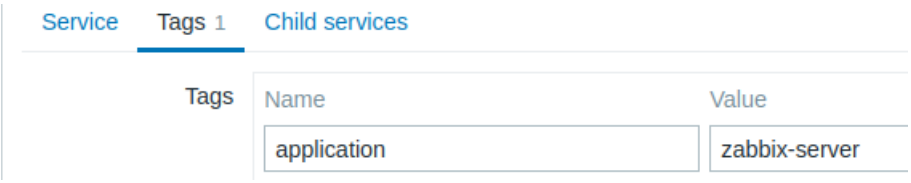


Zabbix cluster będzie mieć dwie usługi podrzędne — po jednej dla każdego z węzłów HA. Jeśli oba węzły HA mają problemy o statusie co najmniej *Warning*, status usługi nadrzędnej powinien zostać ustawiony na *Disaster*. Aby to osiągnąć, dodatkową regułę należy skonfigurować następująco:

- Set status to: Disaster
- Condition: If at least `N` child services have `Status` status or above
- N: 2
- Status: Warning

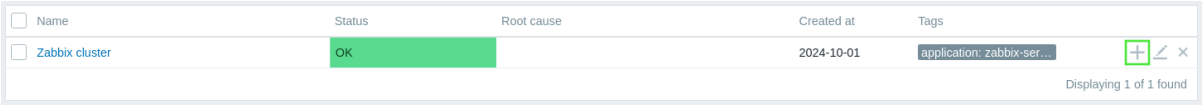
Przejdź do zakładki *Tags* i dodaj tag `application:zabbix-server`. Ten tag zostanie później użyty dla akcji usług i raportów

SLA.



Zapisz nową usługę.

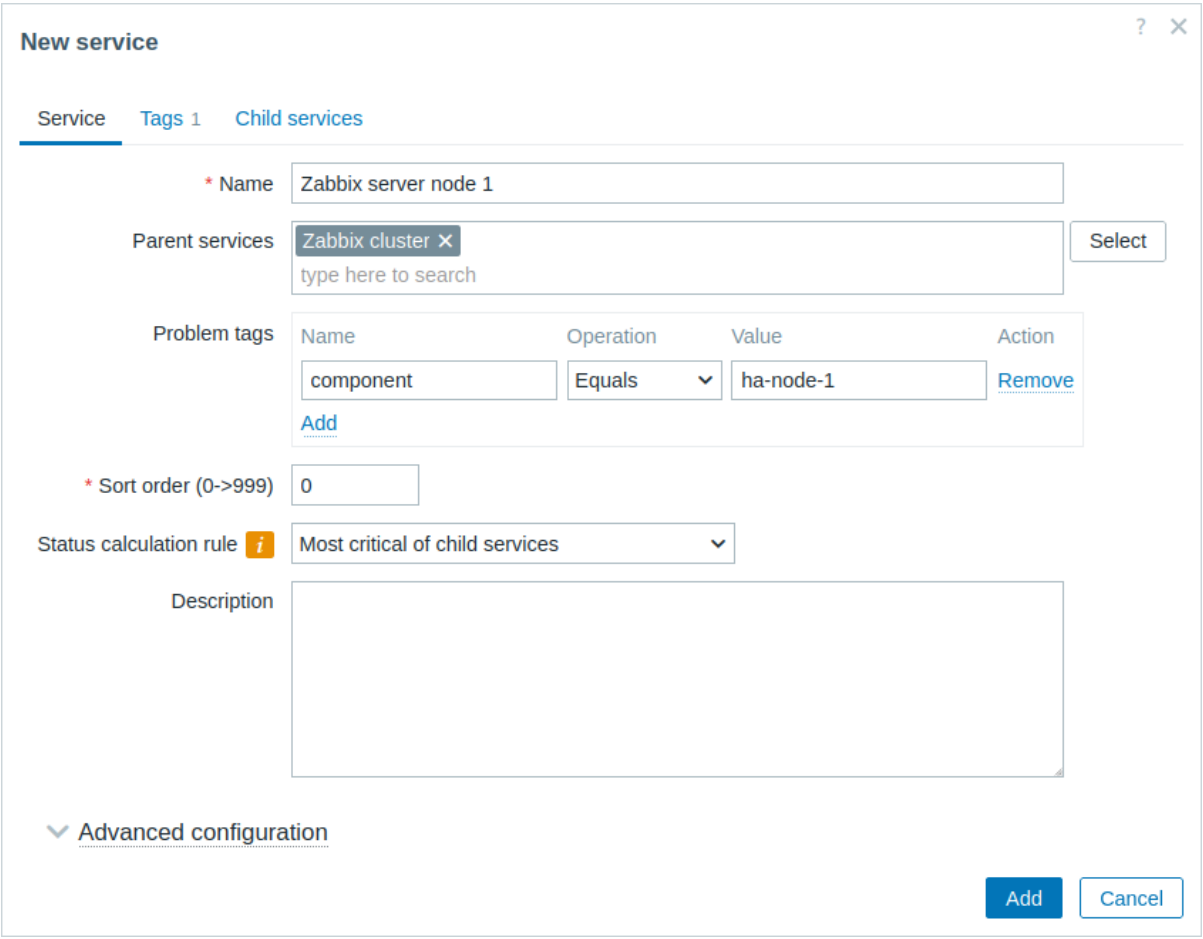
Aby dodać usługę podrzędną, naciśnij ikonę plusa obok usługi Zabbix cluster (ikona jest widoczna tylko w trybie *Edit*).



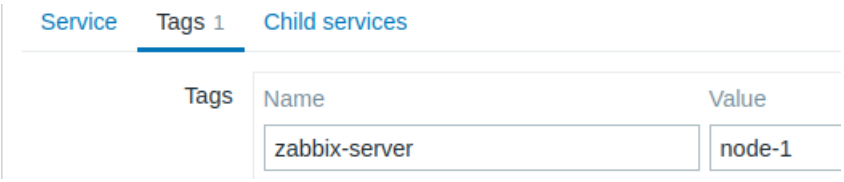
W oknie konfiguracji usługi wprowadź nazwę *Zabbix server node 1*. Zwróć uwagę, że parametr Parent services jest już wstępnie wypełniony wartością *Zabbix cluster*.

Dostępność tej usługi zależy od problemów na hoście *HA node 1*, oznaczonych tagiem problemu `component:ha-node-1`. W parametrze Problem tags wprowadź:

- Nazwa: component
- Operacja: Equals
- Wartość: ha-node-1



Przejdź do zakładki *Tags* i dodaj tag usługi: `zabbix-server:node-1`. Ten tag zostanie później użyty w akcjach usług i raportach SLA.



Zapisz nową usługę.

Utwórz kolejną usługę podrzędną klastra Zabbix o nazwie „Zabbix server node 2”.

Ustaw tagi problemów w następujący sposób:

- Nazwa: component
- Operacja: Równa się
- Wartość: ha-node-2

Przejdź do zakładki *Tagi* i dodaj tag usługi: `zabbix-server:node-2`.

Zapisz nową usługę.

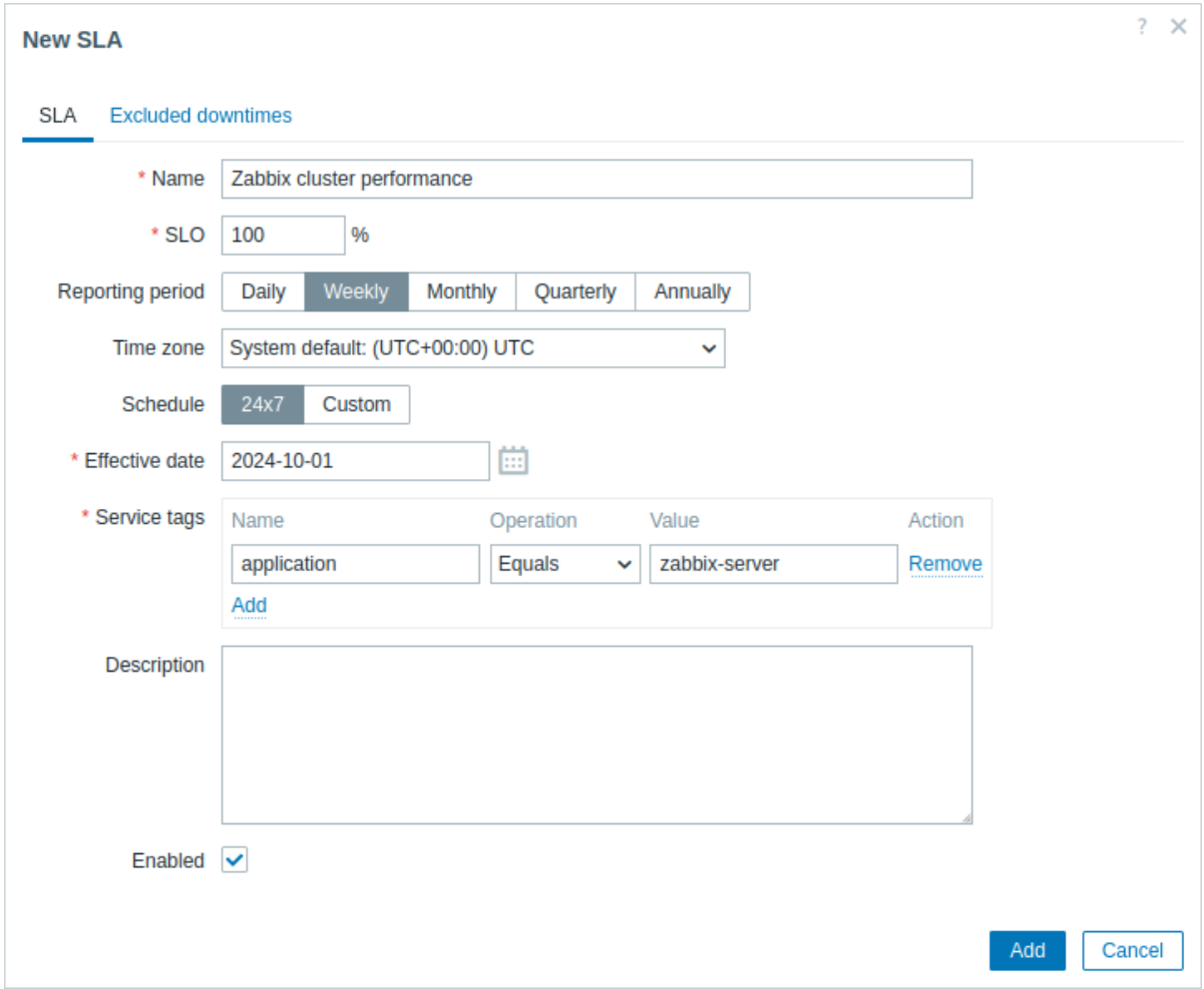
SLA W tym przykładzie oczekiwana wydajność klastra Zabbix wynosi 100%, z wyłączeniem półrocznego, jednogodzinnego okresu konserwacji.

Najpierw należy dodać nową umowę SLA.

Przejdź do sekcji menu *Services->SLA* i naciśnij Create SLA. Wprowadź nazwę *Zabbix cluster performance* i ustaw SLO na 100%.

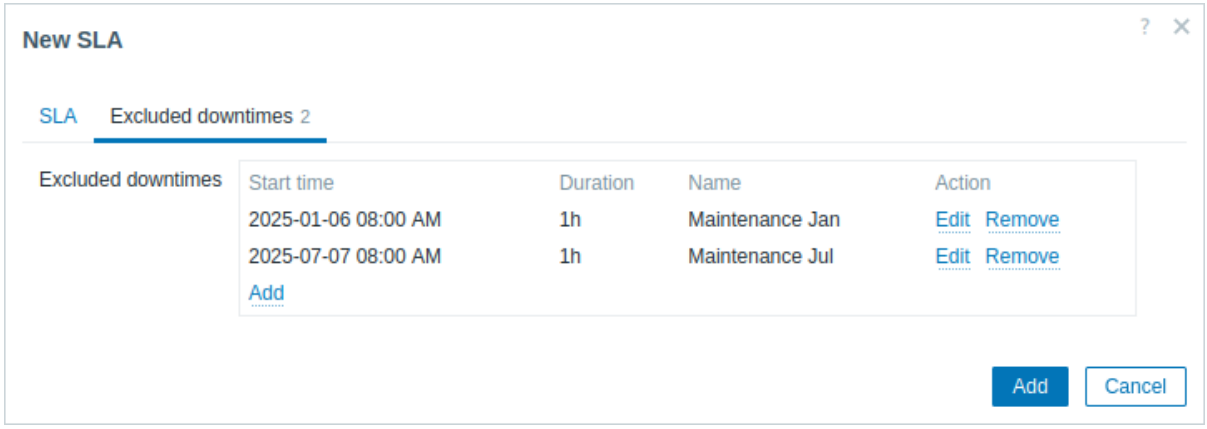
Usługa Zabbix cluster ma tag usługi `application:zabbix-server`. Aby użyć tej umowy SLA do mierzenia wydajności klastra Zabbix, w parametrze *Service tags* podaj:

- Nazwa: application
- Operacja: Equals
- Wartość: zabbix-server



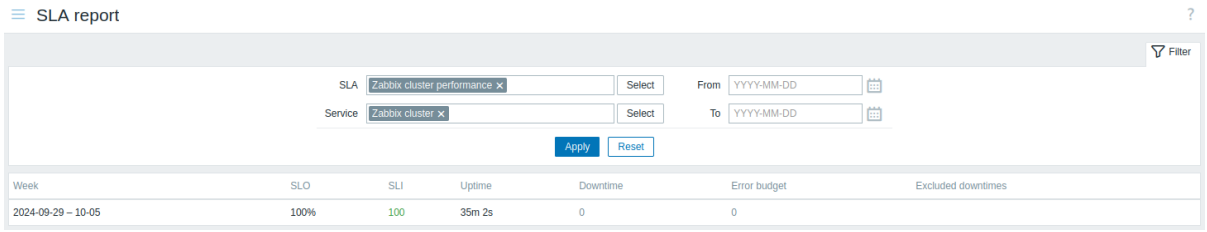
W rzeczywistej konfiguracji można również zaktualizować żądany okres raportowania, strefę czasową i datę rozpoczęcia lub zmienić harmonogram z 24/7 na niestandardowy. W tym przykładzie ustawienia domyślne są wystarczające.

Przejdź do karty *Excluded downtimes* i dodaj przestoje dla zaplanowanych okresów konserwacji, aby wykluczyć te okresy z obliczania SLA. W sekcji Excluded downtimes naciśnij odnośnik Add, wprowadź nazwę przestoju, planowany czas rozpoczęcia i czas trwania.

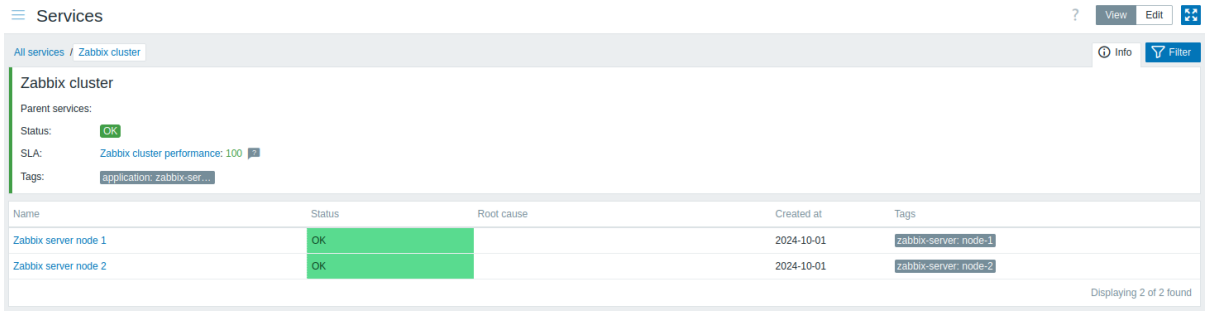


Naciśnij Add, aby zapisać nową umowę SLA.

Przejdź do sekcji raportów SLA, aby wyświetlić raport SLA dla Zabbix cluster.



Informacje o SLA można również sprawdzić w sekcji *Services*.



9 Monitorowanie WWW

Przegląd Za pomocą Zabbix można sprawdzać kilka aspektów dostępności witryn internetowych.

> **Attention:**
> Aby wykonywać monitorowanie WWW, serwer Zabbix musi być początkowo skonfigurowany z obsługą cURL (libcurl).

Aby włączyć monitorowanie WWW, należy zdefiniować scenariusze WWW. Scenariusz WWW składa się z jednego lub kilku żądań HTTP lub ”kroków”. Kroki są okresowo wykonywane przez serwer Zabbix w z góry określonej kolejności. Jeśli host jest monitorowany przez proxy, kroki są wykonywane przez proxy.

Scenariusze WWW są przypisywane do hostów/szablonów w taki sam sposób jak pozycje, wyzwalacze itp. Oznacza to, że scenariusze WWW można również tworzyć na poziomie szablonu, a następnie zastosować do wielu hostów jednym działaniem.

W każdym scenariuszu WWW zbierane są następujące informacje:

- średnia prędkość pobierania na sekundę dla wszystkich kroków całego scenariusza
- numer kroku, który zakończył się niepowodzeniem
- ostatni komunikat o błędzie

W każdym kroku scenariusza WWW zbierane są następujące informacje:

- prędkość pobierania na sekundę
- czas odpowiedzi
- kod odpowiedzi

Więcej informacji można znaleźć w sekcji pozycje monitorowania WWW.

Dane zbierane podczas wykonywania scenariuszy WWW są przechowywane w bazie danych. Dane są automatycznie wykorzystywane do wykresów, wyzwalaczy i powiadomień.

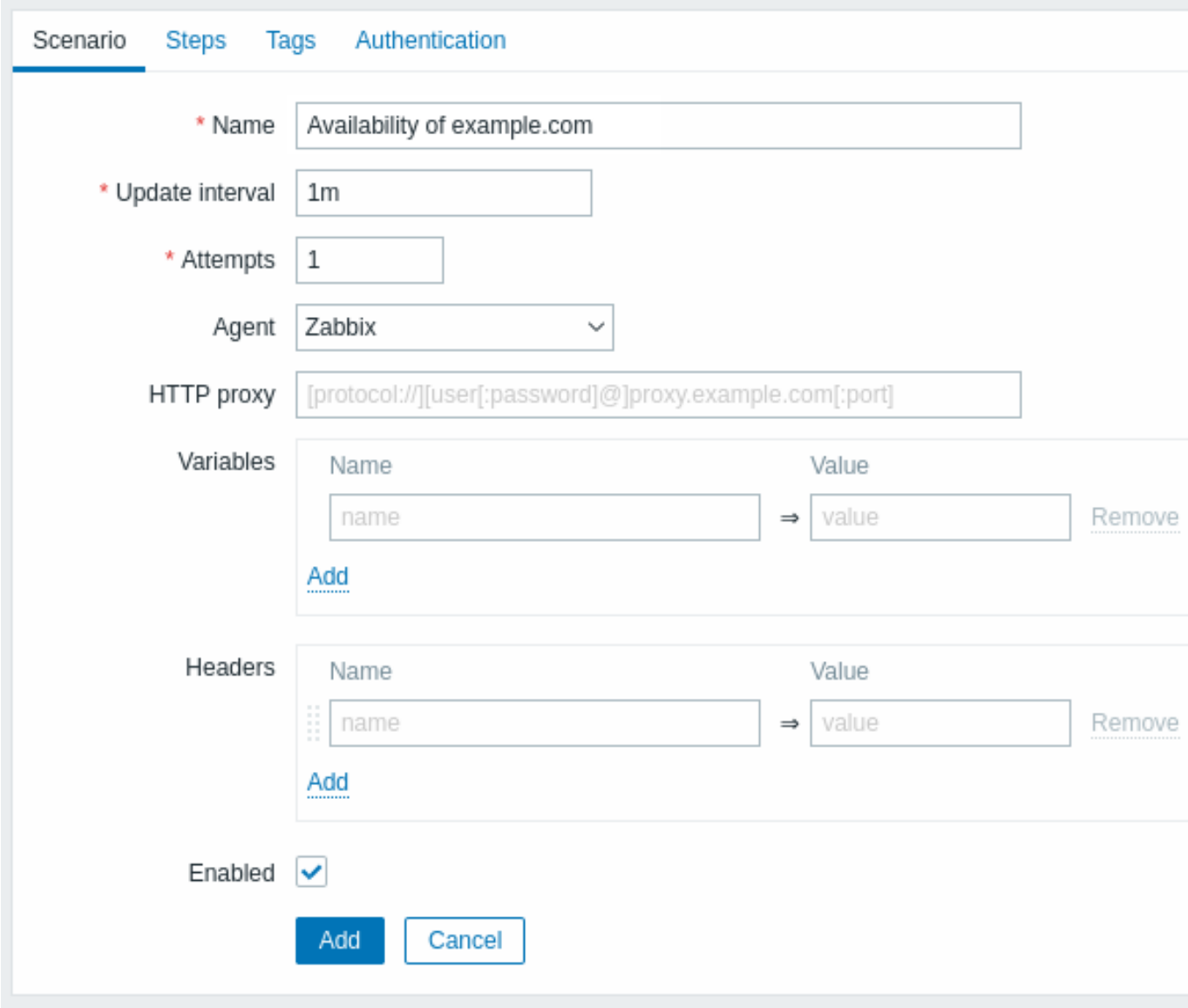
Zabbix może również sprawdzać, czy pobrana strona HTML zawiera z góry określony ciąg znaków. Może wykonać symulowane logowanie i przejść przez ścieżkę symulowanych kliknięć myszą na stronie.

Monitorowanie WWW w Zabbix obsługuje zarówno HTTP, jak i HTTPS. Podczas uruchamiania scenariusza WWW Zabbix może opcjonalnie śledzić przekierowania (zobacz opcję *Follow redirects* poniżej). Maksymalna liczba przekierowań jest na stałe ustawiona na 10 (z użyciem opcji cURL CURLOPT_MAXREDIRS). Wszystkie pliki cookie są zachowywane podczas wykonywania pojedynczego scenariusza.

Konfigurowanie scenariusza WWW Aby skonfigurować scenariusz WWW:

- Przejdź do: *Data collection* → *Hosts* (lub *Templates*)
- Kliknij *Web* w wierszu hosta/szablonu
- Kliknij *Create web scenario* po prawej stronie (lub nazwę scenariusza, aby edytować istniejący scenariusz)
- Wprowadź parametry scenariusza w formularzu

Zakładka **Scenario** umożliwia skonfigurowanie ogólnych parametrów scenariusza WWW.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametry scenariusza:

Parameter	Description
Name	Unikalna nazwa scenariusza. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane są makra użytkownika, makra te pozostaną nierozwinięte w nazwach pozycji monitorowania WWW.
Update interval	Jak często scenariusz będzie wykonywany. Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane jest makro użytkownika i jego wartość zostanie zmieniona (np. 5m → 30s), następne sprawdzenie zostanie wykonane zgodnie z poprzednią wartością (w przykładzie wartości — później). Nowe scenariusze WWW zostaną sprawdzone w ciągu 60 sekund od ich utworzenia.
Attempts	Liczba prób wykonania kroków scenariusza WWW. W przypadku problemów sieciowych (timeout, brak łączności itp.) Zabbix może powtórzyć wykonanie kroku kilka razy. Ustawiona wartość będzie miała jednakowy wpływ na każdy krok scenariusza. Można określić maksymalnie 10 prób, wartość domyślna to 1. *Uwaga*: Zabbix nie powtórzy kroku z powodu nieprawidłowego kodu odpowiedzi lub niedopasowania wymaganego ciągu znaków.
Agent	Wybierz agent klienta. Zabbix będzie udawał wybraną przeglądarkę. Jest to przydatne, gdy witryna zwraca różną zawartość dla różnych przeglądarek. W tym polu można używać makr użytkownika.
HTTP proxy	Możesz określić używany HTTP proxy, stosując format `[protocol://][username[:password]@]proxy.example.com[:port]`. Ustawia to opcję cURL CURLOPT_PROXY. Opcjonalny prefiks `protocol://` może być użyty do określenia alternatywnych protokołów proxy (obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli nie określono protokołu, proxy będzie traktowany jako HTTP proxy. Domyślnie zostanie użyty port 1080. Jeśli określono, proxy nadpisze zmienne środowiskowe związane z proxy, takie jak http_proxy, HTTPS_PROXY. Jeśli nie określono, proxy nie nadpisze zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana „as is”, bez sprawdzania poprawności. Możesz również podać adres SOCKS proxy. Jeśli określisz nieprawidłowy protokół, połączenie nie powiedzie się, a pozycja stanie się nieobsługiwana. *Uwaga*: w przypadku HTTP proxy obsługiwane jest tylko proste uwierzytelnianie. W tym polu można używać makr użytkownika.
Variables	Zmienne, które mogą być używane w krokach scenariusza (URL, zmienne post). Mają one następujący format: **{macro1}**=value1 **{macro2}**=value2 **{macro3}**=regex:<regular expression> **{macro4}**=jsonpath:<jsonpath> **{macro5}**=xmlxpath:<xmlxpath> **{macro6}**={{macro}.function()} (zobacz funkcje makr) Na przykład: {username}=Alexei {password}=kj3h5kJ34bd {hostid}=regex:hostid is ([0-9]+) {url}=jsonpath:$.host_url {status}=xmlxpath://host/response/status {newvar}={{myvar}.btoa()} Do makr można następnie odwoływać się w krokach jako {username}, {password}, {hostid} itd. Zabbix automatycznie zastąpi je rzeczywistymi wartościami. Należy pamiętać, że zmienne z `regex:` wymagają jednego kroku do pobrania wartości wyrażenia regularnego, więc wyodrębniona wartość może zostać zastosowana dopiero w następnym kroku. Jeśli część wartości zaczyna się od `regex:`, część po niej jest traktowana jako wyrażenie regularne przeszukujące stronę WWW i, jeśli zostanie znalezione, zapisujące dopasowanie w zmiennej. Musi być obecna co najmniej jedna podgrupa, aby można było wyodrębnić dopasowaną wartość. Obsługiwane są makra użytkownika oraz makra {HOST.*}. Zmienne są automatycznie kodowane URL przy użyciu w polach zapytania lub danych formularza dla zmiennych post, ale muszą być kodowane URL ręcznie, gdy są używane w surowym post lub bezpośrednio w URL.

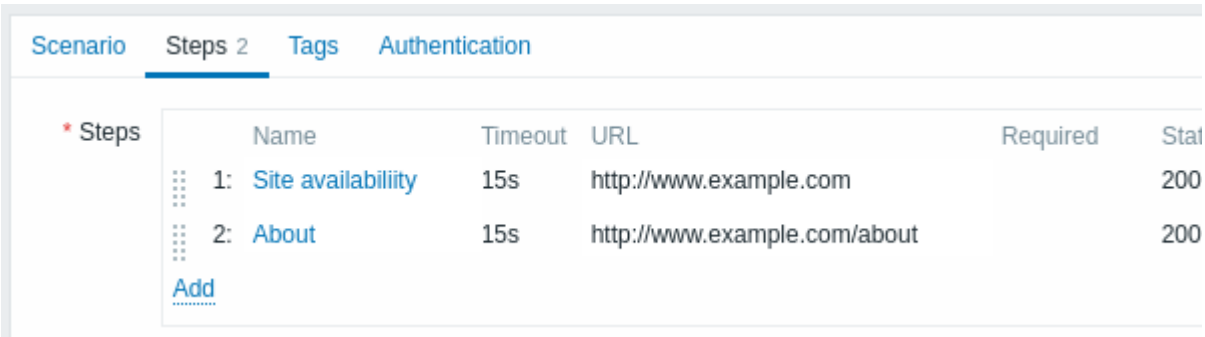
Parameter	Description
Headers	Nagłówki HTTP są używane podczas wykonywania żądania. Można używać nagłówków domyślnych i niestandardowych. Nagłówki będą przypisywane przy użyciu ustawień domyślnych zależnych od typu Agent wybranego z listy rozwijanej na poziomie scenariusza i będą stosowane do wszystkich kroków, chyba że zostaną niestandardowo zdefiniowane na poziomie kroku. **Należy pamiętać, że zdefiniowanie nagłówka na poziomie kroku automatycznie odrzuca wszystkie wcześniej zdefiniowane nagłówki, z wyjątkiem domyślnego nagłówka przypisanego przez wybranie 'User-Agent' z listy rozwijanej na poziomie scenariusza.** Jednak nawet domyślny nagłówek 'User-Agent' może zostać nadpisany przez określenie go na poziomie kroku. Aby usunąć nagłówek na poziomie scenariusza, należy podać nazwę nagłówka i przypisać mu pustą wartość na poziomie kroku. Nagłówki powinny być podawane przy użyciu tej samej składni, w jakiej występują w protokole HTTP, opcjonalnie z użyciem dodatkowych funkcji obsługiwanych przez opcję cURL CURLOPT_HTTPHEADER. Na przykład: Accept-Charset=utf-8 Accept-Language=en-US Content-Type=application/xml; charset=utf-8 Obsługiwane są makra użytkownika oraz makra {HOST.*}.
Enabled	Scenariusz jest aktywny, jeśli to pole jest zaznaczone, w przeciwnym razie — wyłączony.

Należy pamiętać, że podczas edycji istniejącego scenariusza w formularzu dostępne są dwa dodatkowe przyciski:

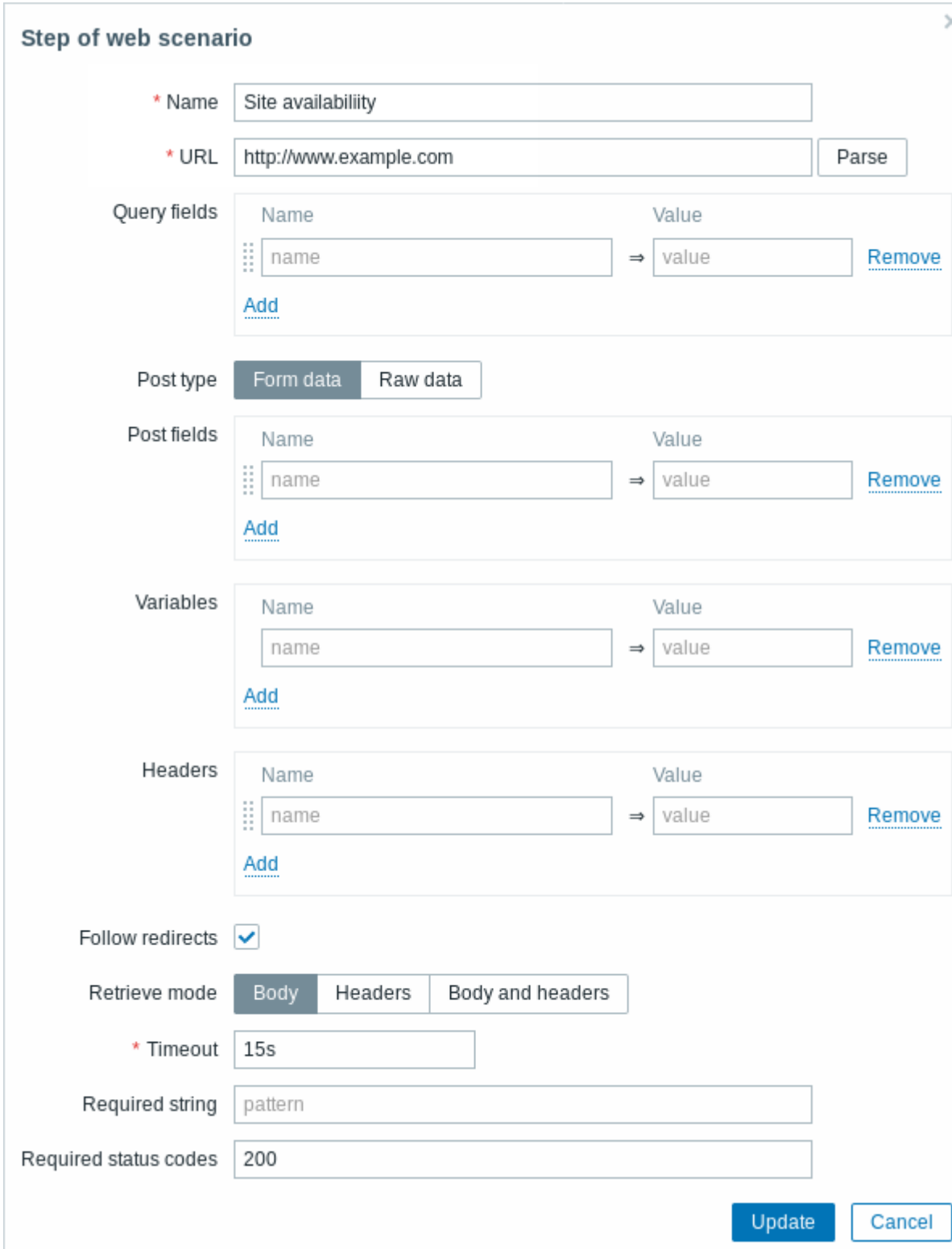
Clone	Utwórz kolejny scenariusz na podstawie właściwości istniejącego.
Clear history and trends	Usuń dane historyczne i trendy dla scenariusza. Spowoduje to, że serwer wykona scenariusz natychmiast po usunięciu danych.

Note:
Jeśli pole *HTTP proxy* pozostanie puste, innym sposobem użycia HTTP proxy jest ustawienie zmiennych środowiskowych związanych z proxy.
Dla sprawdzeń HTTP — ustaw zmienną środowiskową **http_proxy** dla użytkownika serwera Zabbix. Na przykład, `http_proxy=http://proxy_ip:proxy_port`.
Dla sprawdzeń HTTPS — ustaw zmienną środowiskową **HTTPS_PROXY**. Na przykład `HTTPS_PROXY=http://proxy_ip:proxy_port`. Więcej szczegółów jest dostępnych po uruchomieniu polecenia powłoki: *# man curl*.

Zakładka **Steps** umożliwia skonfigurowanie kroków scenariusza WWW. Aby dodać krok scenariusza WWW, kliknij *Add* w bloku *Steps*.



Note:
Tajne makra użytkownika nie mogą być używane w adresach URL, ponieważ zostaną rozwinięte do "******".



Konfigurowanie kroków

Parametry kroku:

Parametr	Opis
Nazwa	Unikalna nazwa kroku. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane są makra użytkownika, te makra pozostaną nierozwiązane w nazwach pozycji monitorowania WWW.

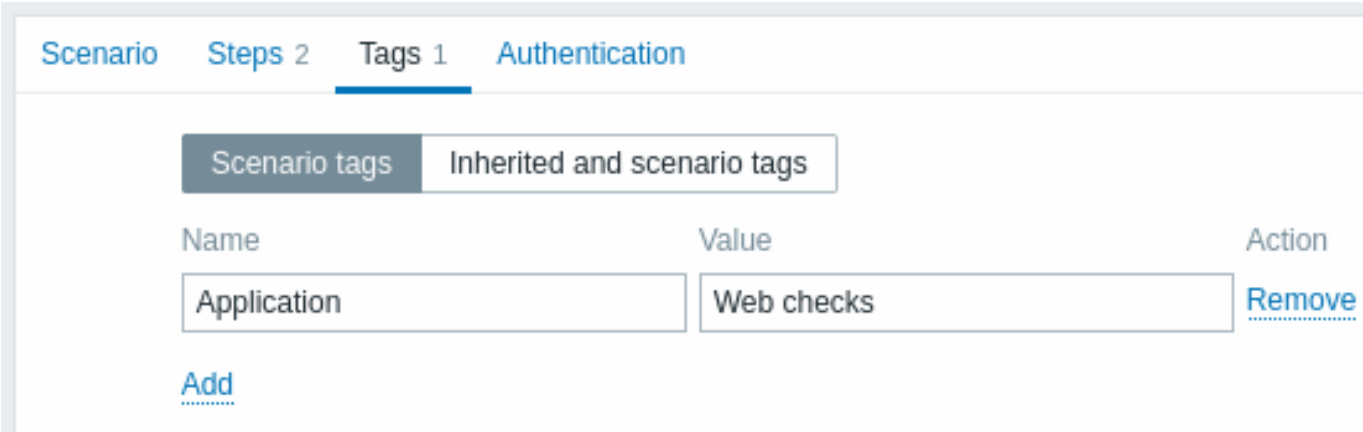
Parametr	Opis
URL	URL, z którym należy się połączyć i z którego należy pobrać dane. Na przykład: https://www.example.com http://www.example.com/download Nazwy domen mogą być podawane znakami Unicode. Podczas wykonywania kroku scenariusza WWW są one automatycznie konwertowane z punycode do ASCII. Przycisk *Parse* może zostać użyty do oddzielenia opcjonalnych pól zapytania (takich jak ?name=Admin&password=mypassword) od URL, przenosząc atrybuty i wartości do *Pól zapytania* w celu automatycznego kodowania URL. W URL można używać zmiennych, stosując składnię {macro}. Zmienne mogą być kodowane URL ręcznie przy użyciu składni {{macro}.urlencode()}. Obsługiwane są makra użytkownika oraz makra {HOST.*}. Ograniczone do 2048 znaków.
Pola zapytania	Zmienne HTTP GET dla URL. Określane jako pary atrybut-wartość. Wartości są automatycznie kodowane URL. Wartości ze zmiennych scenariusza, makr użytkownika lub makr {HOST.*} są rozwiązywane, a następnie automatycznie kodowane URL. Użycie składni {{macro}.urlencode()} spowoduje ich podwójne zakodowanie URL. Obsługiwane są makra użytkownika oraz makra {HOST.*}.
Post	Zmienne HTTP POST. W trybie **Form data** określane jako pary atrybut-wartość. Wartości są automatycznie kodowane URL. Wartości ze zmiennych scenariusza, makr użytkownika lub makr {HOST.*} są rozwiązywane, a następnie automatycznie kodowane URL. W trybie **Raw data** atrybuty/wartości są wyświetlane w jednym wierszu i łączone symbolem **&**. Surowe wartości mogą być ręcznie kodowane/dekodowane URL przy użyciu składni {{macro}.urlencode()} lub {{macro}.urldecode()}. Na przykład: id=2345&userid={user} Jeśli {user} jest zdefiniowane jako zmienna scenariusza WWW, zostanie zastąpione swoją wartością podczas wykonywania kroku. Jeśli chcesz zakodować URL zmienną, zastąp {user} przez {{user}.urlencode()}. Obsługiwane są makra użytkownika oraz makra {HOST.*}.
Zmienne	Zmienne na poziomie kroku, które mogą być używane przez funkcje GET i POST. Określane jako pary atrybut-wartość. Zmienne na poziomie kroku zastępują zmienne na poziomie scenariusza lub zmienne z poprzedniego kroku. Jednak wartość zmiennej na poziomie kroku wpływa tylko na następny krok (a nie na bieżący krok). Mają one następujący format: **{macro}**=value **{macro}**=regex:<regular expression> Więcej informacji można znaleźć w opisie zmiennych na poziomie scenariusza. Zmienne są automatycznie kodowane URL, gdy są używane w polach zapytania lub danych formularza dla zmiennych post, ale muszą być kodowane URL ręcznie, gdy są używane w surowym post lub bezpośrednio w URL.
Nagłówki	Niestandardowe nagłówki HTTP, które zostaną wysłane podczas wykonywania żądania. Określane jako pary atrybut-wartość. Nagłówek zdefiniowany na poziomie kroku będzie używany dla tego konkretnego kroku. **Należy pamiętać, że zdefiniowanie nagłówka na poziomie kroku automatycznie odrzuca wszystkie wcześniej zdefiniowane nagłówki, z wyjątkiem domyślnego nagłówka przypisanego przez wybranie 'User-Agent' z listy rozwijanej na poziomie scenariusza.** Jednak nawet domyślny nagłówek 'User-Agent' może zostać nadpisany przez określenie go na poziomie kroku. Na przykład przypisanie nazwy nagłówkowi, ale bez ustawienia wartości, spowoduje usunięcie domyślnego nagłówka na poziomie scenariusza. Obsługiwane są makra użytkownika oraz makra {HOST.*}. To ustawia opcję cURL CURLOPT_HTTPHEADER.
Podążaj za przekierowaniami	Zaznacz pole wyboru, aby podążać za przekierowaniami HTTP. To ustawia opcję cURL CURLOPT_FOLLOWLOCATION.
Tryb pobierania	Wybierz tryb pobierania: **Body** - pobierz tylko treść z odpowiedzi HTTP **Headers** - pobierz tylko nagłówki z odpowiedzi HTTP **Body and headers** - pobierz treść i nagłówki z odpowiedzi HTTP

Parametr	Opis
Timeout	Zabbix nie poświęci więcej niż ustawiony czas na przetwarzanie URL (od jednej sekundy do maksymalnie 1 godziny). W praktyce ten parametr określa maksymalny czas na nawiązanie połączenia z URL oraz maksymalny czas wykonania żądania HTTP. Dlatego Zabbix nie poświęci więcej niż **2 x Timeout** sekund na krok. Obsługiwane są sufiksy czasu, np. 30s, 1m, 1h. Obsługiwane są makra użytkownika.
Wymagany ciąg	Wymagany wzorzec wyrażenia regularnego. Jeśli pobrana zawartość (HTML) nie pasuje do wymaganego wzorca, krok zakończy się niepowodzeniem. Jeśli pole jest puste, sprawdzanie wymaganego ciągu nie jest wykonywane. Na przykład: Homepage of Zabbix Welcome.*admin *Uwaga*: Odwoływanie się w tym polu do wyrażeń regularnych utworzonych w frontend Zabbix nie jest obsługiwane. Obsługiwane są makra użytkownika oraz makra {HOST.*}.
Wymagane kody statusu	Lista oczekiwanych kodów statusu HTTP. Jeśli Zabbix otrzyma kod, którego nie ma na liście, krok zakończy się niepowodzeniem. Jeśli pole jest puste, sprawdzanie kodów statusu nie jest wykonywane. Na przykład: 200,201,210-299 Obsługiwane są makra użytkownika.

Note:
Wszelkie zmiany w krokach scenariusza WWW zostaną zapisane tylko wtedy, gdy zostanie zapisany cały scenariusz.

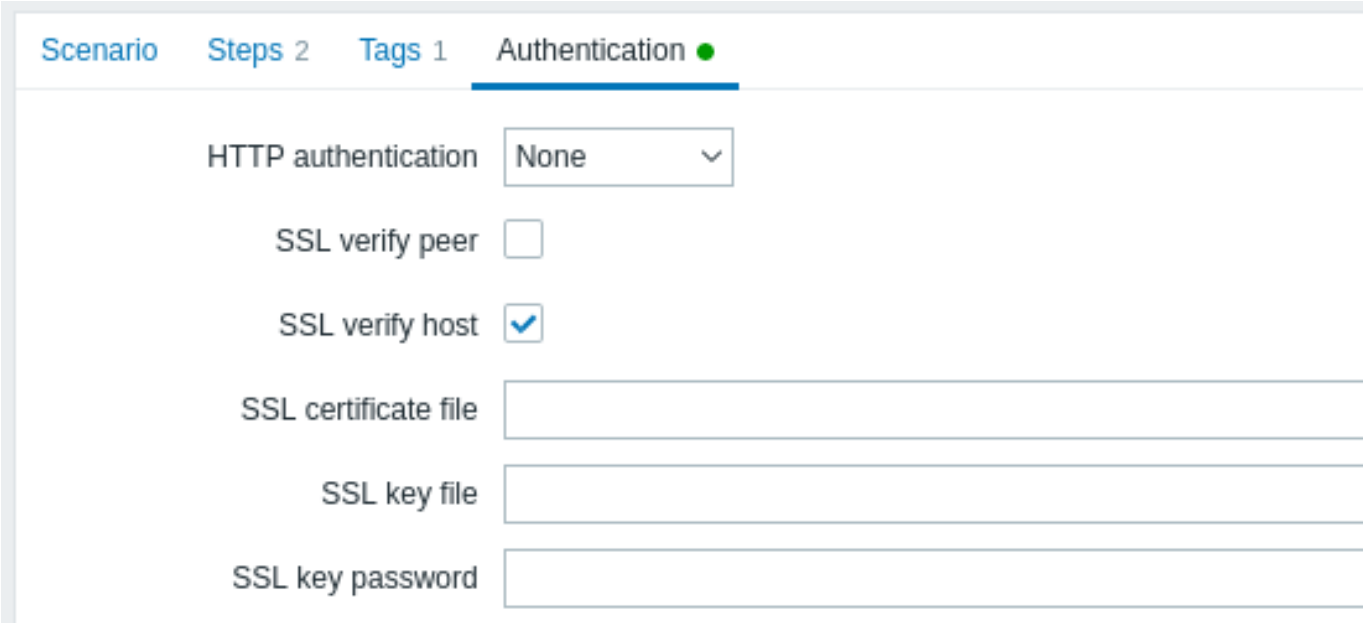
Zobacz także praktyczny przykład, jak można skonfigurować kroki monitorowania WWW.

Konfigurowanie tagów Zakładka **Tagi** umożliwia definiowanie tagów na poziomie scenariusza.



Tagowanie umożliwia filtrowanie scenariuszy WWW oraz pozycji monitorowania WWW.

Konfigurowanie uwierzytelniania Zakładka **Authentication** umożliwia skonfigurowanie opcji uwierzytelniania scenariusza. Zielona kropka obok nazwy zakładki wskazuje, że włączony jest jakiś typ uwierzytelniania HTTP.



Parametry uwierzytelniania:

Parameter	Description
HTTP authentication	Wybierz opcję uwierzytelniania: **None** — bez uwierzytelniania; **Basic** — używane jest uwierzytelnianie podstawowe; **NTLM** — używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** — używane jest uwierzytelnianie Kerberos (zobacz także: Konfigurowanie Kerberos w Zabbix); **Digest** — używane jest uwierzytelnianie Digest.
User	Wprowadź nazwę użytkownika (do 255 znaków). To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika.
Password	Wprowadź hasło użytkownika (do 255 znaków). To pole jest dostępne, jeśli *HTTP authentication* jest ustawione na Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika.
SSL verify peer	Zaznacz pole wyboru, aby weryfikować certyfikat SSL serwera WWW. Certyfikat serwera zostanie automatycznie pobrany z systemowej lokalizacji urzędu certyfikacji (CA). Lokalizację plików CA można nadpisać za pomocą parametru konfiguracyjnego serwera lub proxy Zabbix SSLCALocation. Ustawia to opcję cURL CURLOPT_SSL_VERIFYPEER.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub pole *Subject Alternate Name* certyfikatu serwera WWW jest zgodne. Ustawia to opcję cURL CURLOPT_SSL_VERIFYHOST.
SSL certificate file	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole *SSL key file* puste. Jeśli klucz jest zaszyfrowany, podaj hasło w polu *SSL key password*. Katalog zawierający ten plik jest określony przez parametr konfiguracyjny serwera lub proxy Zabbix SSLCertLocation. W tym polu można używać makr `HOST.*` i makr użytkownika. Ustawia to opcję cURL CURLOPT_SSLCERT.
SSL key file	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Katalog zawierający ten plik jest określony przez parametr konfiguracyjny serwera lub proxy Zabbix SSLKeyLocation. W tym polu można używać makr `HOST.*` i makr użytkownika. Ustawia to opcję cURL CURLOPT_SSLKEY.
SSL key password	Hasło do pliku klucza prywatnego SSL. W tym polu można używać makr użytkownika. Ustawia to opcję cURL CURLOPT_KEYPASSWD.

Attention:
[1] Zabbix obsługuje wyłącznie pliki certyfikatów i kluczy prywatnych w formacie PEM. Jeśli dane certyfikatu i klucza prywatnego są zapisane w pliku w formacie PKCS #12 (zwykle z rozszerzeniem *.p12 lub *.pfx), można wygenerować z niego plik PEM za pomocą następujących poleceń:

```
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:
Serwer Zabbix wykrywa zmiany w certyfikatach bez konieczności ponownego uruchamiania.

Note:
Jeśli certyfikat klienta i klucz prywatny znajdują się w jednym pliku, po prostu podaj go w polu ”SSL certificate file” i pozostaw pole ”SSL key file” puste. Certyfikat i klucz nadal muszą być w formacie PEM. Połączenie certyfikatu i klucza jest proste:

```
cat client.crt client.key > client.pem
```

Wyświetlanie Aby wyświetlić scenariusze WWW skonfigurowane dla hosta, przejdź do *Monitoring* → *Hosts*, odszukaj host na liście i kliknij hiperłącze *Web* w ostatniej kolumnie. Kliknij nazwę scenariusza, aby uzyskać szczegółowe informacje.



Przegląd scenariuszy WWW można również wyświetlić w *Dashboards* za pomocą widżetu monitorowania WWW.

Ostatnie wyniki wykonania scenariusza WWW są dostępne w sekcji *Monitoring* → *Latest data*.

Rozszerzone monitorowanie Czasami konieczne jest rejestrowanie otrzymanej zawartości strony HTML. Jest to szczególnie przydatne, jeśli jakiś krok scenariusza WWW zakończy się niepowodzeniem. Do tego celu służy poziom debugowania 5 (trace). Ten poziom można ustawić w plikach konfiguracyjnych serwer i proxy lub za pomocą opcji runtime control (`-R log_level_increase="http poller,N"`, gdzie N to numer procesu). Poniższe przykłady pokazują, jak można uruchomić rozszerzone monitorowanie, zakładając, że poziom debugowania 4 jest już ustawiony:

```
# Zwiększ poziom logowania wszystkich procesów http poller:
zabbix_server -R log_level_increase="http poller"

# Zwiększ poziom logowania drugiego procesu http poller:
zabbix_server -R log_level_increase="http poller,2"
```

Jeśli rozszerzone monitorowanie WWW nie jest wymagane, można je zatrzymać za pomocą opcji `-R log_level_decrease`.

1 Pozycje monitorowania WWW

Przegląd

Niektóre nowe pozycje są automatycznie dodawane do monitorowania podczas tworzenia scenariuszy WWW.

Wszystkie pozycje dziedziczą tagi ze scenariusza WWW.

Pozycje scenariusza

Gdy tylko scenariusz zostanie utworzony, Zabbix automatycznie dodaje następujące pozycje do monitorowania.

Pozycja	Opis
Prędkość pobierania dla scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o prędkości pobierania (bajty na sekundę) dla całego scenariusza, tj. średnią dla wszystkich kroków. Klucz pozycji: web.test.in[Scenario„bps] Typ: *Numeric(float)*
Nieudany krok scenariusza <Scenario>	Ta pozycja będzie wyświetlać numer kroku, który zakończył się niepowodzeniem w scenariuszu. Jeśli wszystkie kroki zostaną wykonane pomyślnie, zwracane jest 0. Klucz pozycji: web.test.fail[Scenario] Typ: *Numeric(unsigned)*
Treść ostatniego komunikatu o błędzie scenariusza <Scenario>	Ta pozycja zwraca treść ostatniego komunikatu o błędzie scenariusza. Nowa wartość jest zapisywana tylko wtedy, gdy scenariusz ma nieudany krok. Jeśli wszystkie kroki są poprawne, żadna nowa wartość nie jest zbierana. Klucz pozycji: web.test.error[Scenario] Typ: *Character*

Zamiast „Scenario” zostanie użyta rzeczywista nazwa scenariusza.

Note:
Jeśli nazwa scenariusza zawiera makra użytkownika, makra te pozostaną nierozwinięte w nazwach pozycji monitorowania WWW.

 Jeśli nazwa scenariusza zaczyna się od cudzysłowu lub zawiera przecinek albo nawias kwadratowy, zostanie poprawnie ujęta w cudzysłowy w kluczach pozycji. W pozostałych przypadkach nie zostanie zastosowane dodatkowe ujmowanie w cudzysłowy.

Note:
Pozycje monitorowania WWW są dodawane z 30-dniowym okresem przechowywania historii i 90-dniowym okresem przechowywania trendów.

Te pozycje mogą być używane do tworzenia wyzwalaczy i definiowania warunków powiadomień.

Przykład 1

Aby utworzyć wyzwalacz „Web scenario failed”, możesz zdefiniować wyrażenie wyzwalacza:

`last(/host/web.test.fail[Scenario])<>0`

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą swojego scenariusza.

Przykład 2

Aby utworzyć wyzwalacz „Web scenario failed” z użytecznym opisem problemu w nazwie wyzwalacza, możesz zdefiniować wyzwalacz o nazwie:

`Web scenario "Scenario" failed: {ITEM.VALUE}`

oraz wyrażeniu wyzwalacza:

`length(last(/host/web.test.error[Scenario]))>0 and last(/host/web.test.fail[Scenario])>0`

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą scenariusza.

Przykład 3

Aby utworzyć wyzwalacz „Aplikacja web jest wolna”, możesz zdefiniować wyrażenie wyzwalacza:

`last(/host/web.test.in[Scenario,,bps])<10000`

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą swojego scenariusza.

Pozycje kroków scenariusza

Gdy tylko krok zostanie utworzony, Zabbix automatycznie dodaje następujące pozycje do monitorowania.

Pozycja	Opis
Prędkość pobierania dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o prędkości pobierania (bajty na sekundę) dla kroku. Klucz pozycji: web.test.in[Scenario,Step,bps] Typ: *Numeric(float)*
Czas odpowiedzi dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o czasie odpowiedzi kroku w sekundach. Czas odpowiedzi jest liczony od początku żądania do momentu przesłania wszystkich informacji. Klucz pozycji: web.test.time[Scenario,Step,resp] Typ: *Numeric(float)*
Kod odpowiedzi dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać kody odpowiedzi kroku. Klucz pozycji: web.test.rspcode[Scenario,Step] Typ: *Numeric(unsigned)*

Zamiast ”Scenario” i ”Step” zostaną użyte odpowiednio rzeczywiste nazwy scenariusza i kroku.

Note:
Pozycje monitorowania WWW są dodawane z 30-dniowym okresem przechowywania historii i 90-dniowym okresem przechowywania trendów.

Note:
Jeśli nazwa scenariusza zaczyna się od cudzysłowu lub zawiera przecinek albo nawias kwadratowy, zostanie poprawnie ujęta w cudzysłowy w kluczach pozycji. W innych przypadkach nie zostanie zastosowane dodatkowe ujmowanie w cudzysłowy.

Te pozycje mogą być używane do tworzenia wyzwalaczy i definiowania warunków powiadomień. Na przykład, aby utworzyć wyzwalacz ”Logowanie do GUI Zabbix jest zbyt wolne”, można zdefiniować wyrażenie wyzwalacza:

`last(/zabbix/web.test.time[ZABBIX GUI,Login,resp])>3`

2 Scenariusz z życia wzięty

Przegląd

Ta sekcja przedstawia krok po kroku rzeczywisty przykład użycia monitorowania WWW.

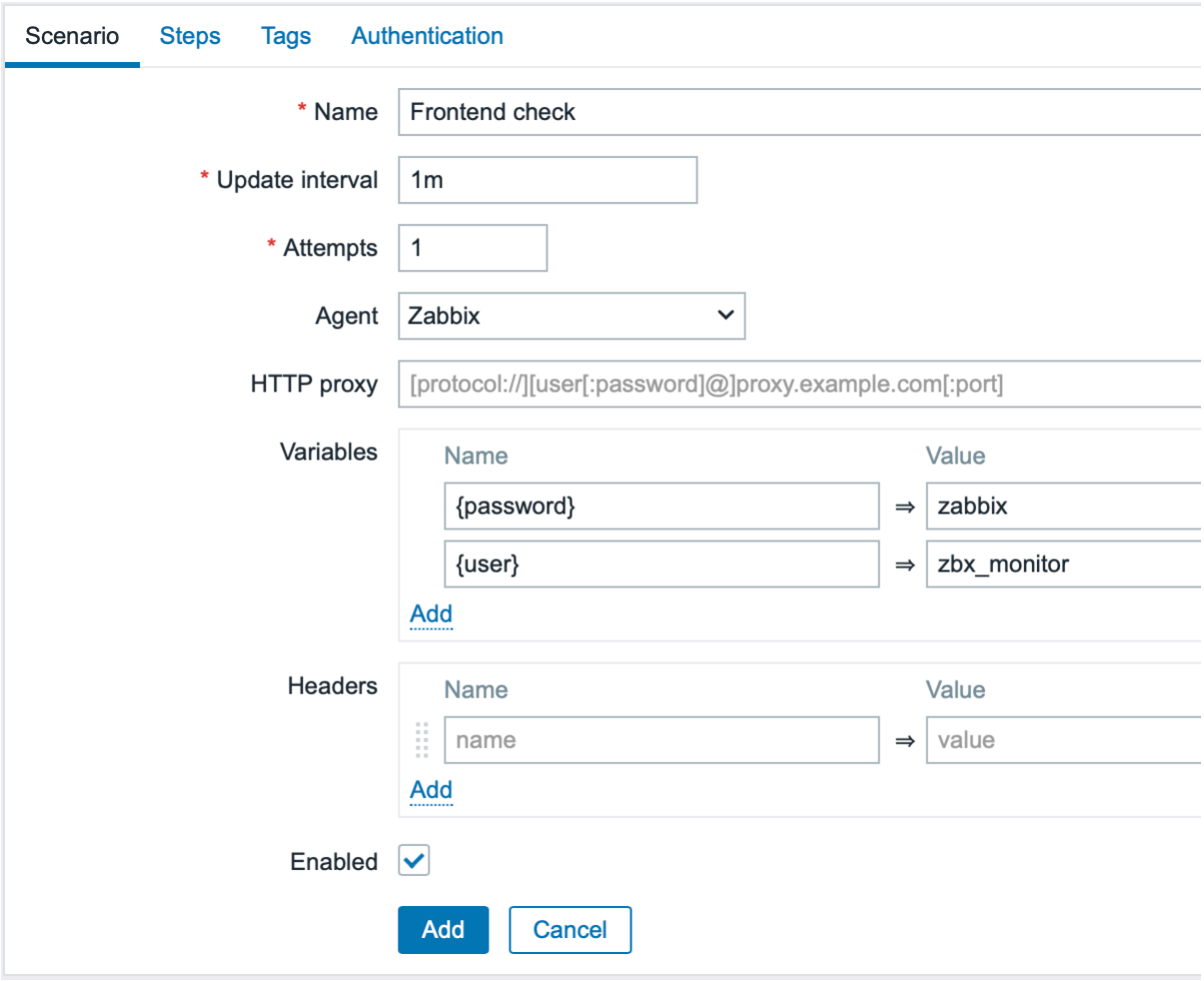
Monitorowanie WWW w Zabbix zostanie użyte do monitorowania frontend Zabbix. Celem jest określenie, czy jest on dostępny, dostarcza właściwą treść oraz jak szybko działa. Aby to osiągnąć, wymaganych jest kilka kroków, w tym sprawdzenie dostępności

pierwszej strony, zalogowanie się przy użyciu nazwy użytkownika i hasła, zweryfikowanie poprawności logowania, wylogowanie się oraz potwierdzenie wylogowania.

Scenariusz

Dodawanie nowego scenariusza WWW

Przejdź do *Zbieranie danych* → *Hosty*, wybierz host i kliknij *WWW* w wierszu tego hosta.
Następnie kliknij *Utwórz scenariusz WWW*.



W formularzu nowego scenariusza uzupełnij następujące pola:

- **Nazwa** - Frontend check
- **Interwał aktualizacji** - 1m
- **Próby** - 1
- **Agent** - Zabbix

W sekcji *Zmienne* dodaj dwie zmienne: *{password}* oraz *{user}*.
Jako wartości wprowadź dane logowania istniejącego użytkownika Zabbix.

Ze względów bezpieczeństwa zaleca się utworzenie osobnego użytkownika z minimalnymi uprawnieniami, który będzie używany do celów monitorowania.

Opcjonalnie przejdź do zakładki *Tagi* i dodaj tagi scenariusza WWW.

Note:
Po pełnej konfiguracji ten scenariusz WWW automatycznie doda do hosta pozycja typu Zabbix trapper.
Możesz używać tagów scenariusza WWW do szybkiej identyfikacji powiązanych pozycji i wyzwalaczy lub do przeszukiwania zebranych danych.
Na przykład odpowiednimi tagami dla tego samouczka są `component: web-scenario` i/lub `target: frontend`.

Konfiguracja kroków scenariusza web

Przejdź do zakładki *Steps* i zdefiniuj kroki dla scenariusza. Kliknij przycisk *Add*, aby dodać pojedynczy krok.

Wspólne pola

Dla każdego kroku opisanego poniżej wypełnij następujące pola oprócz pól specyficznych dla danego kroku:

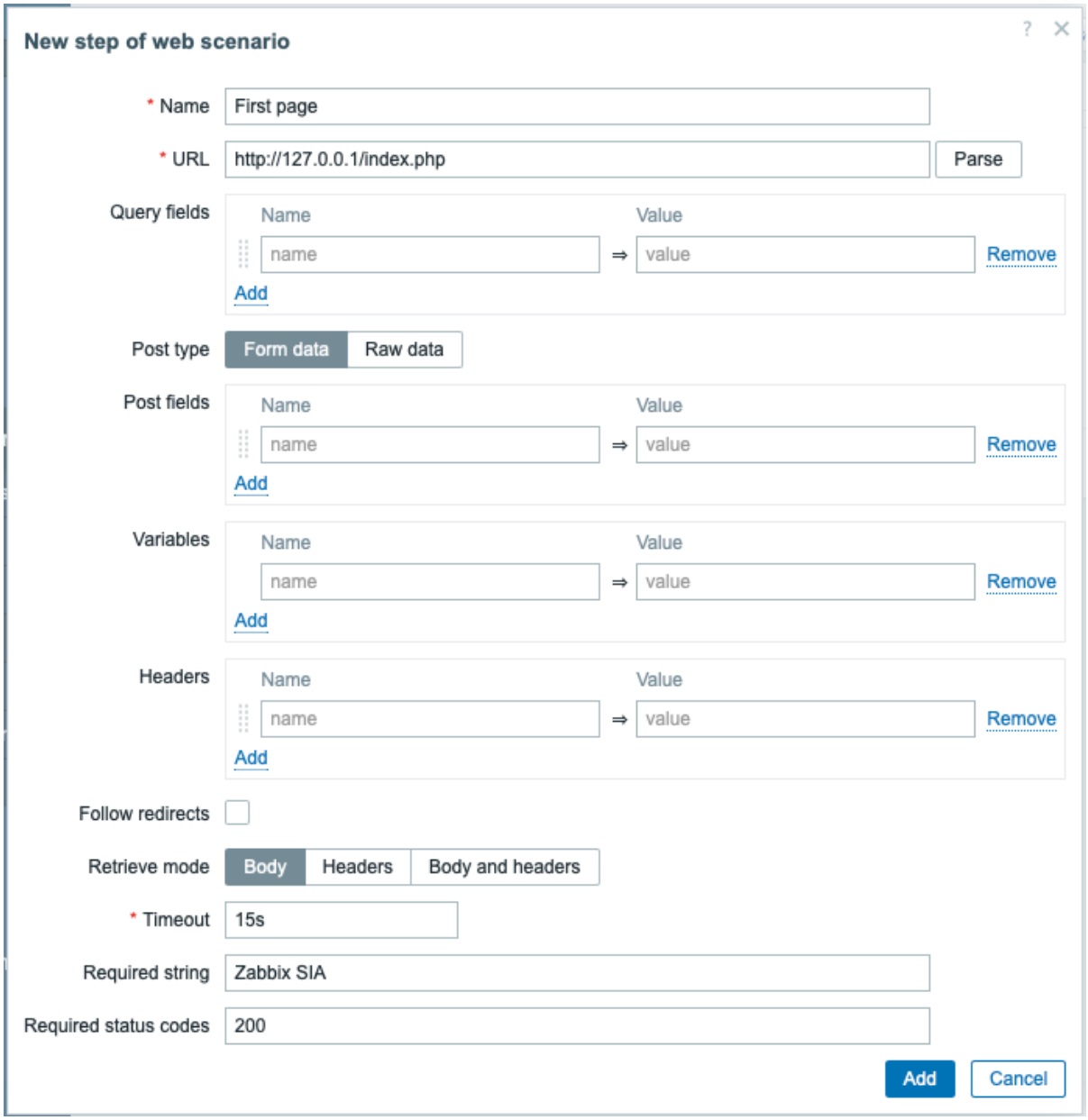
- **URL** - adres URL frontend Zabbix
- **Timeout** - 15s
- **Required status codes** - 200

Krok 1 scenariusza WWW

Sprawdź, czy pierwsza strona odpowiada poprawnie, zwraca kod odpowiedzi HTTP 200 i zawiera tekst „Zabbix SIA”.

- W polu **Nazwa** wpisz *Pierwsza strona*.
- W polu **Wymagany ciąg** wpisz *Zabbix SIA*.
- Wypełnij wspólne pola.

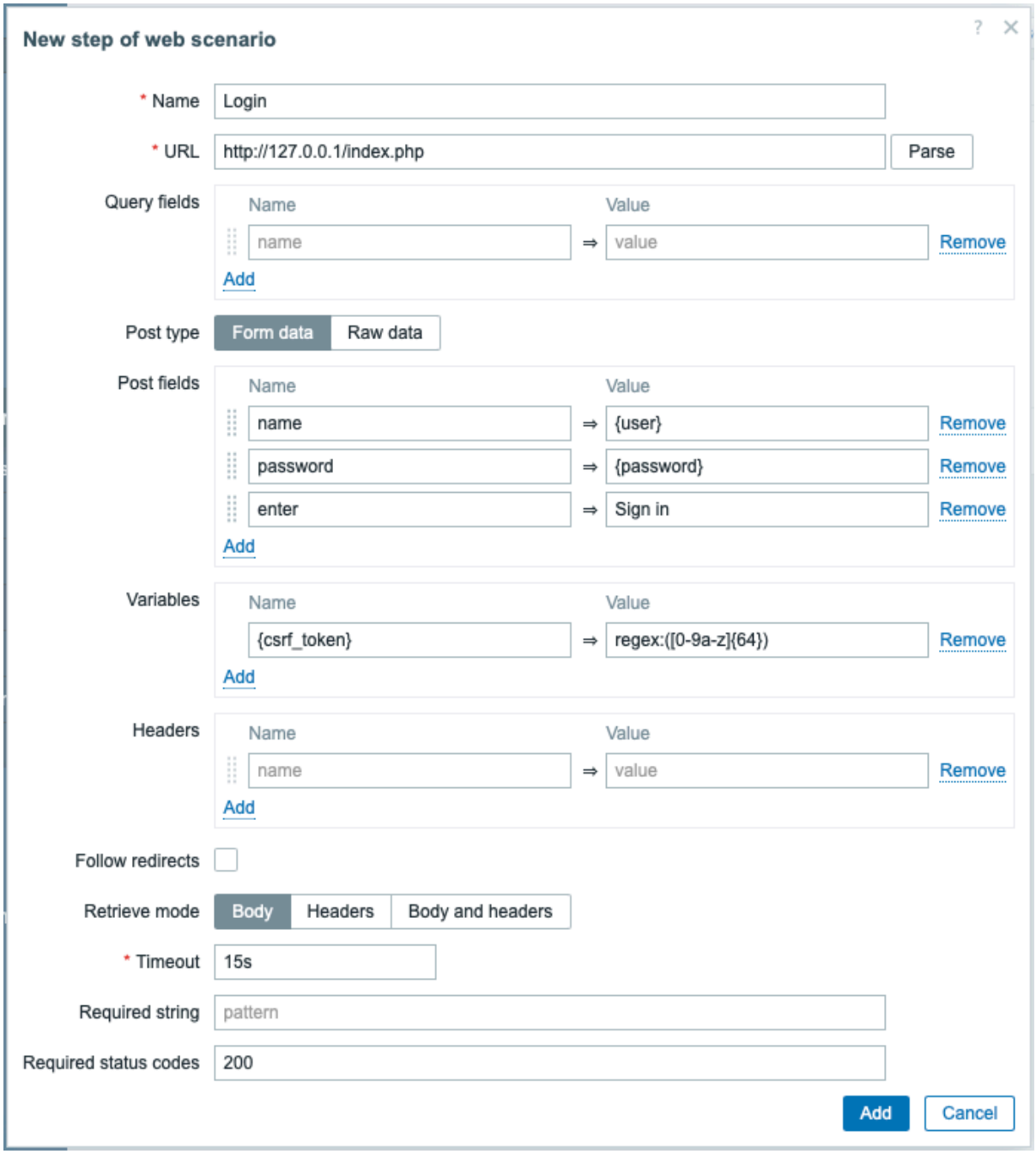
Po zakończeniu konfiguracji kroku naciśnij przycisk *Dodaj*.



Krok 2 scenariusza WWW

Zaloguj się do frontend Zabbix, używając makr (zmiennych) zdefiniowanych na poziomie scenariusza - *{user}* i *{password}*.

- W polu **Name** wpisz *Login*.
- W sekcji **Post fields** dodaj trzy pola post:
  - *name* z wartością *{user}*
  - *password* z wartością *{password}*
  - *enter* z wartością *Sign in*
- W sekcji **Variables** dodaj nową zmienną *{csrf_token}* z wartością *regex:([0-9a-z]{64})*. Ta zmienna przechwyci wartość przypisanego tokena CSRF, aby użyć jej ponownie w kroku 4.
- Uzupełnij wspólne pola.

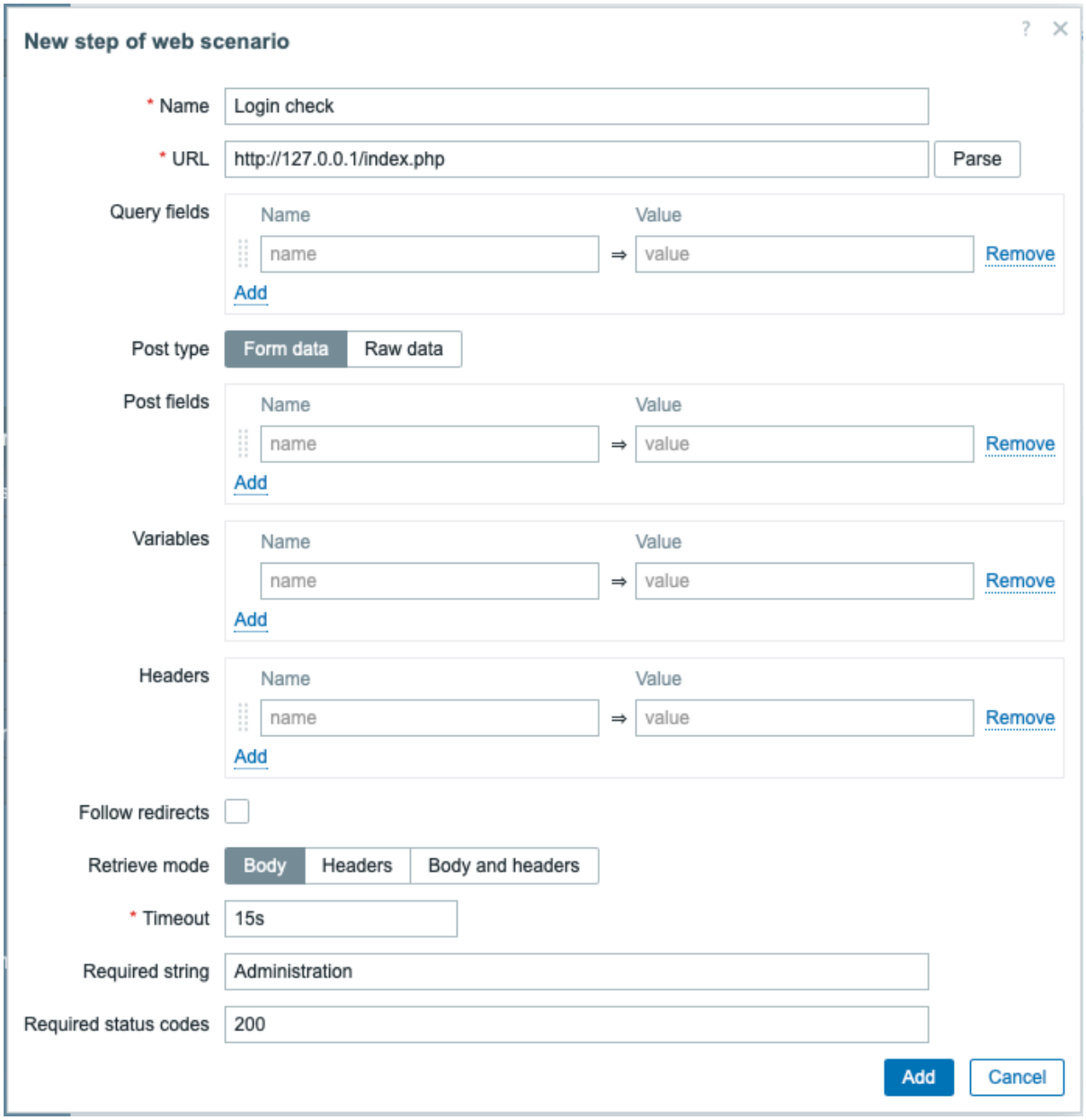


Attention:
Zwróć uwagę, że frontend Zabbix używa przekierowania JavaScript podczas logowania, dlatego logowanie musi nastąpić najpierw, a funkcje dostępne po zalogowaniu można sprawdzać dopiero w kolejnych krokach. Dodatkowo krok logowania musi używać pełnego adresu URL do pliku **index.php**.

Krok 3 scenariusza WWW

Po zalogowaniu zweryfikuj powodzenie, sprawdzając obecność ciągu znaków widocznego tylko po zalogowaniu — na przykład *Administration*.

- W polu **Nazwa** wprowadź *Login check*.
- W polu **Wymagany ciąg znaków** wprowadź *Administration*.
- Wypełnij wspólne pola.

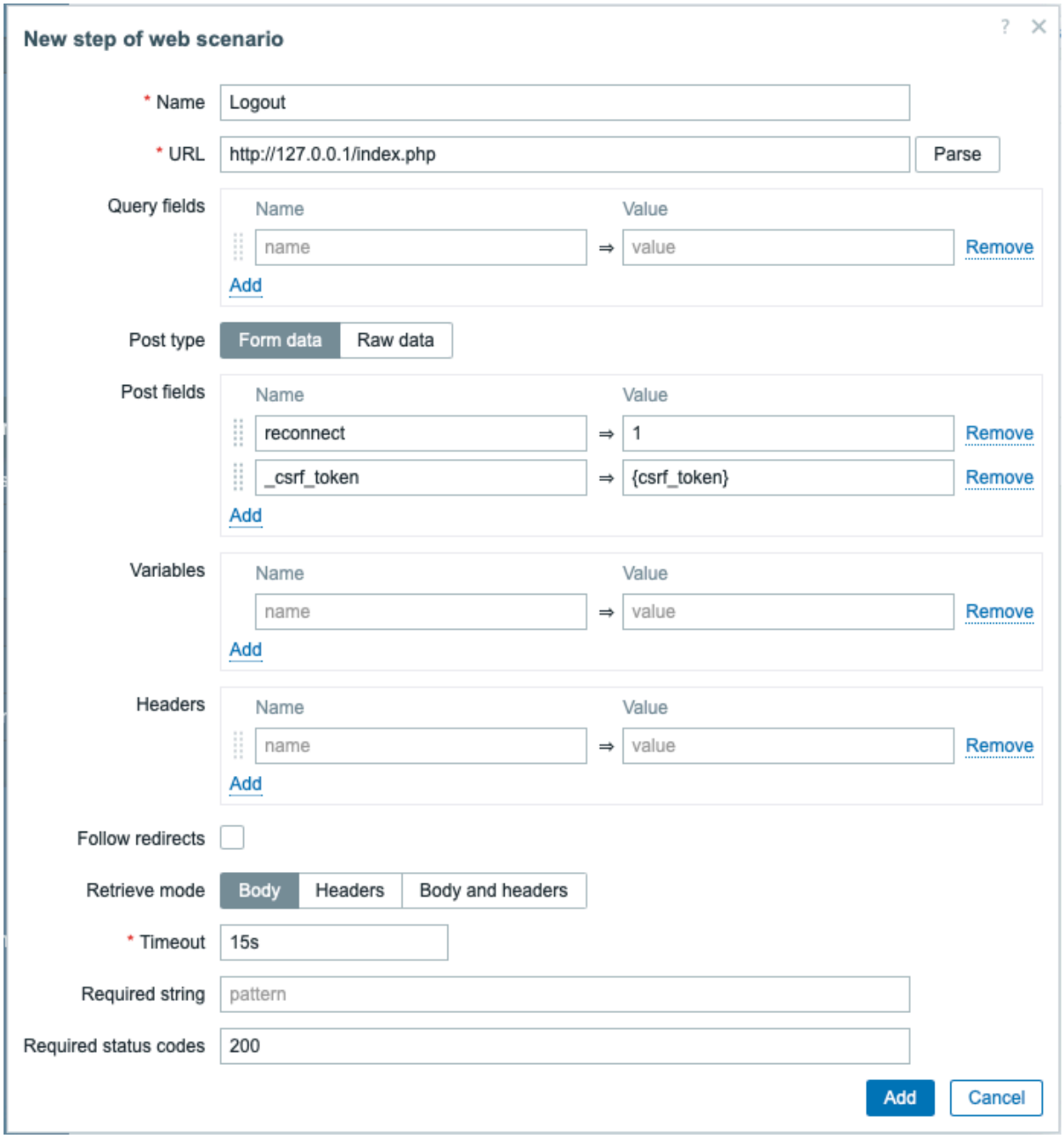


Krok 4 scenariusza WWW

Po zweryfikowaniu dostępności frontend oraz logowania dodaj krok wylogowania — w przeciwnym razie baza danych Zabbix będzie zaśmiecona wieloma rekordami otwartych sesji.

- W polu **Name** wpisz *Logout*.
- W sekcji **Post fields** dodaj dwa pola post:
  - *reconnect* o wartości *1*
  - *_csrf_token* o wartości *{csrf_token}*.
- Wypełnij wspólne pola.

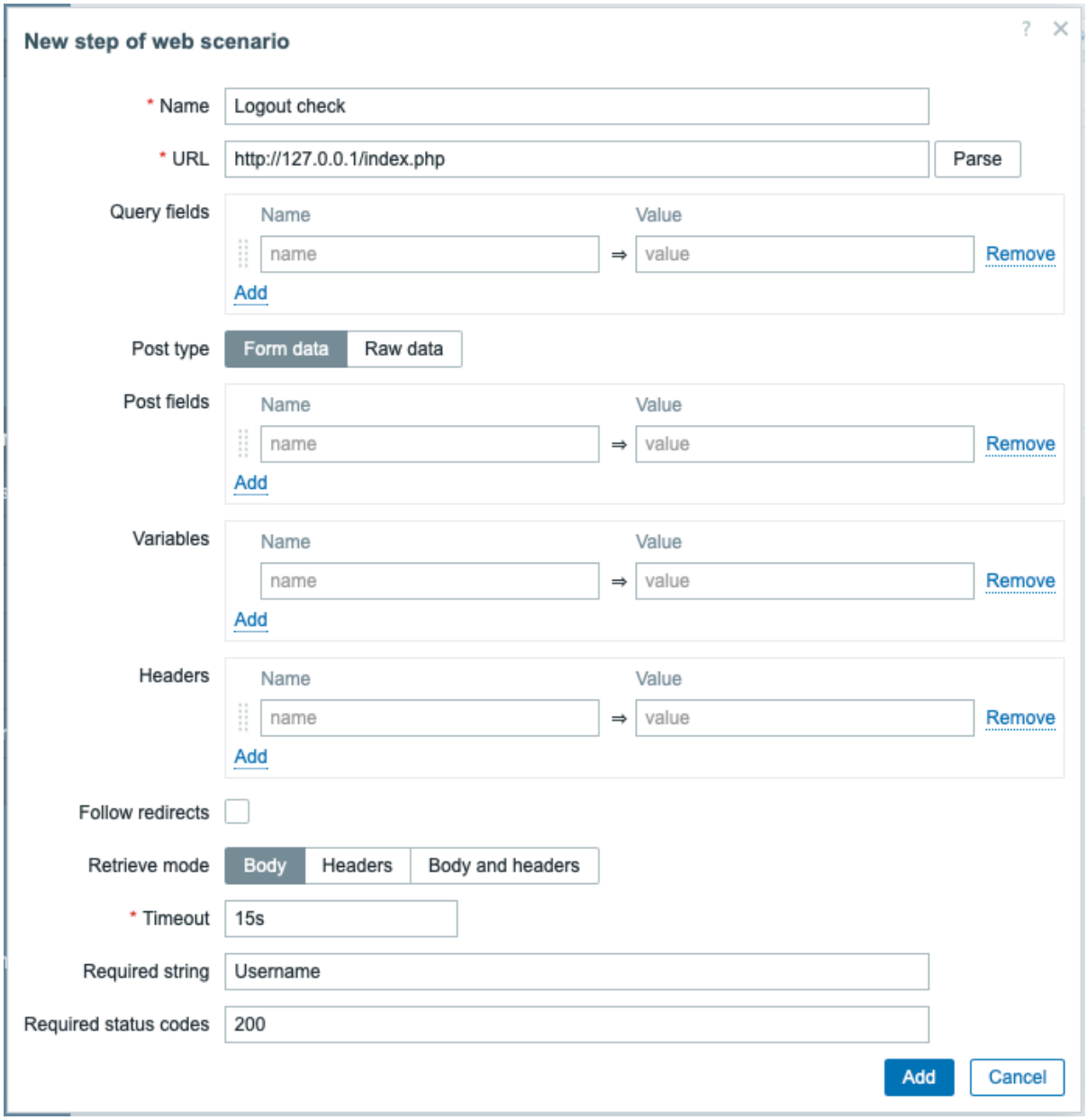
Ten krok używa zmiennej {csrf_token} uzyskanej w kroku 2



Krok 5 scenariusza WWW

Aby potwierdzić wylogowanie, sprawdź ciąg **Username**.

- W polu **Nazwa** wprowadź *Logout check*.
- W polu **Wymagany ciąg** wprowadź *Username*.
- Uzupełnij wspólne pola.



Pełna konfiguracja kroków

Pełna konfiguracja kroków scenariusza WWW powinna wyglądać następująco:

	Name	Timeout	URL	Required	Status codes	Action
1:	First page	15s	http://127.0.0.1/index.php	Zabbix SIA	200	Remove
2:	Log in	15s	http://127.0.0.1/index.php		200	Remove
3:	Log in Check	15s	http://127.0.0.1/index.php	Administrati…	200	Remove
4:	Log out	15s	http://127.0.0.1/index.php		200	Remove
5:	Log out check	15s	http://127.0.0.1/index.php	Username	200	Remove

Add

Sprawdź wyniki

Zapisz ukończony scenariusz monitorowania WWW.

Scenariusz zostanie dodany do hosta. Aby wyświetlić informacje o scenariuszu WWW, przejdź do *Monitoring* → *Hosts*, znajdź host na liście i kliknij hiperłącze Web w ostatniej kolumnie.



Kliknij nazwę scenariusza, aby zobaczyć bardziej szczegółowe statystyki:



10 Monitorowanie maszyn wirtualnych

Przegląd Zabbix może używać reguł odkrywania niskiego poziomu do automatycznego wykrywania hiperwizorów VMware i maszyn wirtualnych oraz tworzenia hostów do ich monitorowania na podstawie wstępnie zdefiniowanych prototypów hostów.

Zabbix zawiera również gotowe do użycia szablony do monitorowania VMware vCenter lub hiperwizorów ESXi.

Minimalna wymagana wersja VMware vCenter lub vSphere to 5.1.

Zbieranie danych Monitorowanie maszyn wirtualnych składa się z dwóch etapów:

1. Procesy Zabbix *vmware collector* przetwarzają dane maszyn wirtualnych — procesy te pobierają niezbędne informacje z usług sieciowych VMware za pośrednictwem protokołu SOAP, wstępnie je przetwarzają i zapisują we współdzielonej pamięci serwera Zabbix.

2. Procesy Zabbix *poller* pobierają dane przy użyciu prostego sprawdzenia Zabbix klucze pozycji monitorowania VMware.

Zabbix dzieli zebrane dane na dane konfiguracyjne VMware oraz dane liczników wydajności VMware. Oba typy danych są zbierane niezależnie przez procesy *vmware collector*.

Na podstawie informacji z liczników wydajności VMware dostępne są następujące statystyki:

- Magazyn danych
- Urządzenie dyskowe
- CPU
- Zasilanie
- Interfejs sieciowy
- Niestandardowe pozycje liczników wydajności

Pełną listę pozycji pobierających dane z liczników wydajności VMware można znaleźć w *kluczach pozycji monitorowania VMware*.

Należy pamiętać, że częstotliwość pobierania zdarzeń VMware zależy od interwału odpytywania vmware.eventlog, ale nie może być mniejsza niż 5 sekund.

Konfiguracja Jeśli serwer Zabbix jest skompilowany ze źródeł, musi być skompilowany z opcjami konfiguracji `--with-libcurl --with-libxml2`, aby włączyć monitorowanie maszyn wirtualnych. Pakiety Zabbix są już skompilowane z włączonymi tymi opcjami.

Następujące parametry pliku konfiguracyjnego serwera Zabbix można zmodyfikować na potrzeby monitorowania maszyn wirtualnych:

- `StartVMwareCollectors`

Note:
Zaleca się włączenie większej liczby kolektorów niż liczba monitorowanych usług VMware; w przeciwnym razie pobieranie statystyk liczników wydajności VMware może zostać opóźnione przez pobieranie danych konfiguracyjnych VMware (co w przypadku dużych instalacji trwa pewien czas).

 Ogólnie wartość `StartVMwareCollectors` nie powinna być mniejsza niż 2 ani większa niż dwukrotność liczby monitorowanych usług VMware: Liczba usług < `StartVMwareCollectors` < (Liczba usług * 2). Na przykład podczas monitorowania jednej usługi VMware ustaw `StartVMwareCollectors` na 2; podczas monitorowania trzech usług ustaw `StartVMwareCollectors` na 5.

 Należy pamiętać, że wymagana liczba kolektorów zależy również od zakresu środowiska VMware oraz od parametrów konfiguracyjnych `VMwareFrequency` i `VMwarePerfFrequency`.

- `VMwareCacheSize`
- `VMwareFrequency`
- `VMwarePerfFrequency`
- `VMwareTimeout`

Attention:
Aby obsługiwać metryki pojemności magazynu danych, upewnij się, że wartość klucza VMware `vpxd.stats.maxQueryMetrics` jest ustawiona na co najmniej 64. Więcej informacji można znaleźć w artykule *VMware Knowledge Base*.

Wykrywanie

Zabbix może używać reguł wykrywania niskiego poziomu (na przykład *vmware.hv.discovery[{$VMWARE.URL}]*) do automatycznego wykrywania hiperwizorów VMware i maszyn wirtualnych. Ponadto Zabbix może używać prototypów hostów do automatycznego generowania rzeczywistych hostów dla wykrytych obiektów. Aby uzyskać więcej informacji, zobacz *Prototypy hostów*.

Configuration examples

For a basic example of how to set up Zabbix for monitoring VMware using the *VMware FQDN* template, see *Monitor VMware with Zabbix*.

For a more detailed example of how to create a host, a low-level discovery rule, and a host prototype for monitoring VMware, see *Setup example*.

Rozszerzone logowanie Dane zbierane przez procesy *vmware collector* mogą być zapisywane w logach na potrzeby szczegółowego debugowania przy użyciu poziomu debugowania 5. Poziom debugowania można skonfigurować w plikach konfiguracyjnych serwer i proxy lub za pomocą opcji kontroli w czasie działania `-R log_level_increase="vmware collector,N"`, gdzie „N” oznacza numer procesu.

Na przykład, aby zwiększyć poziom debugowania z 4 do 5 dla wszystkich procesów *vmware collector*, uruchom następujące polecenie:

```
zabbix_server -R log_level_increase="vmware collector"
```

Aby zwiększyć poziom debugowania z 4 do 5 dla drugiego procesu *vmware collector*, uruchom następujące polecenie:

```
zabbix_server -R log_level_increase="vmware collector,2"
```

Gdy rozszerzone logowanie danych kolektora VMware nie jest już wymagane, zaleca się zmniejszenie poziomu debugowania do wartości domyślnej (3) przez uruchomienie polecenia `-R log_level_decrease`.

Rozwiązywanie problemów

- W przypadku niedostępnych metryk upewnij się, że nie są one domyślnie niedostępne lub wyłączone w nowszych wersjach VMware vSphere oraz że nie nałożono żadnych limitów na zapytania do bazy danych metryk wydajności. Więcej informacji można znaleźć w ZBX-12094.
- Jeśli występuje błąd `config.vpxd.stats.maxQueryMetrics` is invalid or exceeds the maximum number of characters permitted, dodaj parametr `config.vpxd.stats.maxQueryMetrics` do ustawień vCenter Server. Wartość tego parametru powinna być taka sama jak wartość `maxQuerysize` w pliku *web.xml* VMware. Więcej informacji można znaleźć w artykule *VMware Knowledge Base*.
- Jeśli podejrzewasz, że twoja instalacja Zabbix używa zbyt dużo pamięci, zobacz Profiling excessive memory usage with tcmalloc.

1 Klucze pozycji monitorowania VMware

Przegląd Ta strona zawiera szczegółowe informacje o prostych kontrolach, które mogą być używane do monitorowania środowisk VMware. Metryki są pogrupowane według celu monitorowania.

Obsługiwane klucze pozycji Klucze pozycji są wymienione bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Item key	Description	Item group
vmware.eventlog	Dziennik zdarzeń VMware.	Ogólna usługa
vmware.fullname	Pełna nazwa usługi VMware.	
vmware.version	Wersja usługi VMware.	
vmware.cl.perfcounter	Metryki licznika wydajności klastra VMware.	Klaster
vmware.cluster.alarms.get	Dane alarmów klastra VMware.	
vmware.cluster.discovery	Wykrywanie klastrów VMware.	
vmware.cluster.property	Właściwość klastra VMware.	
vmware.cluster.status	Stan klastra VMware.	
vmware.cluster.tags.get	Tablica tagów klastra VMware.	
vmware.datastore.alarms.get	Dane alarmów magazynu danych VMware.	Magazyn danych
vmware.datastore.discovery	Wykrywanie magazynów danych VMware.	
vmware.datastore.hv.list	Lista hypervisorów magazynu danych.	
vmware.datastore.perfcounter	Wartość licznika wydajności magazynu danych VMware.	
vmware.datastore.property	Właściwość magazynu danych VMware.	
vmware.datastore.read	Czas trwania operacji odczytu z magazynu danych.	
vmware.datastore.size	Przestrzeń magazynu danych VMware w bajtach lub jako procent całości.	
vmware.datastore.tags.get	Tablica tagów magazynu danych VMware.	
vmware.datastore.write	Czas trwania operacji zapisu do magazynu danych.	
vmware.dc.alarms.get	Dane alarmów centrum danych VMware.	Centrum danych
vmware.dc.discovery	Wykrywanie centrów danych VMware.	
vmware.dc.tags.get	Tablica tagów centrum danych VMware.	

Item key	Description	Item group
vmware.dvswitch.discovery	Wykrywanie przełączników VMware vSphere Distributed Switch.	vSphere Distributed Switch
vmware.dvswitch.fetchportsget	Dane portów VMware vSphere Distributed Switch.	
vmware.hv.alarms.get	Dane alarmów hypervisora VMware.	Hypervisor
vmware.hv.cluster.name	Nazwa klastra hypervisora VMware.	
vmware.hv.connectionstate	Stan połączenia hypervisora VMware.	
vmware.hv.cpu.usage	Użycie procesora hypervisora VMware (Hz).	
vmware.hv.cpu.usage.perf	Użycie procesora hypervisora VMware jako wartość procentowa w danym przedziale czasu.	
vmware.hv.cpu.utilization	Użycie procesora hypervisora VMware jako wartość procentowa w danym przedziale czasu, zależne od zarządzania energią lub HT.	
vmware.hv.datacenter.name	Nazwa centrum danych hypervisora VMware.	
vmware.hv.datastore.discovery	Wykrywanie magazynów danych hypervisora VMware.	
vmware.hv.datastore.list	Lista magazynów danych hypervisora VMware.	
vmware.hv.datastore.multipath	Liczba dostępnych ścieżek magazynu danych.	
vmware.hv.datastore.read	Średni czas trwania operacji odczytu z magazynu danych.	
vmware.hv.datastore.size	Przestrzeń magazynu danych VMware w bajtach lub jako procent całości.	
vmware.hv.datastore.write	Średni czas trwania operacji zapisu do magazynu danych.	
vmware.hv.discovery	Wykrywanie hypervisorów VMware.	
vmware.hv.diskinfo.get	Dane dysków hypervisora VMware.	
vmware.hv.fullname	Pełna nazwa produktu, łącznie z informacją o wersji.	
vmware.hv.hw.cpu.freq	Częstotliwość procesora hypervisora VMware.	
vmware.hv.hw.cpu.model	Model procesora hypervisora VMware.	
vmware.hv.hw.cpu.num	Liczba rdzeni procesora w hypervisorze VMware.	
vmware.hv.hw.cpu.threads	Liczba wątków procesora w hypervisorze VMware.	
vmware.hv.hw.memory	Całkowity rozmiar pamięci hypervisora VMware.	
vmware.hv.hw.model	Model hypervisora VMware.	
vmware.hv.hw.sensors.get	Wartość czujników sprzętowych hypervisora VMware.	
vmware.hv.hw.serialnumber	Numer seryjny hypervisora VMware.	
vmware.hv.hw.uuid	UUID BIOS hypervisora VMware.	
vmware.hv.hw.vendor	Nazwa dostawcy hypervisora VMware.	
vmware.hv.maintenance	Stan konserwacji hypervisora VMware.	
vmware.hv.memory.size.ballooned	Rozmiar pamięci ballooned hypervisora VMware.	
vmware.hv.memory.used	Rozmiar używanej pamięci hypervisora VMware.	
vmware.hv.net.if.discovery	Wykrywanie interfejsów sieciowych hypervisora VMware.	
vmware.hv.network.in	Statystyki ruchu przychodzącego sieci hypervisora VMware.	
vmware.hv.network.linkspeed	Prędkość interfejsu sieciowego hypervisora VMware.	
vmware.hv.network.out	Statystyki ruchu wychodzącego sieci hypervisora VMware.	
vmware.hv.perfcounter	Wartość licznika wydajności hypervisora VMware.	
vmware.hv.property	Właściwość hypervisora VMware.	
vmware.hv.power	Zużycie energii hypervisora VMware.	
vmware.hv.sensor.health.state	Zbiorczy czujnik stanu kondycji hypervisora VMware.	
vmware.hv.sensors.get	Czujniki stanu dostawcy sprzętu hypervisora VMware.	
vmware.hv.status	Stan hypervisora VMware.	
vmware.hv.tags.get	Tablica tagów hypervisora VMware.	
vmware.hv.uptime	Czas pracy hypervisora VMware.	
vmware.hv.version	Wersja hypervisora VMware.	
vmware.hv.vm.num	Liczba maszyn wirtualnych w hypervisorze VMware.	
vmware.rp.cpu.usage	Użycie CPU w hercach w danym przedziale czasu w puli zasobów VMware.	Pula zasobów
vmware.rp.memory	Metryki pamięci puli zasobów VMware.	
vmware.alarms.get	Dane alarmów wirtualnego centrum VMware.	Wirtualne centrum
vmware.vm.alarms.get	Dane alarmów maszyny wirtualnej VMware.	Maszyna wirtualna
vmware.vm.attribute	Wartość niestandardowego atrybutu maszyny wirtualnej VMware.	
vmware.vm.cluster.name	Nazwa maszyny wirtualnej VMware.	
vmware.vm.consolidationneeded	Dysk maszyny wirtualnej VMware wymaga konsolidacji.	
vmware.vm.cpu.latency	Procent czasu, w którym maszyna wirtualna nie może działać, ponieważ konkuruje o dostęp do fizycznych procesorów CPU.	

Item key	Description	Item group
vmware.vm.cpu.num	Liczba procesorów w maszynie wirtualnej VMware.	
vmware.vm.cpu.readiness	Procent czasu, w którym maszyna wirtualna była gotowa do działania, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU.	
vmware.vm.cpu.ready	Czas, w którym maszyna wirtualna była gotowa do działania, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU.	
vmware.vm.cpu.swapwait	Procent czasu CPU spędzonego na oczekiwaniu na swap-in.	
vmware.vm.cpu.usage	Użycie procesora maszyny wirtualnej VMware (Hz).	
vmware.vm.cpu.usage.perf	Użycie procesora maszyny wirtualnej VMware jako wartość procentowa w danym przedziale czasu.	
vmware.vm.datacenter.name	Nazwa centrum danych maszyny wirtualnej VMware.	
vmware.vm.discovery	Wykrywanie maszyn wirtualnych VMware.	
vmware.vm.guest.memory.swapped	Ilość fizycznej pamięci gościa przeniesionej do przestrzeni wymiany.	
vmware.vm.guest.osuptime	Całkowity czas, jaki upłynął od ostatniego uruchomienia systemu operacyjnego.	
vmware.vm.hv.maintenance	Stan konserwacji hypervisora maszyny wirtualnej VMware.	
vmware.vm.hv.name	Nazwa hypervisora maszyny wirtualnej VMware.	
vmware.vm.memory.size	Całkowity rozmiar pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.ballooned	Rozmiar pamięci ballooned maszyny wirtualnej VMware.	
vmware.vm.memory.size.compressed	Rozmiar skompresowanej pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.consumed	Ilość fizycznej pamięci hosta zużytej na podtrzymanie stron fizycznej pamięci gościa.	
vmware.vm.memory.size.private	Rozmiar pamięci prywatnej maszyny wirtualnej VMware.	
vmware.vm.memory.size.shared	Rozmiar pamięci współdzielonej maszyny wirtualnej VMware.	
vmware.vm.memory.size.swapped	Rozmiar pamięci wymiany maszyny wirtualnej VMware.	
vmware.vm.memory.size.usage.guest	Użycie pamięci gościa maszyny wirtualnej VMware.	
vmware.vm.memory.size.usage.host	Użycie pamięci hosta maszyny wirtualnej VMware.	
vmware.vm.memory.usage	Procent fizycznej pamięci hosta, który został zużyty.	
vmware.vm.net.if.discovery	Wykrywanie interfejsów sieciowych maszyny wirtualnej VMware.	
vmware.vm.net.if.in	Statystyki ruchu przychodzącego interfejsu sieciowego maszyny wirtualnej VMware.	
vmware.vm.net.if.out	Statystyki ruchu wychodzącego interfejsu sieciowego maszyny wirtualnej VMware.	
vmware.vm.net.if.usage	Wykorzystanie sieci maszyny wirtualnej VMware w danym przedziale czasu.	
vmware.vm.perfcounter	Wartość licznika wydajności maszyny wirtualnej VMware.	
vmware.vm.powerstate	Stan zasilania maszyny wirtualnej VMware.	
vmware.vm.property	Właściwość maszyny wirtualnej VMware.	
vmware.vm.snapshot.get	Stan migawki maszyny wirtualnej VMware.	
vmware.vm.state	Stan maszyny wirtualnej VMware.	
vmware.vm.storage.committed	Przydzielona przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.readoio	Średnia liczba oczekujących żądań odczytu do dysku wirtualnego w czasie przedziału zbierania danych.	
vmware.vm.storage.totalreadlatency	Średni czas trwania odczytu z dysku wirtualnego.	
vmware.vm.storage.totalwritelatency	Średni czas trwania zapisu do dysku wirtualnego.	
vmware.vm.storage.uncommitted	Nieprzydzielona przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.unshared	Niewspółdzielona przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.writeoio	Średnia liczba oczekujących żądań zapisu do dysku wirtualnego w czasie przedziału zbierania danych.	
vmware.vm.tags.get	Tablica tagów maszyny wirtualnej VMware.	
vmware.vm.tools	Stan lub wersja narzędzi gościa maszyny wirtualnej VMware.	
vmware.vm.uptime	Czas pracy maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.discovery	Wykrywanie urządzeń dyskowych maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.read	Statystyki odczytu urządzenia dyskowego maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.write	Statystyki zapisu urządzenia dyskowego maszyny wirtualnej VMware.	
vmware.vm.vfs.fs.discovery	Wykrywanie systemów plików maszyny wirtualnej VMware.	
vmware.vm.vfs.fs.size	Statystyki systemu plików maszyny wirtualnej VMware.	

Szczegóły klucza pozycji Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

vmware.eventlog[url,<mode>,<severity>]

 Dziennik zdarzeń VMware.
 Zwracana wartość: *Log*.

Parametry:

- **url** - URL usługi VMware;
- **mode** - *all* (domyślnie) lub *skip* - pomiń przetwarzanie starszych danych;
- **severity** - filtrowanie według poziomu ważności: *error*, *warning*, *info* lub *user*. Ten parametr musi być ujęty w cudzysłów, jeśli określono więcej niż jeden poziom ważności na liście rozdzielanej przecinkami (np. `"error,warning,info,user"`). Domyślnie wyłączone.

Uwagi:

- Dla każdego URL może istnieć tylko jeden klucz pozycji `vmware.eventlog`;
- Zobacz także przykład filtrowania rekordów dziennika zdarzeń VMware.

vmware.fullnameurl

 Pełna nazwa usługi VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware.

vmware.versionurl

 Wersja usługi VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - URL usługi VMware.

vmware.cl.perfcounter[url,id,path,<instance>]

 Metryki licznika wydajności klastra VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - URL usługi VMware;
- **id** - ID klastra VMware. `id` można uzyskać z `vmware.cluster.discovery[]` jako {#CLUSTER.ID}.
- **path** - ścieżka licznika wydajności[1];
- **instance** - instancja licznika wydajności.

vmware.cluster.alarms.get[url,id]

 Dane alarmów klastra VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator klastra VMware.

vmware.cluster.discoveryurl

 Wykrywanie klastrów VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.cluster.property[url,id,prop]

 Właściwość klastra VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **id** — identyfikator klastra VMware;
- **prop** — ścieżka właściwości, będąca nazwą właściwości obiektu VM zgodnie z definicją w VMware SDK.

Przykłady:

```
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},overallStatus]
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},runtime.powerState]
```

vmware.cluster.status[url,name]

 Status klastra VMware.
 Zwracana wartość: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **name** - nazwa klastra VMware.

vmware.cluster.tags.get[url,id]

 Tablica tagów klastra VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator klastra VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.datastore.alarms.get[url,uuid]

 Dane alarmów magazynu danych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator magazynu danych VMware.

vmware.datastore.discovery[url,<filter_uuid>]

 Odkrywanie magazynów danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **filter_uuid** — pojedynczy unikalny identyfikator HV lub VM (domyślnie: pusty). Jeśli zostanie określony, zostaną odkryte tylko powiązane magazyny danych.

vmware.datastore.hv.list[url,datastore]

 Lista hypervisorów datastore.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **datastore** - UUID lub nazwa datastore.

Przykład danych wyjściowych:

```
esx7-01-host.zabbix.sandbox
esx7-02-host.zabbix.sandbox
```

vmware.datastore.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności magazynu danych VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalnie unikalny identyfikator magazynu danych VMware;
- **path** — ścieżka licznika wydajności[1];
- **instance** — instancja licznika wydajności. Dla wartości zagregowanych (domyślnie) użyj pustej instancji. Wartość `instance` można otrzymać z `vmware.datastore.discovery[]` jako część tablicy {#DATASTORE.EXTENT}.

vmware.datastore.property[url,uuid,prop]

 Właściwość magazynu danych VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator magazynu danych VMware;
- **prop** — ścieżka właściwości.

vmware.datastore.read[url,datastore,<mode>]

 Czas trwania operacji odczytu z datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (wartość średnia, domyślnie) lub *maxlatency* (wartość maksymalna).

vmware.datastore.size[url,datastore,<mode>]

 Przestrzeń VMware datastore w bajtach lub jako wartość procentowa całkowitej przestrzeni.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.

Parametry:

- **url** - adres URL usługi VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *pfree* (procent wolnej przestrzeni), *uncommitted*.

vmware.datastore.tags.get[url,uuid]

 Tablica tagów magazynu danych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator magazynu danych VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.datastore.write[url,datastore,<mode>]

 Czas trwania operacji zapisu do datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - *latency* (wartość średnia, domyślnie) lub *maxlatency* (wartość maksymalna).

vmware.dc.alarms.get[url,id]

 Dane alarmów centrum danych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator centrum danych VMware.

vmware.dc.discoveryurl

 Wykrywanie centrów danych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware.

vmware.dc.tags.get[url,id]

 Tablica tagów centrum danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator centrum danych VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.dvswitch.discoveryurl

 Wykrywanie przełączników rozproszonych VMware vSphere.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.dvswitch.fetchports.get[url,uuid,<filter>,<mode>]

 Dane portów VMware vSphere Distributed Switch.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator DVSwitch VMware.
- **filter** - pojedynczy ciąg znaków z kryteriami rozdzielonymi przecinkami do wybierania portów;
- **mode** - *state* (wszystkie XML bez węzłów XML ”config”, domyślnie) lub *full*.

Parametr **filter** obsługuje kryteria dostępne w obiekcie danych VMware DistributedVirtualSwitchPortCriteria.

Przykład:

```
vmware.dvswitch.fetchports.get[{$VMWARE.URL},{$VMWARE.DVS.UUID},"connected:true,active:true,uplinkPort:fal
```

vmware.hv.alarms.get[url,uuid]

 Dane alarmów hipernadzorcy VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.cluster.name[url,uuid]

 Nazwa klastra hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.connectionstate[url,uuid]

 Stan połączenia z hiperwizorem VMware.
 Zwracana wartość: *String*: *connected*, *disconnected* lub *notResponding*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.cpu.usage[url,uuid]

 Użycie procesora hypervisora VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.cpu.usage.perf[url,uuid]

 Użycie procesora hypervisora VMware jako wartość procentowa w danym przedziale czasu.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.cpu.utilization[url,uuid]

 Użycie procesora hipernadzorcy VMware jako wartość procentowa w danym przedziale czasu; zależy od zarządzania energią lub HT.
 Zwracana wartość: *Float*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datacenter.name[url,uuid]

 Nazwa centrum danych hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datastore.discovery[url,uuid]

 Wykrywanie magazynów danych hipernadzorcy VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datastore.list[url,uuid]

 Lista magazynów danych hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

Przykład danych wyjściowych:

```
SSD-RAID1-VAULT1
SSD-RAID1-VAULT2
SSD-RAID10
```

vmware.hv.datastore.multipath[url,uuid,<datastore>,<partitionid>]

 Liczba dostępnych ścieżek datastore.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware;
- **datastore** - UUID lub nazwa datastore;
- **partitionid** - wewnętrzny identyfikator urządzenia fizycznego z `vmware.hv.datastore.discovery`.

vmware.hv.datastore.read[url,uuid,datastore,<mode>]

 Średni czas trwania operacji odczytu z datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (domyślnie).

vmware.hv.datastore.size[url,uuid,datastore,<mode>]

 Miejsce w VMware datastore w bajtach lub jako wartość procentowa całkowitej pojemności.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *pfree* (procent wolnego miejsca), *uncommitted*.

vmware.hv.datastore.write[url,uuid,datastore,<mode>]

 Średni czas trwania operacji zapisu do datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (domyślnie).

vmware.hv.discoveryurl

 Wykrywanie hiperwizorów VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware.

vmware.hv.diskinfo.get[url,uuid]

 Dane dysku hipernadzorcy VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.fullname[url,uuid]

 Pełna nazwa produktu, włącznie z informacją o wersji.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.cpu.freq[url,uuid]

 Częstotliwość procesora hiperwizora VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.model[url,uuid]

 Model procesora hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.num[url,uuid]

 Liczba rdzeni procesora w hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.threads[url,uuid]

 Liczba wątków procesora w hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.memory[url,uuid]

 Całkowity rozmiar pamięci hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.model[url,uuid]

 Model hypervisora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hypervisora VMware.

vmware.hv.hw.sensors.get[url,uuid]

 Wartość czujników sprzętowych hipernadzorcy VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.serialnumber[url,uuid]

 Numer seryjny hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.hw.uuid[url,uuid]

 UUID BIOS hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.hw.vendor[url,uuid]

 Nazwa dostawcy hypervisora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hypervisora VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.maintenance[url,uuid]

 Status trybu konserwacji hipernadzorcy VMware.
 Zwracana wartość: *0* - nie jest w trybie konserwacji; *1* - jest w trybie konserwacji.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.memory.size.ballooned[url,uuid]

 Rozmiar pamięci ballooned hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.memory.used[url,uuid]

 Rozmiar używanej pamięci hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.net.if.discovery[url,uuid]

 Wykrywanie interfejsów sieciowych hypervisora VMware.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.network.in[url,uuid,<mode>]

 Statystyki ruchu przychodzącego sieci hipernadzorcy VMware (bajty na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **mode** — *bps* (domyślnie), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.network.linkspeed[url,uuid,ifname]

 Prędkość interfejsu sieciowego hipernadzorcy VMware.
 Wartość zwracana: *Integer*. Zwraca *0*, jeśli interfejs sieciowy jest wyłączony, w przeciwnym razie zwraca wartość prędkości interfejsu.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;

- **ifname** — nazwa interfejsu.

vmware.hv.network.out[url,uuid,<mode>]

 Statystyki wyjścia sieciowego hypervisora VMware (bajty na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware;
- **mode** - *bps* (domyślnie), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności hipernadzorcy VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware;
- **path** - ścieżka licznika wydajności [1];
- **instance** - instancja licznika wydajności. Użyj pustej instancji dla wartości zagregowanych (domyślnie).

vmware.hv.property[url,uuid,prop]

 Właściwość hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware;
- **prop** — ścieżka właściwości.

vmware.hv.power[url,uuid,<max>]

 Zużycie energii hiperwizora VMware (W).
 Wartość zwracana: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware;
- **max** - maksymalne dozwolone zużycie energii.

vmware.hv.sensor.health.state[url,uuid]

 Sensor zbiorczego stanu kondycji hiperwizora VMware.
 Zwracana wartość: *Integer*: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

Należy pamiętać, że ta pozycja może nie działać w VMware vSphere 6.5 i nowszych wersjach, ponieważ VMware wycofał sensor *VMware Rollup Health State*.

vmware.hv.sensors.get[url,uuid]

 Czujniki stanu producenta sprzętu hypervisora VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.status[url,uuid]

 Status hipernadzorcy VMware.
 Zwracana wartość: *Integer*: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja używa właściwości ogólnego stanu systemu hosta.

vmware.hv.tags.get[url,uuid]

 Tablica tagów hipernadzorcy VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.hv.uptime[url,uuid]

 Czas działania hipernadzorcy VMware (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja używa właściwości ogólnego stanu systemu hosta.

vmware.hv.version[url,uuid]

 Wersja hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.vm.num[url,uuid]

 Liczba maszyn wirtualnych na hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

vmware.rp.cpu.usage[url,rpid]

 Użycie CPU w hercach w danym przedziale czasu dla puli zasobów VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **rpid** — identyfikator puli zasobów VMware.

vmware.rp.memory[url,rpid,<mode>]

 Metryki pamięci puli zasobów VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **rpid** - identyfikator puli zasobów VMware;
- **mode** - możliwe wartości:
consumed (domyślnie) - ilość fizycznej pamięci hosta zużywanej do obsługi stron fizycznej pamięci gościa
 ballooned - ilość fizycznej pamięci gościa odzyskanej z maszyny wirtualnej przez sterownik balloon w systemie gościa
overhead - fizyczna pamięć hosta zużywana przez struktury danych ESXi do uruchamiania maszyn wirtualnych

vmware.alarms.geturl

 Dane alarmów VMware virtual center.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.vm.alarms.get[url,uuid]

 Dane alarmów maszyny wirtualnej VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.attribute[url,uuid,name]

 Wartość atrybutu niestandardowego maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **name** - nazwa atrybutu niestandardowego.

vmware.vm.cluster.name[url,uuid]

 Nazwa maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.consolidationneeded[url,uuid]

 Dysk maszyny wirtualnej VMware wymaga konsolidacji.
 Zwracana wartość: *String*: *true* - konsolidacja jest wymagana; *false* - konsolidacja nie jest wymagana.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.latency[url,uuid]

 Procent czasu, w którym maszyna wirtualna nie może działać, ponieważ konkuruje o dostęp do fizycznych procesorów CPU.
 Wartość zwracana: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.num[url,uuid]

 Liczba procesorów w maszynie wirtualnej VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.readiness[url,uuid,<instance>]

 Procent czasu, w którym maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU.
 Zwracana wartość: *Float*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja CPU.

vmware.vm.cpu.ready[url,uuid]

 Czas (w milisekundach), przez który maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU. Czas gotowości CPU zależy od liczby maszyn wirtualnych na hoście i ich obciążeń CPU (%).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.swapwait[url,uuid,<instance>]

 Procent czasu CPU spędzonego na oczekiwaniu na swap-in.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja CPU.

vmware.vm.cpu.usage[url,uuid]

 Użycie procesora maszyny wirtualnej VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.usage.perf[url,uuid]

 Użycie procesora maszyny wirtualnej VMware jako wartość procentowa w danym przedziale czasu.
 Zwracana wartość: *Float*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.datacenter.name[url,uuid]

 Nazwa centrum danych maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.discoveryurl

 Wykrywanie maszyn wirtualnych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.vm.guest.memory.size.swapped[url,uuid]

 Ilość fizycznej pamięci gościa przeniesionej do przestrzeni wymiany (swap) (KB).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.guest.osuptime[url,uuid]

 Całkowity czas, jaki upłynął od ostatniego uruchomienia systemu operacyjnego (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.hv.maintenance[url,uuid]

 Status konserwacji hypervisora maszyny wirtualnej VMware.
 Zwracana wartość: *0* - nie jest w trybie konserwacji; *1* - jest w trybie konserwacji.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.hv.name[url,uuid]

 Nazwa hypervisora maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size[url,uuid]

 Całkowity rozmiar pamięci maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.ballooned[url,uuid]

 Rozmiar pamięci ballooned maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.compressed[url,uuid]

 Rozmiar skompresowanej pamięci maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.consumed[url,uuid]

 Ilość fizycznej pamięci hosta zużywanej do obsługi stron fizycznej pamięci gościa (KB).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.private[url,uuid]

 Rozmiar pamięci prywatnej maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.shared[url,uuid]

 Rozmiar pamięci współdzielonej maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Liczba całkowita*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.swapped[url,uuid]

 Rozmiar pamięci wymienionej do przestrzeni wymiany maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.usage.guest[url,uuid]

 Użycie pamięci gościa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.usage.host[url,uuid]

 Użycie pamięci hosta maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.usage[url,uuid]

 Procent pamięci fizycznej hosta, która została zużyta.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;

- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.net.if.discovery[url,uuid]

 Wykrywanie interfejsów sieciowych maszyn wirtualnych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.net.if.in[url,uuid,instance,<mode>]

 Statystyki wejściowe interfejsu sieciowego maszyny wirtualnej VMware (bajty/pakiety na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja interfejsu sieciowego;
- **mode** - *bps* (domyślnie) lub *pps* - bajty lub pakiety na sekundę.

vmware.vm.net.if.out[url,uuid,instance,<mode>]

 Statystyki ruchu wychodzącego interfejsu sieciowego maszyny wirtualnej VMware (bajty/pakiety na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja interfejsu sieciowego;
- **mode** — *bps* (domyślnie) lub *pps* — bajty lub pakiety na sekundę.

vmware.vm.net.if.usage[url,uuid,<instance>]

 Wykorzystanie sieci maszyny wirtualnej VMware (łączne szybkości wysyłania i odbierania) w danym przedziale czasu (KBps).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja interfejsu sieciowego.

vmware.vm.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności maszyny wirtualnej VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **path** - ścieżka licznika wydajności [1];
- **instance** - instancja licznika wydajności. Dla wartości zagregowanych (domyślnie) użyj pustej instancji.

vmware.vm.powerstate[url,uuid]

 Stan zasilania maszyny wirtualnej VMware.
 Zwracana wartość: *0* - poweredOff; *1* - poweredOn; *2* - suspended.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.property[url,uuid,prop]

 Właściwość maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **prop** - ścieżka właściwości.

vmware.vm.snapshot.get[url,uuid]

 Stan migawki maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.state[url,uuid]

 Stan maszyny wirtualnej VMware.
 Zwracana wartość: *String*: *notRunning*, *resetting*, *running*, *shuttingDown*, *standby* lub *unknown*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.committed[url,uuid]

 Zajęta przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.readoio[url,uuid,instance]

 Średnia liczba oczekujących żądań odczytu do dysku wirtualnego w czasie interwału zbierania danych.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja urządzenia dyskowego.

vmware.vm.storage.totalreadlatency[url,uuid,instance]

 Średni czas trwania odczytu z dysku wirtualnego (w milisekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.storage.totalwritelatency[url,uuid,instance]

 Średni czas trwania zapisu na dysk wirtualny (w milisekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.storage.uncommitted[url,uuid]

 Niezatwierdzona przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.unshared[url,uuid]

 Niewspółdzielona przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.writeoio[url,uuid,instance]

 Średnia liczba oczekujących żądań zapisu do dysku wirtualnego w czasie interwału zbierania danych.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.tags.get[url,uuid]

 Tablica tagów maszyny wirtualnej VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.vm.tools[url,uuid,<mode>]

 Stan narzędzi gościa maszyny wirtualnej VMware.
 Zwracana wartość: *String*.
 W trybie *status*: *guestToolsExecutingScripts* - VMware Tools uruchamia się; *guestToolsNotRunning* - VMware Tools nie jest uruchomione; *guestToolsRunning* - VMware Tools jest uruchomione.
 W trybie *version*: wersja.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **mode** - *version*, *status* (domyślnie).

vmware.vm.uptime[url,uuid]

 Czas działania maszyny wirtualnej VMware (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.vfs.dev.discovery[url,uuid]

 Wykrywanie urządzeń dyskowych maszyny wirtualnej VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.vfs.dev.read[url,uuid,instance,<mode>]

 Statystyki odczytu urządzenia dyskowego maszyny wirtualnej VMware (bajty/operacje na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego;
- **mode** — *bps* (domyślnie) lub *ops* — bajty lub operacje na sekundę.

vmware.vm.vfs.dev.write[url,uuid,instance,<mode>]

 Statystyki zapisu urządzenia dyskowego maszyny wirtualnej VMware (bajty/operacje na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego;
- **mode** — *bps* (domyślnie) lub *ops* — bajty lub operacje na sekundę.

vmware.vm.vfs.fs.discovery[url,uuid]

 Wykrywanie systemów plików maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

Aby ta pozycja działała, w gościnnej maszynie wirtualnej muszą być zainstalowane VMware Tools.

vmware.vm.vfs.fs.size[url,uuid,fsname,<mode>]

 Statystyki systemu plików maszyny wirtualnej VMware (bajty/procenty).
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowych.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **fsname** - nazwa systemu plików;
- **mode** - *total*, *free*, *used*, *pfree* lub *pused*.

Aby ta pozycja działała, na gościnnej maszynie wirtualnej muszą być zainstalowane VMware Tools.

Przypisy

[1] Zobacz Tworzenie niestandardowych nazw liczników wydajności dla VMware.

[2] Wartość tych pozycji jest pobierana z liczników wydajności VMware, a parametr VMwarePerfFrequency parameter jest używany do odświeżania ich danych w pamięci podręcznej VMware Zabbix:

- vmware.cl.perfcounter
- 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

Więcej informacji

Szczegółowe informacje o tym, jak skonfigurować Zabbix do monitorowania środowisk VMware, można znaleźć w sekcji Monitorowanie maszyn wirtualnych.

2 Kluczowe pola wykrywania maszyn wirtualnych

Poniższa tabela zawiera pola zwracane przez klucze wykrywania związane z maszynami wirtualnymi.

Klucz pozycji		
Opis	**Pole**	**Pobrana zawartość**
vmware.cluster.discovery		
Wykonuje wykrywanie klastrów.	{#CLUSTER.ID}	Identyfikator klastra.
	{#CLUSTER.NAME}	Nazwa klastra.

Klucz pozycji		

	”resource_pool”	Tablica zawierająca dane puli zasobów, w tym identyfikator grupy zasobów, tablicę tagów, ścieżkę puli zasobów, liczbę maszyn wirtualnych.
		
		Struktura tablicy:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`
		
		Dla struktury tablicy "tags" zobacz pole ”tags”.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.datastore.discovery		
Wykonuje wykrywanie magazynów danych.	{#DATASTORE}	Nazwa magazynu danych.

	{#DATASTORE.EXTENT}	Tablica zawierająca identyfikator partycji extent magazynu danych oraz nazwę instancji.
		
		Struktura tablicy:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	{#DATASTORE.TYPE}	Typ magazynu danych. Przykładowe wartości: VMFS, NFS, vsan itp.
	{#DATASTORE.UUID}	Identyfikator magazynu danych.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dc.discovery		
Wykonuje wykrywanie centrów danych.	{#DATACENTER}	Nazwa centrum danych.
	{#DATACENTERID}	Identyfikator centrum danych.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dvswitch.discovery		
Wykonuje wykrywanie przełączników rozproszonych vSphere.	{#DVS.NAME}	Nazwa przełącznika.

Klucz pozycji		
	{#DVS.UUID}	Identyfikator przełącznika.
vmware.hv.discovery		
Wykonuje wykrywanie hypervisorów.	{#HV.UUID}	Unikalny identyfikator hypervisora.
	{#HV.ID}	Identyfikator hypervisora (nazwa obiektu zarządzanego HostSystem).
	{#HV.NAME}	Nazwa hypervisora.
	{#HV.NETNAME}	Nazwa sieciowa hosta hypervisora.
	{#HV.IP}	Adres IP hypervisora, może być pusty. W przypadku konfiguracji HA z wieloma interfejsami sieciowymi obowiązuje następujący priorytet wyboru interfejsu: - preferowany jest adres IP współdzielący podsieć IP z adresem IP vCenter; - preferowany jest adres IP z podsieci IP z bramą domyślną; - preferowany jest adres IP z interfejsu o najniższym ID.
	{#CLUSTER.NAME}	Nazwa klastra, może być pusta.
	{#DATACENTER.NAME}	Nazwa centrum danych.
	{#PARENT.NAME}	Nazwa kontenera przechowującego hypervisor.
	{#PARENT.TYPE}	Typ kontenera, w którym przechowywany jest hypervisor. Wartości mogą być następujące: `Datacenter`, `Folder`, `ClusterComputeResource`, `VMware`, gdzie ”VMware” oznacza nieznany typ kontenera.

	”resource_pool”	Tablica zawierająca dane puli zasobów, w tym identyfikator grupy zasobów, tablicę tagów, ścieżkę puli zasobów, liczbę maszyn wirtualnych.
		
		Struktura tablicy:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`
		
		Dla struktury tablicy `"tags"` zobacz pole ”tags”.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.datastore.discovery		
Wykonuje wykrywanie magazynów danych hypervisora. Należy pamiętać, że wiele hypervisorów może używać tego samego magazynu danych.	{#DATASTORE}	Nazwa magazynu danych.
	{#DATASTORE.TYPE}	Typ magazynu danych. Przykładowe wartości: `VMFS`, `NFS`, `vsan` itp.
	{#DATASTORE.UUID}	Identyfikator magazynu danych.
	{#MULTIPATH.COUNT}	Zarejestrowana liczba ścieżek magazynu danych.
	{#MULTIPATH.PARTITION.COUNT}	Liczba dostępnych partycji dysku.

Klucz pozycji		

	”datastore_extent”	Tablica zawierająca nazwę instancji extent magazynu danych oraz identyfikator partycji.
		
		Struktura tablicy:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.net.if.discovery		
Wykonuje wykrywanie interfejsów sieciowych hypervisora.	{#IFNAME}	Nazwa interfejsu.
	{#IFDRIVER}	Sterownik interfejsu.
	{#IFDUPLEX}	Ustawienia dupleksu interfejsu.
	{#IFSPEED}	Prędkość interfejsu.
	{#IFMAC}	Adres MAC interfejsu.
vmware.vm.discovery		
Wykonuje wykrywanie maszyn wirtualnych.	{#VM.UUID}	Unikalny identyfikator maszyny wirtualnej.
	{#VM.ID}	Identyfikator maszyny wirtualnej (nazwa obiektu zarządzanego VirtualMachine).
	{#VM.NAME}	Nazwa maszyny wirtualnej.
	{#HV.NAME}	Nazwa hypervisora.
	{#HV.UUID}	Unikalny identyfikator hypervisora.
	{#HV.ID}	Identyfikator hypervisora (nazwa obiektu zarządzanego HostSystem).
	{#CLUSTER.NAME}	Nazwa klastra, może być pusta.
	{#DATACENTER.NAME}	Nazwa centrum danych.
	{#DATASTORE.NAME}	Nazwa magazynu danych.
	{#DATASTORE.UUID}	Identyfikator magazynu danych.
	{#VM.IP}	Adres IP maszyny wirtualnej, może być pusty.
	{#VM.DNS}	Nazwa DNS maszyny wirtualnej, może być pusta.
	{#VM.GUESTFAMILY}	Rodzina systemu operacyjnego gościa maszyny wirtualnej, może być pusta.
	{#VM.GUESTFULLNAME}	Pełna nazwa systemu operacyjnego gościa maszyny wirtualnej, może być pusta.
	{#VM.FOLDER}	Łańcuch folderów nadrzędnych maszyny wirtualnej; może być używany jako wartość dla grup zagnieżdżonych; nazwy folderów są łączone znakiem ”/”. Może być pusty.
	{#VM.TOOLS.STATUS}	Stan narzędzi maszyny wirtualnej VMware.
	{#VM.POWERSTATE}	Stan zasilania maszyny wirtualnej VMware (`poweredOff`, `poweredOn` lub `suspended`).
	{#VM.RPOOL.ID}	Identyfikator puli zasobów.
	{#VM.RPOOL.PATH}	Pełna ścieżka puli zasobów z wyłączeniem nazwy głównej ”Resources”. Nazwy folderów są łączone znakiem ”/”.
	{#VM.SNAPSHOT.COUNT}	Liczba migawek VM.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

Klucz pozycji		

	”vm_customattribute”	Tablica niestandardowych atrybutów maszyny wirtualnej (jeśli są zdefiniowane).
		
		Struktura tablicy:
		`[{`
		`"name":"custom field name",`
		`"value":"custom field value"`
		`}]`

	”net_if”	Tablica interfejsów sieciowych maszyny wirtualnej.
		
		Struktura tablicy:
		`[{`
		`"ifname": "interface name",`
		`"ifdesc": "interface description",`
		`"ifmac": "00:00:00:00:00:00",`
		`"ifconnected": true,`
		`"iftype": "interface type",`
		`"ifbackingdevice": "interface backing device",`
		`"ifdvswitch_uuid": "interface switch uuid",`
		`"ifdvswitch_portgroup": "interface switch port group",`
		`"ifdvswitch_port": "interface switch port",`
		`"ifip": ["interface ip addresses"]`
		`}]`
		
		Opis zwracanych danych znajduje się w kluczu pozycji ”vmware.vm.net.if.discovery”.

vmware.vm.net.if.discovery		
Wykonuje wykrywanie interfejsów sieciowych maszyny wirtualnej.	{#IFNAME}	Nazwa interfejsu sieciowego.
	{#IFDESC}	Opis interfejsu.
	{#IFMAC}	Adres MAC interfejsu.
	{#IFCONNECTED}	Stan połączenia interfejsu (`false` - rozłączony; `true` - połączony).
	{#IFTYPE}	Typ interfejsu.
	{#IFBACKINGDEVICE}	Nazwa urządzenia bazowego.
	{#IFDVSWITCH.UUID}	Unikalny identyfikator vSphere Distributed Switch.
	{#IFDVSWITCH.PORTGROUP}	Rozproszona grupa portów.
	{#IFDVSWITCH.PORT}	Port vSphere Distributed Switch.
	”ifip”	Tablica adresów interfejsu.
vmware.vm.vfs.dev.discovery		
Wykonuje wykrywanie urządzeń dyskowych maszyny wirtualnej.	{#DISKNAME}	Nazwa urządzenia dyskowego.
vmware.vm.vfs.fs.discovery		
Wykonuje wykrywanie systemów plików maszyny wirtualnej.	{#FSNAME}	Nazwa systemu plików.

3 przykłady JSON dla pozycji VMware

Przegląd Ta sekcja zawiera dodatkowe informacje o obiektach JSON zwracanych przez różne pozycje.

vmware.*.alarms.get Pozycje **vmware.alarms.get[]**, **vmware.cluster.alarms.get[]**, **vmware.datastore.alarms.get[]**, **vmware.dc.alarms.get[]**, **vmware.hv.alarms.get[]**, **vmware.vm.alarms.get[]** zwracają obiekty JSON o następującej strukturze (wartości podano jako przykład; obiekt `entity` jest zwracany od Zabbix 7.4.10):

```
{
    "alarms": [
        {
            "name": "Host connection and power state",
            "system_name": "alarm.HostConnectionStateAlarm",
            "description": "Default alarm to monitor host connection and power state",
            "enabled": true,
            "key": "alarm-1.host-2013",
            "time": "2022-06-27T05:27:38.759976Z",
            "overall_status": "red",
            "acknowledged": false,
            "entity": {
                "id": "host-2013",
                "uuid": "4201b707-f737-7cf8-881f-49215d1aad2a",
                "type": "HostSystem"
            }
        },
        {
            "name": "Host memory usage",
            "system_name": "alarm.HostMemoryUsageAlarm",
            "description": "Default alarm to monitor host memory usage",
            "enabled": true,
            "key": "alarm-4.host-1004",
            "time": "2022-05-16T13:32:42.47863Z",
            "overall_status": "yellow",
            "acknowledged": false,
            "entity": {
                "id": "host-1004",
                "uuid": "5315c818-g848-8dg9-992g-50326e2bbe3a",
                "type": "HostSystem"
            }
        },
        {
            // other alarms
        }
    ]
}
```

vmware.*.tags.get Pozycje **vmware.cluster.tags.get[]**, **vmware.datastore.tags.get[]**, **vmware.dc.tags.get[]**, **vmware.hv.tags.g**
vmware.vm.tags.get[] zwracają obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "tags": [
        {
            "name": "Windows",
        "description": "tag dla kategorii typ systemu operacyjnego",
        "category": "typ systemu operacyjnego"
    },
    {
            "name": "SQL Server",
        "description": "tag dla kategorii nazwa aplikacji",
        "category": "nazwa aplikacji"
        },
    {
            // inne tagi
        }
    ]
}
```

vmware.hv.diskinfo.get Pozycja **vmware.hv.diskinfo.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
[
  {
    "instance": "mpx.vmhba32:C0:T0:L0",
    "hv_uuid": "8002299e-d7b9-8728-d224-76004bbb6100",
    "datastore_uuid": "",
    "operational_state": [
      "ok"
    ],
    "lun_type": "disk",
    "queue_depth": 1,
    "model": "USB DISK",
    "vendor": "SMI Corp",
    "revision": "1100",
    "serial_number": "CCYYMMDDHHmmSS9S62CK",
    "vsan": {}
  },
  {
    // other instances
  }
]
```

vmware.dvswitch.fetchports.get Pozycja **vmware.dvswitch.fetchports.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "FetchDVPortsResponse":
    {
        "returnval": [
            {
                "key": "0",
                "dvsUuid": "50 36 6a 24 25 c0 10 9e-05 4a f6 ea 4e 3d 09 88",
                "portgroupKey": "dvportgroup-2023",
                "proxyHost":
                {
                    "@type": "HostSystem",
                    "#text": "host-2021"
                },
                "connectee":
                {
                    "connectedEntity":
                    {
                        "@type": "HostSystem",
                        "#text": "host-2021"
                    },
                    "nicKey": "vmnic0",
                    "type": "pnic"
                },
                "conflict": "false",
                "state":
                {
                    "runtimeInfo":
                    {
                        "linkUp": "true",
                        "blocked": "false",
                        "vlanIds":
                        {
                            "start": "0",
                            "end": "4094"
                        },
                        "trunkingMode": "true",
                        "linkPeer": "vmnic0",
                        "macAddress": "00:00:00:00:00:00",
                        "statusDetail": null,
```

```
                        "vmDirectPathGen2Active": "false",
                        "vmDirectPathGen2InactiveReasonOther": "portNptIncompatibleConnectee"
                    },
                    "stats":
                    {
                        "packetsInMulticast": "2385470",
                        "packetsOutMulticast": "45",
                        "bytesInMulticast": "309250248",
                        "bytesOutMulticast": "5890",
                        "packetsInUnicast": "155601537",
                        "packetsOutUnicast": "113008658",
                        "bytesInUnicast": "121609489384",
                        "bytesOutUnicast": "47240279759",
                        "packetsInBroadcast": "1040420",
                        "packetsOutBroadcast": "7051",
                        "bytesInBroadcast": "77339771",
                        "bytesOutBroadcast": "430392",
                        "packetsInDropped": "0",
                        "packetsOutDropped": "0",
                        "packetsInException": "0",
                        "packetsOutException": "0"
                    }
                },
                "connectionCookie": "1702765133",
                "lastStatusChange": "2022-03-25T14:01:11Z",
                "hostLocalPort": "false"
            },
            {
                //inne klucze
            }
        ]
    }
}
```

vmware.hv.hw.sensors.get Pozycja **vmware.hv.hw.sensors.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "val":
        {
            "@type": "HostHardwareStatusInfo",
            "storageStatusInfo": [
                {
                    "name": "Intel Corporation HD Graphics 630 #2",
                "status":
                        {
                            "label": "Unknown",
                    "summary": "Cannot report on the current status of the physical element",
                    "key": "Unknown"
                }
            },
            {
                    "name": "Intel Corporation 200 Series/Z370 Chipset Family USB 3.0 xHCI Controller #20"
                "status":
                        {
                     "label": "Unknown",
                     "summary": "Cannot report on the current status of the physical element",
                     "key": "Unknown"
                 }
            },
            {
                    // inne czujniki sprzętowe hv
                }
```

```
        ]
        }
}
```

vmware.hv.sensors.get Pozycja **vmware.hv.sensors.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "val":
        {
            "@type": "ArrayOfHostNumericSensorInfo", "HostNumericSensorInfo": [
                {
                    "@type": "HostNumericSensorInfo",
                    "name": "System Board 1 PwrMeter Output --- Normal",
                    "healthState":
                        {
                            "label": "Green",
                            "summary": "Sensor is operating under normal conditions",
                    "key": "green"
                },
                    "currentReading": "10500",
                "unitModifier": "-2",
                "baseUnits": "Watts",
                "sensorType": "other"
            },
                {
                    "@type": "HostNumericSensorInfo",
                    "name": "Power Supply 1 PS 1 Output --- Normal",
                    "healthState":
                        {
                            "label": "Green",
                    "summary": "Sensor is operating under normal conditions",
                    "key": "green"
                },
                    "currentReading": "10000",
            "unitModifier": "-2",
            "baseUnits": "Watts",
            "sensorType": "power"
        },
            {
                    // inne sensory hv
                }
        ]
        }
}
```

vmware.vm.snapshot.get Jeśli istnieją jakiekolwiek snapshoty, pozycja **vmware.snapshot.get[]** zwraca obiekt JSON o następującej strukturze (wartości podano jako przykład):

```
{
  "snapshot": [
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:16:39 AM",
      "description": "Descr 1",
      "createtime": "2022-04-01T06:16:51.761Z",
      "size": 5755795171,
      "uniquesize": 5755795171
    },
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:18:21 AM",
      "description": "Descr 2",
      "createtime": "2022-04-01T06:18:29.164999Z",
      "size": 118650595,
```

```
      "uniquesize": 118650595
    },
    {
      "name": "VM Snapshot 4%2f1%2f2022, 9:37:29 AM",
      "description": "Descr 3",
      "createtime": "2022-04-01T06:37:53.534999Z",
      "size": 62935016,
      "uniquesize": 62935016
    }
  ],
  "count": 3,
  "latestdate": "2022-04-01T06:37:53.534999Z",
  "latestage": 22729203,
  "oldestdate": "2022-04-01T06:16:51.761Z",
  "oldestage": 22730465,
  "size": 5937380782,
  "uniquesize": 5937380782
}
```


Jeśli nie istnieje żaden snapshot, pozycja **vmware.snapshot.get[]** zwraca obiekt JSON z pustymi wartościami:


```
{
  "snapshot": [],
  "count": 0,
  "latestdate": null,
  "latestage": 0,
  "oldestdate": null,
  "oldestage": 0,
  "size": 0,
  "uniquesize": 0
}
```


4 Przykład konfiguracji monitorowania VMware

Przegląd

Poniższy przykład opisuje, jak skonfigurować Zabbix do monitorowania maszyn wirtualnych VMware. Obejmuje to:

- utworzenie hosta reprezentującego środowisko VMware;
- utworzenie reguły wykrywania niskiego poziomu, która wykrywa maszyny wirtualne w środowisku VMware;
- utworzenie prototypu hosta, na podstawie którego Zabbix wygeneruje rzeczywiste hosty dla maszyn wirtualnych wykrytych przez regułę wykrywania niskiego poziomu.

Wymagania wstępne

> **Note:**
> Ten przykład nie obejmuje konfiguracji VMware. Zakłada się, że VMware jest już skonfigurowane.

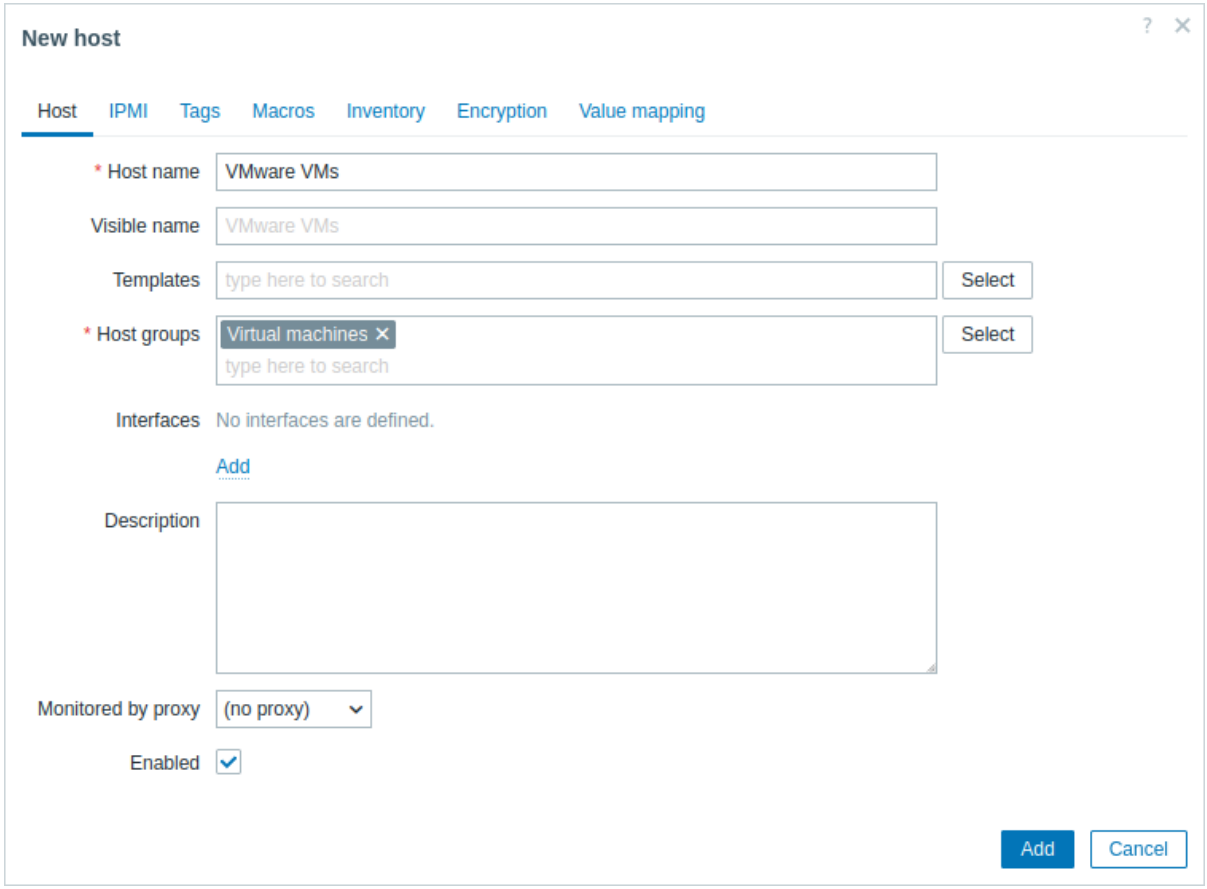
Przed kontynuowaniem ustaw parametr `StartVMwareCollectors` w pliku konfiguracyjnym serwera Zabbix na 2 lub więcej (wartość domyślna to 0).

Utwórz host

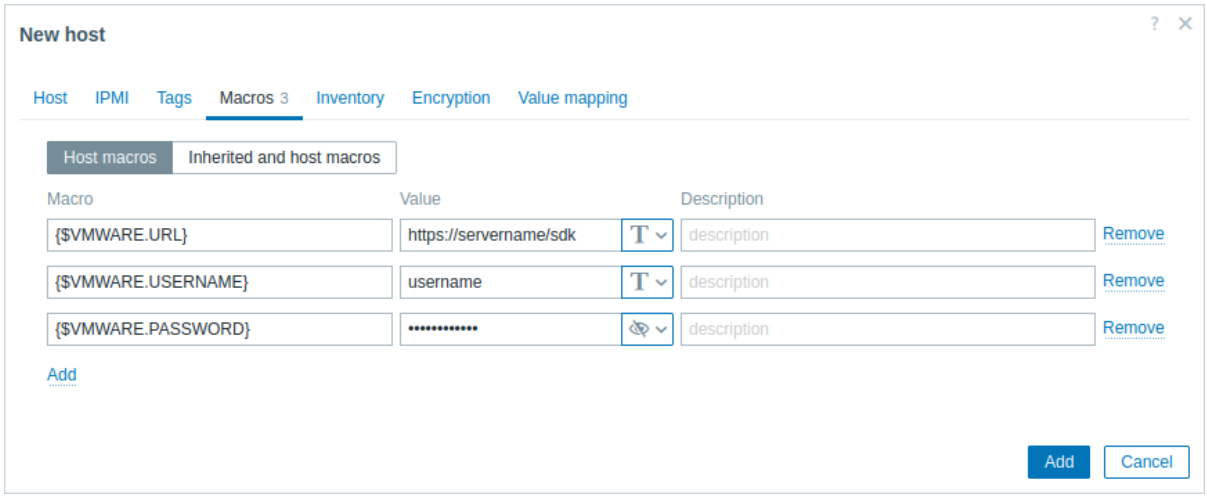
1. Przejdź do *Data collection* → *Hosts*.

2. Utwórz host:

- W polu *Host name* wprowadź nazwę hosta (na przykład „VMware VMs”).
- W polu *Host groups* wpisz lub wybierz grupę hostów (na przykład „Virtual machines”).



- Na karcie *Macros* ustaw następujące makra hosta:
  - {$VMWARE.URL} — adres URL SDK usługi VMware (hipernadzorcy ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} — nazwa użytkownika usługi VMware
  - {$VMWARE.PASSWORD} — hasło użytkownika {$VMWARE.USERNAME} usługi VMware



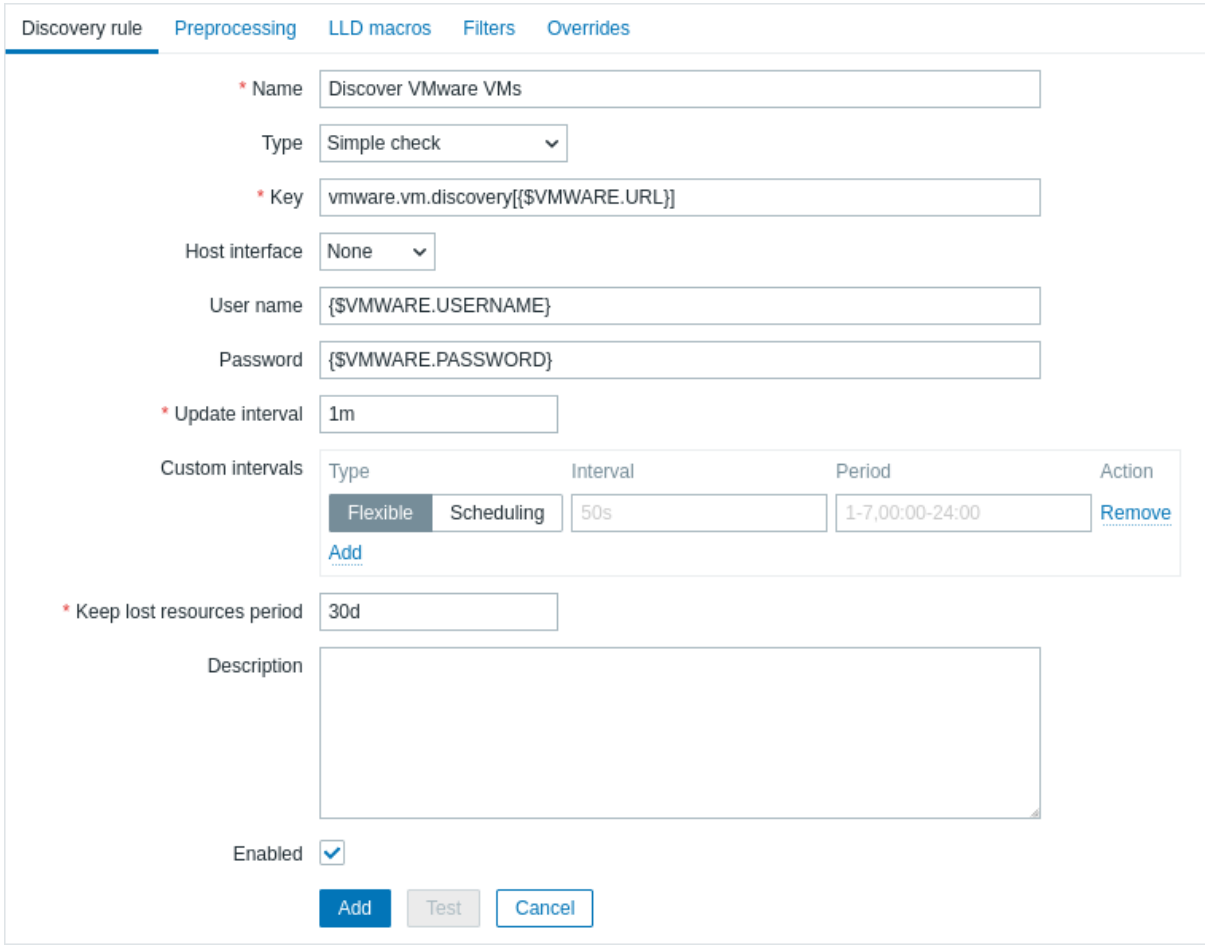
3. Kliknij przycisk *Add*, aby utworzyć host. Ten host będzie reprezentować Twoje środowisko VMware.

Utwórz regułę wykrywania niskiego poziomu

1. Kliknij *Discovery* dla utworzonego hosta, aby przejść do listy reguł wykrywania niskiego poziomu dla tego hosta.

2. Utwórz regułę wykrywania niskiego poziomu:

- W polu *Name* wprowadź nazwę reguły wykrywania niskiego poziomu (na przykład „Wykryj maszyny wirtualne VMware”).
- W polu *Type* wybierz „Simple check”.
- W polu *Key* wprowadź wbudowany klucz pozycji do wykrywania maszyn wirtualnych VMware: `vmware.vm.discovery[{$VMWARE.URL`
- W polach *User name* i *Password* wprowadź odpowiednie makra wcześniej skonfigurowane na hoście.



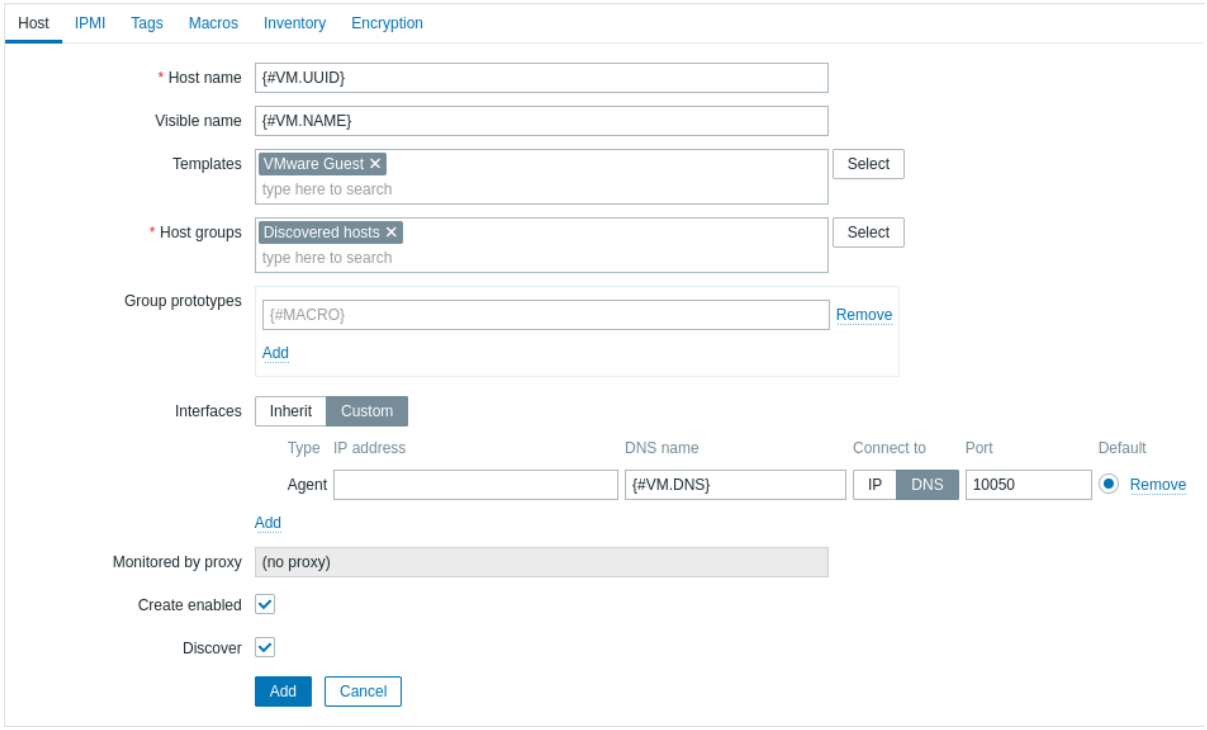
3. Kliknij przycisk *Add*, aby utworzyć regułę wykrywania niskiego poziomu. Ta reguła wykrywania wykryje maszyny wirtualne w Twoim środowisku VMware.

Utwórz prototyp hosta

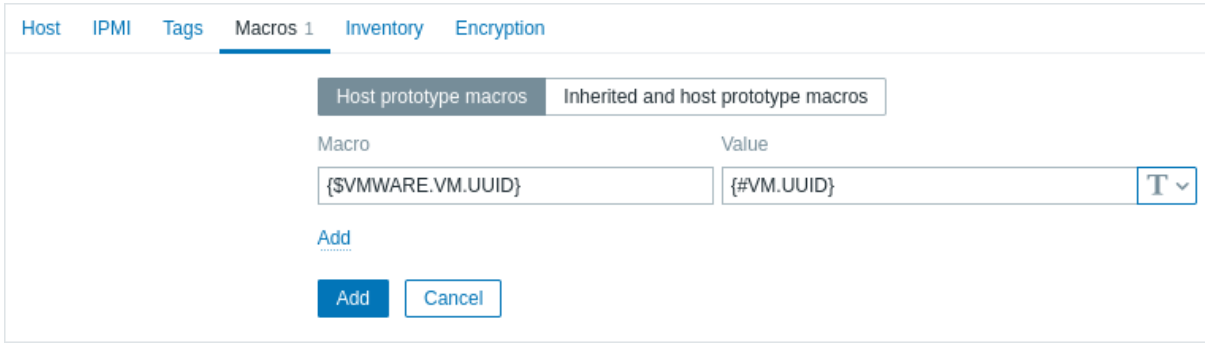
1. Na liście reguł wykrywania niskiego poziomu kliknij *Host prototypes* dla wcześniej utworzonej reguły wykrywania niskiego poziomu.

2. Utwórz prototyp hosta. Ponieważ prototypy hostów są wzorcami do tworzenia hostów za pomocą reguł wykrywania niskiego poziomu, większość pól będzie zawierać makra wykrywania niskiego poziomu. Zapewnia to tworzenie hostów z właściwościami opartymi na zawartości pobranej przez wcześniej utworzoną regułę wykrywania niskiego poziomu.

- W polu *Host name* wprowadź makro {#VM.UUID}.
- W polu *Visible name* wprowadź makro {#VM.NAME}.
- W polu *Templates* wprowadź lub wybierz szablon ”VMware Guest”. Ten szablon zawiera pozycje VMware oraz reguły wykrywania do monitorowania stanu zasilania maszyny wirtualnej, użycia CPU, użycia pamięci, urządzeń sieciowych itp.
- W polu *Host groups* wprowadź lub wybierz grupę hostów (na przykład ”Discovered hosts”).
- W polu *Interfaces* dodaj niestandardowy interfejs hosta. Następnie wprowadź makro {#VM.DNS} w polu *DNS name* lub wprowadź makro {#VM.IP} w polu *IP address*. Alternatywnie, jeśli maszyny wirtualne w Twoim środowisku VMware mają wiele interfejsów, przejdź do sekcji *Zaawansowana konfiguracja interfejsu hosta*. Skonfigurowanie niestandardowego interfejsu hosta jest konieczne do prawidłowego działania szablonu *VMware Guest*.



- Na karcie *Macros* ustaw makro {$VMWARE.VM.UUID} z wartością {#VM.UUID}. Jest to konieczne do prawidłowego działania szablonu *VMware Guest*, który używa tego makra jako makra użytkownika na poziomie hosta w parametrach pozycji (na przykład *vmware.vm.net.if.discovery[{$VMWARE.URL}, {$VMWARE.VM.UUID}]*).

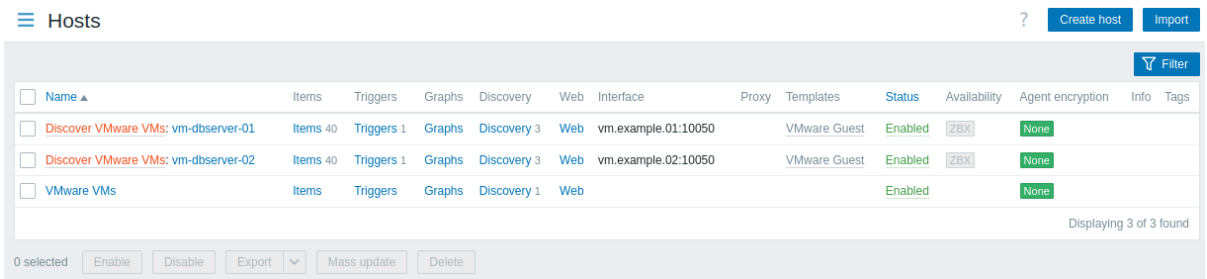


3. Kliknij przycisk *Add*, aby utworzyć prototyp hosta. Ten prototyp hosta będzie używany do tworzenia hostów dla maszyn wirtualnych wykrytych przez wcześniej utworzoną regułę wykrywania niskiego poziomu.

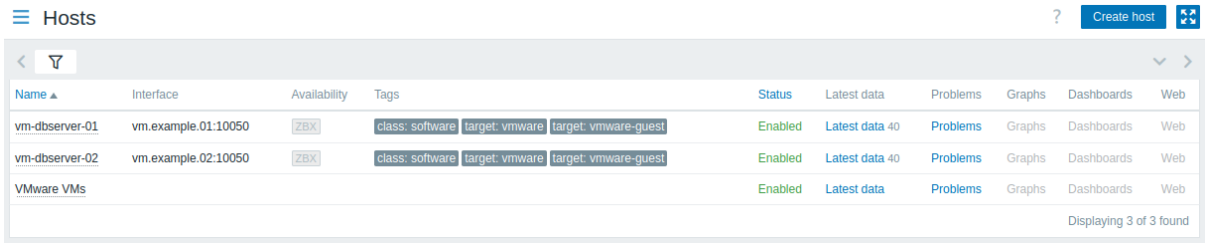
Wyświetlanie hostów i metryk

Po utworzeniu prototypu hosta reguła wykrywania niskiego poziomu utworzy hosty dla wykrytych maszyn wirtualnych VMware, a Zabbix rozpocznie ich monitorowanie. Należy pamiętać, że wykrywanie i tworzenie hostów można również wykonać ręcznie, jeśli to konieczne.

Aby wyświetlić utworzone hosty, przejdź do sekcji menu *Zbieranie danych → Hosty*.



Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitorowanie → Hosty* i kliknij *Najnowsze dane* dla jednego z hostów.



Zaawansowana konfiguracja interfejsu hosta

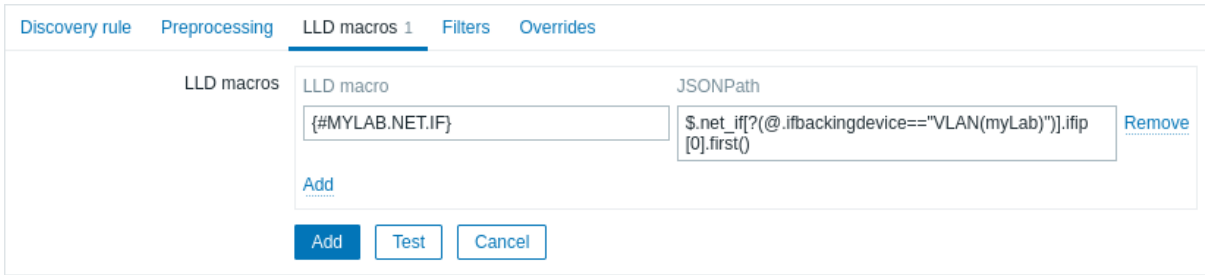
Klucz pozycji `vmware.vm.discovery[{$VMWARE.URL}]`, skonfigurowany w sekcji *Utwórz regułę wykrywania niskiego poziomu*, zwraca dane interfejsów sieciowych w polu "net_if":

```
"net_if": [
      {
        "ifname": "5000",
        "ifdesc": "Network adapter 1",
        "ifmac": "00:11:22:33:44:55",
        "ifconnected": true,
        "iftype": "VirtualVmxnet3",
        "ifbackingdevice": "VLAN(myLab)",
        "ifdvswitch_uuid": "",
        "ifdvswitch_portgroup": "",
        "ifdvswitch_port": "",
        "ifip": [
          "127.0.0.1",
          "::1"
        ]
      },
      {
        "ifname": "5001",
        "ifdesc": "Network adapter 2",
        "ifmac": "00:11:22:33:44:55",
        "ifconnected": false,
        "iftype": "VirtualVmxnet3",
        "ifbackingdevice": "VLAN(myLab2)",
        "ifdvswitch_uuid": "",
        "ifdvswitch_portgroup": "",
        "ifdvswitch_port": "",
        "ifip": []
      }
    ]
```

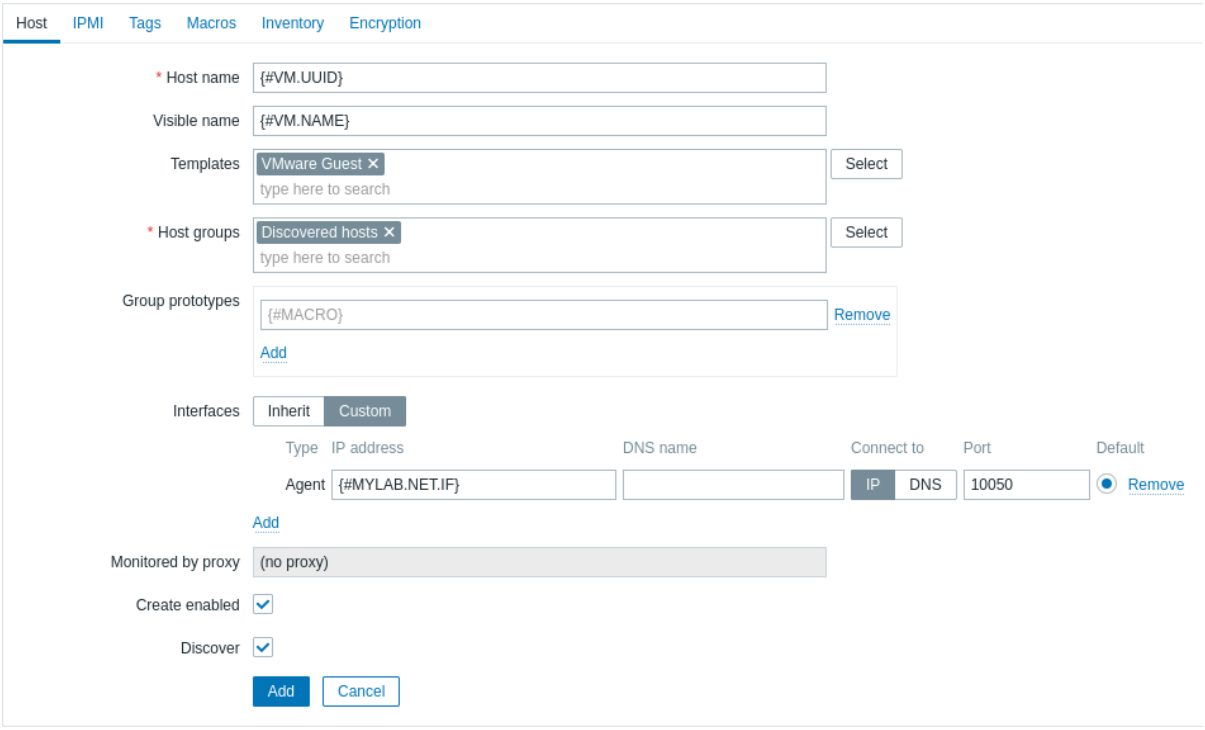
Dane te można wykorzystać do skonfigurowania niestandardowego interfejsu hosta.

1. Podczas tworzenia reguły wykrywania niskiego poziomu dodatkowo skonfiguruj makro wykrywania niskiego poziomu. Na karcie *Makra LLD* utwórz niestandardowe makro LLD z wartością JSONPath. Na przykład:

- `{#MYLAB.NET.IF}` - `$.net_if[?(@.ifbackingdevice=="VLAN(myLab)")].ifip[0].first()`



2. Podczas tworzenia prototypu hosta dodaj niestandardowy interfejs hosta i wprowadź makro LLD w polu *Nazwa DNS* lub *Adres IP*.



11 Konserwacja

Przegląd Konserwacja służy do tłumienia problemów w zdefiniowanych wcześniej przedziałach czasu.

W Zabbix można definiować okresy konserwacji dla hostów i grup hostów.

Ponadto możliwe jest zdefiniowanie konserwacji tylko dla pojedynczego wyzwalacza (lub podzbioru wyzwalaczy) przez określenie tagów wyzwalacza. W takim przypadku konserwacja zostanie aktywowana tylko dla tych wyzwalaczy; wszystkie pozostałe wyzwalacze hosta lub grupy hostów nie będą objęte konserwacją.

Istnieją dwa typy konserwacji: *Z gromadzeniem danych* oraz *Bez gromadzenia danych*.

Podczas konserwacji *Z gromadzeniem danych* wyzwalacze są przetwarzane jak zwykle, a zdarzenia są tworzone, gdy jest to wymagane. Jednak jeśli w konfiguracji akcji zaznaczona jest opcja *Pause operations for suppressed problems*, eskalacje problemów są wstrzymywane dla hostów/wyzwalaczy objętych konserwacją. W takim przypadku kroki eskalacji, które mogą obejmować wysyłanie powiadomień lub zdalnych poleceń, będą ignorowane tak długo, jak trwa okres konserwacji. Należy pamiętać, że operacje przywrócenia i aktualizacji problemu nie są tłumione podczas konserwacji, a jedynie eskalacje. Powiadomienia o przywróceniu nie są wysyłane, jeśli problem rozpoczął się podczas konserwacji.

Na przykład, jeśli kroki eskalacji są zaplanowane po 0, 30 i 60 minutach od wystąpienia problemu, a półgodzinna konserwacja trwa od 10. do 40. minuty po wystąpieniu rzeczywistego problemu, kroki drugi i trzeci zostaną wykonane pół godziny później, czyli po 60 i 90 minutach (o ile problem nadal istnieje). Podobnie, jeśli problem wystąpi podczas konserwacji, eskalacja rozpocznie się po zakończeniu konserwacji.

Aby otrzymywać powiadomienia o problemach podczas konserwacji w normalny sposób (bez opóźnienia), należy odznaczyć opcję *Pause operations for suppressed problems* w konfiguracji akcji.

> **Note:**
> Jeśli co najmniej jeden host (użyty w wyrażeniu wyzwalacza) nie jest w trybie konserwacji, Zabbix wyśle powiadomienie o problemie.

serwer Zabbix musi działać podczas konserwacji. Konserwacje są przeliczane co minutę lub natychmiast po przeładowaniu pamięci podręcznej konfiguracji, jeśli wprowadzono zmiany w okresie konserwacji.

Procesy timera sprawdzają, czy status hosta powinien zostać zmieniony na/z trybu konserwacji przy 0 sekundzie każdej minuty. Dodatkowo co sekundę proces timera sprawdza, czy jakiekolwiek konserwacje powinny zostać uruchomione/zatrzymane na podstawie zmian w [okresach konserwacji] po aktualizacji konfiguracji. Dlatego szybkość uruchamiania/zatrzymywania okresów konserwacji zależy od interwału aktualizacji konfiguracji (domyślnie 10 sekund). Należy pamiętać, że zmiany okresu konserwacji nie obejmują

ustawień *Active since/Active till*. Ponadto, jeśli host/grupa hostów zostanie dodana do istniejącego aktywnego okresu konserwacji, zmiany zostaną aktywowane przez proces timera dopiero na początku następnej minuty.

Należy pamiętać, że gdy host przechodzi w tryb konserwacji, procesy timera serwera Zabbix odczytają wszystkie otwarte problemy, aby sprawdzić, czy wymagają one tłumienia. Może to mieć wpływ na wydajność, jeśli istnieje wiele otwartych problemów. serwer Zabbix odczyta również wszystkie otwarte problemy podczas uruchamiania, nawet jeśli w danym momencie nie skonfigurowano żadnych konserwacji.

Należy pamiętać, że serwer Zabbix (lub proxy) zawsze zbiera dane niezależnie od typu konserwacji (w tym konserwacji *Bez gromadzenia danych*). Dane są później ignorowane przez serwer, jeśli ustawiono *Bez gromadzenia danych*.

Po zakończeniu konserwacji *Bez gromadzenia danych* wyzwalacze używające funkcji `nodata()` nie zostaną uruchomione przed następnym sprawdzeniem w okresie, który sprawdzają.

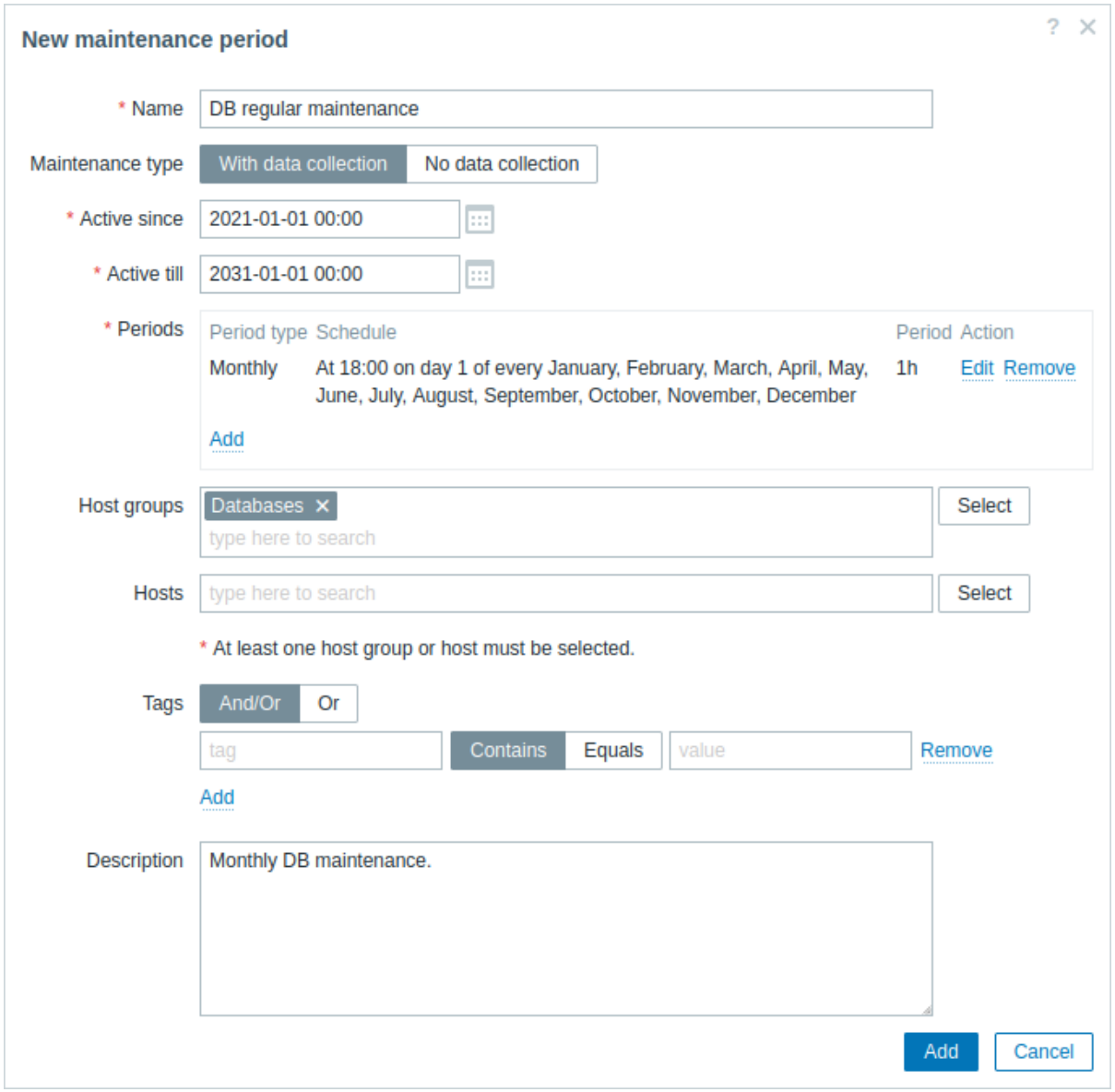
Jeśli pozycja logu zostanie dodana, gdy host jest w trakcie konserwacji, a konserwacja się zakończy, zostaną zebrane tylko nowe wpisy pliku dziennika od momentu zakończenia konserwacji.

Jeśli wartość ze znacznikiem czasu zostanie wysłana dla hosta, który jest objęty konserwacją *Bez gromadzenia danych* (np. przy użyciu Zabbix sender), wartość ta zostanie odrzucona; jednak możliwe jest wysłanie wartości ze znacznikiem czasu dla wygasłego okresu konserwacji i zostanie ona zaakceptowana.

Jeśli okres konserwacji, hosty, grupy lub tagi zostaną zmienione przez użytkownika, zmiany zaczną obowiązywać dopiero po synchronizacji pamięci podręcznej konfiguracji.

Konfiguracja Aby skonfigurować okres utrzymaniowy:

1. Przejdź do: *Data collection* > *Maintenance*.
2. Kliknij *Create maintenance period* (lub nazwę istniejącego okresu utrzymaniowego).
3. Wprowadź parametry utrzymania w formularzu.

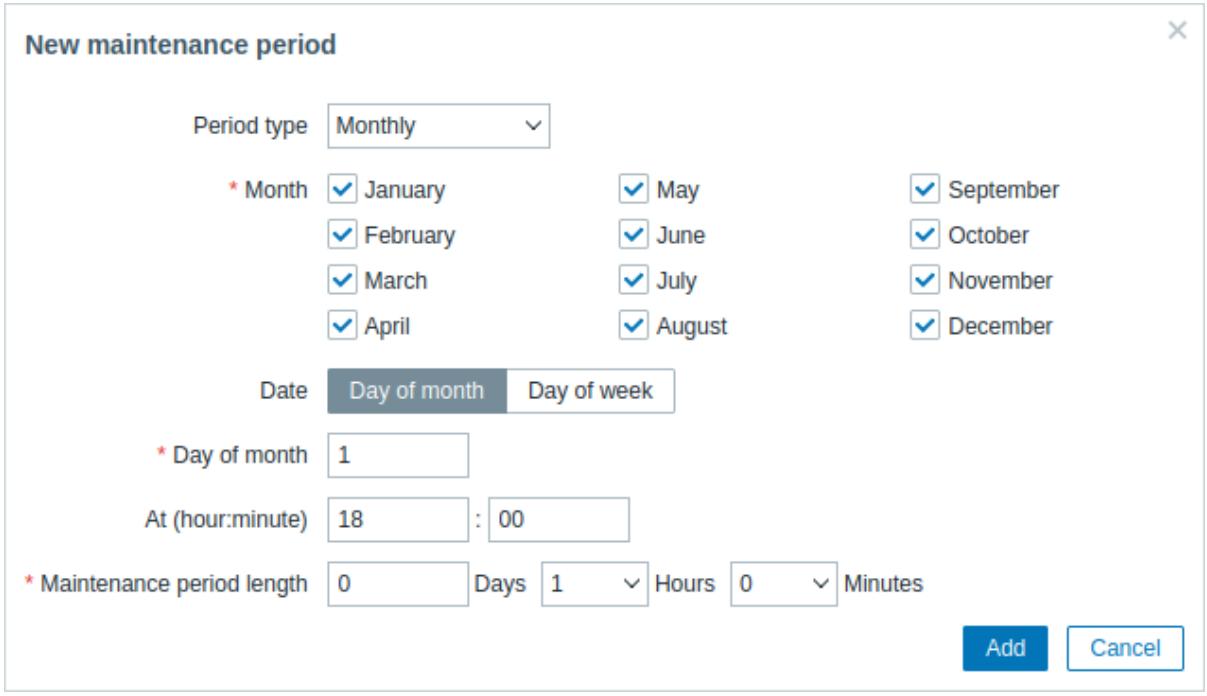


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Name	Nazwa okresu utrzymaniowego.
Maintenance type	Można ustawić dwa typy utrzymania: **With data collection** — dane są zbierane przez serwer podczas utrzymania, a wyzwalacze są przetwarzane; **No data collection** — dane mogą być nadal zbierane, ale nie są zapisywane do bazy danych podczas utrzymania, a wyzwalacze (w tym funkcja `nodata()`) nie są uruchamiane. Zobacz Wpływ okresów utrzymaniowych, aby dowiedzieć się, jak każdy typ wpływa na raport dostępności.
Active since	Data i godzina, od których wykonywanie okresów utrzymaniowych staje się aktywne. *Uwaga:* Ustawienie tylko tego czasu nie aktywuje okresu utrzymaniowego; okresy utrzymaniowe muszą być skonfigurowane w sekcji *Periods* (zobacz poniżej).
Active till	Data i godzina, do których wykonywanie okresów utrzymaniowych pozostaje aktywne.
Periods	Ten blok umożliwia zdefiniowanie dokładnych dni i godzin, w których odbywa się utrzymanie. Kliknięcie Add otwiera wyskakujące okno z elastycznym formularzem *Maintenance period*, w którym można zdefiniować harmonogram utrzymania. Szczegółowy opis znajduje się w sekcji Okresy utrzymaniowe.
Host groups	Wybierz grupy hostów, dla których utrzymanie zostanie aktywowane. Utrzymanie zostanie aktywowane dla wszystkich hostów z określonych grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania zostanie wyświetlona lista rozwijana wszystkich dostępnych grup hostów. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. W rezultacie utrzymanie zostanie również aktywowane na hostach z grup zagnieżdżonych.
Hosts	Wybierz hosty, dla których utrzymanie zostanie aktywowane. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania zostanie wyświetlona lista rozwijana wszystkich dostępnych hostów.
Tags	Określ tagi, aby wyciszać problemy z pasującymi tagami na hostach objętych utrzymaniem. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są dwa operatory: **Contains** — uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, z uwzględnieniem wielkości liter); **Equals** — uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** — wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** — wystarczy spełnienie jednego warunku. Tagi można określić tylko wtedy, gdy wybrano typ utrzymania *With data collection*.
Description	Opis okresu utrzymaniowego.

Okresy konserwacji

Okno okresu konserwacji służy do planowania czasu dla cyklicznej lub jednorazowej konserwacji. Formularz jest dynamiczny, a dostępne pola zmieniają się w zależności od wybranego *Typu okresu*.



Typ okresu	Opis
Tylko jednorazowo	Skonfiguruj jednorazowy okres konserwacji: *Data* - data i godzina okresu konserwacji; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna.
Codziennie	Skonfiguruj codzienny okres konserwacji: *Co ile dni* - częstotliwość konserwacji (1 - *(domyślnie)* codziennie, 2 - co dwa dni itd.); *O (godzina:minuta)* - pora dnia, o której rozpoczyna się konserwacja; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna. Gdy parametr *Co ile dni* jest większy niż „1”, dniem początkowym jest dzień, na który przypada czas *Aktywne od*. Przykłady: - jeśli *Aktywne od* ustawiono na „2021-01-01 12:00”, *Co ile dni* ustawiono na „2”, a *O (godzina:minuta)* ustawiono na „23:00”, to pierwszy okres konserwacji rozpocznie się 1 stycznia o 23:00, a drugi okres konserwacji rozpocznie się 3 stycznia o 23:00; - jeśli *Aktywne od* ustawiono na „2021-01-01 12:00”, *Co ile dni* ustawiono na „2”, a *O (godzina:minuta)* ustawiono na „01:00”, to pierwszy okres konserwacji rozpocznie się 3 stycznia o 01:00, a drugi okres konserwacji rozpocznie się 5 stycznia o 01:00.
Co tydzień	Skonfiguruj tygodniowy okres konserwacji: *Co ile tygodni* - częstotliwość konserwacji (1 - *(domyślnie)* co tydzień, 2 - co dwa tygodnie itd.); *Dzień tygodnia* - dzień, w którym ma odbywać się konserwacja; *O (godzina:minuta)* - pora dnia, o której rozpoczyna się konserwacja; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna. Gdy parametr *Co ile tygodni* jest większy niż „1”, tygodniem początkowym jest tydzień, na który przypada czas *Aktywne od*. Przykłady można znaleźć w opisie parametru *Codziennie* powyżej.
Co miesiąc	Skonfiguruj miesięczny okres konserwacji: *Miesiąc* - wybierz wszystkie miesiące, w których wykonywana jest regularna konserwacja; *Data:* ***Dzień miesiąca*** - wybierz tę opcję, jeśli konserwacja ma odbywać się tego samego dnia każdego miesiąca (na przykład każdego 1. dnia miesiąca), a następnie wybierz wymagany dzień w polu *Dzień miesiąca*, które się pojawi; *Data:* ***Dzień tygodnia*** - wybierz tę opcję, jeśli konserwacja ma odbywać się tylko w określone dni (na przykład w każdy pierwszy poniedziałek miesiąca), a następnie wybierz (z listy rozwijanej) wymagany tydzień miesiąca (pierwszy, drugi, trzeci, czwarty lub ostatni), po czym zaznacz pola wyboru dla dnia/dni konserwacji; *O (godzina:minuta)* - pora dnia, o której rozpoczyna się konserwacja; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna.

Attention:
Podczas tworzenia okresu utrzymania używana jest strefa czasowa użytkownika, który go tworzy. Jednak gdy planowane są cykliczne okresy utrzymania (*Codziennie*, *Co tydzień*, *Co miesiąc*), używana jest strefa czasowa serwera Zabbix. Aby zapewnić przewidywalne działanie cyklicznych okresów utrzymania, należy używać wspólnej strefy czasowej we wszystkich częściach Zabbix.

Po zakończeniu naciśnij *Dodaj*, aby dodać okres konserwacji do bloku *Okresy*.

Należy pamiętać, że zmiany czasu letniego (DST) nie wpływają na długość trwania konserwacji. Na przykład załóżmy, że mamy skonfigurowaną dwugodzinną konserwację, która zwykle rozpoczyna się o 01:00 i kończy o 03:00:

- jeśli po jednej godzinie konserwacji (o 02:00) nastąpi zmiana czasu letniego i bieżący czas zmieni się z 02:00 na 03:00, konserwacja będzie kontynuowana jeszcze przez jedną godzinę (do 04:00);
- jeśli po dwóch godzinach konserwacji (o 03:00) nastąpi zmiana czasu letniego i bieżący czas zmieni się z 03:00 na 02:00, konserwacja zostanie zatrzymana, ponieważ minęły dwie godziny;
- jeśli okres konserwacji rozpoczyna się w godzinie pomijanej z powodu zmiany czasu letniego, konserwacja nie rozpocznie się.

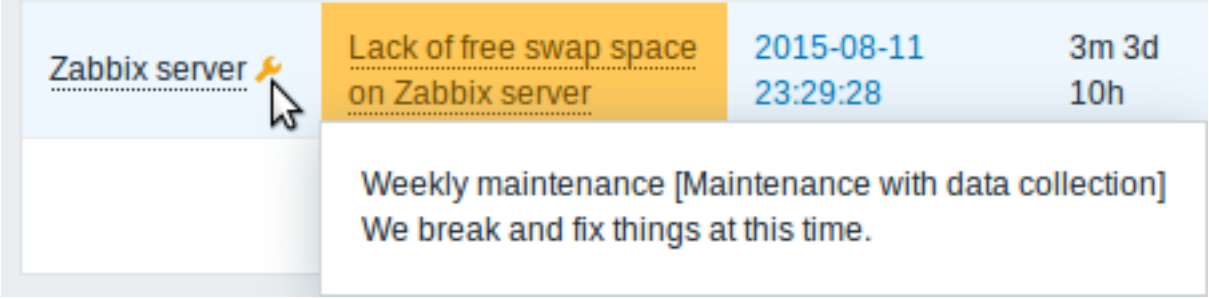
Jeśli okres konserwacji jest ustawiony na „1 dzień" (rzeczywisty okres konserwacji wynosi 24 godziny, ponieważ Zabbix oblicza dni w godzinach), rozpoczyna się o 00:00 i kończy o 00:00 następnego dnia:

- konserwacja zakończy się o 01:00 następnego dnia, jeśli bieżący czas zostanie przesunięty do przodu o jedną godzinę;
- konserwacja zakończy się o 23:00 tego dnia, jeśli bieżący czas zostanie cofnięty o jedną godzinę.

Wyświetlanie Wyświetlanie hostów w konserwacji

Pomarańczowa ikona klucza obok nazwy hosta wskazuje, że ten host jest w konserwacji w:

- *Dashboardach*
- *Monitoring* > *Problemy*
- *Inwentaryzacja* > *Hosty* > *Szczegóły inwentaryzacji hosta*
- *Zbieranie danych* > *Hosty* (zobacz kolumnę „Status")



Szczegóły konserwacji są wyświetlane po umieszczeniu wskaźnika myszy nad ikoną.

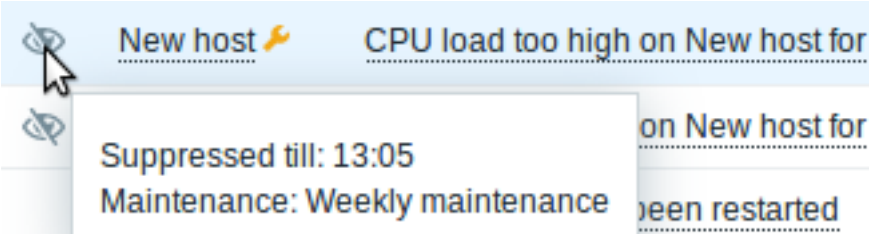
Dodatkowo hosty w konserwacji mają pomarańczowe tło w *Monitoring* > *Mapy*.

Wyświetlanie ukrytych problemów

Zazwyczaj problemy dla hostów w trybie konserwacji są ukrywane, tzn. nie są wyświetlane we frontendzie. Można jednak skonfigurować wyświetlanie ukrytych problemów, wybierając opcję *Pokaż ukryte problemy* w następujących miejscach:

- *Pulpity* (w konfiguracji widżetów *Hosty z problemami*, *Problemy*, *Problemy według ważności*, *Przegląd wyzwalaczy*)
- *Monitoring* > *Problemy* (w filtrze)
- *Monitoring* > *Mapy* (w konfiguracji mapy)
- Globalne powiadomienia (w konfiguracji profilu użytkownika)

Gdy ukryte problemy są wyświetlane, pojawia się następująca ikona: . Najechanie kursorem myszy na ikonę wyświetla więcej szczegółów.



Obliczanie kolejek podczas konserwacji Kolejki wyświetlane w frontendzie Zabbix (Administration > Queue) są obliczane przez serwer Zabbix. Nie obejmują one pozycji w konserwacji *No data collection*, a długość kolejki dla tych pozycji jest zawsze równa zero, nawet jeśli ich wartości są opóźnione. Opóźnione pozycje w konserwacji *With data collection* są nadal uwzględniane w kolejce.

Zabbix proxy nie jest świadomy okresów konserwacji, ponieważ nie ma synchronizacji konfiguracji konserwacji między serwerem Zabbix a proxy. Kontrole wewnętrzne obliczane na proxy Zabbix (na przykład `zabbix[queue,,]` i `zabbix[stats,,,queue,,]`) zgłaszają opóźnione pozycje niezależnie od stanu konserwacji na serwerze Zabbix.

W rezultacie frontend Zabbix oraz kontrole wewnętrzne na proxy Zabbix mogą raportować różne długości kolejek dla tych samych pozycji objętych konserwacją *No data collection*.

12 Wyrażenia regularne

Przegląd Perl Compatible Regular Expressions (PCRE, PCRE2) są obsługiwane w Zabbix.

W Zabbix istnieją dwa sposoby używania wyrażeń regularnych:

- ręczne wprowadzanie wyrażenia regularnego
- użycie globalnego wyrażenia regularnego utworzonego w Zabbix

Wyrażenia regularne W obsługiwanych miejscach można ręcznie wprowadzić wyrażenie regularne. Należy pamiętać, że wyrażenie nie może zaczynać się od @, ponieważ ten symbol jest używany w Zabbix do odwoływania się do globalnych wyrażeń regularnych.

> **Warning:**
> Podczas używania wyrażeń regularnych może dojść do wyczerpania stosu. Więcej informacji można znaleźć na stronie podręcznika pcrestack.

Należy pamiętać, że w dopasowywaniu wielowierszowym kotwice ^ i $ dopasowują się odpowiednio do początku/końca każdej linii, zamiast do początku/końca całego ciągu.

Zobacz także przykłady dotyczące poprawnego escapowania w różnych kontekstach.

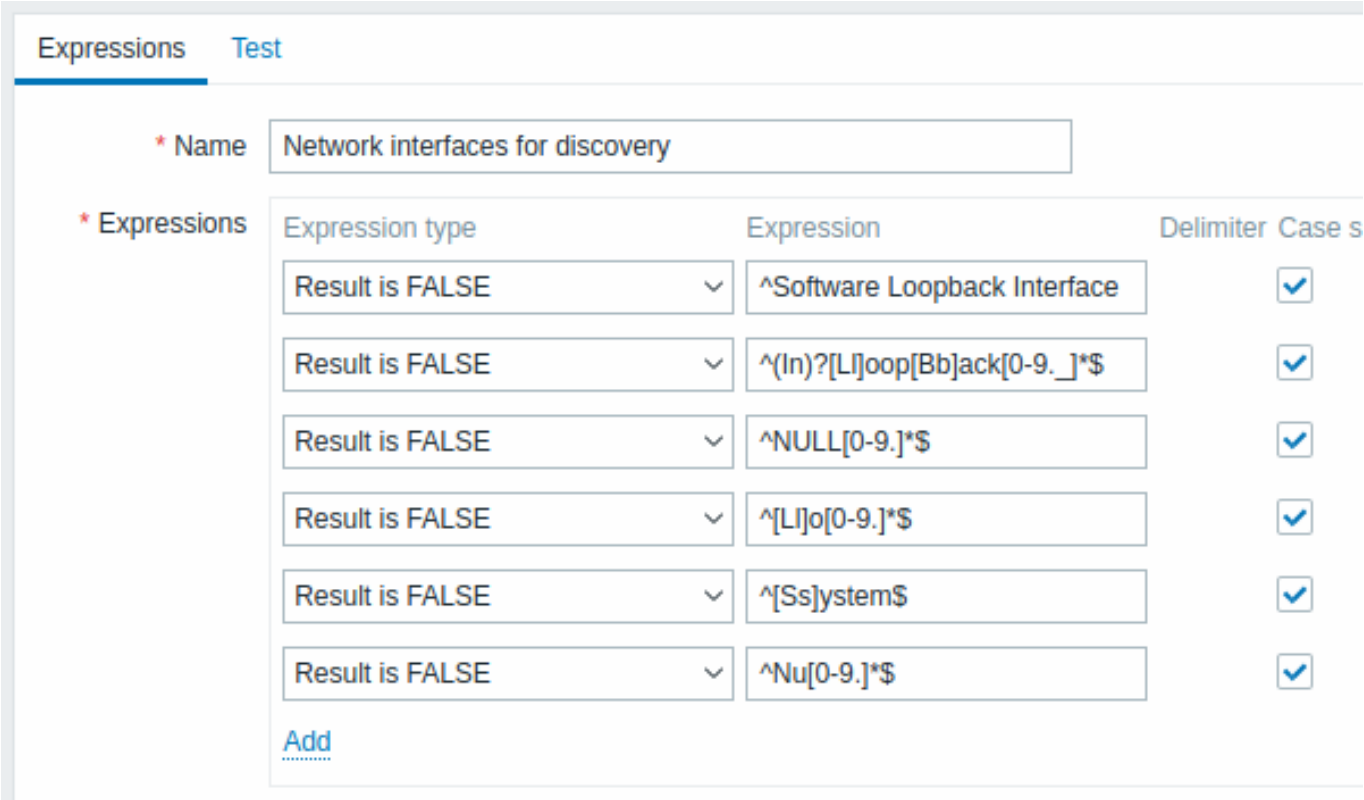
Globalne wyrażenia regularne W frontendzie Zabbix dostępny jest zaawansowany edytor do tworzenia i testowania złożonych wyrażeń regularnych.

Po utworzeniu wyrażenia regularnego w ten sposób można go używać w kilku miejscach w frontendzie, odwołując się do jego nazwy poprzedzonej symbolem @, na przykład *@mycustomregexp*.

Aby utworzyć globalne wyrażenie regularne:

- Przejdź do: *Administration* > *General*
- Wybierz *Regular expressions* z listy rozwijanej
- Kliknij *New regular expression*

Karta **Expressions** umożliwia ustawienie nazwy wyrażenia regularnego i dodanie podwyrażeń.



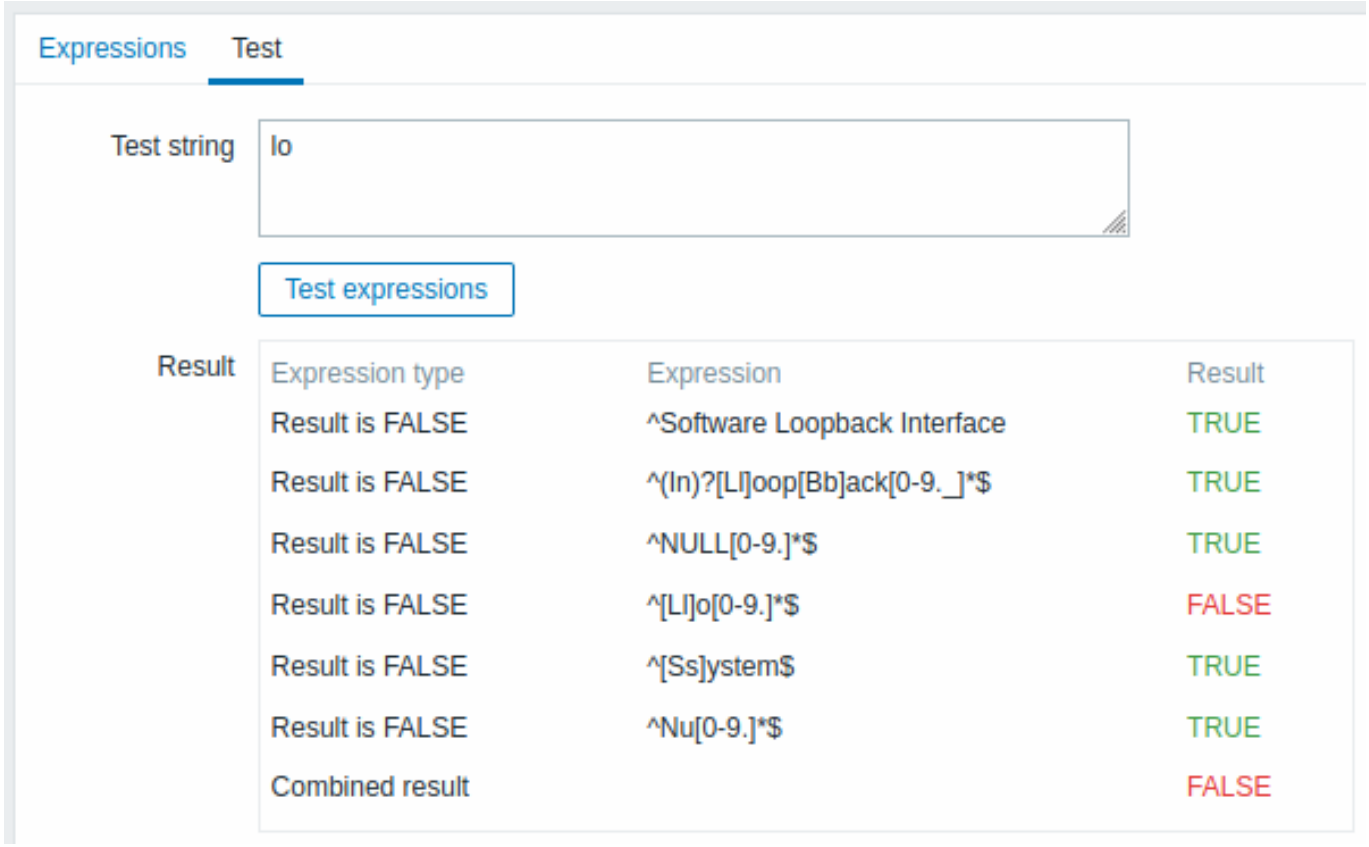
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Ustaw nazwę wyrażenia regularnego. Dozwolone są dowolne znaki Unicode.
Expressions	Kliknij *Add* w bloku Expressions, aby dodać nowe podwyrażenie.
Expression type	Wybierz typ wyrażenia: **Character string included** - dopasuj podciąg **Any character string included** - dopasuj dowolny podciąg z listy rozdzielanej znakami. Lista rozdzielana obejmuje przecinek (,), kropkę (.) lub ukośnik (/). **Character string not included** - dopasuj dowolny ciąg z wyjątkiem podciągu **Result is TRUE** - dopasuj wyrażenie regularne **Result is FALSE** - nie dopasowuj wyrażenia regularnego
Expression	Wprowadź podciąg/wyrażenie regularne.
Delimiter	Przecinek (,), kropka (.) lub ukośnik (/) służący do oddzielania ciągów tekstowych w wyrażeniu regularnym. Ten parametr jest aktywny tylko wtedy, gdy wybrano typ wyrażenia *"Any character string included"*.
Case sensitive	Pole wyboru określające, czy wyrażenie regularne rozróżnia wielkość liter.

Ukośnik (/) w wyrażeniu jest traktowany dosłownie, a nie jako separator. Dzięki temu można zapisywać wyrażenia zawierające ukośnik bez błędów.

Attention:
Niestandardowa nazwa wyrażenia regularnego w Zabbix może zawierać przecinki, spacje itp. W przypadkach, gdy może to prowadzić do błędnej interpretacji podczas odwoływania się do niej (na przykład przecinek w parametrze klucza pozycja), całe odwołanie można ująć w cudzysłów, na przykład: *"@My custom regexp for purpose1, purpose2"*.
 Nazw wyrażeń regularnych nie należy ujmować w cudzysłów w innych miejscach (na przykład we właściwościach reguły LLD).

Na karcie **Test** można testować wyrażenie regularne i jego podwyrażenia, podając ciąg testowy.



Wyniki pokazują status każdego podwyrażenia oraz łączny status niestandardowego wyrażenia.

Łączny status niestandardowego wyrażenia jest definiowany jako *Combined result*. Jeśli zdefiniowano kilka podwyrażeń, Zabbix używa operatora logicznego AND do obliczenia *Combined result*. Oznacza to, że jeśli co najmniej jeden wynik ma wartość False, *Combined result* również ma status False.

Domyślne globalne wyrażenia regularne Zabbix zawiera kilka globalnych wyrażeń regularnych w swoim domyślnym zestawie danych.

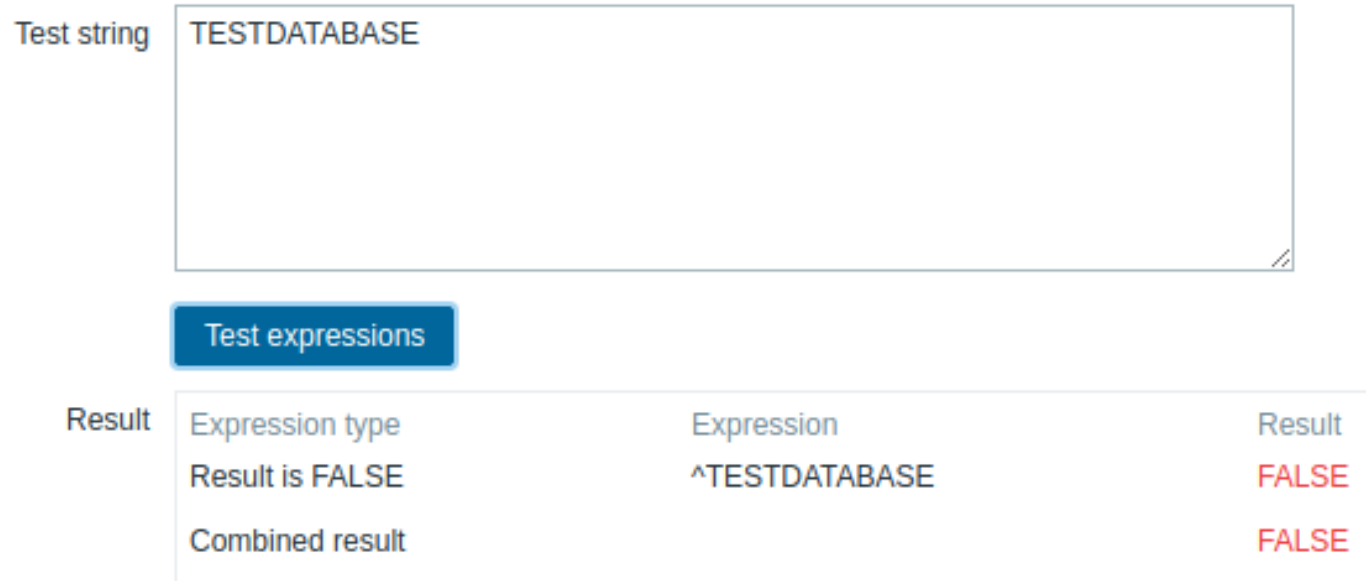
Name	Expression	Matches
File systems for discovery	`^(btrfs|ext2|ext3|ext4|jfs|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|refs|apfs|ntfs|fat32|zfs)$`	”btrfs” lub ”ext2” lub ”ext3” lub ”ext4” lub ”jfs” lub ”reiser” lub ”xfs” lub ”ffs” lub ”ufs” lub ”jfs” lub ”jfs2” lub ”vxfs” lub ”hfs” lub ”refs” lub ”apfs” lub ”ntfs” lub ”fat32” lub ”zfs”
Network interfaces for discovery	`^Software Loopback Interface`	Łańcuchy zaczynające się od ”Software Loopback Interface”.
	`^lo$`	”lo”
	`^(In)?[Ll]oop[Bb]ack[0-9._]*$`	Łańcuchy, które opcjonalnie zaczynają się od ”In”, następnie zawierają ”L” lub ”l”, potem ”oop”, następnie ”B” lub ”b”, potem ”ack”, po czym opcjonalnie może wystąpić dowolna liczba cyfr, kropek lub podkreśleń.
	`^NULL[0-9.]*$`	Łańcuchy zaczynające się od ”NULL”, po których opcjonalnie może wystąpić dowolna liczba cyfr lub kropek.
	`^[Ll]o[0-9.]*$`	Łańcuchy zaczynające się od ”Lo” lub ”lo” i opcjonalnie zakończone dowolną liczbą cyfr lub kropek.
	`^[Ss]ystem$`	”System” lub ”system”
	`^Nu[0-9.]*$`	Łańcuchy zaczynające się od ”Nu”, po których opcjonalnie może wystąpić dowolna liczba cyfr lub kropek.
Storage devices for SNMP discovery	`^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$`	”Physical memory” lub ”Virtual memory” lub ”Memory buffers” lub ”Cached memory” lub ”Swap space”

Name	Expression	Matches
Windows service names for discovery	`^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$`	”MMCSS”, ”gupdate” lub ”SysmonLog” lub łańcuchy takie jak ”clr_optimization_v2.0.50727_32” i ”clr_optimization_v4.0.30319_32”, gdzie zamiast kropek można wstawić dowolny znak z wyjątkiem znaku nowej linii.
Windows service startup states for discovery	`^(automatic|automatic delayed)$`	”automatic” lub ”automatic delayed”

Przykłady Przykład 1

Użycie poniższego wyrażenia w wykrywaniu niskiego poziomu do wykrywania baz danych z wyjątkiem bazy danych o określonej nazwie:

```
^TESTDATABASE$
```

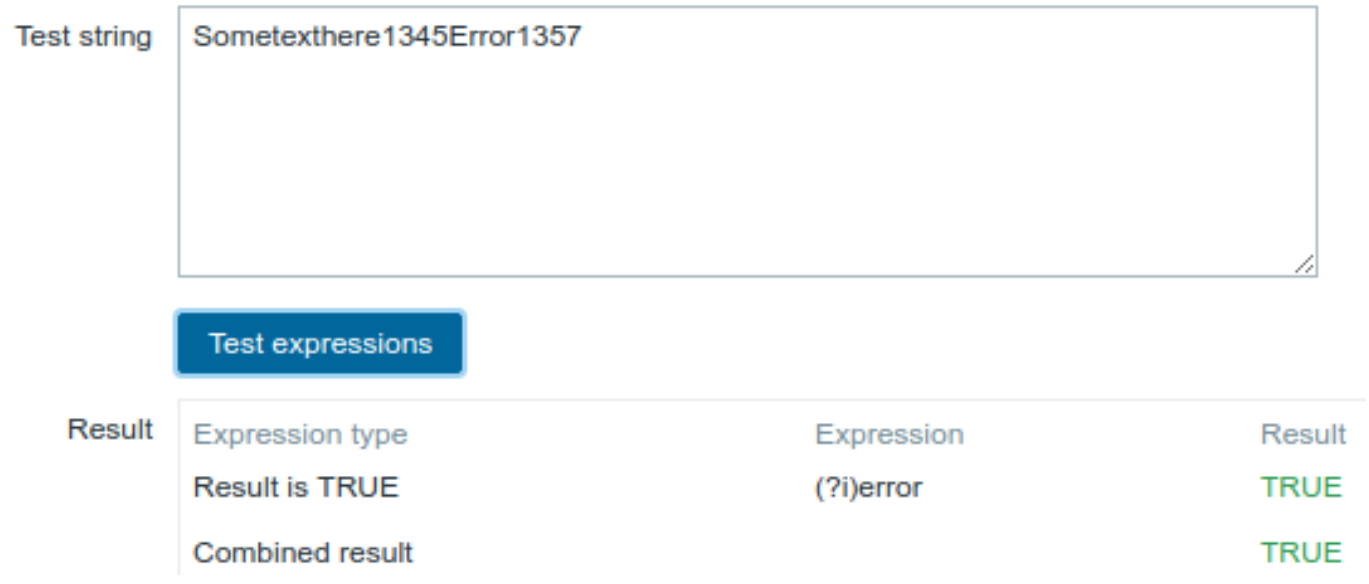


Wybrany *typ wyrażenia*: ”Wynik to FALSE”. Nie pasuje do nazwy zawierającej ciąg ”*TESTDATABASE*”.

Przykład z wbudowanym modyfikatorem wyrażenia regularnego

Użycie poniższego wyrażenia regularnego, w tym wbudowanego modyfikatora (?i), aby dopasować znaki ”error”:

```
(?i)error
```

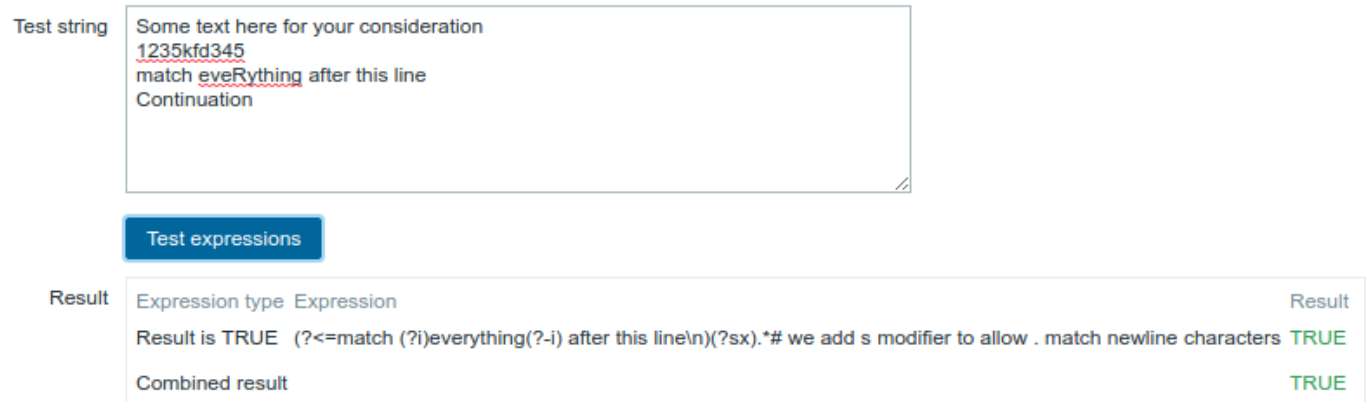


Wybrany *typ wyrażenia*: ”Wynik to TRUE”. Znaki ”error” są dopasowane.

Kolejny przykład z modyfikatorem regex w linii

Użycie następującego wyrażenia regularnego, w tym wielu modyfikatorów w linii, aby dopasować znaki po określonej linii:

```
(?<=match (?i)everything(?-i) after this line\n)(?sx).*# we add s modifier to allow . match newline charac
```



Wybrany typ wyrażenia: "Result is TRUE". Znaki po określonej linii są dopasowywane.

Attention:
Modyfikator **g** nie może być określony w linii. Listę dostępnych modyfikatorów można znaleźć na stronie podręcznika pcresyntax. Więcej informacji na temat składni PCRE można znaleźć w dokumentacji HTML PCRE.

Obsługa wyrażeń regularnych według lokalizacji

Lokalizacja	Wyrażenie regularne	Globalne wyrażenie regularne	Dopasowanie wielowierszowe	Komentarze
Pozycje agenta				
eventlog[]	Yes	Yes	Yes	parametry `regexp`, `severity`, `source`, `eventid`
eventlog.count[]				parametry `regexp`, `severity`, `source`, `eventid`
log[]				parametr `regexp`
log.count[]				
logrt[]		Yes/No		parametr `regexp` obsługuje oba, parametr `file_regexp` obsługuje tylko wyrażenia nieglobalne
logrt.count[]				
proc.cpu.util[]		No	No	parametr `cmdline`
proc.get[]				
proc.mem[]				
proc.num[]				
sensor[]				parametry `device` i `sensor` w systemie Linux 2.4
system.hw.macaddr[]				parametr `interface`
system.sw.packages[]				parametr `regexp`
system.sw.packages.get[]				parametr `regexp`
vfs.dir.count[]				parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
vfs.dir.get[]				parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
vfs.dir.size[]				parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
vfs.file.regexp[]			Yes	parametr `regexp`
vfs.file.regmatch[]				
web.page.regexp[]				
Trapy SNMP				
snmptrap[]	Yes	Yes	No	parametr `regexp`

Lokalizacja	Wyrażenie regularne	Globalne wyrażenie regularne	Dopasowanie wielowierszowe	Komentarze
Wstępne przetwarzanie wartości pozycji	Yes	No	No	pole `pattern`
Funkcje dla wyzwalaczy/pozycji obliczanych				
count()	Yes	Yes	Yes	parametr `pattern`, jeśli parametr `operator` ma wartość *regexp* lub *iregexp*
countunique()	Yes	Yes		
find()	Yes	Yes		
logeventid()	Yes	Yes	No	parametr `pattern`
logsource()				
Wykrywanie niskiego poziomu				
Filtry	Yes	Yes	No	pole *Regular expression*
Nadpisania	Yes	No		w opcjach *matches*, *does not match* dla warunków *Operation*
Warunki akcji	Yes	No	No	w opcjach *matches*, *does not match* dla warunków automatycznej rejestracji *Host name* i *Host metadata*
Skrypty	Yes	Yes	No	pole *Input validation rule*
Monitorowanie WWW	Yes	No	Yes	*Variables* z prefiksem **regex:** pole *Required string*
Kontekst makra użytkownika	Yes	No	No	w kontekście makra z prefiksem **regex:**
Funkcje makr				
regsub()	Yes	No	No	parametr `pattern`
iregsub()				
Wskaźniki łączy na mapach	Yes	No	No	pole *Pattern* (dla pozycji tekstowych)
Mapowanie ikon	Yes	Yes	No	pole *Expression*
Mapowanie wartości	Yes	No	No	pole *Value*, jeśli typ mapowania to `regexp`

13 Potwierdzanie problemu

Przegląd Zdarzenia problemowe w Zabbix mogą być potwierdzane przez użytkowników.

Jeśli użytkownik otrzyma powiadomienie o zdarzeniu problemowym, może przejść do frontend Zabbix, otworzyć okno podręczne aktualizacji problemu dla tego problemu jedną z metod wymienionych poniżej i potwierdzić problem. Podczas potwierdzania może wpisać komentarz, informując, że pracuje nad nim lub cokolwiek innego, co uzna za stosowne.

W ten sposób, jeśli inny użytkownik systemu zauważy ten sam problem, natychmiast zobaczy, czy został on potwierdzony, oraz jakie komentarze zostały dotychczas dodane.

W ten sposób proces rozwiązywania problemów przez więcej niż jednego użytkownika systemu może przebiegać w skoordynowany sposób.

Status potwierdzenia jest również używany podczas definiowania operacji akcji. Można na przykład zdefiniować, że powiadomienie zostanie wysłane do menedżera wyższego szczebla tylko wtedy, gdy zdarzenie nie zostanie potwierdzone przez pewien czas.

Aby potwierdzać zdarzenia i dodawać do nich komentarze, użytkownik musi mieć co najmniej uprawnienia do odczytu odpowiednich wyzwalaczy. Aby zmienić istotność problemu lub zamknąć problem, użytkownik musi mieć uprawnienia do odczytu i zapisu odpowiednich wyzwalaczy.

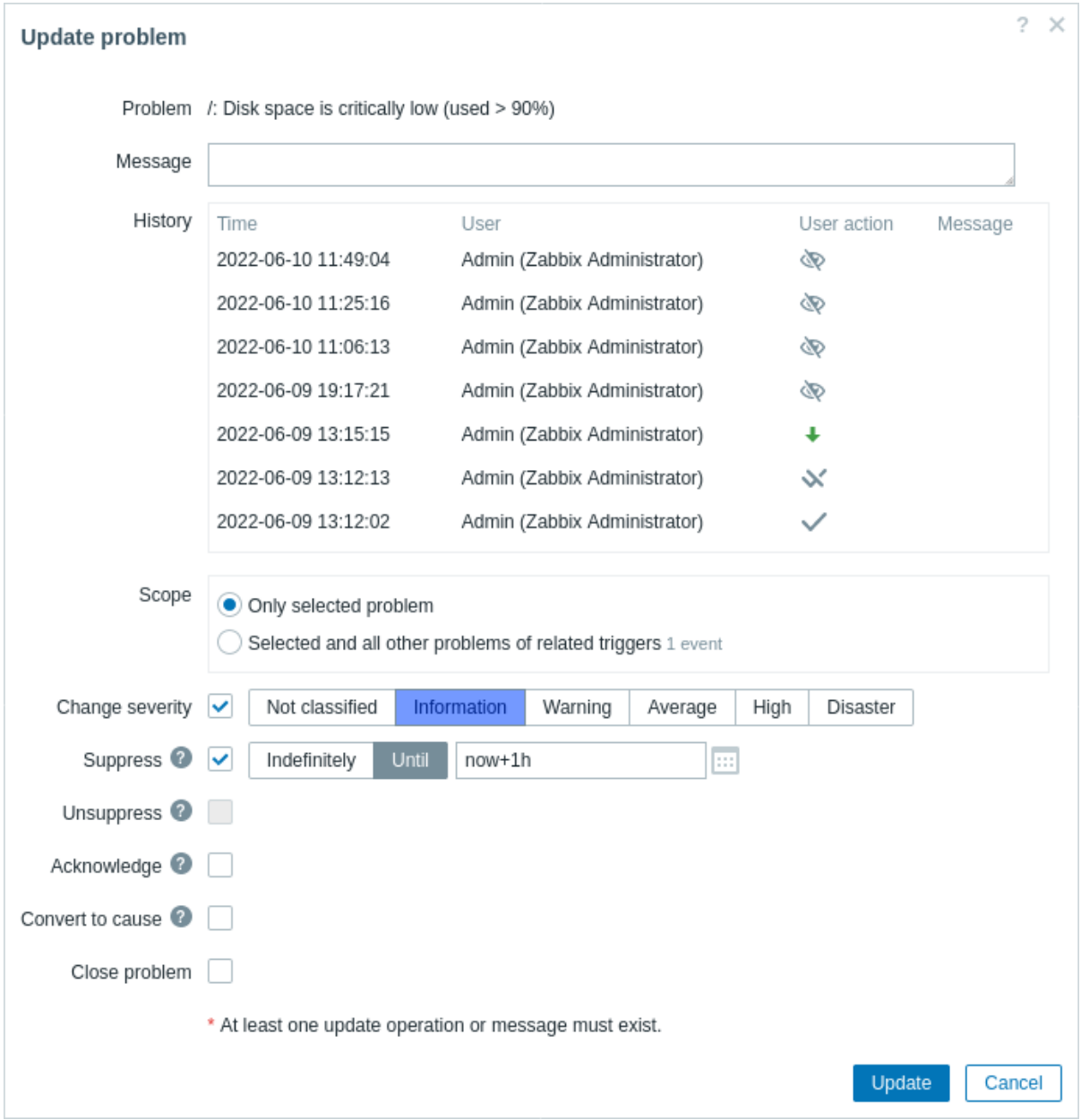
Istnieje **kilka** sposobów uzyskania dostępu do okna podręcznego aktualizacji problemu, które umożliwia potwierdzenie problemu.

- Możesz wybrać problemy w *Monitoring* > *Problemy*, a następnie kliknąć *Masowa aktualizacja* pod listą
- Możesz kliknąć *Aktualizuj* w kolumnie *Aktualizuj* dla problemu w:
  - *Pulpity* (widżety *Problemy* i *Problemy według istotności*)
  - *Monitoring* > *Problemy*
  - *Monitoring* > *Problemy* > *Szczegóły zdarzenia*
- Możesz kliknąć komórkę nierozwiązanego problemu w:
  - *Pulpity* (widżet *Przegląd wyzwalaczy*)

Menu podręczne zawiera opcję *Aktualizuj*, która przeniesie Cię do okna aktualizacji problemu.

Aktualizowanie problemów Wyskakujące okno aktualizacji problemów umożliwia:

- dodanie komentarza do problemu
- wyświetlenie dotychczasowych komentarzy i działań
- zmianę ważności problemu
- tłumienie/odtłumianie problemu
- potwierdzanie/cofanie potwierdzenia problemu
- zmianę problemu objawowego na problem przyczynowy
- ręczne zamknięcie problemu



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Problem	Jeśli wybrano tylko jeden problem, wyświetlana jest nazwa problemu. Jeśli wybrano kilka problemów, wyświetlany jest napis *Wybrano N problemów*.
Message	Wprowadź tekst komentarza do problemu (maksymalnie 2048 znaków).
History	Wyświetlana jest lista wcześniejszych działań i komentarzy dotyczących problemu wraz z czasem i danymi użytkownika. Znaczenie ikon używanych do oznaczania działań użytkownika opisano na stronie szczegółów zdarzenia. Należy pamiętać, że historia jest wyświetlana tylko wtedy, gdy do aktualizacji wybrano jeden problem.
Scope	Określ zakres działań takich jak zmiana ważności, potwierdzanie lub ręczne zamykanie problemów: **Only selected problem** - wpłynie tylko na to zdarzenie **Selected and all other problems of related triggers** - w przypadku potwierdzania/zamykania problemu wpłynie na to zdarzenie oraz wszystkie inne problemy, które nie zostały jeszcze potwierdzone/zamknięte. Jeśli zakres obejmuje problemy już potwierdzone lub zamknięte, nie będą one potwierdzane/zamykane ponownie. Z drugiej strony liczba operacji zmiany wiadomości i ważności nie jest ograniczona.

Parametr	Opis
Change severity	Zaznacz pole wyboru i kliknij przycisk ważności, aby zaktualizować ważność problemu. Pole wyboru zmiany ważności jest dostępne, jeśli istnieją uprawnienia do odczytu i zapisu dla co najmniej jednego z wybranych problemów. Po kliknięciu *Update* zaktualizowane zostaną tylko te problemy, dla których dostępny jest odczyt i zapis. Jeśli dla żadnego z wybranych wyzwalaczy nie istnieją uprawnienia do odczytu i zapisu, pole wyboru jest wyłączone.
Suppress	Zaznacz pole wyboru, aby stłumić problem: **Indefinitely** - tłumienie bezterminowe **Until** - tłumienie do określonego czasu. Obsługiwane są formaty czasu bezwzględnego i względnego, na przykład: `now+1d` - na jeden dzień od teraz (domyślnie) `now/w` - do końca bieżącego tygodnia `2022-05-28 12:00:00` - do bezwzględnej daty/godziny Należy pamiętać, że prosty okres (np. `1d`, `1w`) nie jest obsługiwany. Dostępność tej opcji zależy od ustawień roli użytkownika „Suppress problems". Zobacz także: Tłumienie problemów
Unsuppress	Zaznacz pole wyboru, aby odtłumić problem. To pole wyboru jest aktywne tylko wtedy, gdy co najmniej jeden z wybranych problemów jest stłumiony. Dostępność tej opcji zależy od ustawień roli użytkownika „Suppress problems".
Acknowledge	Zaznacz pole wyboru, aby potwierdzić problem. To pole wyboru jest dostępne, jeśli wśród wybranych problemów znajduje się co najmniej jeden niepotwierdzony problem. Nie można dodać kolejnego potwierdzenia do już potwierdzonego problemu (można jednak dodać kolejny komentarz).
Unacknowledge	Zaznacz pole wyboru, aby cofnąć potwierdzenie problemu. To pole wyboru jest dostępne, jeśli wśród wybranych problemów znajduje się co najmniej jeden potwierdzony problem.
Convert to cause	Zaznacz pole wyboru, aby przekształcić problem(y) objawowy(e) w problem(y) przyczynowy(e).
Close problem	Zaznacz pole wyboru, aby ręcznie zamknąć wybrany problem(y). Pole wyboru zamykania problemu jest dostępne, jeśli opcja *Allow manual close* jest zaznaczona w konfiguracji wyzwalacza dla co najmniej jednego z wybranych problemów. Po kliknięciu *Update* zamknięte zostaną tylko te problemy, które można zamknąć. Jeśli żaden problem nie może zostać ręcznie zamknięty, pole wyboru jest wyłączone. Problemy już zamknięte nie będą zamykane ponownie.

Wyświetlanie Na podstawie informacji o potwierdzeniach można skonfigurować sposób wyświetlania liczby problemów na pulpicie lub mapach. Aby to zrobić, należy dokonać wyboru w opcji *Wyświetlanie problemów*, dostępnej zarówno w konfiguracji mapy, jak i w widżecie pulpitu *Problemy według ważności* dashboard widget. Możliwe jest wyświetlanie całkowitej liczby problemów, liczby niepotwierdzonych problemów oddzielnie od całkowitej liczby lub tylko liczby niepotwierdzonych problemów.

Na podstawie informacji o aktualizacji problemu (potwierdzenie itp.) można skonfigurować operacje aktualizacji — wysłanie wiadomości lub wykonanie zdalnych poleceń.

1 Tłumienie problemów

Przegląd

Wyciszenie problemu umożliwia tymczasowe ukrycie problemu, którym można zająć się później. Jest to przydatne do uporządkowania listy problemów, aby nadać najwyższy priorytet najpilniejszym kwestiom. Na przykład czasami problem może pojawić się w weekend i nie być na tyle pilny, aby zajmować się nim natychmiast, więc można go „odłożyć" do poniedziałkowego poranka.

Wyciszenie problemu pozwala ukryć *pojedynczy* problem, w przeciwieństwie do wyciszenia problemów poprzez konserwację hosta, gdy ukrywane są wszystkie problemy hosta objętego konserwacją.

Operacje dla akcji wyzwalaczy będą wstrzymane dla wyciszonych problemów w taki sam sposób, jak ma to miejsce w przypadku konserwacji hosta.

Konfiguracja

Problem można stłumić za pomocą okna **aktualizacja problemu**, gdzie stłumienie jest jedną z opcji aktualizacji problemu, obok komentowania, zmiany powagi, potwierdzania znajomości itd.

Problem można również przywrócić za pomocą tego samego okna aktualizacji problemu.

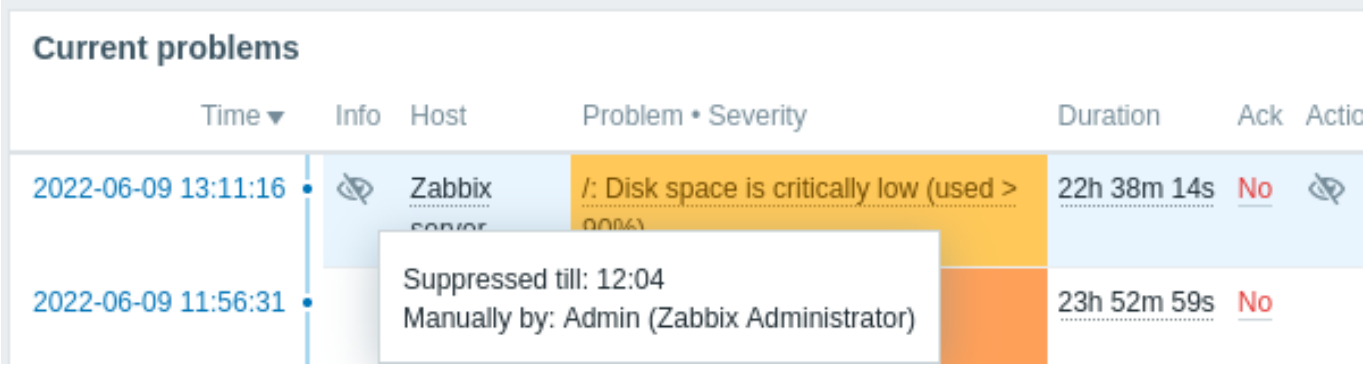
Wyświetlanie

Po wyciszeniu problem jest oznaczany migającą ikoną wyciszenia w kolumnie *Info*, zanim zostanie ukryty.

Ikona wyciszenia miga, gdy zadanie wyciszenia znajduje się na liście oczekujących. Gdy menedżer zadań wyciszy problem, ikona przestanie migać. Jeśli ikona wyciszenia miga przez dłuższy czas, może to wskazywać na problem z serwerem, na przykład gdy serwer jest niedostępny i menedżer zadań nie może ukończyć zadania. Ta sama logika dotyczy cofnięcia wyciszenia. W krótkim czasie po przesłaniu zadania, gdy serwer jeszcze go nie ukończył, ikona cofnięcia wyciszenia miga.

Wyciszony problem może być zarówno ukryty, jak i wyświetlany, w zależności od ustawień filtra problemów/widgetu.

Gdy jest wyświetlany na liście problemów, wyciszony problem jest oznaczony ikoną wyciszenia, a szczegóły wyciszenia są wyświetlane po najechaniu kursorem:



Szczegóły wyciszenia są również wyświetlane w wyskakującym oknie po najechaniu kursorem na ikonę wyciszenia w kolumnie *Actions*.

14 Eksport/import konfiguracji

Przegląd Funkcjonalność eksportu/importu w Zabbix umożliwia wymianę różnych elementów konfiguracji między jednym systemem Zabbix a drugim.

Typowe przypadki użycia tej funkcjonalności:

- udostępnianie szablonów lub map sieciowych — użytkownicy Zabbix mogą udostępniać swoje parametry konfiguracji
- przesyłanie szablonu do Zabbix Community templates. Następnie inni mogą pobrać szablon i zaimportować plik do Zabbix.
- integracja z narzędziami innych firm — uniwersalne formaty YAML, XML i JSON umożliwiają integrację oraz import/eksport danych za pomocą narzędzi i aplikacji innych firm

Co można eksportować/importować

Obiekty, które można eksportować/importować to:

- Grupy hostów (*tylko przez Zabbix API*)
- Grupy szablonów (*tylko przez Zabbix API*)
- Szablony
- Hosty
- Mapy sieci
- Typy mediów
- Obrazy

Format eksportu

Dane można eksportować za pomocą interfejsu użytkowania Zabbix lub Zabbix API. Obsługiwane formaty eksportu to YAML, XML i JSON.

Szczegóły eksportu

- Wszystkie obsługiwane elementy są eksportowane do jednego pliku.
- Encje hosta i szablonu (pozycje, wyzwalacze, wykresy, reguły wykrywania), które są dziedziczone z podłączonych szablonów, nie są eksportowane. Wszelkie zmiany wprowadzone w tych encjach na poziomie hosta (takie jak zmieniony interwał pozycji, zmodyfikowane wyrażenie regularne lub dodane prototypy do reguły wykrywania niskiego poziomu) zostaną utracone

podczas eksportu; podczas importu wszystkie encje z podłączonych szablonów są odtwarzane tak jak w oryginalnym podłączonym szablonie.
- Encje utworzone przez wykrywanie niskiego poziomu oraz wszelkie encje od nich zależne nie są eksportowane. Na przykład wyzwalacz utworzony dla pozycji wygenerowanej przez regułę LLD nie zostanie wyeksportowany.
- Jeśli eksportowany host/szablon zawiera encje obsługujące limity czasu, wartości limitu czasu zostaną wyeksportowane, jeśli dla tych encji skonfigurowano własne limity czasu.

Szczegóły dotyczące importu

- Import zatrzymuje się przy pierwszym błędzie.
- Podczas aktualizacji istniejących obrazów w trakcie importu obrazów pole "imagetype" jest ignorowane, tj. nie można zmienić typu obrazu za pomocą importu.
- Podczas importowania hostów/szablonów z użyciem opcji "Delete missing" makra hosta/szablonu, które nie są obecne w pliku importu, zostaną usunięte z
hosta/szablonu po imporcie.
- Puste tagi dla pozycji, wyzwalaczy, wykresów, discoveryRules, itemPrototypes, triggerPrototypes, graphPrototypes nie mają znaczenia, tj. to tak, jakby ich brakowało.
- Jeśli encje importowanego hosta/szablonu mają skonfigurowane własne limity czasu, zostaną one zastosowane; w przeciwnym razie zostaną zastosowane limity czasu proxy/globalne.
- Import obsługuje YAML, XML i JSON, a plik importu musi mieć poprawne rozszerzenie pliku: .yaml i .yml dla YAML, .xml dla XML oraz .json dla JSON. Zobacz informacje o kompatybilności dotyczące obsługiwanych wersji XML.
- Import obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 (z BOM lub bez); inne kodowania (UTF16LE, UTF16BE, UTF32LE, UTF32BE itd.) spowodują błąd konwersji podczas importu.

Podstawowy format YAML Format eksportu YAML zawiera następujące węzły:

- Węzeł główny dla eksportu YAML Zabbix
- Wersja eksportu

```
zabbix_export:
  version: '7.4'
```

Pozostałe węzły zależą od eksportowanych obiektów.

Format XML Format eksportu XML zawiera następujące znaczniki:

- Domyślny nagłówek dokumentów XML
- Znacznik główny dla eksportu XML Zabbix
- Wersja eksportu

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>7.4</version>
</zabbix_export>
```

Pozostałe znaczniki zależą od eksportowanych obiektów.

Format JSON Format eksportu JSON zawiera następujące obiekty:

- Obiekt główny dla eksportu JSON Zabbix
- Wersja eksportu

```
{
    "zabbix_export": {
        "version": "7.4"
    }
}
```

Pozostałe obiekty zależą od eksportowanych obiektów.

1 Grupy szablonów

Przegląd

W frontend grupy szablonów mogą być eksportowane tylko razem z eksportem szablonów. Gdy szablon jest eksportowany, wszystkie grupy, do których należy, są eksportowane wraz z nim automatycznie.

API umożliwia eksportowanie grup szablonów niezależnie od szablonów.

Format eksportu

```
  template_groups:
    - uuid: 36bff6c29af64692839d077febfc7079
      name: 'Urządzenia sieciowe'
```

Wyeksportowane elementy

Element	Type	Description
uuid	string	Unikalny identyfikator tej grupy szablonów.
name	string	Nazwa grupy.

2 Grupy hostów

Przegląd

W interfejsie użytkownika grupy hostów mogą być eksportowane tylko za pomocą eksportu hosta. Gdy host jest eksportowany, wszystkie grupy, do których należy, są automatycznie eksportowane wraz z nim.

API umożliwia eksportowanie grup hostów niezależnie od hostów.

Format eksportu

```
host_groups:
- uuid: 6f6799aa69e844b4b3918f779f2abf08
name: 'Serwery Zabbix'
```

Eksportowane elementy

Element	Typ	Opis
uuid	string	Unikalny identyfikator grupy hostów.
name	string	Nazwa grupy.

###3 Szablony {#manual-xml_export_import-templates}

Przegląd

Szablony są eksportowane wraz z wieloma powiązanymi obiektami i relacjami między obiektami.

Eksport szablonu zawiera:

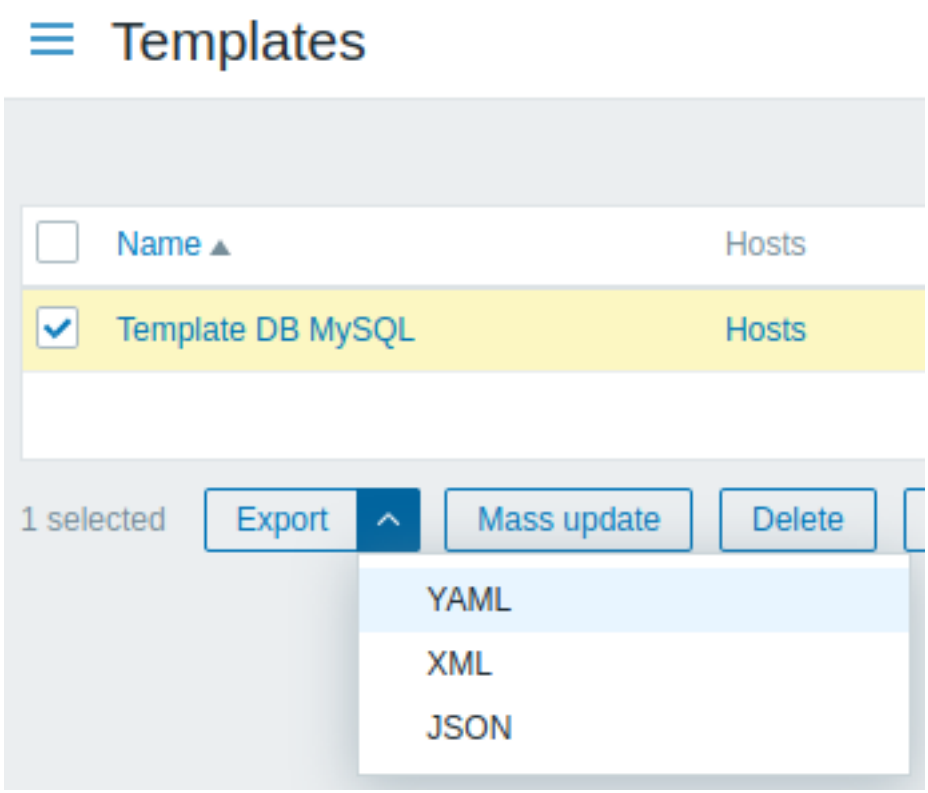
- Powiązane grupy szablonów
- Powiązane grupy hostów (jeśli są używane w konfiguracji prototypu hosta)
- Dane szablonu
- Powiązania z innymi szablonami
- Powiązania z grupami szablonów
- Bezpośrednio powiązane pozycje
- Bezpośrednio powiązane wyzwalacze
- Bezpośrednio powiązane wykresy
- Bezpośrednio powiązane pulpity
- Bezpośrednio powiązane reguły wykrywania ze wszystkimi prototypami
- Bezpośrednio powiązane scenariusze WWW
- Mapy wartości

Eksportowanie

Aby wyeksportować szablony, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Templates*.
2. Zaznacz pola wyboru szablonów do wyeksportowania.

3. Kliknij *Export* poniżej listy.



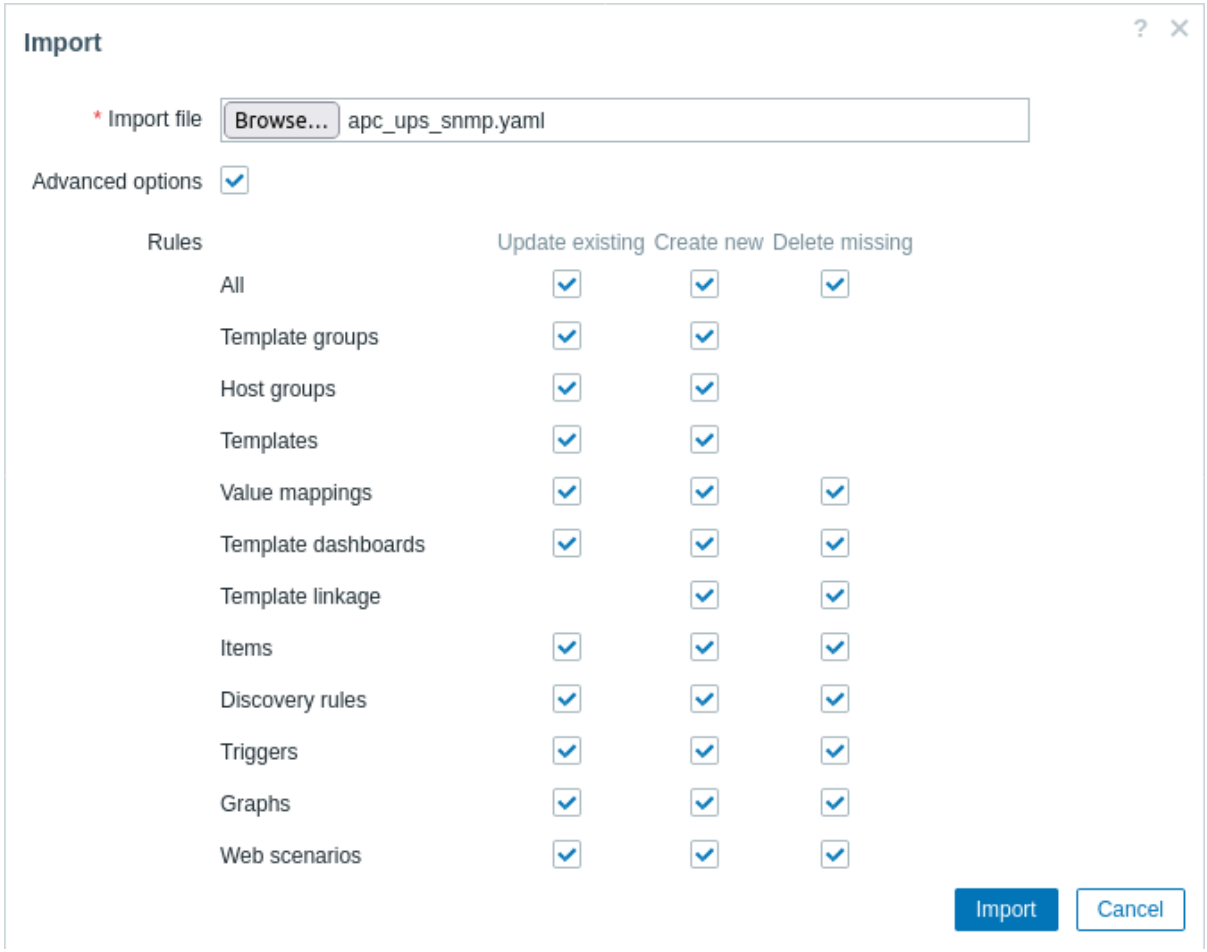
W zależności od wybranego formatu szablony są eksportowane do pliku lokalnego z domyślną nazwą:

- `zabbix_export_templates.yaml` — w eksporcie YAML (domyślna opcja eksportu);
- `zabbix_export_templates.xml` — w eksporcie XML;
- `zabbix_export_templates.json` — w eksporcie JSON.

Importowanie

Aby zaimportować szablony, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Templates*.
2. Kliknij *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Kliknij *Import* w prawym dolnym rogu formularza konfiguracji.



Jeśli zaznaczysz pole wyboru *Advanced options*, zostanie wyświetlona szczegółowa lista wszystkich elementów, które można importować — zaznacz lub odznacz każdą regułę importu zgodnie z potrzebami.

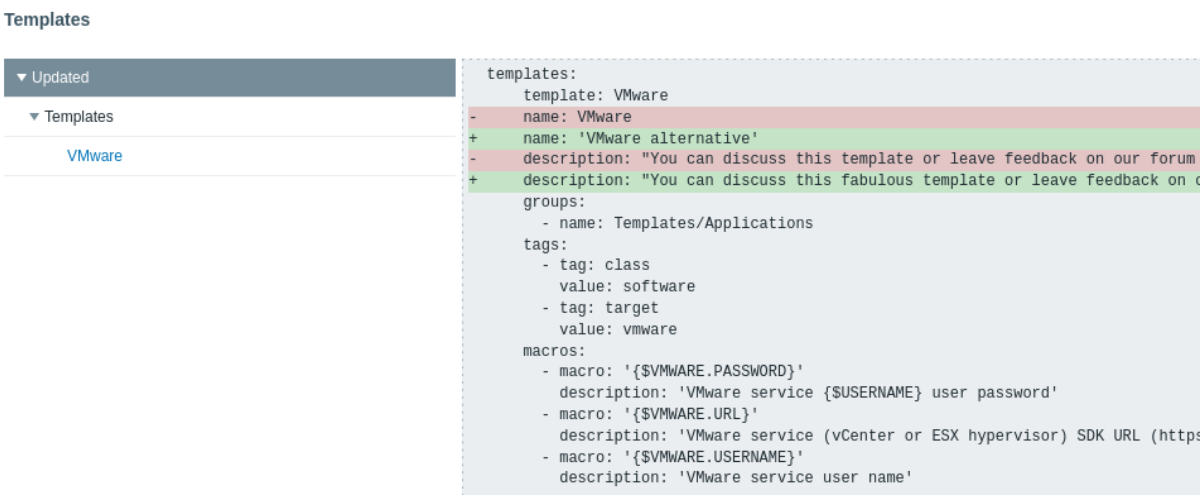
Jeśli klikniesz pole wyboru w wierszu *All*, wszystkie elementy poniżej zostaną zaznaczone/odznaczone.

Reguły importu:

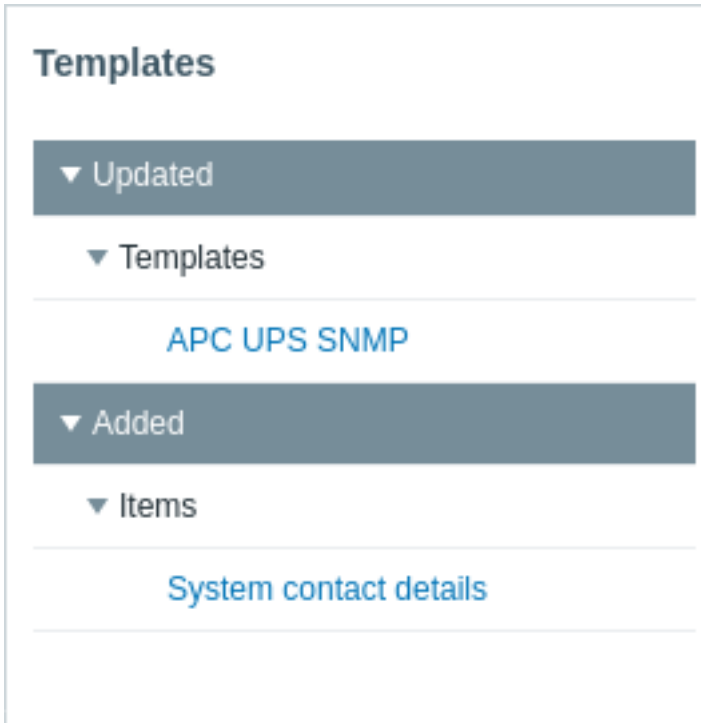
Rule	Description
Update existing	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.
Delete missing	Istniejące elementy, których nie ma w pliku importu, zostaną usunięte. W przeciwnym razie nie zostaną usunięte. Jeśli opcja *Delete missing* jest zaznaczona dla *Template linkage*, bieżące powiązania szablonów, których nie ma w pliku importu, zostaną odłączone. Encje (pozycje, wyzwalacze, wykresy itd.) dziedziczone z odłączonych szablonów nie zostaną usunięte (chyba że opcja *Delete missing* zostanie również zaznaczona dla każdej encji).

Attention:
Jeśli szablony o tych samych nazwach już istnieją, podczas importu należy zaznaczyć opcje *Delete missing*, aby uzyskać czysty import. W ten sposób stare pozycje, których nie ma już w zaktualizowanym szablonie, zostaną usunięte (należy pamiętać, że będzie to oznaczać utratę historii tych starych pozycji).

Na następnym ekranie będzie można wyświetlić zawartość importowanego szablonu. Jeśli jest to nowy szablon, wszystkie elementy będą wyświetlane na zielono. W przypadku aktualizacji istniejącego szablonu nowe elementy szablonu będą wyróżnione na zielono; usunięte elementy szablonu będą wyróżnione na czerwono; elementy, które nie uległy zmianie, będą wyświetlane na szarym tle.



Menu po lewej stronie może służyć do poruszania się po liście zmian. Sekcja *Updated* wyróżnia wszystkie zmiany wprowadzone w istniejących elementach szablonu. Sekcja *Added* zawiera listę nowych elementów szablonu. Elementy w każdej sekcji są pogrupowane według typu elementu; kliknij szarą strzałkę, aby rozwinąć lub zwinąć grupę elementów.



Przejrzyj zmiany w szablonie, a następnie kliknij *Import*, aby wykonać import szablonu. W frontend zostanie wyświetlony komunikat o powodzeniu lub niepowodzeniu importu.

Format eksportu Po wyeksportowaniu szablonu Zabbix generuje uporządkowany format YAML, JSON lub XML. Eksport obejmuje elementy szablonu, takie jak metadane szablonu, pozycje, makra, wyzwalacze, pulpity i inne.

Każdy element pełni określoną funkcję i może zawierać elementy zagnieżdżone.

Poniższe sekcje opisują każdy element w formacie eksportu. W przykładach użyto szablonu Linux by Zabbix agent.

Wielokropek (`...`) oznacza elementy pominięte dla zwięzłości. Uwaga (`see table below`) jest używana, gdy dany element jest wyjaśniony bardziej szczegółowo w dalszych sekcjach.

```
zabbix_export:
  version: '7.4'
  template_groups:
    - uuid: 846977d1dfed4968bc5f8bdb363285bc
      name: 'Templates/Operating systems'
  templates: (see table below)
```

```
  triggers: (see table below)
  graphs: (see table below)
```

Metadane szablonu

Element	Type	Opis
version	string	(wymagane) Wersja eksportu.
template_groups		(wymagane) Element główny dla grup szablonów.
uuid	string	(wymagane) Unikalny identyfikator tej grupy szablonów.
name	string	(wymagane) Nazwa grupy szablonów.
host_groups		Element główny dla grup hostów używanych przez prototypy hostów.
uuid	string	(wymagane) Unikalny identyfikator tej grupy hostów.
name	string	(wymagane) Nazwa grupy hostów.
templates		Element główny dla szablonów.
triggers		Element główny dla elementów wyzwalaczy szablonu, które są takie same jak dla wyzwalaczy pozycji szablonu.
graphs		Element główny dla elementów wykresów szablonu, które są takie same jak dla wykresów hosta.

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      template: 'Linux by Zabbix agent'
      name: 'Linux by Zabbix agent'
      description: |
        To oficjalny szablon dla systemu Linux. Wymaga Zabbix agent 7.4 lub nowszego. (...)
      wizard_ready: 'YES'
      readme: |
        ## Omówienie

        To oficjalny szablon dla systemu Linux. Wymaga Zabbix agent 7.4 lub nowszego. (...)
      vendor:
        name: Zabbix
        version: 7.4-2
      groups:
        - name: 'Templates/Operating systems'
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: class
          value: os
        - tag: target
          value: linux
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          value: 3m
          description: 'Limit czasu, po którym agent jest uznawany za niedostępny. Działa tylko dla agentó
          config: (see table below)
        - macro: '{$CPU.UTIL.CRIT}'
          value: '90'
          description: 'Krytyczny próg użycia CPU wyrażony w %.'
          config: (see table below)
        (...)
      dashboards: (see table below)
      valuemaps: (see table below)
    (...)
```

Szablony

Element	Type	Description
uuid	string	(wymagane) Unikalny identyfikator tego szablonu.
template	string	(wymagane) Unikalna nazwa szablonu.
name	string	Widoczna nazwa szablonu.
description	text	Opis szablonu.
wizard_ready	text	Określa, czy szablon jest dostępny do wyboru w Kreatorze hosta. Możliwe wartości:[1] NO (0, domyślnie), YES (1). Zobacz także: Obiekt Template (`wizard_ready`).
readme	text	Instrukcje konfiguracji specyficzne dla szablonu, wyświetlane w Kreatorze hosta. Obsługuje formatowanie Markdown.
vendor		Element nadrzędny dla dostawcy szablonu (obecny, jeśli wyeksportowany szablon zawiera dane dostawcy).
name	string	(wymagane) Nazwa dostawcy szablonu.
version	string	(wymagane) Wersja szablonu. Dla szablonów gotowych do użycia wersja jest wyświetlana w następujący sposób: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład 7.0-0, 7.0-3, 7.4-0, 7.4-3.
templates		Element nadrzędny dla połączonych szablonów.
name	string	(wymagane) Nazwa szablonu.
groups		Element nadrzędny dla grup szablonów.
name	string	(wymagane) Nazwa grupy szablonów.
items		Element nadrzędny dla pozycji szablonu.
discovery_rules		Element nadrzędny dla reguł niskopoziomowego wykrywania szablonu.
httptests		Element nadrzędny dla scenariuszy web szablonu.
tags		Element nadrzędny dla tagów szablonu.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
macros		Element nadrzędny dla makr użytkownika szablonu.
macro	string	(wymagane) Nazwa makra użytkownika.
type	string	Typ makra użytkownika. Możliwe wartości:[1] TEXT (0, domyślnie), SECRET_TEXT (1), VAULT (2). Zobacz także: Obiekt User macro (`type`).
value	string	Wartość makra użytkownika.
description	string	Opis makra użytkownika.
config		Element nadrzędny dla konfiguracji makra szablonu, odpowiedzialny za sposób wyświetlania makra w Kreatorze hosta.
dashboards		Element nadrzędny dla dashboardów szablonu.
valuemaps		Element nadrzędny dla map wartości szablonu.

Pozycje szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        - uuid: f94f9f4699e94c369e6c98b2a2f485ce
          name: 'Zabbix agent ping'
          key: agent.ping
          description: 'Agent zawsze zwraca dla tej pozycji wartość "1". Może być używany w połączeniu z `
          valuemap:
            name: 'Zabbix agent ping status'
          tags:
            - tag: component
              value: system
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          name: 'Liczba zainstalowanych pakietów'
          key: system.sw.packages.get
          delay: 1h
          preprocessing: (zobacz tabelę poniżej)
```

```
      tags:
        - tag: component
          value: os
      triggers: (zobacz tabelę poniżej)
    - uuid: 403cebed115441369e94d35d070ca7b8
      name: 'Wykorzystanie pamięci'
      type: DEPENDENT
      key: vm.memory.utilization
      value_type: FLOAT
      units: '%'
      description: 'Procent użytej pamięci jest obliczany jako `100-pavailable`.'
      preprocessing: (zobacz tabelę poniżej)
      master_item:
        key: 'vm.memory.size[pavailable]'
      tags:
        - tag: component
          value: memory
      triggers: (zobacz tabelę poniżej)
    (...)
```

Element	Type	Description
uuid	string	(wymagane) Unikalny identyfikator tej pozycji.
name	string	(wymagane) Nazwa pozycji.
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), CALCULATED (15), JMX (16), SNMP_TRAP (17), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22).
snmp_oid	string	(wymagane dla pozycji SNMP_AGENT) ID obiektu SNMP.
key	string	(wymagane) Klucz pozycji.
delay	string	Interwał aktualizacji pozycji. Domyślnie: 1m. Wartość będzie zawsze równa 0 dla pozycji TRAP.
history	string	Okres czasu (z użyciem przyrostka czasu, makra użytkownika lub makra LLD), przez jaki dane historyczne mają być przechowywane. Domyślnie: 31d.
trends	string	Okres czasu (z użyciem przyrostka czasu, makra użytkownika lub makra LLD), przez jaki dane trendów mają być przechowywane. Domyślnie: 365d.
status	string	Status pozycji. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
value_type	string	Typ odebranej wartości. Możliwe wartości:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, domyślnie), TEXT (4), BINARY (5).
allowed_hosts	string	Lista adresów IP hostów, oddzielonych przecinkami, które mogą wysyłać dane dla tej pozycji. Obsługiwane dla pozycji TRAP i HTTP_AGENT.
units	string	Jednostki odebranej wartości (bps, B, itp.).
params	text	Dodatkowe parametry zależne od typu pozycji (wykonywany skrypt dla pozycji SSH i TELNET; zapytanie SQL dla pozycji ODBC; formuła dla pozycji CALCULATED; skrypt dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER).
ipmi_sensor	string	Czujnik IPMI. Obsługiwane dla pozycji IPMI.
authtype	string	Typ uwierzytelniania. Obsługiwane dla pozycji SSH i HTTP_AGENT. Możliwe wartości dla pozycji SSH:[1] PASSWORD (0, domyślnie), PUBLIC_KEY (1). Możliwe wartości dla pozycji HTTP_AGENT:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
username	string	(wymagane dla pozycji SSH i TELNET) Nazwa użytkownika do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `password` (patrz poniżej) albo pozostawić oba elementy puste.

Element	Type	Description
password	string	(wymagane dla pozycji SSH i TELNET) Hasło do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `username` (patrz powyżej) albo pozostawić oba elementy puste.
publickey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza publicznego.
privatekey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza prywatnego.
description	text	Opis pozycji.
inventory_link	string	Pole inwentarza hosta, które jest wypełniane przez pozycję. Możliwe wartości:[1] NONE (0), ALIAS (4) itd. (obsługiwane pola: zobacz Host inventory).
valuemap		Element główny map wartości pozycji.
name	string	(wymagane) Nazwa mapy wartości używanej dla pozycji.
logtimefmt	string	Format czasu w wpisach dziennika. Obsługiwane dla pozycji typu LOG.
preprocessing		Element główny wstępnego przetwarzania wartości pozycji.
step		Element główny kroków wstępnego przetwarzania wartości pozycji szablonu.
jmx_endpoint	string	Punkt końcowy JMX. Obsługiwane dla pozycji JMX.
master_item		(wymagane dla pozycji DEPENDENT) Element główny dla pozycji nadrzędnej pozycji zależnej.
key	string	(wymagane) Klucz pozycji nadrzędnej pozycji zależnej.
timeout	string	Limit czasu żądania odpytywania danych pozycji. Obsługiwane dla listy typów pozycji Timeouts.
url	string	(wymagane dla pozycji HTTP_AGENT) Ciąg URL.
query_fields		Element główny parametrów zapytania. Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania. Obsługiwane dla pozycji HTTP_AGENT.
parameters		Element główny parametrów zdefiniowanych przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
name	string	(wymagane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER) Nazwa parametru zdefiniowanego przez użytkownika.
value	string	Wartość parametru zdefiniowanego przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
posts	string	Dane treści żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługiwane dla pozycji HTTP_AGENT.
follow_redirects	string	Podążaj za przekierowaniami odpowiedzi podczas odpytywania danych. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
post_type	string	Typ treści danych POST. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (2), XML (3).
http_proxy	string	Ciąg połączenia proxy HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
headers		Element główny nagłówków żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa nagłówka.
value	string	(wymagane dla pozycji HTTP_AGENT) Wartość nagłówka.
retrieve_mode	string	Która część odpowiedzi ma zostać zapisana. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
request_method	string	Typ metody żądania. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] GET (0, domyślnie), POST (1), PUT (2), HEAD (3).
output_format	string	Sposób przetwarzania odpowiedzi. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (1).
allow_traps	string	Zezwól na wypełnianie wartości podobnie jak w przypadku pozycji trapper. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).

Element	Type	Description
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
verify_peer	string	Czy sprawdzać, czy certyfikat hosta jest autentyczny. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Czy sprawdzać, czy nazwa hosta dla połączenia odpowiada nazwie w certyfikacie hosta. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
tags		Element główny tagów pozycji.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
triggers		Element główny wyzwalaczy pozycji szablonu.

Note:
Zobacz też: Item object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Kroki wstępnego przetwarzania wartości pozycji szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.length()
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 12h
          (...)
```

Element	Type	Opis
type	string	(wymagane) Typ kroku wstępnego przetwarzania wartości pozycji. Możliwe wartości:[1] MULTIPLIER (1), RTRIM (2), LTRIM (3), TRIM (4), REGEX (5), BOOL_TO_DECIMAL (6), OCTAL_TO_DECIMAL (7), HEX_TO_DECIMAL (8), SIMPLE_CHANGE (9, obliczane: otrzymana wartość - poprzednia wartość), CHANGE_PER_SECOND (10, obliczane: (otrzymana wartość - poprzednia wartość)/(bieżący czas - czas ostatniego sprawdzenia)), XMLPATH (11), JSONPATH (12), IN_RANGE (13), MATCHES_REGEX (14), NOT_MATCHES_REGEX (15), CHECK_JSON_ERROR (16), CHECK_XML_ERROR (17), CHECK_REGEX_ERROR (18), DISCARD_UNCHANGED (19), DISCARD_UNCHANGED_HEARTBEAT (20), JAVASCRIPT (21), PROMETHEUS_PATTERN (22), PROMETHEUS_TO_JSON (23), CSV_TO_JSON (24), STR_REPLACE (25), CHECK_NOT_SUPPORTED (26), XML_TO_JSON (27), SNMP_WALK_VALUE (28), SNMP_WALK_TO_JSON (29), SNMP_GET_VALUE (30).
parameters		(wymagane) Element główny dla parametrów kroku wstępnego przetwarzania wartości pozycji.
parameter	string	Indywidualny parametr kroku wstępnego przetwarzania wartości pozycji.
error_handler	string	Typ akcji używany w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości:[1] ORIGINAL_ERROR (0, domyślnie), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parametry obsługi błędów.

Note:
Zobacz także: Item preprocessing object (odpowiednia właściwość o pasującej nazwie).

Wyzwalacze pozycji szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          triggers:
            - uuid: b950c306394f4b3c902060a8273cbcde
              expression: 'change(/Linux by Zabbix agent/system.sw.packages.get)<>0'
              name: 'Linux: Liczba zainstalowanych pakietów uległa zmianie'
              priority: WARNING
              manual_close: 'YES'
              tags:
                - tag: scope
                  value: notice
        (...)
        - uuid: 403cebed115441369e94d35d070ca7b8
          (...)
          triggers:
            - uuid: cfd395b1cde74ef18a5e5f840bd5142a
              expression: 'min(/Linux by Zabbix agent/vm.memory.utilization,5m)>{$MEMORY.UTIL.MAX}'
              name: 'Linux: Wysokie wykorzystanie pamięci'
              event_name: 'Linux: Wysokie wykorzystanie pamięci (>{$MEMORY.UTIL.MAX}% przez 5m)'
              priority: AVERAGE
              description: 'Systemowi kończy się wolna pamięć.'
              dependencies:
                - name: 'Linux: Brak dostępnej pamięci'
                  expression: 'max(/Linux by Zabbix agent/vm.memory.size[available],5m)<{$MEMORY.AVAILABLE
              tags:
                - tag: scope
                  value: capacity
                - tag: scope
                  value: performance
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego wyzwalacza.
expression	string	(wymagane) Wyrażenie wyzwalacza.
recovery_mode	string	Podstawa generowania zdarzeń OK. Możliwe wartości:[1] EXPRESSION (0, domyślnie), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza.
correlation_mode	string	Tryb korelacji (brak korelacji zdarzeń lub korelacja zdarzeń według tagu). Możliwe wartości:[1] DISABLED (0, domyślnie), TAG_VALUE (1).
correlation_tag	string	Nazwa tagu używanego do korelacji zdarzeń.
name	string	(wymagane) Nazwa wyzwalacza.
event_name	string	Nazwa zdarzenia.
opdata	string	Dane operacyjne.
url_name	string	Etykieta adresu URL powiązanego z wyzwalaczem.
url	string	Adres URL powiązany z wyzwalaczem.
status	string	Status wyzwalacza. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
priority	string	Poziom ważności wyzwalacza. Możliwe wartości:[1] NOT_CLASSIFIED (0, domyślnie), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).
description	text	Opis wyzwalacza.

Element	Type	Opis
type	string	Typ generowania zdarzeń (pojedyncze zdarzenie problemu lub wiele zdarzeń problemu). Możliwe wartości:[1] SINGLE (0, domyślnie), MULTIPLE (1).
manual_close	string	Ręczne zamykanie zdarzeń problemu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
dependencies		Element główny dla zależności.
name	string	(wymagane) Nazwa zależnego wyzwalacza.
expression	string	(wymagane) Wyrażenie zależnego wyzwalacza.
recovery_expression	string	Wyrażenie odzyskiwania zależnego wyzwalacza.
tags		Element główny dla tagów wyzwalacza.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz także: Trigger object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Reguły wykrywania niskiego poziomu szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      discovery_rules:
        - uuid: acfdea9c46ef48c68e6636f43b8f96a2
          name: 'Wykrywanie interfejsów sieciowych'
          key: net.if.discovery
          delay: 1h
          filter: (zobacz tabelę poniżej)
          description: 'Wykrywanie interfejsów sieciowych.'
          item_prototypes: (zobacz tabelę poniżej)
          trigger_prototypes: (zobacz tabelę poniżej)
          graph_prototypes: (zobacz tabelę poniżej)
        (...)
```

Attention:
Większość elementów reguł wykrywania niskiego poziomu szablonu jest taka sama jak w przypadku pozycji szablonu. Poniższa tabela opisuje te elementy, które różnią się od pozycji szablonu.

Element	Type	Opis
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), JMX (16), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22).
key	string	(wymagane) Klucz reguły wykrywania niskiego poziomu.
filter		Element główny dla filtrów reguł wykrywania niskiego poziomu szablonu.
lifetime	string	Okres czasu (z użyciem sekund, sufiksu czasu lub makra użytkownika), po którym zasoby, które nie są już wykrywane, zostaną usunięte. Domyślnie: 7d.
lifetime_type	string	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Okres czasu (z użyciem sekund, sufiksu czasu lub makra użytkownika), po którym zasoby, które nie są już wykrywane, zostaną wyłączone.
enabled_lifetime_type	string	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Element główny dla elementów prototypów pozycji szablonu, które są takie same jak dla pozycji szablonu.
trigger_prototypes		Element główny dla elementów prototypów wyzwalaczy szablonu, które są takie same jak dla wyzwalaczy pozycji szablonu.

Element	Type	Opis
graph_prototypes		Element główny dla elementów prototypów wykresów szablonu, które są takie same jak dla wykresów hosta.
host_prototypes		Element główny dla elementów prototypów hostów szablonu, które są takie same jak dla hostów.
parent_discovery_rule		Element główny dla nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły) prototypu reguły wykrywania niskiego poziomu. Ta właściwość oznacza, że jest to prototyp reguły LLD, będący bezpośrednim elementem podrzędnym wskazanej reguły/prototypu reguły.
key	string	(wymagane) Klucz nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły).
master_item	string	(wymagane dla reguł DEPENDENT) Element główny dla pozycji nadrzędnej reguły zależnej.
lld_macro_paths		Element główny dla ścieżek makr reguły wykrywania niskiego poziomu.
lld_macro	string	(wymagane) Nazwa makra reguły wykrywania niskiego poziomu.
path	string	(wymagane) Selektor wartości, która zostanie przypisana do odpowiedniego makra.
preprocessing		Element główny dla wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu.
step		Element główny dla elementów kroków wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu, które są takie same jak dla kroków wstępnego przetwarzania wartości pozycji szablonu, z wyjątkiem mniejszej liczby możliwych wartości. Zobacz także: obiekt przetwarzania wstępnego reguły LLD.
overrides		Element główny dla reguł nadpisywania reguły wykrywania niskiego poziomu.
name	string	(wymagane) Unikalna nazwa nadpisania.
step	string	(wymagane) Unikalny numer kolejności nadpisania.
stop	string	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli pasuje.
filter		Element główny dla elementów filtrów reguł nadpisywania reguły wykrywania niskiego poziomu szablonu, które są takie same jak dla filtrów reguł wykrywania niskiego poziomu szablonu.
operations		Element główny dla operacji nadpisywania reguły wykrywania niskiego poziomu szablonu.

Note:
Zobacz także: obiekt reguły LLD (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Filtry reguł wykrywania niskiego poziomu szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      discovery_rules:
        - uuid: acfdea9c46ef48c68e6636f43b8f96a2
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.MATCHES}'
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
          (...)
        - uuid: 2bbdc79f082d4c618e01bec625e9c90a
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.MATCHES}'
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.NOT_MATCHES}'
```

```
            operator: NOT_MATCHES_REGEX
          - macro: '{#DEVTYPE}'
            value: disk
      (...)
```

Element	Type	Opis
evaltype	string	Zastąp metodę oceny warunków filtra. Możliwe wartości:[1] AND_OR (0, domyślnie), AND (1), OR (2), FORMULA (3).
formula	string	Niestandardowa formuła obliczeniowa dla warunków filtra.
conditions		Element główny dla warunków filtra.
macro	string	(wymagane) Nazwa makra wykrywania niskiego poziomu, dla którego ma zostać wykonane sprawdzenie.
value	string	Wartość do porównania.
operator	string	Operator warunku. Możliwe wartości:[1] MATCHES_REGEX (8, domyślnie), NOT_MATCHES_REGEX (9).
formulaid	string	(wymagane) Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków.

Note:
Zobacz także: LLD rule filter object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Operacje nadpisania reguły wykrywania niskiego poziomu szablonu

Element	Type	Opis
operationobject	string	Obiekt, do którego stosowana jest operacja. Możliwe wartości:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3).
operator	string	Operator warunku nadpisania. Możliwe wartości:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Wyrażenie regularne lub ciąg znaków dla operatora warunku nadpisania.
status	string	Status obiektu po operacji nadpisania.
discover	string	Czy obiekt jest dodawany w wyniku wykrywania.
delay	string	Interwał aktualizacji ustawiony dla prototypu pozycji po operacji nadpisania.
history	string	Okres przechowywania historii ustawiony dla prototypu pozycji po operacji nadpisania.
trends	string	Okres przechowywania trendów ustawiony dla prototypu pozycji po operacji nadpisania.
severity	string	Ważność prototypu wyzwalacza ustawiona po operacji nadpisania.
tags		Element główny dla tagów ustawionych dla obiektu po operacji nadpisania.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
templates		Element główny dla szablonów podłączonych do prototypu hosta po operacji nadpisania.
name	string	(wymagane) Nazwa szablonu.
inventory_mode	string	Tryb inwentarza prototypu hosta ustawiony po operacji nadpisania.

Note:
Zobacz także: obiekt operacji nadpisania reguły LLD (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Scenariusze WWW szablonu

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego scenariusza WWW.
name	string	(wymagane) Nazwa scenariusza WWW.
delay	string	Częstotliwość wykonywania scenariusza WWW (z użyciem sekund, sufiksu czasu lub makra użytkownika). Domyślnie: 1m.

Element	Type	Opis
attempts	integer	Liczba prób wykonania kroków scenariusza WWW. Możliwe wartości:[1] 1-10 (domyślnie: 1).
agent	string	agent klienta. Zabbix będzie udawał wybraną przeglądarkę. Jest to przydatne, gdy strona internetowa zwraca różną zawartość dla różnych przeglądarek. Domyślnie: Zabbix.
http_proxy	string	proxy, który będzie używany przez scenariusz WWW, podany w postaci: `http://[username[:password]@]proxy.example.com[:port]`
variables		Element główny dla zmiennych scenariusza WWW, które mogą być używane w krokach scenariusza.
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny dla nagłówków HTTP wysyłanych podczas wykonywania żądania. Nagłówki powinny być wymienione przy użyciu tej samej składni, w jakiej występują w protokole HTTP.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
status	string	Status scenariusza WWW. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
authentication	string	Metoda uwierzytelniania. Możliwe wartości:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
http_user	string	Nazwa użytkownika używana do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
http_password	string	Hasło używane do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
verify_peer	string	Weryfikacja certyfikatu SSL serwera WWW. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Weryfikacja zgodności pola *Common Name* lub pola *Subject Alternate Name* certyfikatu serwera WWW. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło pliku klucza prywatnego SSL.
steps		(wymagane) Element główny dla kroków scenariusza WWW szablonu.
tags		Element główny dla tagów scenariusza WWW.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz także: Obiekt scenariusza WWW (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Kroki scenariusza WWW szablonu

Element	Type	Opis
name	string	(wymagane) Nazwa kroku scenariusza WWW.
url	string	(wymagane) URL do monitorowania.
query_fields		Element główny parametrów zapytania (tablica pól HTTP, które mają zostać dodane do URL podczas wykonywania żądania).
name	string	(wymagane) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania.
posts		Element główny zmiennych HTTP POST (ciąg znaków (surowe dane POST) lub tablica pól HTTP (dane pól formularza)).
name	string	(wymagane) Nazwa pola POST.
value	string	(wymagane) Wartość pola POST.

Element	Type	Opis
variables		Element główny zmiennych na poziomie kroku (makr), które powinny zostać zastosowane po tym kroku. Jeśli wartość zmiennej ma prefiks ’regex:’, jej wartość jest wyodrębniana z danych zwróconych przez ten krok zgodnie ze wzorcem wyrażenia regularnego następującym po prefiksie ’regex:’
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny nagłówków HTTP, które mają zostać wysłane podczas wykonywania żądania.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
follow_redirects	string	Podążanie za przekierowaniami HTTP. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
retrieve_mode	string	Tryb pobierania odpowiedzi HTTP. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
timeout	string	Limit czasu wykonania kroku (z użyciem sekund, sufiksu czasu lub makra użytkownika). Domyślnie: 15s.
required	string	Tekst, który musi występować w odpowiedzi (ignorowany, jeśli pusty).
status_codes	string	Rozdzielana przecinkami lista akceptowanych kodów statusu HTTP (np. 200-201,210-299; ignorowana, jeśli pusta).

Note:
Zobacz także: Obiekt kroku scenariusza WWW (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Konfiguracja makr szablonu

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      (...)
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          (...)
          config:
            type: TEXT
            priority: '1'
            label: 'Liczba sekund od ostatniego wykrycia agenta Zabbix'
            description: 'Limit czasu, po którym agent jest uznawany za niedostępny.'
        - macro: '{$CPU.UTIL.CRIT}'
          (...)
          config:
            type: TEXT
            priority: '2'
            section_name: Progi
            label: 'Próg wykorzystania CPU wyrażony'
            description: 'Krytyczny próg wykorzystania CPU wyrażony w %. W zakresie od 0 do 100 włącznie.'
            regex: '^-?([0-9]+|(([0-9]+)\.([0-9]+)))$'
        (...)
        - macro: '{$IFCONTROL}'
          (...)
          config:
            type: CHECKBOX
            priority: '19'
            label: 'Kontrola interfejsów'
            description: 'Uruchom wyzwalacz, gdy stan operacyjny interfejsu zmieni się na "Link down".'
            options:
              - checked: '1'
                unchecked: '0'
        (...)
```

Element	Type	Opis
type	string	(wymagane) Typ pola wejściowego makra. Możliwe wartości:[1] NOCONF (0), TEXT (1), LIST (2), CHECKBOX (3).
priority	string	Pozycja makra na liście makr.
section_name	string	Etykieta zwijanej sekcji, w której makro jest grupowane.
label	string	(wymagane dla makr TEXT, LIST, CHECKBOX) Etykieta makra.
description	text	Tekst pomocy makra. Obsługuje formatowanie Markdown.
required	string	Oznacza makro jako obowiązkowe. Możliwe wartości:[1] NO (0), YES (1). Obsługiwane dla makr TEXT i LIST.
regex	string	Wyrażenie regularne do walidacji danych wejściowych użytkownika w polu tekstowym. Obsługiwane dla makr TEXT.
options		Element główny dla elementów LIST lub wartości CHECKBOX.
value	string	(wymagane dla makr LIST) Wartość elementu LIST.
text	string	(wymagane dla makr LIST) Tekst elementu LIST.
checked	string	(wymagane dla makr CHECKBOX) Wartość reprezentująca stan zaznaczony. Możliwe wartości:[1] 0 (false), 1 (true). Obsługiwane dla makr CHECKBOX.
unchecked	string	(wymagane dla makr CHECKBOX) Wartość reprezentująca stan niezaznaczony. Możliwe wartości:[1] 0 (false), 1 (true). Obsługiwane dla makr CHECKBOX.

Note:
Zobacz także: Macro configuration object (odpowiednia właściwość o pasującej nazwie).

Pulpity szablonów

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
          pages:
            - name: Overview
              widgets: (zobacz tabelę poniżej)
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego pulpitu.
name	string	(wymagane) Nazwa pulpitu szablonu.
display pe-riod	integer	Okres wyświetlania stron pulpitu.
auto_start	string	Automatyczne uruchamianie pokazu slajdów. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
pages		Element główny dla stron pulpitu szablonu.
name	string	Nazwa strony.
display period	integer	Okres wyświetlania strony.
sortorder	integer	Kolejność sortowania strony.
widgets		Element główny dla widgetów pulpitu szablonu.

Note:
Zobacz także: obiekt pulpitu szablonu (odpowiednia właściwość o pasującej nazwie).

Widżety pulpitu szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
          pages:
            - name: Overview
              widgets:
                - type: graphprototype
                  width: '72'
                  height: '5'
                  fields:
                    - type: INTEGER
                      name: columns
                      value: '3'
                    - type: GRAPH_PROTOTYPE
                      name: graphid.0
                      value:
                        host: 'Linux by Zabbix agent'
                        name: 'FS [{#FSTYPE}({#FSNAME})]: Space utilization chart (relative to total)'
                    - type: STRING
                      name: reference
                      value: AAAEB
                - type: graphprototype
                  'y': '5'
                  width: '72'
                  height: '5'
                  fields:
                    - type: INTEGER
                      name: columns
                      value: '1'
                    - type: GRAPH_PROTOTYPE
                      name: graphid.0
                      value:
                        host: 'Linux by Zabbix agent'
                        name: 'FS [{#FSTYPE}({#FSNAME})]: Space usage graph, in % (relative to max availab
                    - type: STRING
                      name: reference
                      value: AAAEC
        (...)
```

Element	Type	Opis
type	string	(wymagane) Typ widżetu.
name	string	Nazwa widżetu.
x	integer	Pozycja pozioma od lewej strony pulpitu szablonu. Możliwe wartości:[1] 0-71.
y	integer	Pozycja pionowa od górnej krawędzi pulpitu szablonu. Możliwe wartości:[1] 0-63.
width	integer	Szerokość widżetu. Możliwe wartości:[1] 1-72.
height	integer	Wysokość widżetu. Możliwe wartości:[1] 1-64.
hide_header	string	Ukryj nagłówek widżetu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
fields		Element główny dla pól widżetu pulpitu szablonu.
type	string	(wymagane) Typ pola widżetu. Możliwe wartości:[1] INTEGER (0), STRING (1), ITEM (4), ITEM_PROTOTYPE (5), GRAPH (6), GRAPH_PROTOTYPE (7), MAP (8), SERVICE (9), SLA (10), USER (11), ACTION (12), MEDIA_TYPE (13).
name	string	(wymagane) Nazwa pola widżetu.

Element	Type	Opis
value	mixed	(wymagane) Wartość pola widżetu, zależna od typu pola.

Note:
Zobacz także: Template dashboard widget object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Mapowania wartości szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      valuemaps:
        - uuid: 8c048c6cca8248f2860c208e8db0f59e
          name: ifOperStatus
          mappings:
            - value: '0'
              newvalue: unknown
            - value: '1'
              newvalue: notpresent
            - value: '2'
              newvalue: down
            - value: '3'
              newvalue: lowerlayerdown
            - value: '4'
              newvalue: testing
            - value: '5'
              newvalue: dormant
            - value: '6'
              newvalue: up
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego mapowania wartości.
name	string	(wymagane) Nazwa mapowania wartości.
mapping		Element główny dla mapowań.
type	string	Typ dopasowania mapowania. Możliwe wartości:[1] EQUAL (0, domyślnie), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
value	string	Wartość oryginalna.
newvalue	string	(wymagane) Wartość, na którą mapowana jest wartość oryginalna.

Note:
Zobacz także: Value map object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Przypisy [1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie w celach informacyjnych. Więcej informacji można znaleźć na stronie powiązanego obiektu API wskazanej w odpowiednim wpisie tabeli lub na końcu każdej sekcji.

4 Hosty

Przegląd

Hosty są eksportowane z wieloma powiązanymi obiektami i relacjami obiektów.

Eksport hosta zawiera:

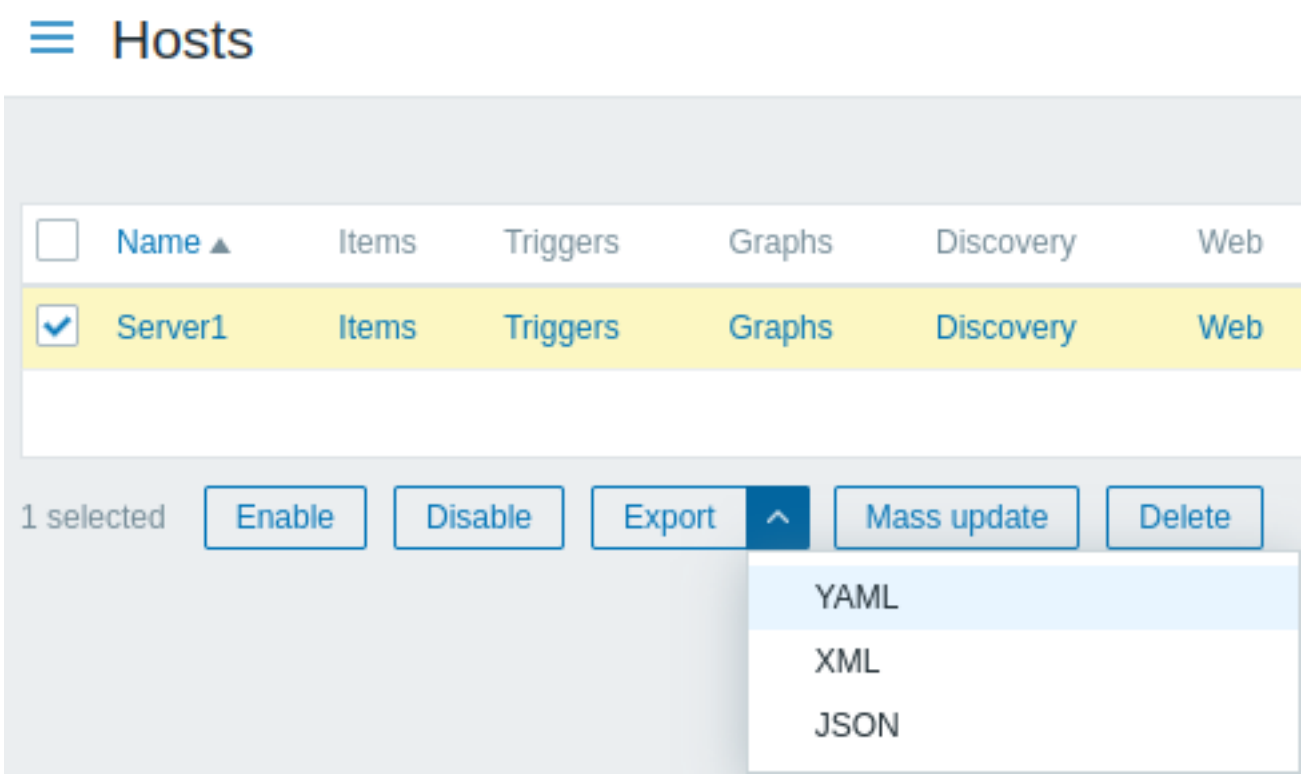
- Połączone grupy hostów
- Dane hosta

- Powiązanie szablonu
- Powiązanie grupy hosta
- Interfejsy hosta
- Bezpośrednio połączone pozycje
- Bezpośrednio połączone wyzwalacze
- Bezpośrednio połączone reguły wykrywania ze wszystkimi prototypami
- Bezpośrednio połączone scenariusze sieci WWW
- Makra hosta
- Dane inwentarza hosta
- Mapy wartości
- Połączone wykresy

Eksportowanie

Aby wyeksportować hosty, wykonaj następujące kroki:

1. Przejdź do *Data collection* → *Hosts*.
2. Zaznacz pola wyboru hostów, które chcesz wyeksportować.
3. Kliknij *Export* pod listą.



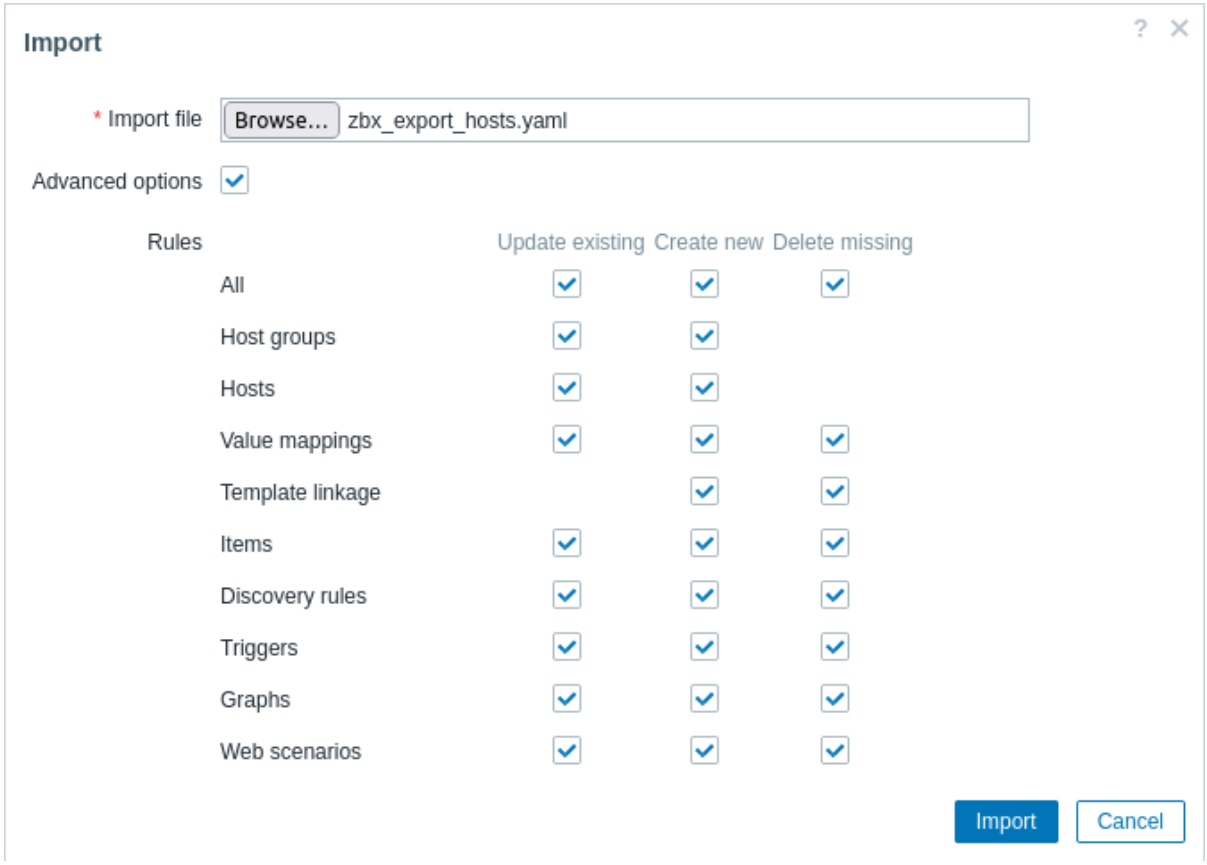
W zależności od wybranego formatu hosty są eksportowane do pliku lokalnego z domyślną nazwą:

- `zabbix_export_hosts.yaml` — w eksporcie YAML (domyślna opcja eksportu);
- `zabbix_export_hosts.xml` — w eksporcie XML;
- `zabbix_export_hosts.json` — w eksporcie JSON.

Importowanie

Aby zaimportować hosty, wykonaj następujące czynności:

1. Przejdź do *Data collection* → *Hosts*.
2. Kliknij *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Kliknij *Import* w prawym dolnym rogu formularza konfiguracji.



Jeśli zaznaczysz pole wyboru *Advanced options*, zostanie wyświetlona szczegółowa lista wszystkich elementów możliwych do importu — zaznacz lub odznacz każdą regułę importu zgodnie z potrzebami.

Jeśli klikniesz pole wyboru w wierszu *All*, wszystkie elementy poniżej zostaną zaznaczone/odznaczone.

Reguły importu:

Reguła	Opis
Update existing	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.
Delete missing	Istniejące elementy, które nie są obecne w pliku importu, zostaną usunięte. W przeciwnym razie nie zostaną usunięte. Jeśli opcja *Delete missing* jest zaznaczona dla *Template linkage*, bieżące powiązania z szablonami, których nie ma w pliku importu, zostaną odłączone. Encje (pozycje, wyzwalacze, wykresy itp.) dziedziczone z odłączonych szablonów nie zostaną usunięte (chyba że opcja *Delete missing* zostanie również wybrana dla każdej encji).

Komunikat o powodzeniu lub niepowodzeniu importu zostanie wyświetlony we frontend.

Format eksportu

Gdy host jest eksportowany, Zabbix generuje uporządkowany format YAML, JSON lub XML. Eksport obejmuje elementy hosta, takie jak metadane hosta, pozycje, makra, wyzwalacze i inne.

Każdy element pełni określoną funkcję i może zawierać elementy zagnieżdżone.

W kolejnych sekcjach opisano każdy element w formacie eksportu. W przykładach użyto hosta z szablonem MySQL by Zabbix agent 2. Na potrzeby demonstracji elementów szablon jest odłączany po utworzeniu hosta. Odłączenie zachowuje wszystkie elementy w konfiguracji (w przeciwieństwie do odłączenia i wyczyszczenia, które je usuwa).

Wielokropek (`...`) oznacza elementy pominięte dla zwięzłości. Uwaga (`see table below`) jest używana, gdy dany element jest bardziej szczegółowo wyjaśniony w dalszych sekcjach.

```
zabbix_export:
  version: '7.4'
```

```
  host_groups:
    - uuid: 748ad4d098d447d492bb935c907f652f
      name: Databases
  hosts: (patrz tabela poniżej)
  graphs: (patrz tabela poniżej)
```

Metadane hosta

Element		Type	Description
version		string	(wymagane) Wersja Zabbix.
host_groups			(wymagane) Element główny dla grup hostów.
	uuid	string	(wymagane) Unikalny identyfikator tej grupy hostów.
	name	string	(wymagane) Nazwa grupy hostów.
hosts			Element główny dla hostów.
graphs			Element główny dla wykresów hosta.

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      name: 'MySQL server'
      groups:
        - name: Databases
      interfaces: (see table below)
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: location
          value: Riga
      macros:
        (...)
        - macro: '{$MYSQL.DSN}'
          value: 192.0.2.1
        (...)
        - macro: '{$MYSQL.PASSWORD}'
          type: SECRET_TEXT
          description: 'Hasło użytkownika MySQL.'
        (...)
      valuemaps: (see table below)
  (...)
```

Hosty

Element	Type	Description
host	string	(wymagane) Unikalna nazwa hosta.
name	string	Widoczna nazwa hosta.
description	text	Opis hosta.
monitored_by	string	Sposób monitorowania hosta. Dostępne wartości:[1] SERVER (0, domyślnie), PROXY (1) lub PROXY_GROUP (2).
proxy		Element główny dla proxy.
name	string	(wymagane) Nazwa proxy (jeśli dotyczy), które monitoruje host.
proxy_group		Element główny dla grupy proxy.
name	string	(wymagane) Nazwa grupy proxy (jeśli dotyczy), używanej do monitorowania hosta.
status	string	Status hosta. Dostępne wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
ipmi_authtype	string	Typ uwierzytelniania sesji IPMI. Dostępne wartości:[1] DEFAULT (-1, domyślnie), NONE (0), MD2 (1), MD5 (2), STRAIGHT (4), OEM (5), RMCP_PLUS (6).

Element	Type	Description
ipmi_privilege	string	Poziom uprawnień sesji IPMI. Dostępne wartości:[1] CALLBACK (1), USER (2, domyślnie), OPERATOR (3), ADMIN (4), OEM (5).
ipmi_username	string	Nazwa użytkownika do sprawdzeń IPMI.
ipmi_password	string	Hasło do sprawdzeń IPMI.
templates		Element główny dla powiązanych szablonów.
name	string	(wymagane) Nazwa szablonu.
groups		Element główny dla grup hostów, do których należy host.
name	string	(wymagane) Nazwa grupy hostów.
interfaces		Element główny dla interfejsów hosta.
items		Element główny dla pozycji hosta.
discovery_rules		Element główny dla reguł niskopoziomowego wykrywania hosta.
http_tests		Element główny dla scenariuszy web hosta.
tags		Element główny dla tagów hosta.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
macros		Element główny dla makr hosta.
macro	string	(wymagane) Nazwa makra użytkownika.
type	string	Typ makra użytkownika. Dostępne wartości:[1] TEXT (0, domyślnie), SECRET_TEXT (1), VAULT (2).
value	string	Wartość makra użytkownika.
description	string	Opis makra użytkownika.
inventory		Element główny dla inwentarza hosta.
<inventory_property>	string	Właściwość inwentarza. Wszystkie właściwości mają odpowiadający im element (`type`, `name`, `os` itd.; na przykład zobacz Format eksportu).
inventory_mode	string	Tryb inwentarza. Dostępne wartości:[1] DISABLED (-1), MANUAL (0, domyślnie), AUTOMATIC (1).
valuemaps		Element główny dla map wartości hosta.

Note:
Zobacz także: Obiekt Host (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Interfejsy hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      interfaces:
        - ip: 192.0.2.1
          interface_ref: if1
      (...)
```

Element	Type	Opis
default	string	Czy jest to główny interfejs hosta. Należy pamiętać, że na hoście może istnieć tylko jeden główny interfejs danego typu. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
type	string	Typ interfejsu. Możliwe wartości:[1] ZABBIX (1, domyślnie), SNMP (2), IPMI (3), JMX (4).
useip	string	Czy do łączenia z hostem ma być używany adres IP (w przeciwnym razie zostanie użyty DNS). Możliwe wartości:[1] NO (0), YES (1, domyślnie).
ip	string	(wymagane dla połączeń IP) Adres IP (IPv4 lub IPv6).
dns	string	(wymagane dla połączeń DNS) Nazwa DNS.
port	string	Numer portu.
details		Element główny dla szczegółów interfejsu.
version	string	Użyj tej wersji SNMP. Możliwe wartości:[1] SNMPV1 (1), SNMP_V2C (2, domyślnie), SNMP_V3 (3).
community	string	(wymagane dla pozycji SNMPv1 i SNMPv2) Społeczność SNMP.

Element	Type	Opis
max_repetitions	string	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Obsługiwane dla pozycji SNMPv2 i SNMPv3 (`discovery[]` i `walk[]`). Domyślnie: 10.
contextname	string	Nazwa kontekstu SNMPv3. Obsługiwane dla pozycji SNMPv3.
securityname	string	Nazwa zabezpieczeń SNMPv3. Obsługiwane dla pozycji SNMPv3.
securitylevel	string	Poziom zabezpieczeń SNMPv3. Obsługiwane dla pozycji SNMPv3. Możliwe wartości:[1] NOAUTHNOPRIV (0, domyślnie), AUTHNOPRIV (1), AUTHPRIV (2).
authprotocol	string	Protokół uwierzytelniania SNMPv3. Obsługiwane dla pozycji SNMPv3. Możliwe wartości:[1] MD5 (0, domyślnie), SHA1 (1), SHA224 (2), SHA256 (3), SHA384 (4), SHA512 (5).
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Obsługiwane dla pozycji SNMPv3.
privprotocol	string	Protokół prywatności SNMPv3. Obsługiwane dla pozycji SNMPv3. Możliwe wartości:[1] DES (0, domyślnie), AES128 (1), AES192 (2), AES256 (3), AES192C (4), AES256C (5).
privpassphrase	string	Hasło prywatności SNMPv3. Obsługiwane dla pozycji SNMPv3.
bulk	string	Używaj żądań zbiorczych dla SNMP. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
interface_ref	string	Nazwa referencyjna interfejsu, która ma być używana w pozycjach (format: `if<N>`).

Note:
Zobacz także: Obiekt interfejsu hosta (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Pozycje hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Binlog cache disk use'
          type: DEPENDENT
          key: mysql.binlog_cache_disk_use
          value_type: FLOAT
          description: 'Liczba transakcji, które użyły tymczasowej pamięci podręcznej na dysku, ponieważ n
          preprocessing: (see table below)
          master_item:
            key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          tags:
            - tag: component
              value: cache
        (...)
        - name: 'Buffer pool utilization'
          type: CALCULATED
          key: mysql.buffer_pool_utilization
          value_type: FLOAT
          units: '%'
          params: |
            ( last(//mysql.innodb_buffer_pool_pages_total) -
            last(//mysql.innodb_buffer_pool_pages_free) ) /
            ( last(//mysql.innodb_buffer_pool_pages_total) +
            ( last(//mysql.innodb_buffer_pool_pages_total) = 0 ) ) * 100 *
```

```
        ( last(//mysql.innodb_buffer_pool_pages_total) > 0 )
      description: 'Stosunek użytych stron do wszystkich stron w puli buforów.'
      tags:
        - tag: component
          value: memory
      triggers: (see table below)
    (...)
    - name: Uptime
      type: DEPENDENT
      key: mysql.uptime
      units: uptime
      description: 'Liczba sekund, przez które serwer był uruchomiony.'
      preprocessing: (see table below)
      master_item:
        key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
      tags:
        - tag: component
          value: application
      triggers: (see table below)
    (...)
```

Element	Type	Description
name	string	(wymagane) Nazwa pozycji.
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), CALCULATED (15), JMX (16), SNMP_TRAP (17), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22), ITEM_TYPE_NESTED (23).
snmp_oid	string	(wymagane dla pozycji SNMP_AGENT) Identyfikator obiektu SNMP.
key	string	(wymagane) Klucz pozycji.
delay	string	Interwał aktualizacji pozycji. Domyślnie: 1m. Wartość będzie zawsze równa 0 dla pozycji TRAP.
history	string	Okres czasu (z użyciem sufiksu czasu, makra użytkownika lub makra LLD), przez jaki dane historyczne mają być przechowywane. Domyślnie: 31d.
trends	string	Okres czasu (z użyciem sufiksu czasu, makra użytkownika lub makra LLD), przez jaki dane trendów mają być przechowywane. Domyślnie: 365d.
status	string	Status pozycji. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
value_type	string	Typ odebranej wartości. Możliwe wartości:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, domyślnie), TEXT (4), BINARY (5).
allowed_hosts	string	Lista adresów IP hostów, oddzielonych przecinkami, które mogą wysyłać dane dla pozycji. Obsługiwane dla pozycji TRAP i HTTP_AGENT.
units	string	Jednostki odebranej wartości (bps, B, itd.).
params	text	Dodatkowe parametry zależne od typu pozycji (wykonywany skrypt dla pozycji SSH i TELNET; zapytanie SQL dla pozycji ODBC; formuła dla pozycji CALCULATED; skrypt dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER).
ipmi_sensor	string	Czujnik IPMI. Obsługiwane dla pozycji IPMI.
authtype	string	Typ uwierzytelniania. Obsługiwane dla pozycji SSH i HTTP_AGENT. Możliwe wartości dla pozycji SSH:[1] PASSWORD (0, domyślnie), PUBLIC_KEY (1). Możliwe wartości dla pozycji HTTP_AGENT:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
username	string	(wymagane dla pozycji SSH i TELNET) Nazwa użytkownika do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `password` (patrz poniżej) albo pozostawić oba elementy puste.

Element	Type	Description
password	string	(wymagane dla pozycji SSH i TELNET) Hasło do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `username` (patrz powyżej) albo pozostawić oba elementy puste.
publickey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza publicznego.
privatekey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza prywatnego.
description	text	Opis pozycji.
inventory_link	string	Pole inwentarza hosta, które jest wypełniane przez pozycję. Możliwe wartości:[1] NONE (0), ALIAS (4) itd. (obsługiwane pola: zobacz Inwentarz hosta).
valuemap		Element główny map wartości pozycji.
name	string	(wymagane) Nazwa mapy wartości używanej dla pozycji.
logtimefmt	string	Format czasu w wpisach dziennika. Obsługiwane dla pozycji typu LOG.
preprocessing		Element główny wstępnego przetwarzania wartości pozycji.
step		Element główny kroków wstępnego przetwarzania wartości pozycji hosta.
interface_ref	string	Odwołanie do interfejsu hosta (format: `if<N>`).
jmx_endpoint	string	Punkt końcowy JMX. Obsługiwane dla pozycji JMX.
master_item		(wymagane dla pozycji DEPENDENT) Element główny dla pozycji nadrzędnej pozycji zależnej.
key	string	(wymagane) Klucz pozycji nadrzędnej pozycji zależnej.
timeout	string	Limit czasu żądania odpytywania danych pozycji. Obsługiwane dla listy typów pozycji Timeouts.
url	string	(wymagane dla pozycji HTTP_AGENT) Ciąg URL.
query_fields		Element główny parametrów zapytania. Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania. Obsługiwane dla pozycji HTTP_AGENT.
parameters		Element główny parametrów zdefiniowanych przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
name	string	(wymagane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER) Nazwa parametru zdefiniowanego przez użytkownika.
value	string	Wartość parametru zdefiniowanego przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
posts	string	Dane treści żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługiwane dla pozycji HTTP_AGENT.
follow_redirects	string	Podążaj za przekierowaniami odpowiedzi podczas odpytywania danych. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
post_type	string	Typ treści danych POST. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (2), XML (3).
http_proxy	string	Ciąg połączenia proxy HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
headers		Element główny nagłówków żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa nagłówka.
value	string	(wymagane dla pozycji HTTP_AGENT) Wartość nagłówka.
retrieve_mode	string	Która część odpowiedzi ma zostać zapisana. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
request_method	string	Typ metody żądania. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] GET (0, domyślnie), POST (1), PUT (2), HEAD (3).
output_format	string	Sposób przetwarzania odpowiedzi. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (1).

Element	Type	Description
allow_traps	string	Zezwól na wypełnianie wartości podobnie jak w pozycji trapper. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Ścieżka do publicznego pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_file	string	Ścieżka do prywatnego pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
verify_peer	string	Czy sprawdzać, czy certyfikat hosta jest autentyczny. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Czy sprawdzać, czy nazwa hosta dla połączenia odpowiada nazwie w certyfikacie hosta. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
tags		Element główny tagów pozycji.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
triggers		Element główny wyzwalaczy pozycji hosta.

Note:
Zobacz też: Obiekt pozycji (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Kroki wstępnego przetwarzania wartości pozycji hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Binlog cache disk use'
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.Binlog_cache_disk_use
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 6h
          (...)
```

Element	Type	Description
type	string	(wymagane) Typ kroku wstępnego przetwarzania wartości pozycji. Możliwe wartości:[1] MULTIPLIER (1), RTRIM (2), LTRIM (3), TRIM (4), REGEX (5), BOOL_TO_DECIMAL (6), OCTAL_TO_DECIMAL (7), HEX_TO_DECIMAL (8), SIMPLE_CHANGE (9, obliczane: otrzymana wartość - poprzednia wartość), CHANGE_PER_SECOND (10, obliczane: (otrzymana wartość - poprzednia wartość)/(bieżący czas - czas ostatniego sprawdzenia)), XMLPATH (11), JSONPATH (12), IN_RANGE (13), MATCHES_REGEX (14), NOT_MATCHES_REGEX (15), CHECK_JSON_ERROR (16), CHECK_XML_ERROR (17), CHECK_REGEX_ERROR (18), DISCARD_UNCHANGED (19), DISCARD_UNCHANGED_HEARTBEAT (20), JAVASCRIPT (21), PROMETHEUS_PATTERN (22), PROMETHEUS_TO_JSON (23), CSV_TO_JSON (24), STR_REPLACE (25), CHECK_NOT_SUPPORTED (26), XML_TO_JSON (27), SNMP_WALK_VALUE (28), SNMP_WALK_TO_JSON (29), SNMP_GET_VALUE (30).
parameters		(wymagane) Element główny dla parametrów kroku wstępnego przetwarzania wartości pozycji.
parameter	string	Indywidualny parametr kroku wstępnego przetwarzania wartości pozycji.

Element	Type	Description
error_handler	string	Typ akcji używany w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości:[1] ORIGINAL_ERROR (0, domyślnie), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parametry obsługi błędów.

Note:
Zobacz także: Item preprocessing object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Wyzwalacze pozycji hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: Uptime
          (...)
          triggers:
            - expression: 'nodata(/MySQL server/mysql.uptime,30m)=1'
              name: 'MySQL: Failed to fetch info data'
              event_name: 'MySQL: Failed to fetch info data (or no data for 30m)'
              priority: INFO
              description: 'Zabbix has not received any data for items for the last 30 minutes.'
              dependencies:
                - name: 'MySQL: Service is down'
                  expression: 'last(/MySQL server/mysql.ping["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSW
              tags:
                - tag: scope
                  value: availability
            - expression: 'last(/MySQL server/mysql.uptime)<10m'
              name: 'MySQL: Service has been restarted'
              event_name: 'MySQL: Service has been restarted (uptime < 10m)'
              priority: INFO
              description: 'MySQL uptime is less than 10 minutes.'
              tags:
                - tag: scope
                  value: notice
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego wyzwalacza.
expression	string	(wymagane) Wyrażenie wyzwalacza.
recovery_mode	string	Podstawa generowania zdarzeń OK. Możliwe wartości:[1] EXPRESSION (0, domyślnie), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza.
correlation_mode	string	Tryb korelacji (brak korelacji zdarzeń lub korelacja zdarzeń według tagu). Możliwe wartości:[1] DISABLED (0, domyślnie), TAG_VALUE (1).
correlation_tag	string	Nazwa tagu używana do korelacji zdarzeń.
name	string	(wymagane) Nazwa wyzwalacza.
event_name	string	Nazwa zdarzenia.
opdata	string	Dane operacyjne.
url_name	string	Etykieta adresu URL powiązanego z wyzwalaczem.
url	string	Adres URL powiązany z wyzwalaczem.
status	string	Status wyzwalacza. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
priority	string	Poziom ważności wyzwalacza. Możliwe wartości:[1] NOT_CLASSIFIED (0, domyślnie), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).

Element	Type	Opis
description	text	Opis wyzwalacza.
type	string	Typ generowania zdarzeń (pojedyncze zdarzenie problemu lub wiele zdarzeń problemu). Możliwe wartości:[1] SINGLE (0, domyślnie), MULTIPLE (1).
manual_close	string	Ręczne zamykanie zdarzeń problemu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
dependencies		Element główny dla zależności.
name	string	(wymagane) Nazwa wyzwalacza zależności.
expression	string	(wymagane) Wyrażenie wyzwalacza zależności.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza zależności.
tags		Element główny dla tagów wyzwalacza.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz także: Trigger object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Reguły wykrywania niskiego poziomu hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Database discovery'
          key: 'mysql.db.discovery["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          delay: 1h
          filter: (see table below)
          description: 'Scanning databases in DBMS.'
          interface_ref: if1
          item_prototypes:
            - name: 'Size of database {#DATABASE}'
              key: 'mysql.db.size["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}","{#DATABASE}"]'
              delay: 5m
              units: B
              description: 'Database size.'
              preprocessing:
                - type: DISCARD_UNCHANGED_HEARTBEAT
                  parameters:
                    - 1h
              interface_ref: if1
              tags:
                - tag: component
                  value: storage
                - tag: database
                  value: '{#DATABASE}'
          lld_macro_paths:
            - lld_macro: '{#DATABASE}'
              path: $.Database
          preprocessing:
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 1d
        (...)
```

Attention:
Większość elementów reguł wykrywania niskiego poziomu hosta jest taka sama jak w przypadku pozycji hosta. Poniższa tabela opisuje te elementy, które różnią się od pozycji hosta.

Element	Type	Opis
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), TRAP (2), SIMPLE (3), INTERNAL (5), ZABBIX_ACTIVE (7), EXTERNAL (10), ODBC (11), IPMI (12), SSH (13), TELNET (14), JMX (16), DEPENDENT (18), HTTP_AGENT (19), SNMP_AGENT (20), ITEM_TYPE_SCRIPT (21), ITEM_TYPE_BROWSER (22).
key	string	(wymagane) Klucz reguły wykrywania niskiego poziomu.
filter		Element główny dla filtrów reguł wykrywania niskiego poziomu hosta.
lifetime	string	Okres czasu (z użyciem sekund, sufiksu czasu lub makra użytkownika), po którym zasoby, które nie są już wykrywane, zostaną usunięte. Domyślnie: 7d.
lifetime_type	string	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Okres czasu (z użyciem sekund, sufiksu czasu lub makra użytkownika), po którym zasoby, które nie są już wykrywane, zostaną wyłączone.
enabled_lifetime_type	string	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Element główny dla elementów prototypów pozycji hosta, które są takie same jak dla pozycji hosta.
trigger_prototypes		Element główny dla elementów prototypów wyzwalaczy hosta, które są takie same jak dla wyzwalaczy pozycji hosta.
graph_prototypes		Element główny dla prototypów wykresów hosta, które są takie same jak dla wykresów hosta.
host_prototypes		Element główny dla prototypów hosta, które są takie same jak dla hostów.
parent_discovery_rule		Element główny dla nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły) prototypu reguły wykrywania niskiego poziomu. Ta właściwość oznacza, że jest to prototyp reguły LLD, będący bezpośrednim elementem podrzędnym wskazanej reguły/prototypu reguły.
key	string	(wymagane) Klucz nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły).
master_item	string	(wymagane dla reguł DEPENDENT) Element główny dla pozycji nadrzędnej reguły zależnej.
lld_macro_paths		Element główny dla ścieżek makr reguły wykrywania niskiego poziomu.
lld_macro	string	(wymagane) Nazwa makra reguły wykrywania niskiego poziomu.
path	string	(wymagane) Selektor wartości, która zostanie przypisana do odpowiedniego makra.
preprocessing		Element główny dla wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu.
step		Element główny dla elementów kroków wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu, które są takie same jak dla kroków wstępnego przetwarzania wartości pozycji hosta, z wyjątkiem mniejszej liczby możliwych wartości. Zobacz także: obiekt wstępnego przetwarzania reguły LLD.
overrides		Element główny dla reguł nadpisywania reguły wykrywania niskiego poziomu.
name	string	(wymagane) Unikalna nazwa nadpisania.
step	string	(wymagane) Unikalny numer kolejności nadpisania.
stop	string	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli pasuje.
filter		Element główny dla elementów filtra reguły nadpisywania reguły wykrywania niskiego poziomu, które są takie same jak dla filtrów reguł wykrywania niskiego poziomu hosta.
operations		Element główny dla operacji nadpisywania reguły wykrywania niskiego poziomu hosta.

Note:
Zobacz także: obiekt reguły LLD (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Filtry reguły wykrywania niskiego poziomu hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Database discovery'
          (...)
```

```
        filter:
          evaltype: AND
          conditions:
            - macro: '{#DATABASE}'
              value: '{$MYSQL.DBNAME.MATCHES}'
            - macro: '{#DATABASE}'
              value: '{$MYSQL.DBNAME.NOT_MATCHES}'
              operator: NOT_MATCHES_REGEX
        (...)
```

Element	Type	Opis
evaltype	string	Nadpisuje metodę oceny warunków filtra. Możliwe wartości:[1] AND_OR (0, domyślnie), AND (1), OR (2), FORMULA (3).
formula	string	Niestandardowa formuła obliczeniowa dla warunków filtra.
conditions		Element główny dla warunków filtra.
macro	string	(wymagane) Nazwa makra wykrywania niskiego poziomu, dla którego ma zostać wykonane sprawdzenie.
value	string	Wartość do porównania.
operator	string	Operator warunku. Możliwe wartości:[1] MATCHES_REGEX (8, domyślnie), NOT_MATCHES_REGEX (9).
formulaid	string	(wymagane) Dowolny unikalny identyfikator używany do odwoływania się do warunku z poziomu wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu ich pobrania.

Note:
Zobacz także: obiekt filtra reguły LLD (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Operacje nadpisania reguły wykrywania niskopoziomowego hosta

Element	Type	Opis
operationobject	string	Obiekt, do którego stosowana jest operacja. Możliwe wartości:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3), DISCOVERY_PROTOTYPE (4).
operator	string	Operator warunku nadpisania. Możliwe wartości:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Wyrażenie regularne lub ciąg znaków dla operatora warunku nadpisania.
status	string	Status obiektu po operacji nadpisania.
discover	string	Czy obiekt jest dodawany w wyniku wykrywania.
delay	string	Interwał aktualizacji ustawiony dla prototypu pozycji po operacji nadpisania.
history	string	Okres przechowywania historii ustawiony dla prototypu pozycji po operacji nadpisania.
trends	string	Okres przechowywania trendów ustawiony dla prototypu pozycji po operacji nadpisania.
severity	string	Istotność prototypu wyzwalacza ustawiona po operacji nadpisania.
tags		Element główny dla tagów ustawionych dla obiektu po operacji nadpisania.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
templates		Element główny dla szablonów podłączonych do prototypu hosta po operacji nadpisania.
name	string	(wymagane) Nazwa szablonu.
inventory_mode	string	Tryb inwentaryzacji prototypu hosta ustawiony po operacji nadpisania.

Note:
Zobacz także: Obiekt operacji nadpisania reguły LLD (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Scenariusze WWW hosta

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego scenariusza WWW.
name	string	(wymagane) Nazwa scenariusza WWW.
delay	string	Częstotliwość wykonywania scenariusza WWW (z użyciem sekund, sufiksu czasu lub makra użytkownika). Domyślnie: 1m.
attempts	integer	Liczba prób wykonania kroków scenariusza WWW. Możliwe wartości: 1-10 (domyślnie: 1).
agent	string	Agent klienta. Zabbix będzie udawał wybraną przeglądarkę. Jest to przydatne, gdy witryna zwraca różną zawartość dla różnych przeglądarek. Domyślnie: Zabbix.
http_proxy	string	Proxy, który będzie używany przez scenariusz WWW, podany w postaci: `http://[username[:password]@]proxy.example.com[:port]`
variables		Element główny dla zmiennych scenariusza WWW, które mogą być używane w krokach scenariusza.
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny dla nagłówków HTTP wysyłanych podczas wykonywania żądania. Nagłówki powinny być podane z użyciem tej samej składni, w jakiej występują w protokole HTTP.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
status	string	Status scenariusza WWW. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
authentication	string	Metoda uwierzytelniania. Możliwe wartości:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
http_user	string	Nazwa użytkownika używana do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
http_password	string	Hasło używane do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
verify_peer	string	Weryfikacja certyfikatu SSL serwera WWW. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Weryfikacja zgodności pola *Common Name* lub pola *Subject Alternate Name* certyfikatu serwera WWW. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło pliku klucza prywatnego SSL.
steps		(wymagane) Element główny dla kroków scenariusza WWW hosta.
tags		Element główny dla tagów scenariusza WWW.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz także: Obiekt scenariusza WWW (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Kroki scenariusza WWW hosta

Element	Type	Opis
name	string	(wymagane) Nazwa kroku scenariusza WWW.
url	string	(wymagane) URL do monitorowania.
query_fields		Element główny parametrów zapytania (tablica pól HTTP, które mają zostać dodane do URL podczas wykonywania żądania).
name	string	(wymagane) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania.
posts		Element główny zmiennych HTTP POST (ciąg znaków (surowe dane POST) lub tablica pól HTTP (dane pól formularza)).
name	string	(wymagane) Nazwa pola POST.

Element	Type	Opis
value	string	(wymagane) Wartość pola POST.
variables		Element główny zmiennych na poziomie kroku (makr), które powinny zostać zastosowane po tym kroku. Jeśli wartość zmiennej ma prefiks 'regex:', to jej wartość jest wyodrębniana z danych zwróconych przez ten krok zgodnie ze wzorcem wyrażenia regularnego następującym po prefiksie 'regex:'
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny nagłówków HTTP, które mają zostać wysłane podczas wykonywania żądania.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
follow_redirects	string	Podążanie za przekierowaniami HTTP. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
retrieve_mode	string	Tryb pobierania odpowiedzi HTTP. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
timeout	string	Limit czasu wykonania kroku (z użyciem sekund, sufiksu czasu lub makra użytkownika). Domyślnie: 15s.
required	string	Tekst, który musi występować w odpowiedzi (ignorowany, jeśli pusty).
status_codes	string	Lista akceptowanych kodów statusu HTTP rozdzielonych przecinkami (np. 200-201,210-299; ignorowana, jeśli pusta).

Note:
Zobacz także: Obiekt kroku scenariusza WWW (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Wykresy hosta

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items: (zobacz tabelę poniżej)
    (...)
```

Element	Type	Opis
uuid	string	Unikalny identyfikator wykresu.
name	string	(wymagane) Nazwa wykresu.
width	integer	Szerokość wykresu w pikselach. Używana do podglądu oraz dla wykresów kołowych/rozsuniętych. Możliwe wartości:[1] 20-65535 (domyślnie: 900).
height	integer	Wysokość wykresu w pikselach. Używana do podglądu oraz dla wykresów kołowych/rozsuniętych. Możliwe wartości:[1] 20-65535 (domyślnie: 900).
yaxismin	double	Wartość minimum osi Y. Obsługiwane dla STAŁEJ wartości minimalnej osi Y. Domyślnie: 0.
yaxismax	double	Wartość maksimum osi Y. Obsługiwane dla STAŁEJ wartości minimalnej osi X. Domyślnie: 0.
show_work_period	string	Wyróżnij godziny poza czasem pracy. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
show_triggers	string	Wyświetl proste wartości wyzwalaczy jako linię. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
type	string	Typ wykresu. Możliwe wartości:[1] NORMAL (0, domyślnie), STACKED (1), PIE (2), EXPLODED (3).
show_legend	string	Wyświetl legendę wykresu. Możliwe wartości:[1] NO (0), YES (1, domyślnie).

Element	Type	Opis
show_3d	string	Włącz styl 3D. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
percent_left	double	Pokaż linię percentyla dla lewej osi. Obsługiwane dla wykresów NORMAL. Domyślnie: 0.
percent_right	double	Pokaż linię percentyla dla prawej osi. Obsługiwane dla wykresów NORMAL. Domyślnie: 0.
ymin_type_1	string	Minimalna wartość osi Y. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] CALCULATED (0, domyślnie), FIXED (1), ITEM (2).
ymin_item_1		(wymagane, jeśli `ymin_type_1` jest ustawione na `ITEM`) Element główny dla szczegółów indywidualnej pozycji.
host	string	(wymagane) Host pozycji.
key	string	(wymagane) Klucz pozycji.
ymax_type_1	string	Maksymalna wartość osi Y. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] CALCULATED (0, domyślnie), FIXED (1), ITEM (2).
ymax_item_1		(wymagane, jeśli `ymax_type_1` jest ustawione na `ITEM`) Element główny dla szczegółów indywidualnej pozycji.
host	string	(wymagane) Host pozycji.
key	string	(wymagane) Klucz pozycji.
graph_items		(wymagane) Element główny dla pozycji wykresu hosta.

Note:
Zobacz także: Graph object (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Elementy wykresu hosta

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items:
        - drawtype: GRADIENT_LINE
          color: 199C0D
          item:
            host: 'MySQL server'
            key: mysql.bytes_received.rate
        - sortorder: '1'
          drawtype: GRADIENT_LINE
          color: F63100
          item:
            host: 'MySQL server'
            key: mysql.bytes_sent.rate
    (...)
```

Element	Type	Opis
sortorder	integer	Kolejność rysowania. Mniejsza wartość jest rysowana jako pierwsza. Można jej użyć do rysowania linii lub obszarów za (lub przed) innymi.
drawtype	string	Styl rysowania elementu wykresu. Obsługiwane dla wykresów NORMAL. Możliwe wartości:[1] SINGLE_LINE (0, domyślnie), FILLED_REGION (1), BOLD_LINE (2), DOTTED_LINE (3), DASHED_LINE (4), GRADIENT_LINE (5).
color	string	Kolor elementu (6 znaków, zapis szesnastkowy).
yaxisside	string	Strona wykresu, po której będzie rysowana skala Y elementu wykresu. Obsługiwane dla wykresów NORMAL i STACKED.

Element		Type	Opis
calc_fnc		string	Dane do narysowania, jeśli dla pozycja istnieje więcej niż jedna wartość. Możliwe wartości:[1] MIN (1), AVG (2, domyślnie), MAX (4), ALL (7; minimum, średnia i maksimum; obsługiwane dla wykresów prostych), LAST (9, obsługiwane dla wykresów kołowych/rozsadzonych).
type		string	Typ elementu wykresu. Możliwe wartości:[1] SIMPLE (0, domyślnie), GRAPH_SUM (2; wartość pozycji reprezentuje cały wykres kołowy; obsługiwane dla wykresów kołowych/rozsadzonych).
item			(wymagane) Pojedyncza pozycja.
	host	string	(wymagane) Host pozycji.
	key	string	(wymagane) Klucz pozycji.

Note:
Zobacz też: Obiekt elementu wykresu (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Mapy wartości hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      valuemaps:
        - name: Przykładowa mapa wartości
          mappings:
            - value: '1'
              newvalue: Przykładowa wartość
  (...)
```

Element		Type	Opis
uuid		string	(wymagane) Unikalny identyfikator tej mapy wartości.
name		string	(wymagane) Nazwa mapy wartości.
mapping			Element główny dla mapowań.
	type	string	Typ dopasowania mapowania. Możliwe wartości:[1] EQUAL (0, domyślnie), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
	value	string	Wartość oryginalna.
	newvalue	string	(wymagane) Wartość, na którą mapowana jest wartość oryginalna.

Note:
Zobacz także: Value map object (odpowiednia właściwość o pasującej nazwie).

Przypisy [1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie w celach informacyjnych. Więcej informacji można znaleźć na stronie powiązanego obiektu API wskazanej w odpowiednim wpisie tabeli lub na końcu każdej sekcji.

5 Mapy sieciowe

Przegląd

Export mapy sieci zawiera:

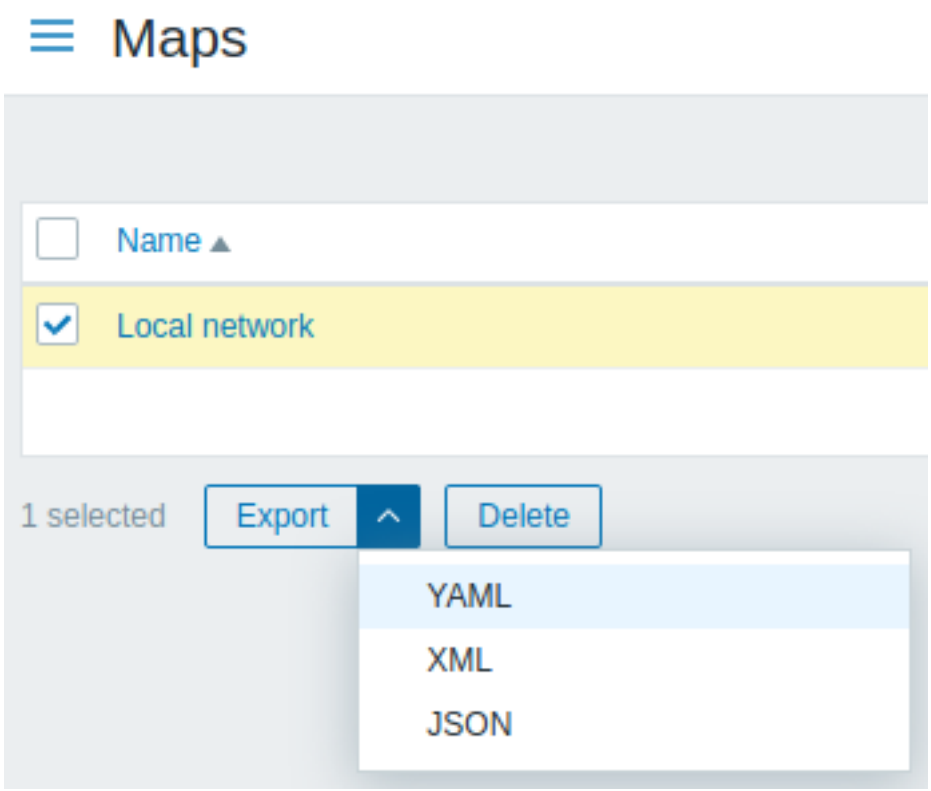
- Wszystkie powiązane obrazy
- Strukturę mapy (wszystkie ustawienia mapy, wszystkie zawarte elementy wraz z ich ustawieniami, łącza mapy oraz wskaźniki statusu łączy mapy)

Warning:
Żadne grupy hostów, hosty, wyzwalacze, inne mapy ani inne elementy, które mogą być powiązane z eksportowaną mapą, nie są eksportowane. Dlatego jeśli brakuje co najmniej jednego z elementów, do których odwołuje się mapa, jej import się nie powiedzie.

Eksportowanie

Aby wyeksportować mapy sieci, wykonaj następujące czynności:

1. Przejdź do *Monitoring* → *Maps*.
2. Zaznacz pola wyboru map sieci, które chcesz wyeksportować.
3. Kliknij *Export* pod listą.



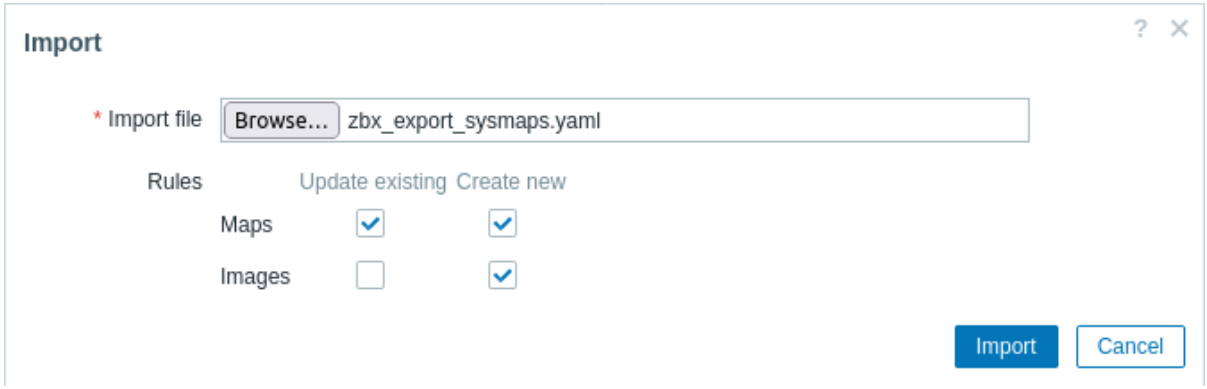
W zależności od wybranego formatu mapy są eksportowane do pliku lokalnego z domyślną nazwą:

- `zabbix_export_maps.yaml` — w eksporcie YAML (domyślna opcja eksportu);
- `zabbix_export_maps.xml` — w eksporcie XML;
- `zabbix_export_maps.json` — w eksporcie JSON.

Importowanie

Aby zaimportować mapy sieci, wykonaj następujące czynności:

1. Przejdź do *Monitoring* → *Maps*.
2. Kliknij *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz wymagane opcje w regułach importu.
5. Kliknij *Import* w prawym dolnym rogu formularza konfiguracji.



Reguły importu:

Reguła	Opis
Update existing	Istniejące mapy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe mapy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.

Jeśli odznaczysz obie opcje map i zaznaczysz odpowiednie opcje dla obrazów, zaimportowane zostaną tylko obrazy. Importowanie obrazów jest dostępne tylko dla użytkowników *Super admin*.

W frontend zostanie wyświetlony komunikat o powodzeniu lub niepowodzeniu importu.

Warning:
Jeśli zastąpisz istniejący obraz, wpłynie to na wszystkie mapy, które używają tego obrazu.

Format eksportu

Eksport do YAML:

```
zabbix_export:
  version: '7.4'
  images:
    - name: Zabbix_server_3D_(128)
      imagetype: '1'
      encodedImage: iVBOR...5CYII=
  maps:
    - name: 'Local network'
      width: '680'
      height: '200'
      label_type: '0'
      label_location: '0'
      highlight: '1'
      expandproblem: '1'
      markelements: '1'
      show_unack: '0'
      severity_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'
    background: {  }
    iconmap: {  }
    urls: {  }
    selements:
      - elementtype: '0'
        elements:
          - host: 'Zabbix server'
        label: |
          {HOST.NAME}
          {HOST.CONN}
        label_location: '0'
        x: '111'
        'y': '61'
        elementsubtype: '0'
        areatype: '0'
        width: '200'
        height: '200'
        viewtype: '0'
        use_iconmap: '0'
        selementid: '1'
        icon_off:
          name: Zabbix_server_3D_(128)
        icon_on: {  }
        icon_disabled: {  }
        icon_maintenance: {  }
        urls: {  }
        evaltype: '0'
    shapes:
      - 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: {  }
    links: {  }
```

Eksportowane elementy Wyeksportowane elementy są objaśnione w poniższej tabeli.

Element	Type	Description
images		Element główny dla obrazów.
name	string	Unikalna nazwa obrazu.
imagetype	integer	Typ obrazu. Możliwe wartości: 1 - Obraz; 2 - Tło.
encodedImage	string	Obraz zakodowany w Base64.
maps		Element główny dla map.

Mapy

Element	Type	Opis
name	string	Unikalna nazwa mapy.
width	integer	Szerokość mapy w pikselach.
height	integer	Wysokość mapy w pikselach.
label_type	integer	Typ etykiety elementu mapy. Możliwe wartości: 0 - Etykieta; 1 - Adres IP hosta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Nic.
label_location	integer	Domyślne położenie etykiety elementu mapy. Możliwe wartości: 0 - Dół; 1 - Lewo; 2 - Prawo; 3 - Góra.
highlight	integer	Włącz podświetlanie ikon dla aktywnych wyzwalaczy i statusów hostów. Możliwe wartości: 0 - Nie; 1 - Tak.
expandproblem	integer	Wyświetlaj wyzwalacz problemu dla elementów z pojedynczym problemem. Możliwe wartości: 0 - Nie; 1 - Tak.
markelements	integer	Podświetlaj elementy mapy, których status ostatnio się zmienił. Możliwe wartości: 0 - Nie; 1 - Tak.
show_unack	integer	Wyświetlanie problemów. Możliwe wartości: 0 - Liczba wszystkich problemów; 1 - Liczba niepotwierdzonych problemów; 2 - Liczba potwierdzonych i niepotwierdzonych problemów osobno.
severity_min	integer	Minimalny poziom ważności wyzwalacza wyświetlany domyślnie na mapie. Możliwe wartości: 0 - Niesklasyfikowany; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa.
show_suppressed	integer	Wyświetlaj problemy, które w przeciwnym razie byłyby wyciszone (niewyświetlane) z powodu konserwacji hosta. Możliwe wartości: 0 - Nie; 1 - Tak.
grid_size	integer	Rozmiar komórki siatki mapy w pikselach. Obsługiwane, jeśli `grid_show` jest ustawione na 0. Możliwe wartości: 20, 40, 50, 75 lub 100.
grid_show	integer	Wyświetlaj siatkę w konfiguracji mapy. Możliwe wartości: 0 - Tak; 1 - Nie.
grid_align	integer	Automatycznie wyrównuj ikony w konfiguracji mapy. Możliwe wartości: 0 - Tak; 1 - Nie.

Element	Type	Opis
label_format	integer	Użyj zaawansowanej konfiguracji etykiet. Możliwe wartości: 0 - Nie; 1 - Tak.
label_type_host	integer	Wyświetlaj etykietę jako etykietę hosta. Obsługiwane, jeśli `label_format` jest ustawione na 1. Możliwe wartości: 0 - Etykieta; 1 - Adres IP hosta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Nic; 5 - Etykieta niestandardowa.
label_type_hostgroup	integer	Wyświetlaj etykietę jako etykietę grupy hostów. Obsługiwane, jeśli `label_format` jest ustawione na 1. Możliwe wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Nic; 5 - Etykieta niestandardowa.
label_type_trigger	integer	Wyświetlaj etykietę jako etykietę wyzwalacza. Obsługiwane, jeśli `label_format` jest ustawione na 1. Możliwe wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Nic; 5 - Etykieta niestandardowa.
label_type_map	integer	Wyświetlaj etykietę jako etykietę mapy. Obsługiwane, jeśli `label_format` jest ustawione na 1. Możliwe wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Nic; 5 - Etykieta niestandardowa.
label_type_image	integer	Wyświetlaj etykietę jako etykietę obrazu. Obsługiwane, jeśli `label_format` jest ustawione na 1. Możliwe wartości: 0 - Etykieta; 2 - Nazwa elementu; 4 - Nic; 5 - Etykieta niestandardowa.
label_string_host	string	Etykieta niestandardowa dla elementów hosta. Obsługiwane, jeśli `label_type_host` jest ustawione na 5.
label_string_hostgroup	string	Etykieta niestandardowa dla elementów grupy hostów. Obsługiwane, jeśli `label_type_hostgroup` jest ustawione na 5.
label_string_trigger	string	Etykieta niestandardowa dla elementów wyzwalacza. Obsługiwane, jeśli `label_type_trigger` jest ustawione na 5.
label_string_map	string	Etykieta niestandardowa dla elementów mapy. Obsługiwane, jeśli `label_type_map` jest ustawione na 5.
label_string_image	string	Etykieta niestandardowa dla elementów obrazu. Obsługiwane, jeśli `label_type_image` jest ustawione na 5.
expand_macros	integer	Rozwijaj makra w etykietach w konfiguracji mapy. Możliwe wartości: 0 - Nie; 1 - Tak.
background		Element główny obrazu tła (jeśli istnieje). Obsługiwane, jeśli `imagetype` jest ustawione na 2.
name	string	Nazwa obrazu tła.

Element		Type	Opis
iconmap			Element główny mapowania ikon (jeśli istnieje).
	name	string	Nazwa mapowania ikon.
urls			Element główny adresów URL używanych przez mapy lub każdy element mapy.
	name	string	Nazwa linku.
	url	string	Adres URL linku.
	elementtype	integer	Typ pozycji mapy, do którego należy link. Możliwe wartości: 0 - Host; 1 - Mapa; 2 - Wyzwalacz; 3 - Grupa hostów; 4 - Obraz.
selements			Element główny dla selements mapy.
shapes			Element główny dla kształtów mapy.
	type	integer	Typ kształtu. Możliwe wartości: 0 - Prostokąt; 1 - Elipsa.
	x	integer	Współrzędne X kształtu w pikselach.
	y	integer	Współrzędne Y kształtu w pikselach.
	width	integer	Szerokość kształtu.
	height	integer	Wysokość kształtu.
	text	string	Tekst wewnątrz kształtu.
	font	integer	Styl czcionki tekstu. Możliwe wartości: 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.
	font_size	integer	Rozmiar czcionki w pikselach.
	font_color	string	Kolor czcionki przedstawiony w kodzie szesnastkowym.
	text_halign	integer	Wyrównanie poziome tekstu. Możliwe wartości: 0 - Środek; 1 - Lewo; 2 - Prawo.
	text_valign	integer	Wyrównanie pionowe tekstu. Możliwe wartości: 0 - Środek; 1 - Góra; 2 - Dół.
	border_type	integer	Typ obramowania kształtu. Możliwe wartości: 0 - Brak; 1 - Linia pogrubiona; 2 - Linia kropkowana; 3 - Linia przerywana.
	border_width	integer	Szerokość obramowania w pikselach.
	border_color	string	Kolor obramowania przedstawiony w kodzie szesnastkowym.
	background_color	string	Kolor tła (wypełnienia) przedstawiony w kodzie szesnastkowym.
	zindex	integer	Wartość określająca kolejność wszystkich kształtów i linii (z-index).
lines			Element główny dla linii mapy.

Element		Type	Opis
	x1	integer	Współrzędne X punktu 1 linii w pikselach.
	y1	integer	Współrzędne Y punktu 1 linii w pikselach.
	x2	integer	Współrzędne X punktu 2 linii w pikselach.
	y2	integer	Współrzędne Y punktu 2 linii w pikselach.
	line_type	integer	Typ linii. Możliwe wartości: 0 - Brak; 1 - Linia pogrubiona; 2 - Linia kropkowana; 3 - Linia przerywana.
	line_width	integer	Szerokość linii w pikselach.
	line_color	string	Kolor linii przedstawiony w kodzie szesnastkowym.
	zindex	integer	Wartość określająca kolejność wszystkich kształtów i linii (z-index).
links			Element główny dla połączeń między elementami mapy.
	drawtype	integer	Styl połączenia. Możliwe wartości: 0 - Linia; 2 - Linia pogrubiona; 3 - Linia kropkowana; 4 - Linia przerywana.
	color	string	Kolor połączenia (6 znaków, hex).
	label	string	Etykieta połączenia.
	selementid1	id	ID jednego elementu do połączenia.
	selementid2	id	ID drugiego elementu do połączenia.
linktriggers			Element główny dla wskaźników statusu połączeń.

Note:
Zobacz także: Map object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Elementy mapy

Element	Type	Description
elementtype	integer	Typ elementu mapy. Możliwe wartości: 0 - Host; 1 - Mapa; 2 - Wyzwalacz; 3 - Grupa hostów; 4 - Obraz.
elements		Element nadrzędny dla encji Zabbix (host, grupa hostów, mapa itp.), które są reprezentowane na mapie. Każda encja ma odpowiadający jej element (`host` itd.; zobacz na przykład Format eksportu).
label	string	Etykieta ikony.
label_location	integer	Położenie etykiety. Możliwe wartości: -1 - Użyj domyślnego ustawienia mapy; 0 - Dół; 1 - Lewo; 2 - Prawo; 3 - Góra.
x	integer	Położenie na osi X.
y	integer	Położenie na osi Y.
elementsubtype	integer	Podtyp elementu. Obsługiwane, jeśli `elementtype` ma wartość 3. Możliwe wartości: 0 - Pojedyncza grupa hostów; 1 - Wszystkie grupy hostów.

Element	Type	Description
areatype	integer	Rozmiar obszaru. Obsługiwane, jeśli `elementtype` ma wartość 1. Możliwe wartości: 0 - Taki sam jak cała mapa; 1 - Rozmiar niestandardowy.
width	integer	Szerokość obszaru. Obsługiwane, jeśli `areatype` ma wartość 1.
height	integer	Wysokość obszaru. Obsługiwane, jeśli `areatype` ma wartość 1.
viewtype	integer	Algorytm rozmieszczenia obszaru. Obsługiwane, jeśli `elementsubtype` ma wartość 1. Możliwe wartości: 0 - Rozmieść równomiernie w obszarze.
use_iconmap	integer	Użyj mapowania ikon dla tego elementu. Istotne tylko wtedy, gdy mapowanie ikon jest aktywowane na poziomie mapy. Możliwe wartości: 0 - Nie; 1 - Tak.
selementid	id	Unikalny identyfikator rekordu elementu.
icon_off		Element nadrzędny dla obrazu używanego, gdy element ma status 'OK'.
name	string	Unikalna nazwa obrazu.
icon_on		Element nadrzędny dla obrazu używanego, gdy element ma status 'Problem'.
name	string	Unikalna nazwa obrazu.
icon_disabled		Element nadrzędny dla obrazu używanego, gdy element jest wyłączony.
name	string	Unikalna nazwa obrazu.
icon_maintenance		Element nadrzędny dla obrazu używanego, gdy element jest w trybie konserwacji.
name	string	Unikalna nazwa obrazu.
urls		Element nadrzędny dla adresów URL używanych przez mapy lub każdy element mapy.
name	string	Nazwa łącza.
url	string	Adres URL łącza.
evaltype	integer	Typ oceny dla tagów.
tags		Element nadrzędny dla tagów problemów (dla elementów host i grup hostów). Jeśli podano tagi, na mapie będą wyświetlane tylko problemy z tymi tagami.
tag	string	Nazwa tagu.
value	string	Wartość tagu.
operator	integer	Operator.

Note:
Zobacz też: Obiekt elementu mapy (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Wskaźniki statusu łącza mapy

Element	Type	Opis
drawtype	integer	Styl łącza, gdy wyzwalacz jest w stanie „Problem". Możliwe wartości: 0 - Linia; 2 - Linia pogrubiona; 3 - Linia kropkowana; 4 - Linia kreskowana.
color	string	Kolor łącza (6 znaków, hex), gdy wyzwalacz jest w stanie „Problem".
trigger		Element główny dla wyzwalacza używanego do wskazywania statusu łącza.
description	string	Nazwa wyzwalacza.
expression	string	Wyrażenie wyzwalacza.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza.

Note:
Zobacz także: Obiekt wyzwalacza łącza mapy (odpowiednia właściwość o pasującej nazwie).

6 Typy mediów

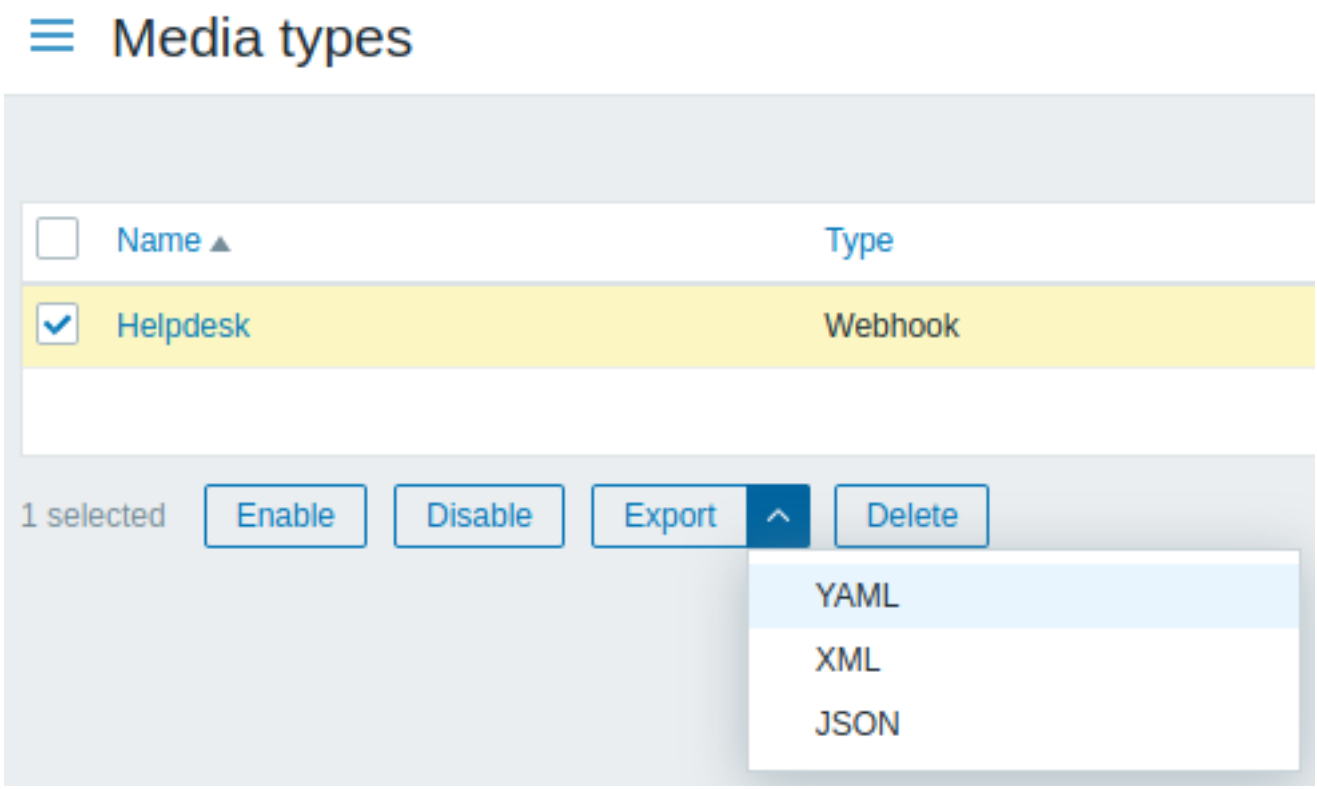
Przegląd

Typy mediów są eksportowane ze wszystkimi powiązanymi obiektami i relacjami obiektów.

Eksportowanie

Aby wyeksportować typy mediów, wykonaj następujące czynności:

1. Przejdź do *Alerty* → *Typy mediów*.
2. Zaznacz pola wyboru typów mediów, które chcesz wyeksportować.
3. Kliknij *Eksportuj* poniżej listy.



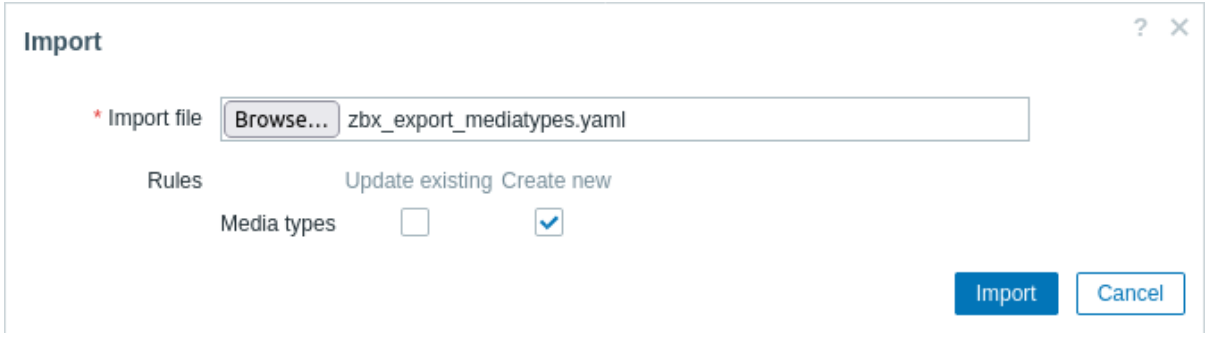
W zależności od wybranego formatu typy mediów są eksportowane do pliku lokalnego, który posiada domyślną nazwię:

- `zabbix_export_mediatypes.yaml` - przy eksporcie do YAML (domyślna opcja eksportu);
- `zabbix_export_mediatypes.xml` - przy eksporcie do XML;
- `zabbix_export_mediatypes.json` - przy eksporcie do JSON.

Importowanie

Aby zaimportować typy mediów, wykonaj następujące kroki:

1. Przejdź do *Alerts* → *Media types*.
2. Kliknij *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz wymagane opcje w regułach importu.
5. Kliknij *Import* w prawym dolnym rogu formularza konfiguracji.



Reguły importu:

Rule	Description
Update existing	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.

W frontend zostanie wyświetlony komunikat o powodzeniu lub niepowodzeniu importu.

Format eksportu

Eksport do YAML:

```yaml
zabbix_export:
  version: '7.4'
  media_types:
    - name: Pushover
      type: WEBHOOK
      parameters:
        - name: endpoint
          value: 'https://api.pushover.net/1/messages.json'
        - name: eventid
          value: '{EVENT.ID}'
        - name: event_nseverity
          value: '{EVENT.NSEVERITY}'
        - name: event_source
          value: '{EVENT.SOURCE}'
        - name: event_value
          value: '{EVENT.VALUE}'
        - name: expire
          value: '1200'
        - name: message
          value: '{ALERT.MESSAGE}'
        - name: priority_average
          value: '0'
        - name: priority_default
          value: '0'
        - name: priority_disaster
          value: '0'
        - name: priority_high
          value: '0'
        - name: priority_information
          value: '0'
        - name: priority_not_classified
          value: '0'
        - name: priority_warning
          value: '0'
        - name: retry
          value: '60'
        - name: title
          value: '{ALERT.SUBJECT}'
        - name: token
          value: '<PUSHOVER TOKEN HERE>'
        - name: triggerid
          value: '{TRIGGER.ID}'
        - name: url
          value: '{$ZABBIX.URL}'
        - name: url_title
          value: Zabbix
        - name: user
          value: '{ALERT.SENDTO}'
      status: DISABLED
```

```
    max_sessions: '0'
    script: |
      try {
          var params = JSON.parse(value),
              request = new HttpRequest(),
              data,
              response,
              severities = [
                  {name: 'not_classified', color: '#97AAB3'},
                  {name: 'information', color: '#7499FF'},
                  {name: 'warning', color: '#FFC859'},
                  {name: 'average', color: '#FFA059'},
                  {name: 'high', color: '#E97659'},
                  {name: 'disaster', color: '#E45959'},
                  {name: 'resolved', color: '#009900'},
                  {name: 'default', color: '#000000'}
              ],
              priority;

          if (typeof params.HTTPProxy === 'string' && params.HTTPProxy.trim() !== '') {
              request.setProxy(params.HTTPProxy);
          }

          if ([0, 1, 2, 3].indexOf(parseInt(params.event_source)) === -1) {
              throw 'Nieprawidłowy parametr "event_source": "' + params.event_source + '".\nMusi mieć wa
          }

          if (params.event_value !== '0' && params.event_value !== '1'
              && (params.event_source === '0' || params.event_source === '3')) {
              throw 'Nieprawidłowy parametr "event_value": ' + params.event_value + '\nMusi mieć wartość
          }

          if ([0, 1, 2, 3, 4, 5].indexOf(parseInt(params.event_nseverity)) === -1) {
              params.event_nseverity = '7';
          }

          if (params.event_value === '0') {
              params.event_nseverity = '6';
          }

          priority = params['priority_' + severities[params.event_nseverity].name] || params.priority_de

          if (isNaN(priority) || priority < -2 || priority > 2) {
              throw '"priority" powinno mieć wartość -2..2';
          }

          if (params.event_source === '0' && isNaN(params.triggerid)) {
              throw 'pole "triggerid" nie jest liczbą';
          }

          if (isNaN(params.eventid)) {
              throw 'pole "eventid" nie jest liczbą';
          }

          if (typeof params.message !== 'string' || params.message.trim() === '') {
              throw 'pole "message" nie może być puste';
          }

          data = {
              token: params.token,
              user: params.user,
              title: params.title,
```

```
            message: params.message,
            url: (params.event_source === '0')
                ? params.url + '/tr_events.php?triggerid=' + params.triggerid + '&eventid=' + params.e
                : params.url,
            url_title: params.url_title,
            priority: priority
        };

        if (priority == 2) {
            if (isNaN(params.retry) || params.retry < 30) {
                throw 'pole "retry" powinno być liczbą o wartości co najmniej 30, jeśli "priority" jes
            }

            if (isNaN(params.expire) || params.expire > 10800) {
                throw 'pole "expire" powinno być liczbą o wartości co najwyżej 10800, jeśli "priority"
            }

            data.retry = params.retry;
            data.expire = params.expire;
        }

        data = JSON.stringify(data);
        Zabbix.log(4, '[ Pushover Webhook ] Wysyłanie żądania: ' + params.endpoint + '\n' + data);

        request.addHeader('Content-Type: application/json');
        response = request.post(params.endpoint, data);

        Zabbix.log(4, '[ Pushover Webhook ] Otrzymano odpowiedź z kodem statusu ' + request.getStatus(

        if (response !== null) {
            try {
                response = JSON.parse(response);
            }
            catch (error) {
                Zabbix.log(4, '[ Pushover Webhook ] Nie udało się przeanalizować odpowiedzi otrzymanej
                response = null;
            }
        }

        if (request.getStatus() != 200 || response === null || typeof response !== 'object' || respons
            if (response !== null && typeof response === 'object' && typeof response.errors === 'objec
                    && typeof response.errors[0] === 'string') {
                throw response.errors[0];
            }
            else {
                throw 'Nieznany błąd. Sprawdź dziennik debugowania, aby uzyskać więcej informacji.';
            }
        }

        return 'OK';
    }
    catch (error) {
        Zabbix.log(4, '[ Pushover Webhook ] Wysyłanie powiadomienia Pushover nie powiodło się: ' + err
        throw 'Wysyłanie powiadomienia Pushover nie powiodło się: ' + error;
    }
description: |
    Zapoznaj się z instrukcją konfiguracji tutaj: https://git.zabbix.com/projects/ZBX/repos/zabbix/bro

    Ustaw parametr token na klucz aplikacji Pushover.
    Przypisując media Pushover do użytkownika Zabbix, dodaj klucz użytkownika do pola Wyślij do.
message_templates:
    - event_source: TRIGGERS
      operation_mode: PROBLEM
```

```
      subject: 'Problem: {EVENT.NAME}'
      message: |
        Problem rozpoczął się o {EVENT.TIME} dnia {EVENT.DATE}
        Nazwa problemu: {EVENT.NAME}
        Host: {HOST.NAME}
        Ważność: {EVENT.SEVERITY}
        Dane operacyjne: {EVENT.OPDATA}
        Oryginalny ID problemu: {EVENT.ID}
        {TRIGGER.URL}
    - event_source: TRIGGERS
      operation_mode: RECOVERY
      subject: 'Rozwiązano w {EVENT.DURATION}: {EVENT.NAME}'
      message: |
        Problem został rozwiązany o {EVENT.RECOVERY.TIME} dnia {EVENT.RECOVERY.DATE}
        Nazwa problemu: {EVENT.NAME}
        Czas trwania problemu: {EVENT.DURATION}
        Host: {HOST.NAME}
        Ważność: {EVENT.SEVERITY}
        Oryginalny ID problemu: {EVENT.ID}
        {TRIGGER.URL}
    - event_source: TRIGGERS
      operation_mode: UPDATE
      subject: 'Zaktualizowany problem w {EVENT.AGE}: {EVENT.NAME}'
      message: |
        {USER.FULLNAME} {EVENT.UPDATE.ACTION} problem o {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}.
        {EVENT.UPDATE.MESSAGE}

        Bieżący status problemu to {EVENT.STATUS}, wiek wynosi {EVENT.AGE}, potwierdzony: {EVENT.ACK.S
    - event_source: DISCOVERY
      operation_mode: PROBLEM
      subject: 'Odkrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}'
      message: |
        Reguła odkrywania: {DISCOVERY.RULE.NAME}

        Adres IP urządzenia: {DISCOVERY.DEVICE.IPADDRESS}
        DNS urządzenia: {DISCOVERY.DEVICE.DNS}
        Status urządzenia: {DISCOVERY.DEVICE.STATUS}
        Czas działania urządzenia: {DISCOVERY.DEVICE.UPTIME}

        Nazwa usługi urządzenia: {DISCOVERY.SERVICE.NAME}
        Port usługi urządzenia: {DISCOVERY.SERVICE.PORT}
        Status usługi urządzenia: {DISCOVERY.SERVICE.STATUS}
        Czas działania usługi urządzenia: {DISCOVERY.SERVICE.UPTIME}
    - event_source: AUTOREGISTRATION
      operation_mode: PROBLEM
      subject: 'Automatyczna rejestracja: {HOST.HOST}'
      message: |
        Nazwa hosta: {HOST.HOST}
        IP hosta: {HOST.IP}
        Port agenta: {HOST.PORT}
```

Elementy eksportowane

Eksportowane elementy są wyjaśnione w poniższej tabeli.

Element	Typ	Opis
name	string	(wymagane) Nazwa typu mediów.
type	string	(wymagane) Typ komunikacji używany przez typ mediów. Możliwe wartości:[1] EMAIL (0), SMS (1), SCRIPT (2), WEBHOOK (4).
status	string	Czy typ mediów jest włączony. Możliwe wartości:[1] ENABLED (0, domyślne), DISABLED (1)

Element	Typ	Opis
max_sessions	integer	Maksymalna liczba alertów, które można przetwarzać równolegle. Możliwe wartości dla wiadomości SMS:[1] 1 (domyślnie). Możliwe wartości dla innych typów mediów:[1] 0-100 (gdzie 0 — bez ograniczeń).
attempts	integer	Maksymalna liczba prób wysłania alertu. Możliwe wartości:[1] 1-10 (domyślnie: 3).
attempt_interval	string	Interwał (z użyciem sekund lub przyrostka czasu) między kolejnymi próbami. Możliwe wartości:[1] 0-60 s (domyślnie: 10 s).
description	string	Opis typu mediów.
message_templates		Element główny szablonu wiadomości dla typu mediów.
event_source	string	(wymagane) Źródło zdarzenia. Możliwe wartości:[1] TRIGGERS (0), DISCOVERY (1), AUTOREGISTRATION (2), INTERNAL (3), SERVICE (4).
operation_mode	string	Tryb działania. Możliwe wartości:[1] PROBLEM (0), RECOVERY (1), UPDATE (2).
subject	string	Temat wiadomości.
message	string	Treść wiadomości.

Note:
Zobacz także: Obiekt typu mediów (odnieś się do odpowiedniej właściwości o pasującej nazwie).

E-mail

Następujące dodatkowe elementy są eksportowane tylko dla typu nośnika *E-mail*.

Element	Type	Opis
provider	string	Dostawca poczty e-mail.
smtp_server	string	Serwer SMTP.
smtp_port	integer	Port serwera SMTP, z którym ma zostać nawiązane połączenie. Domyślnie: 25.
smtp_helo	string	HELO SMTP.
smtp_email	string	Adres e-mail, z którego będą wysyłane powiadomienia.
smtp_security	string	Poziom zabezpieczeń połączenia SMTP do użycia. Możliwe wartości:[1] NONE (0, domyślnie), STARTTLS (1), SSL_OR_TLS (2).
smtp_verify_host	string	Weryfikacja hosta SSL dla SMTP. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
smtp_verify_peer	string	Weryfikacja peera SSL dla SMTP. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
smtp_authentication	string	Metoda uwierzytelniania SMTP do użycia. Możliwe wartości:[1] NONE (0, domyślnie), PASSWORD (1), OAUTH (2).
username	string	Nazwa użytkownika.
password	string	Hasło uwierzytelniające.
redirection_url	string	Adres URL frontendu Zabbix, do którego nastąpi przekierowanie po autoryzacji OAuth.
client_id	string	Identyfikator klienta zarejestrowany na serwerze autoryzacji OAuth.
authorization_url	string	Adres URL OAuth wraz z parametrami służący do uzyskania tokenów dostępu i odświeżania.
token_url	string	Adres URL OAuth służący do wymiany tokenu autoryzacyjnego na tokeny dostępu i odświeżania.
message_format	string	Format wiadomości. Możliwe wartości:[1] TEXT (0), HTML (1, domyślnie).

Note:
Zobacz także: Media type object (odpowiednia właściwość o pasującej nazwie).

SMS

Następujące dodatkowe elementy są eksportowane tylko dla typu mediów *SMS*.

Element	Typ	Opis
gsm_modem	string	(wymagane) Nazwa urządzenia szeregowego modemu GSM.

Note:
Zobacz także: Obiekt typu mediów (sprawdź odpowiednią właściwość z pasującą nazwą).

Skrypt

Następujące dodatkowe elementy są eksportowane tylko dla typu nośnika *Skrypt*.

Element	Typ	Opis
nazwa skryptu	string	(wymagane) Nazwa skryptu.
parametry		Element główny dla parametrów skryptu.
sortorder	string	(wymagane) Kolejność parametrów skryptu przekazywanych do skryptu jako argumenty wiersza poleceń.
wartość	string	Wartość parametru skryptu.

Note:
Zobacz także: Obiekt typu nośnika (sprawdź odpowiednią właściwość o pasującej nazwie).

Webhook

Następujące dodatkowe elementy są eksportowane tylko dla typu mediów *Webhook*.

Element	Typ	Opis
script	string	Skrypt.
timeout	string	Limitu czasu żądania HTTP dla skryptu JavaScript. Możliwe wartości:[1] 1-60s (domyślnie: 30s).
process_tags	string	Czy przetwarzać zwrócone tagi. Możliwe wartości:[1] NIE (0, domyślnie), TAK (1).
show_event_menu	string	Wskaż obecność wpisu w menu zdarzeń„ jeśli makro {EVENT.TAGS.*} zostało pomyślnie przetworzone w polach `event_menu_url` i `event_menu_name`. Możliwe wartości:[1] NIE (0, domyślnie), TAK (1).
event_menu_url	string	Adres URL wpisu menu zdarzeń. Obsługuje makro {EVENT.TAGS.*}.
event_menu_name	string	Nazwa wpisu menu zdarzeń. Obsługuje makro {EVENT.TAGS.*}.
parameters		Element główny dla parametrów typu mediów webhook.
name	string	(wymagane) Nazwa parametru webhook.
value	string	Wartość parametru webhook.

Note:
Zobacz także: Obiekt typu media (odnieś się do odpowiedniej właściwości o pasującej nazwie).

Przypisy

[1] Wartości liczb całkowitych API w nawiasach, na przykład ENABLED (0), są wymienione tylko w celach informacyjnych. Aby uzyskać więcej informacji, zobacz powiązaną stronę obiektu API w tabeli lub na końcu każdej sekcji.

15 Odkrywanie

Przegląd Wykrywanie zapewnia automatyczne sposoby dodawania hostów i konfigurowania monitorowania bez ręcznej konfiguracji. Zabbix obsługuje wiele metod wykrywania, aby dostosować się do różnych scenariuszy.

Metody wykrywania

- Wykrywanie sieci — automatycznie wykrywaj hosty i usługi dostępne w Twojej sieci.
- Autorejestracja aktywnego agenta — umożliwia aktywnym agentom automatyczną rejestrację w serwerze.
- Wykrywanie niskiego poziomu — automatycznie twórz pozycje, wyzwalacze, wykresy i hosty dla wykrytych obiektów, takich jak systemy plików lub interfejsy sieciowe, przy użyciu prototypów i zagnieżdżonych reguł wykrywania.

1 Wykrywanie sieci

Przegląd

Zabbix oferuje funkcję automatycznego wykrywania sieci, która jest skuteczna i bardzo elastyczna.

Przy prawidłowo skonfigurowanym wykrywaniu sieci można:

- przyspieszyć wdrożenie Zabbix
- uprościć administrację
- używać Zabbix w szybko zmieniających się środowiskach bez nadmiernych nakładów administracyjnych

Wykrywanie sieci w Zabbix opiera się na następujących informacjach:

- zakresy adresów IP
- dostępność usług zewnętrznych (FTP, SSH, WEB, POP3, IMAP, TCP itp.)
- informacje otrzymane od Zabbix agent (obsługiwany jest tylko tryb nieszyfrowany)
- informacje otrzymane od agenta SNMP

Nie zapewnia ono:

- wykrywania topologii sieci

Wykrywanie sieci zasadniczo składa się z dwóch faz: wykrywania i działań.

Wykrywanie

Zabbix okresowo skanuje zakresy IP zdefiniowane w regułach wykrywania sieci. Częstotliwość sprawdzania można skonfigurować indywidualnie dla każdej reguły.

Każda reguła ma zdefiniowany zestaw kontroli usług do wykonania dla zakresu IP.

Reguły wykrywania są przetwarzane przez menedżera wykrywania. Menedżer wykrywania tworzy zadanie dla każdej reguły z listą zadań cząstkowych (kontroli sieciowych). Kontrole sieciowe są wykonywane równolegle przez dostępne procesy robocze wykrywania (ich liczba jest konfigurowalna w frontendzie dla każdej reguły). Tylko kontrole z tym samym adresem IP i portem są planowane sekwencyjnie, ponieważ niektóre urządzenia nie akceptują równoległych połączeń na tym samym porcie.

Rozmiar kolejki kontroli sieciowych jest ograniczony do 2000000 lub około 4 GB pamięci. Jeśli kolejka się zapełni, reguła wykrywania zostanie pominięta, a w logu zostanie zapisany komunikat ostrzegawczy. Możesz użyć wewnętrznej pozycji `zabbix[discovery_queue]`, aby monitorować liczbę kontroli wykrywania w kolejce.

Kontrole wykrywania są przetwarzane niezależnie od innych kontroli. Jeśli jakiekolwiek kontrole nie znajdą usługi (lub zakończą się niepowodzeniem), pozostałe kontrole nadal będą przetwarzane.

> **Note:**
> Jeśli reguła wykrywania zostanie zmieniona podczas wykonywania, bieżące wykonanie wykrywania zostanie przerwane.

Każda kontrola usługi i hosta (IP) wykonywana przez moduł wykrywania sieci generuje zdarzenie wykrywania.

Zdarzenie	Wynik kontroli usługi
Usługa wykryta	Usługa jest „up” po tym, jak była „down”, lub gdy została wykryta po raz pierwszy.
Usługa up	Usługa jest „up” po tym, jak była już „up”.
Usługa utracona	Usługa jest „down” po tym, jak była „up”.
Usługa down	Usługa jest „down” po tym, jak była już „down”.
Host wykryty	Co najmniej jedna usługa hosta jest „up” po tym, jak wszystkie usługi tego hosta były „down”, lub została wykryta usługa należąca do niezarejestrowanego hosta.
Host up	Co najmniej jedna usługa hosta jest „up” po tym, jak co najmniej jedna usługa była już „up”.
Host utracony	Wszystkie usługi hosta są „down” po tym, jak co najmniej jedna była „up”.
Host down	Wszystkie usługi hosta są „down” po tym, jak były już „down”.

Akcje

Zdarzenia wykrywania mogą stanowić podstawę odpowiednich akcji, takich jak:

- Wysyłanie powiadomień
- Dodawanie/usuwanie hostów
- Włączanie/wyłączanie hostów
- Dodawanie hostów do grupy
- Usuwanie hostów z grupy
- Dodawanie tagów do hosta
- Usuwanie tagów z hosta
- Podłączanie szablonu do hostów/odłączanie szablonu od hostów
- Wykonywanie zdalnych skryptów

Akcje te można skonfigurować z uwzględnieniem typu urządzenia, adresu IP, statusu, czasu działania/przestoju itd. Pełne informacje na temat konfigurowania akcji dla zdarzeń opartych na wykrywaniu sieci można znaleźć na stronach operacji i warunków akcji.

Ponieważ akcje wykrywania sieci są oparte na zdarzeniach, będą uruchamiane zarówno wtedy, gdy wykryty host jest online, jak i wtedy, gdy jest offline. Zdecydowanie zaleca się dodanie warunku akcji *Discovery status: up*, aby uniknąć uruchamiania takich akcji jak *Add host* przy zdarzeniach *Service Lost*/*Service Down*. W przeciwnym razie, jeśli wykryty host zostanie usunięty ręcznie, nadal będzie generował zdarzenia *Service Lost*/*Service Down* i zostanie odtworzony podczas następnego cyklu wykrywania.

Note:
Podłączanie szablonów do wykrytego hosta zakończy się zbiorczym niepowodzeniem, jeśli którykolwiek z szablonów możliwych do podłączenia zawiera unikalną encję (np. klucz pozycji), która jest taka sama jak unikalna encja (np. klucz pozycji) już istniejąca na hoście lub w innym z szablonów możliwych do podłączenia.

Tworzenie hosta

Host jest dodawany, jeśli wybrano operację *Add host*. Host jest również dodawany, nawet jeśli operacja *Add host* nie występuje, jeśli wybierzesz operacje skutkujące działaniami na hoście. Takimi operacjami są:

- włącz host
- wyłącz host
- dodaj host do grupy hostów
- podłącz szablon do hosta

Utworzone hosty są dodawane do grupy *Discovered hosts* (domyślnie; można to skonfigurować w *Administration* > *General* > *Other*). Jeśli chcesz, aby hosty były dodawane do innej grupy, dodaj operację *Remove from host groups* (wskazując „Discovered hosts") oraz operację *Add to host groups* (wskazując inną grupę hostów), ponieważ host musi należeć do grupy hostów.

Adres IP wykrytego urządzenia, wraz ze źródłem wykrywania (serwer Zabbix, proxy Zabbix lub grupa proxy) oraz typem interfejsu, jest używany jako kryterium wyszukiwania hosta w systemie. Jeśli host z tym samym adresem IP, typem interfejsu i źródłem wykrywania już istnieje, ten host będzie celem wykonywania operacji. Jeśli źródło wykrywania jest inne, wykryty obiekt jest traktowany jako inny host i może zostać utworzony nowy host.

Jeśli adres IP wykrytego hosta zostanie zmieniony lub interfejs zostanie usunięty, przy następnym wykryciu zostanie utworzony nowy host.

Nadawanie nazw hostom

Podczas dodawania hostów nazwa hosta jest wynikiem odwrotnego wyszukiwania DNS lub adresem IP, jeśli odwrotne wyszukiwanie się nie powiedzie. Wyszukiwanie jest wykonywane z serwera Zabbix lub proxy Zabbix, w zależności od tego, który z nich wykonuje wykrywanie. Jeśli wyszukiwanie nie powiedzie się na proxy, nie jest ponawiane na serwerze. Jeśli host o takiej nazwie już istnieje, do nazwy kolejnego hosta zostanie dodane **_2**, następnie **_3** itd.

Możliwe jest również zastąpienie wyszukiwania DNS/IP i użycie zamiast tego wartości pozycji jako nazwy hosta, na przykład:

- Można wykryć wiele serwerów z uruchomionym agent Zabbix przy użyciu pozycji agenta Zabbix do wykrywania i automatycznie przypisać im odpowiednie nazwy na podstawie wartości tekstowej zwróconej przez tę pozycję
- Można wykryć wiele urządzeń sieciowych SNMP przy użyciu pozycji agenta SNMP do wykrywania i automatycznie przypisać im odpowiednie nazwy na podstawie wartości tekstowej zwróconej przez tę pozycję

Jeśli nazwa hosta została ustawiona przy użyciu wartości pozycji, nie jest aktualizowana podczas kolejnych kontroli wykrywania. Jeśli nie można ustawić nazwy hosta przy użyciu wartości pozycji, używana jest wartość domyślna (nazwa DNS).

Jeśli host z wykrytym adresem IP już istnieje, a źródło wykrywania (serwer Zabbix, proxy lub grupa proxy) nie uległo zmianie, nowy host nie jest tworzony. Jeśli źródło wykrywania jest inne, wykryta jednostka jest traktowana jako odrębna i może zostać utworzony nowy host. Jednak jeśli akcja wykrywania zawiera operacje (podłączenie szablonu, dodanie do grupy hostów itp.), są one wykonywane na istniejącym hoście dopasowanym według adresu IP, typu interfejsu i źródła wykrywania.

Usuwanie hostów

Hosty wykryte przez regułę wykrywania sieci są usuwane automatycznie z *Monitoring* > *Discovery*, jeśli wykryta jednostka nie znajduje się już w zakresie adresów IP tej reguły. Hosty są usuwane natychmiast.

Tworzenie interfejsów podczas dodawania hostów

Gdy hosty są dodawane w wyniku wykrywania sieci, ich interfejsy są tworzone zgodnie z następującymi zasadami:

- Wykryte usługi — na przykład, jeśli sprawdzenie SNMP zakończyło się powodzeniem, zostanie utworzony interfejs SNMP.
- Jeśli host odpowiedział zarówno na żądania agenta Zabbix, jak i SNMP, zostaną utworzone oba typy interfejsów.
- Jeśli kryteriami unikalności są dane zwrócone przez agenta Zabbix lub SNMP, pierwszy interfejs znaleziony dla hosta zostanie utworzony jako domyślny. Pozostałe adresy IP zostaną dodane jako dodatkowe interfejsy. Warunki akcji (takie jak IP hosta) nie mają wpływu na dodawanie interfejsów. *Należy pamiętać*, że będzie to działać, jeśli wszystkie interfejsy zostaną wykryte przez tę samą regułę wykrywania. Jeśli inna reguła wykrywania wykryje inny interfejs tego samego hosta, zostanie dodany dodatkowy host.
- Jeśli host odpowiadał tylko na sprawdzenia agenta, zostanie utworzony tylko z interfejsem agenta. Jeśli później zacznie odpowiadać na SNMP, zostaną dodane dodatkowe interfejsy SNMP.
- Jeśli początkowo zostały utworzone 3 oddzielne hosty, wykryte na podstawie kryteriów unikalności „IP", a następnie reguła wykrywania zostanie zmodyfikowana tak, że hosty A, B i C będą miały identyczny wynik kryteriów unikalności, hosty B i C zostaną utworzone jako dodatkowe interfejsy dla A, pierwszego hosta. Poszczególne hosty B i C pozostaną. W *Monitoring* > *Discovery* dodane interfejsy będą wyświetlane w kolumnie „Discovered device", czarną czcionką i z wcięciem, ale kolumna „Monitored host" będzie wyświetlać tylko A, pierwszego utworzonego hosta. „Uptime/Downtime" nie jest mierzone dla adresów IP, które są traktowane jako dodatkowe interfejsy.

Zmiana ustawienia proxy

Hosty wykryte przez różne proxy nie zawsze są traktowane jako różne hosty. Wykrywanie i sprawdzanie unikalności zależą od struktury grup proxy: gdy proxy uruchamia regułę wykrywania i tworzy host, ten host jest dodawany do nadrzędnej grupy proxy danego proxy, a nie przypisywany do samego proxy. Gdy Zabbix podczas wykrywania sprawdza unikalność adresu IP, sprawdza hosty monitorowane przez nadrzędną grupę proxy. Hosty monitorowane przez poszczególne proxy w tej grupie (w tym proxy, które uruchomiło wykrywanie) są pomijane podczas sprawdzania unikalności, co może skutkować zduplikowanymi hostami, jeśli wiele proxy monitoruje nakładające się podsieci.

Chociaż takie zachowanie pozwala na działanie wykrywania w nakładających się zakresach adresów IP używanych przez różne podsieci, zmiana proxy przypisanego do już monitorowanej podsieci jest bardziej skomplikowana, ponieważ zmiany proxy muszą zostać spójnie zastosowane do wykrytych hostów oraz do członkostwa w nadrzędnej grupie proxy, aby uniknąć duplikatów.

Na przykład kroki wymagane do zastąpienia proxy w regule wykrywania:

1. wyłącz regułę wykrywania
2. zsynchronizuj konfigurację proxy
3. zastąp proxy w regule wykrywania
4. zastąp proxy dla wszystkich hostów wykrytych przez tę regułę (upewnij się, że hosty w nadrzędnej grupie proxy oraz wszelkie hosty monitorowane przez poszczególne proxy w tej grupie zostały zaktualizowane, aby uniknąć duplikatów)
5. włącz regułę wykrywania

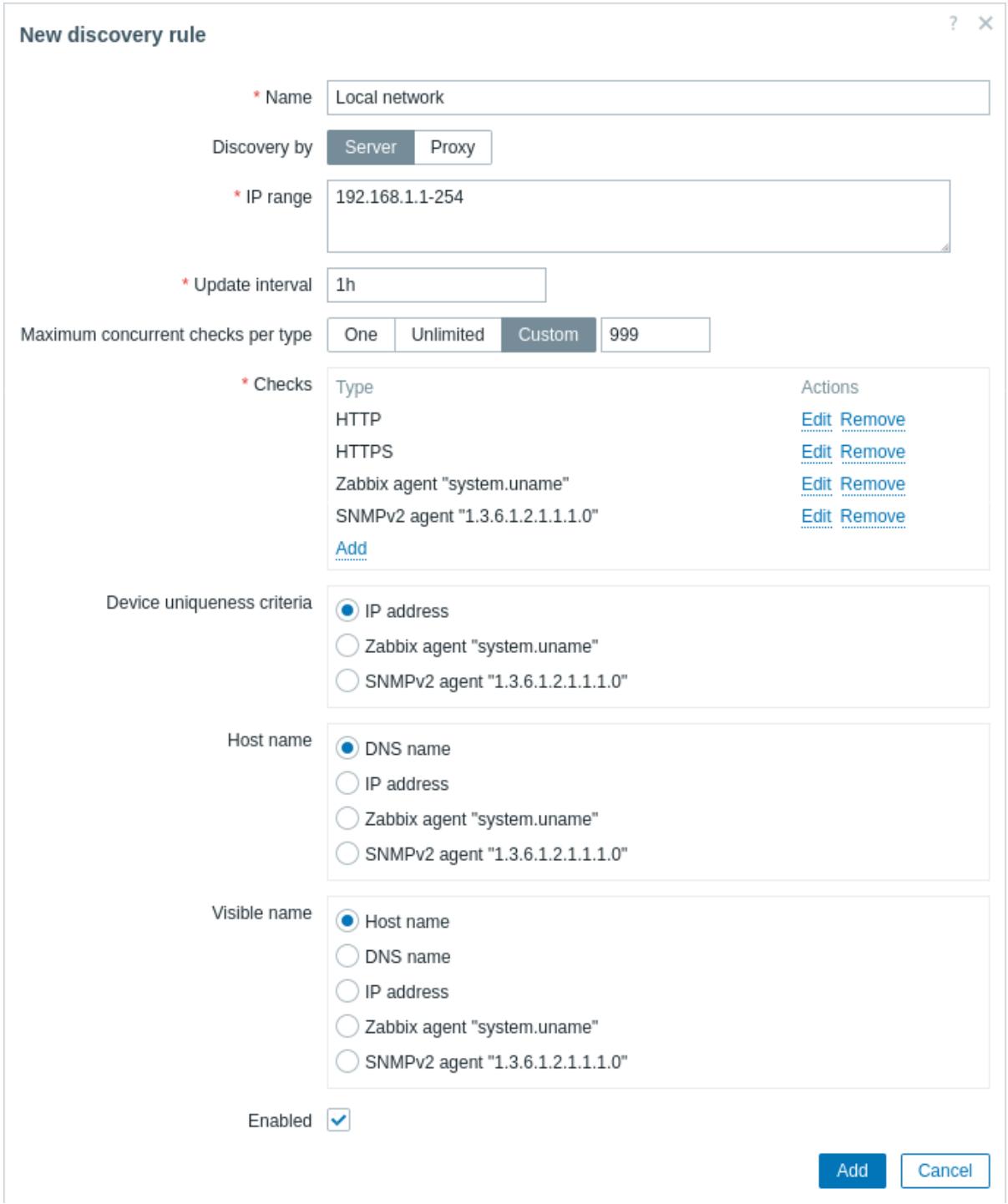
1 Konfigurowanie reguły wykrywania sieci

Przegląd

Aby skonfigurować regułę wykrywania sieci używaną przez Zabbix do wykrywania hostów i usług:

- Przejdź do *Data collection* → *Discovery*
- Kliknij *Create discovery rule* (lub nazwę reguły, aby edytować istniejącą)
- Edytuj atrybuty reguły wykrywania

Atrybuty reguły



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Unikalna nazwa reguły. Na przykład „Sieć lokalna”.
Wykrywanie przez	Wykrywanie jest wykonywane przez: **Serwer** - przez serwer Zabbix **Proxy** - przez proxy Zabbix (wybrane w polu nazwy proxy)
Zakres IP	Zakres adresów IP do wykrywania. Może mieć następujące formaty: Pojedynczy adres IP: 192.168.1.33 Zakres adresów IP: 192.168.1-10.1-255. Zakres jest ograniczony całkowitą liczbą objętych adresów (mniej niż 64K). Maska IP: 192.168.4.0/24 obsługiwane maski IP: /16 - /30 dla adresów IPv4 /112 - /128 dla adresów IPv6 Lista: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 To pole obsługuje spacje, tabulatory i wiele linii.

Parametr	Opis
Interwał aktualizacji	Ten parametr określa, jak często Zabbix będzie wykonywać regułę. Interwał jest liczony po zakończeniu wykonywania poprzedniego wystąpienia wykrywania, więc nie dochodzi do nakładania się. Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane jest makro użytkownika i jego wartość zostanie zmieniona (np. 1w → 1h), następne sprawdzenie zostanie wykonane zgodnie z poprzednią wartością (w podanym przykładzie — znacznie później).
Maksymalna liczba równoczesnych sprawdzeń na typ	Ustaw maksymalną liczbę wątków wykrywania (workerów) na jedno sprawdzenie usługi dla równoległego przetwarzania sprawdzeń wykrywania: **Jeden** - jeden wątek **Bez ograniczeń** - nieograniczona liczba wątków (ale nie większa niż w parametrze StartDiscoverers) **Niestandardowa** - ustaw niestandardową liczbę wątków (0-999) Zwróć uwagę, że wszystkie reguły wykrywania ze sprawdzeniami usług asynchronicznych SNMPv3 są przetwarzane przez jeden worker ze względu na specyfikę implementacji libsnmp, tj. zwiększenie liczby workerów nie zwiększy szybkości wykrywania.
Sprawdzenia	Zabbix użyje tej listy sprawdzeń do wykrywania. Kliknij Add, aby skonfigurować nowe sprawdzenie w wyskakującym oknie. Obsługiwane sprawdzenia: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, agent Zabbix, agent SNMPv1, agent SNMPv2, agent SNMPv3, ICMP ping. Wykrywanie oparte na protokole używa funkcji **net.tcp.service[]** do testowania każdego hosta, z wyjątkiem SNMP, które odpytuje OID SNMP. agent Zabbix jest testowany przez odpytywanie pozycji w trybie nieszyfrowanym. Więcej szczegółów znajduje się w sekcji pozycje agenta. Parametr „Porty” może mieć jedną z następujących postaci: Pojedynczy port: 22 Zakres portów: 22-45 Lista: 22-45,55,60-70 Od Zabbix 7.0. wszystkie sprawdzenia usług są wykonywane asynchronicznie, z wyjątkiem sprawdzeń LDAP. Od Zabbix 7.0 sprawdzanie HTTP/HTTPs odbywa się przez libcurl. Jeśli serwer/proxy Zabbix został skompilowany bez libcurl, sprawdzenia HTTP będą działać jak w poprzednich wersjach (tj. jako sprawdzenia TCP), ale sprawdzenia HTTPS nie będą działać.
Kryteria unikalności urządzenia	Kryteriami unikalności mogą być: **Adres IP** - nie przetwarzaj wielu urządzeń z pojedynczym adresem IP. Jeśli urządzenie z tym samym adresem IP już istnieje, zostanie uznane za już wykryte i nowy host nie zostanie dodany. **<discovery check>** - sprawdzenie agenta Zabbix albo agenta SNMP. Zwróć uwagę, że kryteria unikalności używane podczas wykrywania nie są tym samym co identyfikacja hosta w systemie podczas wykonywania akcji. Kryteria unikalności podczas wykrywania określają, czy dwa lub więcej wykrytych urządzeń to to samo urządzenie (lub różne urządzenia), podczas gdy w Zabbix jedynym kryterium identyfikacji hosta jest adres IP (zobacz Tworzenie hosta).
Nazwa hosta	Ustaw techniczną nazwę hosta tworzonego hosta, używając: **Nazwa DNS** - nazwa DNS (domyślnie) **Adres IP** - adres IP **<discovery check>** - odebrana wartość tekstowa sprawdzenia wykrywania (np. agent Zabbix, sprawdzenie agenta SNMP) Zobacz także: Nazewnictwo hostów.
Widoczna nazwa	Ustaw widoczną nazwę hosta tworzonego hosta, używając: **Nazwa hosta** - techniczna nazwa hosta (domyślnie) **Nazwa DNS** - nazwa DNS **Adres IP** - adres IP **<discovery check>** - odebrana wartość tekstowa sprawdzenia wykrywania (np. agent Zabbix, sprawdzenie agenta SNMP) Zobacz także: Nazewnictwo hostów.
Włączone	Po zaznaczeniu pola wyboru reguła jest aktywna i będzie wykonywana przez serwer Zabbix. Jeśli pole nie jest zaznaczone, reguła nie jest aktywna. Nie będzie wykonywana.

Przekroczenie limitu deskryptorów plików

W przypadku dużej liczby równoczesnych kontroli możliwe jest wyczerpanie limitu deskryptorów plików dla discovery manager.

Liczba deskryptorów plików wymaganych do wykrywania jest równa liczbie discovery workers * 1000. Domyślnie istnieje 5 discovery workers, podczas gdy miękki limit systemu wynosi około 1024.

Jeśli ten limit zostanie osiągnięty, Zabbix zmniejszy domyślną liczbę równoczesnych kontroli na typ dla każdego workera i zapisze ostrzeżenie do pliku dziennika. Jeśli jednak użytkownik ustawił wyższą wartość dla *Maximum concurrent checks per type* niż wartość obliczona przez Zabbix, Zabbix użyje wartości zdefiniowanej przez użytkownika dla jednego workera.

Przykład z życia

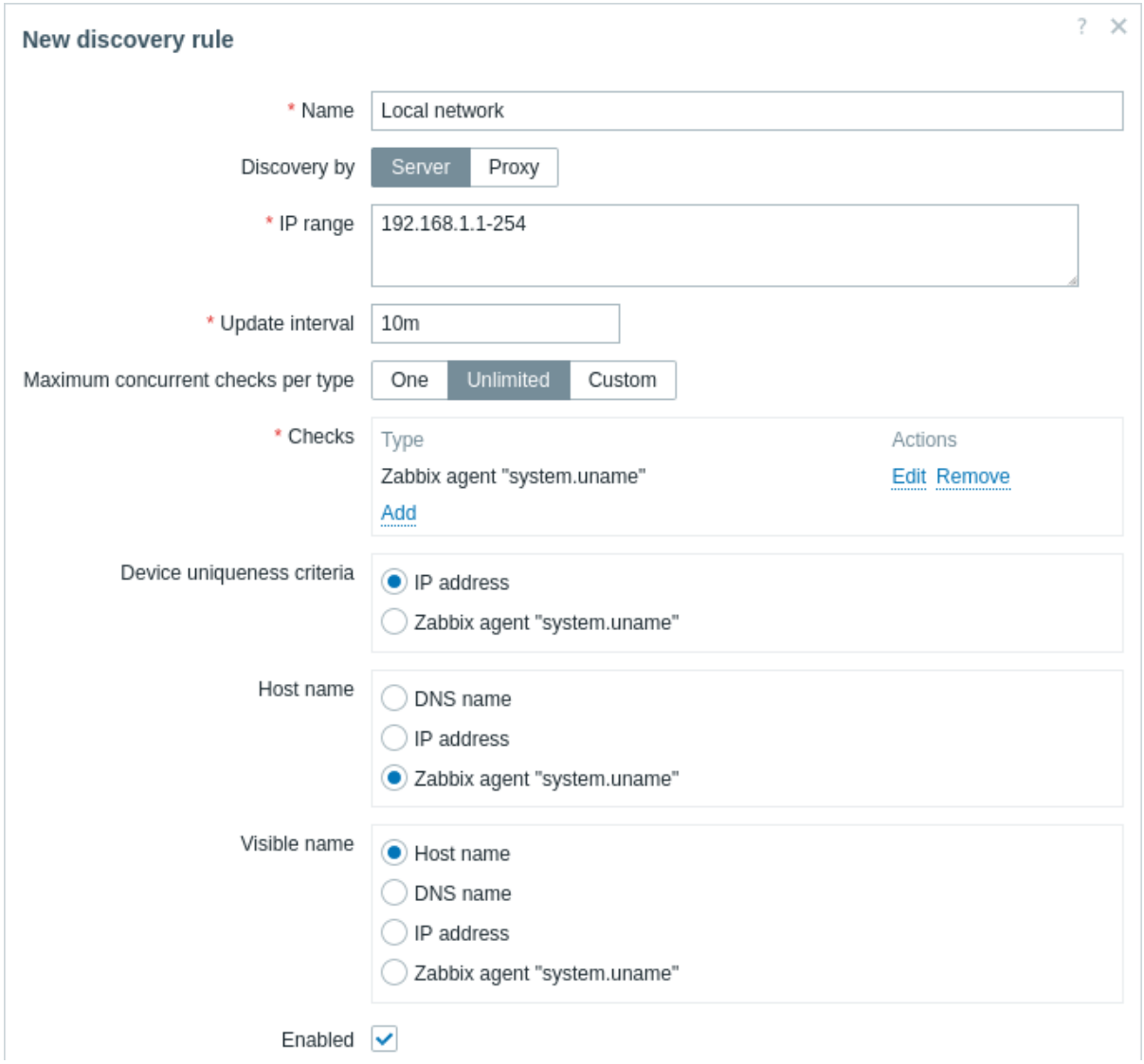
W tym przykładzie chcemy skonfigurować wykrywanie sieci dla sieci lokalnej o zakresie adresów IP 192.168.1.1-192.168.1.254.

W naszym scenariuszu chcemy:

- wykrywać te hosty, na których działa agent Zabbix
- uruchamiać wykrywanie co 10 minut
- dodać host do monitorowania, jeśli czas działania hosta wynosi więcej niż 1 godzinę
- usuwać hosty, jeśli czas niedostępności hosta przekracza 24 godziny
- dodawać hosty z systemem Linux do grupy ”Linux servers”
- dodawać hosty z systemem Windows do grupy ”Windows servers”
- używać szablonu *Linux* dla hostów z systemem Linux
- używać szablonu *Windows* dla hostów z systemem Windows

Krok 1

Definiowanie reguły wykrywania sieci dla naszego zakresu adresów IP.



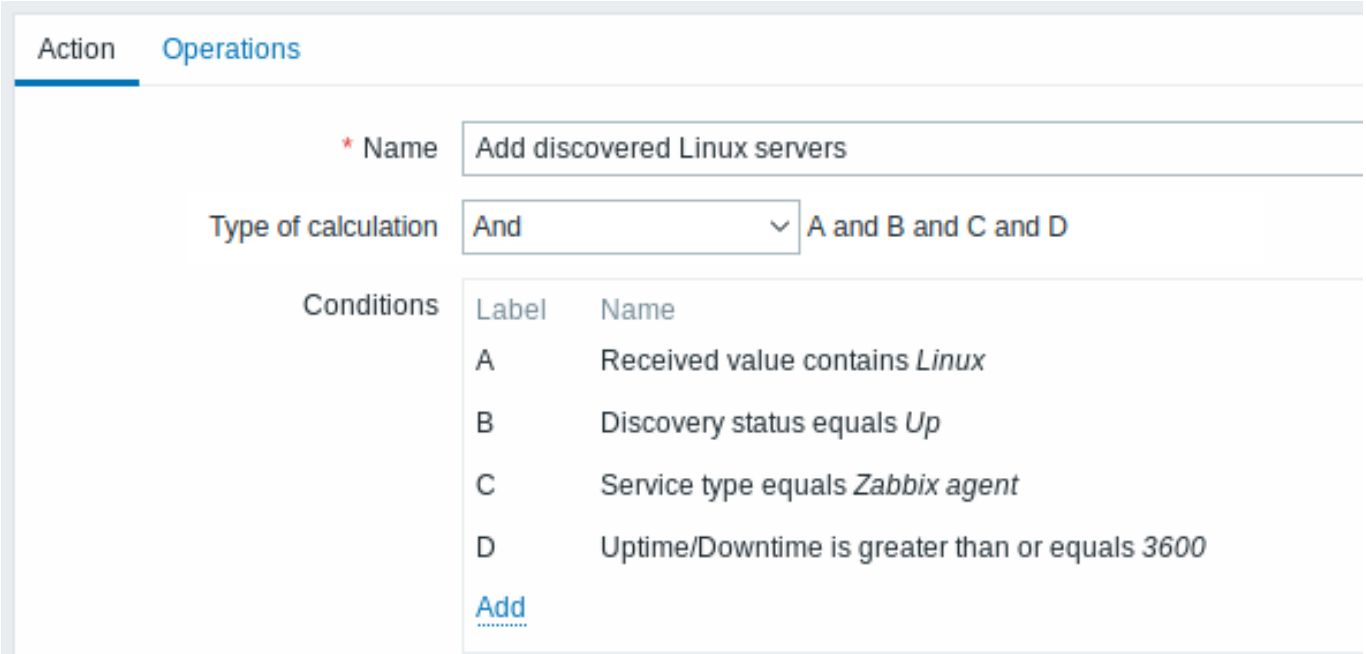
Zabbix spróbuje wykryć hosty w zakresie adresów IP 192.168.1.1-192.168.1.254, łącząc się z agentami Zabbix i pobierając wartość z klucza **system.uname**. Wartość otrzymana od agent może zostać użyta do nadania nazw hostom, a także do zastosowania różnych akcji dla różnych systemów operacyjnych. Na przykład powiązać serwery Windows z szablonem *Windows*, a serwery Linux z szablonem *Linux*.

Reguła będzie wykonywana co 10 minut.

Po dodaniu tej reguły Zabbix automatycznie rozpocznie wykrywanie i generowanie zdarzeń opartych na wykrywaniu do dalszego przetwarzania.

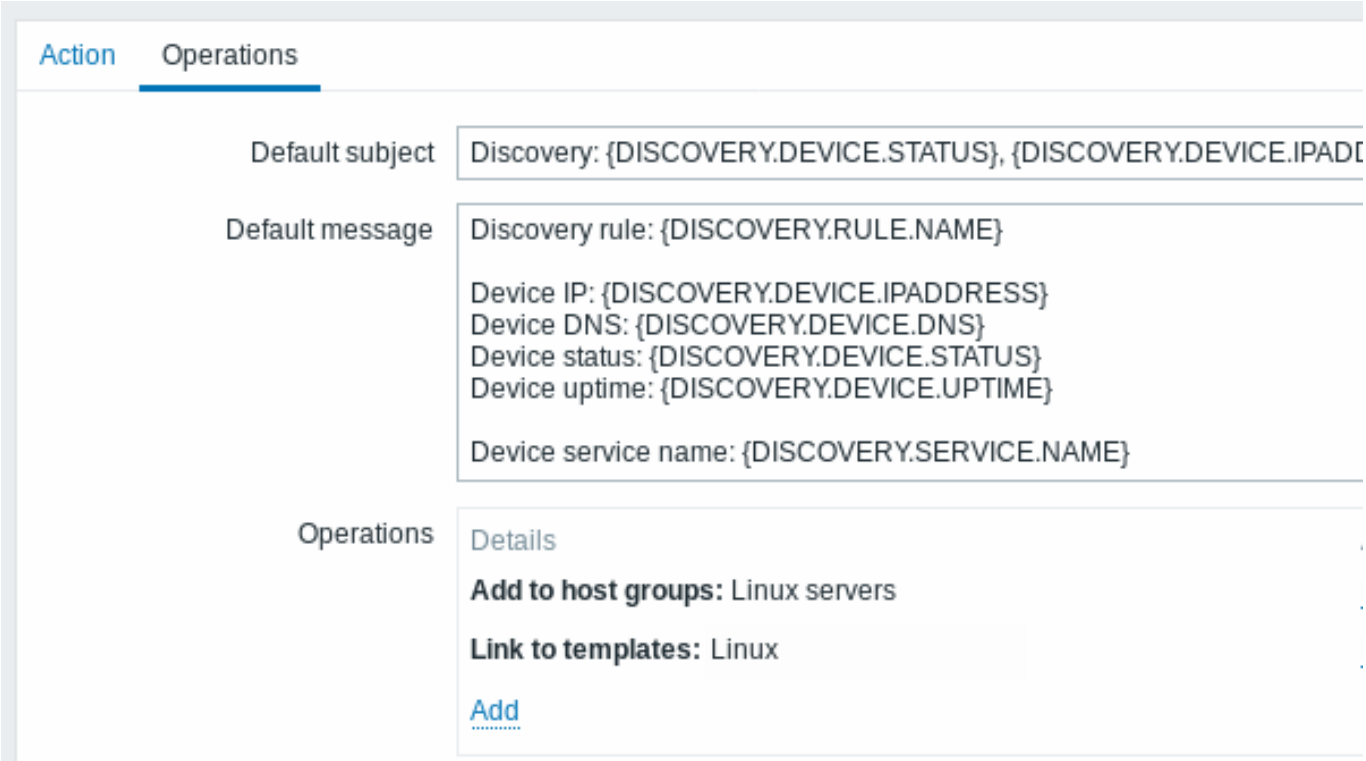
Krok 2

Definiowanie akcji wykrywania do dodawania wykrytych serwerów Linux do odpowiedniej grupy/szablonu.



Akcja zostanie uruchomiona, jeśli:

- usługa „Zabbix agent" ma stan „up"
- wartość system.uname (klucz Zabbix agent użyty przez nas w definicji reguły) zawiera „Linux"
- czas działania wynosi 1 godzinę (3600 sekund) lub więcej

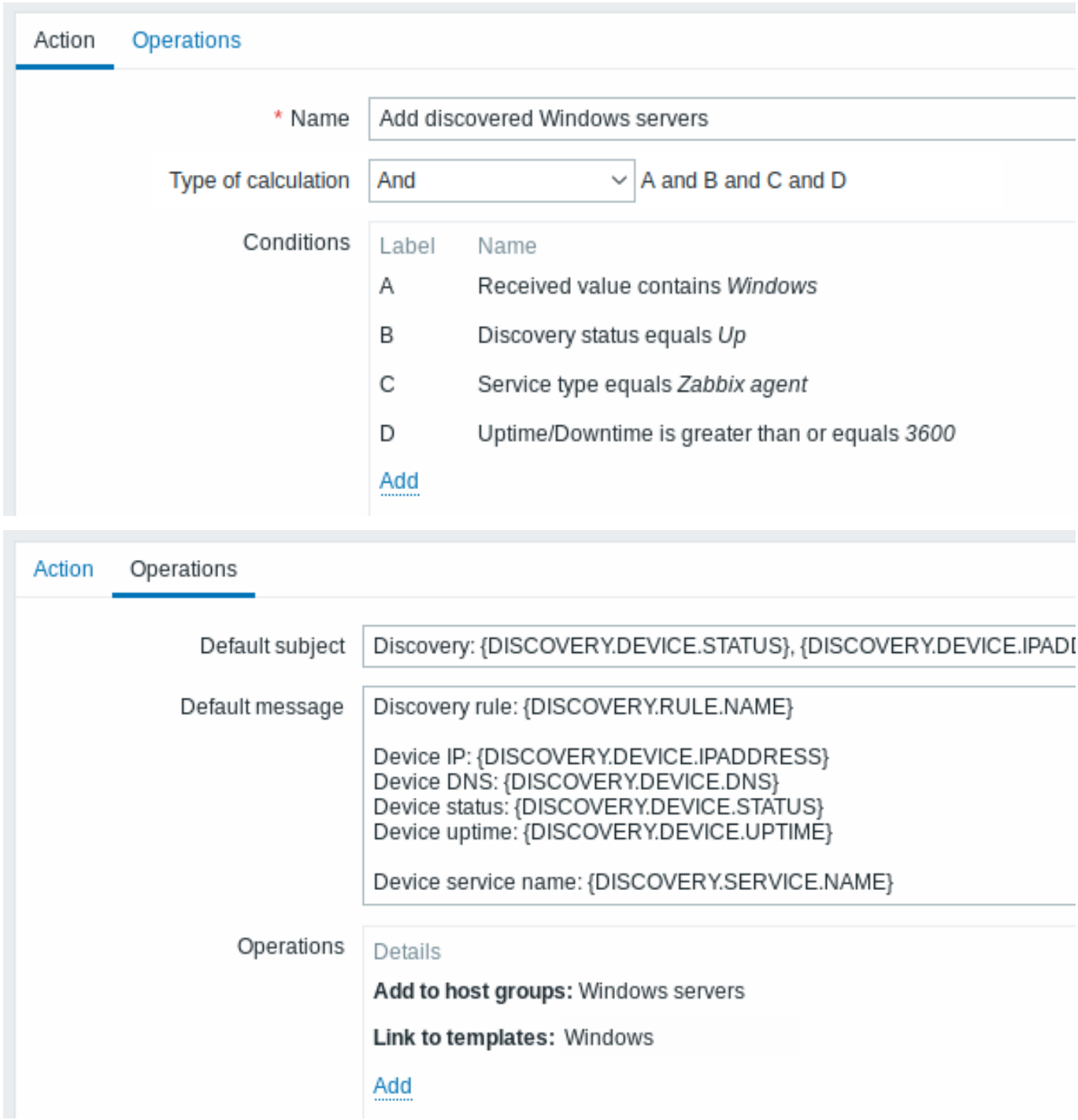


Akcja wykona następujące operacje:

- doda wykryty host do grupy „Linux servers" (a także doda host, jeśli nie został dodany wcześniej)
- podłączy host do szablonu *Linux*. Zabbix automatycznie rozpocznie monitorowanie hosta przy użyciu pozycji i wyzwalaczy z szablonu „Linux".

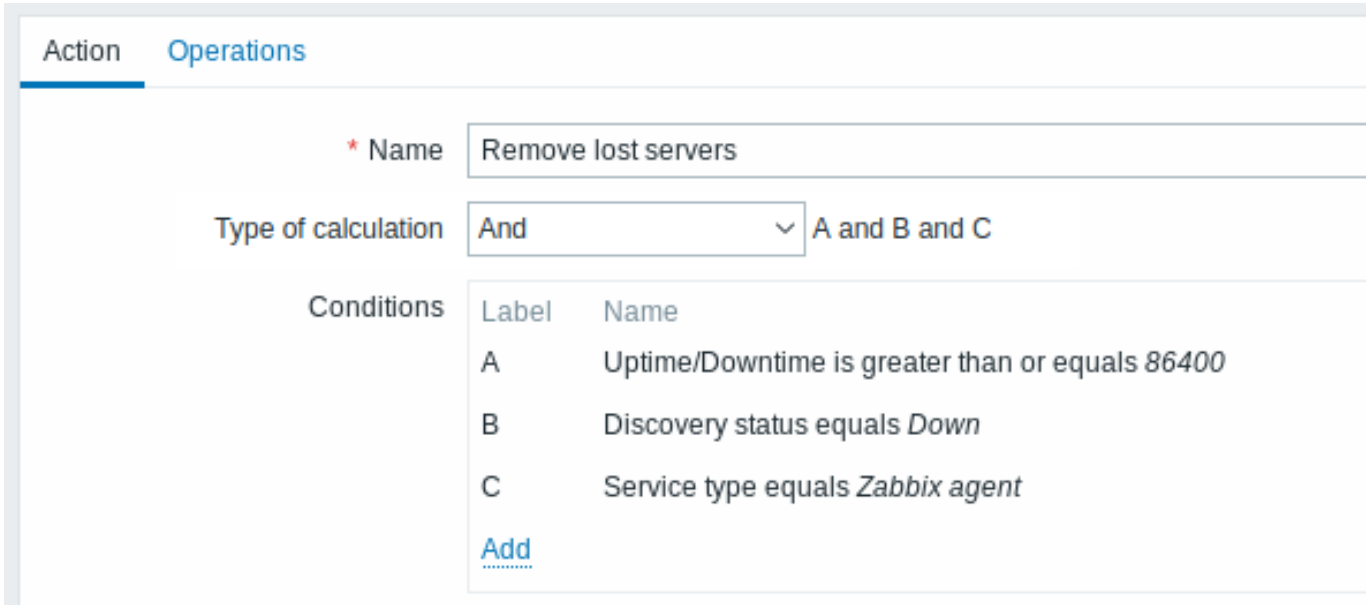
Krok 3

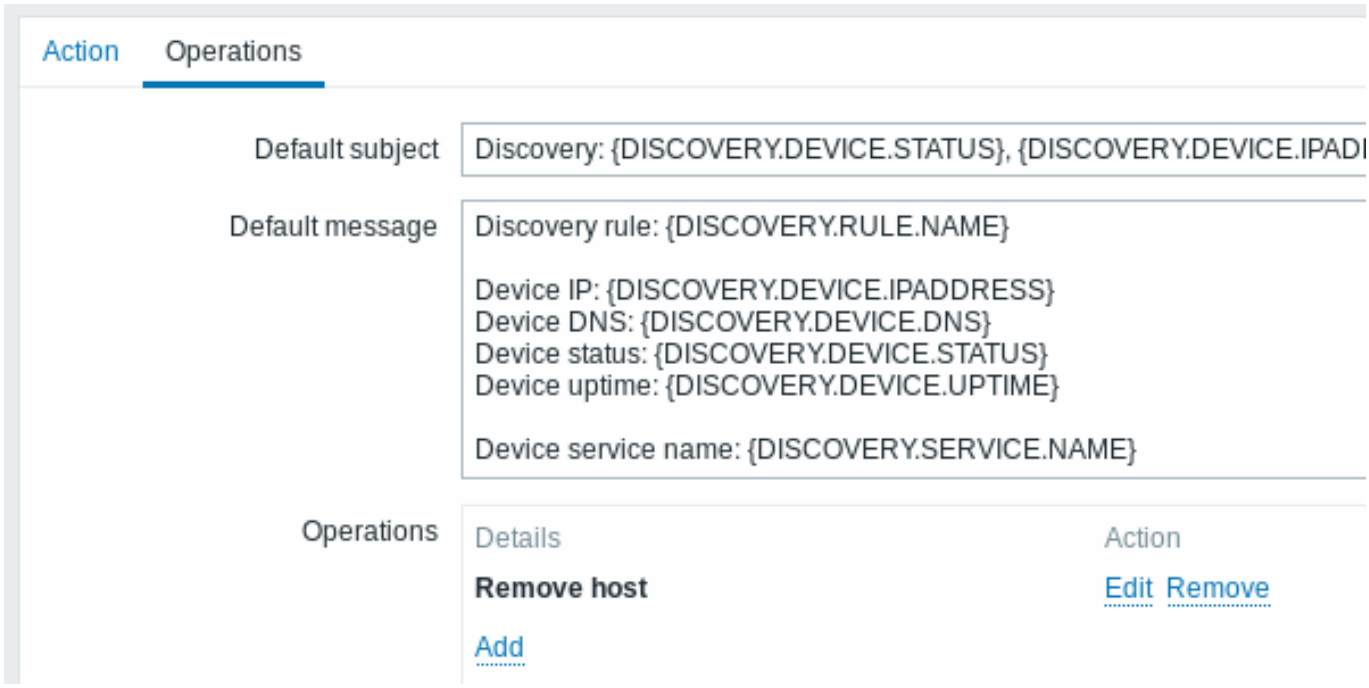
Definiowanie akcji wykrywania służącej do dodawania wykrytych serwerów Windows do odpowiedniej grupy/szablonu.



Krok 4

Definiowanie akcji wykrywania do usuwania utraconych serwerów.





Serwer zostanie usunięty, jeśli usługa „Zabbix agent” będzie w stanie „down” przez ponad 24 godziny (86400 sekund).

2 Autorejestracja agenta aktywnego

Przegląd

Możliwe jest zezwolenie na autorejestrację aktywnego agenta Zabbix, po której serwer może rozpocząć ich monitorowanie. W ten sposób nowe hosty mogą być dodawane do monitorowania bez ręcznej konfiguracji na serwerze.

Autorejestracja może nastąpić, gdy wcześniej nieznany aktywny agent poprosi o kontrole.

Funkcja ta może być bardzo przydatna do automatycznego monitorowania nowych węzłów Cloud. Gdy tylko w Cloud pojawi się nowy węzeł, Zabbix automatycznie rozpocznie zbieranie danych o wydajności i dostępności hosta.

Autorejestracja aktywnego agenta obsługuje również monitorowanie dodanych hostów za pomocą kontroli pasywnych. Gdy aktywny agent prosi o kontrole, pod warunkiem że ma zdefiniowane parametry konfiguracyjne `ListenIP` lub `ListenPort` w pliku konfiguracyjnym, są one przesyłane do serwera. Jeśli określono wiele adresów IP, do serwera wysyłany jest pierwszy z nich.

Serwer, dodając nowy autorejestrowany host, używa otrzymanego adresu IP i portu do skonfigurowania agenta. Jeśli nie zostanie odebrana żadna wartość adresu IP, używany jest adres wykorzystany dla połączenia przychodzącego. Jeśli nie zostanie odebrana żadna wartość portu, używany jest port 10050.

Możliwe jest określenie, że host powinien zostać autorejestrowany z nazwą DNS jako domyślnym interfejsem agenta.

Autorejestracja jest uruchamiana ponownie:

- jeśli zmienią się informacje o metadanych hosta:
  - z powodu zmiany HostMetadata i ponownego uruchomienia agenta
  - z powodu zmiany wartości zwracanej przez HostMetadataItem
- dla hostów utworzonych ręcznie, jeśli brakuje metadanych
- jeśli host zostanie ręcznie zmieniony tak, aby był monitorowany przez inny proxy Zabbix
- jeśli autorejestracja dla tego samego hosta pochodzi z nowego proxy Zabbix

Interwał heartbeat autorejestracji aktywnego agenta dla serwera Zabbix i proxy Zabbix wynosi 120 sekund. Dlatego w przypadku usunięcia wykrytego hosta autorejestracja zostanie uruchomiona ponownie po 120 sekundach.

Konfiguracja

Określ serwer

Upewnij się, że serwer Zabbix jest wskazany w pliku konfiguracyjnym agenta - `zabbix_agentd.conf`:

```
ServerActive=10.0.0.1
```

Jeśli nie zdefiniujesz jawnie parametru *Hostname* w `zabbix_agentd.conf`, serwer użyje nazwy systemowej hosta z lokalizacji agenta do nazwania hosta. Nazwę systemową hosta w systemie Linux można uzyskać, uruchamiając polecenie `hostname`.

Jeśli parametr *Hostname* jest zdefiniowany w konfiguracji agenta Zabbix jako lista hostów rozdzielona przecinkami, hosty zostaną utworzone dla wszystkich wymienionych nazw hostów.

Po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym uruchom ponownie agenta.

Akcja dla aktywnej autorejestracji agent

Gdy serwer otrzyma żądanie autorejestracji od agent, wywołuje akcję. Dla autorejestracji agent musi być skonfigurowana akcja ze źródłem zdarzeń „Autorejestracja".

Note:
Skonfigurowanie wykrywania sieci nie jest wymagane, aby aktywne agenty mogły się autorejestrować.

W frontend Zabbix przejdź do *Alerty* → *Akcje*, wybierz *Akcje autorejestracji* i kliknij *Utwórz akcję*:

- Na karcie Akcja nadaj swojej akcji nazwę
- Opcjonalnie określ warunki. Możesz użyć dopasowania podciągu lub wyrażenia regularnego w warunkach dla nazwy hosta/metadanych hosta. Jeśli zamierzasz użyć warunku „Metadane hosta", zobacz następną sekcję.
- Na karcie Operacje dodaj odpowiednie operacje, takie jak: „Dodaj host", „Dodaj do grupy hostów" (na przykład *Wykryte hosty*), „Podłącz szablony" itd.

Note:
Jeśli hosty, które będą się autorejestrować, prawdopodobnie będą obsługiwane wyłącznie przez aktywne monitorowanie (na przykład hosty odgrodzone zaporą od serwera Zabbix), możesz utworzyć specjalny szablon, taki jak *Template_Linux-active*, do podłączenia.

Utworzone hosty są dodawane do grupy *Wykryte hosty* (domyślnie; można to skonfigurować w *Administracja* > *Ogólne* > *Inne*). Jeśli chcesz, aby hosty były dodawane do innej grupy, dodaj operację *Usuń z grupy hostów* (wskazując „Wykryte hosty"), a także operację *Dodaj do grupy hostów* (wskazując inną grupę hostów), ponieważ host musi należeć do grupy hostów.

Bezpieczna autorejestracja

Bezpieczny sposób autorejestracji jest możliwy dzięki skonfigurowaniu uwierzytelniania opartego na PSK z szyfrowanymi połączeniami.

Poziom szyfrowania jest konfigurowany globalnie w *Administracja* > *Ogólne* > *Autorejestracja*. Można wybrać brak szyfrowania, szyfrowanie TLS z uwierzytelnianiem PSK lub obie opcje jednocześnie (tak aby niektóre hosty mogły rejestrować się bez szyfrowania, a inne z użyciem szyfrowania).

Uwierzytelnianie za pomocą PSK jest weryfikowane przez serwer Zabbix przed dodaniem hosta. Jeśli zakończy się powodzeniem, host zostanie dodany, a ustawienie *Połączenia z/do hosta* zostanie ustawione wyłącznie na „PSK", z tożsamością/kluczem współdzielonym takim samym jak w globalnym ustawieniu autorejestracji.

Attention:
Aby zapewnić bezpieczeństwo autorejestracji w instalacjach korzystających z proxy, należy włączyć szyfrowanie między serwerem Zabbix a proxy.

Używanie DNS jako domyślnego interfejsu

Parametry konfiguracyjne HostInterface i HostInterfaceItem configuration parameters umożliwiają określenie niestandardowej wartości interfejsu hosta podczas autorejestracji.

Dokładniej mówiąc, są one przydatne, jeśli host ma zostać automatycznie zarejestrowany z nazwą DNS jako domyślnym interfejsem agent zamiast jego adresu IP. W takim przypadku nazwa DNS powinna zostać określona lub zwrócona jako wartość parametru HostInterface albo HostInterfaceItem. Jeśli wartość jednego z tych parametrów zmieni się — na przykład z adresu IP na nazwę DNS lub odwrotnie — domyślny interfejs automatycznie zarejestrowanego hosta zostanie odpowiednio zaktualizowany. Ta aktualizacja jest stosowana do istniejącego hosta, a nie przez utworzenie nowego. Aby wysłać nową wartość, agent musi zostać uruchomiony ponownie, tak aby ponownie zainicjował proces autorejestracji.

Note:
Jeśli parametry HostInterface lub HostInterfaceItem nie są skonfigurowane, zamiast nich używany jest parametr listen_dns. Ta wartość jest określana przez wykonanie odwrotnego wyszukiwania DNS dla adresu IP agenta. Jeśli odwrotne DNS nie jest poprawnie skonfigurowane lub zwraca nieprawidłową nazwę, może to skutkować nieprawidłową lub nieudaną autorejestracją z powodu nieprawidłowej wartości interfejsu.

Używanie metadanych hosta

Gdy agent wysyła do serwera żądanie autorejestracji, przesyła swoją nazwę hosta. W niektórych przypadkach (na przykład węzły chmury Amazon) nazwa hosta nie wystarcza, aby serwer Zabbix mógł rozróżnić wykryte hosty. Metadane hosta mogą być opcjonalnie używane do przesyłania innych informacji z agenta do serwera.

Metadane hosta konfiguruje się w pliku konfiguracyjnym agenta - `zabbix_agentd.conf`. Istnieją 2 sposoby określenia metadanych hosta w pliku konfiguracyjnym:

```
HostMetadata
HostMetadataItem
```

Opis tych opcji znajduje się pod powyższym linkiem.

Parametr HostMetadataItem może zwrócić maksymalnie 65535 punktów kodowych UTF-8. Dłuższa wartość zostanie obcięta.

Należy pamiętać, że w MySQL efektywna maksymalna długość w znakach będzie mniejsza, jeśli zwracana wartość zawiera znaki wielobajtowe. Na przykład wartość zawierająca wyłącznie znaki 3-bajtowe będzie ograniczona łącznie do 21844 znaków, natomiast wartość zawierająca wyłącznie znaki 4-bajtowe będzie ograniczona do 16383 znaków.

Attention:
Próba autorejestracji następuje za każdym razem, gdy aktywny agent wysyła do serwera żądanie odświeżenia aktywnych kontroli. Opóźnienie między żądaniami jest określone w parametrze RefreshActiveChecks agenta. Pierwsze żądanie jest wysyłane natychmiast po ponownym uruchomieniu agenta.

Przykłady

Autorejestracja według systemu operacyjnego przy użyciu HostMetadata

Załóżmy, że chcesz, aby hosty były automatycznie rejestrowane przez serwer Zabbix. W swojej sieci masz aktywne agenty Zabbix (zobacz sekcję „Configuration" powyżej). W sieci znajdują się hosty z systemem Windows i Linux, a w frontendzie Zabbix masz dostępne szablony „Linux by Zabbix agent" i „Windows by Zabbix agent". Dlatego podczas rejestracji hosta chcesz, aby do rejestrowanego hosta został zastosowany odpowiedni szablon Linux/Windows. Domyślnie podczas autorejestracji do serwera wysyłana jest tylko nazwa hosta, co może nie być wystarczające. Aby mieć pewność, że do hosta zostanie zastosowany właściwy szablon, należy użyć metadanych hosta.

Konfiguracja frontendu

Pierwszą rzeczą, którą należy zrobić, jest skonfigurowanie frontendu. Utwórz 2 akcje. Pierwsza akcja:

- Nazwa: Automatyczna rejestracja hosta Linux
- Warunki: Metadane hosta zawierają *Linux*
- Operacje: Podłącz szablony: Linux by Zabbix agent

Note:
W tym przypadku można pominąć operację „Dodaj host”. Podłączenie szablonu do hosta wymaga najpierw dodania hosta, więc serwer zrobi to automatycznie.

Druga akcja:

- Nazwa: Automatyczna rejestracja hosta Windows
- Warunki: Metadane hosta zawierają *Windows*
- Operacje: Podłącz szablony: Windows by Zabbix agent

Konfiguracja agenta

Teraz należy skonfigurować agenty. Dodaj następujący wiersz do plików konfiguracyjnych agenta:

```
HostMetadataItem=system.uname
```

W ten sposób masz pewność, że metadane hosta będą zawierać „Linux” lub „Windows” w zależności od hosta, na którym uruchomiony jest agent. Przykład metadanych hosta w tym przypadku:

```
Linux: Linux server3 3.2.0-4-686-pae #1 SMP Debian 3.2.41-2 i686 GNU/Linux
Windows: Windows WIN-0PXGGSTYNHO 6.0.6001 Windows Server 2008 Service Pack 1 Intel IA-32
```

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

Używanie HostMetadata do kontrolowania autorejestracji i szablonów

Krok 1 — Chroń rejestrację za pomocą HostMetadata

Użycie metadanych hosta pozwala zapewnić podstawową ochronę przed rejestracją niepożądanych hostów.

Konfiguracja frontendu

Utwórz akcję we frontendzie, używając trudnego do odgadnięcia tajnego kodu, aby nie dopuścić niepożądanych hostów:

- Nazwa: Akcja autorejestracji Linux
- Warunki:
  - Typ obliczeń: AND
  - Warunek (A): Metadane hosta zawierają //Linux//
  - Warunek (B): Metadane hosta zawierają //21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae//
- Operacje:
  - Wyślij wiadomość do użytkowników: Admin przez wszystkie media
  - Dodaj do grup hostów: Serwery Linux
  - Podłącz szablony: Linux by Zabbix agent

Należy pamiętać, że sama ta metoda nie zapewnia silnej ochrony, ponieważ dane są przesyłane otwartym tekstem. Aby zmiany zaczęły obowiązywać natychmiast, wymagane jest przeładowanie pamięci podręcznej konfiguracji.

Konfiguracja agenta

Dodaj następujący wiersz do pliku konfiguracyjnego agenta:

```
HostMetadata=Linux    21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

gdzie „Linux” to platforma, a pozostała część ciągu to trudny do odgadnięcia tajny tekst.

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

Krok 2 — Dodaj szablon do zarejestrowanego hosta

Możliwe jest dodanie dodatkowych szablonów do już zarejestrowanego hosta. W takim przypadku szablon MySQL by Zabbix agent zostanie podłączony tylko do hostów, których HostMetadata zawiera token `MySQL`.

Konfiguracja frontend

Zaktualizuj akcję we frontend:

- Nazwa: Akcja autorejestracji Linux
- Warunki:
  - Typ obliczania: AND
  - Warunek (A): Metadane host zawierają Linux
  - Warunek (B): Metadane host zawierają 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
  - Warunek (C): Metadane host zawierają MySQL
- Operacje:
  - Wyślij wiadomość do użytkowników: Admin przez wszystkie media

- Dodaj do grup hostów: Serwery Linux
- Podłącz szablony: Linux by Zabbix agent
- Podłącz szablony: MySQL by Zabbix Agent

Konfiguracja agenta

Zaktualizuj następną linię w pliku konfiguracyjnym agenta:

```
HostMetadata=MySQL on Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

3 Odkrywanie niskiego poziomu

Przegląd Odkrywanie niskiego poziomu (LLD) zapewnia sposób automatycznego tworzenia pozycji, wyzwalaczy i wykresów dla różnych encji na hoście. Na przykład Zabbix może automatycznie rozpocząć monitorowanie systemów plików lub interfejsów sieciowych na twojej maszynie, bez konieczności ręcznego tworzenia pozycji dla każdego systemu plików lub interfejsu sieciowego. LLD może również tworzyć hosty, na przykład w celu uzupełnienia maszyn wirtualnych wykrytych na hiperwizorze, a także zagnieżdżone reguły odkrywania, umożliwiające wielopoziomowe odkrywanie. Dodatkowo możliwe jest skonfigurowanie Zabbix tak, aby automatycznie usuwał niepotrzebne encje na podstawie rzeczywistych wyników okresowo wykonywanego odkrywania.

Użytkownik może zdefiniować własne typy odkrywania, pod warunkiem że są zgodne z określonym protokołem JSON.

Ogólna architektura procesu odkrywania wygląda następująco.

Najpierw użytkownik tworzy regułę odkrywania w *Data collection* > *Templates*, w kolumnie *Discovery*. Reguła odkrywania składa się z (1) pozycji, która wykrywa wymagane encje (na przykład systemy plików lub interfejsy sieciowe), oraz (2) prototypów pozycji, wyzwalaczy i wykresów, które mają zostać utworzone na podstawie wartości tej pozycji.

Pozycja wykrywająca wymagane encje jest podobna do zwykłej pozycji spotykanej w innych miejscach: serwer pyta agent Zabbix (lub inny komponent, zależnie od ustawionego typu pozycji) o wartość tej pozycji, agent odpowiada wartością tekstową. Różnica polega na tym, że wartość zwracana przez agent powinna zawierać listę wykrytych encji w formacie JSON. Chociaż szczegóły tego formatu są istotne tylko dla implementujących własne kontrole odkrywania, warto wiedzieć, że zwracana wartość zawiera listę par makro → wartość. Na przykład pozycja ”net.if.discovery” może zwrócić dwie pary: ”{#IFNAME}” → ”lo” oraz ”{#IFNAME}” → ”eth0”.

Te makra są używane w nazwach, kluczach i innych polach prototypów, gdzie są następnie zastępowane otrzymanymi wartościami w celu utworzenia rzeczywistych pozycji, wyzwalaczy, wykresów, a nawet hostów dla każdej wykrytej encji. Zobacz pełną listę opcji użycia makr LLD.

Gdy serwer otrzyma wartość dla pozycji odkrywania, analizuje pary makro → wartość i dla każdej pary generuje rzeczywiste pozycje, wyzwalacze i wykresy na podstawie ich prototypów. W powyższym przykładzie z ”net.if.discovery” serwer wygenerowałby jeden zestaw pozycji, wyzwalaczy i wykresów dla interfejsu pętli zwrotnej ”lo” oraz drugi zestaw dla interfejsu ”eth0”.

Należy pamiętać, że od **Zabbix 4.2** format JSON zwracanego przez reguły odkrywania niskiego poziomu został zmieniony. Nie oczekuje się już, że JSON będzie zawierał obiekt ”data”. Odkrywanie niskiego poziomu akceptuje teraz zwykły JSON zawierający tablicę, aby obsługiwać nowe funkcje, takie jak wstępne przetwarzanie wartości pozycji oraz niestandardowe ścieżki do wartości makr odkrywania niskiego poziomu w dokumencie JSON.

Wbudowane klucze odkrywania zostały zaktualizowane tak, aby zwracały tablicę wierszy LLD w korzeniu dokumentu JSON. Zabbix automatycznie wyodrębni makro i wartość, jeśli pole tablicy używa składni {#MACRO} jako klucza. Wszelkie nowe natywne kontrole odkrywania będą używać nowej składni bez elementów ”data”. Podczas przetwarzania wartości odkrywania niskiego poziomu najpierw lokalizowany jest korzeń (tablica pod `$.` lub `$.data`).

Chociaż element ”data” został usunięty ze wszystkich natywnych pozycji związanych z odkrywaniem, dla zachowania zgodności wstecznej Zabbix nadal będzie akceptował notację JSON z elementem ”data”, choć jego użycie nie jest zalecane. Jeśli JSON zawiera obiekt tylko z jednym elementem tablicy ”data”, zawartość tego elementu zostanie automatycznie wyodrębniona przy użyciu JSONPath `$.data`. Odkrywanie niskiego poziomu akceptuje teraz opcjonalne makra LLD zdefiniowane przez użytkownika z niestandardową ścieżką określoną w składni JSONPath.

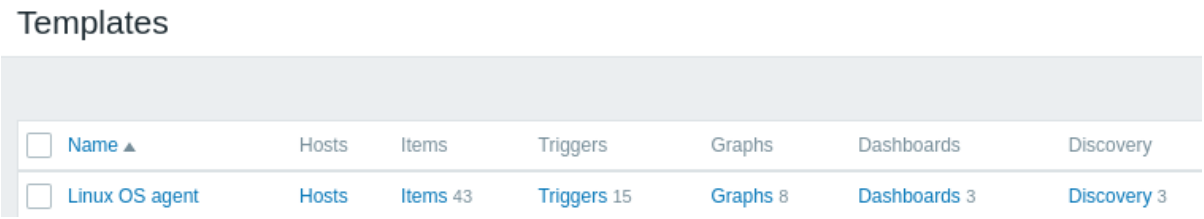
> **Warning:**
> W rezultacie powyższych zmian nowsze agenty nie będą już mogły współpracować ze starszym serwerem Zabbix.

Zobacz także: Wykryte encje

Konfigurowanie wykrywania niskiego poziomu Przedstawimy wykrywanie niskiego poziomu na przykładzie wykrywania systemów plików.

Aby skonfigurować wykrywanie, wykonaj następujące czynności:

- Przejdź do: *Zbieranie danych* > *Szablony* lub *Hosty*.
- Kliknij *Wykrywanie* w wierszu odpowiedniego szablonu/hosta.



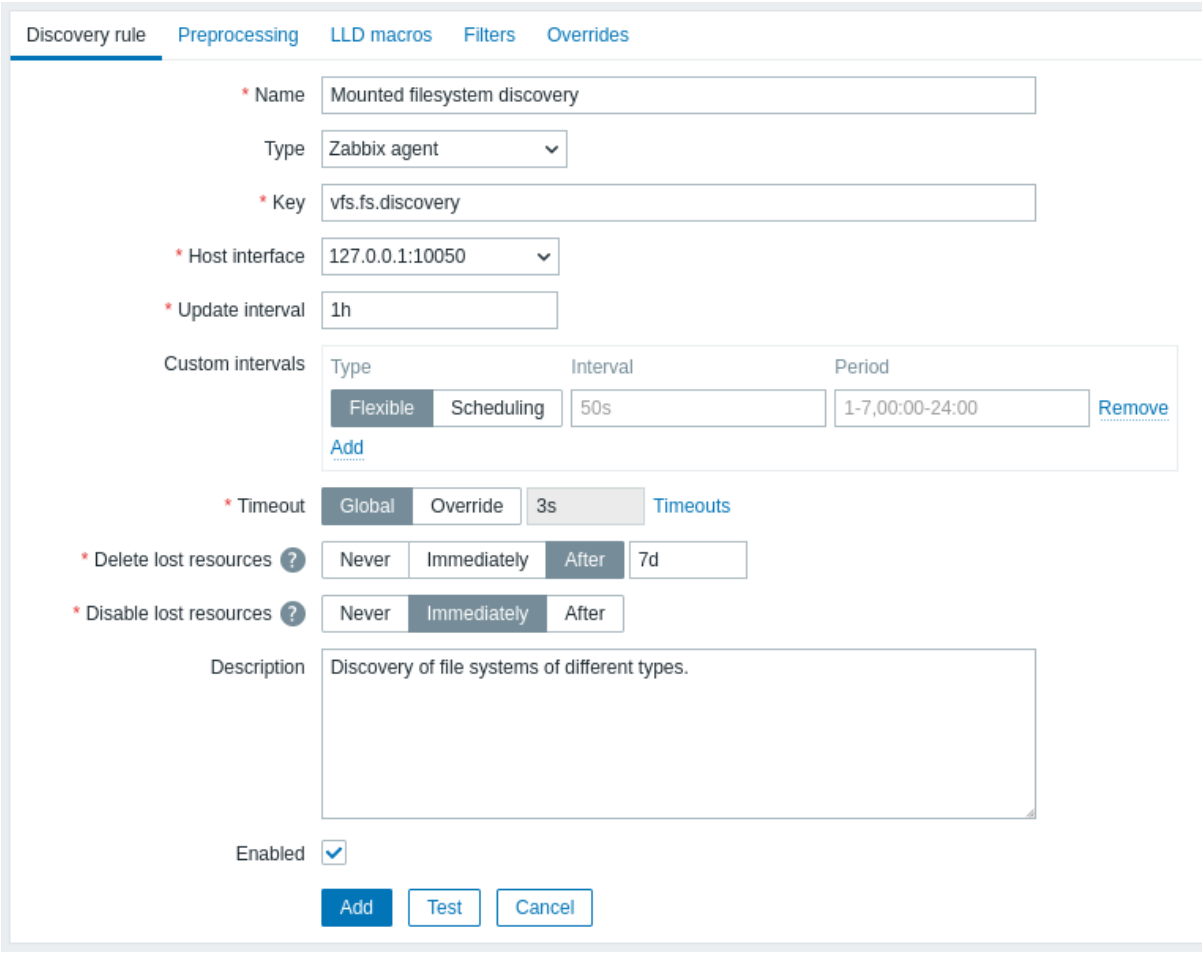
- Kliknij *Utwórz regułę wykrywania* w prawym górnym rogu ekranu.
- Uzupełnij formularz reguły wykrywania, podając wymagane informacje.

Reguła wykrywania

Formularz reguły wykrywania zawiera pięć kart, które od lewej do prawej przedstawiają przepływ danych podczas wykrywania:

- *Reguła wykrywania* — określa przede wszystkim wbudowaną pozycję lub niestandardowy skrypt służący do pobierania danych wykrywania.
- *Przetwarzanie wstępne* — stosuje przetwarzanie wstępne do wykrytych danych.
- *Makra LLD* — umożliwia wyodrębnienie wartości makr do użycia w wykrytych pozycjach, wyzwalaczach itp.
- *Filtry* — umożliwia filtrowanie wykrytych wartości.
- *Nadpisania* — umożliwia modyfikację pozycji, wyzwalaczy, wykresów lub prototypów hostów podczas stosowania do określonych wykrytych obiektów.

Karta **Reguła wykrywania** zawiera klucz pozycji używany do wykrywania (a także niektóre ogólne atrybuty reguły wykrywania):



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa reguły wykrywania.
Type	Typ kontroli używany do wykrywania. W tym przykładzie używany jest typ pozycji *Zabbix agent*. Reguła wykrywania może być także pozycją zależną, zależną od zwykłej pozycji. Nie może zależeć od innej reguły wykrywania. W przypadku pozycji zależnej wybierz odpowiedni typ (*Dependent item*) i określ pozycję nadrzędną w polu 'Master item'. Pozycja nadrzędna musi istnieć.
Key	Wprowadź klucz pozycji wykrywania (do 2048 znaków). Na przykład możesz użyć wbudowanego klucza pozycji "vfs.fs.discovery", aby zwrócić ciąg JSON z listą systemów plików obecnych na komputerze, ich typami i opcjami montowania. Zwróć uwagę, że inną opcją wykrywania systemów plików jest użycie wyników wykrywania przez klucz agenta "vfs.fs.get" (zobacz przykład).
Update interval	To pole określa, jak często Zabbix wykonuje wykrywanie. Na początku, gdy dopiero konfigurujesz wykrywanie systemów plików, możesz chcieć ustawić mały interwał, ale gdy już wiesz, że działa poprawnie, możesz ustawić 30 minut lub więcej, ponieważ systemy plików zwykle nie zmieniają się zbyt często. Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: Interwał aktualizacji można ustawić na '0' tylko wtedy, gdy istnieją niestandardowe interwały z wartością różną od zera. Jeśli ustawiono '0' i istnieje niestandardowy interwał (elastyczny lub harmonogram) z wartością różną od zera, pozycja będzie odpytywana w czasie trwania niestandardowego interwału. Nowe reguły wykrywania będą sprawdzane w ciągu 60 sekund od ich utworzenia, chyba że mają Harmonogram lub Elastyczny interwał aktualizacji, a *Update interval* jest ustawiony na 0. *Uwaga*, że dla istniejącej reguły wykrywania wykrywanie można wykonać natychmiast, naciskając przycisk *Execute now*.
Custom intervals	Możesz utworzyć niestandardowe reguły sprawdzania pozycji: **Flexible** — utwórz wyjątek od *Update interval* (interwał z inną częstotliwością) **Scheduling** — utwórz niestandardowy harmonogram odpytywania. Szczegółowe informacje znajdują się w sekcji Custom intervals.
Timeout	Ustaw limit czasu kontroli wykrywania. Wybierz opcję limitu czasu: **Global** — używany jest limit czasu proxy/globalny (wyświetlany w wyszarzonym polu *Timeout*); **Override** — używany jest niestandardowy limit czasu (ustawiany w polu *Timeout*; dozwolony zakres: 1 - 600s). Obsługiwane są sufiksy czasu, np. 30s, 1m, oraz makra użytkownika. Kliknięcie odnośnika *Timeouts* umożliwia skonfigurowanie limitów czasu proxy lub limitów czasu globalnych (jeśli proxy nie jest używany). Zwróć uwagę, że odnośnik *Timeouts* jest widoczny tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administration > General* lub *Administration > Proxies*.
Delete lost resources	Określ, jak szybko wykryta jednostka zostanie usunięta, gdy jej status wykrywania zmieni się na "Not discovered anymore": *Never* — nie zostanie usunięta; *Immediately* — zostanie usunięta natychmiast; *After* — zostanie usunięta po określonym czasie. Wartość musi być większa niż wartość *Disable lost resources*. Obsługiwane są sufiksy czasu, np. 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: Użycie opcji "Immediately" nie jest zalecane, ponieważ nawet błędna edycja filtra może spowodować usunięcie jednostki wraz z wszystkimi danymi historycznymi. Zwróć uwagę, że zasoby wyłączone ręcznie nie zostaną usunięte przez wykrywanie niskiego poziomu.
Disable lost resources	Określ, jak szybko wykryta jednostka zostanie wyłączona, gdy jej status wykrywania zmieni się na "Not discovered anymore": *Never* — nie zostanie wyłączona; *Immediately* — zostanie wyłączona natychmiast; *After* — zostanie wyłączona po określonym czasie. Wartość powinna być większa niż interwał aktualizacji reguły wykrywania. Zwróć uwagę, że zasoby wyłączone automatycznie zostaną ponownie włączone, jeśli zostaną ponownie wykryte przez wykrywanie niskiego poziomu. Zasoby wyłączone ręcznie nie zostaną ponownie włączone po ponownym wykryciu. To pole nie jest wyświetlane, jeśli *Delete lost resources* jest ustawione na "Immediately". Obsługiwane są sufiksy czasu, np. 2h, 1d. Obsługiwane są makra użytkownika.

Parameter	Description
Description	Wprowadź opis.
Enabled	Jeśli zaznaczone, reguła będzie przetwarzana.

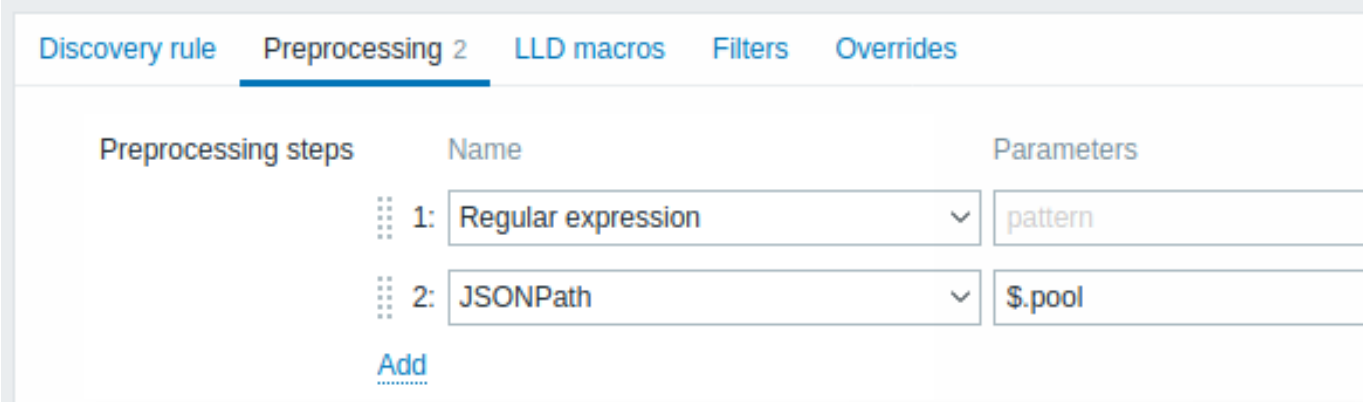
Note:
Historia reguły wykrywania nie jest zachowywana.

Preprocessing

Zakładka **Preprocessing** pozwala zdefiniować reguły transformacji stosowane do wyniku discovery. Na tym etapie możliwa jest jedna lub kilka transformacji. Transformacje są wykonywane w kolejności, w jakiej zostały zdefiniowane. Całe preprocessing jest wykonywane przez serwer Zabbix.

Zobacz także:

- Szczegóły preprocessing
- Testowanie preprocessing



Type		
	Transformation	Description
Text		
	Regular expression	Dopasuj otrzymaną wartość do wyrażenia regularnego <pattern> i zastąp wartość wyodrębnionym <output>. Wyrażenie regularne obsługuje wyodrębnianie maksymalnie 10 przechwyconych grup za pomocą sekwencji \N. Parameters: **pattern** - wyrażenie regularne **output** - szablon formatowania wyjścia. Sekwencja ucieczki \N (gdzie N=1...9) jest zastępowana przez N-tą dopasowaną grupę. Sekwencja ucieczki \0 jest zastępowana przez dopasowany tekst. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
	Replace	Znajdź ciąg wyszukiwania i zastąp go innym (lub niczym). Wszystkie wystąpienia ciągu wyszukiwania zostaną zastąpione. Parameters: **search string** - ciąg do znalezienia i zastąpienia, z uwzględnieniem wielkości liter (wymagane) **replacement** - ciąg, którym należy zastąpić ciąg wyszukiwania. Ciąg zastępczy może być również pusty, co w praktyce pozwala usunąć znaleziony ciąg. Można używać sekwencji ucieczki do wyszukiwania lub zastępowania znaków nowej linii, powrotu karetki, tabulatorów i spacji ”\n \r \t \s”; znak ukośnika odwrotnego można zapisać jako ”\\” a sekwencje ucieczki można zapisać jako ”\\n”. Ucieczka znaków nowej linii, powrotu karetki i tabulatorów jest wykonywana automatycznie podczas low-level discovery.
Structured data		
	JSONPath	Wyodrębnij wartość lub fragment z danych JSON za pomocą funkcjonalności JSONPath. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.

Type	

XML XPath	Wyodrębnij wartość lub fragment z danych XML za pomocą funkcjonalności XPath.
	Aby ta opcja działała, serwer Zabbix musi być skompilowany z obsługą libxml.
	Przykłady:
	`number(/document/item/value)` wyodrębni 10 z `<document><item><value>10</value></item></document>`
	`number(/document/item/@attribute)` wyodrębni 10 z `<document><item attribute="10"></item></document>`
	`/document/item` wyodrębni `<item><value>10</value></item>` z `<document><item><value>10</value></item></document>`
	Uwaga: przestrzenie nazw nie są obsługiwane.
	Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.

CSV to JSON	Konwertuj dane pliku CSV do formatu JSON. Więcej informacji: CSV to JSON preprocessing.
XML to JSON	Konwertuj dane w formacie XML do JSON. Więcej informacji: Zasady serializacji. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
SNMP	
SNMP walk value	Wyodrębnij wartość dla określonej nazwy OID/MIB i zastosuj opcje formatowania: **Unchanged** - zwróć Hex-STRING jako niezmieniony ciąg szesnastkowy (*uwaga*: wskazówki wyświetlania są nadal stosowane); **UTF-8 from Hex-STRING** - konwertuj Hex-STRING do ciągu UTF-8; **MAC from Hex-STRING** - konwertuj Hex-STRING do ciągu adresu MAC (w którym `' '` zostanie zastąpione przez `':'`); **Integer from BITS** - konwertuj pierwsze 8 bajtów ciągu bitowego wyrażonego jako sekwencja znaków szesnastkowych (np. ”1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitowych dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
SNMP walk to JSON	Konwertuj wartości SNMP do JSON. Określ nazwę pola w JSON oraz odpowiadającą jej ścieżkę SNMP OID. Wartości pól zostaną wypełnione wartościami z określonej ścieżki SNMP OID. Ten krok preprocessing można wykorzystać do SNMP OID discovery. Dostępne są podobne opcje formatowania wartości jak w kroku *SNMP walk value*. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
SNMP get value	Zastosuj opcje formatowania do wartości SNMP get: **UTF-8 from Hex-STRING** - konwertuj Hex-STRING do ciągu UTF-8; **MAC from Hex-STRING** - konwertuj Hex-STRING do ciągu adresu MAC (w którym `' '` zostanie zastąpione przez `':'`); **Integer from BITS** - konwertuj pierwsze 8 bajtów ciągu bitowego wyrażonego jako sekwencja znaków szesnastkowych (np. ”1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitowych dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
Custom scripts	
JavaScript	Wprowadź kod JavaScript w edytorze modalnym, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Pamiętaj, że dostępna długość JavaScript zależy od używanej bazy danych. Więcej informacji: Javascript preprocessing
Validation	
Does not match regular expression	Określ wyrażenie regularne, z którym wartość nie może się zgadzać. Np. `Error:(.*?)\.` Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.

Type	
Check for error in JSON	Sprawdź komunikat o błędzie na poziomie aplikacji znajdujący się pod JSONPath. Zatrzymaj przetwarzanie, jeśli operacja się powiedzie, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością, która była przed tym krokiem preprocessing. Pamiętaj, że te błędy usług zewnętrznych są zgłaszane użytkownikowi bez zmian, bez dodawania informacji o kroku preprocessing. Np. `$.errors`. Jeśli zostanie odebrany JSON taki jak `{"errors":"e1"}`, następny krok preprocessing nie zostanie wykonany. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
Check for error in XML	Sprawdź komunikat o błędzie na poziomie aplikacji znajdujący się pod Xpath. Zatrzymaj przetwarzanie, jeśli operacja się powiedzie, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością, która była przed tym krokiem preprocessing. Pamiętaj, że te błędy usług zewnętrznych są zgłaszane użytkownikowi bez zmian, bez dodawania informacji o kroku preprocessing. W przypadku niepowodzenia parsowania nieprawidłowego XML nie zostanie zgłoszony żaden błąd. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
Matches regular expression	Określ wyrażenie regularne, z którym wartość musi się zgadzać. Jeśli zaznaczysz pole wyboru *Custom on fail*, możliwe jest określenie niestandardowych opcji obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.
Throttling	
Discard unchanged with heartbeat	Odrzuć wartość, jeśli nie zmieniła się w zdefiniowanym okresie czasu (w sekundach). Obsługiwane są dodatnie wartości całkowite określające liczbę sekund (minimum - 1 sekunda). W tym polu można używać sufiksów czasu (np. 30s, 1m, 2h, 1d). W tym polu można używać makr użytkownika i makr low-level discovery. Dla jednej pozycji discovery można określić tylko jedną opcję throttling. Np. `1m`. Jeśli identyczny tekst zostanie przekazany do tej reguły dwa razy w ciągu 60 sekund, zostanie odrzucony. *Uwaga*: Zmiana prototypów pozycji nie resetuje throttling. Throttling jest resetowany tylko wtedy, gdy zmieniane są kroki preprocessing.
Prometheus	
Prometheus to JSON	Konwertuj wymagane metryki Prometheus do JSON. Więcej szczegółów: Sprawdzenia Prometheus.

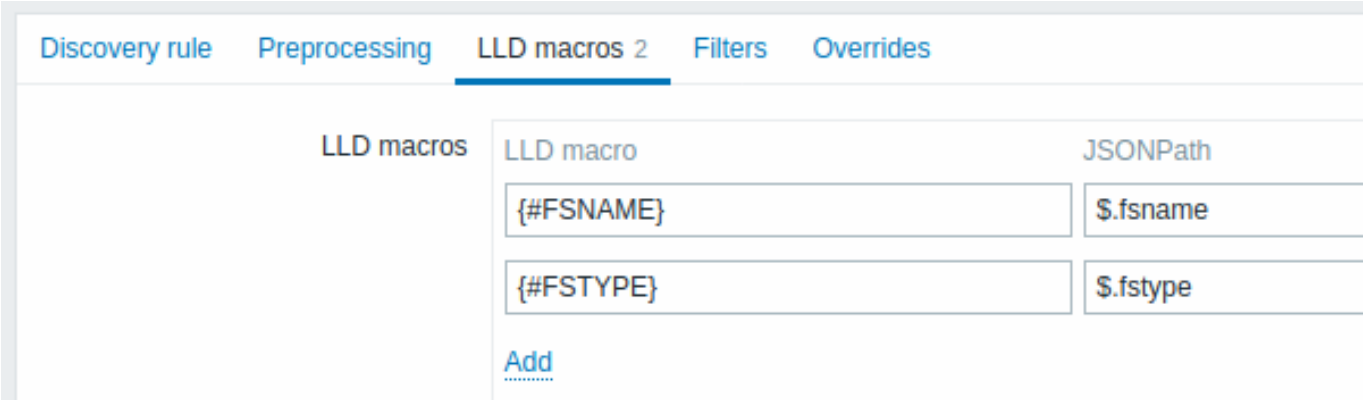
Pamiętaj, że jeśli reguła discovery została zastosowana do hosta za pośrednictwem szablonu, zawartość tej zakładki jest tylko do odczytu.

Makra niestandardowe

Zakładka **Makra LLD** umożliwia określenie niestandardowych makr wykrywania niskiego poziomu.

Makra niestandardowe są przydatne w przypadkach, gdy zwrócony JSON nie ma już zdefiniowanych wymaganych makr. Na przykład:

- Natywny klucz `vfs.fs.discovery` do wykrywania systemów plików zwraca JSON z pewnymi wstępnie zdefiniowanymi makrami LLD, takimi jak {#FSNAME}, {#FSTYPE}. Te makra mogą być używane bezpośrednio w prototypach pozycji i wyzwalaczy (zobacz kolejne sekcje strony); definiowanie makr niestandardowych nie jest potrzebne;
- Pozycja agenta `vfs.fs.get` również zwraca JSON z danymi systemu plików, ale bez żadnych wstępnie zdefiniowanych makr LLD. W takim przypadku możesz samodzielnie zdefiniować makra i odwzorować je na wartości w JSON za pomocą JSONPath:

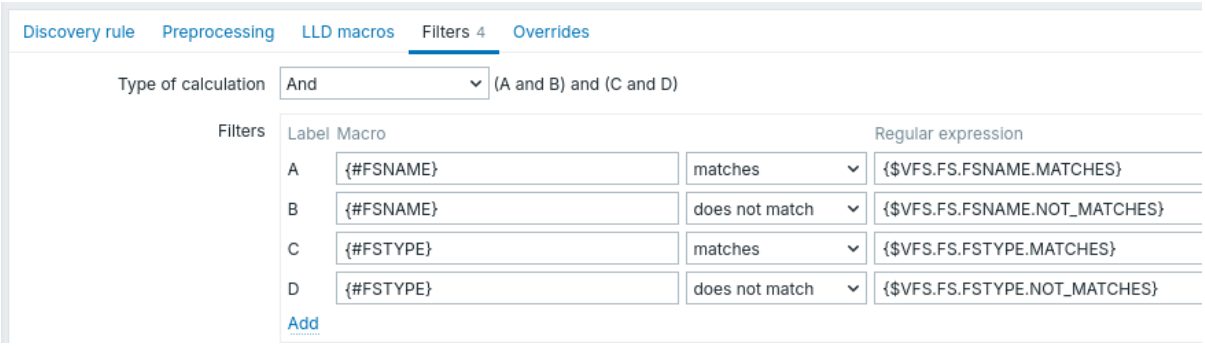


Wyodrębnione wartości mogą być używane w wykrytych pozycjach, wyzwalaczach itd. Pamiętaj, że wartości będą wyodrębniane z wyniku wykrywania oraz wszystkich dotychczasowych kroków przetwarzania wstępnego.

Parameter	Description
Makro LLD *JSONPath*	Nazwa makra wykrywania niskiego poziomu, z użyciem następującej składni: {#MACRO}. Ścieżka używana do wyodrębnienia wartości makra LLD z wiersza LLD, z użyciem składni JSONPath. Wartości wyodrębnione ze zwróconego JSON są używane do zastępowania makr LLD w polach prototypów pozycji, wyzwalaczy itd. JSONPath można określić przy użyciu notacji kropkowej lub notacji nawiasowej. Notacja nawiasowa powinna być używana w przypadku znaków specjalnych i Unicode, na przykład `$['unicode + special chars #1']['unicode + special chars #2']`. Na przykład `$.foo` wyodrębni „bar” i „baz” z tego JSON: `[{"foo":"bar"}, {"foo":"baz"}]` Zwróć uwagę, że `$.foo` wyodrębni „bar” i „baz” również z tego JSON: `{"data":[{"foo":"bar"}, {"foo":"baz"}]}` ponieważ pojedynczy obiekt „data” jest przetwarzany automatycznie (dla zgodności wstecznej z implementacją wykrywania niskiego poziomu w wersjach Zabbix wcześniejszych niż 4.2).

Filtr

Filtr może być użyty do generowania rzeczywistych pozycji, wyzwalaczy i wykresów tylko dla encji spełniających kryteria. Karta **Filtry** zawiera definicje filtrów reguły wykrywania, umożliwiające filtrowanie wartości wykrywania:



Parametr	Opis
Typ obliczania	Dostępne są następujące opcje obliczania filtrów: **And** - wszystkie filtry muszą zostać spełnione; **Or** - wystarczy, jeśli jeden filtr zostanie spełniony; **And/Or** - używa *And* dla różnych nazw makr i *Or* dla tej samej nazwy makra; **Custom expression** - oferuje możliwość zdefiniowania własnego sposobu obliczania filtrów. Formuła musi zawierać wszystkie filtry z listy. Ograniczenie do 255 znaków.

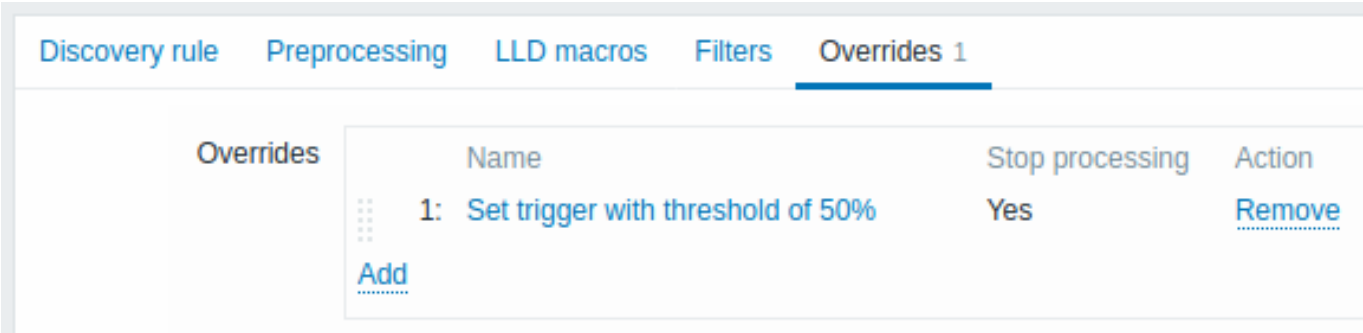
Parametr	Opis
Filtry	Dostępne są następujące operatory warunków filtra: *matches*, *does not match*, *exists*, *does not exist*. Operatory *matches* i *does not match* oczekują wyrażenia regularnego zgodnego z Perl (PCRE). Na przykład, jeśli interesują Cię tylko systemy plików C:, D: i E:, możesz wpisać {#FSNAME} w polu „Macro", a wyrażenie regularne "^C|^D|^E" w polu tekstowym „Regular expression". Filtrowanie jest również możliwe według typów systemów plików przy użyciu makra {#FSTYPE} (np. "^ext|^reiserfs") oraz według typów dysków (obsługiwane tylko przez agent Windows) przy użyciu makra {#FSDRIVETYPE} (np. "fixed"). W polu „Regular expression" można wprowadzić wyrażenie regularne lub odwołać się do globalnego wyrażenia regularnego. Aby przetestować wyrażenie regularne, można użyć „grep -E", na przykład: `for f in ext2 nfs reiserfs smbfs; do echo $f | grep -E '^ext|^reiserfs' || echo "SKIP: $f"; done` Operatory *exists* i *does not exist* umożliwiają filtrowanie encji na podstawie obecności lub braku określonego makra LLD w odpowiedzi. Zwróć uwagę, że jeśli makro z filtra nie występuje w odpowiedzi, znaleziona encja zostanie zignorowana, chyba że dla tego makra określono warunek „does not exist". Wyświetlone zostanie ostrzeżenie, jeśli brak makra wpływa na wynik wyrażenia. Na przykład, jeśli brakuje {#B} w: {#A} matches 1 and {#B} matches 2 - spowoduje ostrzeżenie {#A} matches 1 or {#B} matches 2 - bez ostrzeżenia

Warning:
Błąd lub literówka w wyrażeniu regularnym użytym w regule LLD (na przykład nieprawidłowe wyrażenie regularne „File systems for discovery") może spowodować usunięcie tysięcy elementów konfiguracji, wartości historycznych i zdarzeń dla wielu hostów.

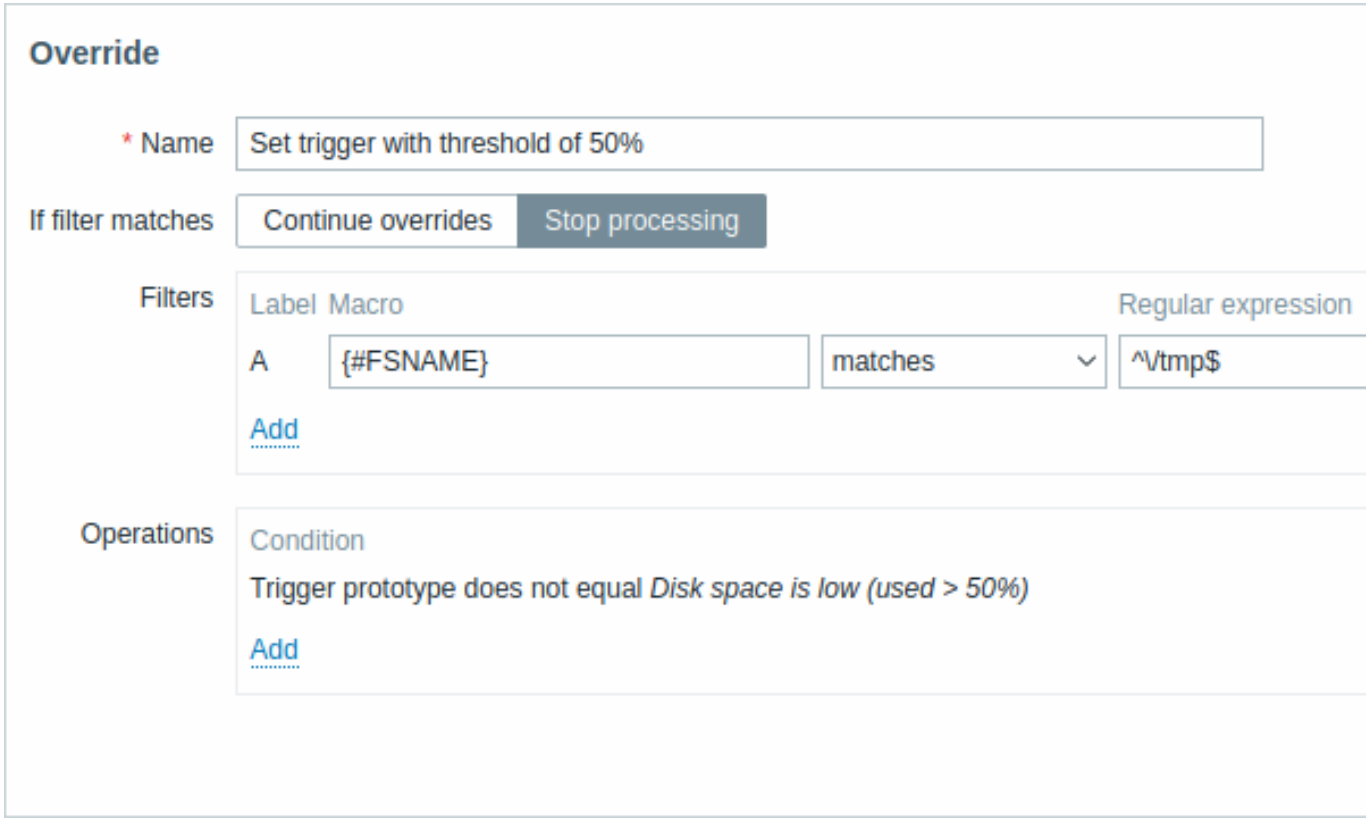
Attention:
Baza danych Zabbix w MySQL musi zostać utworzona z rozróżnianiem wielkości liter, jeśli nazwy systemów plików różniące się wyłącznie wielkością liter mają być wykrywane poprawnie.

Nadpisanie

Karta **Overrides** umożliwia definiowanie reguł modyfikujących listę prototypów item, trigger, graph, host i discovery lub ich atrybuty dla wykrytych obiektów spełniających podane kryteria.



Nadpisania, jeśli istnieją, są wyświetlane na liście typu drag-and-drop, którą można zmieniać kolejnością, i są wykonywane w kolejności, w jakiej zostały zdefiniowane. Aby skonfigurować szczegóły nowego nadpisania, kliknij Add w bloku *Overrides*. Aby edytować istniejące nadpisanie, kliknij jego nazwę. Otworzy się okno popup umożliwiające edycję szczegółów reguły nadpisania.

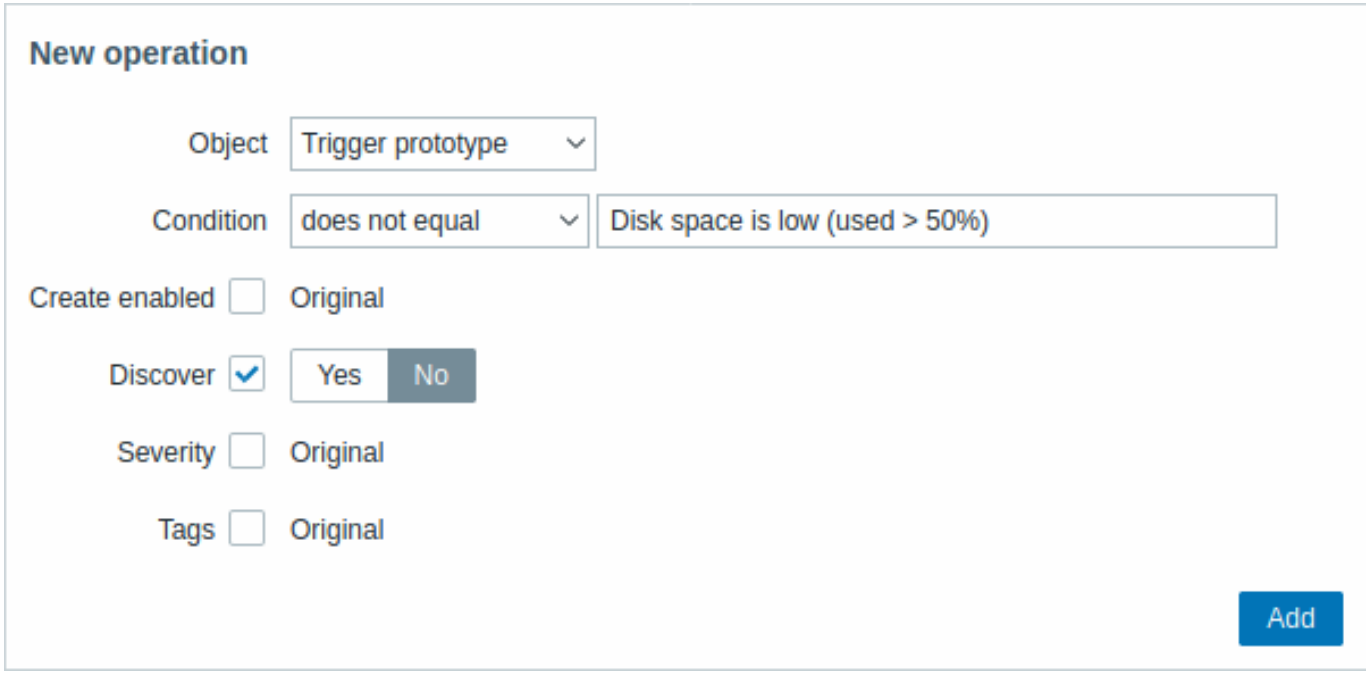


Wszystkie wymagane parametry są oznaczone czerwonymi gwiazdkami.

Parametr	Opis
Name	Unikalna nazwa nadpisania (w obrębie reguły LLD).
If filter matches	Określa, czy kolejne nadpisania mają być przetwarzane po spełnieniu warunków filtra: **Continue overrides** - kolejne nadpisania będą przetwarzane. **Stop processing** - zostaną wykonane operacje z poprzednich (jeśli istnieją) oraz z tego nadpisania, a kolejne nadpisania zostaną zignorowane dla dopasowanych wierszy LLD.
Filters	Określa, do których wykrytych encji ma być zastosowane nadpisanie. Filtry nadpisania są przetwarzane po filtrach reguły discovery i mają taką samą funkcjonalność.
Operations	Operacje nadpisania są wyświetlane z następującymi szczegółami: **Condition** - typ obiektu i warunek, jaki ma spełniać nazwa obiektu; na przykład: Trigger prototype nie jest równe *Disk space is low (used > 50%)*. **Actions** - wyświetlane są linki do edycji i usuwania operacji.

Konfigurowanie operacji

Aby skonfigurować szczegóły nowej operacji, kliknij Add w bloku Operations. Aby edytować istniejącą operację, kliknij Edit obok operacji. Otworzy się okno popup, w którym można edytować szczegóły operacji.



Parametr	Opis
Object	Dostępnych jest pięć typów obiektów: Item prototype Trigger prototype Graph prototype Host prototype Discovery prototype
Condition	Umożliwia filtrowanie encji, do których ma zostać zastosowana operacja.
Operator	Obsługiwane operatory: **equals** - zastosuj do tego prototypu **does not equal** - zastosuj do wszystkich prototypów z wyjątkiem tego **contains** - zastosuj, jeśli nazwa prototypu zawiera ten ciąg **does not contain** - zastosuj, jeśli nazwa prototypu nie zawiera tego ciągu **matches** - zastosuj, jeśli nazwa prototypu pasuje do wyrażenia regularnego **does not match** - zastosuj, jeśli nazwa prototypu nie pasuje do wyrażenia regularnego
Pattern	Wyrażenie regularne lub ciąg do wyszukania.
Object: *Item prototype*	
Create enabled	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu item: *Yes* - item zostanie dodany w stanie włączonym. *No* - item zostanie dodany do wykrytej encji, ale w stanie wyłączonym.
Discover	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu item: *Yes* - item zostanie dodany. *No* - item nie zostanie dodany.
Update interval	Gdy pole wyboru jest zaznaczone, pojawią się dwie opcje umożliwiające ustawienie innego interwału dla item: *Delay* - interwał aktualizacji item. Obsługiwane są makra użytkownika i sufiksy czasu (np. 30s, 1m, 2h, 1d). Należy ustawić 0, jeśli używany jest *Custom interval*. *Custom interval* - kliknij Add, aby określić elastyczne/interwały harmonogramu. Szczegółowe informacje znajdują się w sekcji Custom intervals.

Parametr		Opis
	History	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające ustawienie innego okresu przechowywania historii dla item: *Do not store* - jeśli wybrane, historia nie będzie przechowywana. *Store up to* - jeśli wybrane, po prawej stronie pojawi się pole do określenia okresu przechowywania. Obsługiwane są makra użytkownika i makra LLD.
	Trends	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające ustawienie innego okresu przechowywania trendów dla item: *Do not store* - jeśli wybrane, trendy nie będą przechowywane. *Store up to* - jeśli wybrane, po prawej stronie pojawi się pole do określenia okresu przechowywania. Obsługiwane są makra użytkownika i makra LLD.
	Tags	Gdy pole wyboru jest zaznaczone, pojawi się nowy blok umożliwiający określenie par tag-wartość. Te tagi zostaną dołączone do tagów określonych w prototypie item, nawet jeśli nazwy tagów są takie same.
Object: *Trigger prototype*		
	Create enabled	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu trigger: *Yes* - trigger zostanie dodany w stanie włączonym. *No* - trigger zostanie dodany do wykrytej encji, ale w stanie wyłączonym.
	Discover	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu trigger: *Yes* - trigger zostanie dodany. *No* - trigger nie zostanie dodany.
	Severity	Gdy pole wyboru jest zaznaczone, pojawią się przyciski poziomu ważności trigger, umożliwiające jego modyfikację.
	Tags	Gdy pole wyboru jest zaznaczone, pojawi się nowy blok umożliwiający określenie par tag-wartość. Te tagi zostaną dołączone do tagów określonych w prototypie trigger, nawet jeśli nazwy tagów są takie same.
Object: *Graph prototype*		
	Discover	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu graph: *Yes* - graph zostanie dodany. *No* - graph nie zostanie dodany.
Object: *Host prototype*		
	Create enabled	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu host: *Yes* - host zostanie utworzony w stanie włączonym. *No* - host zostanie utworzony w stanie wyłączonym.
	Discover	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu host: *Yes* - host zostanie wykryty. *No* - host nie zostanie wykryty.
	Link templates	Gdy pole wyboru jest zaznaczone, pojawi się pole do określenia szablonów. Zacznij wpisywać nazwę szablonu lub kliknij *Select* obok pola i wybierz szablony z listy w oknie popup. Szablony z tego nadpisania są dodawane do wszystkich szablonów już powiązanych z prototypem host.

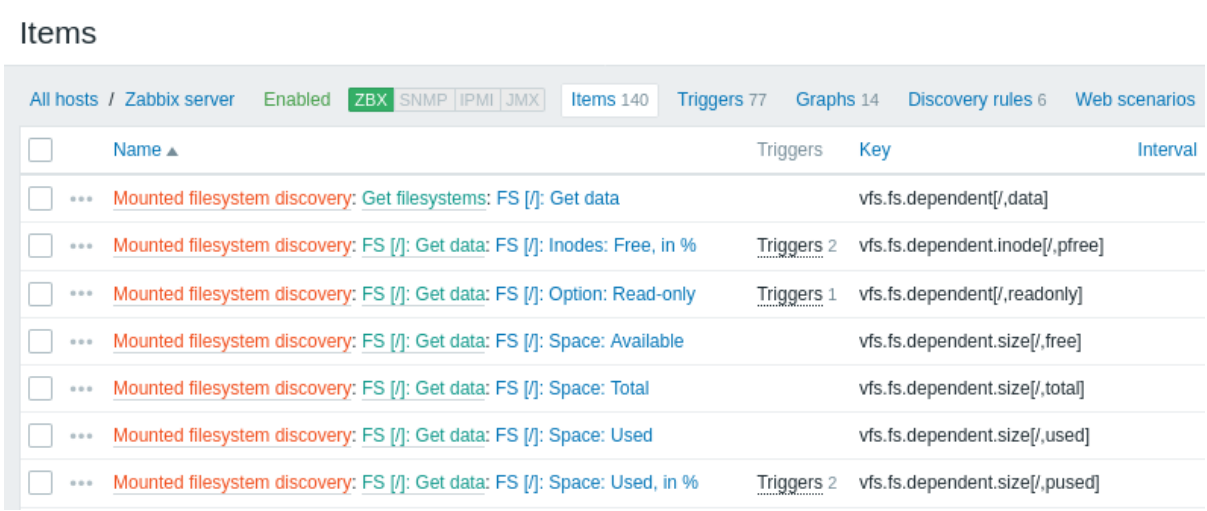
Parametr	Opis
Tags	Gdy pole wyboru jest zaznaczone, pojawi się nowy blok umożliwiający określenie par tag-wartość. Te tagi zostaną dołączone do tagów określonych w prototypie host, nawet jeśli nazwy tagów są takie same.
Host inventory	Gdy pole wyboru jest zaznaczone, pojawią się przyciski umożliwiające wybór innego trybu inwentarza dla prototypu host: *Disabled* - nie wypełniaj inwentarza host *Manual* - podaj szczegóły ręcznie *Automated* - automatycznie wypełnij dane inwentarza host na podstawie zebranych metryk.

Przyciski formularza

Przyciski na dole formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj regułę wykrywania. Ten przycisk jest dostępny tylko dla nowych reguł wykrywania.
Update	Zaktualizuj właściwości reguły wykrywania. Ten przycisk jest dostępny tylko dla istniejących reguł wykrywania.
Clone	Utwórz kolejną regułę wykrywania na podstawie właściwości bieżącej reguły wykrywania.
Execute now	Wykonaj wykrywanie natychmiast na podstawie reguły wykrywania. Reguła wykrywania musi już istnieć. Zobacz więcej szczegółów. *Uwaga:* podczas natychmiastowego wykonywania wykrywania pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wynik nie będzie odzwierciedlał bardzo niedawnych zmian w konfiguracji reguły wykrywania.
Test	Przetestuj konfigurację reguły wykrywania. Użyj tego przycisku, aby zweryfikować ustawienia konfiguracji (takie jak łączność i poprawność parametrów) bez trwałego stosowania jakichkolwiek zmian.
Delete	Usuń regułę wykrywania.
Cancel	Anuluj edycję właściwości reguły wykrywania.

Odkryte encje Poniższe zrzuty ekranu ilustrują, jak odkryte pozycje, wyzwalacze i wykresy wyglądają w konfiguracji hosta. Odkryte encje są poprzedzone pomarańczowym odnośnikiem do reguły wykrywania, z której pochodzą.



Należy pamiętać, że odkryte encje nie zostaną utworzone, jeśli istnieją już encje o tych samych kryteriach unikalności, na przykład pozycja z takim samym kluczem lub wykres o tej samej nazwie. W takim przypadku w frontend jest wyświetlany komunikat o

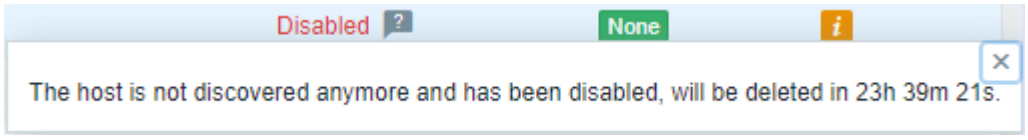
błędzie informujący, że reguła wykrywania niskiego poziomu nie mogła utworzyć niektórych encji. Sama reguła wykrywania nie stanie się jednak nieobsługiwana tylko dlatego, że nie udało się utworzyć jakiejś encji i trzeba było ją pominąć. Reguła wykrywania będzie nadal tworzyć/aktualizować inne encje.

Jeśli odkryta encja (host, system plików, interfejs itp.) przestanie być wykrywana (lub nie przejdzie już filtra), encje utworzone na jej podstawie mogą zostać automatycznie wyłączone, a ostatecznie usunięte.

Utracone zasoby mogą być automatycznie wyłączane na podstawie wartości parametru *Disable lost resources*. Dotyczy to utraconych hostów, pozycji i wyzwalaczy.

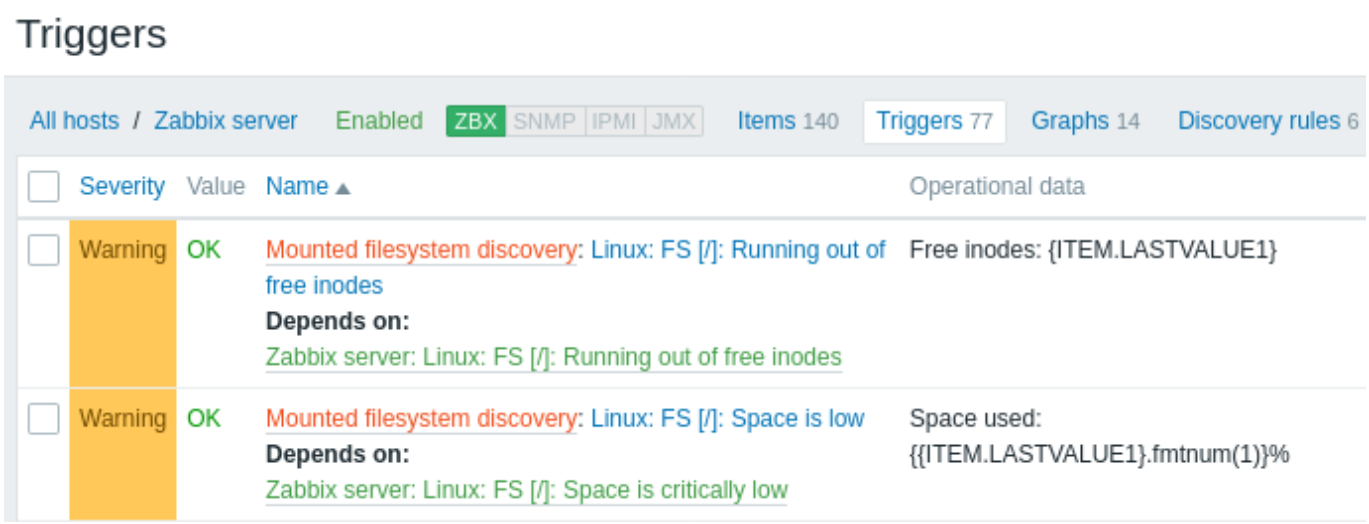
Utracone zasoby mogą być automatycznie usuwane na podstawie wartości parametru *Delete lost resources*. Dotyczy to utraconych hostów, grup hostów, pozycji, wyzwalaczy i wykresów.

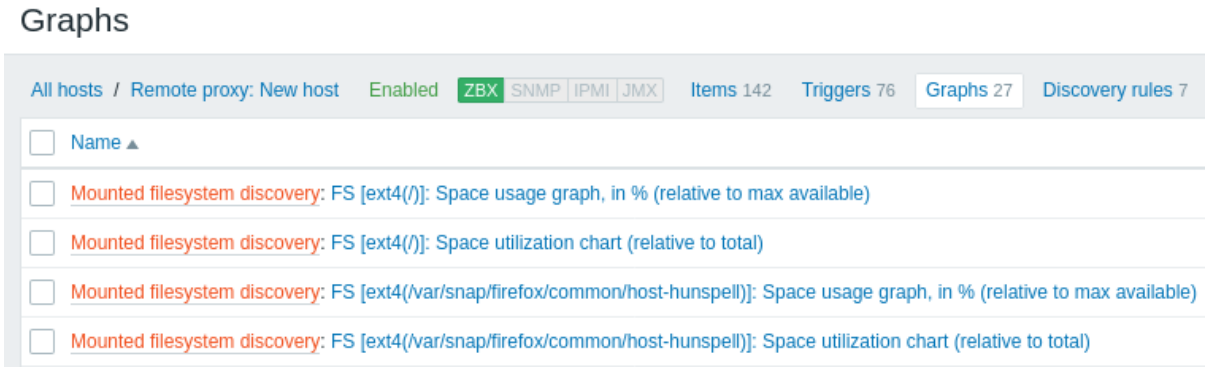
Gdy odkryte encje przechodzą do stanu „Not discovered anymore", na liście encji wyświetlany jest wskaźnik czasu życia. Po najechaniu na niego kursorem myszy zostanie wyświetlony komunikat zawierający szczegóły statusu.



Jeśli encje zostały oznaczone do usunięcia, ale nie zostały usunięte w oczekiwanym czasie (wyłączona reguła wykrywania lub host pozycji), zostaną usunięte przy następnym przetwarzaniu reguły wykrywania.

Encje zawierające inne encje oznaczone do usunięcia nie będą aktualizowane, jeśli zostaną zmienione na poziomie reguły wykrywania. Na przykład wyzwalacze oparte na LLD nie będą aktualizowane, jeśli zawierają pozycje oznaczone do usunięcia.





Inne typy wykrywania Więcej szczegółów oraz instrukcje dotyczące innych gotowych typów wykrywania są dostępne w następujących sekcjach:

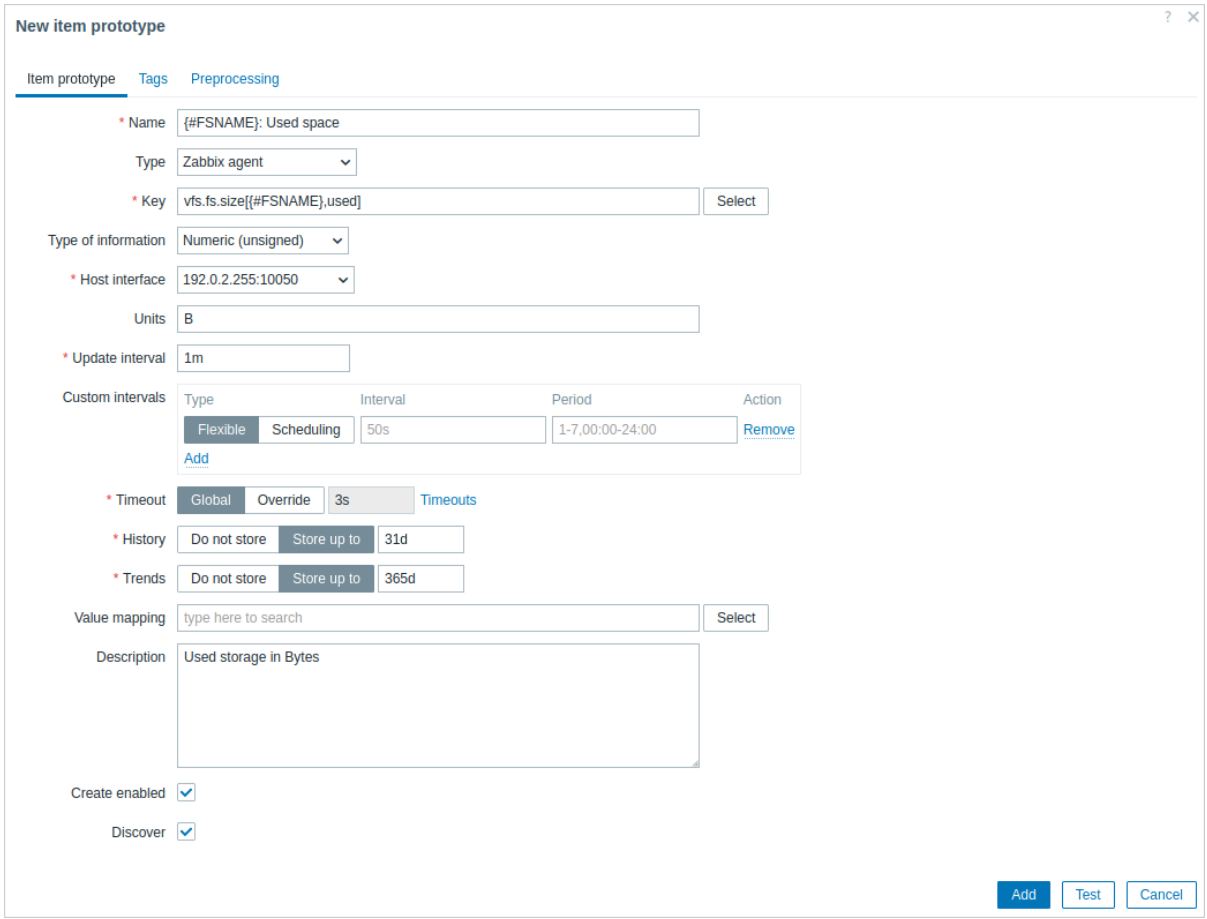
- wykrywanie interfejsów sieciowych
- wykrywanie procesorów i rdzeni CPU
- wykrywanie OID-ów SNMP
- wykrywanie obiektów JMX;
- wykrywanie przy użyciu zapytań SQL ODBC
- wykrywanie usług Windows
- wykrywanie interfejsów hosta w Zabbix

Więcej szczegółów na temat formatu JSON dla pozycji wykrywania oraz przykład implementacji własnego mechanizmu wykrywania systemu plików jako skryptu Perl można znaleźć w sekcji tworzenie niestandardowych reguł LLD.

1 Prototypy pozycji

Po utworzeniu reguły przejdź do pozycji dla tej reguły i naciśnij „Utwórz prototyp pozycji”, aby utworzyć prototyp pozycji.

Zwróć uwagę, jak makro {#FSNAME} jest używane tam, gdzie wymagana jest nazwa systemu plików. Użycie makra wykrywania niskiego poziomu jest obowiązkowe w kluczu pozycji, aby mieć pewność, że wykrywanie zostanie przetworzone poprawnie. Gdy reguła wykrywania zostanie przetworzona, to makro zostanie zastąpione wykrytym systemem plików.



Note:
Gdy parametry klucza pozycji nie są używane, umieść makro LLD wewnątrz nawiasów klucza pozycji `[...]` (na przykład `v_[{#MACRO}]`).

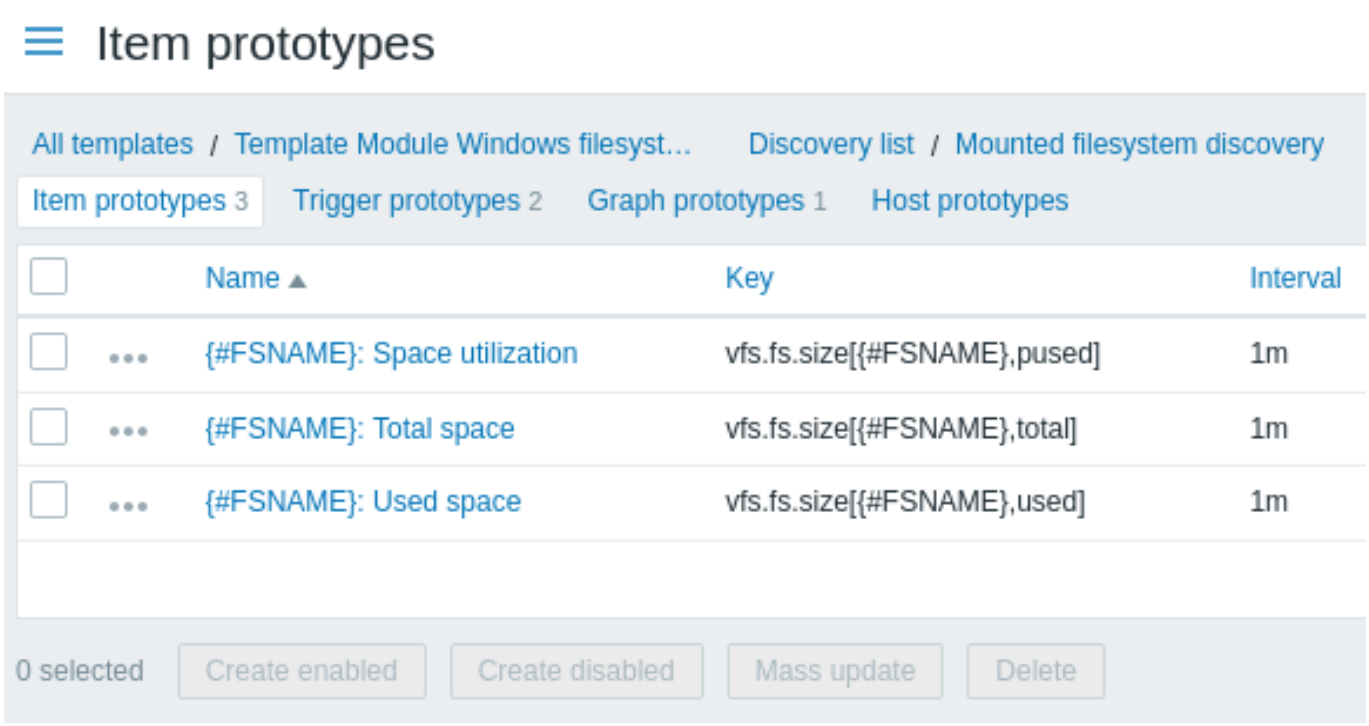
Makra wykrywania niskiego poziomu macros oraz makra użytkownika macros są obsługiwane w konfiguracji prototypu pozycji i parametrach wstępnego przetwarzania wartości pozycji. Zwróć uwagę, że w przypadku użycia w interwałach aktualizacji pojedyncze makro musi wypełniać całe pole. Wiele makr w jednym polu lub makra połączone z tekstem nie są obsługiwane.

Note:
Dla bezpiecznego użycia w parametrach wstępnego przetwarzania wyrażeń regularnych i XPath wykonywane jest odpowiednie dla kontekstu maskowanie makr wykrywania niskiego poziomu.

Atrybuty specyficzne dla prototypów pozycji:

Parameter	Description
Create enabled	Jeśli zaznaczone, pozycja zostanie dodana w stanie włączonym. Jeśli niezaznaczone, pozycja zostanie dodana do wykrytego obiektu, ale w stanie wyłączonym.
Discover	Jeśli zaznaczone (domyślnie), pozycja zostanie dodana do wykrytego obiektu. Jeśli niezaznaczone, pozycja nie zostanie dodana do wykrytego obiektu, chyba że to ustawienie zostanie nadpisane w regule wykrywania.

Możemy utworzyć kilka prototypów pozycji dla każdej metryki systemu plików, która nas interesuje:



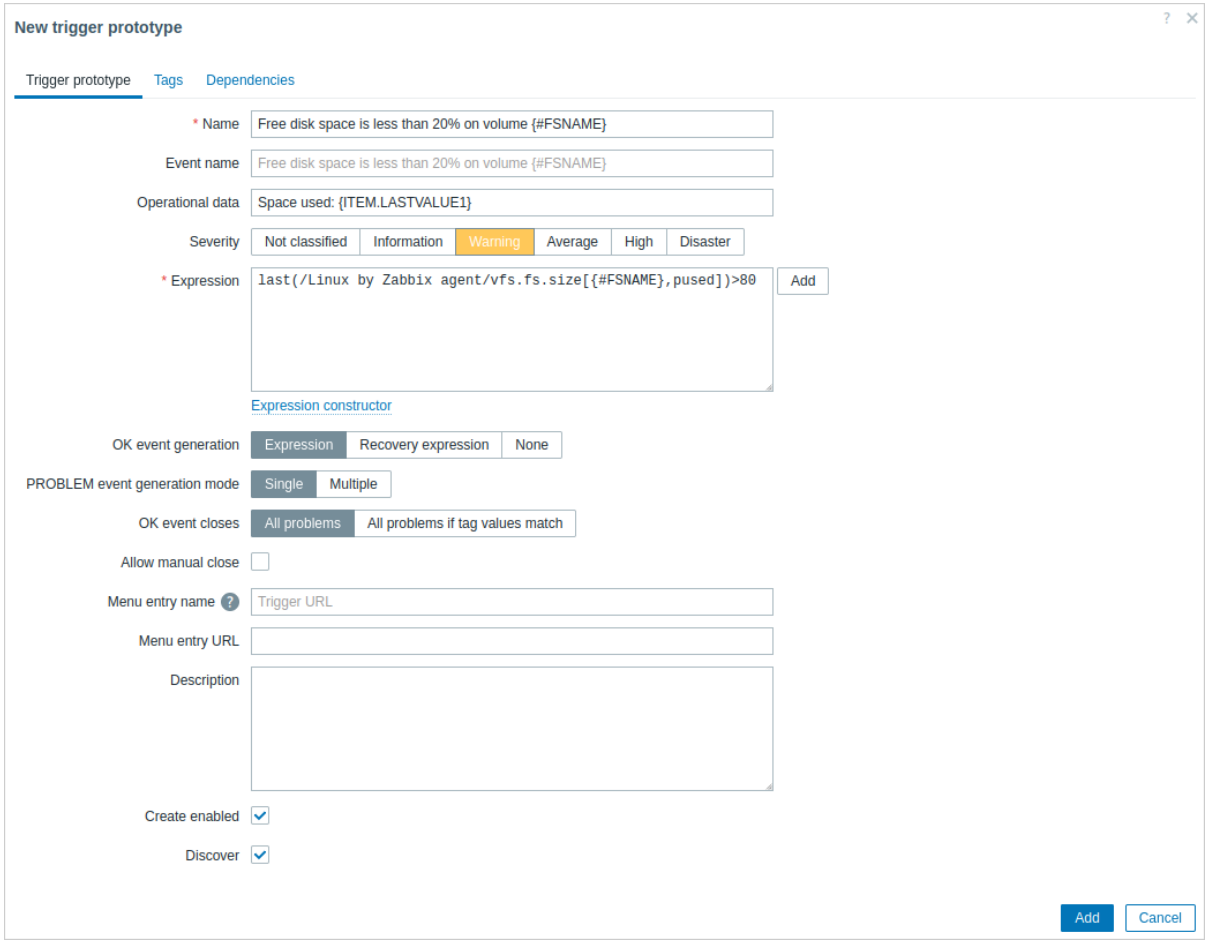
Kliknij ikonę z trzema kropkami, aby otworzyć menu dla konkretnego prototypu pozycji z następującymi opcjami:

- *Create trigger prototype* - utwórz prototyp wyzwalacza na podstawie tego prototypu pozycji
- *Trigger prototypes* - kliknij, aby zobaczyć listę z odnośnikami do już skonfigurowanych prototypów wyzwalaczy tego prototypu pozycji
- *Create dependent item* - utwórz pozycję zależną dla tego prototypu pozycji

Opcja *Mass update* jest dostępna, jeśli chcesz jednocześnie zaktualizować właściwości kilku prototypów pozycji.

2 Prototypy wyzwalaczy

Prototypy wyzwalaczy tworzy się w podobny sposób jak prototypy pozycji:

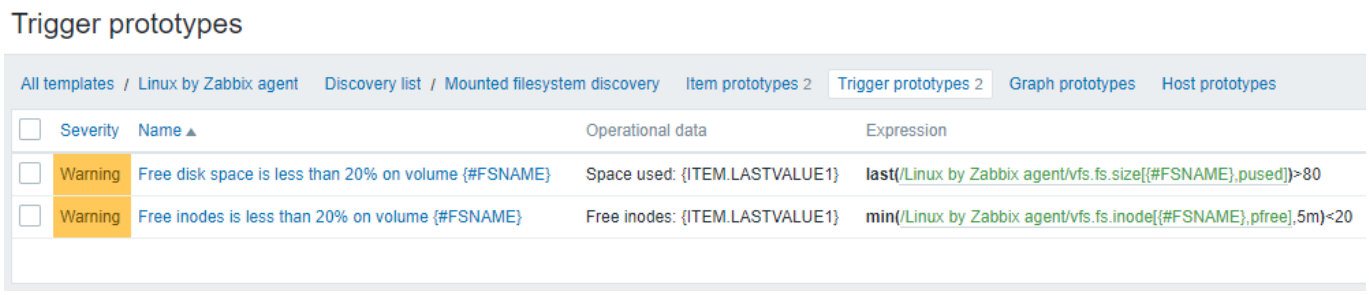


Atrybuty specyficzne dla prototypów wyzwalaczy:

Parametr	Opis
Create enabled	Jeśli zaznaczone, wyzwalacz zostanie dodany w stanie włączonym. Jeśli odznaczone, wyzwalacz zostanie dodany do wykrytego obiektu, ale w stanie wyłączonym.
Discover	Jeśli zaznaczone (domyślnie), wyzwalacz zostanie dodany do wykrytego obiektu. Jeśli odznaczone, wyzwalacz nie zostanie dodany do wykrytego obiektu, chyba że to ustawienie zostanie nadpisane w regule wykrywania.

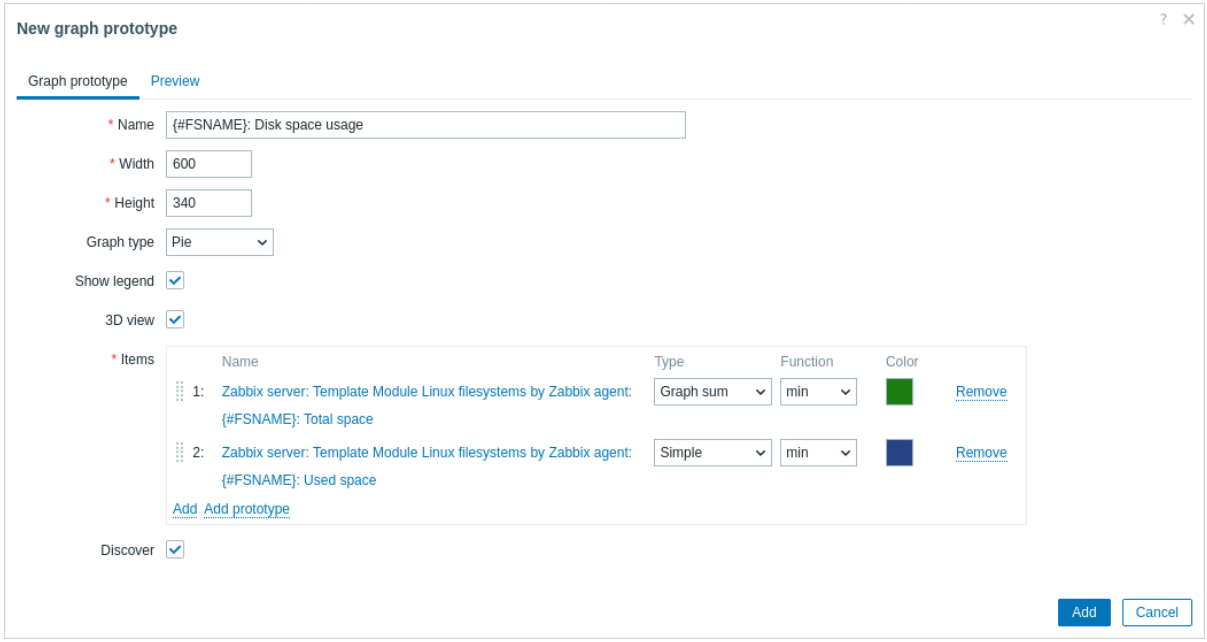
Gdy z prototypów tworzone są rzeczywiste wyzwalacze, może pojawić się potrzeba elastycznego określenia, jaka stała („20” w naszym przykładzie) ma być użyta do porównania w wyrażeniu. Zobacz, jak makra użytkownika z kontekstem mogą być przydatne do uzyskania takiej elastyczności.

Można zdefiniować zależności między prototypami wyzwalaczy. Aby to zrobić, przejdź do zakładki *Dependencies*. Prototyp wyzwalacza może zależeć od innego prototypu wyzwalacza z tej samej reguły wykrywania niskiego poziomu (LLD) lub od zwykłego wyzwalacza. Prototyp wyzwalacza nie może zależeć od prototypu wyzwalacza z innej reguły LLD ani od wyzwalacza utworzonego z prototypu wyzwalacza. Prototyp wyzwalacza hosta nie może zależeć od wyzwalacza z szablonu.



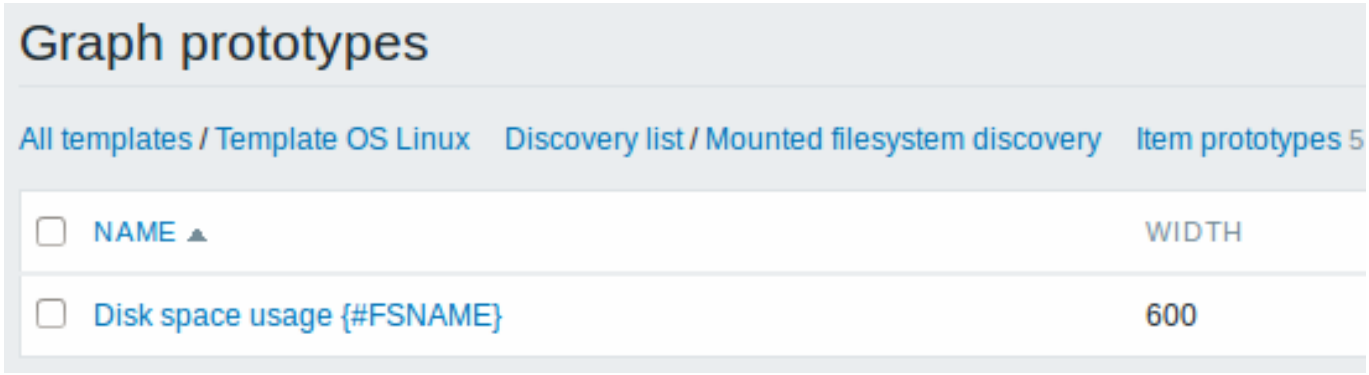
3 Prototypy wykresów

Możemy również tworzyć prototypy wykresów:

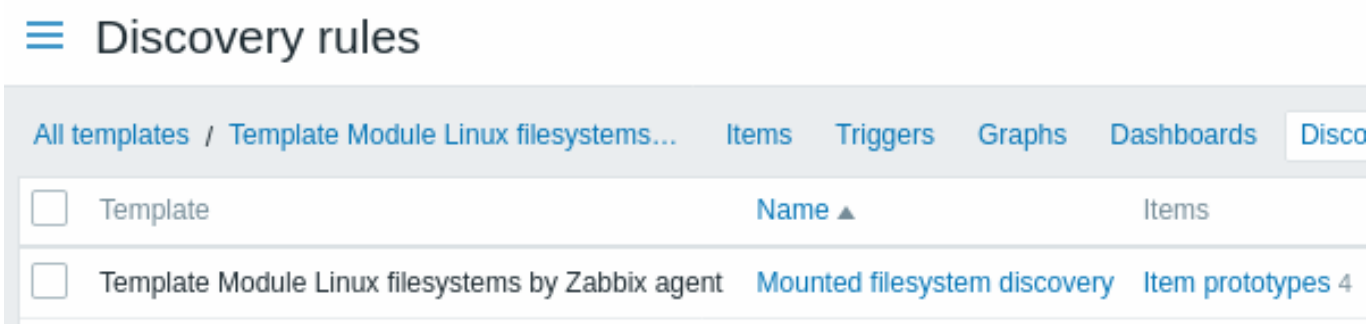


Atrybuty specyficzne dla prototypów wykresów:

Parametr	Opis
Discover	Jeśli zaznaczone (domyślnie), wykres zostanie dodany do wykrytego obiektu. Jeśli odznaczone, wykres nie zostanie dodany do wykrytego obiektu, chyba że to ustawienie zostanie nadpisane w regule wykrywania.



Na koniec utworzyliśmy regułę wykrywania, która wygląda jak pokazano poniżej. Ma ona pięć prototypów pozycji, dwa prototypy wyzwalaczy i jeden prototyp wykresu.



4 Prototypy hostów

Prototypy hostów są wzorcami do tworzenia hostów za pomocą reguł odkrywania niskiego poziomu. Zanim zostaną odkryte jako hosty, te prototypy nie mogą mieć pozycji ani wyzwalaczy, z wyjątkiem tych podłączonych z szablonów.

Konfiguracja

Prototypy hostów są konfigurowane w ramach reguł wykrywania niskiego poziomu.

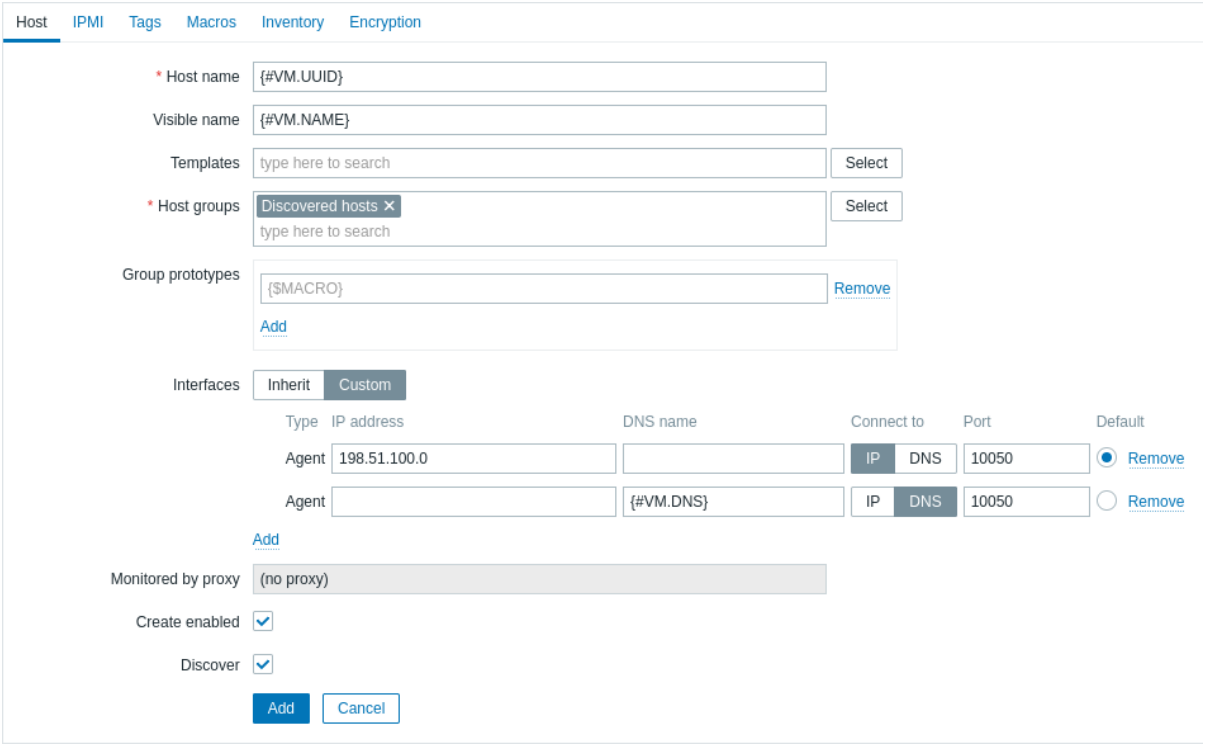
Aby utworzyć prototyp hosta:

1. Przejdź do *Zbieranie danych → Hosty*.

2. Kliknij *Wykrywanie* dla wymaganego hosta, aby przejść do listy reguł wykrywania niskiego poziomu skonfigurowanych dla tego hosta.

3. Kliknij *Prototypy hostów* dla wymaganej reguły wykrywania.

4. Kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu.



Prototypy hostów mają te same parametry co zwykłe hosty; jednak następujące parametry obsługują inną lub dodatkową konfigurację:

Parameter	Description
Host name	Ten parametr musi zawierać co najmniej jedno makro wykrywania niskiego poziomu, aby zapewnić unikalne nazwy hostów dla tworzonych hostów.
Visible name	Obsługiwane są makra wykrywania niskiego poziomu.
Group prototypes	Umożliwia określanie prototypów grup hostów za pomocą makr wykrywania niskiego poziomu. Na podstawie określonych prototypów grup zostaną wykryte, utworzone i powiązane z utworzonymi hostami grupy hostów; wykryte grupy, które zostały już utworzone przez inne reguły wykrywania niskiego poziomu, również zostaną powiązane z utworzonymi hostami. Jednak wykryte grupy hostów, które odpowiadają ręcznie utworzonym grupom hostów, nie zostaną powiązane z utworzonymi hostami.
Interfaces	Określa, czy wykryte hosty dziedziczą adres IP z hosta, do którego należy reguła wykrywania (domyślnie), czy otrzymują niestandardowe interfejsy. Obsługiwane są makra wykrywania niskiego poziomu oraz makra użytkownika.
Create enabled	Ustawia status wykrytych hostów; jeśli opcja nie jest zaznaczona, hosty zostaną utworzone jako wyłączone.
Discover	Określa, czy hosty będą tworzone na podstawie prototypu hosta; jeśli opcja nie jest zaznaczona, hosty nie będą tworzone na podstawie prototypu hosta (chyba że to ustawienie zostanie zastąpione w regule wykrywania niskiego poziomu).

Note:
Makra wykrywania niskiego poziomu są również obsługiwane dla wartości tagów oraz wartości makr użytkownika prototypu hosta.
 Mapy wartości nie są obsługiwane dla prototypów hostów.

Przykład konfiguracji prototypu hosta znajduje się w sekcji *Monitorowanie maszyny wirtualnej*.

Interfejsy hosta

Aby dodać niestandardowe interfejsy, przełącz selektor *Interfaces* z „Inherit” na „Custom”. Kliknij Add i wybierz typ interfejsu — Zabbix agent, SNMP, JMX, IPMI.

Note:
Jeśli wybrano *Custom*, ale nie ustawiono żadnych interfejsów, hosty zostaną utworzone bez interfejsów.
 Jeśli wybrano *Inherit* i prototyp hosta należy do szablonu, wszystkie wykryte hosty odziedziczą interfejs hosta od hosta, do którego podłączony jest szablon.

Jeśli określono wiele niestandardowych interfejsów, podstawowy interfejs można ustawić w kolumnie *Default*.

Przykład konfiguracji niestandardowego interfejsu hosta znajduje się w *przykładzie konfiguracji monitorowania VMware*.

Warning:
Host zostanie utworzony tylko wtedy, gdy interfejs hosta zawiera poprawne dane.

Wykryte hosty

Na liście hostów wykryte hosty są poprzedzone nazwą reguły wykrywania, która je utworzyła.

Wykryte hosty dziedziczą większość parametrów z prototypów hostów jako *tylko do odczytu*. Skonfigurować można tylko następujące parametry wykrytego hosta:

- *Szablony* - połączyć nowe szablony lub odłączyć dodane ręcznie. Szablonów odziedziczonych z prototypu hosta nie można odłączyć.
- *Opis* - dodać lub edytować opis hosta.
- *Status* - włączyć lub wyłączyć host.
- *Tagi* - dodać nowe tagi lub usunąć dodane ręcznie. Tagów odziedziczonych z prototypu hosta nie można usunąć. Należy pamiętać, że tagi nie mogą się powielać (tagi o tej samej nazwie i wartości). Jeśli do prototypu hosta zostanie dodany nowy tag i będzie on odpowiadał tagowi dodanemu ręcznie w wykrytym hoście, podczas wykrywania tag dodany ręcznie zostanie zastąpiony tagiem odziedziczonym.
- *Makra* - dodać nowe makra hosta lub usunąć dodane ręcznie, a także te odziedziczone z prototypu hosta; zmienić wartości makr i typy.

Note:
Wykryte hosty dziedziczą również makra użytkownika z hosta, na którym skonfigurowano regułę wykrywania. Te odziedziczone makra można usunąć, a ich wartości i typy można zmienić.

Wykryte hosty mogą również wykrywać inne hosty. Na przykład, jeśli prototyp hosta (używany do wykrywania hiperwizorów) ma podłączony szablon zawierający regułę wykrywania niskiego poziomu z własnym prototypem hosta (używanym do wykrywania maszyn wirtualnych), Zabbix wykryje zarówno hiperwizory, jak i ich maszyny wirtualne. Aby dodatkowo wykrywać kontenery na tych maszynach wirtualnych, można utworzyć reguły wykrywania niskiego poziomu lub podłączyć nowy szablon z prototypami hostów do wykrytych maszyn wirtualnych, albo wstępnie skonfigurować prototyp hosta maszyny wirtualnej przez podłączenie szablonu, który sam zawiera prototypy hostów.

Wykryte hosty można usuwać ręcznie. Należy jednak pamiętać, że zostaną wykryte ponownie, jeśli wykrywanie będzie dla nich włączone.

Hosty, które nie są już wykrywane, mogą zostać:

- automatycznie wyłączone (na podstawie wartości *Disable lost resources* reguły wykrywania)
- automatycznie usunięte (na podstawie wartości *Delete lost resources* reguły wykrywania).

5 Prototypy wykrywania

Przegląd

Prototypy wykrywania to **zagnieżdżone reguły wykrywania niskiego poziomu** w ramach „nadrzędnej” reguły wykrywania, umożliwiające tworzenie wielopoziomowego wykrywania obiektów z ich własnymi pozycjami, wyzwalaczami itp. Na przykład możesz chcieć wykryć wszystkie instancje bazy danych na serwerze bazy danych, następnie wykryć przestrzenie tabel dla każdej instancji, a potem wykryć tabele dla każdej przestrzeni tabel.

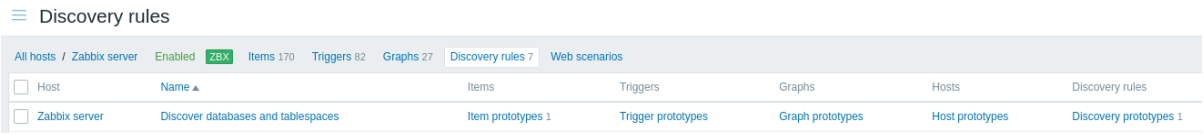
Prototypy wykrywania mają własne prototypy pozycji, wyzwalaczy, wykresów, hostów i wykrywania. Zagnieżdżony prototyp wykrywania będzie używać tej samej wartości JSON co reguła nadrzędna, jeśli określisz typ *Nested*.

Liczba poziomów zagnieżdżenia dla prototypów wykrywania jest nieograniczona.

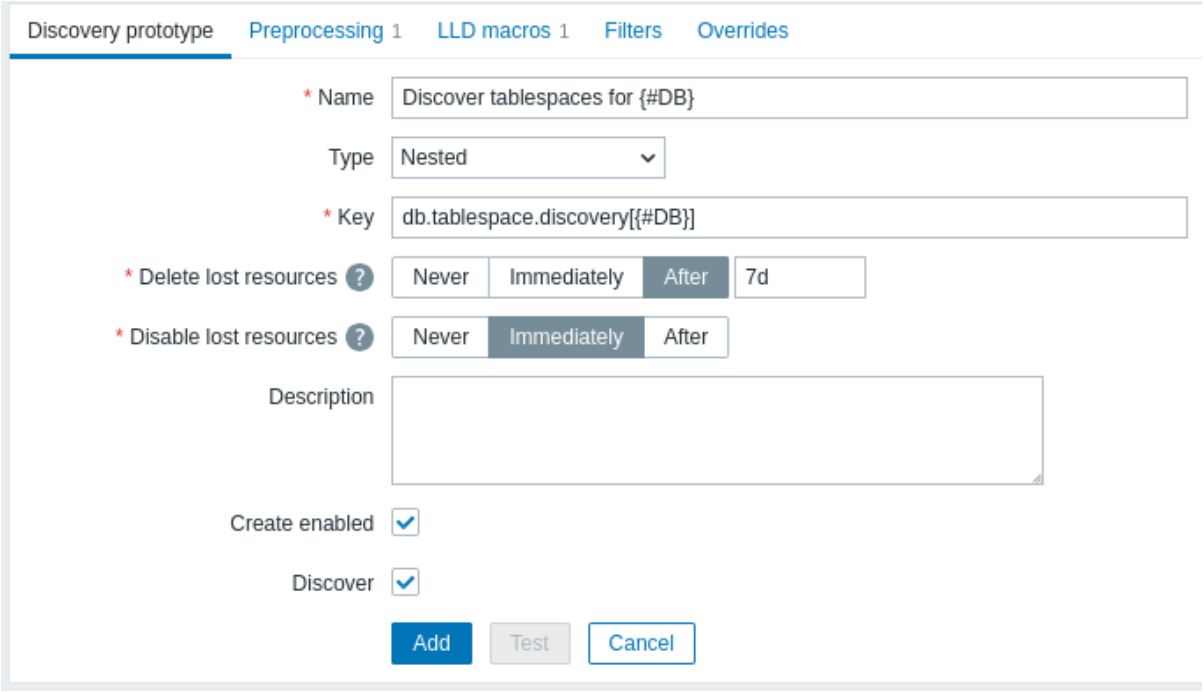
Konfiguracja

Aby utworzyć prototyp wykrywania:

- Kliknij *Discovery prototypes* w wierszu istniejącej reguły wykrywania



- Kliknij *Create discovery prototype*



Pola konfiguracyjne tego formularza są wspólne ze standardowym wykrywaniem niskiego poziomu.

Jeśli w otwartym formularzu prototypu wykrywania wybierzesz „Nested" jako *Type*, reguły wykrywania (z prototypu wykrywania) będą generowane na podstawie obiektu JSON z tej samej wartości JSON co nadrzędna reguła wykrywania. Na przykład, jeśli oryginalny JSON ma postać [<object A>, <object B>] i istnieje jeden prototyp zagnieżdżonej reguły wykrywania, wówczas zostaną wygenerowane dwie reguły wykrywania odpowiednio na podstawie danych obiektu A i obiektu B.

W takim przypadku prototyp wykrywania jest aktywowany w tym samym czasie co reguła nadrzędna. Dzięki temu reguła zagnieżdżona może używać przetwarzania wstępnego, aby pracować na innym „wycinku" tych samych danych, już pobranych przez regułę nadrzędną.

Makra LLD z nadrzędnej reguły LLD są dostępne dla zagnieżdżonych reguł wykrywania.

Zagnieżdżone reguły LLD na wykrytych hostach

Zagnieżdżona reguła wykrywania niskiego poziomu może być używana w szablonie hosta przypisanym do prototypu hosta. Jeśli na wykrytym hoście istnieje *zagnieżdżona* reguła wykrywania, wówczas obiekt JSON użyty do wykrycia hosta jest również przekazywany do wszystkich reguł LLD typu zagnieżdżonego na tym hoście. Więcej szczegółów znajduje się w przykładzie.

Makra LLD z reguły wykrywania, która utworzyła host, są dostępne dla zagnieżdżonych reguł wykrywania.

Przykład

Zilustrujmy możliwe zastosowanie prototypów wykrywania na podstawie otrzymania następującego przykładu wielopoziomowego JSON.

```
[
  {
    "database": "db1",
    "created_at": "2024-02-01T12:30:00Z",
    "encoding": "UTF8",
    "tablespaces": [
      { "name": "ts1", "max_size": "10GB" },
      { "name": "ts2", "max_size": "20GB" },
      { "name": "ts3", "max_size": "15GB" }
    ]
```

```
  },
  {
    "database": "db2",
    "created_at": "2023-11-15T08:45:00Z",
    "encoding": "UTF16",
    "tablespaces": [
      { "name": "ts1", "max_size": "5GB" },
      { "name": "ts2", "max_size": "25GB" },
      { "name": "ts3", "max_size": "30GB" }
    ]
  },
  {
    "database": "db3",
    "created_at": "2024-01-05T15:10:00Z",
    "encoding": "UTF8",
    "tablespaces": [
      { "name": "ts1", "max_size": "12GB" },
      { "name": "ts2", "max_size": "18GB" },
      { "name": "ts3", "max_size": "22GB" }
    ]
  }
]
```

Przypadek 1

Wykrywanie instancji baz danych na serwerze bazy danych, a następnie wykrywanie tablespace'ów dla każdej instancji.

1. Masz co najmniej jeden host powiązany z wykrywaniem serwera bazy danych.
2. Utwórz regułę LLD dla tego hosta o nazwie *Wykryj bazy danych i tablespace'y*.
3. Przejdź do zakładki *Makra LLD* w tej regule i dodaj makro `{#DB}=$.database`.
4. Dodaj prototyp pozycji dla tej reguły o nazwie *Aktywne połączenia do {#DB}* (Typ: Agent, klucz: `db.connections[{#DB}]`).
5. Zostaną wykryte pozycje powiązane z każdą bazą danych:

```
Active connections to db1, Active connections to db2, Active connections to db3.
```

6. Utwórz prototyp wykrywania dla tej reguły o nazwie *Wykryj tablespace'y dla {#DB}* (Typ: Zagnieżdżony, klucz: `db.tablespace.discovery[{#DB}]`).
7. Przejdź do zakładki *Przetwarzanie wstępne* tego prototypu wykrywania i dodaj krok `JSONPath=$.tablespaces`.
8. Przejdź do zakładki *Makra LLD* tego prototypu wykrywania i dodaj makro `{#TSNAME}=$.name`.
9. Utwórz prototyp pozycji dla tego prototypu wykrywania o nazwie *Rozmiar tablespace'a {#TSNAME} dla {#DB}* (Typ: Agent, klucz: `db.ts.size[{#DB}, {#TSNAME}]`).
10. Zostaną wykryte pozycje powiązane z każdym tablespace'em każdej bazy danych:

```
Size of tablespace ts1 for db1, Size of tablespace ts2 for db1, Size of tablespace ts3 for db1,
Size of tablespace ts1 for db2, Size of tablespace ts2 for db2, Size of tablespace ts3 for db2,
Size of tablespace ts1 for db3, Size of tablespace ts2 for db3, Size of tablespace ts3 for db3.
```

z kluczami `db.ts.size[db1,ts1]`, `db.ts.size[db1,ts2]`, ... `db.ts.size[db3,ts3]`.

Przypadek 2

Odkrywanie instancji baz danych na serwerze bazy danych poprzez reprezentowanie ich jako odkrytych hostów, a następnie odkrywanie tablespace dla każdej instancji.

1. Masz co najmniej jeden host (host główny) powiązany z odkrywaniem serwera bazy danych.
2. Utwórz szablon do odkrywania tablespace dla każdej bazy danych.
3. Utwórz w tym szablonie pozycję o nazwie *Aktywne połączenia do {#DB}* (Typ: Agent, Klucz: `db.connections[{#DB}]`).
4. Utwórz regułę LLD dla tego szablonu o nazwie *Odkryj tablespace* (Typ: Zagnieżdżony).
5. Przejdź do zakładki *Preprocessing* tej reguły i dodaj krok `JSONPath=$.tablespaces`.
6. Przejdź do zakładki *LLD Macros* tej reguły, dodaj makro `{#TSNAME}=$.name`.

7. Utwórz prototyp pozycji dla tej reguły o nazwie *Rozmiar tablespace {#TSNAME} dla {#DB}* (Typ: Agent, Klucz: `db.ts.size[{#DB}, {#TSNAME}]`).
8. Wracając do hosta głównego, utwórz regułę LLD dla tego hosta o nazwie *Odkryj bazy danych i tablespace*.
9. Przejdź do zakładki *LLD Macros* tej reguły, dodaj makro `{#DB}=$.database`.
10. Dodaj prototyp hosta dla tej reguły o nazwie *Host dla bazy danych {#DB}*.
11. Przejdź do zakładki *Macros* tego prototypu hosta, dodaj makro `{$DB}={#DB}` (dla nazwy i klucza pozycji z kroku 3).
12. Podłącz szablon z kroku 2 do tego prototypu hosta.
13. Odkryte hosty zawierają odkryte pozycje powiązane z każdą bazą danych i jej tablespace:

Host	Pozycje
Host dla bazy danych db1	Aktywne połączenia do db1
	Rozmiar tablespace ts1 dla db1
	Rozmiar tablespace ts2 dla db1
	Rozmiar tablespace ts3 dla db1
Host dla bazy danych db2	Aktywne połączenia do db2
	Rozmiar tablespace ts1 dla db2
	Rozmiar tablespace ts2 dla db2
	Rozmiar tablespace ts3 dla db2
Host dla bazy danych db3	Aktywne połączenia do db3
	Rozmiar tablespace ts1 dla db3
	Rozmiar tablespace ts2 dla db3
	Rozmiar tablespace ts3 dla db3

6 Uwagi dotyczące wykrywania niskiego poziomu

Używanie makr LLD w kontekstach makr użytkownika

Makra LLD mogą być używane wewnątrz kontekstu makr użytkownika, na przykład w prototypach wyzwalaczy.

Wiele reguł LLD dla tej samej pozycji

Możliwe jest zdefiniowanie kilku reguł wykrywania niskiego poziomu z tą samą pozycją wykrywania.

Aby to zrobić, należy zdefiniować parametr agenta Alias parameter, który pozwala używać zmodyfikowanych kluczy pozycji wykrywania w różnych regułach wykrywania, na przykład `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

Limity danych dla wartości zwracanych

Nie ma limitu dla danych JSON reguły wykrywania niskiego poziomu, jeśli są one odbierane bezpośrednio przez serwer Zabbix. Dzieje się tak, ponieważ wartości zwracane są przetwarzane bez zapisywania ich w bazie danych.

Nie ma również limitu dla niestandardowych reguł wykrywania niskiego poziomu. Jeśli jednak dane niestandardowej reguły wykrywania niskiego poziomu są pobierane za pomocą parametru UserParameter, obowiązuje limit wartości zwracanej dla parametru UserParameter.

Jeśli dane muszą przejść przez proxy Zabbix, proxy musi zapisać te dane w bazie danych. W takim przypadku obowiązują limity bazy danych.

7 Reguły wykrywania

Ta sekcja zawiera praktyczne przykłady konfiguracji reguł wykrywania dla różnych typów encji, takich jak systemy plików, interfejsy sieciowe, usługi, czujniki, obiekty SNMP, aplikacje Java i bazy danych.

Każdy przykład pokazuje, jak może być zbudowana reguła wykrywania, w tym użyty klucz pozycji lub zapytanie, dane wyjściowe JSON z makrami LLD oraz sposób wykorzystania tych wyników do tworzenia prototypów pozycji, wyzwalaczy i wykresów.

Aby przejść do określonego typu reguły wykrywania, użyj paska bocznego, aby otworzyć odpowiednią stronę.

1 Wykrywanie zamontowanych systemów plików

Przegląd

Możliwe jest wykrywanie zamontowanych systemów plików i ich właściwości:

- nazwa punktu montowania
- typ systemu plików
- rozmiar systemu plików
- statystyki inode
- opcje montowania

Aby to zrobić, można użyć połączenia:

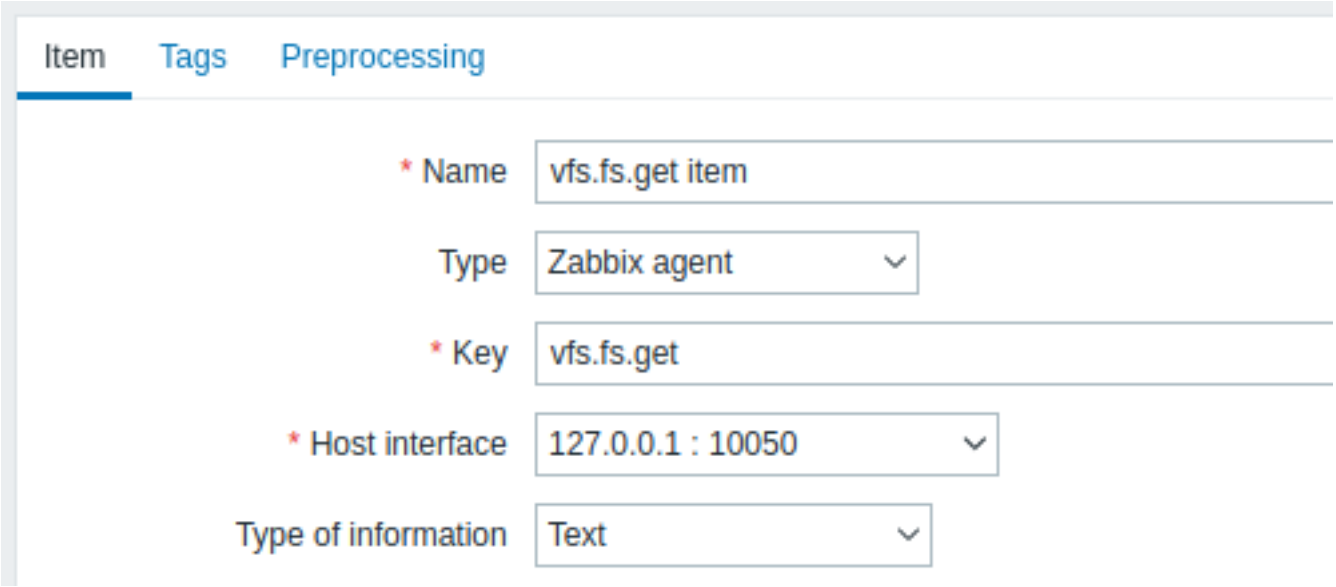
- pozycji agent `vfs.fs.get` jako pozycji nadrzędnej
- zależnej reguły wykrywania niskiego poziomu oraz prototypów pozycji

Konfiguracja

Główna pozycja

Utwórz pozycję agenta Zabbix, używając następującego klucza:

`vfs.fs.get`



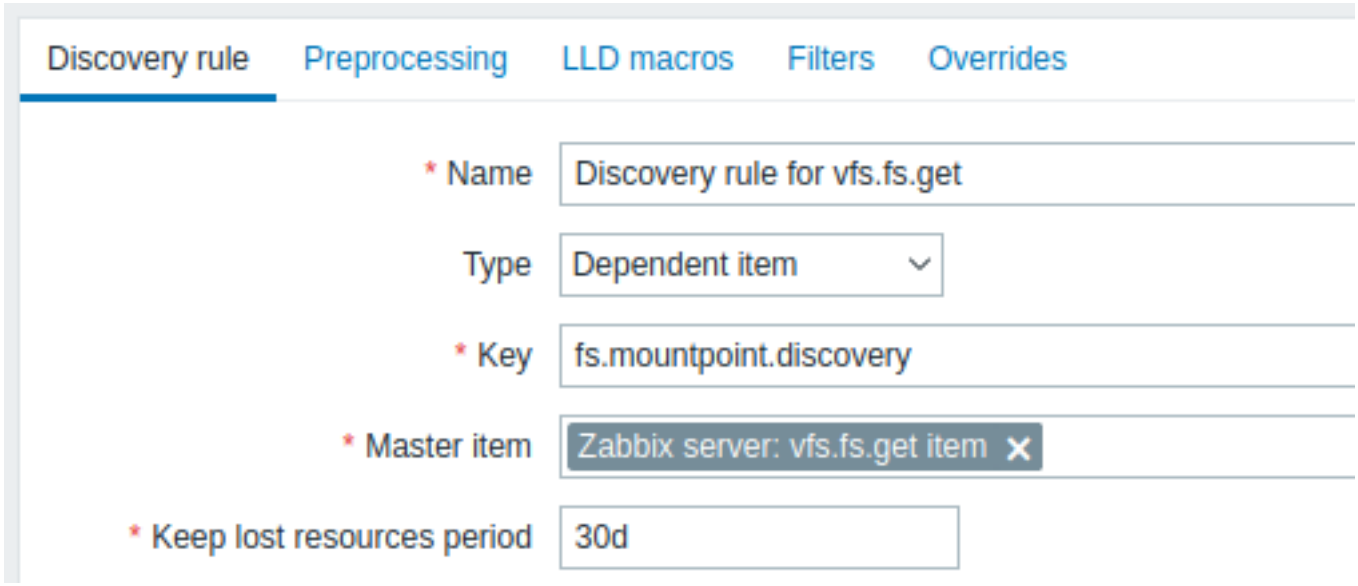
Ustaw typ informacji na „Tekst” dla potencjalnie dużych danych JSON.

Dane zwracane przez tę pozycję będą zawierać coś podobnego do poniższego dla zamontowanego systemu plików:

```
[
  {
    "fsname": "/",
    "fstype": "ext4",
    "bytes": {
      "total": 249405239296,
      "free": 24069537792,
      "used": 212595294208,
      "pfree": 10.170306,
      "pused": 89.829694
    },
    "inodes": {
      "total": 15532032,
      "free": 12656665,
      "used": 2875367,
      "pfree": 81.487503,
      "pused": 18.512497
    },
    "options": "rw,noatime,errors=remount-ro"
  }
]
```

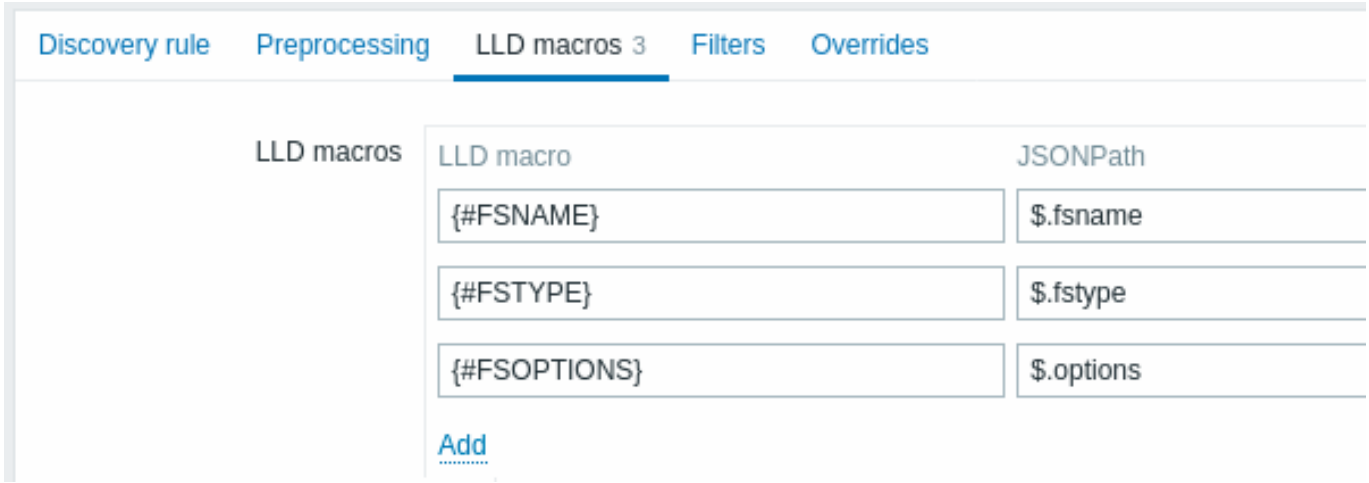
Zależna reguła LLD

Utwórz regułę wykrywania niskiego poziomu typu „Dependent item”:

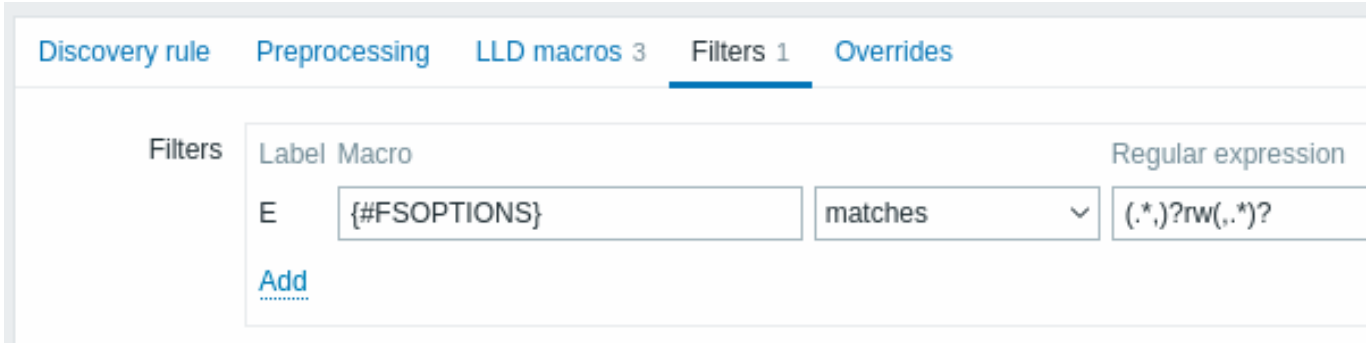


Jako pozycję nadrzędną wybierz utworzoną przez nas pozycję `vfs.fs.get`.

Na karcie „LLD macros” zdefiniuj własne makra z odpowiadającymi im ścieżkami JSONPath:

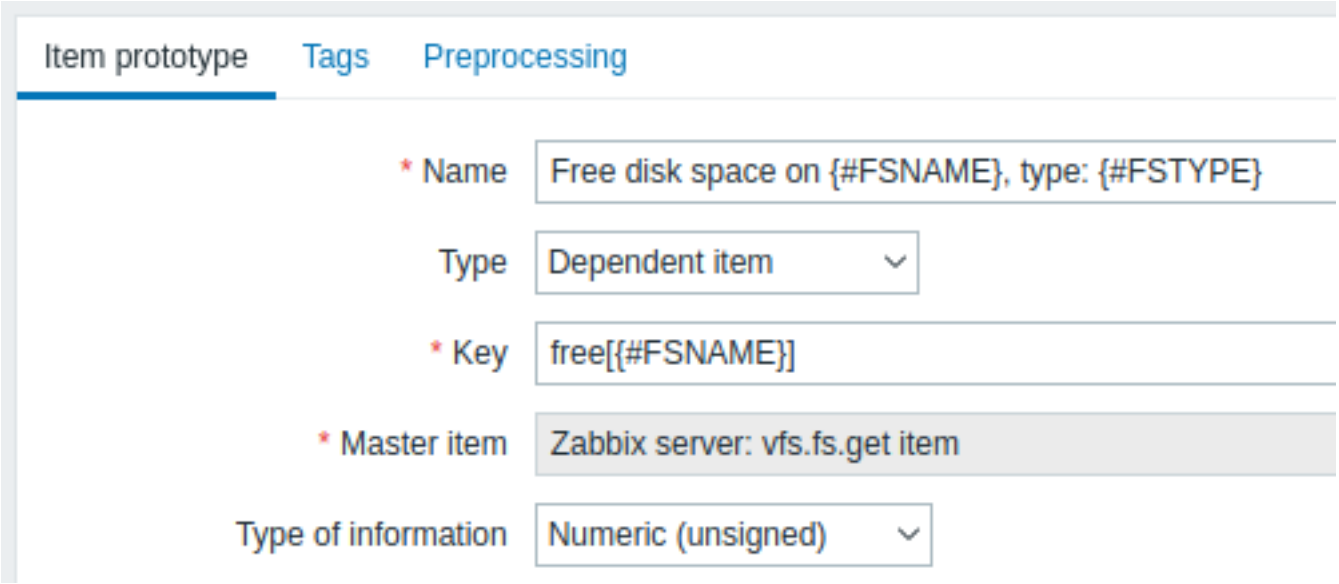


Na karcie „Filters” możesz dodać wyrażenie regularne, które odfiltruje tylko systemy plików **do odczytu i zapisu**:



Zależna pozycja prototypu

Utwórz prototyp pozycji z typem „Dependent item” w tej regule LLD. Jako pozycję nadrzędną dla tego prototypu wybierz utworzoną przez nas pozycję `vfs.fs.get`.



Zwróć uwagę na użycie własnych makr w nazwie i kluczu prototypu pozycji:

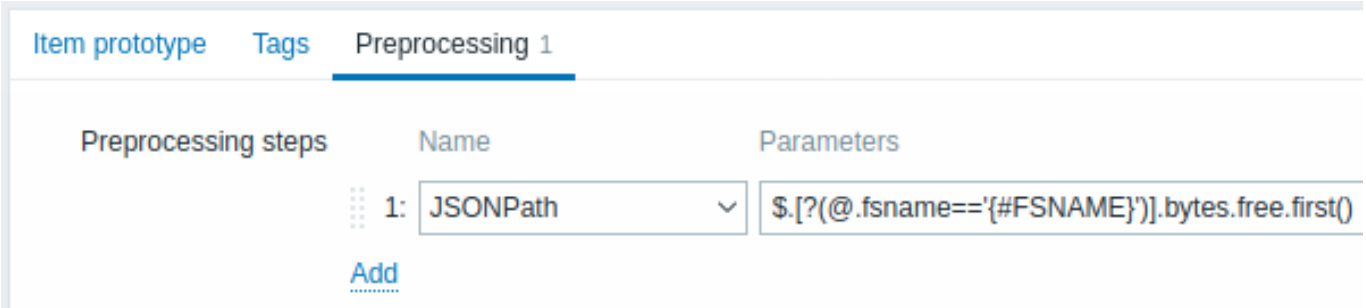
- *Nazwa*: Wolne miejsce na dysku na {#FSNAME}, typ: {#FSTYPE}
- *Klucz*: Free[{#FSNAME}]

Jako typ informacji użyj:

- *Numeric (unsigned)* dla metryk takich jak „free”, „total”, „used”
- *Numeric (float)* dla metryk takich jak „pfree”, „pused” (procent)

Na karcie „Preprocessing” prototypu pozycji wybierz JSONPath i użyj następującego wyrażenia JSONPath jako parametru:

```
$.[?(@.fsname=='{#FSNAME}')].bytes.free.first()
```



Po rozpoczęciu wykrywania zostanie utworzona jedna pozycja dla każdego punktu montowania. Ta pozycja będzie zwracała liczbę wolnych bajtów dla danego punktu montowania.

2 Wykrywanie interfejsów sieciowych

Podobnie jak wykrywane są systemy plików, możliwe jest również wykrywanie interfejsów sieciowych.

Klucz pozycji

Klucz pozycji używany w regule odkrywania to

```
net.if.discovery
```

Obsługiwane makra

Możesz użyć makra {#IFNAME} w filtrze reguły wykrywania oraz w prototypach pozycji, wyzwalaczy i wykresów.

Przykłady prototypów pozycji, które możesz chcieć utworzyć na podstawie ”net.if.discovery”:

- ”net.if.in[{#IFNAME},bytes]”,
- ”net.if.out[{#IFNAME},bytes]”.

Zwróć uwagę, że w systemie Windows zwracane jest również {#IFGUID}.

3 Wykrywanie procesorów CPU i rdzeni CPU

Podobnie jak wykrywane są systemy plików, możliwe jest również wykrywanie procesorów CPU i rdzeni CPU.

Klucz pozycji

Klucz pozycji używany w regule odkrywania to

`system.cpu.discovery`

Obsługiwane makra

Ten klucz wykrywania zwraca dwa makra - {#CPU.NUMBER} oraz {#CPU.STATUS}, identyfikujące odpowiednio numer porządkowy CPU i jego stan. Wylicza procesory od 0 do N - 1, gdzie N oznacza całkowitą liczbę procesorów.

```
[
    {
        "{#CPU.NUMBER}": 0,
        "{#CPU.STATUS}": "online"
    },
    {
        "{#CPU.NUMBER}": 1,
        "{#CPU.STATUS}": "offline"
    },
    {
        "{#CPU.NUMBER}": 2,
        "{#CPU.STATUS}": "unknown" /* "unknown" występuje tylko w systemie Windows */
    },
    {
        "{#CPU.NUMBER}": 3,
        "{#CPU.STATUS}": "online"
    }
]
```

Należy pamiętać, że nie da się jednoznacznie rozróżnić rzeczywistych, fizycznych procesorów, rdzeni i wątków hyper-threading. {#CPU.STATUS} w systemach Linux, UNIX i BSD zwraca stan procesora, który może mieć wartość ”online” lub ”offline”. W systemach Windows to samo makro może przyjmować trzecią wartość - ”unknown” - co oznacza, że procesor został wykryty, ale nie zebrano jeszcze dla niego żadnych informacji.

Wykrywanie CPU opiera się na procesie collectora agenta, aby zachować spójność z danymi dostarczanymi przez collector i oszczędzać zasoby podczas pozyskiwania danych. W efekcie ten klucz pozycji nie działa z flagą wiersza poleceń test (-t) pliku binarnego agenta, która zwróci stan NOT_SUPPORTED oraz towarzyszący komunikat wskazujący, że proces collectora nie został uruchomiony.

Prototypy pozycji, które można utworzyć na podstawie wykrywania CPU, obejmują na przykład:

- `system.cpu.util[{#CPU.NUMBER},<type>,<mode>]`
- `system.hw.cpu[{#CPU.NUMBER},<info>]`

Szczegółowy opis kluczy pozycji można znaleźć w sekcji Klucze pozycji agenta Zabbix.

Uwagi dotyczące wykrytych pozycji liczników wydajności w systemach NUMA

Niektóre liczniki wydajności Windows mogą nie być dostępne dla części procesorów logicznych w systemach NUMA.

Na przykład pozycje wykryte za pomocą prototypu pozycji z poniższym kluczem mogą działać tylko dla pierwszego węzła NUMA. Pozycje dla pozostałych węzłów NUMA mogą być w stanie nieobsługiwanym.

`perf_counter[\Processor({#CPU.NUMBER})\% Processor Time,60]`

Ponadto reguła wykrywania z kluczem ”system.cpu.discovery” nie dostarcza żadnych wartości makr wykrywania niskiego poziomu, które można by podstawić do takich liczników wydajności jak:

`perf_counter["\Processor Information(<NUMA node index>,<CPU index in NUMA node>)\% Processor Time",60]`

Natomiast pozycje system.cpu.util opierają się na innych licznikach wydajności do monitorowania procentowego wykorzystania CPU, w zależności od liczby grup procesorów i liczby procesorów. Występuje również rzadki błąd związany z grupami procesorów w Windows.

4 Odkrywanie OID-ów SNMP

Przegląd

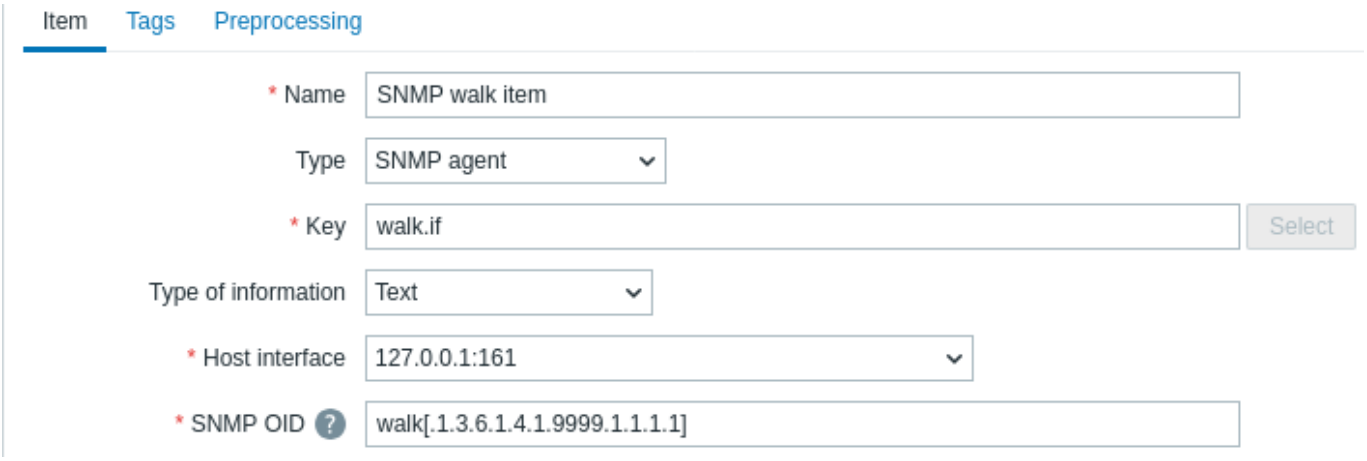
W tej sekcji przeprowadzimy odkrywanie niskiego poziomu na urządzeniu SNMP.

Ta metoda odkrywania OID SNMP jest obsługiwana od wersji 6.4 serwera/proxy Zabbix.

Przykładowa konfiguracja

1. Utwórz pozycję agenta SNMP z kluczem takim jak:

```
walk[.1.3.6.1.4.1.9999.1.1.1.1]
```



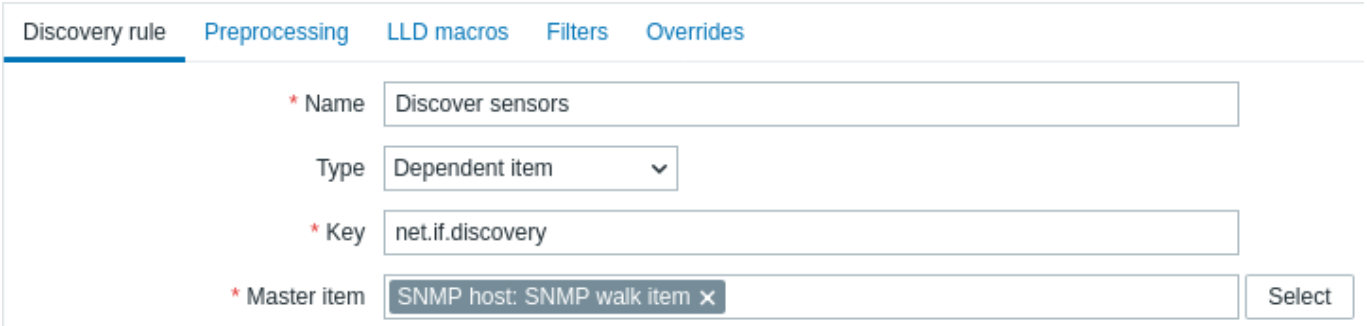
Ta pozycja wykonuje pojedyncze przejście po tabeli SNMP i zwraca wszystkie wpisy tabeli w jednym żądaniu, w formacie odpowiadającym danym wyjściowym narzędzia `snmpwalk` z opcjami formatowania `-Oe -Ot -On`.

Zwróci następującą wielowierszową wartość tekstową:

```
.1.3.6.1.4.1.9999.1.1.1.1.1.1 = STRING: "Temperature Sensor"
.1.3.6.1.4.1.9999.1.1.1.1.2.1 = STRING: "temp"
.1.3.6.1.4.1.9999.1.1.1.1.3.1 = 100
.1.3.6.1.4.1.9999.1.1.1.1.1.2 = STRING: "Humidity Sensor"
.1.3.6.1.4.1.9999.1.1.1.1.2.2 = STRING: "humidity"
.1.3.6.1.4.1.9999.1.1.1.1.3.2 = 200
```

2. Utwórz regułę wykrywania:

- W polu *Name* wprowadź opisową nazwę reguły wykrywania (np. „Wykryj czujniki”).
- W polu *Type* wybierz „Dependent item”.
- W polu *Key* wprowadź opisowy klucz (np. „net.if.discovery”).
- W polu *Master item* wybierz „SNMP walk item”.



3. Na karcie *Preprocessing* dodaj krok przetwarzania wstępnego z opcją „SNMP walk to JSON” wybraną na liście rozwijanej *Name* i z 3 parametrami:

- *Field name*: ”{#SENSORNAME}”; *OID prefix*: ”.1.3.6.1.4.1.9999.1.1.1.1.1”: *Format*: ”Unchanged”.
- *Field name*: ”{#SENSORTYPE}”; *OID prefix*: ”.1.3.6.1.4.1.9999.1.1.1.1.2”: *Format*: ”Unchanged”.
- *Field name*: ”{#SENSORVALUE}”; *OID prefix*: ”.1.3.6.1.4.1.9999.1.1.1.1.3”: *Format*: ”Unchanged”.

Po przetwarzaniu wstępnym reguła wykrywania zwraca tablicę JSON zestawów makr.

Na przykład:

```
[
    {
        "{#SNMPINDEX}": "1",
        "{#SENSORNAME}": "Temperature Sensor",
        "{#SENSORTYPE}": "temp",
        "{#SENSORVALUE}": "100"
    },
    {
        "{#SNMPINDEX}": "2",
        "{#SENSORNAME}": "Humidity Sensor",
        "{#SENSORTYPE}": "humidity",
        "{#SENSORVALUE}": "200"
    }
]
```

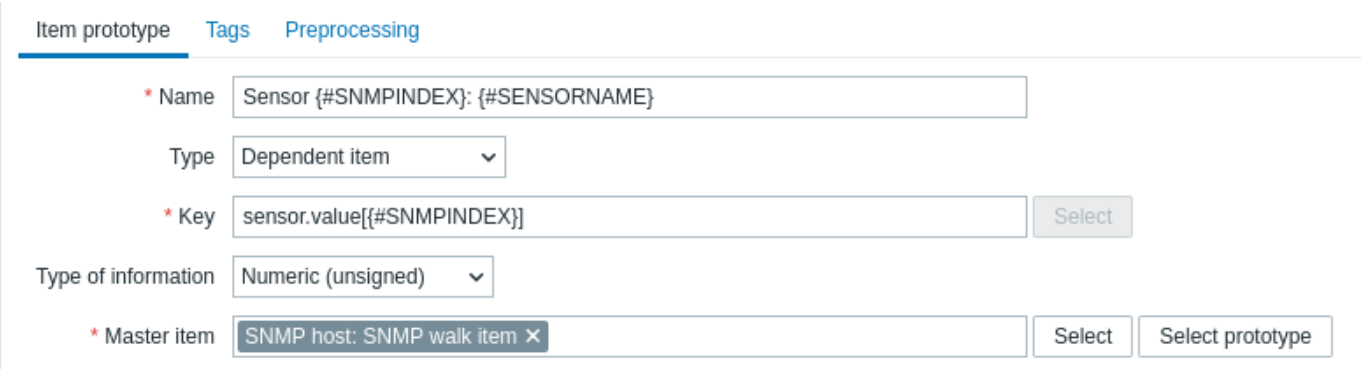
Każdy obiekt reprezentuje jeden wykryty czujnik i udostępnia makra takie jak `{#SNMPINDEX}`, `{#SENSORNAME}`, `{#SENSORTYPE}` oraz `{#SENSORVALUE}`.

Są one grupowane według indeksu SNMP, który jest numerycznym sufiksem na końcu każdego OID (np. .1, .2) — ten indeks jednoznacznie identyfikuje każdy wiersz w tabeli SNMP i jest automatycznie wyodrębniany jako `{#SNMPINDEX}`.

4. W ramach reguły wykrywania utwórz jeden lub więcej prototypów pozycji (z regułą wykrywania jako pozycją nadrzędną).

Na przykład zależna pozycja wartości czujnika:

- W polu *Name* wprowadź „Sensor {#SNMPINDEX}: {#SENSORNAME}”.
- W polu *Type* wybierz „Dependent item”.
- W polu *Key* wprowadź „sensor.value[{#SNMPINDEX}]”.
- W polu *Master item* wybierz „SNMP walk item”.



Na karcie *Preprocessing* dodaj krok przetwarzania wstępnego o nazwie „SNMP walk value” z OID ”.1.3.6.1.4.1.9999.1.1.1.1.3.{#SNMPINDEX}” w polu *Parameter*. *Format*: ”Unchanged”.

Zostaną wykryte następujące pozycje:

Name	Key	OID from which value is extracted	Item value
Sensor 1: Temperature Sensor	sensor.value[1]	1.3.6.1.4.1.9999.1.1.1.1.3.1	100
Sensor 2: Humidity Sensor	sensor.value[2]	1.3.6.1.4.1.9999.1.1.1.1.3.2	200

Po uruchomieniu reguły wykrywania zostaną utworzone pozycje takie jak `sensor.value[1]`, `sensor.value[2]`.

Każda pozycja zależna wyodrębnia swoją wartość z wyniku SNMP walk pozycji nadrzędnej przy użyciu przetwarzania wstępnego, bez wykonywania osobnych żądań SNMP.

5. Odwołuj się do prototypów pozycji zależnych w prototypach wyzwalaczy, używając tych samych makr z reguły wykrywania. Przykład:

```
{Template_Sensor:sensor.value[{#SNMPINDEX}].last()} > 75
```

Powoduje to utworzenie wyzwalacza dla każdego wykrytego czujnika (na przykład sensor.value[1], sensor.value[2]) i jego uruchomienie, jeśli najnowsza wartość (temperatury lub wilgotności) przekroczy 75.

6. Uwzględnij pozycje zależne dla każdej wykrytej jednostki. Przykładowy klucz elementu wykresu:

```
sensor.value[{#SNMPINDEX}]
```

Dla każdego {#SNMPINDEX} tworzony jest jeden wykres, przedstawiający temperaturę i wilgotność w czasie.

Ta konfiguracja wykonuje tylko jedno żądanie SNMP walk na cykl odpytywania, niezależnie od liczby wykrytych pozycji. Wszystkie pozycje zależne wyodrębniają swoje wartości z nadrzędnego wyniku SNMP walk przy użyciu przetwarzania wstępnego, co znacząco zmniejsza ruch SNMP i obciążenie.

Dynamiczne indeksy z walk[]

Dynamiczne indeksy (na przykład indeksy interfejsów) mogą się zmieniać po rekonfiguracji sprzętu. Aby uwzględnić takie zachowanie, tworzona jest nadrzędna reguła wykrywania SNMP walk z kluczem takim jak:

```
walk[1.3.6.1.2.1.2.2.1.10]
```

Po przetwarzaniu wstępnym SNMP walk do JSON wynik może wyglądać następująco:

```
[
    {
        "{#SNMPINDEX}": "2",
        "{#VALUE}": "123456"
    },
    {
        "{#SNMPINDEX}": "3",
        "{#VALUE}": "654321"
    }
]
```

Prototyp pozycji zależnej używa makra {#SNMPINDEX} do zbudowania klucza:

```
net.if.in[{#SNMPINDEX}]
```

Przetwarzanie wstępne dla tego prototypu obejmuje nazwę „SNMP walk value" z OID „1.3.6.1.2.1.2.2.1.10.{#SNMPINDEX}" w polu *Parameter*. *Format*: „Unchanged".

W czasie działania tworzone są rzeczywiste pozycje, takie jak `net.if.in[2]` i `net.if.in[3]`. Jeśli dany indeks interfejsu ulegnie zmianie (na przykład jeśli indeks 2 zostanie zastąpiony przez 5 w tabeli SNMP), to przy następnym uruchomieniu reguły wykrywania:

- Stara zależna pozycja net.if.in[2] zostanie oznaczona jako „lost" lub usunięta i nie będą już zbierane dla niej nowe dane.
- Zostanie utworzona nowa zależna pozycja net.if.in[5], rozpoczynająca z pustą historią.
- Dane historyczne z net.if.in[2] nie są automatycznie przenoszone do net.if.in[5].

Przykład prototypu wyzwalacza:

```
{Template_Interface:net.if.in[{#SNMPINDEX}].last()} > 1000000000
```

Przykład prototypu wykresu obejmuje pozycje:

```
net.if.in[{#SNMPINDEX}]
net.if.out[{#SNMPINDEX}]
```

Ta konfiguracja zapewnia niezawodne monitorowanie tabel z dynamicznymi indeksami przy jednoczesnym zminimalizowaniu ruchu SNMP — wymagany jest tylko jeden SNMP walk na cykl odpytywania, a prototypy pozycji zależnych wyodrębniają potrzebne wartości.

Wykryte encje

Gdy serwer jest uruchomiony, utworzy rzeczywiste zależne pozycje, wyzwalacze i wykresy na podstawie wartości zwracanych przez regułę wykrywania SNMP.

5 Odkrywanie OID-ów SNMP (starsza wersja)

Przegląd

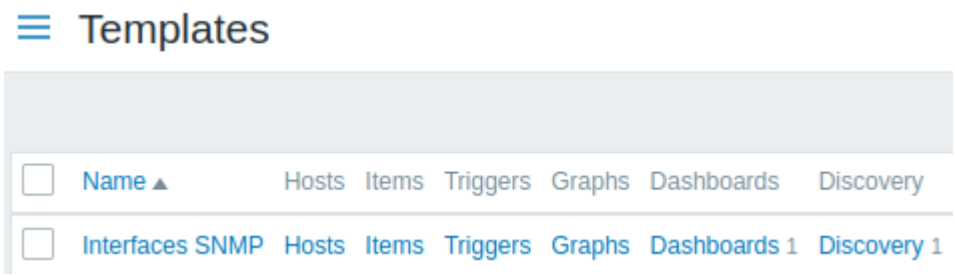
W tej sekcji przeprowadzimy wykrywanie SNMP na przełączniku.

Klucz pozycji

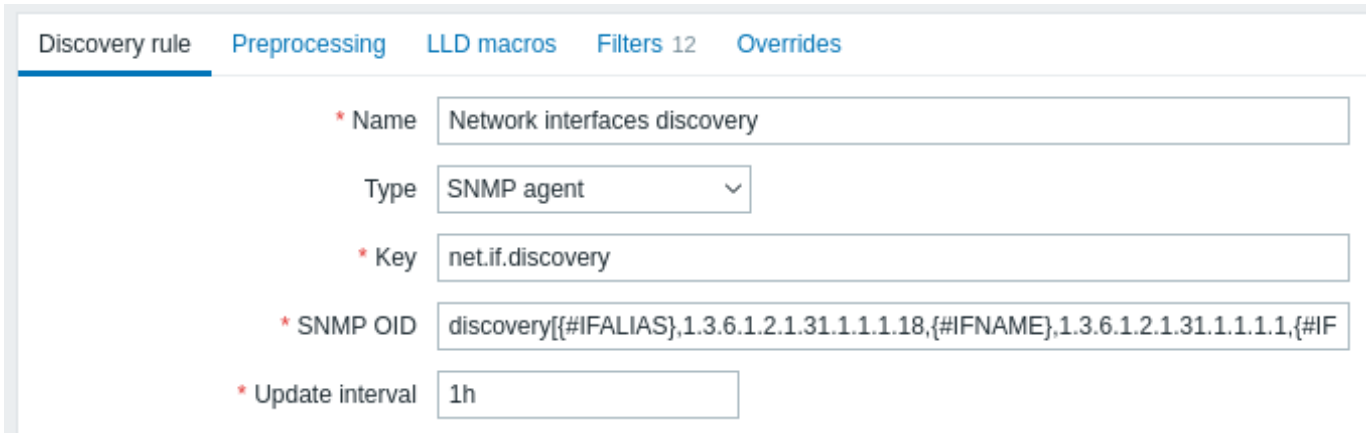
W przeciwieństwie do wykrywania systemów plików i interfejsów sieciowych, pozycja niekoniecznie musi mieć klucz "snmp.discovery" — wystarczy typ pozycji SNMP agent.

Aby skonfigurować regułę wykrywania, wykonaj następujące kroki:

- Przejdź do: *Data collection* → *Templates*
- Kliknij *Discovery* w wierszu odpowiedniego szablonu



- Kliknij *Create discovery rule* w prawym górnym rogu ekranu
- Wypełnij formularz reguły wykrywania wymaganymi danymi, jak pokazano na zrzucie ekranu poniżej



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

OID-y do wykrycia są definiowane w polu SNMP OID w następującym formacie: `discovery[{#MACRO1}, oid1, {#MACRO2}, oid2, …,]`

gdzie *{#MACRO1}*, *{#MACRO2}* ... są prawidłowymi nazwami makr lld, a *oid1*, *oid2*... są OID-ami zdolnymi do generowania znaczących wartości dla tych makr. Wbudowane makro *{#SNMPINDEX}*, zawierające indeks wykrytego OID, jest stosowane do wykrytych encji. Wykryte encje są grupowane według wartości makra *{#SNMPINDEX}*.

Oto przykład, w którym na przełączniku wykonywanych jest kilka poleceń snmpwalk:

```
$ 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
```

A SNMP OID jest ustawiony na: `discovery[{#IFDESCR}, ifDescr, {#IFPHYSADDRESS}, ifPhysAddress]`

Teraz ta reguła wykryje encje z makrami {#IFDESCR} ustawionymi na **WAN**, **LAN1** i **LAN2**, makrami {#IFPHYSADDRESS} ustawionymi na **8:0:27:90:7a:75**, **8:0:27:90:7a:76** oraz **8:0:27:2b:af:9e**, a makrami {#SNMPINDEX} ustawionymi na indeksy wykrytych OID-ów **1**, **2** i **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"
    }
]
```

Jeśli encja nie ma określonego OID, odpowiednie makro zostanie pominięte dla tej encji. Na przykład, jeśli mamy następujące dane:

```
ifDescr.1 "Interface #1"
ifDescr.2 "Interface #2"
ifDescr.4 "Interface #4"

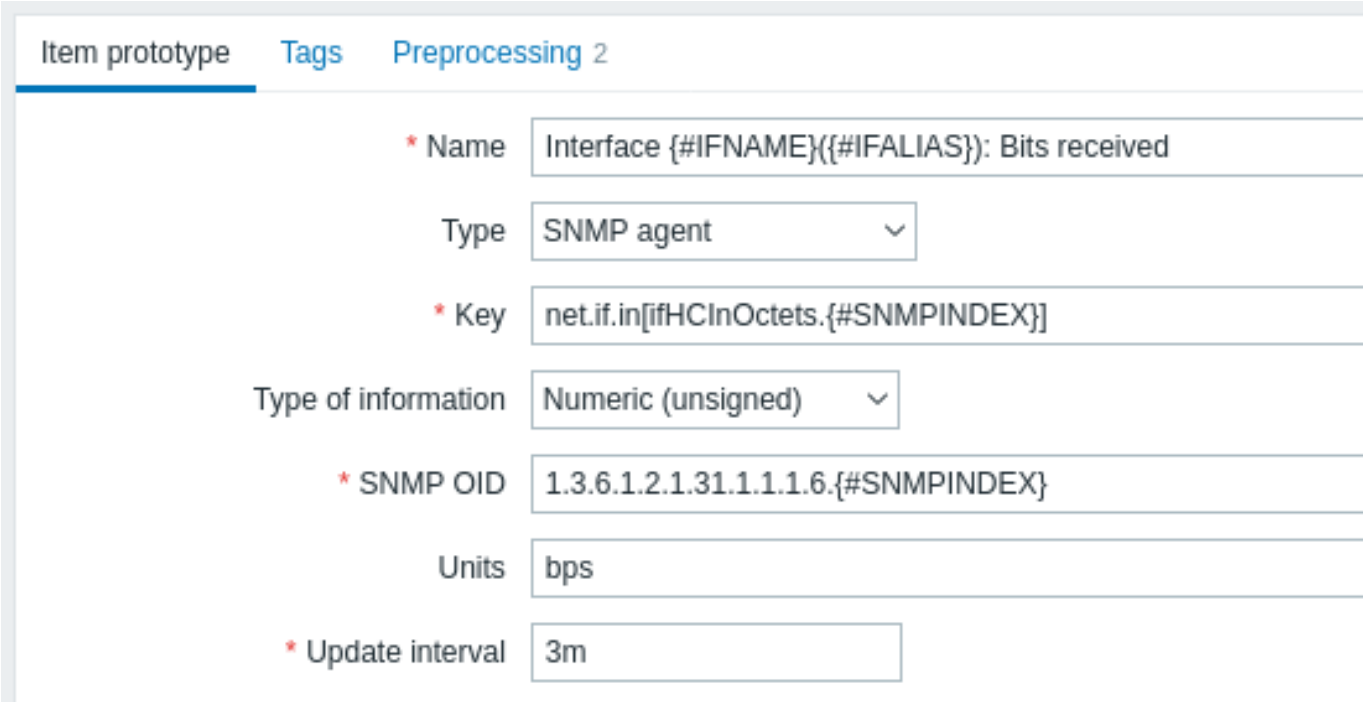
ifAlias.1 "eth0"
ifAlias.2 "eth1"
ifAlias.3 "eth2"
ifAlias.5 "eth4"
```

W takim przypadku wykrywanie SNMP `discovery[{#IFDESCR}, ifDescr, {#IFALIAS}, ifAlias]` zwróci następującą strukturę:

```
[
    {
        "{#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"
    }
]
```

Prototypy pozycji

Poniższy zrzut ekranu pokazuje, jak można używać tych makr w prototypach pozycji:



Możesz utworzyć tyle prototypów pozycji, ile potrzeba:

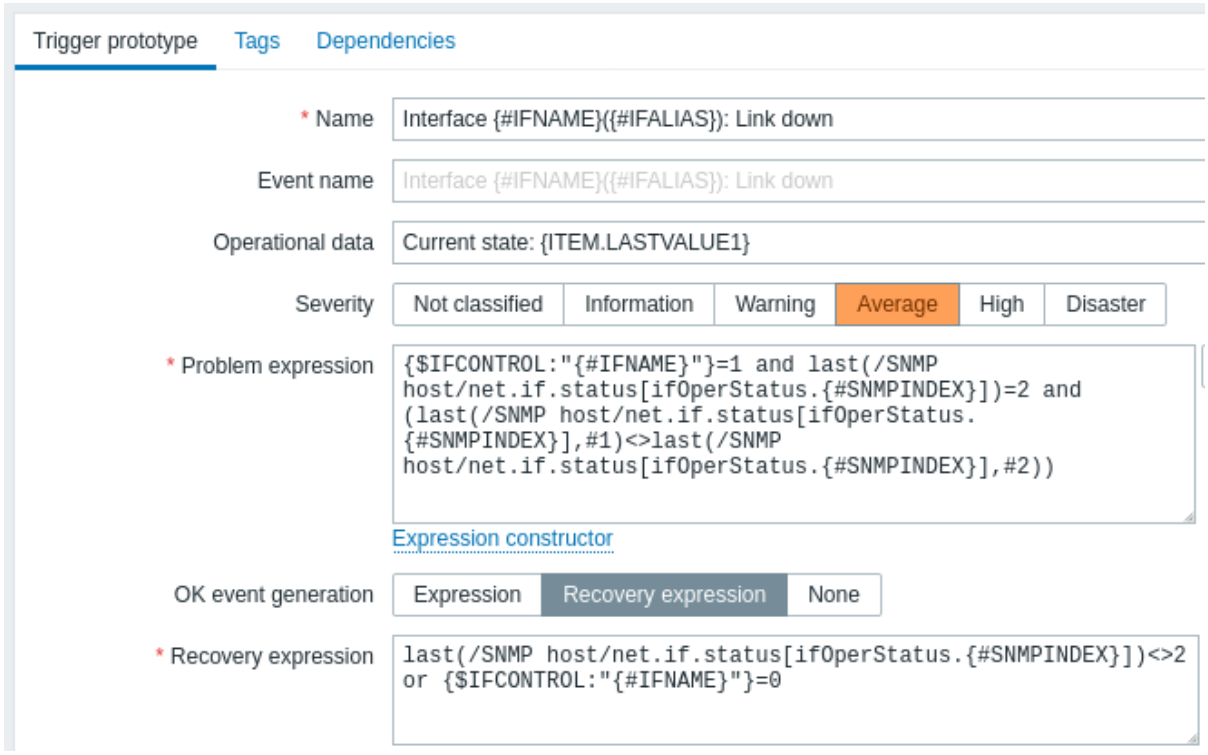
Item prototypes

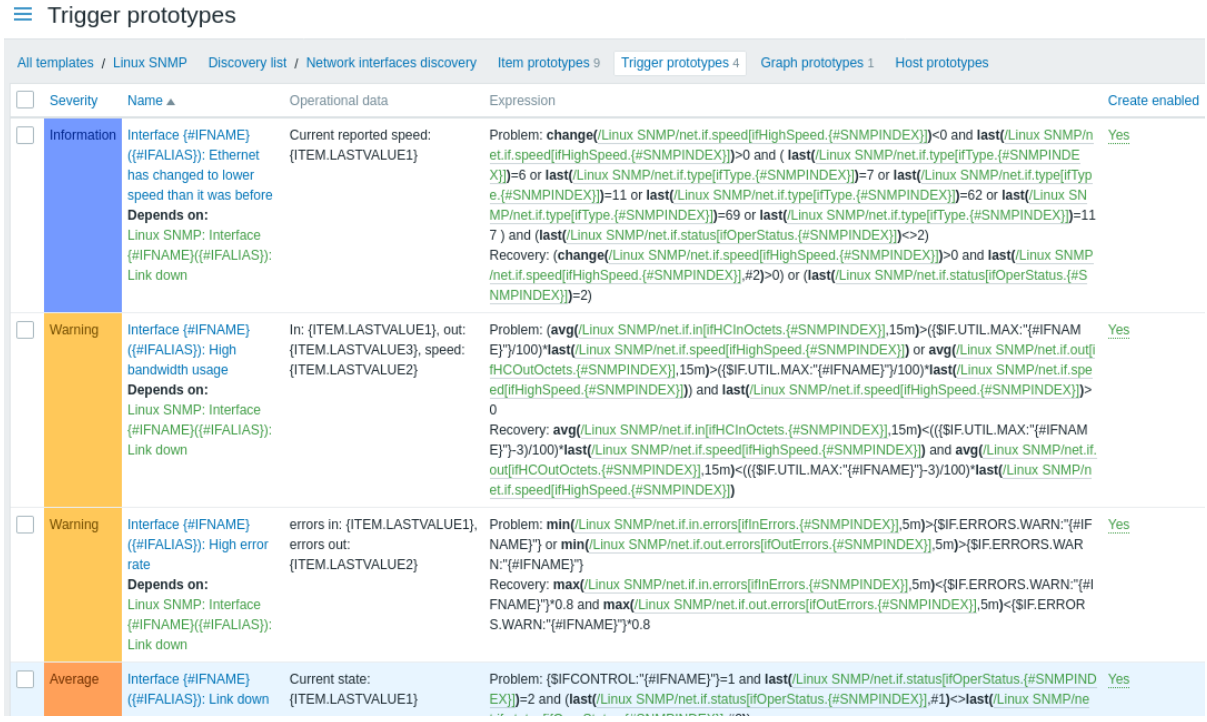
All templates / Linux SNMP Discovery list / Network interfaces discovery Item prototypes 9 Trigger prototypes 4 Graph prototypes 1 Host prototypes

Name ▲	Key	Interval	History	Trends	Type	Create enabled
Interface {#IFNAME}({#IFALIAS}): Bits received	net.if.in[ifHCInOctets.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Bits sent	net.if.out[ifHCOutOctets.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Inbound packets discarded	net.if.in.discards[ifInDiscards.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Inbound packets with errors	net.if.in.errors[ifInErrors.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Interface type	net.if.type[ifType.{#SNMPINDEX}]	1h	7d	0d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Operational status	net.if.status[ifOperStatus.{#SNMPINDEX}]	1m	7d	0	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Outbound packets discarded	net.if.out.discards[ifOutDiscards.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Outbound packets with errors	net.if.out.errors[ifOutErrors.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes
Interface {#IFNAME}({#IFALIAS}): Speed	net.if.speed[ifHighSpeed.{#SNMPINDEX}]	5m	7d	0d	SNMP agent	Yes

Prototypy wyzwalaczy

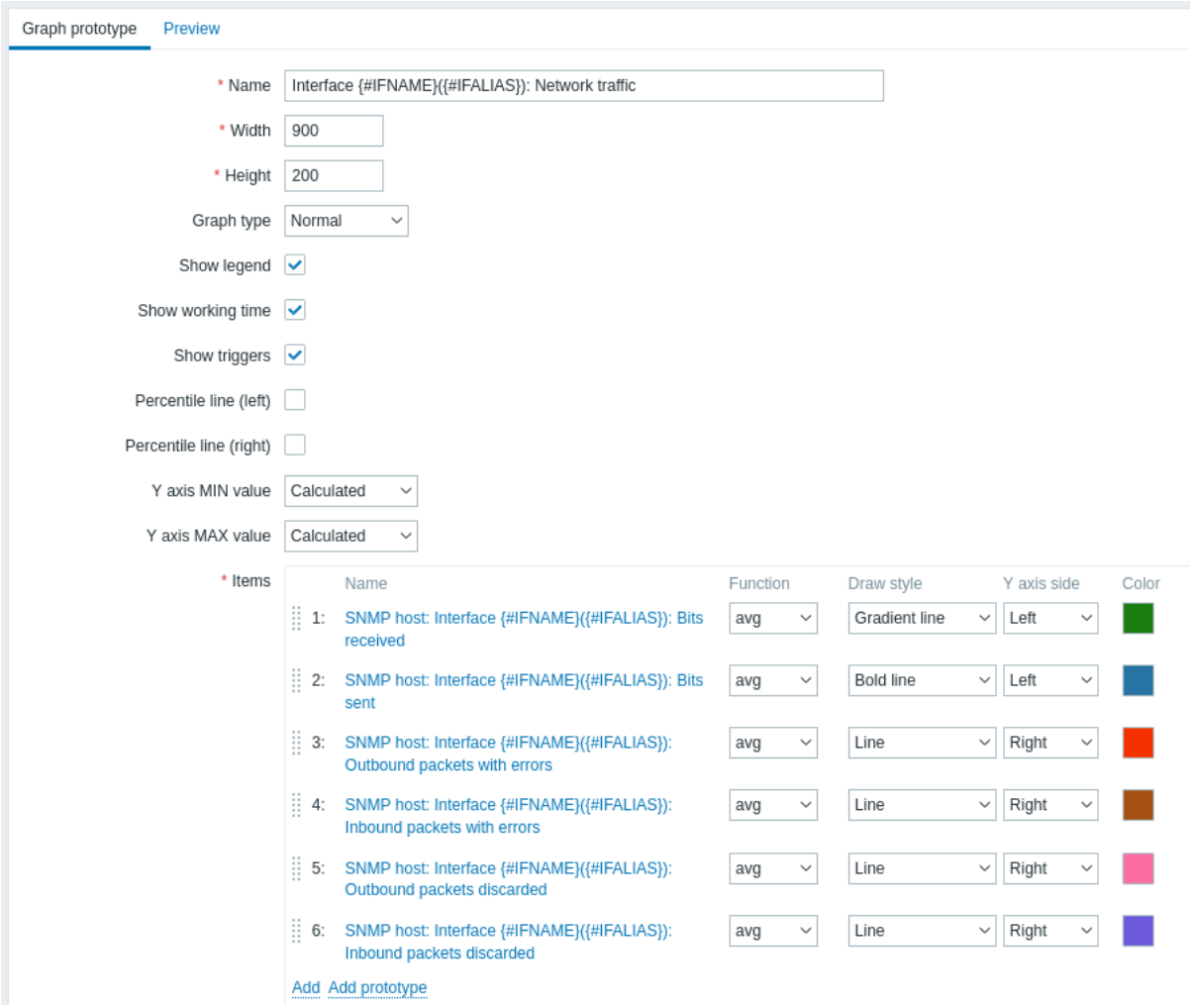
Poniższy zrzut ekranu ilustruje, jak możemy używać tych makr w prototypach wyzwalaczy:

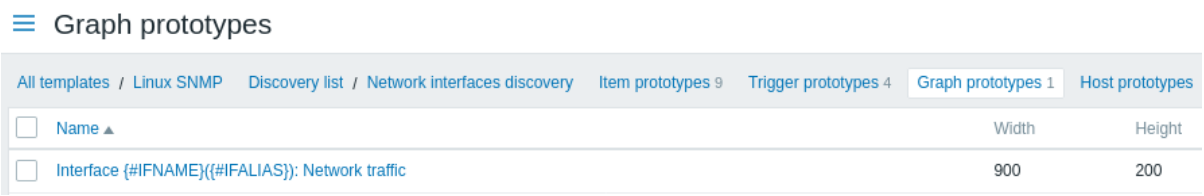




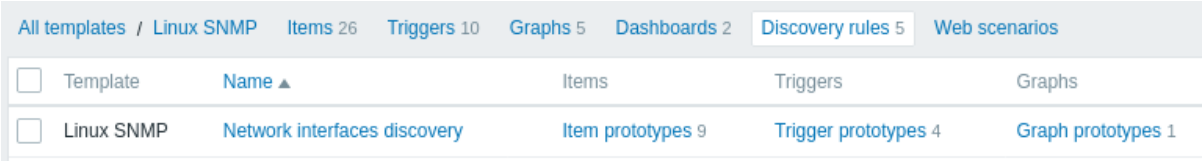
Prototypy wykresów

Poniższy zrzut ekranu pokazuje, jak możemy używać tych makr w prototypach wykresów:





Podsumowanie naszej reguły wykrywania:



Wykryte encje

Gdy serwer jest uruchomiony, utworzy rzeczywiste pozycje, wyzwalacze i wykresy na podstawie wartości zwracanych przez regułę wykrywania SNMP. W konfiguracji hosta są one poprzedzone pomarańczowym odnośnikiem do reguły wykrywania, z której pochodzą.

Items

All hosts / SNMP host Enabled SNMP Items 81 Triggers 23 Graphs 14 Discovery rules 6 Web scenarios

Name ▲	Triggers	Key	Interval	History	Trends	Type	Status
Network interfaces discovery: Interface enp4s0(): Bits received	Triggers 1	net.if.in[ifHCInOctets.2]	3m	7d	365d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Bits sent	Triggers 1	net.if.out[ifHCOutOctets.2]	3m	7d	365d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Inbound packets discarded		net.if.in.discards[ifInDiscards.2]	3m	7d	365d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Inbound packets with errors	Triggers 1	net.if.in.errors[ifInErrors.2]	3m	7d	365d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Interface type	Triggers 1	net.if.type[ifType.2]	1h	7d	0d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Operational status	Triggers 2	net.if.status[ifOperStatus.2]	1m	7d	0	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Outbound packets discarded		net.if.out.discards[ifOutDiscards.2]	3m	7d	365d	SNMP agent	Enabled
Network interfaces discovery: Interface enp4s0(): Outbound packets with errors	Triggers 1	net.if.out.errors[ifOutErrors.2]	3m	7d	365d	SNMP agent	Enabled
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

Severity	Value	Name ▲	Operational data	Expression
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)
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]**)**
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
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

6 Wykrywanie obiektów JMX

Przegląd

Możliwe jest wykrywanie wszystkich JMX MBeanów lub atrybutów MBeanów albo określenie wzorca do wykrywania tych obiektów.

Dla konfiguracji reguły wykrywania konieczne jest zrozumienie różnicy między MBeanem a atrybutami MBeana. MBean jest obiektem, który może reprezentować urządzenie, aplikację lub dowolny zasób, który wymaga zarządzania.

Na przykład istnieje MBean reprezentujący serwer WWW. Jego atrybutami są liczba połączeń, liczba wątków, limit czasu żądania, pamięć podręczna plików HTTP, użycie pamięci itp. Wyrażając tę myśl w języku bardziej zrozumiałym dla człowieka, możemy zdefiniować ekspres do kawy jako MBean, który ma następujące atrybuty do monitorowania: ilość wody na filiżankę, średnie zużycie wody w określonym czasie, liczba ziaren kawy wymagana na filiżankę, czas uzupełniania ziaren kawy i wody itp.

Klucz pozycji

W konfiguracji reguły odkrywania wybierz **JMX agent** w polu *Typ*.

Dla odkrywania obiektów JMX obsługiwane są dwa klucze pozycji - jmx.discovery[] oraz jmx.get[]:

Klucz pozycji	Wartość zwracana	Parametry	Komentarz
jmx.discovery[<tryb odkrywania>,<nazwa obiektu>,<unikalny krótki opis>]	Ta pozycja zwraca tablicę JSON z makrami LLD opisującymi obiekty MBean lub ich atrybuty.	**tryb odkrywania** - jedna z następujących wartości: *attributes* (pobierz atrybuty JMX MBean, domyślnie) lub *beans* (pobierz JMX MBeans) **nazwa obiektu** - wzorzec nazwy obiektu (zobacz documentation) identyfikujący nazwy MBean do pobrania (domyślnie puste, co powoduje pobranie wszystkich zarejestrowanych beanów) **unikalny krótki opis** - unikalny opis umożliwiający użycie wielu pozycji JMX z tym samym trybem odkrywania i nazwą obiektu na hoście (opcjonalnie)	Przykłady: → jmx.discovery - pobierz wszystkie atrybuty JMX MBean → jmx.discovery[beans] - pobierz wszystkie JMX MBeans → jmx.discovery[attributes,"*:type=GarbageCollector,name - pobierz wszystkie atrybuty garbage collectora → jmx.discovery[beans,"*:type=GarbageCollector,name=* - pobierz wszystkie garbage collectory Istnieją pewne ograniczenia dotyczące tego, jakie właściwości MBean może zwrócić ta pozycja, ze względu na ograniczony zestaw znaków obsługiwanych przy generowaniu nazw makr (obsługiwane znaki można opisać za pomocą następującego wyrażenia regularnego: `A-Z0-9_\.`). Dlatego na przykład, aby odkrywać właściwości MBean zawierające słowo z łącznikiem lub znaki spoza ASCII, należy użyć `jmx.get[]`.
jmx.get[<tryb odkrywania>,<nazwa obiektu>,<unikalny krótki opis>]			

Klucz pozycji			
	Ta pozycja zwraca tablicę JSON z obiektami MBean lub ich atrybutami. W porównaniu do `jmx.discovery[]` nie definiuje makr LLD.	**tryb odkrywania** - jedna z następujących wartości: *attributes* (pobierz atrybuty JMX MBean, domyślnie) lub *beans* (pobierz JMX MBeans) **nazwa obiektu** - wzorzec nazwy obiektu (zobacz documentation) identyfikujący nazwy MBean do pobrania (domyślnie puste, co powoduje pobranie wszystkich zarejestrowanych beanów) **unikalny krótki opis** - unikalny opis umożliwiający użycie wielu pozycji JMX z tym samym trybem odkrywania i nazwą obiektu na hoście (opcjonalnie)	Podczas używania tej pozycji należy zdefiniować własne makra odkrywania niskiego poziomu, wskazujące wartości wyodrębnione ze zwróconego JSON przy użyciu JSONPath.

Attention:
Jeśli nie zostaną przekazane żadne parametry, zostaną pobrane wszystkie atrybuty MBean z JMX. Nieokreślenie parametrów dla odkrywania JMX lub próba pobrania wszystkich atrybutów dla szerokiego zakresu, takiego jak `*:type=*,name=*`, może prowadzić do potencjalnych problemów z wydajnością.

Użycie jmx.discovery

Ta pozycja zwraca obiekt JSON z makrami wykrywania niskiego poziomu opisującymi obiekty MBean lub atrybuty. Na przykład przy wykrywaniu atrybutów MBean (przeformatowano dla przejrzystości):

```
[
    {
        "{#JMXVALUE}":"0",
        "{#JMXTYPE}":"java.lang.Long",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,CollectionCount",
        "{#JMXATTR}":"CollectionCount"
    },
    {
        "{#JMXVALUE}":"0",
        "{#JMXTYPE}":"java.lang.Long",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,CollectionTime",
        "{#JMXATTR}":"CollectionTime"
    },
    {
        "{#JMXVALUE}":"true",
        "{#JMXTYPE}":"java.lang.Boolean",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,Valid",
        "{#JMXATTR}":"Valid"
    },
    {
        "{#JMXVALUE}":"PS Scavenge",
        "{#JMXTYPE}":"java.lang.String",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,Name",
        "{#JMXATTR}":"Name"
    },
    {
        "{#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}":"ObjectName"
    }
]
```

Przy wykrywaniu MBeanów (przeformatowano dla przejrzystości):

```
[
    {
        "{#JMXDOMAIN}":"java.lang",
        "{#JMXTYPE}":"GarbageCollector",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXNAME}":"PS Scavenge"
    }
]
```

Obsługiwane makra

Następujące makra są obsługiwane do użycia w filtrze reguły wykrywania oraz w prototypach pozycji, wyzwalaczy i wykresów:

Makro	Opis
Wykrywanie atrybutów MBean	
{#JMXVALUE}	Wartość atrybutu.
{#JMXTYPE}	Typ atrybutu.
{#JMXOBJ}	Nazwa obiektu.
{#JMXDESC}	Nazwa obiektu wraz z nazwą atrybutu.
{#JMXATTR}	Nazwa atrybutu.
Wykrywanie MBeanów	
{#JMXDOMAIN}	Domena MBean. (*Nazwa zastrzeżona przez Zabbix*)
{#JMXOBJ}	Nazwa obiektu. (*Nazwa zastrzeżona przez Zabbix*)
{#JMX<key property>}	Właściwości MBean (takie jak {#JMXTYPE}, {#JMXNAME}) (zobacz Ograniczenia poniżej).

Ograniczenia

Istnieją pewne ograniczenia związane z algorytmem tworzenia nazw makr LLD z nazw właściwości MBean:

- nazwy atrybutów są zmieniane na wielkie litery
- nazwy atrybutów są ignorowane (nie są generowane żadne makra LLD), jeśli składają się z znaków nieobsługiwanych w nazwach makr LLD. Obsługiwane znaki można opisać za pomocą następującego wyrażenia regularnego: `A-Z0-9_\.`
- jeśli atrybut ma nazwę ”obj” lub ”domain”, zostanie zignorowany z powodu nakładania się z wartościami zastrzeżonych właściwości Zabbix {#JMXOBJ} i {#JMXDOMAIN}

Proszę rozważyć ten przykład jmx.discovery (w trybie ”beans”). MBean ma zdefiniowane następujące właściwości (niektóre z nich zostaną zignorowane; patrz niżej):

```
name=test
  =Type
attributes []=1,2,3
Name=NameOfTheTest
domAin=some
```

W wyniku wykrywania JMX zostaną wygenerowane następujące makra LLD:

- {#JMXDOMAIN} - wewnętrzne Zabbix, opisujące domenę MBean
- {#JMXOBJ} - wewnętrzne Zabbix, opisujące obiekt MBean
- {#JMXNAME} - utworzone z właściwości ”name”

Ignorowane właściwości to:

- тип : jego nazwa zawiera nieobsługiwane znaki (spoza ASCII)
- attributes[] : jego nazwa zawiera nieobsługiwane znaki (nawiasy kwadratowe nie są obsługiwane)
- Name : jest już zdefiniowana (name=test)
- domAin : jest to nazwa zastrzeżona przez Zabbix

Przykłady

Przyjrzyjmy się jeszcze dwóm praktycznym przykładom tworzenia reguły LLD z wykorzystaniem MBean. Aby lepiej zrozumieć różnicę między regułą LLD zbierającą MBeans a regułą LLD zbierającą atrybuty MBean, prosimy spojrzeć na poniższą tabelę:

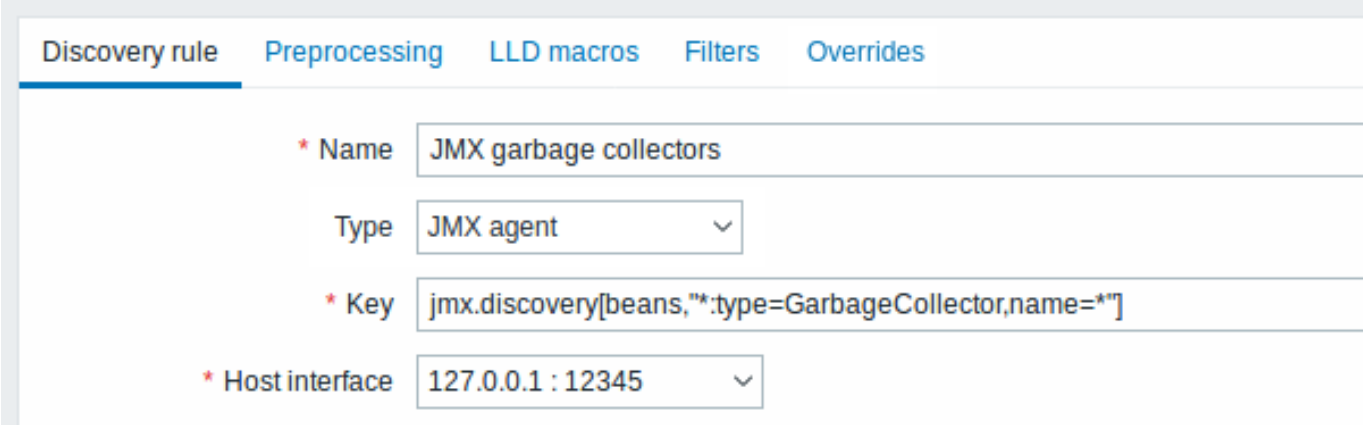
MBean1	**MBean2**	**MBean3**
MBean1Attribute1	MBean2Attribute1	MBean3Attribute1
MBean1Attribute2	MBean2Attribute2	MBean3Attribute2
MBean1Attribute3	MBean2Attribute3	MBean3Attribute3

Przykład 1: Odkrywanie MBeanów

Ta reguła zwróci 3 obiekty: górny wiersz kolumny: MBean1, MBean2, MBean3.

Aby uzyskać więcej informacji o obiektach, zobacz tabelę obsługiwanych makr w sekcji *Odkrywanie MBeanów*.

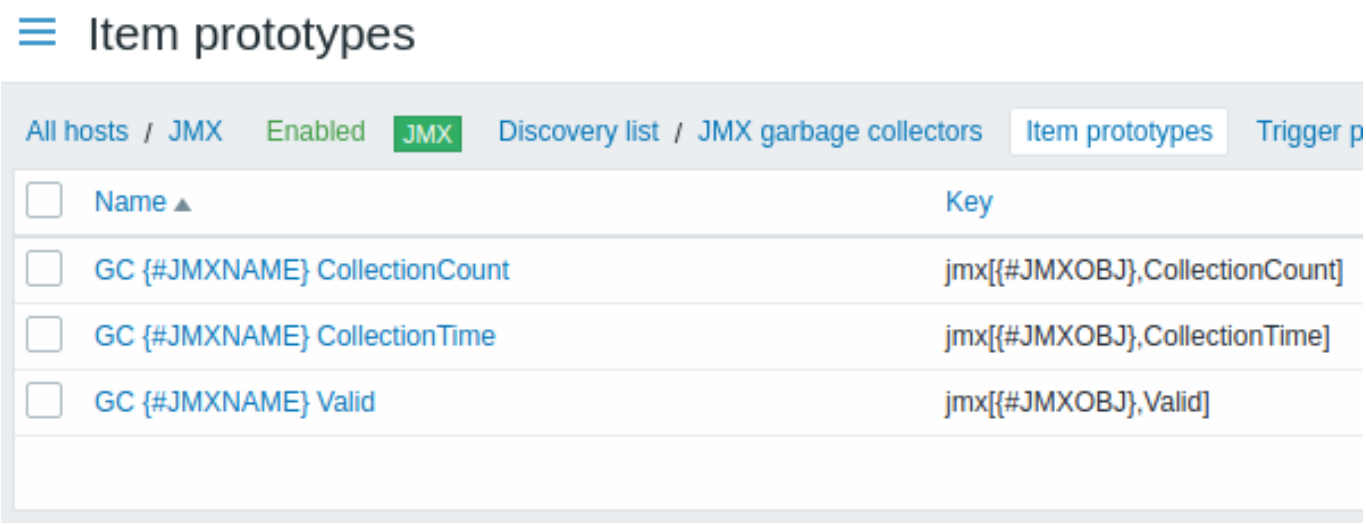
Konfiguracja reguły odkrywania zbierającej MBeany (bez atrybutów) wygląda następująco:



Klucz użyty tutaj:

```
jmx.discovery[beans,"*:type=GarbageCollector,name=*"]
```

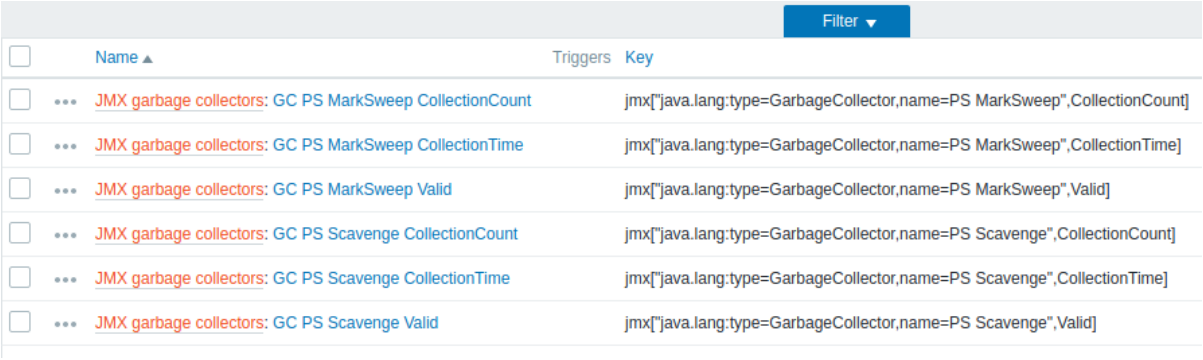
Wszystkie garbage collectory bez atrybutów zostaną odkryte. Ponieważ garbage collectory mają ten sam zestaw atrybutów, możemy użyć żądanych atrybutów w prototypach pozycji w następujący sposób:



Klucze użyte tutaj:

```
jmx[{#JMXOBJ},CollectionCount]
jmx[{#JMXOBJ},CollectionTime]
jmx[{#JMXOBJ},Valid]
```

Reguła odkrywania LLD da w rezultacie coś zbliżonego do tego (pozycje są odkrywane dla dwóch garbage collectorów):

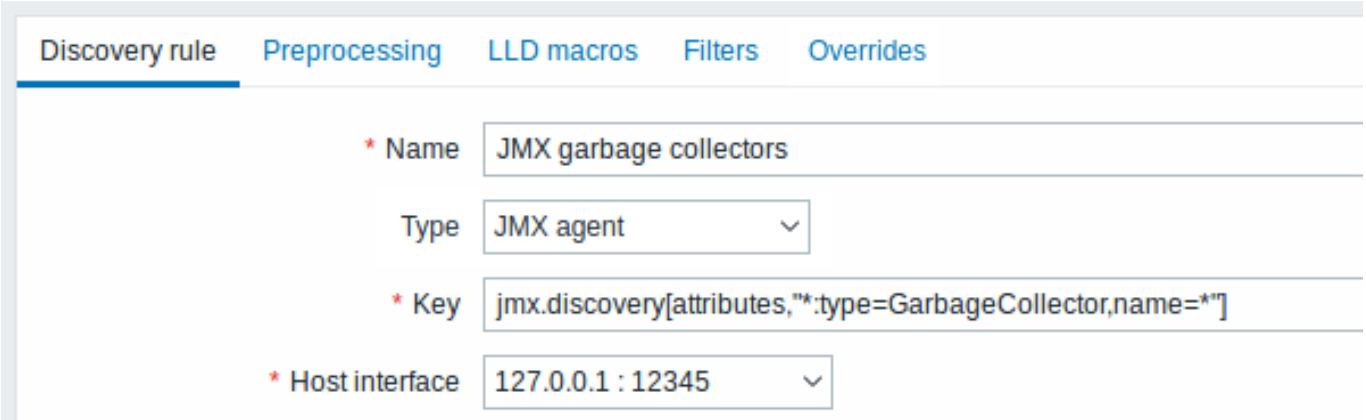


Przykład 2: Odkrywanie atrybutów MBean

Ta reguła zwróci 9 obiektów z następującymi polami: MBean1Attribute1, MBean2Attribute1, MBean3Attribute1,MBean1Attribute2,MBean2Attri MBean3Attribute2, MBean1Attribute3, MBean2Attribute3, MBean3Attribute3.

Aby uzyskać więcej informacji o obiektach, zobacz tabelę supported macros, sekcja *Discovery of MBean attributes*.

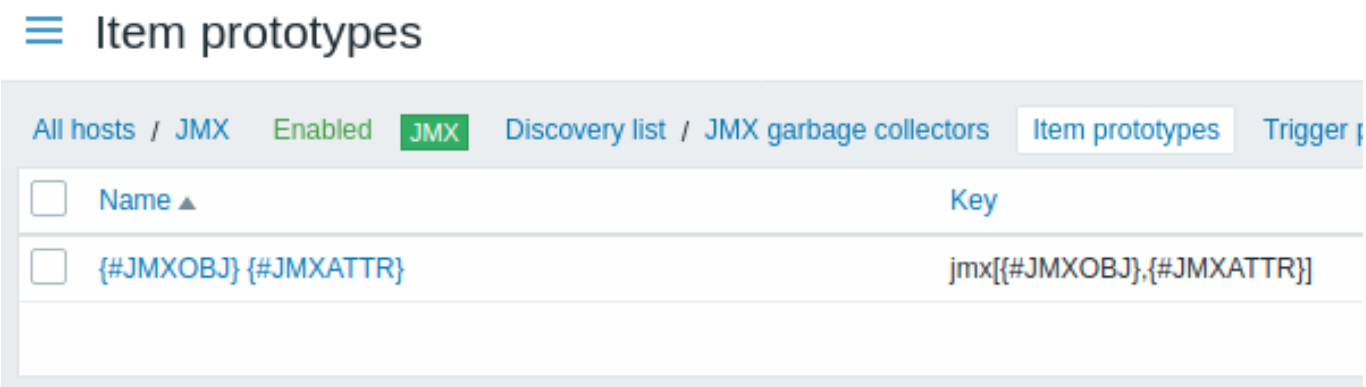
Konfiguracja reguły wykrywania zbierającej atrybuty MBean wygląda następująco:



Użyty tutaj klucz:

```
jmx.discovery[attributes,"*:type=GarbageCollector,name=*"]
```

Wszystkie garbage collectory z atrybutem pojedynczej pozycji zostaną wykryte.



W tym konkretnym przypadku dla każdego atrybutu MBean zostanie utworzona pozycja z prototypu. Główną wadą tej konfiguracji jest to, że utworzenie wyzwalaczy z prototypów wyzwalaczy jest niemożliwe, ponieważ istnieje tylko jeden prototyp pozycji dla wszystkich atrybutów. Dlatego ta konfiguracja może być używana do zbierania danych, ale nie jest zalecana do automatycznego monitorowania.

Używanie jmx.get

`jmx.get[]` jest podobna do pozycji `jmx.discovery[]`, ale nie zamienia właściwości obiektów Java na nazwy makr wykrywania niskiego poziomu i dlatego może zwracać wartości bez ograniczeń, które są związane z generowaniem nazw makr LLD, takich jak myślniki lub znaki spoza ASCII.

Podczas używania `jmx.get[]` do wykrywania makra wykrywania niskiego poziomu mogą być definiowane oddzielnie na niestandardowej karcie makr LLD w konfiguracji reguły wykrywania, z użyciem JSONPath do wskazania wymaganych wartości.

Odkrywanie MBeans

Pozycja wykrywania: `jmx.get[beans,"com.example:type=*,*"]`

Odpowiedź:

```
[
    {
        "object": "com.example:type=Hello,data-src=data-base,  =    ",
        "domain": "com.example",
        "properties": {
            "data-src": "data-base",
            "  ": "    ",
            "type": "Hello"
        }
    },
    {
        "object": "com.example:type=Atomic",
        "domain": "com.example",
        "properties": {
            "type": "Atomic"
        }
    }
]
```

Odkrywanie atrybutów MBean

Pozycja odkrywania: `jmx.get[attributes,"com.example:type=*,*"]`

Odpowiedź:

```
[
    {
        "object": "com.example:type=*",
        "domain": "com.example",
        "properties": {
            "type": "Simple"
        }
    },
    {
        "object": "com.zabbix:type=yes,domain=zabbix.com,data-source=/dev/rand,  =    ,obj=true",
        "domain": "com.zabbix",
        "properties": {
            "type": "Hello",
            "domain": "com.example",
            "data-source": "/dev/rand",
            "  ": "    ",
            "obj": true
        }
    }
]
```

7 Wykrywanie sensorów IPMI

Przegląd

Możliwe jest automatyczne wykrywanie sensorów IPMI.

Aby to zrobić, można użyć kombinacji:

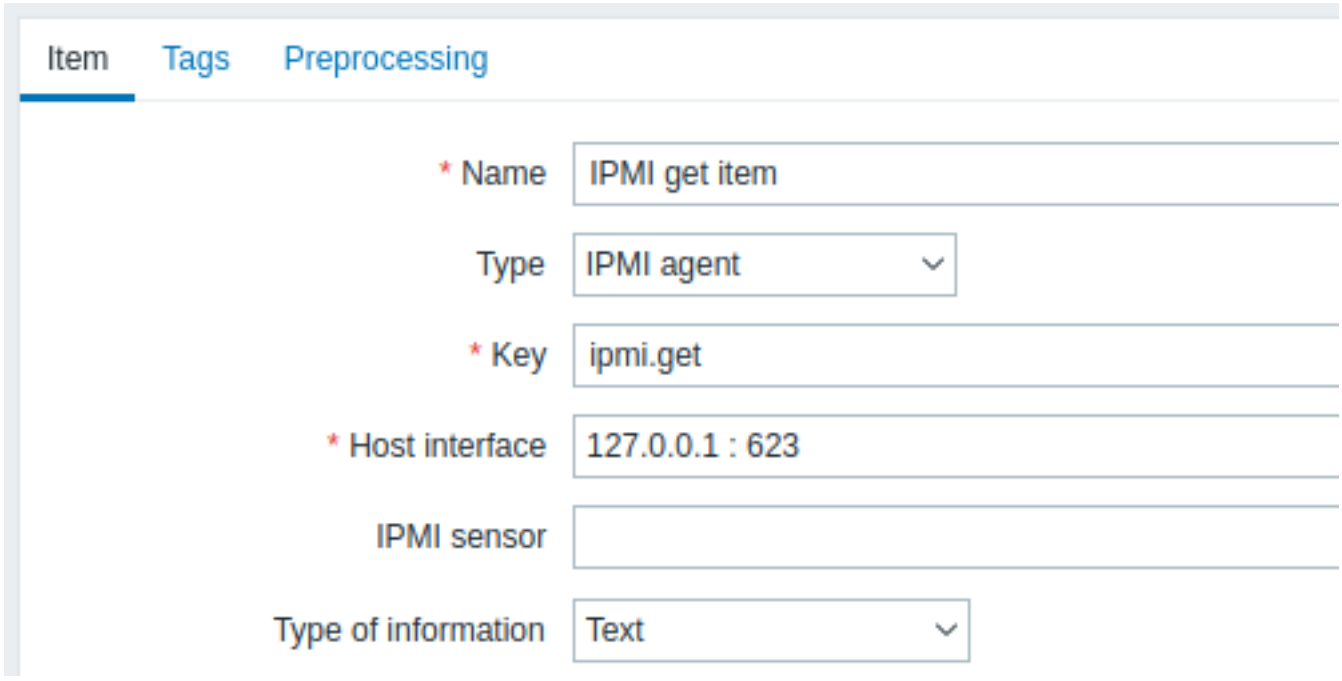
- pozycji IPMI `ipmi.get` jako pozycji nadrzędnej
- zależnej reguły wykrywania niskiego poziomu oraz prototypów pozycji

Konfiguracja

Główna pozycja

Utwórz pozycję IPMI przy użyciu następującego klucza:

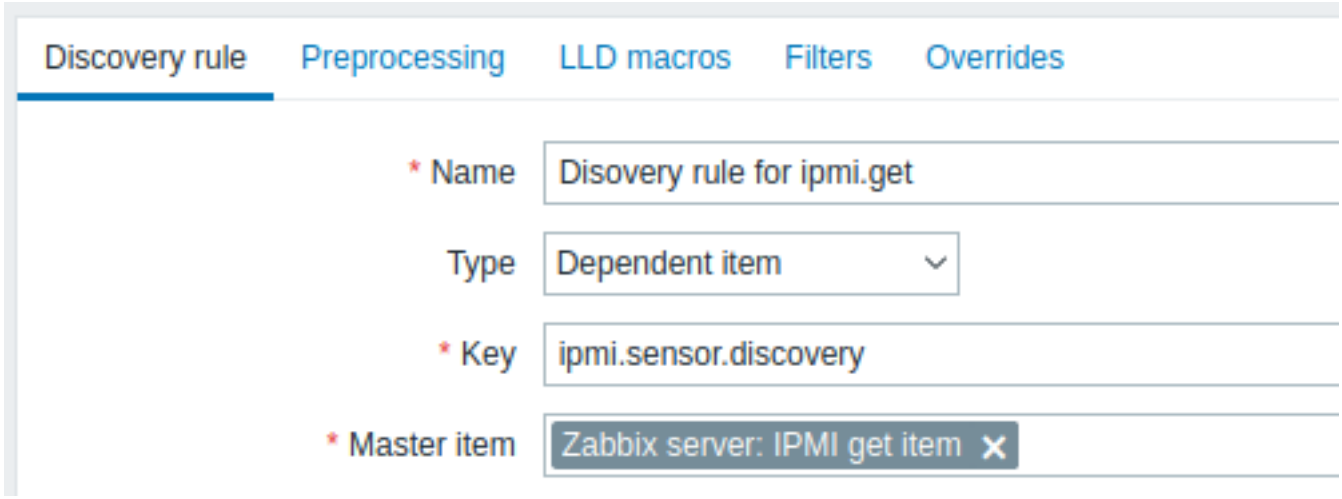
`ipmi.get`



Ustaw typ informacji na „Tekst” dla potencjalnie dużych danych JSON.

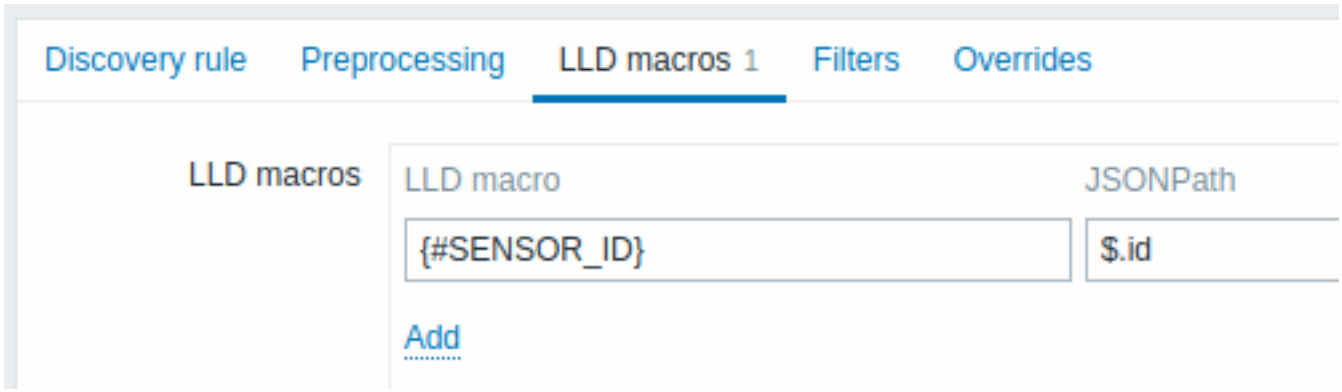
Zależna reguła LLD

Utwórz regułę wykrywania niskiego poziomu typu „Dependent item”:



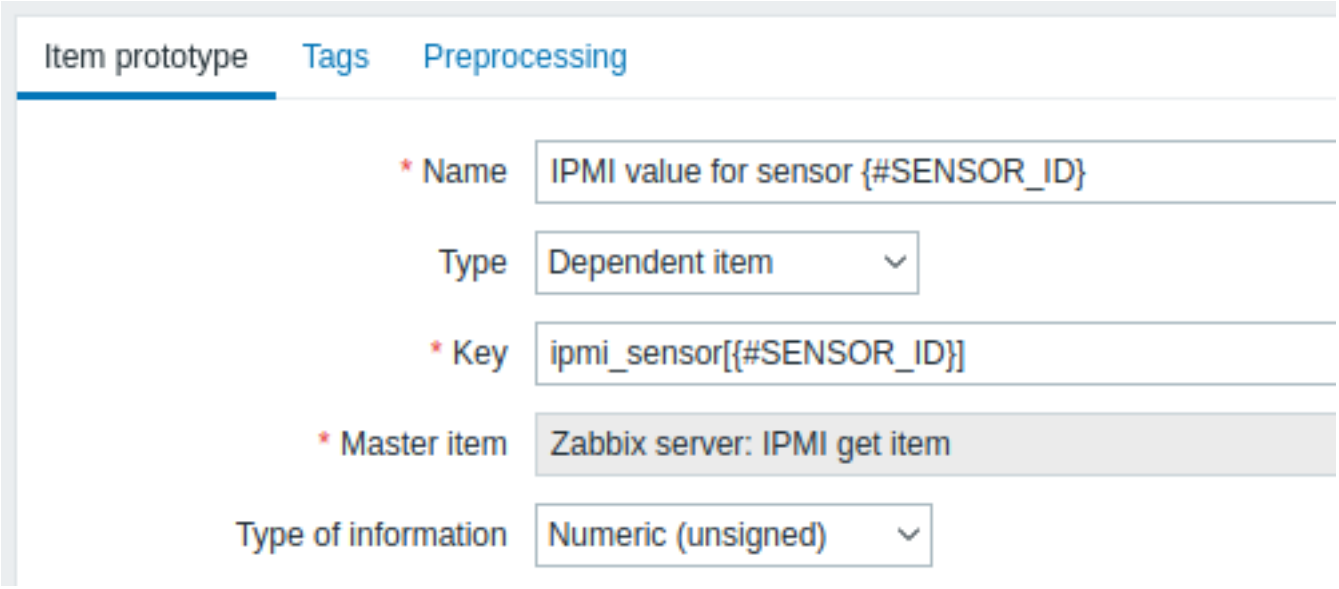
Jako nadrzędną pozycja wybierz utworzoną przez nas pozycja `ipmi.get`.

Na karcie „LLD macros” zdefiniuj niestandardowe makro z odpowiadającym JSONPath:



Prototyp pozycji zależnej

Utwórz prototyp pozycji typu „Dependent item” w tej regule LLD. Jako pozycję nadrzędną dla tego prototypu wybierz pozycję `ipmi.get`, którą utworzyliśmy.



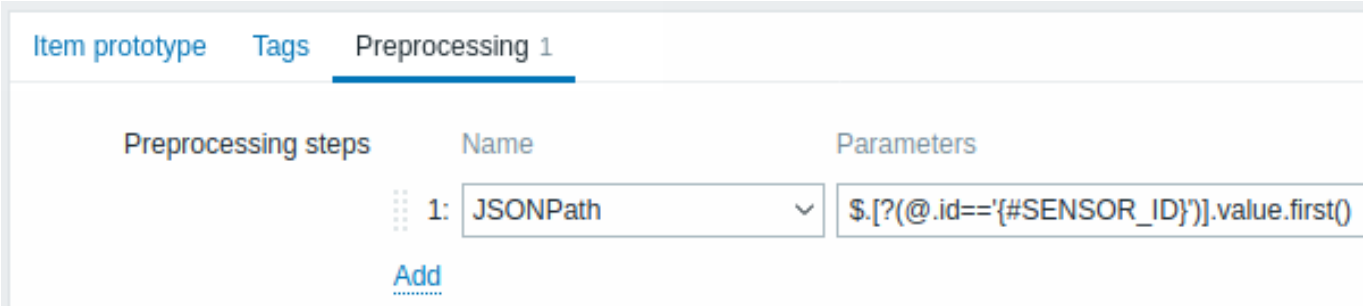
Zwróć uwagę na użycie makra {#SENSOR_ID} w nazwie i kluczu prototypu pozycji:

- *Nazwa*: Wartość IPMI dla sensora {#SENSOR_ID}
- *Klucz*: ipmi_sensor[{#SENSOR_ID}]

Jako typ informacji wybierz *Numeric (unsigned)*.

Na karcie „Preprocessing" prototypu pozycji wybierz JSONPath i użyj następującego wyrażenia JSONPath jako parametru:

```
$.[?(@.id=='{#SENSOR_ID}')].value.first()
```



Po rozpoczęciu wykrywania zostanie utworzona jedna pozycja dla każdego sensora IPMI. Ta pozycja będzie zwracać wartość całkowitą dla danego sensora.

8 Wykrywanie usług systemd

Przegląd

Możliwe jest wykrywanie jednostek systemd (domyślnie usług) za pomocą Zabbix.

Klucz pozycji

Pozycją do użycia w regule odkrywania jest

```
systemd.unit.discovery
```

Attention:
Ten klucz pozycji jest obsługiwany tylko w Zabbix agent 2.

Ta pozycja zwraca JSON z informacjami o jednostkach systemd, na przykład:

```
[{
    "{#UNIT.NAME}": "mysqld.service",
    "{#UNIT.DESCRIPTION}": "MySQL Server",
    "{#UNIT.LOADSTATE}": "loaded",
    "{#UNIT.ACTIVESTATE}": "active",
    "{#UNIT.SUBSTATE}": "running",
    "{#UNIT.FOLLOWED}": "",
```

```
    "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/mysqld_2eservice",
    "{#UNIT.JOBID}": 0,
    "{#UNIT.JOBTYPE}": "",
    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "enabled"
    "{#UNIT.SERVICETYPE}": "simple"
}, {
    "{#UNIT.NAME}": "systemd-journald.socket",
    "{#UNIT.DESCRIPTION}": "Journal Socket",
    "{#UNIT.LOADSTATE}": "loaded",
    "{#UNIT.ACTIVESTATE}": "active",
    "{#UNIT.SUBSTATE}": "running",
    "{#UNIT.FOLLOWED}": "",
    "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/systemd_2djournald_2esocket",
    "{#UNIT.JOBID}": 0,
    "{#UNIT.JOBTYPE}": "",
    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "enabled"
}]
```

Wykrywanie wyłączonych jednostek systemd

Możliwe jest również wykrywanie **wyłączonych** jednostek systemd. W takim przypadku w wynikowym JSON zwracane są trzy makra:

- {#UNIT.PATH}
- {#UNIT.ACTIVESTATE}
- {#UNIT.UNITFILESTATE}.

Attention:
Aby pozycje i wyzwalacze były tworzone z prototypów dla wyłączonych jednostek systemd, upewnij się, że dostosowano (lub usunięto) blokujące filtry LLD dla {#UNIT.ACTIVESTATE} i {#UNIT.UNITFILESTATE}.

Obsługiwane makra

Następujące makra są obsługiwane do użycia w filtrze reguły wykrywania oraz w prototypach pozycji, wyzwalaczy i wykresów:

Makro	Opis
{#UNIT.NAME}	Nazwa podstawowej jednostki.
{#UNIT.DESCRIPTION}	Opis czytelny dla człowieka.
{#UNIT.LOADSTATE}	Stan załadowania (tj. czy plik jednostki został pomyślnie załadowany).
{#UNIT.ACTIVESTATE}	Stan aktywności (tj. czy jednostka jest obecnie uruchomiona, czy nie).
{#UNIT.SUBSTATE}	Stan podrzędny (bardziej szczegółowa wersja stanu aktywności, specyficzna dla typu jednostki, w przeciwieństwie do samego stanu aktywności).
{#UNIT.FOLLOWED}	Jednostka, której stan jest śledzony przez tę jednostkę, jeśli taka istnieje; w przeciwnym razie pusty ciąg znaków.
{#UNIT.PATH}	Ścieżka obiektu jednostki.
{#UNIT.JOBID}	Numeryczny identyfikator zadania, jeśli dla jednostki zadania znajduje się zadanie w kolejce; w przeciwnym razie 0.
{#UNIT.JOBTYPE}	Typ zadania.
{#UNIT.JOBPATH}	Ścieżka obiektu zadania.
{#UNIT.UNITFILESTATE}	Stan instalacji pliku jednostki.
{#UNIT.SERVICETYPE}	Typ jednostki usługi (np. `simple`, `forking`, `oneshot`, `idle` itd.). To makro jest zwracane tylko wtedy, gdy jednostka jest usługą.

Prototypy pozycji

Prototypy pozycji, które można utworzyć na podstawie wykrywania usług systemd, obejmują na przykład:

- Nazwa pozycji: `{#UNIT.DESCRIPTION} active state info`; klucz pozycji: `systemd.unit.info["{#UNIT.NAME}"]`
- Nazwa pozycji: `{#UNIT.DESCRIPTION} load state info`; klucz pozycji: `systemd.unit.info["{#UNIT.NAME}",LoadState]`

9 Wykrywanie usług Windows

Przegląd

W podobny sposób, w jaki wykrywane są systemy plików, możliwe jest również wykrywanie usług Windows.

Klucz pozycji

Pozycja używana w regule odkrywania to

`service.discovery`

Obsługiwane makra

Następujące makra są obsługiwane do użycia w filtrze reguły wykrywania oraz w prototypach pozycji, wyzwalaczy i wykresów:

Makro	Opis
{#SERVICE.NAME}	Nazwa usługi.
{#SERVICE.DISPLAYNAME}	Wyświetlana nazwa usługi.
{#SERVICE.DESCRIPTION}	Opis usługi.
{#SERVICE.STATE}	Wartość liczbowa stanu usługi. Szczegóły znajdują się w pozycji service.info.
{#SERVICE.STATENAME}	Nazwa stanu usługi. Szczegóły znajdują się w pozycji service.info.
{#SERVICE.PATH}	Ścieżka usługi.
{#SERVICE.USER}	Użytkownik usługi.
{#SERVICE.STARTUP}	Wartość liczbowa typu uruchamiania usługi. Szczegóły znajdują się w pozycji service.info.
{#SERVICE.STARTUPNAME}	Nazwa typu uruchamiania usługi. Szczegóły znajdują się w pozycji service.info.
{#SERVICE.STARTUPTRIGGER}	Wartość liczbowa wskazująca, czy typ uruchamiania usługi ma: 0 - brak wyzwalaczy uruchamiania 1 - ma wyzwalacze uruchamiania Jest to przydatne do wykrywania takich typów uruchamiania usług jak *Automatyczny (uruchamianie wyzwalane)*, *Automatyczny opóźniony (uruchamianie wyzwalane)* oraz *Ręczny (uruchamianie wyzwalane).*

Na podstawie wykrywania usług Windows można utworzyć prototyp pozycji taki jak

`service.info[{#SERVICE.NAME},<param>]`

gdzie `param` przyjmuje następujące wartości: *state*, *displayname*, *path*, *user*, *startup* lub *description*.

Na przykład, aby pobrać wyświetlaną nazwę usługi, można użyć pozycji "service.info[{#SERVICE.NAME},displayname]". Jeśli wartość `param` nie zostanie określona ("service.info[{#SERVICE.NAME}]"), używany jest domyślny parametr *state*.

10 Odkrywanie instancji liczników wydajności systemu Windows

Przegląd

Możliwe jest wykrywanie instancji obiektów liczników wydajności systemu Windows. Jest to przydatne w przypadku liczników wydajności z wieloma instancjami.

Klucz pozycji

Aby skonfigurować regułę wykrywania, użyj następującej pozycji:

- `perf_instance.discovery[object]`

Zwróć uwagę, że nazwa obiektu może być zlokalizowana. Na przykład:

```
perf_instance.discovery[Processor] # Nazwa obiektu jest w języku angielskim.
perf_instance.discovery[Processador] # Nazwa obiektu jest w języku portugalskim.
```

Alternatywnie, aby mieć pewność, że nazwa obiektu jest podawana w języku angielskim, niezależnie od lokalizacji systemu operacyjnego, użyj następującej pozycji:

- `perf_instance_en.discovery[object]`

Na przykład:

```
perf_instance_en.discovery[Processor]
perf_instance_en.discovery[Memory]
```

Obsługiwane makra

Proces wykrywania zwróci wszystkie instancje określonego obiektu w makrze {#INSTANCE}:

```
[
    {"{#INSTANCE}":"0"},
    {"{#INSTANCE}":"1"},
    {"{#INSTANCE}":"_Total"}
]
```

To makro może być używane w prototypach pozycji perf_counter[] i perf_counter_en[].

Na przykład, jeśli kluczem pozycji użytym w regule wykrywania jest perf_instance.discovery[Processor], możesz utworzyć następujący prototyp pozycji:

```
perf_counter["\Processor({#INSTANCE})\% Processor Time"]
```

Uwaga:

- Jeśli określony obiekt nie zostanie znaleziony lub nie obsługuje zmiennych instancji, pozycja wykrywania otrzyma stan NOT-SUPPORTED.
- Jeśli określony obiekt obsługuje zmienne instancje, ale obecnie nie ma żadnych instancji, zostanie zwrócona pusta tablica JSON.
- Zduplikowane instancje zostaną pominięte.

11 Wykrywanie przy użyciu zapytań WMI

Przegląd

WMI to potężny interfejs w systemie Windows, który może być używany do pobierania różnych informacji o komponentach Windows, usługach, stanie systemu oraz zainstalowanym oprogramowaniu.

Może być używany do wykrywania dysków fizycznych i zbierania danych o ich wydajności, wykrywania interfejsów sieciowych, wykrywania gości Hyper-V, monitorowania usług Windows oraz wielu innych zadań w systemie operacyjnym Windows.

Ten typ wykrywania niskiego poziomu jest realizowany przy użyciu zapytań WQL, których wyniki są automatycznie przekształcane w obiekt JSON odpowiedni do wykrywania niskiego poziomu.

Klucz pozycji

Pozycja do użycia w regule odkrywania to

```
wmi.getall[<namespace>,<query>]
```

Ta pozycja przekształca wynik zapytania w tablicę JSON. Na przykład:

```
select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'
```

może zwrócić coś takiego:

```
[
    {
        "DeviceID" : "\\.\PHYSICALDRIVE0",
        "BytesPerSector" : 512,
        "Capabilities" : [
            3,
            4
        ],
        "CapabilityDescriptions" : [
            "Random Access",
            "Supports Writing"
        ],
        "Caption" : "VBOX HARDDISK ATA Device",
        "ConfigManagerErrorCode" : 0,
        "ConfigManagerUserConfig" : "False",
        "CreationClassName" : "Win32_DiskDrive",
        "Description" : "Disk drive",
```

```
        "FirmwareRevision" : "1.0",
        "Index" : 0,
        "InterfaceType" : "IDE"
    },
    {
        "DeviceID" : "\\.\PHYSICALDRIVE1",
        "BytesPerSector" : 512,
        "Capabilities" : [
            3,
            4
        ],
        "CapabilityDescriptions" : [
            "Random Access",
            "Supports Writing"
        ],
        "Caption" : "VBOX HARDDISK ATA Device",
        "ConfigManagerErrorCode" : 0,
        "ConfigManagerUserConfig" : "False",
        "CreationClassName" : "Win32_DiskDrive",
        "Description" : "Disk drive",
        "FirmwareRevision" : "1.0",
        "Index" : 1,
        "InterfaceType" : "IDE"
    }
]
```


Makra wykrywania niskiego poziomu

Mimo że w zwróconym JSON nie są tworzone żadne makra wykrywania niskiego poziomu, makra te mogą zostać zdefiniowane przez użytkownika jako dodatkowy krok, z użyciem funkcjonalności niestandardowych makr LLD z JSONPath wskazującym na wykryte wartości w zwróconym JSON.

Makra te mogą następnie zostać użyte do tworzenia prototypów pozycji, wyzwalaczy itp.

12 Wykrywanie za pomocą zapytań SQL poprzez ODBC

Przegląd

Ten typ wykrywania niskiego poziomu jest realizowany przy użyciu zapytań SQL, których wyniki są automatycznie przekształcane w obiekt JSON odpowiedni do wykrywania niskiego poziomu.

Klucz pozycja

Zapytania SQL są wykonywane przy użyciu typu pozycja ”Database monitor”. Dlatego większość instrukcji na stronie monitorowania ODBC ma zastosowanie, aby uzyskać działającą regułę wykrywania ”Database monitor”.

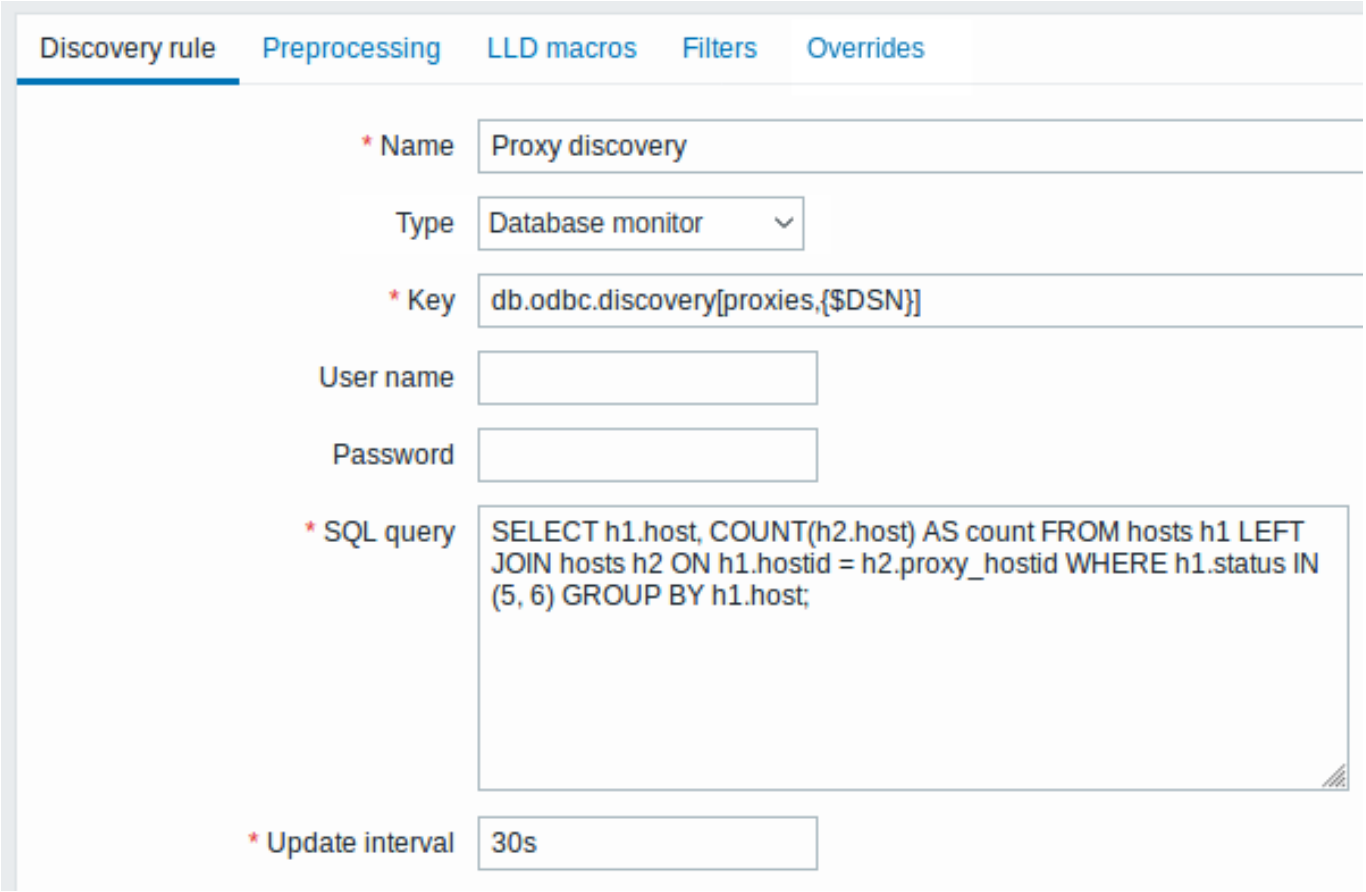
W regułach wykrywania ”Database monitor” można użyć dwóch kluczy pozycji:

- **db.odbc.discovery**[<unique short description>,<dsn>,<connection string>] - ta pozycja przekształca wynik zapytania SQL w tablicę JSON, zamieniając nazwy kolumn z wyniku zapytania na nazwy makr niskopoziomowego wykrywania sparowane z wykrytymi wartościami pól. Te makra można wykorzystać podczas tworzenia prototypów pozycji, wyzwalaczy itp. Zobacz także: Using db.odbc.discovery.
- **db.odbc.get**[<unique short description>,<dsn>,<connection string>] - ta pozycja przekształca wynik zapytania SQL w tablicę JSON, zachowując oryginalne nazwy kolumn z wyniku zapytania jako nazwę pola w JSON sparowaną z wykrytymi wartościami. W porównaniu z `db.odbc.discovery[]`, ta pozycja nie tworzy makr niskopoziomowego wykrywania w zwracanym JSON, dlatego nie ma potrzeby sprawdzania, czy nazwy kolumn mogą być prawidłowymi nazwami makr. Makra niskopoziomowego wykrywania można zdefiniować jako dodatkowy krok w razie potrzeby, korzystając z funkcji niestandardowego makra LLD z JSONPath wskazującym wykryte wartości w zwracanym JSON. Zobacz także: Using db.odbc.get.

Użycie db.odbc.discovery

Poniższy przykład pokazuje, jak zapytanie SQL jest przekształcane do formatu JSON przy użyciu wykrywania niskiego poziomu proxy Zabbix, na podstawie zapytania ODBC do bazy danych Zabbix. Jest to przydatne do automatycznego tworzenia ”zabbix[proxy,<name>,lastaccess]” pozycji wewnętrznych w celu monitorowania, które proxy są aktywne.

Zacznij od konfiguracji reguły wykrywania:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Tutaj używane jest następujące bezpośrednie zapytanie do bazy danych Zabbix w celu wybrania wszystkich proxy Zabbix wraz z liczbą hostów, które monitorują. Liczba hostów może zostać użyta na przykład do odfiltrowania pustych proxy:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxyid
+---------+-------+
| host    | count |
+---------+-------+
| Japan 1 |     5 |
| Japan 2 |    12 |
| Latvia  |     3 |
+---------+-------+
3 rows in set (0.01 sec)
```

Dzięki wewnętrznemu działaniu pozycji "db.odbc.discovery[,{$DSN}]" wynik tego zapytania jest automatycznie przekształcany do następującego formatu JSON:

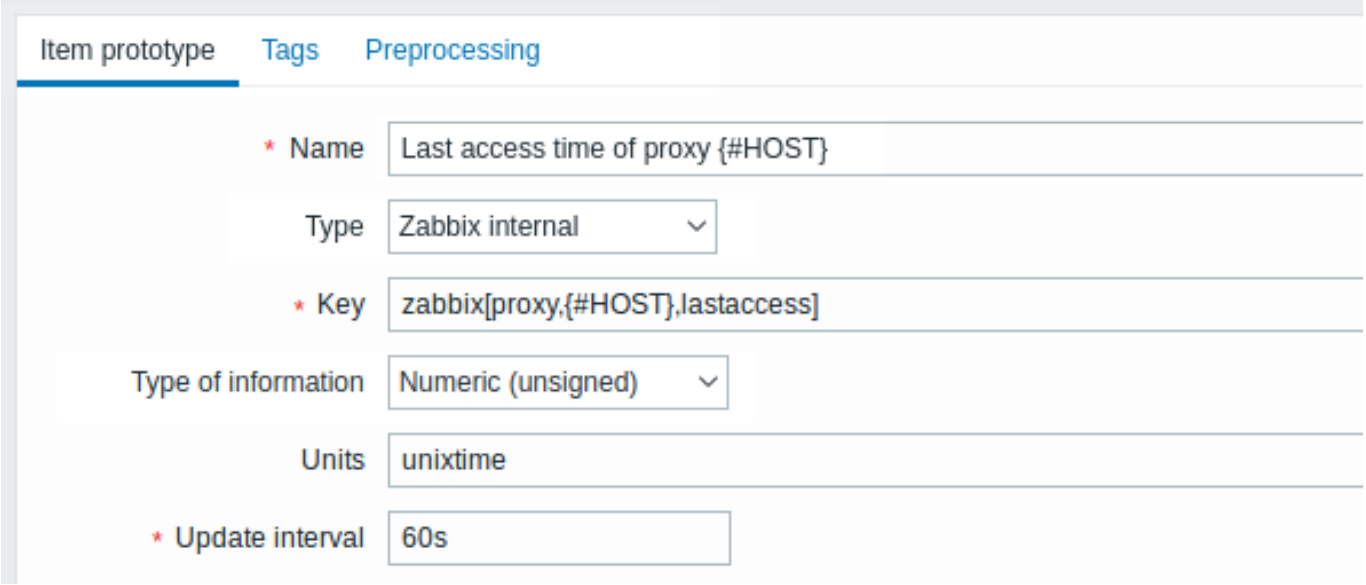
```
[
    {
        "{#HOST}": "Japan 1",
        "{#COUNT}": "5"
    },
    {
        "{#HOST}": "Japan 2",
        "{#COUNT}": "12"
    },
    {
        "{#HOST}": "Latvia",
        "{#COUNT}": "3"
    }
]
```

Jak widać, nazwy kolumn stają się nazwami makr, a wybrane wiersze stają się wartościami tych makr.

Note:
Jeśli nie jest oczywiste, w jaki sposób nazwa kolumny zostanie przekształcona w nazwę makra, zaleca się użycie aliasów kolumn, takich jak "COUNT(h2.host) AS count" w powyższym przykładzie.
Jeśli nazwa kolumny nie może zostać przekonwertowana na prawidłową nazwę makra, reguła wykrywania stanie się nieobsługiwana, a komunikat błędu będzie zawierał numer problematycznej kolumny. Jeśli potrzebna jest dodatkowa pomoc, uzyskane nazwy kolumn są podawane przy DebugLevel=4 w pliku dziennika serwera Zabbix:

```
$ grep db.odbc.discovery /tmp/zabbix_server.log
 ...
 23876:20150114:153410.856 In db_odbc_discovery() query:'SELECT h1.host, COUNT(h2.host) FROM hosts h1 L
 23876:20150114:153410.860 db_odbc_discovery() column[1]:'host'
 23876:20150114:153410.860 db_odbc_discovery() column[2]:'COUNT(h2.host)'
 23876:20150114:153410.860 End of db_odbc_discovery():NOTSUPPORTED
 23876:20150114:153410.860 Item [Zabbix server:db.odbc.discovery[proxies,{$DSN}]] error: Cannot convert
```

Teraz, gdy rozumiemy, jak zapytanie SQL jest przekształcane w obiekt JSON, możemy użyć makra {#HOST} w prototypach pozycji:



Po wykonaniu wykrywania zostanie utworzona pozycja dla każdego proxy:



Używanie db.odbc.get

Używając db.odbc.get[,{$DSN}] oraz następującego przykładu SQL:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxyid
+---------+-------+
| host    | count |
+---------+-------+
| Japan 1 |     5 |
| Japan 2 |    12 |
| Latvia  |     3 |
+---------+-------+
3 rows in set (0.01 sec)
```

zostanie zwrócony następujący JSON:

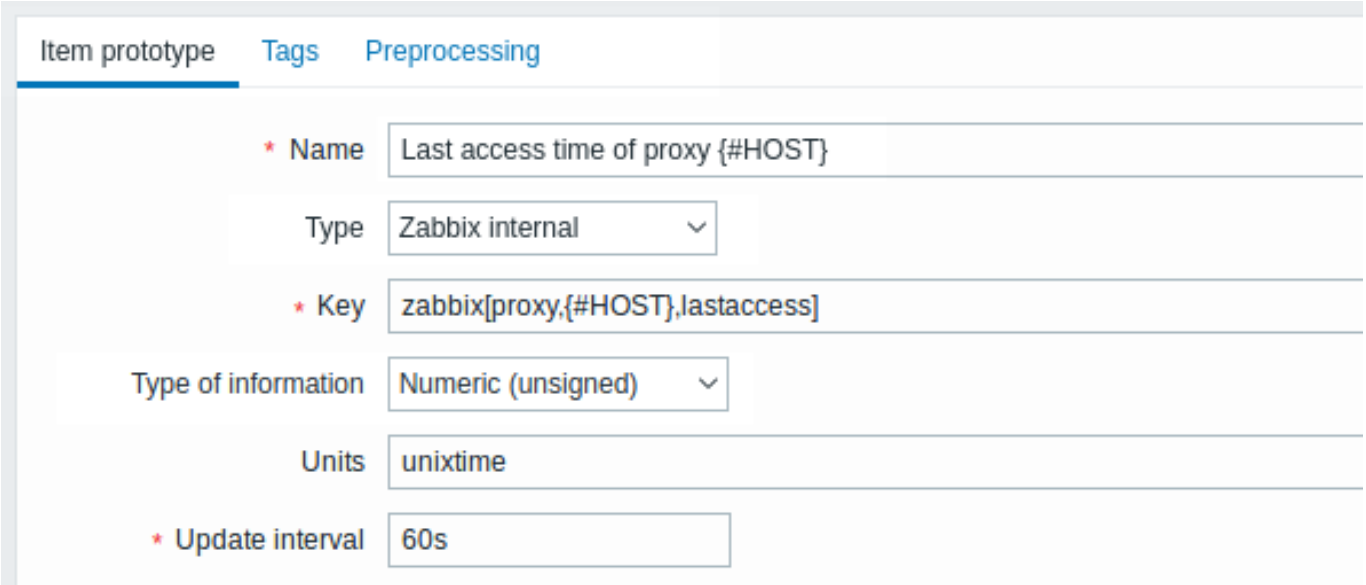
```
[
    {
```

```
        "host": "Japan 1",
        "count": "5"
    },
    {
        "host": "Japan 2",
        "count": "12"
    },
    {
        "host": "Latvia",
        "count": "3"
    }
]
```

Jak widać, nie ma tam makr wykrywania niskopoziomowego. Jednak niestandardowe makra wykrywania niskopoziomowego można utworzyć na karcie Makra LLD reguły wykrywania przy użyciu JSONPath, na przykład:

```
{#HOST} → $.host
```

Teraz to makro {#HOST} może być używane w prototypach pozycji:



13 Wykrywanie przy użyciu danych Prometheus

Przegląd

Dane dostarczane w formacie linii Prometheus mogą być używane do wykrywania niskiego poziomu.

Zobacz Prometheus checks, aby uzyskać szczegółowe informacje o tym, jak zapytania o dane Prometheus są implementowane w Zabbix.

Konfiguracja

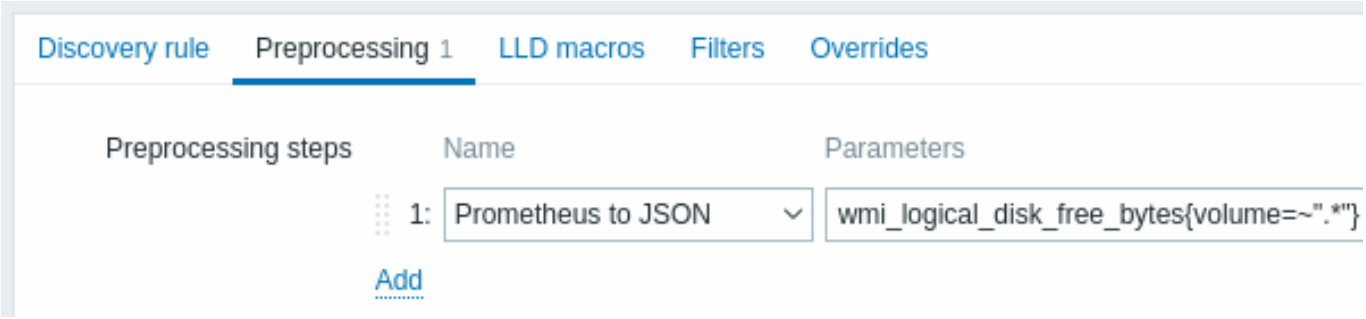
Reguła wykrywania niskiego poziomu powinna zostać utworzona jako pozycja zależna względem nadrzędnej pozycji HTTP, która zbiera dane Prometheus.

Prometheus do JSON

W regule wykrywania przejdź do zakładki Preprocessing i wybierz opcję przetwarzania wstępnego *Prometheus do JSON*. Dane w formacie JSON są potrzebne do wykrywania, a opcja przetwarzania wstępnego *Prometheus do JSON* zwróci dokładnie takie dane z następującymi atrybutami:

- nazwa metryki
- wartość metryki
- help (jeśli występuje)
- type (jeśli występuje)
- labels (jeśli występują)
- surowy wiersz

Na przykład zapytanie `wmi_logical_disk_free_bytes`:



z tych wierszy Prometheus:

```
# HELP wmi_logical_disk_free_bytes Free space in bytes (LogicalDisk.PercentFreeSpace)
# TYPE wmi_logical_disk_free_bytes gauge
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
```

zwróci:

```
[
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "C:"
         },
        "value": "3.5180249088e+11",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"C:\"} 3.5180249088e+11"
    },
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "D:"
         },
        "value": "2.627731456e+09",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"D:\"} 2.627731456e+09"
    },
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "HarddiskVolume4"
         },
        "value": "4.59276288e+08",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"HarddiskVolume4\"} 4.59276288e+08"
    }
]
```

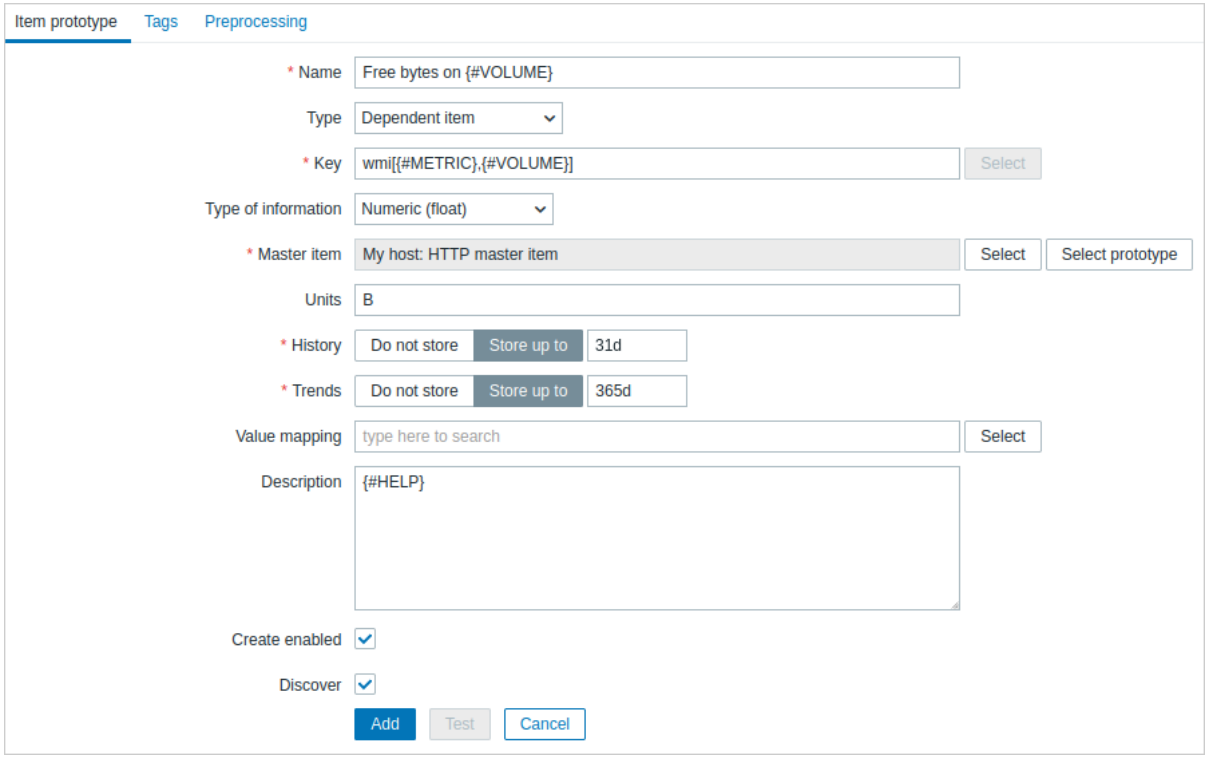
Mapowanie makr LLD

Następnie należy przejść do zakładki makr LLD i utworzyć następujące mapowania:

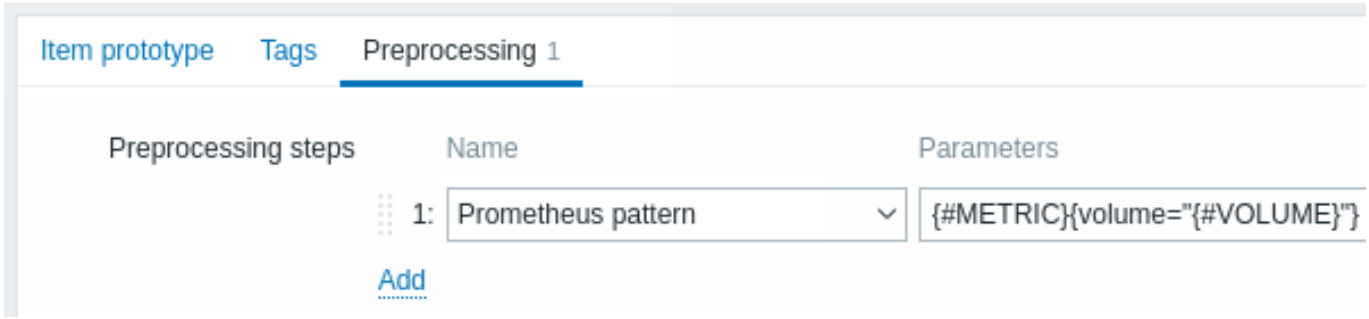
```
{#VOLUME}=$.labels['volume']
{#METRIC}=$['name']
{#HELP}=$['help']
```

Prototyp pozycji

Możesz chcieć utworzyć prototyp pozycji w taki sposób:



z opcjami przetwarzania wstępnego:



14 Wykrywanie urządzeń blokowych

Podobnie jak wykrywane są systemy plików, możliwe jest również wykrywanie urządzeń blokowych oraz ich typu.

Klucz pozycji

Klucz pozycji używany w regule wykrywania to

`vfs.dev.discovery`

Ta pozycja jest obsługiwana tylko na platformach Linux.

Możesz tworzyć reguły wykrywania przy użyciu tej pozycji wykrywania oraz:

- filtra: **{#DEVNAME} pasuje do** `sd[\D]$` - aby wykrywać urządzenia o nazwach „sd0”, „sd1”, „sd2”, ...
- filtra: **{#DEVTYPE} pasuje do** `disk` **ORAZ {#DEVNAME} nie pasuje do** `^loop.*` - aby wykrywać urządzenia typu dysk, których nazwa nie zaczyna się od „loop”

Obsługiwane makra

Ten klucz wykrywania zwraca dwa makra - {#DEVNAME} i {#DEVTYPE}, identyfikujące odpowiednio nazwę i typ urządzenia blokowego, na przykład:

```
[
   {
      "{#DEVNAME}":"loop1",
      "{#DEVTYPE}":"disk"
   },
   {
      "{#DEVNAME}":"dm-0",
```

```
        "{#DEVTYPE}":"disk"
    },
    {
        "{#DEVNAME}":"sda",
        "{#DEVTYPE}":"disk"
    },
    {
        "{#DEVNAME}":"sda1",
        "{#DEVTYPE}":"partition"
    }
]
```

Wykrywanie urządzeń blokowych pozwala używać pozycji `vfs.dev.read[]` i `vfs.dev.write[]` do tworzenia prototypów pozycji z użyciem makra {#DEVNAME}, na przykład:

- ”vfs.dev.read[{#DEVNAME},sps]”
- ”vfs.dev.write[{#DEVNAME},sps]”

Makro {#DEVTYPE} jest przeznaczone do filtrowania urządzeń.

15 Odkrywanie interfejsów hosta w Zabbix

Przegląd

Możliwe jest wykrywanie wszystkich interfejsów skonfigurowanych w frontend Zabbix dla hosta.

Klucz pozycji

Pozycją używaną w regule odkrywania jest wewnętrzna pozycja Zabbix:

`zabbix[host,discovery,interfaces]`

Ta pozycja zwraca JSON z opisem interfejsów, w tym:

- adres IP/nazwę hosta DNS (w zależności od ustawienia hosta „Połącz z”)
- numer portu
- typ interfejsu (agent Zabbix, SNMP, JMX, IPMI)
- informację, czy jest to interfejs domyślny, czy nie
- informację, czy funkcja żądań zbiorczych jest włączona — tylko dla interfejsów SNMP.

Na przykład:

```
[{"{#IF.CONN}":"192.168.3.1","{#IF.IP}":"192.168.3.1","{#IF.DNS}":"","{#IF.PORT}":"10050","{#IF.TYPE}":"AG
```

W przypadku wielu interfejsów ich rekordy w JSON są uporządkowane według:

- typu interfejsu,
- domyślności — interfejs domyślny jest umieszczany przed interfejsami niedomyślnymi,
- ID interfejsu (w porządku rosnącym).

Obsługiwane makra

Następujące makra są obsługiwane do użycia w filtrze reguły wykrywania oraz w prototypach pozycji, wyzwalaczy i wykresów:

Macro	Description
{#IF.CONN}	Adres IP interfejsu lub nazwa hosta DNS.
{#IF.IP}	Adres IP interfejsu.
{#IF.DNS}	Nazwa hosta DNS interfejsu.
{#IF.PORT}	Numer portu interfejsu.
{#IF.TYPE}	Typ interfejsu (”AGENT”, ”SNMP”, ”JMX” lub ”IPMI”).
{#IF.DEFAULT}	Status domyślny interfejsu: 0 - interfejs niedomyślny 1 - interfejs domyślny
{#IF.SNMP.BULK}	Status przetwarzania zbiorczego SNMP dla interfejsu: 0 - wyłączone 1 - włączone To makro jest zwracane tylko wtedy, gdy typ interfejsu to ”SNMP”.

8 Własne reguły LLD

Przegląd

Możliwe jest również utworzenie całkowicie niestandardowej reguły LLD, wykrywającej dowolny typ encji — na przykład bazy danych na serwerze bazodanowym.

Aby to zrobić, należy utworzyć niestandardową pozycja, która zwraca ciąg JSON, określający znalezione obiekty oraz opcjonalnie — niektóre ich właściwości. Liczba makr na encję nie jest ograniczona — podczas gdy wbudowane reguły wykrywania zwracają jedno lub dwa makra (na przykład dwa w przypadku wykrywania systemów plików), możliwe jest zwrócenie większej liczby.

Przykład

Wymagany format ciągu JSON najlepiej zilustrować na przykładzie. Załóżmy, że używasz starego agenta Zabbix 1.8 (takiego, który nie obsługuje klucza `vfs.fs.discovery`), ale nadal musisz wykrywać systemy plików. Poniżej znajduje się prosty skrypt Perl dla systemu Linux, który wykrywa zamontowane systemy plików i zwraca dane w formacie JSON, zawierające zarówno nazwę systemu plików, jak i jego typ. Jednym ze sposobów użycia go byłoby zastosowanie go jako UserParameter z kluczem `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\"{#FSNAME}\":\"$fsname\",\n";
    print "\t\t\"{#FSTYPE}\":\"$fstype\"\n";
    print "\t}\n";
}

print "]\n";
```

Attention:
Dozwolone symbole w nazwach makr LLD to **0-9** , **A-Z** , **_** , **.**. Małe litery nie są obsługiwane w nazwach.

Przykład danych wyjściowych (przeformatowany dla większej czytelności) pokazano poniżej. JSON dla niestandardowych kontroli wykrywania musi mieć ten sam format.

```
[
    { "{#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"  }
]
```

W poprzednim przykładzie wymagane jest, aby klucze odpowiadały nazwom makr LLD używanym w prototypach; alternatywnie można wyodrębnić wartości makr LLD za pomocą JSONPath {#FSNAME} → `$.fsname` oraz {#FSTYPE} → `$.fstype`, co umożliwia użycie takiego skryptu:

```
####!/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";
```

Przykład danych wyjściowych (przeformatowany dla większej czytelności) pokazano poniżej.

```
[
    { "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"  }
]
```

Następnie w polu *Filter* reguły wykrywania możesz określić `{#FSTYPE}` jako makro, a `rootfs|ext3` jako wyrażenie regularne.

Note:
Nie musisz używać nazw makr takich jak `FSNAME`/`FSTYPE` w niestandardowych regułach LLD; możesz używać dowolnych nazw. Jeśli używany jest JSONPath, wiersz LLD będzie elementem tablicy, który może być obiektem, ale może też być inną tablicą lub wartością.

Pamiętaj, że w przypadku użycia parametru użytkownika wartość zwracana jest ograniczona do 16 MB. Więcej informacji znajdziesz w sekcji limity danych dla wartości zwracanych LLD.

16 Monitorowanie rozproszone

Przegląd Zabbix zapewnia skuteczny i niezawodny sposób monitorowania rozproszonej infrastruktury IT przy użyciu proxy.

Proxy mogą być używane do lokalnego zbierania danych w imieniu scentralizowanego serwera Zabbix, a następnie przekazywania tych danych do serwera.

Funkcje proxy

Przy podejmowaniu decyzji o użyciu lub nieużyciu proxy należy wziąć pod uwagę kilka kwestii.

	Proxy
Lekki	**Tak**

	Proxy
GUI	Nie
Działa niezależnie	**Tak**
Łatwe utrzymanie	**Tak**
Automatyczne tworzenie bazy danych	**Tak**[1]
Lokalna administracja	Nie
Gotowy do użycia na sprzęcie wbudowanym	**Tak**
Jednokierunkowe połączenia TCP	**Tak**
Scentralizowana konfiguracja	**Tak**
Generuje powiadomienia	Nie

[1] Funkcja automatycznego tworzenia bazy danych działa tylko z SQLite. Inne obsługiwane bazy danych wymagają ręcznej konfiguracji.

Attention:
Proxy Zabbix nie uwzględnia okresów konserwacji; szczegóły znajdują się w sekcji Obliczanie kolejek podczas konserwacji.

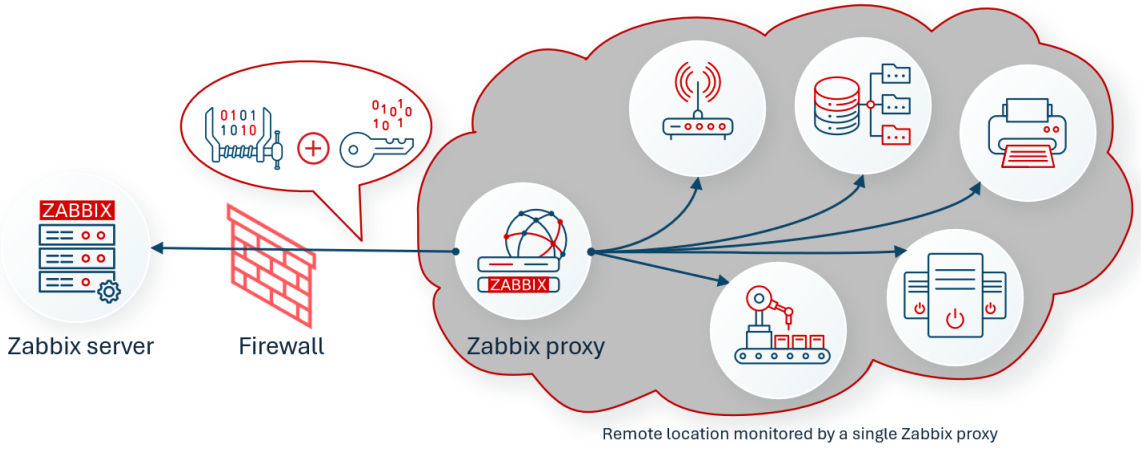
1 Proxy

Przegląd Proxy Zabbixa może zbierać dane o wydajności i dostępności w imieniu serwera Zabbixa. W ten sposób proxy może przejąć część obciążenia związanego ze zbieraniem danych i odciążyć serwer Zabbixa.

Ponadto użycie proxy jest najprostszym sposobem wdrożenia scentralizowanego i rozproszonego monitorowania, gdy wszystkie agent i proxy raportują do jednego serwera Zabbixa, a wszystkie dane są zbierane centralnie.

Proxy Zabbixa można użyć do:

- Monitorowania zdalnych lokalizacji
- Monitorowania lokalizacji o zawodnej łączności
- Odciążenia serwera Zabbixa podczas monitorowania tysięcy urządzeń
- Uproszczenia utrzymania monitorowania rozproszonego



Proxy wymaga tylko jednego połączenia TCP z serwerem Zabbixa. W ten sposób łatwiej obejść zaporę sieciową, ponieważ trzeba skonfigurować tylko jedną regułę zapory.

Attention:
Proxy Zabbixa musi używać oddzielnej bazy danych. Wskazanie go na bazę danych serwera Zabbixa spowoduje uszkodzenie konfiguracji.

Wszystkie dane zbierane przez proxy są przechowywane lokalnie przed ich przesłaniem do serwera. Dzięki temu żadne dane nie zostaną utracone z powodu tymczasowych problemów komunikacyjnych z serwerem. Parametry *ProxyLocalBuffer* i *ProxyOfflineBuffer* w pliku konfiguracji proxy określają, jak długo dane są przechowywane lokalnie.

Attention:
Może się zdarzyć, że proxy, które otrzymuje najnowsze zmiany konfiguracji bezpośrednio z bazy danych serwera Zabbixa, ma bardziej aktualną konfigurację niż serwer Zabbixa, którego konfiguracja może nie być aktualizowana tak szybko ze względu na wartość CacheUpdateFrequency. W rezultacie proxy może rozpocząć zbieranie danych i wysyłać je do serwera Zabbixa, który ignoruje te dane.

Proxy Zabbixa jest kolektorem danych. Nie oblicza wyzwalaczy, nie przetwarza zdarzeń ani nie wysyła alertów. Aby zapoznać się z przeglądem funkcjonalności proxy, przejrzyj poniższą tabelę:

Function		Supported by proxy
Items		
	Zabbix agent checks	**Yes**
	Zabbix agent checks (active)	**Yes** [1]
	Simple checks	**Yes**
	Trapper items	**Yes**
	SNMP checks	**Yes**
	SNMP traps	**Yes**
	IPMI checks	**Yes**
	JMX checks	**Yes**
	Log file monitoring	**Yes**
	Internal checks	**Yes**
	SSH checks	**Yes**
	Telnet checks	**Yes**
	External checks	**Yes**
	Dependent items	**Yes**
	Script items	**Yes**
	Browser items	**Yes**
Built-in web monitoring		**Yes**
Item value preprocessing		**Yes**
Network discovery		**Yes**
Active agent autoregistration		**Yes**
Low-level discovery		**Yes** [2]
Remote commands		**Yes**
Calculating triggers		*No*
Processing events		*No*
Event correlation		*No*
Sending alerts		*No*

Note:
[1] Aby upewnić się, że agent pyta proxy (a nie serwer) o aktywne sprawdzenia, proxy musi być wymieniony w parametrze **ServerActive** w pliku konfiguracji agenta.
 [2] W przypadku LLD proxy Zabbixa tylko zbiera i wstępnie przetwarza dane, a następnie wysyła je do serwera Zabbixa do dalszego przetwarzania.

Ochrona przed przeciążeniem

Jeśli serwer Zabbix był przez jakiś czas niedostępny, a proxy zebrały dużo danych, a następnie serwer zostanie uruchomiony, może dojść do jego przeciążenia (użycie pamięci podręcznej historii utrzymuje się na poziomie 95-100% przez pewien czas). Takie przeciążenie może spowodować spadek wydajności, w wyniku czego kontrole są przetwarzane wolniej, niż powinny. Ochronę przed takim scenariuszem zaimplementowano, aby uniknąć problemów wynikających z przeciążenia pamięci podręcznej historii.

Gdy pamięć podręczna historii serwera Zabbix jest pełna, dostęp do zapisu w pamięci podręcznej historii jest ograniczany, co wstrzymuje procesy zbierania danych przez serwer. Najczęstszym przypadkiem przeciążenia pamięci podręcznej historii jest sytuacja po przestoju serwera, gdy proxy przesyłają zebrane dane. Aby temu zapobiec, dodano ograniczanie przepustowości proxy (obecnie nie można go wyłączyć).

Serwer Zabbix przestanie przyjmować dane od proxy, gdy użycie pamięci podręcznej historii osiągnie 80%. Zamiast tego proxy zostaną umieszczone na liście ograniczania przepustowości. Będzie to trwało do momentu, gdy użycie pamięci podręcznej spadnie do 60%. Wtedy serwer zacznie ponownie przyjmować dane od proxy, pojedynczo, zgodnie z listą ograniczania przepustowości. Oznacza to, że pierwsze proxy, które próbowało przesłać dane w okresie ograniczania przepustowości, zostanie obsłużone jako pierwsze i dopóki nie zakończy przesyłania, serwer nie będzie przyjmował danych od innych proxy.

Ten tryb ograniczania przepustowości będzie trwał do momentu, gdy użycie pamięci podręcznej ponownie osiągnie 80%, spadnie

do 20% albo lista ograniczania przepustowości będzie pusta. W pierwszym przypadku serwer ponownie przestanie przyjmować dane od proxy. W dwóch pozostałych przypadkach serwer zacznie działać normalnie, przyjmując dane od wszystkich proxy.

Powyższe informacje można przedstawić w następującej tabeli:

Użycie pamięci podręcznej zapisu historii	Tryb serwera Zabbix	Działanie serwera Zabbix
Osiąga 80%	Oczekiwanie	Przestaje przyjmować dane od proxy, ale utrzymuje *listę ograniczania przepustowości* (priorytetową listę proxy, z którymi należy skontaktować się później).
Spada do 60%	Ograniczanie przepustowości	Rozpoczyna przetwarzanie listy ograniczania przepustowości, ale nadal nie przyjmuje danych od proxy.
Spada do 20%	Normalny	Usuwa listę ograniczania przepustowości i zaczyna normalnie przyjmować dane od proxy.

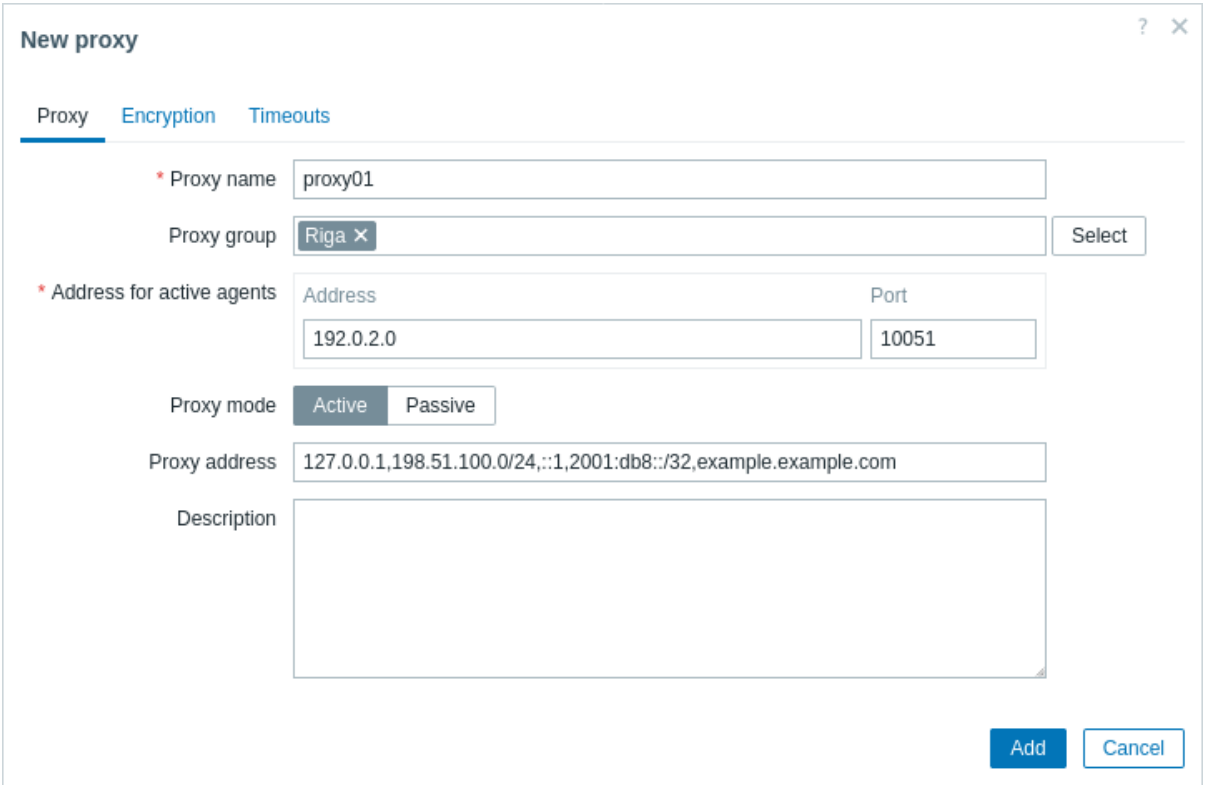
Możesz użyć wewnętrznej pozycji zabbix[wcache,history,pused], aby powiązać to zachowanie serwera Zabbix z metryką.

Konfiguracja Po zainstalowaniu i skonfigurowaniu proxy, czas skonfigurować je w frontend Zabbix.

Dodawanie proxy

Aby skonfigurować proxy w frontend Zabbix:

- Przejdź do: *Administration* → *Proxies*
- Kliknij *Create proxy*



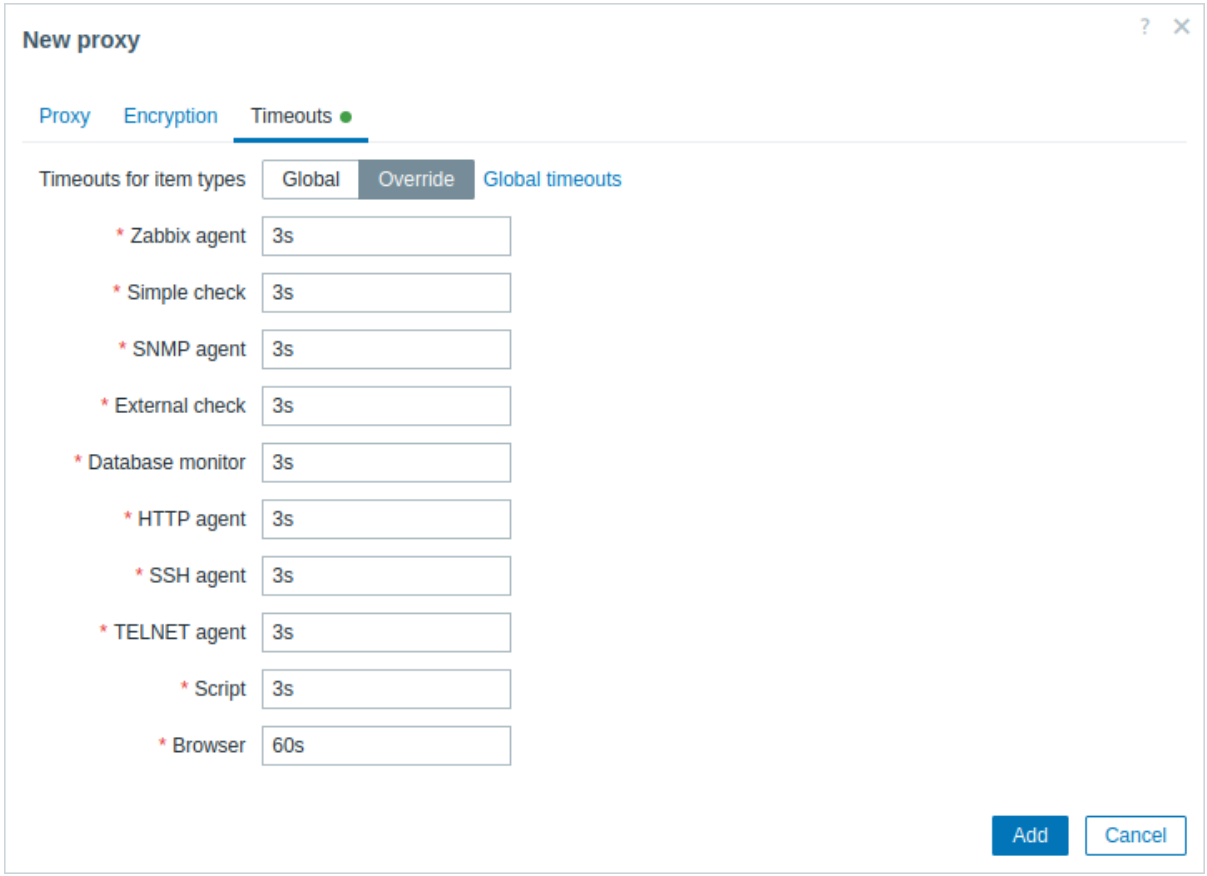
Parameter	Description
Proxy name	Wprowadź nazwę proxy. Musi być taka sama jak nazwa w parametrze *Hostname* w pliku konfiguracyjnym proxy.
Proxy group	Wybierz jedną grupę proxy do równoważenia obciążenia/wysokiej dostępności.
Address for active agents	Wprowadź adres, z którym muszą łączyć się monitorowane aktywne agent lub nadawcy. Obsługiwane **tylko** dla agentów Zabbix 7.0 lub nowszych. Ten adres jest używany do łączenia się zarówno z aktywnymi, jak i pasywnymi proxy. To pole jest dostępne tylko wtedy, gdy w polu *Proxy group* wybrano grupę proxy.
Address	Adres IP/nazwa DNS, z którym należy się połączyć.

Parameter	Description
Port	Numer portu TCP (domyślnie 10051), z którym należy się połączyć. Obsługiwane są makra użytkownika.
Proxy mode	Wybierz tryb proxy. **Active** - proxy połączy się z serwerem Zabbix i pobierze dane konfiguracyjne **Passive** - serwer Zabbix łączy się z proxy *Uwaga*: bez szyfrowanej komunikacji (wrażliwe) dane konfiguracyjne proxy mogą stać się dostępne dla osób mających dostęp do portu trapper serwera Zabbix podczas korzystania z aktywnego proxy. Jest to możliwe, ponieważ każdy może podszyć się pod aktywne proxy i zażądać danych konfiguracyjnych, jeśli nie odbywa się uwierzytelnianie lub adresy proxy nie są ograniczone w polu *Proxy address*.
Proxy address	Jeśli określono, żądania aktywnego proxy są akceptowane tylko z tej listy adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS aktywnego proxy Zabbix. To pole jest dostępne tylko wtedy, gdy w polu *Proxy mode* wybrano aktywne proxy. Makra nie są obsługiwane.
Interface	Wprowadź szczegóły interfejsu dla pasywnego proxy. To pole jest dostępne tylko wtedy, gdy w polu Proxy mode wybrano pasywne proxy.
Address	Adres IP/nazwa DNS pasywnego proxy.
Port	Numer portu TCP pasywnego proxy (domyślnie 10051). Obsługiwane są makra użytkownika.
Description	Wprowadź opis proxy.

Karta **Encryption** umożliwia wymaganie szyfrowanych połączeń z proxy.

Parameter	Description
Connections to proxy	Sposób, w jaki serwer łączy się z pasywnym proxy: bez szyfrowania (domyślnie), z użyciem PSK (pre-shared key) lub certyfikatu.
Connections from proxy	Wybierz, jaki typ połączeń jest dozwolony z aktywnego proxy. Można wybrać kilka typów połączeń jednocześnie (przydatne podczas testowania i przełączania na inny typ połączenia). Domyślnie: ”No encryption”.
Issuer	Dozwolony wystawca certyfikatu. Certyfikat jest najpierw weryfikowany z CA (certificate authority). Jeśli jest poprawny i podpisany przez CA, pole *Issuer* może służyć do dalszego ograniczenia dozwolonego CA. To pole jest opcjonalne i przeznaczone do użycia, jeśli instalacja Zabbix korzysta z certyfikatów od wielu CA.
Subject	Dozwolony subject certyfikatu. Certyfikat jest najpierw weryfikowany z CA. Jeśli jest poprawny i podpisany przez CA, pole *Subject* może służyć do zezwolenia tylko na jedną wartość ciągu *Subject*. Jeśli to pole jest puste, akceptowany jest każdy poprawny certyfikat podpisany przez skonfigurowane CA.
PSK identity	Ciąg identyfikacyjny pre-shared key. Nie umieszczaj wrażliwych informacji w identyfikatorze PSK, ponieważ jest on przesyłany przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć.
PSK	Pre-shared key (ciąg szesnastkowy). Maksymalna długość: 512 cyfr szesnastkowych (PSK 256-bajtowy), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, 64 cyfry szesnastkowe (PSK 32-bajtowy), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Karta **Timeouts** umożliwia zastąpienie globalnych limitów czasu dla typów pozycji, które to obsługują.



Parameter	Description
Timeouts for item types	Wybierz opcję limitu czasu: **Global** - używany jest globalny limit czasu (wyświetlany w wyszarzonym polu *Timeout* dla każdego typu pozycji); **Override** - używany jest niestandardowy limit czasu (ustawiany w polu *Timeout* dla każdego typu pozycji). Dozwolony zakres: 1 - 600 s (domyślnie: dziedziczone z globalnych limitów czasu). Obsługiwane są sufiksy czasu, np. 30s, 1m, oraz makra użytkownika. Kliknięcie łącza *Global timeouts* umożliwia skonfigurowanie globalnych limitów czasu. Zwróć uwagę, że łącze *Global timeouts* jest widoczne tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administration* → *General*. Obsługiwane typy pozycji: - Zabbix agent (zarówno kontrole pasywne, jak i aktywne) - Simple check (z wyjątkiem pozycji `icmpping*`, `vmware.*`) - SNMP agent (tylko dla pozycji SNMP `walk[OID]` i `get[OID]`) - External check - Database monitor - HTTP agent - SSH agent - TELNET agent - Script - Browser Zwróć uwagę, że limity czasu ustawione w **Override** mają pierwszeństwo przed globalnymi, ale zostaną zastąpione przez indywidualne limity czasu pozycji, jeśli zostaną ustawione w konfiguracji pozycji.

Note:
Jeśli główna wersja proxy nie jest zgodna z główną wersją serwera, obok *Timeouts for item types* zostanie wyświetlona ikona z komunikatem po najechaniu "Timeouts disabled because the proxy and server versions do not match". W takich przypadkach proxy będzie używać parametru `Timeout` z pliku konfiguracyjnego proxy.

Formularz edycji istniejącego proxy zawiera następujące dodatkowe przyciski:

- *Refresh configuration* - odśwież konfigurację proxy
- *Clone* - utwórz nowe proxy na podstawie właściwości istniejącego proxy
- *Delete* - usuń proxy

Konfiguracja hosta

Możesz określić, że pojedynczy host ma być monitorowany przez proxy lub grupę proxy w formularzu konfiguracji hosta, używając pola *Monitorowany przez*.



Masowa aktualizacja hosta to inny sposób określenia, że hosty mają być monitorowane przez proxy lub grupę proxy.

1 Synchronizacja konfiguracji monitoringu

Przegląd

Ta strona zawiera szczegółowe informacje o aktualizacji konfiguracji monitorowania dla proxy, tj. o tym, jak zmiany wprowadzone w konfiguracji monitorowania na serwerze są synchronizowane z proxy.

Aktualizacja przyrostowa

Aktualizacja konfiguracji proxy jest przyrostowa. Podczas synchronizacji konfiguracji aktualizowane są tylko zmodyfikowane encje (zatem jeśli żadne encje nie zostały zmodyfikowane, nic nie zostanie wysłane). Takie podejście pozwala oszczędzać zasoby i ustawić krótszy interwał aktualizacji konfiguracji proxy (niemal natychmiastowy).

Zmiany konfiguracji proxy są śledzone za pomocą numerów rewizji. Tylko encje z rewizjami większymi niż rewizja konfiguracji proxy są uwzględniane w danych konfiguracyjnych wysyłanych do proxy.

Encje objęte synchronizacją konfiguracji są następujące:

Encja	Szczegóły
dane TLS autorejestracji	Wszystkie dane TLS autorejestracji.
wyrażenia	Wszystkie wyrażenia (wyrażenia regularne, tabele wyrażeń).
konfiguracja globalna	Konfiguracja globalna zdefiniowana w tabeli 'config'
host	Wszystkie właściwości, interfejsy, inwentarz, pozycje, preprocessing pozycji, parametry pozycji, scenariusze WWW hosta.
makra hosta	Wszystkie makra zdefiniowane dla hosta oraz wszystkie identyfikatory szablonów z nim powiązanych.
reguła wykrywania proxy	Reguły wykrywania i kontrole przypisane do proxy.

Oznacza to, że:

- Jeśli pozycja zostanie zmieniona na **hoście**, zsynchronizowana zostanie cała konfiguracja tego hosta.
- Jeśli **wyrażenie regularne** zostanie zmienione, zsynchronizowane zostaną wszystkie wyrażenia.

Wyjątkiem są makra hosta, które są wysyłane również wtedy, gdy cokolwiek na hoście zostało zmienione.

Polecenie `-R config_cache_reload` na proxy również zainicjuje aktualizację przyrostową.

Należy pamiętać, że pełna synchronizacja konfiguracji zostanie przeprowadzona przy uruchomieniu/ponownym uruchomieniu proxy, przełączeniu awaryjnym HA, jeśli token sesji uległ zmianie lub jeśli aktualizacja konfiguracji nie powiodła się na proxy, na przykład gdy połączenie zostało przerwane podczas odbierania danych konfiguracyjnych.

Parametry konfiguracji

Parametr **ProxyConfigFrequency** określa, jak często konfiguracja proxy jest synchronizowana z serwerem (domyślnie co 10 sekund).

Należy pamiętać, że ProxyConfigFrequency jest:

- parametrem serwera dla pasywnych proxy
- parametrem proxy dla aktywnych proxy

W aktywnych proxy parametr ProxyConfigFrequency jest nowym parametrem od wersji Zabbix 6.4 i musi być używany zamiast obecnie przestarzałego parametru ConfigFrequency.

Attention:
Jeśli używane są jednocześnie ProxyConfigFrequency i ConfigFrequency, proxy zapisze błąd w logu i zakończy działanie.

Obliczanie kolejek podczas konserwacji

Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły znajdują się w sekcji Obliczanie kolejek podczas konserwacji.

2 Równoważenie obciążenia proxy i wysoka dostępność

Przegląd

Proxy Zabbix mogą być organizowane w grupy proxy, aby umożliwić równoważenie obciążenia proxy i wysoką dostępność.

Równoważenie obciążenia proxy i wysoka dostępność to automatyczna redystrybucja hostów między proxy w ramach grupy proxy:

- Jeśli proxy przejdzie w tryb offline, jego hosty zostaną przeniesione do innych proxy, co pozwoli utrzymać wysoką dostępność proxy.
- Jeśli proxy ma znacznie większą/mniejszą liczbę hostów niż inne proxy, jego hosty zostaną przeniesione do innych proxy w celu zrównoważenia obciążenia proxy.

Redystrybucja hostów działa tylko między proxy w grupie, które spełniają następujące warunki:

- Proxy działają w wersji Zabbix 7.0 lub nowszej.
- Wersja proxy jest zgodna z wersją serwera Zabbix. W przypadku używania Zabbix agent (passive) wersja proxy musi być zgodna z wersją agenta. Aktywne agenty wymagają jedynie Zabbix 7.0 lub nowszego.
- Grupa proxy ma stan online.
- Hosty są skonfigurowane do monitorowania przez grupę proxy, a nie przez pojedyncze proxy.

Note:
Stan grupy proxy można monitorować za pomocą sprawdzeń wewnętrznych przez dowolny host przypisany do grupy proxy. Jednak aby monitorować stan pojedynczego proxy w grupie, przypisz host do tego proxy; w przeciwnym razie wyniki mogą być niespójne.

Redystrybucja hostów

Równoważenie obciążenia proxy i wysoka dostępność są zarządzane przez serwer Zabbix za pomocą menedżera grup proxy, który stale monitoruje stan wszystkich proxy w każdej grupie proxy oraz rozkład ich hostów.

Wysoka dostępność proxy w obrębie grupy jest zapewniana przez mechanizm przełączania awaryjnego proxy: gdy proxy przechodzi w tryb offline, jego hosty są natychmiast redystrybuowane do innych proxy. Równoważenie obciążenia proxy również ma miejsce, ponieważ hosty są ponownie przypisywane do proxy z najmniejszą liczbą przypisanych hostów.

Dodatkowo równoważenie obciążenia proxy jest uruchamiane, gdy liczba hostów na proxy różni się od średniej grupowej o co najmniej 10 hostów oraz współczynnik 2 (nadmiar hostów lub niedobór hostów). Jeśli nierównowaga utrzymuje się po okresie karencji (10 x opóźnienie przełączenia awaryjnego), grupa proxy jest umieszczana w kolejce do redystrybucji hostów.

Menedżer grup proxy redystrybuuje hosty zgodnie z następującą logiką:

1. Oblicz średnią liczbę hostów na proxy.
2. Dla proxy z nadmiarem hostów — przenieś nadmiarowe hosty do nieprzypisanej puli proxy.
3. Dla proxy z niedoborem hostów — oblicz, ile hostów jest potrzebnych do osiągnięcia równowagi.
4. Usuń wymaganą liczbę hostów z proxy mających najwięcej hostów.
5. Przenieś nieprzypisane hosty do proxy mających najmniej hostów.

Przykłady redystrybucji hostów:

Hosty na proxy	Średnia grupowa	Ponowne przypisanie hostów
100	50	Tak
60	50	Nie
40	50	Nie
25	50	Tak

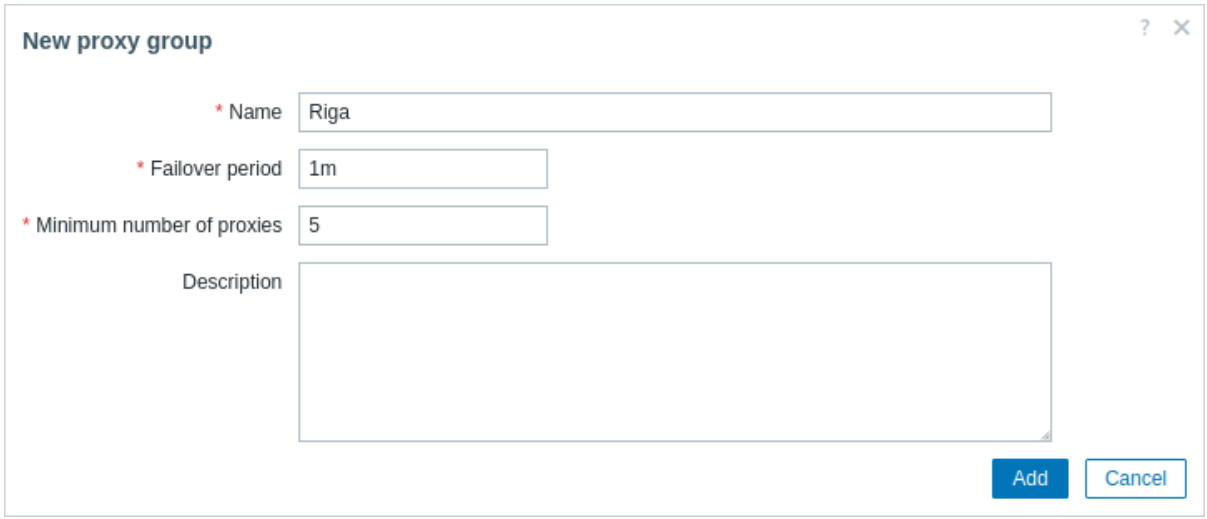
Hosty na proxy	Średnia grupowa	Ponowne przypisanie hostów
15	5	Tak
10	5	Nie

Attention:
Posiadanie mniej niż 10 hostów monitorowanych przez grupę proxy może prowadzić do nierównomiernego rozkładu hostów między proxy w grupie.

Konfigurowanie grupy proxy

Aby skonfigurować grupę proxy w frontend Zabbix:

1. Przejdź do *Administration* > *Proxy groups*
2. Kliknij *Create proxy group*



Parametr	Opis
Name	Nazwa grupy proxy.
Failover period	Okres w sekundach, w którym proxy w grupie proxy musi komunikować się z serwer Zabbix, aby było uznawane za online (domyślnie: 1m; zakres: 10s-15m). Jeśli proxy nie komunikuje się w tym okresie, jego stan zmienia się na *Offline*, a jego hosty są natychmiast redystrybuowane do innych proxy. Równoważenie obciążenia proxy rozpoczyna się po 10 x tym okresie. Obsługuje sufiksy czasu (np. 30s, 1m) oraz makra użytkownika.
Minimum number of proxies	Minimalna liczba proxy online wymagana do utrzymania grupy proxy online (domyślnie: 1; zakres: 1-1000). Obsługuje makra użytkownika. Ta wartość powinna być mniejsza niż całkowita liczba proxy w grupie. Na przykład w grupie 10 proxy ustawienie minimum na 10 spowoduje przejście grupy w stan offline w przypadku awarii dowolnego proxy. Należy pamiętać, że proxy online w grupie offline nadal działają normalnie, ale równoważenie obciążenia/wysoka dostępność nie będą realizowane.
Description	Opis grupy proxy.
Proxies	Wyświetla listę maksymalnie pięciu proxy (jako linki lub zwykły tekst, w zależności od uprawnień użytkownika do proxy) podczas edycji grupy zawierającej proxy.

Konfigurowanie równoważenia obciążenia proxy

Aby używać równoważenia obciążenia proxy, należy skonfigurować grupę proxy w Zabbix frontend (zobacz wyżej) i upewnić się, że hosty są monitorowane przez grupę proxy, a nie przez pojedyncze proxy (można użyć masowej aktualizacji hostów, aby przenieść hosty z proxy do grupy proxy).

W przypadku korzystania z Zabbix agent należy również skonfigurować go w następujący sposób:

- Dla kontroli pasywnych należy wymienić wszystkie proxy z grupy proxy w parametrze Server.
- Dla kontroli aktywnych zaleca się wymienienie wszystkich proxy z grupy proxy lub serwera Zabbix w parametrze ServerActive. Należy pamiętać, że tylko Zabbix agent 7.0 (lub nowszy) będzie działać z grupami proxy w trybie aktywnym.

Jeśli parametr `ServerActive` zawiera tylko jedno proxy z grupy proxy (lub serwer Zabbix), agent nadal będzie mógł połączyć się z właściwym proxy. Gdy usługa agenta uruchomi się i połączy z określonym proxy, agent otrzyma i zapisze w pamięci podręcznej pełną listę adresów IP proxy oraz ich bieżące obciążenie w grupie. Następnie kontrole aktywne zostaną przekierowane do właściwego dostępnego proxy dla hosta, na podstawie bieżącego przypisania proxy-host w grupie proxy.

Warning:
Podanie tylko jednego proxy w parametrze `ServerActive` Zabbix agent może prowadzić do utraty danych monitorowania, jeśli agent zostanie uruchomiony/zrestartowany, gdy wskazane proxy będzie niedostępne.

Note:
Podczas korzystania z Zabbix sender żądania danych są również przekierowywane do właściwego dostępnego proxy dla hosta, na podstawie bieżącego przypisania proxy-host w grupie proxy. Jeśli jednak wysyłasz wartości wielu hostów z pliku wejściowego, użyj opcji −g, aby zapobiec wysłaniu danych do niewłaściwego proxy.

Zabbix agent musi również mieć możliwość łączenia się przez zaporę sieciową ze wszystkimi proxy w grupie proxy. W przeciwnym razie kontrole aktywne mogą zawieszać się lub kończyć niepowodzeniem podczas przekierowania albo przełączenia awaryjnego. Na przykład:

- Podczas kontroli aktywnych proxy może przekierować agenta do innego proxy. Jeśli to proxy jest blokowane przez zaporę sieciową, komunikacja zawiesi się podczas oczekiwania na odpowiedź.
- W stabilnych konfiguracjach wysokiej dostępności bez niedawnego równoważenia agenty mogą nigdy nie kontaktować się z zapasowymi proxy. Jeśli reguły zapory sieciowej uległy zmianie i nie zostały przetestowane, przełączenie awaryjne może się nie powieść.

Testowanie równoważenia obciążenia proxy

Aby przetestować równoważenie obciążenia proxy:

1. Skonfiguruj grupę proxy.
2. Upewnij się, że grupa proxy ma stan online.
3. Upewnij się, że hosty są monitorowane przez grupę proxy, a nie przez pojedyncze proxy (możesz użyć masowej aktualizacji, aby przenieść hosty z proxy do grupy proxy).
4. Odczekaj kilka sekund na aktualizację konfiguracji i dystrybucję hostów między proxy w grupie proxy. Obserwuj zmianę, odświeżając listę hostów w *Administracja* > *Proxy*.

Ważne uwagi

- Pułapki SNMP nie są obsługiwane przez proxy w grupie proxy.
- Kontrole zależne od konfiguracji zewnętrznej (np. skrypty dla kontroli zewnętrznych lub konfiguracja ODBC dla kontroli bazy danych) muszą mieć taką samą konfigurację na wszystkich proxy w grupie proxy.
- Kontrole bazy danych wymagają rozszerzonych uprawnień do obiektu bazy danych/serwera.
- Hosty VMware monitorowane przez grupę proxy będą losowo rozdzielane między proxy w grupie. Powoduje to, że każdy proxy buforuje wszystkie dane VMware, co skutkuje dodatkowym obciążeniem vCenter.
- Hosty utworzone na podstawie danych autorejestracji z proxy w grupie proxy są ustawiane jako monitorowane przez tę grupę proxy. Jednak hosty utworzone na podstawie danych wykrywania sieci z proxy w grupie proxy są ustawiane jako monitorowane przez ten proxy.

17 Szyfrowanie

Przegląd Zabbix obsługuje szyfrowaną komunikację między komponentami Zabbix z użyciem protokołu Transport Layer Security (TLS) w wersji 1.2 i 1.3 (w zależności od biblioteki kryptograficznej). Obsługiwane jest szyfrowanie oparte na certyfikatach oraz na kluczach współdzielonych.

Szyfrowanie można skonfigurować dla połączeń:

- Między Zabbix server, Zabbix proxy, Zabbix agent, usługą Zabbix web service oraz narzędziami zabbix_sender i zabbix_get
- Z bazą danych Zabbix z poziomu Zabbix frontend oraz server/proxy
- Między Zabbix frontend a Zabbix server

Szyfrowanie jest opcjonalne i można je skonfigurować dla poszczególnych komponentów:

- Niektóre proxy i agenty mogą być skonfigurowane do używania szyfrowania opartego na certyfikatach w komunikacji z serwerem, podczas gdy inne mogą używać szyfrowania opartego na kluczach współdzielonych, a jeszcze inne mogą nadal korzystać z nieszyfrowanej komunikacji (jak dotychczas).

- Serwer (proxy) może używać różnych konfiguracji szyfrowania dla różnych hostów.

Programy daemon Zabbix używają jednego portu nasłuchującego zarówno dla szyfrowanych, jak i nieszyfrowanych połączeń przychodzących. Dodanie szyfrowania nie wymaga otwierania nowych portów w zaporach sieciowych.

Ograniczenia

- Klucze prywatne są przechowywane w postaci zwykłego tekstu w plikach dostępnych do odczytu przez komponenty Zabbix podczas uruchamiania.
- Klucze współdzielone są wprowadzane w Zabbix frontend i przechowywane w bazie danych Zabbix w postaci zwykłego tekstu.
- Wbudowane szyfrowanie nie chroni komunikacji między serwerem WWW uruchamiającym Zabbix frontend a przeglądarką internetową użytkownika.
- Obecnie każde szyfrowane połączenie jest otwierane z pełnym uzgadnianiem TLS; nie są zaimplementowane buforowanie sesji ani bilety.
- Dodanie szyfrowania wydłuża czas sprawdzania pozycji i wykonywania akcji, w zależności od opóźnień sieciowych:
  - Na przykład, jeśli opóźnienie pakietów wynosi 100 ms, to otwarcie połączenia TCP i wysłanie niezaszyfrowanego żądania zajmuje około 200 ms. Przy szyfrowaniu dochodzi około 1000 ms na ustanowienie połączenia TLS.
  - Może być konieczne zwiększenie limitów czasu; w przeciwnym razie niektóre pozycje i akcje uruchamiające zdalne skrypty na agentach mogą działać przy połączeniach nieszyfrowanych, ale kończyć się błędem przekroczenia limitu czasu przy połączeniach szyfrowanych.
- Szyfrowanie nie jest obsługiwane przez wykrywanie sieci. Kontrole Zabbix agent wykonywane przez wykrywanie sieci będą nieszyfrowane, a jeśli Zabbix agent jest skonfigurowany tak, aby odrzucać nieszyfrowane połączenia, takie kontrole nie zakończą się powodzeniem.

Kompilowanie Zabbix z obsługą szyfrowania Aby obsługiwać szyfrowanie, Zabbix musi zostać skompilowany i połączony z jedną z obsługiwanych bibliotek kryptograficznych:

- GnuTLS - od wersji 3.1.18
- OpenSSL - wersje 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x
- LibreSSL - testowano z wersjami 2.7.4, 2.8.2:
  - LibreSSL 2.6.x nie jest obsługiwany
  - LibreSSL jest obsługiwany jako kompatybilny zamiennik OpenSSL; nowe funkcje API `tls_*()` specyficzne dla LibreSSL nie są używane. Komponenty Zabbix skompilowane z LibreSSL nie będą mogły używać PSK, będzie można używać wyłącznie certyfikatów.

Note:
Więcej informacji o konfiguracji SSL dla frontend Zabbix można znaleźć w tych najlepszych praktykach.

Bibliotekę wybiera się przez podanie odpowiedniej opcji do skryptu „configure”:

- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]` (używane również dla LibreSSL)

Na przykład, aby skonfigurować źródła dla serwer i agent z użyciem *OpenSSL*, można użyć polecenia podobnego do poniższego:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Różne komponenty Zabbix mogą być kompilowane z różnymi bibliotekami kryptograficznymi (np. serwer z *OpenSSL*, agent z *GnuTLS*).

Attention:
Jeśli planujesz używać kluczy współdzielonych (PSK), rozważ użycie bibliotek *GnuTLS* lub *OpenSSL 1.1.0* (albo nowszych) w komponentach Zabbix korzystających z PSK. Biblioteki *GnuTLS* i *OpenSSL 1.1.0* obsługują pakiety szyfrów PSK z Perfect Forward Secrecy. Starsze wersje biblioteki *OpenSSL* (1.0.1, 1.0.2c) również obsługują PSK, ale dostępne pakiety szyfrów PSK nie zapewniają Perfect Forward Secrecy.

Zarządzanie szyfrowaniem połączeń Połączenia w Zabbix mogą używać:

- braku szyfrowania (domyślnie)
- szyfrowania opartego na certyfikatach RSA
- szyfrowania opartego na PSK

Istnieją dwa ważne parametry używane do określania szyfrowania między komponentami Zabbix:

- TLSConnect - określa, jakiego szyfrowania używać dla połączeń wychodzących (bez szyfrowania, PSK lub certyfikat)

- TLSAccept - określa, jakie typy połączeń są dozwolone dla połączeń przychodzących (bez szyfrowania, PSK lub certyfikat). Można podać jedną lub więcej wartości.

`TLSConnect` jest używany w plikach konfiguracyjnych Zabbix proxy (w trybie aktywnym określa tylko połączenia do serwer) oraz Zabbix agent (dla aktywnych kontroli). W Zabbix frontend odpowiednikiem TLSConnect jest pole *Connections to host* na karcie *Data collection → Hosts → <some host> → Encryption* oraz pole *Connections to proxy* na karcie *Administration → Proxies → <some proxy> → Encryption*. Jeśli skonfigurowany typ szyfrowania dla połączenia zawiedzie, żadne inne typy szyfrowania nie będą próbowane.

`TLSAccept` jest używany w plikach konfiguracyjnych Zabbix proxy (w trybie pasywnym określa tylko połączenia z serwer) oraz Zabbix agent (dla pasywnych kontroli). W Zabbix frontend odpowiednikiem TLSAccept jest pole *Connections from host* na karcie *Data collection → Hosts → <some host> → Encryption* oraz pole *Connections from proxy* na karcie *Administration → Proxies → <some proxy> → Encryption*.

Zwykle konfiguruje się tylko jeden typ szyfrowania dla połączeń przychodzących. Możesz jednak chcieć zmienić typ szyfrowania, np. z braku szyfrowania na szyfrowanie oparte na certyfikatach, przy minimalnym czasie niedostępności i z możliwością wycofania zmian. Aby to osiągnąć:

- Ustaw `TLSAccept=unencrypted,cert` w pliku konfiguracyjnym agent i uruchom ponownie Zabbix agent
- Przetestuj połączenie z agent za pomocą zabbix_get, używając certyfikatu. Jeśli działa, możesz ponownie skonfigurować szyfrowanie dla tego agent w Zabbix frontend na karcie *Data collection → Hosts → <some host> → Encryption*, ustawiając *Connections to host* na "Certificate".
- Gdy pamięć podręczna konfiguracji serwer zostanie zaktualizowana (a konfiguracja proxy również zostanie zaktualizowana, jeśli host jest monitorowany przez proxy), połączenia z tym agent będą szyfrowane
- Jeśli wszystko działa zgodnie z oczekiwaniami, możesz ustawić `TLSAccept=cert` w pliku konfiguracyjnym agent i uruchomić ponownie Zabbix agent. Teraz agent będzie akceptował tylko szyfrowane połączenia oparte na certyfikatach. Połączenia bez szyfrowania oraz oparte na PSK będą odrzucane.

W podobny sposób działa to na serwer i proxy. Jeśli w Zabbix frontend w konfiguracji host pole *Connections from host* jest ustawione na "Certificate", to od agent (aktywne kontrole) i zabbix_sender (trapper pozycja) będą akceptowane tylko szyfrowane połączenia oparte na certyfikatach.

Najprawdopodobniej skonfigurujesz połączenia przychodzące i wychodzące tak, aby używały tego samego typu szyfrowania albo w ogóle bez szyfrowania. Technicznie jednak możliwa jest konfiguracja asymetryczna, np. szyfrowanie oparte na certyfikatach dla połączeń przychodzących i oparte na PSK dla połączeń wychodzących.

Konfiguracja szyfrowania dla każdego host jest wyświetlana w Zabbix frontend, w *Data collection → Hosts* w kolumnie *Agent encryption*. Na przykład:

Przykład	Connections to host	Allowed connections from host	Rejected connections from host
NONE	Bez szyfrowania	Bez szyfrowania	Szyfrowane, oparte na certyfikatach i oparte na PSK
CERT NONE PSK CERT	Szyfrowane, oparte na certyfikatach	Szyfrowane, oparte na certyfikatach	Bez szyfrowania i szyfrowane oparte na PSK
PSK NONE PSK CERT	Szyfrowane, oparte na PSK	Szyfrowane, oparte na PSK	Bez szyfrowania i szyfrowane oparte na certyfikatach
PSK NONE PSK CERT	Szyfrowane, oparte na PSK	Bez szyfrowania i szyfrowane oparte na PSK	Szyfrowane oparte na certyfikatach
CERT NONE PSK CERT	Szyfrowane, oparte na certyfikatach	Bez szyfrowania, szyfrowane oparte na PSK lub oparte na certyfikatach	-

Attention:
Połączenia domyślnie nie są szyfrowane. Szyfrowanie musi być skonfigurowane indywidualnie dla każdego host i proxy.

zabbix_get i zabbix_sender z szyfrowaniem Informacje o używaniu zabbix_get i zabbix_sender z szyfrowaniem można znaleźć w ich stronach podręcznika man.

Zestawy szyfrów Domyślnie zestawy szyfrów są konfigurowane wewnętrznie podczas uruchamiania Zabbix.

Obsługiwane są również zestawy szyfrów konfigurowane przez użytkownika dla GnuTLS i OpenSSL. Użytkownicy mogą skonfigurować zestawy szyfrów zgodnie ze swoimi politykami bezpieczeństwa. Korzystanie z tej funkcji jest opcjonalne (wbudowane domyślne zestawy szyfrów nadal działają).

W przypadku bibliotek kryptograficznych skompilowanych z ustawieniami domyślnymi wbudowane reguły Zabbix zwykle skutkują następującymi zestawami szyfrów (w kolejności od najwyższego do najniższego priorytetu):

Biblioteka	Zestawy szyfrów certyfikatów	Zestawy szyfrów PSK
GnuTLS 3.1.18	TLS_ECDHE_RSA_AES_128_GCM_SHA256 TLS_ECDHE_RSA_AES_128_CBC_SHA256 TLS_ECDHE_RSA_AES_128_CBC_SHA1 TLS_RSA_AES_128_GCM_SHA256 TLS_RSA_AES_128_CBC_SHA256 TLS_RSA_AES_128_CBC_SHA1	TLS_ECDHE_PSK_AES_128_CBC_SHA256 TLS_ECDHE_PSK_AES_128_CBC_SHA1 TLS_PSK_AES_128_GCM_SHA256 TLS_PSK_AES_128_CBC_SHA256 TLS_PSK_AES_128_CBC_SHA1
OpenSSL 1.0.2c	ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA	PSK-AES128-CBC-SHA
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

Pakiety szyfrów skonfigurowane przez użytkownika Wbudowane kryteria wyboru pakietów szyfrów mogą zostać nadpisane przez pakiety szyfrów skonfigurowane przez użytkownika.

Attention:
Pakiety szyfrów skonfigurowane przez użytkownika to funkcja przeznaczona dla zaawansowanych użytkowników, którzy rozumieją pakiety szyfrów TLS, ich bezpieczeństwo oraz konsekwencje błędów, a także dobrze radzą sobie z rozwiązywaniem problemów z TLS.

Wbudowane kryteria wyboru pakietów szyfrów można nadpisać za pomocą następujących parametrów:

Zakres nadpisania	Parametr	Wartość	Opis
Wybór pakietów szyfrów dla certyfikatów	TLSCipherCert13	Prawidłowe cipher strings OpenSSL 1.1.1 dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru pakietów szyfrów opartych na certyfikatach dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.
	TLSCipherCert	Prawidłowe cipher strings OpenSSL dla TLS 1.2 lub prawidłowe priority strings GnuTLS. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru pakietów szyfrów opartych na certyfikatach dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Wybór pakietów szyfrów dla PSK	TLSCipherPSK13	Prawidłowe cipher strings OpenSSL 1.1.1 dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru pakietów szyfrów opartych na PSK dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.

Zakres nadpisania	Parametr	Wartość	Opis
	TLSCipherPSK	Prawidłowe cipher strings OpenSSL dla TLS 1.2 lub prawidłowe priority strings GnuTLS. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru pakietów szyfrów opartych na PSK dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Połączona lista pakietów szyfrów dla certyfikatu i PSK	TLSCipherAll13	Prawidłowe cipher strings OpenSSL 1.1.1 dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru pakietów szyfrów dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.
	TLSCipherAll	Prawidłowe cipher strings OpenSSL dla TLS 1.2 lub prawidłowe priority strings GnuTLS. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru pakietów szyfrów dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Aby nadpisać wybór pakietów szyfrów w narzędziach zabbix_get i zabbix_sender, użyj parametrów wiersza poleceń:

- `--tls-cipher13`
- `--tls-cipher`

Nowe parametry są opcjonalne. Jeśli parametr nie zostanie określony, zostanie użyta wewnętrzna wartość domyślna. Jeśli parametr jest zdefiniowany, nie może być pusty.

Jeśli ustawienie wartości TLSCipher* w bibliotece kryptograficznej zakończy się niepowodzeniem, serwer, proxy lub agent nie uruchomi się, a błąd zostanie zapisany w logu.

Ważne jest, aby rozumieć, kiedy każdy parametr ma zastosowanie.

Połączenia wychodzące

Najprostszy przypadek to połączenia wychodzące:

- Dla połączeń wychodzących z certyfikatem użyj TLSCipherCert13 lub TLSCipherCert
- Dla połączeń wychodzących z PSK użyj TLSCipherPSK13 lub TLSCipherPSK
- W przypadku narzędzi zabbix_get i zabbix_sender można użyć parametrów wiersza poleceń `--tls-cipher13` lub `--tls-cipher` (szyfrowanie jest jednoznacznie określone parametrem `--tls-connect`)

Połączenia przychodzące

W przypadku połączeń przychodzących jest to nieco bardziej skomplikowane, ponieważ reguły są specyficzne dla komponentów i konfiguracji.

Dla Zabbix **agent**:

Konfiguracja połączenia agenta	Konfiguracja szyfrów
TLSConnect=cert	TLSCipherCert, TLSCipherCert13
TLSConnect=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert	TLSCipherCert, TLSCipherCert13
TLSAccept=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert,psk	TLSCipherAll, TLSCipherAll13

Dla Zabbix **serwer** i **proxy**:

Konfiguracja połączenia	Konfiguracja szyfrów
Połączenia wychodzące używające PSK	TLSCipherPSK, TLSCipherPSK13
Połączenia przychodzące używające certyfikatów	TLSCipherAll, TLSCipherAll13
Połączenia przychodzące używające PSK, jeśli serwer nie ma certyfikatu	TLSCipherPSK, TLSCipherPSK13

Konfiguracja połączenia	Konfiguracja szyfrów
Połączenia przychodzące używające PSK, jeśli serwer ma certyfikat	TLSCipherAll, TLSCipherAll13

W dwóch powyższych tabelach można zauważyć pewną prawidłowość:

- TLSCipherAll i TLSCipherAll13 można określić tylko wtedy, gdy używana jest połączona lista pakietów szyfrów opartych na certyfikatach **i** PSK. Występuje to w dwóch przypadkach: serwer (proxy) ze skonfigurowanym certyfikatem (pakiety szyfrów PSK są zawsze skonfigurowane na serwerze, proxy, jeśli biblioteka kryptograficzna obsługuje PSK), agent skonfigurowany do akceptowania zarówno przychodzących połączeń opartych na certyfikatach, jak i PSK
- w pozostałych przypadkach wystarczające są TLSCipherCert* i/lub TLSCipherPSK*

Poniższe tabele pokazują wbudowane wartości domyślne `TLSCipher*`. Mogą one stanowić dobry punkt wyjścia do zdefiniowania własnych wartości niestandardowych.

Parametr	GnuTLS 3.6.12
TLSCipherCert	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
TLSCipherPSK	NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIGN-ALL
TLSCipherAll	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

Parametr	OpenSSL 1.1.1d [1]
TLSCipherCert13	
TLSCipherCert	EECDH+aRSA+AES128:RSA+aRSA+AES128
TLSCipherPSK13	TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
TLSCipherPSK	kECDHEPSK+AES128:kPSK+AES128
TLSCipherAll13	
TLSCipherAll	EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128

[1] Wartości domyślne różnią się w przypadku starszych wersji OpenSSL (1.0.1, 1.0.2, 1.1.0), LibreSSL oraz gdy OpenSSL jest skompilowany bez obsługi PSK.

Przykłady pakietów szyfrów skonfigurowanych przez użytkownika

Poniżej znajdują się przykłady pakietów szyfrów skonfigurowanych przez użytkownika:

- Testowanie ciągów szyfrów i dopuszczanie tylko pakietów szyfrów z PFS
- Przełączanie z AES128 na AES256

Testowanie ciągów szyfrów i zezwalanie tylko na pakiety szyfrów PFS

Aby zobaczyć, które pakiety szyfrów zostały wybrane, należy ustawić `DebugLevel=4` w pliku konfiguracyjnym lub użyć opcji `-vv` dla zabbix_sender.

Przed uzyskaniem żądanych pakietów szyfrów może być konieczne poeksperymentowanie z parametrami `TLSCipher*`. Wielokrotne restartowanie serwera Zabbix, proxy lub agent tylko po to, aby dostosować parametry `TLSCipher*`, jest niewygodne. Wygodniejszymi opcjami są użycie zabbix_sender lub polecenia `openssl`. Pokażmy oba sposoby.

1. Użycie zabbix_sender.

Utwórzmy testowy plik konfiguracyjny, na przykład `/home/zabbix/test.conf`, ze składnią pliku `zabbix_agentd.conf`:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=cert
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/agent.crt
TLSKeyFile=/home/zabbix/agent.key
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agent.psk
```

Do tego przykładu potrzebne są prawidłowe certyfikaty CA i agent oraz PSK. Dostosuj ścieżki i nazwy plików certyfikatów oraz PSK do swojego środowiska.

Jeśli nie używasz certyfikatów, a tylko PSK, możesz utworzyć prostszy plik testowy:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=psk
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agentd.psk
```

Wybrane pakiety szyfrów można zobaczyć, uruchamiając zabbix_sender (przykład skompilowany z OpenSSL 1.1.d):

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k nonexisting_item -o 1 2>&1 | grep ciphersuites
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() certificate ciphersuites: TLS_AES_256_GCM_SHA384 TLS_
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() PSK ciphersuites: TLS_CHACHA20_POLY1305_SHA256 TLS_AE
zabbix_sender [41271]: DEBUG: zbx_tls_init_child() certificate and PSK ciphersuites: TLS_AES_256_GCM_SHA
```

Tutaj widać pakiety szyfrów wybrane domyślnie. Te wartości domyślne zostały dobrane tak, aby zapewnić interoperacyjność z agentami Zabbix działającymi w systemach ze starszymi wersjami OpenSSL (od 1.0.1).

W nowszych systemach można zwiększyć bezpieczeństwo, zezwalając tylko na kilka pakietów szyfrów, np. tylko na pakiety szyfrów z PFS (Perfect Forward Secrecy). Spróbujmy zezwolić tylko na pakiety szyfrów z PFS przy użyciu parametrów `TLSCipher*`.

Attention:
Wynik nie będzie interoperacyjny z systemami używającymi OpenSSL 1.0.1 i 1.0.2, jeśli używany jest PSK. Szyfrowanie oparte na certyfikatach powinno działać.

Dodaj dwie linie do pliku konfiguracyjnego `test.conf`:

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
```

i przetestuj ponownie:

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k nonexisting_item -o 1 2>&1 | grep ciphersuites
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() certificate ciphersuites: TLS_AES_256_GCM_SHA384 TLS_
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() PSK ciphersuites: TLS_CHACHA20_POLY1305_SHA256 TLS_AE
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() certificate and PSK ciphersuites: TLS_AES_256_GCM_SHA
```

Listy „certificate ciphersuites" i „PSK ciphersuites" uległy zmianie — są krótsze niż wcześniej i zawierają tylko pakiety szyfrów TLS 1.3 oraz pakiety szyfrów TLS 1.2 ECDHE-*, zgodnie z oczekiwaniami.

2. TLSCipherAll i TLSCipherAll13 nie mogą być testowane za pomocą zabbix_sender; nie wpływają one na wartość „certificate and PSK ciphersuites" pokazaną w powyższym przykładzie. Aby dostroić TLSCipherAll i TLSCipherAll13, trzeba eksperymentować z agent, proxy lub serwer.

Aby więc zezwolić tylko na pakiety szyfrów PFS, może być konieczne dodanie maksymalnie trzech parametrów

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
TLSCipherAll=EECDH+aRSA+AES128:kECDHEPSK+AES128
```

do zabbix_agentd.conf, zabbix_proxy.conf i zabbix_server_conf, jeśli każdy z nich ma skonfigurowany certyfikat, a agent ma również PSK.

Jeśli środowisko Zabbix używa wyłącznie szyfrowania opartego na PSK i nie używa certyfikatów, wystarczy tylko jeden parametr:

```
TLSCipherPSK=kECDHEPSK+AES128
```

Teraz, gdy już rozumiesz, jak to działa, możesz testować wybór pakietów szyfrów nawet poza Zabbix, za pomocą polecenia `openssl`. Przetestujmy wszystkie trzy wartości parametrów `TLSCipher*`:

```
$ openssl ciphers EECDH+aRSA+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 E
$ openssl ciphers kECDHEPSK+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-PSK-AES128-CBC-SHA256 E
$ openssl ciphers EECDH+aRSA+AES128:kECDHEPSK+AES128 | sed 's/:/ /g'
TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 E
```

Możesz preferować `openssl ciphers` z opcją `-V`, aby uzyskać bardziej szczegółowe dane wyjściowe:

```
$ openssl ciphers -V EECDH+aRSA+AES128:kECDHEPSK+AES128
           0x13,0x02 - TLS_AES_256_GCM_SHA384  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(256) Mac=AEAD
           0x13,0x03 - TLS_CHACHA20_POLY1305_SHA256 TLSv1.3 Kx=any      Au=any  Enc=CHACHA20/POLY1305(256
           0x13,0x01 - TLS_AES_128_GCM_SHA256  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(128) Mac=AEAD
           0xC0,0x2F - ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH     Au=RSA  Enc=AESGCM(128) Mac=AEAD
           0xC0,0x27 - ECDHE-RSA-AES128-SHA256 TLSv1.2 Kx=ECDH     Au=RSA  Enc=AES(128)  Mac=SHA256
           0xC0,0x13 - ECDHE-RSA-AES128-SHA    TLSv1 Kx=ECDH     Au=RSA  Enc=AES(128)  Mac=SHA1
           0xC0,0x37 - ECDHE-PSK-AES128-CBC-SHA256 TLSv1 Kx=ECDHEPSK Au=PSK  Enc=AES(128)  Mac=SHA256
           0xC0,0x35 - ECDHE-PSK-AES128-CBC-SHA TLSv1 Kx=ECDHEPSK Au=PSK  Enc=AES(128)  Mac=SHA1
```

Podobnie można testować ciągi priorytetów dla GnuTLS:

```
$ gnutls-cli -l --priority=NONE:+VERS-TLS1.2:+ECDHE-RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+CURVE-A
Cipher suites for NONE:+VERS-TLS1.2:+ECDHE-RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+CURVE-ALL:+COMP-
TLS_ECDHE_RSA_AES_128_GCM_SHA256                        0xc0, 0x2f      TLS1.2
TLS_ECDHE_RSA_AES_128_CBC_SHA256                        0xc0, 0x27      TLS1.2

Protocols: VERS-TLS1.2
Ciphers: AES-128-GCM, AES-128-CBC
MACs: AEAD, SHA256
Key Exchange Algorithms: ECDHE-RSA
Groups: GROUP-SECP256R1, GROUP-SECP384R1, GROUP-SECP521R1, GROUP-X25519, GROUP-X448, GROUP-FFDHE2048, GR
PK-signatures: SIGN-RSA-SHA256, SIGN-RSA-PSS-SHA256, SIGN-RSA-PSS-RSAE-SHA256, SIGN-ECDSA-SHA256, SIGN-E
```

Przełączanie z AES128 na AES256

Zabbix używa AES128 jako wbudowanego domyślnego szyfrowania danych. Załóżmy, że używasz certyfikatów i chcesz przełączyć się na AES256 w OpenSSL 1.1.1.

Można to osiągnąć, dodając odpowiednie parametry w `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
```

Attention:
Chociaż używane będą tylko pakiety szyfrów związane z certyfikatami, parametry `TLSCipherPSK*` również są zdefiniowane, aby uniknąć ich wartości domyślnych, które obejmują mniej bezpieczne szyfry dla szerszej interoperacyjności. Pakietów szyfrów PSK nie można całkowicie wyłączyć na serwer/proxy.

A w `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 Korzystanie z certyfikatów

Przegląd

Zabbix może używać certyfikatów RSA w formacie PEM, podpisanych przez publiczny lub wewnętrzny urząd certyfikacji (CA).

Weryfikacja certyfikatu jest wykonywana względem wstępnie skonfigurowanego certyfikatu CA. Opcjonalnie można używać list unieważnionych certyfikatów (CRL).

Każdy komponent Zabbix może mieć skonfigurowany tylko jeden certyfikat.

Aby uzyskać więcej informacji na temat konfigurowania i obsługi wewnętrznego CA, generowania i podpisywania żądań certyfikatów oraz unieważniania certyfikatów, zapoznaj się z samouczkami, takimi jak OpenSSL PKI Tutorial v2.0.

Dokładnie rozważ i przetestuj rozszerzenia swoich certyfikatów. Więcej szczegółów znajduje się w sekcji Ograniczenia dotyczące używania rozszerzeń certyfikatów X.509 v3.

Parametry konfiguracji certyfikatów

Następujące parametry konfiguracji są obsługiwane podczas konfigurowania certyfikatów na komponentach Zabbix.

Parameter	Mandatory	Description
TLSCAFile	yes	Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnego urzędu CA do weryfikacji certyfikatu strony przeciwnej. Jeśli używany jest łańcuch certyfikatów z wieloma elementami, certyfikaty należy uporządkować tak, aby najpierw znajdowały się certyfikaty CA niższego poziomu, a następnie certyfikaty CA wyższego poziomu. Certyfikaty z wielu urzędów CA mogą być umieszczone w jednym pliku.
TLSCRLFile	no	Pełna ścieżka do pliku zawierającego listy unieważnionych certyfikatów (CRL).
TLSCertFile	yes	Pełna ścieżka do pliku zawierającego certyfikat. Jeśli używany jest łańcuch certyfikatów z wieloma elementami, certyfikaty należy uporządkować tak, aby najpierw znajdował się certyfikat serwera, proxy lub agenta, następnie certyfikaty CA niższego poziomu, a na końcu certyfikaty CA wyższego poziomu.
TLSKeyFile	yes	Pełna ścieżka do pliku zawierającego klucz prywatny. Upewnij się, że ten plik jest czytelny wyłącznie dla użytkownika Zabbix, ustawiając odpowiednie uprawnienia dostępu.
TLSServerCertIssuer	no	Dozwolony wystawca certyfikatu serwera.
TLSServerCertSubject	no	Dozwolony podmiot certyfikatu serwera.

Przykłady konfiguracji

Po skonfigurowaniu wymaganych certyfikatów skonfiguruj komponenty Zabbix tak, aby używały szyfrowania opartego na certyfikatach.

Poniżej znajdują się szczegółowe kroki konfiguracji dla:

- serwera Zabbix
- proxy Zabbix
- agenta Zabbix

Serwer Zabbix

1. Przygotuj plik certyfikatu CA.

Aby weryfikować certyfikaty partnera, serwer Zabbix musi mieć dostęp do pliku zawierającego certyfikaty głównych, samopodpisanych głównych urzędów certyfikacji CA. Na przykład, jeśli wymagane są certyfikaty z dwóch niezależnych głównych urzędów CA, umieść je w pliku `/home/zabbix/zabbix_ca_file.crt`:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
```

```
            X509v3 Basic Constraints: critical
                CA:TRUE
            ...
-----BEGIN CERTIFICATE-----
MIID2jCCAsKgAwIBAgIBATANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
....
9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVmO
-----END CERTIFICATE-----
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ....
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE
            ....
-----BEGIN CERTIFICATE-----
MIID3DCCAsSgAwIBAgIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQB
...
vdGNYoSfvu41GQAR5Vj5FnRJRzv5XQOZ3B6894GY1zY=
-----END CERTIFICATE-----
```

2. Umieść certyfikat/łańcuch certyfikatów serwera Zabbix w pliku, na przykład w `/home/zabbix/zabbix_server.crt`. Pierwszy certyfikat to certyfikat serwera Zabbix, po którym następuje certyfikat pośredniego urzędu CA:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Zabbix server
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment
            X509v3 Basic Constraints:
                CA:FALSE
            ...
-----BEGIN CERTIFICATE-----
MIIECDCCAvCgAwIBAgIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixk
...
h02u1GHiy46GI+xfR3LsPwFKlkTaaLaL/6aaoQ==
-----END CERTIFICATE-----
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
    Signature Algorithm: sha1WithRSAEncryption
```

```
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ...
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE, pathlen:0
        ...
-----BEGIN CERTIFICATE-----
MIID4TCCAsmgAwIBAgIBAjANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
...
dyCeWnvL7u5sd6ffo8iRny0QzbHKmQt/wUtcVIvWXdMIFJM0Hw==
-----END CERTIFICATE-----
```

Note:
Używaj wyłącznie atrybutów wymienionych powyżej zarówno dla certyfikatów klienta, jak i serwera, aby nie wpływać na proces weryfikacji certyfikatu. Na przykład OpenSSL może nie ustanowić szyfrowanego połączenia, jeśli używane są rozszerzenia *X509v3 Subject Alternative Name* lub *Netscape Cert Type*. Więcej informacji znajduje się w sekcji Ograniczenia dotyczące używania rozszerzeń certyfikatów X.509 v3.

3. Umieść klucz prywatny serwera Zabbix w pliku, na przykład w `/home/zabbix/zabbix_server.key`:

```
-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBKowggSmAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEf47MLhffWa7XvZTY=
-----END PRIVATE KEY-----
```

4. Edytuj parametry konfiguracji TLS w pliku konfiguracyjnym serwera Zabbix:

```
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key
```

Zabbix proxy

1. Przygotuj pliki z certyfikatami głównego urzędu certyfikacji (CA), certyfikatem/łańcuchem certyfikatów Zabbix proxy oraz kluczem prywatnym, zgodnie z opisem w sekcji Zabbix server. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym Zabbix proxy.

2. Edytuj dodatkowe parametry TLS w pliku konfiguracyjnym Zabbix proxy:

- Dla aktywnego proxy: `TLSConnect=cert`
- Dla pasywnego proxy: `TLSAccept=cert`

Note:
Aby zwiększyć bezpieczeństwo proxy, możesz również ustawić parametry `TLSServerCertIssuer` i `TLSServerCertSubject`. Więcej informacji znajduje się w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

Parametry TLS w końcowym pliku konfiguracyjnym proxy mogą wyglądać następująco:

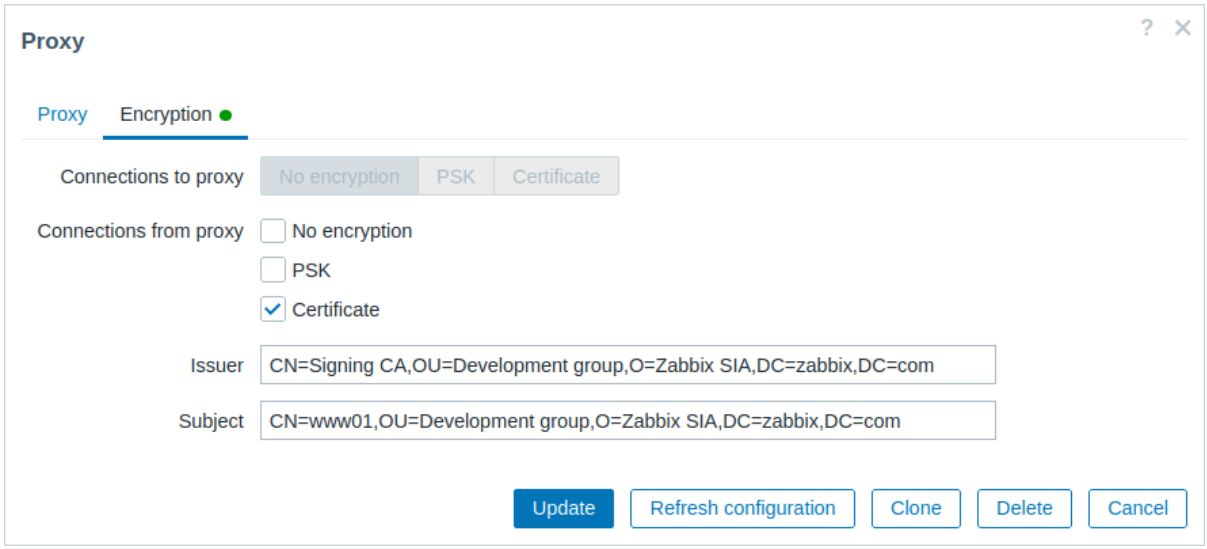
```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key
```

3. Skonfiguruj szyfrowanie dla tego proxy w Zabbix frontend:

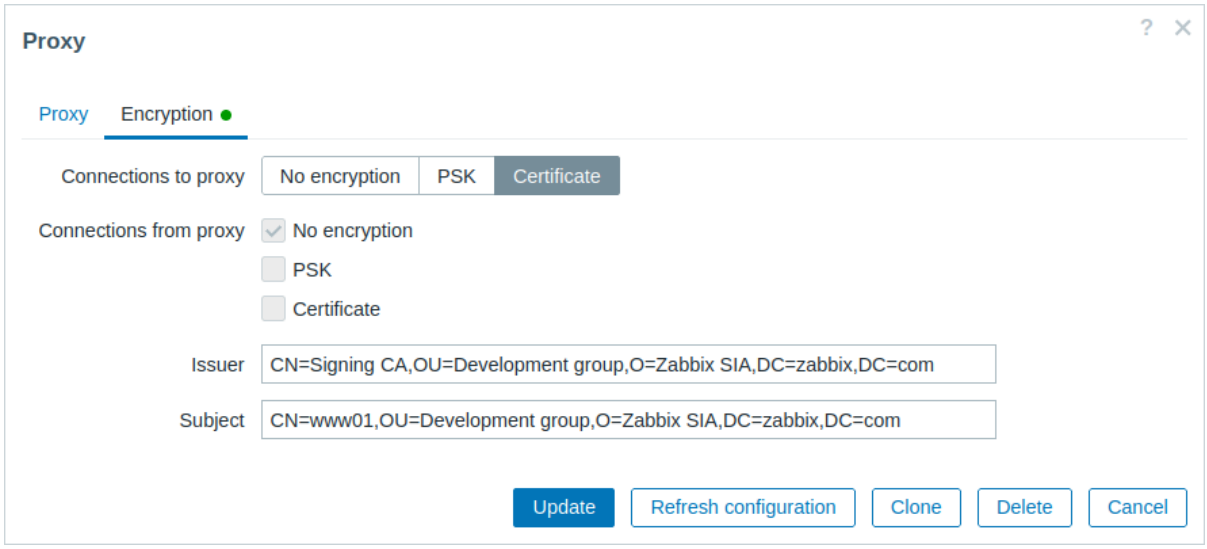
- Przejdź do: *Administracja* → *Proxy*.
- Wybierz proxy i kliknij kartę *Szyfrowanie*.

W poniższych przykładach pola *Issuer* i *Subject* są wypełnione. Więcej informacji o tym, dlaczego i jak używać tych pól, znajduje się w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

Dla aktywnego proxy:



Dla pasywnego proxy:



Zabbix agent

1. Przygotuj pliki z certyfikatami CA najwyższego poziomu, certyfikatem/łańcuchem certyfikatów Zabbix agent oraz kluczem prywatnym, zgodnie z opisem w sekcji Zabbix server. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym Zabbix agent.

2. Edytuj dodatkowe parametry TLS w pliku konfiguracyjnym Zabbix agent:

- Dla aktywnego agent: `TLSConnect=cert`
- Dla pasywnego agent: `TLSAccept=cert`

Note:
Aby zwiększyć bezpieczeństwo agent, możesz ustawić parametry `TLSServerCertIssuer` i `TLSServerCertSubject`. Więcej informacji znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

Parametry TLS w końcowym pliku konfiguracyjnym agent mogą wyglądać następująco. Zwróć uwagę, że przykład zakłada, iż host jest monitorowany przez proxy, dlatego zostało ono określone jako Subject certyfikatu:

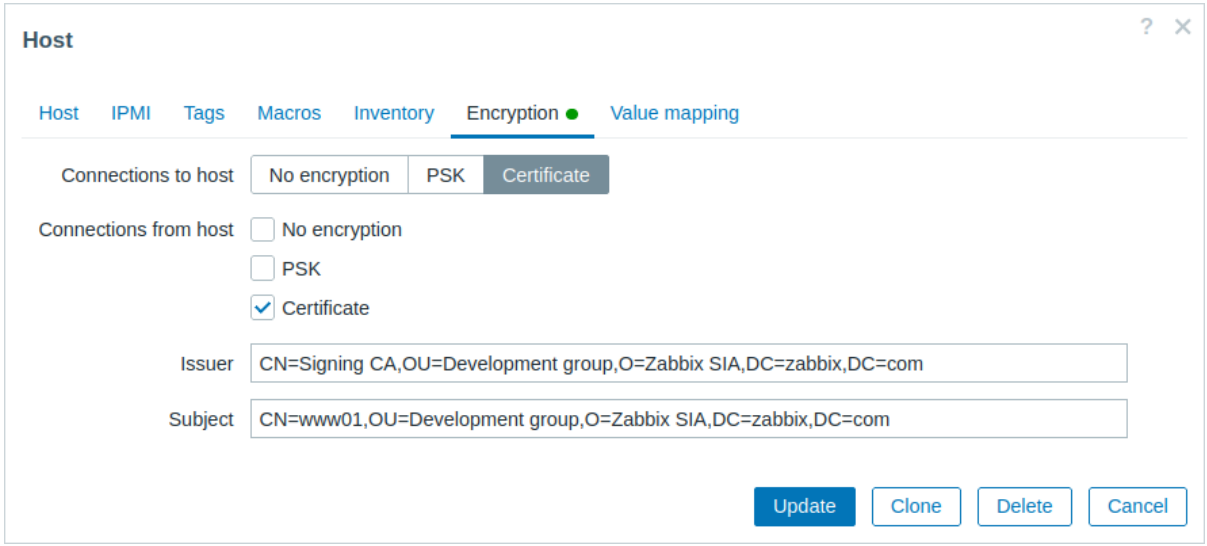
```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
```

```
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_agentd.crt
TLSKeyFile=/home/zabbix/zabbix_agentd.key
```

3. Skonfiguruj szyfrowanie w Zabbix frontend dla hosta monitorowanego przez ten agent.

- Przejdź do: *Data collection* → *Hosts*.
- Wybierz host i kliknij kartę *Encryption*.

W poniższym przykładzie pola *Issuer* i *Subject* są wypełnione. Więcej informacji o tym, dlaczego i jak używać tych pól, znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.



Usługa internetowa Zabbix

1. Przygotuj pliki z certyfikatami głównego urzędu certyfikacji (CA), certyfikatem/łańcuchem certyfikatów usługi internetowej Zabbix oraz kluczem prywatnym, zgodnie z opisem w sekcji serwer Zabbix. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym usługi internetowej Zabbix.

2. Edytuj dodatkowy parametr TLS w pliku konfiguracyjnym usługi internetowej Zabbix: `TLSAccept=cert`

Parametry TLS w końcowym pliku konfiguracyjnym usługi internetowej mogą wyglądać następująco:

```
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_web_service.crt
TLSKeyFile=/home/zabbix/zabbix_web_service.key
```

3. Skonfiguruj serwer Zabbix tak, aby łączył się z usługą internetową Zabbix skonfigurowaną do używania TLS, edytując parametr `WebServiceURL` w pliku konfiguracyjnym serwera Zabbix:

```
WebServiceURL=https://example.com:443/report
```

Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu

Gdy dwa komponenty Zabbix (na przykład serwer i agent) ustanawiają połączenie TLS, weryfikują wzajemnie swoje certyfikaty. Jeśli certyfikat strony zdalnej jest podpisany przez zaufany urząd certyfikacji (z wcześniej skonfigurowanym certyfikatem głównym w `TLSCAFile`), jest prawidłowy, nie wygasł i przechodzi inne kontrole, komunikacja między komponentami może być kontynuowana. W tym najprostszym przypadku wystawca i podmiot certyfikatu nie są weryfikowani.

Jednak stwarza to ryzyko: każdy, kto ma ważny certyfikat, może podszyć się pod kogokolwiek innego (na przykład certyfikat hosta może zostać użyty do podszycia się pod serwer). Choć może to być akceptowalne w małych środowiskach, gdzie certyfikaty są podpisywane przez dedykowany wewnętrzny urząd certyfikacji, a ryzyko podszycia się jest niskie, w większych środowiskach lub środowiskach o wyższych wymaganiach bezpieczeństwa może to być niewystarczające.

Jeśli Twój główny urząd certyfikacji wystawia certyfikaty, które nie powinny być akceptowane przez Zabbix, lub jeśli chcesz zmniejszyć ryzyko podszycia się, możesz ograniczyć dozwolone certyfikaty, określając ich wystawcę i podmiot.

Na przykład w pliku konfiguracyjnym Zabbix proxy można określić:

```
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

Przy tych ustawieniach aktywny proxy nie będzie komunikował się z serwerem Zabbix, którego certyfikat ma innego wystawcę lub podmiot. Podobnie pasywny proxy nie będzie akceptował żądań od takiego serwera.

Zasady dopasowywania ciągów `Issuer` i `Subject`

Zasady dopasowywania ciągów `Issuer` i `Subject` są następujące:

- Ciągi `Issuer` i `Subject` są sprawdzane niezależnie. Oba są opcjonalne.
- Nieokreślony ciąg oznacza, że akceptowany jest dowolny ciąg.
- Ciągi są porównywane *tak jak są* i muszą być identyczne.
- Znaki UTF-8 są obsługiwane. Jednak symbole wieloznaczne (*) ani wyrażenia regularne nie są obsługiwane.
- Zaimplementowano następujące wymagania RFC 4514 — znaki wymagające poprzedzenia znakiem ucieczki (odwrotnym ukośnikiem '\', U+005C):
  - w dowolnym miejscu ciągu: '"' (U+0022), '+' (U+002B), ',' (U+002C), ';' (U+003B), '<' (U+003C), '>' (U+003E), '\\' (U+005C);
  - na początku ciągu: spacja (' ', U+0020) lub znak kratki ('#', U+0023);
  - na końcu ciągu: spacja (' ', U+0020).
- Znaki null (U+0000) nie są obsługiwane. Jeśli zostanie napotkany znak null, dopasowanie zakończy się niepowodzeniem.
- Standardy RFC 4517 i RFC 4518 nie są obsługiwane.

Na przykład, jeśli ciągi organizacji (O) w `Issuer` i `Subject` zawierają końcowe spacje, a ciąg jednostki organizacyjnej (OU) w `Subject` zawiera podwójne cudzysłowy, znaki te muszą być poprzedzone znakiem ucieczki:

```
TLSServerCertIssuer=CN=Signing CA,OU=Development head,O=\ Example SIA\ ,DC=example,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group \"5\",O=\ Example SIA\ ,DC=example,DC=com
```

Kolejność pól i formatowanie

Zabbix stosuje się do zaleceń RFC 4514, które określają „odwrotną" kolejność tych pól, zaczynając od pól najniższego poziomu (CN), przechodząc do pól poziomu pośredniego (OU, O) i kończąc na polach najwyższego poziomu (DC).

```
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
```

Natomiast OpenSSL domyślnie wyświetla ciągi `Issuer` i `Subject` w kolejności od najwyższego do najniższego poziomu. W poniższym przykładzie pola `Issuer` i `Subject` zaczynają się od najwyższego poziomu (DC) i kończą na polu najniższego poziomu (CN). Formatowanie ze spacjami i separatorami pól również różni się w zależności od użytych opcji, dlatego nie będzie zgodne z formatem wymaganym przez Zabbix.

```
$ openssl x509 -noout -in /home/zabbix/zabbix_proxy.crt -issuer -subject
issuer= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Development group/CN=Signing CA
subject= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Development group/CN=Zabbix proxy

$ openssl x509 -noout -text -in /home/zabbix/zabbix_proxy.crt
Certificate:
    ...
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Signing CA
        ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Zabbix proxy
```

Aby poprawnie sformatować ciągi *Issuer* i *Subject* dla Zabbix, uruchom OpenSSL z następującymi opcjami:

```
$ 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
```

Dane wyjściowe będą wtedy w odwrotnej kolejności, rozdzielone przecinkami i gotowe do użycia w plikach konfiguracyjnych Zabbix oraz we frontend:

```
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
```

Ograniczenia dotyczące używania rozszerzeń certyfikatów X.509 v3

Podczas wdrażania certyfikatów X.509 v3 w Zabbix niektóre rozszerzenia mogą nie być w pełni obsługiwane lub mogą powodować niespójne działanie.

Rozszerzenie Subject Alternative Name

Zabbix nie obsługuje rozszerzenia *Subject Alternative Name*, które służy do określania alternatywnych nazw DNS, takich jak adresy IP lub adresy e-mail. Zabbix może weryfikować wyłącznie wartość w polu *Subject* certyfikatu (zobacz Ograniczanie dozwolonego

wystawcy i podmiotu certyfikatu). Jeśli certyfikaty zawierają pole `subjectAltName`, wynik weryfikacji certyfikatu może się różnić w zależności od konkretnych zestawów narzędzi kryptograficznych użytych do kompilacji komponentów Zabbix. W rezultacie Zabbix może zarówno akceptować, jak i odrzucać certyfikaty w zależności od tych kombinacji.

Rozszerzenie Extended Key Usage

Zabbix obsługuje rozszerzenie *Extended Key Usage*. Jednak jeśli jest ono używane, zazwyczaj wymagane jest określenie zarówno atrybutu *clientAuth* (do uwierzytelniania klienta TLS WWW), jak i *serverAuth* (do uwierzytelniania serwera TLS WWW). Na przykład:

- W przypadku kontroli pasywnych, gdzie agent działa jako serwer TLS, atrybut *serverAuth* musi być zawarty w certyfikacie agenta.
- W przypadku kontroli aktywnych, gdzie agent działa jako klient TLS, atrybut *clientAuth* musi być zawarty w certyfikacie agenta.

Chociaż GnuTLS może zgłaszać ostrzeżenie o naruszeniu użycia klucza, zwykle pozwala kontynuować komunikację mimo tych ostrzeżeń.

Rozszerzenie Name Constraints

Obsługa rozszerzenia *Name Constraints* różni się w zależności od zestawu narzędzi kryptograficznych. Upewnij się, że wybrany zestaw narzędzi obsługuje to rozszerzenie. To rozszerzenie może uniemożliwić Zabbix załadowanie certyfikatów CA, jeśli ta sekcja jest oznaczona jako krytyczna, w zależności od używanego zestawu narzędzi.

Listy unieważnionych certyfikatów (CRL)

Jeśli certyfikat zostanie skompromitowany, urząd certyfikacji (CA) może go unieważnić, umieszczając certyfikat na liście unieważnionych certyfikatów (CRL). Listy CRL są zarządzane za pomocą plików konfiguracyjnych i mogą być określone przy użyciu parametru `TLSCRLFile` w plikach konfiguracyjnych serwera, proxy i agenta. Na przykład:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file.crt
```

W tym przypadku `zabbix_crl_file.crt` może zawierać listy CRL z wielu urzędów certyfikacji i może wyglądać następująco:

```
-----BEGIN X509 CRL-----
MIIB/DCB5QIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkARkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7SoOZcAoHQ3GWd9npuctg=
-----END X509 CRL-----
-----BEGIN X509 CRL-----
MIIB+TCB4gIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQBGRYDY29t
...
CAEebS2CND3ShBedZ8YSil59O6JvaDP61lR5lNs=
-----END X509 CRL-----
```

Plik CRL jest wczytywany tylko podczas uruchamiania Zabbix. Aby zaktualizować listę CRL, uruchom ponownie Zabbix.

> **Attention:**
> Jeśli komponenty Zabbix są skompilowane z OpenSSL i używane są listy CRL, upewnij się, że każdy główny i pośredni urząd certyfikacji w łańcuchach certyfikatów ma odpowiadającą mu listę CRL (nawet jeśli jest pusta) dołączoną w `TLSCRLFile`.

2 Używanie kluczy współdzielonych z wyprzedzeniem

Przegląd

Każdy klucz pre-shared key (PSK) w Zabbix jest w rzeczywistości parą składającą się z:

- jawnego ciągu identyfikatora PSK,
- tajnej wartości ciągu PSK.

Ciąg identyfikatora PSK jest niepustym ciągiem UTF-8. Na przykład „PSK ID 001 Zabbix agentd". Jest to unikalna nazwa, za pomocą której ten konkretny PSK jest identyfikowany przez komponenty Zabbix. Nie umieszczaj poufnych informacji w ciągu identyfikatora PSK — jest on przesyłany przez sieć bez szyfrowania.

Wartość PSK to trudny do odgadnięcia ciąg cyfr szesnastkowych, na przykład „e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d

Ograniczenia rozmiaru

W Zabbix obowiązują ograniczenia rozmiaru dla tożsamości i wartości PSK; w niektórych przypadkach biblioteka kryptograficzna może mieć niższy limit:

Komponent	Maks. rozmiar tożsamości PSK	Min. rozmiar wartości PSK	Maks. rozmiar wartości PSK
Zabbix	128 znaków UTF-8	128-bitów (16-bajtowy PSK, wprowadzany jako 32 cyfry szesnastkowe)	2048-bitów (256-bajtowy PSK, wprowadzany jako 512 cyfr szesnastkowych)
GnuTLS	128 bajtów (może zawierać znaki UTF-8)	-	2048-bitów (256-bajtowy PSK, wprowadzany jako 512 cyfr szesnastkowych)
OpenSSL 1.0.x, 1.1.0	127 bajtów (może zawierać znaki UTF-8)	-	2048-bitów (256-bajtowy PSK, wprowadzany jako 512 cyfr szesnastkowych)
OpenSSL 1.1.1	127 bajtów (może zawierać znaki UTF-8)	-	512-bitów (64-bajtowy PSK, wprowadzany jako 128 cyfr szesnastkowych)
OpenSSL 1.1.1a i nowsze	127 bajtów (może zawierać znaki UTF-8)	-	2048-bitów (256-bajtowy PSK, wprowadzany jako 512 cyfr szesnastkowych)

Attention:
Zabbix frontend umożliwia skonfigurowanie ciągu tożsamości PSK o długości do 128 znaków oraz PSK o długości 2048 bitów niezależnie od używanych bibliotek kryptograficznych.
 Jeśli niektóre komponenty Zabbix obsługują niższe limity, obowiązkiem użytkownika jest skonfigurowanie tożsamości i wartości PSK z długością dozwoloną dla tych komponentów.
 Przekroczenie limitów długości skutkuje błędami komunikacji między komponentami Zabbix.

Zanim serwer Zabbix połączy się z agent przy użyciu PSK, serwer wyszukuje w bazie danych tożsamość PSK i wartość PSK skonfigurowane dla tego agent (w praktyce w pamięci podręcznej konfiguracji). Po odebraniu połączenia agent używa tożsamości PSK i wartości PSK ze swojego pliku konfiguracyjnego. Jeśli obie strony mają ten sam ciąg tożsamości PSK i tę samą wartość PSK, połączenie może się powieść.

Attention:
Każda tożsamość PSK musi być powiązana tylko z jedną wartością. Obowiązkiem użytkownika jest dopilnowanie, aby nie było dwóch PSK z tym samym ciągiem tożsamości, ale różnymi wartościami. Niedopełnienie tego może prowadzić do nieprzewidywalnych błędów lub zakłóceń komunikacji między komponentami Zabbix używającymi PSK z tym ciągiem tożsamości PSK.

Generowanie PSK

Na przykład 256-bitowy (32 bajty) PSK można wygenerować za pomocą następujących poleceń:

- przy użyciu *OpenSSL*:

```
$ openssl rand -hex 32
af8ced32dfe8714e548694e2d29e1a14ba6fa13f216cb35c19d0feb1084b0429
```

- przy użyciu *GnuTLS*:

```
$ psktool -u psk_identity -p database.psk -s 32
Generating a random key for user 'psk_identity'
Key stored to database.psk

$ cat database.psk
psk_identity:9b8eafedfaae00cece62e85d5f4792c7d9c9bcc851b23216a1d300311cc4f7cb
```

Należy pamiętać, że powyższe polecenie „psktool” generuje plik bazy danych zawierający tożsamość PSK oraz skojarzony z nią PSK. Zabbix oczekuje w pliku PSK wyłącznie samego PSK, dlatego z pliku należy usunąć ciąg tożsamości oraz dwukropek (':').

Konfigurowanie PSK do komunikacji serwer-agent (przykład)

Na hoście agenta zapisz wartość PSK do pliku, na przykład `/home/zabbix/zabbix_agentd.psk`. Plik musi zawierać PSK w pierwszym ciągu tekstowym, na przykład:

```
1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952
```

Ustaw prawa dostępu do pliku PSK — musi on być czytelny tylko dla użytkownika Zabbix.

Edytuj parametry TLS w pliku konfiguracyjnym agenta `zabbix_agentd.conf`, na przykład ustaw:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

agent będzie łączył się z serwerem (aktywne testy) i akceptował od serwera oraz `zabbix_get` tylko połączenia używające PSK. Tożsamość PSK będzie miała wartość „PSK 001”.

Uruchom ponownie agent. Teraz możesz przetestować połączenie za pomocą `zabbix_get`, na przykład:

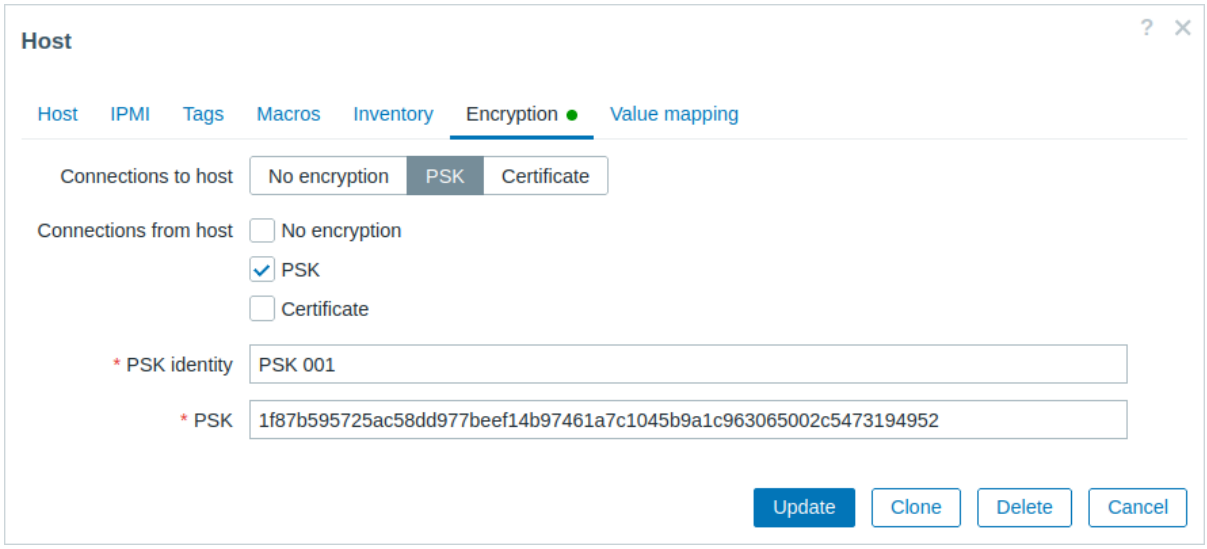
```
zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk --tls-psk-identity="PSK 001" --tl
```

(Aby zminimalizować przestój, zobacz, jak zmienić typ połączenia w zarządzaniu szyfrowaniem połączeń).

Skonfiguruj szyfrowanie PSK dla tego agenta w frontendzie Zabbix:

- Przejdź do: *Data collection → Hosts*
- Wybierz host i kliknij kartę **Encryption**

Przykład:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Gdy pamięć podręczna konfiguracji zostanie zsynchronizowana z bazą danych, nowe połączenia będą używać PSK. Sprawdź pliki dziennika serwera i agenta pod kątem komunikatów o błędach.

Konfigurowanie PSK dla komunikacji serwer - aktywny proxy (przykład)

Na proxy zapisz wartość PSK do pliku, na przykład `/home/zabbix/zabbix_proxy.psk`. Plik musi zawierać PSK w pierwszym wierszu tekstu, na przykład:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Ustaw prawa dostępu do pliku PSK - musi on być dostępny do odczytu tylko dla użytkownika Zabbix.

Edytuj parametry TLS w pliku konfiguracyjnym proxy `zabbix_proxy.conf`, na przykład ustaw:

```
TLSConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

proxy będzie łączył się z serwerem przy użyciu PSK. Tożsamość PSK będzie miała wartość „PSK 002”.

(Aby zminimalizować przestój, zobacz jak zmienić typ połączenia w Zarządzaniu szyfrowaniem połączeń).

Skonfiguruj PSK dla tego proxy w Zabbix frontend. Przejdź do *Administration→Proxies*, wybierz proxy i przejdź do zakładki „Encryption”. W sekcji „Connections from proxy” zaznacz PSK. Wklej do pola „PSK identity” wartość „PSK 002”, a do pola „PSK” wartość „e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9”. Kliknij „Update”.

Uruchom ponownie proxy. Zacznie on używać połączeń szyfrowanych opartych na PSK do serwera. Sprawdź pliki dziennika serwera i proxy pod kątem komunikatów o błędach.

W przypadku pasywnego proxy procedura jest bardzo podobna. Jedyna różnica - ustaw `TLSAccept=psk` w pliku konfiguracyjnym proxy i ustaw „Connections to proxy” w Zabbix frontend na PSK.

3 Rozwiązywanie problemów

Ogólne zalecenia

- Zacznij od ustalenia, który komponent działa jako klient TLS, a który jako serwer TLS w danym przypadku problemowym.
 Serwer Zabbix, proxy i agenty, w zależności od interakcji między nimi, mogą działać zarówno jako serwery TLS, jak i klienci TLS.
 Na przykład serwer Zabbix łączący się z agent w celu wykonania sprawdzenia pasywnego działa jako klient TLS. Agent pełni rolę serwera TLS.
 Agent Zabbix, żądający listy aktywnych sprawdzeń od proxy, działa jako klient TLS. Proxy pełni rolę serwera TLS.
 Narzędzia `zabbix_get` i `zabbix_sender` zawsze działają jako klienci TLS.
- Zabbix używa wzajemnego uwierzytelniania.
 Każda strona weryfikuje swojego partnera i może odrzucić połączenie.
 Na przykład serwer Zabbix łączący się z agent może natychmiast zamknąć połączenie, jeśli certyfikat agent jest nieprawidłowy. I odwrotnie — agent Zabbix akceptujący połączenie od serwera może zamknąć połączenie, jeśli serwer nie jest zaufany przez agent.
- Sprawdź pliki dziennika po obu stronach — po stronie klienta TLS i serwera TLS.
 Strona, która odrzuca połączenie, może zapisać dokładny powód jego odrzucenia. Druga strona często zgłasza raczej ogólny błąd (np. „Połączenie zamknięte przez partnera”, „połączenie zostało nieprawidłowo zakończone”).
- Czasami błędnie skonfigurowane szyfrowanie skutkuje mylącymi komunikatami o błędach, które w żaden sposób nie wskazują rzeczywistej przyczyny.
 W poniższych podsekcjach staramy się przedstawić (daleki od wyczerpującego) zbiór komunikatów i możliwych przyczyn, które mogą pomóc w rozwiązywaniu problemów.
 Należy pamiętać, że różne biblioteki kryptograficzne (OpenSSL, GnuTLS) często generują różne komunikaty o błędach w tych samych sytuacjach problemowych.
 Czasami komunikaty o błędach zależą nawet od konkretnej kombinacji bibliotek kryptograficznych po obu stronach.

1 Problemy z typem połączenia lub uprawnieniami

Serwer jest skonfigurowany do łączenia się z agent przy użyciu PSK, ale agent akceptuje tylko połączenia nieszyfrowane

W logu serwera lub proxy (z *GnuTLS* 3.3.16)

```
Get value from agent failed: zbx_tls_connect(): gnutls_handshake() failed: \
    -110 The TLS connection was non-properly terminated.
```

W logu serwera lub proxy (z *OpenSSL* 1.0.2c)

```
Get value from agent failed: TCP connection successful, cannot establish TLS to [[127.0.0.1]:10050]: \
    Connection closed by peer. Check allowed connection types and access rights
```

Jedna strona łączy się przy użyciu certyfikatu, ale druga strona akceptuje tylko PSK lub odwrotnie

W dowolnym logu (z *GnuTLS*):

```
nie udało się zaakceptować połączenia przychodzącego: z 127.0.0.1: zbx_tls_accept(): gnutls_handshake() ni
    -21 Nie można było uzgodnić obsługiwanego zestawu szyfrów.
```

W dowolnym logu (z *OpenSSL* 1.0.2c):

```
nie udało się zaakceptować połączenia przychodzącego: z 127.0.0.1: uzgadnianie TLS zwróciło kod błędu 1:\
    plik .\ssl\s3_srvr.c linia 1411: error:1408A0C1:procedury SSL:ssl3_get_client_hello:brak wspólnego szy
    zapis TLS krytycznego alertu „handshake failure”
```

Próba użycia Zabbix sender skompilowanego z obsługą TLS do wysyłania danych do serwera/proxy Zabbix skompilowanego bez obsługi TLS

W logu po stronie nawiązującej połączenie:

Linux:

```
...In zbx_tls_init_child()
...Biblioteka OpenSSL (wersja OpenSSL 1.1.1  11 Sep 2018) zainicjalizowana
...
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...End of zbx_tls_connect():FAIL error:'połączenie zamknięte przez peer'
...błąd wysyłania wartości: TCP zakończone powodzeniem, nie można ustanowić TLS do [[localhost]:10051]: po
```

Windows:

```
...Biblioteka OpenSSL (wersja OpenSSL 1.1.1a  20 Nov 2018) zainicjalizowana
...
```

```
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...zbx_psk_client_cb() requested PSK identity "PSK test sender"
...End of zbx_tls_connect():FAIL error:'SSL_connect() błąd wejścia/wyjścia: [0x00000000] Operacja zakończo
...błąd wysyłania wartości: TCP zakończone powodzeniem, nie można ustanowić TLS do [[192.168.1.2]:10051]:
```

W logu po stronie akceptującej:

```
...nie udało się zaakceptować połączenia przychodzącego: z 127.0.0.1: obsługa TLS nie została skompilowana
```

Jedna strona łączy się przy użyciu PSK, ale druga strona używa LibreSSL lub została skompilowana bez obsługi szyfrowania

LibreSSL nie obsługuje PSK.

W logu strony nawiązującej połączenie:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() I/O error: [0] Success
```

W logu strony akceptującej połączenie:

```
...failed to accept an incoming connection: from 192.168.1.2: support for PSK was not compiled in
```

W Zabbix frontend:

```
Get value from agent failed: TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect()
```

Jedna strona łączy się z użyciem PSK, ale druga strona używa OpenSSL z wyłączoną obsługą PSK

W logu strony nawiązującej połączenie:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() set result code to SSL_ERR
```

W logu strony akceptującej połączenie:

```
...failed to accept an incoming connection: from 192.168.1.2: TLS handshake set result code to 1: file ssl
```

2 Problemy z certyfikatem

OpenSSL używany z listami CRL i dla części CA w łańcuchu certyfikatów ich CRL nie jest dołączona w `TLSCRLFile`

W logu serwera TLS w przypadku peera *OpenSSL*:

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake with 127.0.0.1 returned error code
    file s3_srvr.c line 3251: error:14089086: SSL routines:ssl3_get_client_certificate:certificate verify
    TLS write fatal alert "unknown CA"
```

W logu serwera TLS w przypadku peera *GnuTLS*:

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake with 127.0.0.1 returned error code
    file rsa_pk1.c line 103: error:0407006A: rsa routines:RSA_padding_check_PKCS1_type_1:\
    block type is not 01 file rsa_eay.c line 705: error:04067072: rsa routines:RSA_EAY_PUBLIC_DECRYPT:padd
```

Lista CRL wygasła lub wygasa podczas działania serwera

OpenSSL, w logu serwera:

- przed wygaśnięciem:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
    SSL_connect() returned SSL_ERROR_SSL: file s3_clnt.c line 1253: error:14090086:\
    SSL routines:ssl3_get_server_certificate:certificate verify failed:\
    TLS write fatal alert "certificate revoked"
```

- po wygaśnięciu:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
    SSL_connect() returned SSL_ERROR_SSL: file s3_clnt.c line 1253: error:14090086:\
    SSL routines:ssl3_get_server_certificate:certificate verify failed:\
    TLS write fatal alert "certificate expired"
```

Chodzi o to, że przy prawidłowej liście CRL unieważniony certyfikat jest zgłaszany jako "certificate revoked". Gdy lista CRL wygaśnie, komunikat o błędzie zmienia się na "certificate expired", co jest dość mylące.

GnuTLS, w logu serwera:

- przed i po wygaśnięciu tak samo:

```
cannot connect to proxy "proxy-openssl-1.0.1e": TCP successful, cannot establish TLS to [[127.0.0.1]:20004
      invalid peer certificate: The certificate is NOT trusted. The certificate chain is revoked.
```

Certyfikat z podpisem własnym, nieznane CA

OpenSSL, w logu:

```
error:'self signed certificate: SSL_connect() set result code to SSL_ERROR_SSL: file ../ssl/statem/statem_
      line 1924: error:1416F086:SSL routines:tls_process_server_certificate:certificate verify failed:\
      TLS write fatal alert "unknown CA"'
```

Zaobserwowano to, gdy certyfikat serwera przez pomyłkę miał taki sam ciąg znaków Issuer i Subject, mimo że został podpisany przez CA. Issuer i Subject są równe w certyfikacie głównego CA, ale nie mogą być równe w certyfikacie serwera. (To samo dotyczy certyfikatów proxy i agent.)

Aby sprawdzić, czy certyfikat zawiera takie same wpisy Issuer i Subject, uruchom:

```
openssl x509 -in <yourcertificate.crt> -noout -text
```

Dopuszczalne jest, aby certyfikat root (najwyższego poziomu) miał identyczne wartości Issuer i Subject.

3 Problemy z PSK

PSK zawiera nieparzystą liczbę cyfr szesnastkowych

Proxy lub agent nie uruchamia się, komunikat w logu proxy lub agent:

```
nieprawidłowy PSK w pliku "/home/zabbix/zabbix_proxy.psk"
```

Ciąg tożsamości PSK dłuższy niż 128 bajtów jest przekazywany do GnuTLS

W logu po stronie klienta TLS:

```
gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.
```

W logu po stronie serwera TLS.

```
gnutls_handshake() failed: -90 The SRP username supplied is illegal.
```

Zbyt długa wartość PSK używana z OpenSSL 1.1.1

W logu strony nawiązującej połączenie:

```
...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
```

W logu strony akceptującej połączenie:

```
...Message from 123.123.123.123 is missing header. Message ignored.
```

Ten problem zwykle występuje podczas aktualizacji OpenSSL z wersji 1.0.x lub 1.1.0 do 1.1.1, jeśli wartość PSK jest dłuższa niż 512 bitów (64-bajtowy PSK, wprowadzony jako 128 cyfr szesnastkowych).

Zobacz także: Ograniczenia rozmiaru wartości

18 Interfejs webowy

Przegląd Aby zapewnić łatwy dostęp do Zabbix z dowolnego miejsca i z dowolnej platformy, udostępniono interfejs oparty na sieci WWW.

> **Note:**
> Jeśli używasz więcej niż jednej instancji frontend, upewnij się, że ustawienia regionalne i biblioteki (LDAP, SAML itp.) są zainstalowane i skonfigurowane identycznie dla wszystkich frontendów.

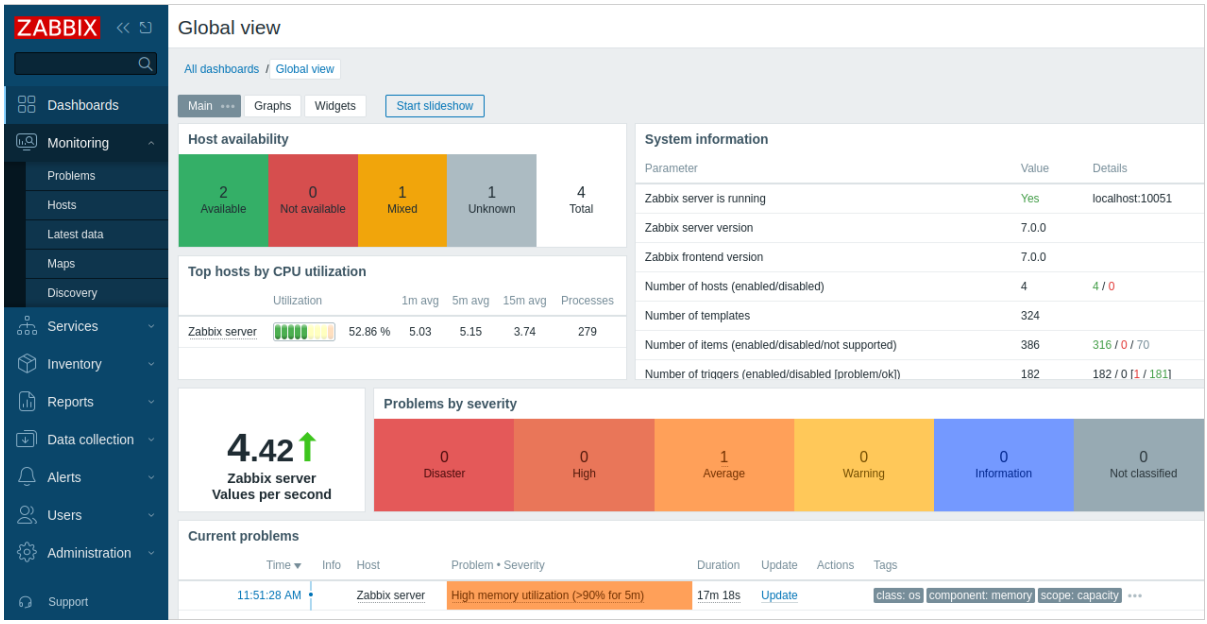
Pomoc frontend Łącze pomocy ? jest dostępne w formularzach frontend Zabbix i zawiera bezpośrednie odnośniki do odpowiednich części dokumentacji.

1 Menu

Przegląd

Pionowe menu na pasku bocznym zapewnia dostęp do różnych sekcji frontendu Zabbix.

Domyślny motyw posiada ciemnoniebieskie menu.

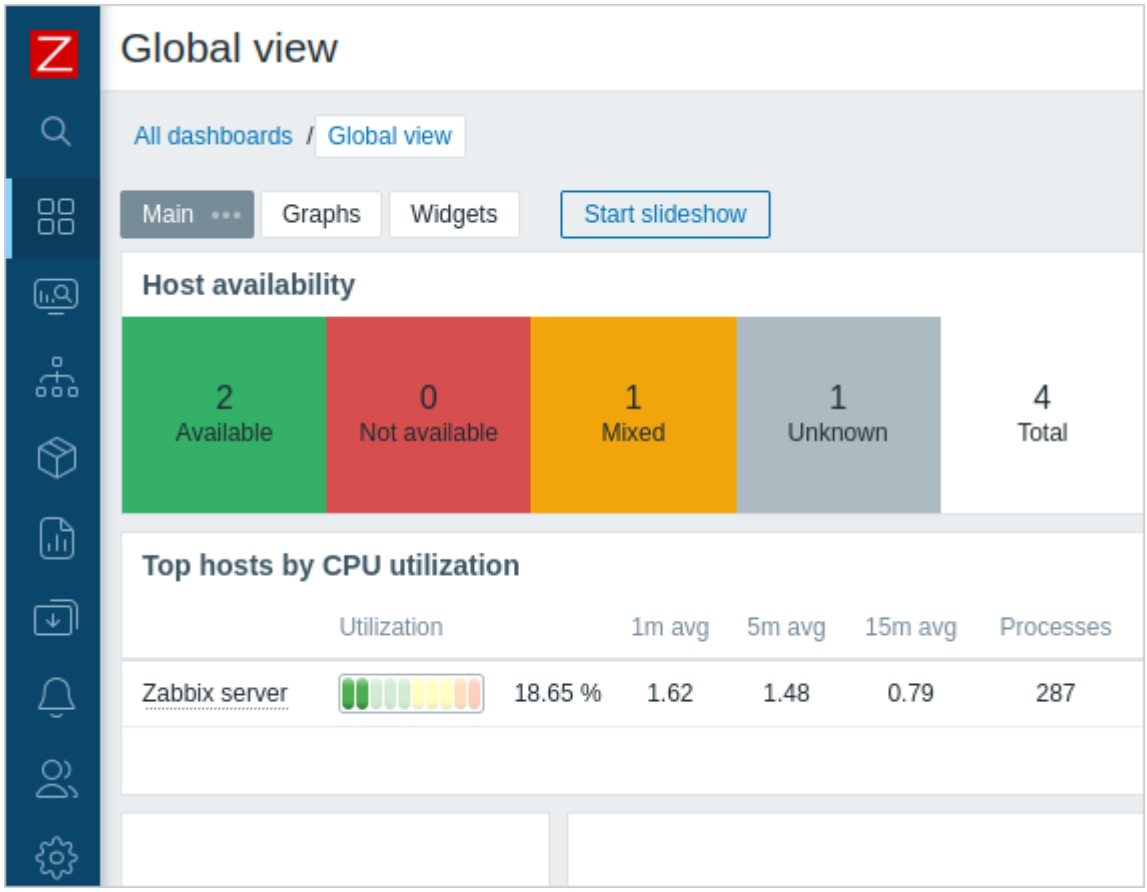


Praca z menu

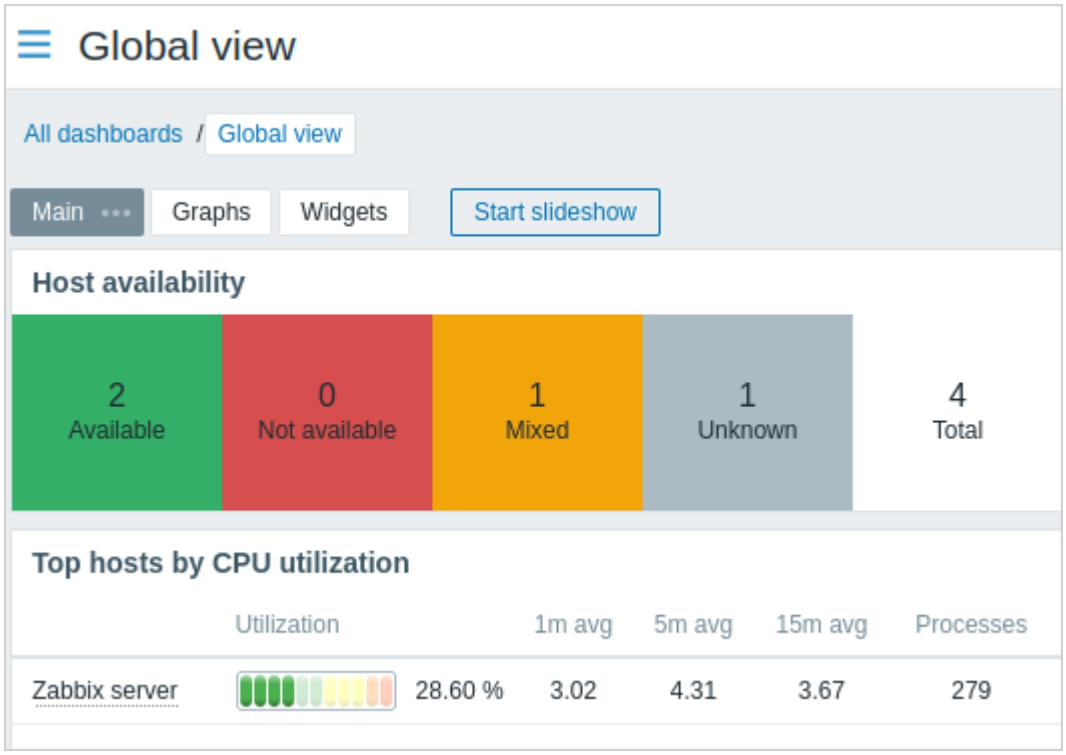
Pole wyszukiwania globalnego znajduje się poniżej logo Zabbix.

Menu można zwinąć lub całkowicie ukryć:

- Aby zwinąć, kliknij  obok logo Zabbix. W zwiniętym menu widoczne są tylko ikony.



- Aby ukryć, kliknij obok logo Zabbix. W ukrytym menu wszystko jest niewidoczne.



Collapsed menu

When the menu is collapsed to icons only, a full menu reappears as soon as the mouse cursor is placed upon it. Note that it reappears over page content; to move page content to the right you have to click on the expand button. If the mouse cursor again is placed outside the full menu, the menu will collapse again after two seconds.

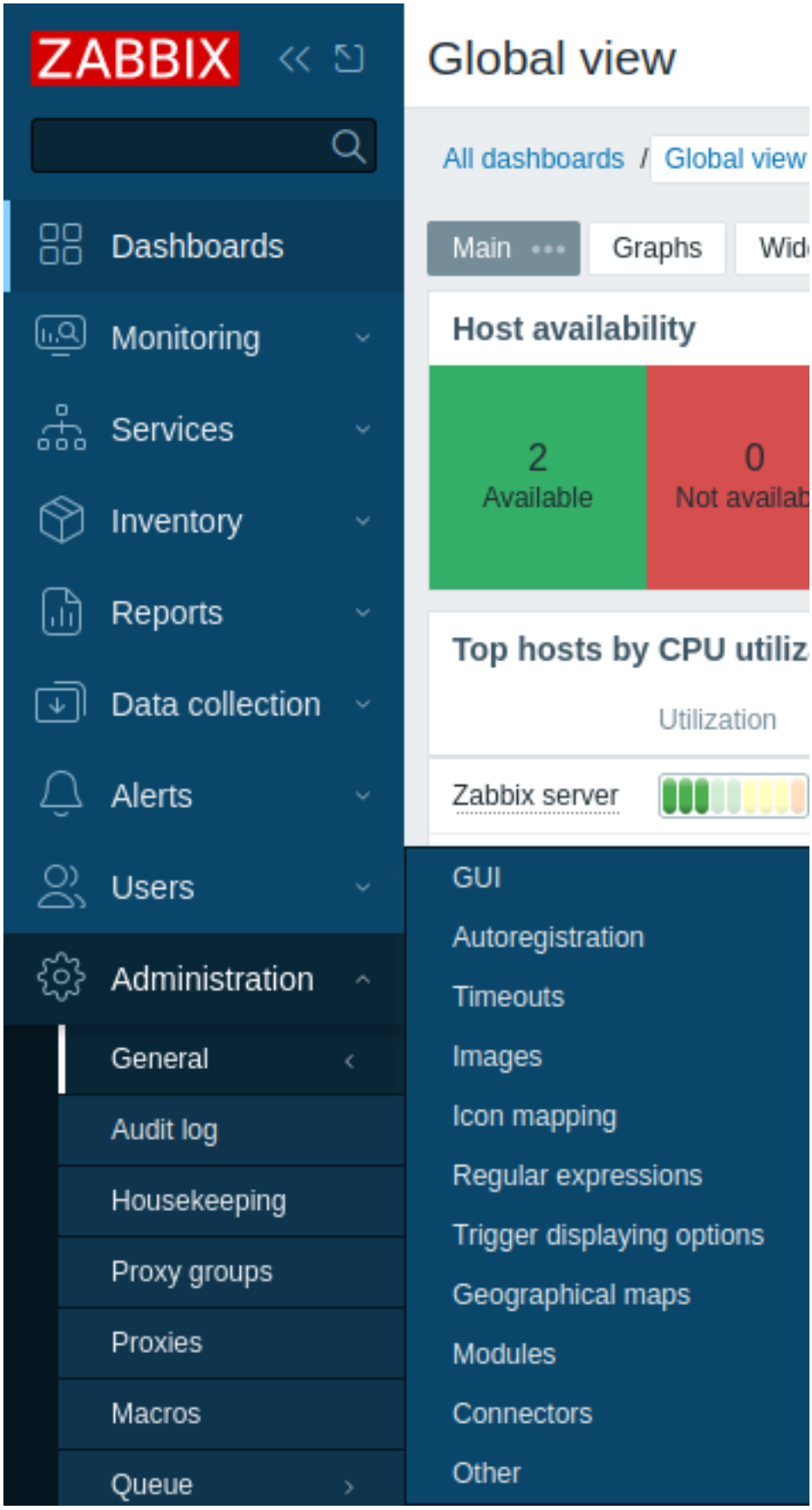
You can also make a collapsed menu reappear fully by hitting the Tab key. Hitting the Tab key repeatedly will allow to focus on the next menu element.

Ukryte menu

Nawet gdy menu jest całkowicie ukryte, pełne menu jest dostępne po jednym kliknięciu myszą — po kliknięciu ikony burgera . Należy pamiętać, że pojawia się ono ponownie nad zawartością strony; aby przesunąć zawartość strony w prawo, trzeba odkryć menu, klikając przycisk pokazania paska bocznego.

Poziomy menu

W menu są maksymalnie trzy poziomy.



Menu kontekstowe

Oprócz menu głównego Zabbix udostępnia menu kontekstowe dla hosta, pozycji oraz zdarzenia, zapewniające szybki dostęp do często używanych elementów, takich jak najnowsze wartości, prosty wykres, formularz konfiguracji, powiązane skrypty lub linki zewnętrzne.

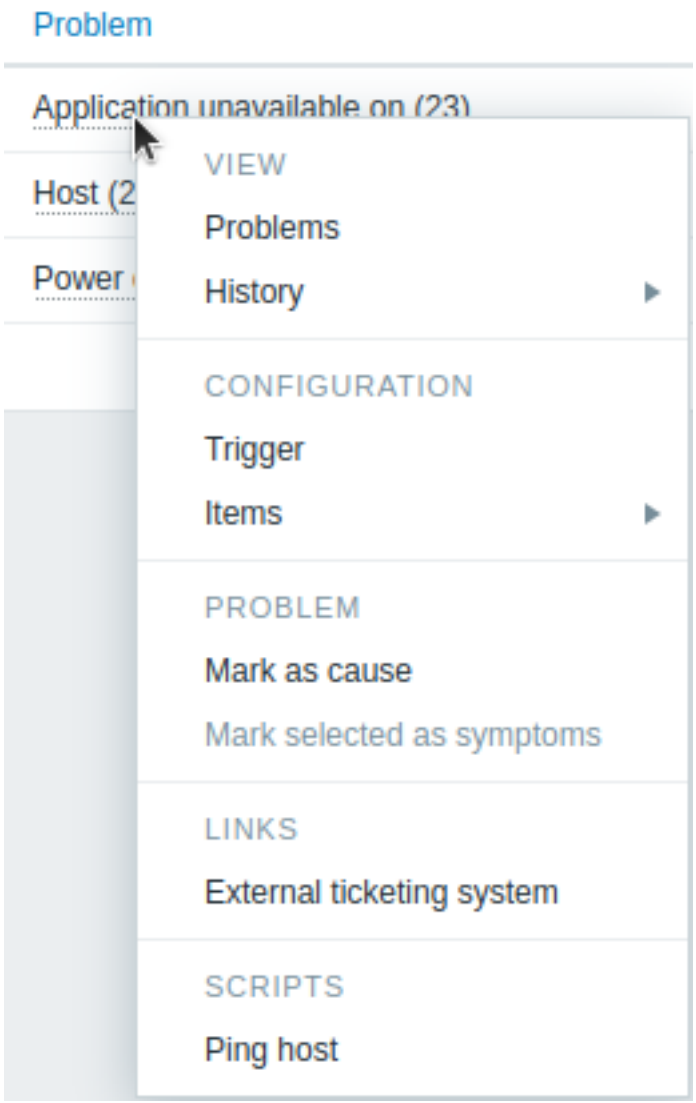
Menu kontekstowe są dostępne po kliknięciu nazwy hosta, pozycji lub problemu/wyzwalacza w obsługiwanych lokalizacjach.

1 Menu Zdarzenia

Przegląd

Menu zdarzenia zawiera skróty do działań lub sekcji frontend, które są często wymagane dla zdarzenia.

Menu zdarzenia można otworzyć, klikając nazwę zdarzenia.



Zawartość

Menu kontekstowe zdarzenia ma sześć sekcji: *Widok*, *Akcje*, *Konfiguracja*, *Problem*, *Linki* i *Skrypty*. Dla encji, które nie są skonfigurowane, linki są wyłączone i wyświetlane na szaro. Sekcje *Skrypty* i *Linki* są wyświetlane, jeśli ich encje są skonfigurowane.

Sekcja *Widok* zawiera linki do:

- **Problemy** - otwiera listę nierozwiązanych problemów powiązanych z bazowym wyzwalacz;
- **Historia** - prowadzi do wykresu/historii pozycja(i) w *Najnowsze dane* dla bazowej pozycji (bazowych pozycji). Jeśli wyzwalacz używa kilku pozycji, linki będą dostępne dla każdej z nich.

Sekcja *Akcje* jest dostępna tylko w widżetach *Przegląd wyzwalaczy*. Zawiera link do:

- **Aktualizuj problem** - otwiera ekran aktualizacji problemu.

Sekcja *Konfiguracja* zawiera linki do konfiguracji:

- **Wyzwalacza**, który wywołał problem;
- **Pozycji** użytych w wyrażeniu wyzwalacza.

> **Note:**
> Należy pamiętać, że sekcja konfiguracji jest dostępna tylko dla użytkowników typu Admin i Super admin.

Sekcja *Problem* zawiera opcje umożliwiające:

- **Oznacz jako przyczynę** - oznaczenie problemu jako przyczyny;
- **Oznacz wybrane jako symptomy** - oznaczenie wybranych problemów jako symptomów tego problemu.

Sekcja *Linki* zawiera linki do:

- dostępu do skonfigurowanego URL wyzwalacza;
- dostępu do niestandardowych linków skonfigurowanych w Skryptach globalnych (z zakresem „Ręczna akcja zdarzenia" i typem „URL");
- dostępu do skonfigurowanego zewnętrznego zgłoszenia dla problemu (zobacz opcję *Include event menu entry* podczas konfigurowania webhooków).

Sekcja *Skrypty* zawiera linki do wykonania globalnego skryptu (z zakresem *Ręczna akcja zdarzenia*). Ta funkcja może być przydatna do uruchamiania skryptów używanych do zarządzania zgłoszeniami problemów w systemach zewnętrznych.

Obsługiwane lokalizacje

Menu kontekstowe zdarzenia jest dostępne po kliknięciu problemu lub nazwy zdarzenia w różnych sekcjach frontend, na przykład:

- Widgety pulpitów, takie jak widget *Problemy*, widget *Przegląd wyzwalaczy* itp.
- Monitorowanie → Problemy
- Monitorowanie → Problemy → Szczegóły zdarzenia
- Raporty → Top 100 wyzwalaczy (skrypty globalne i dostęp do zewnętrznego zgłoszenia nie są obsługiwane w tej lokalizacji)

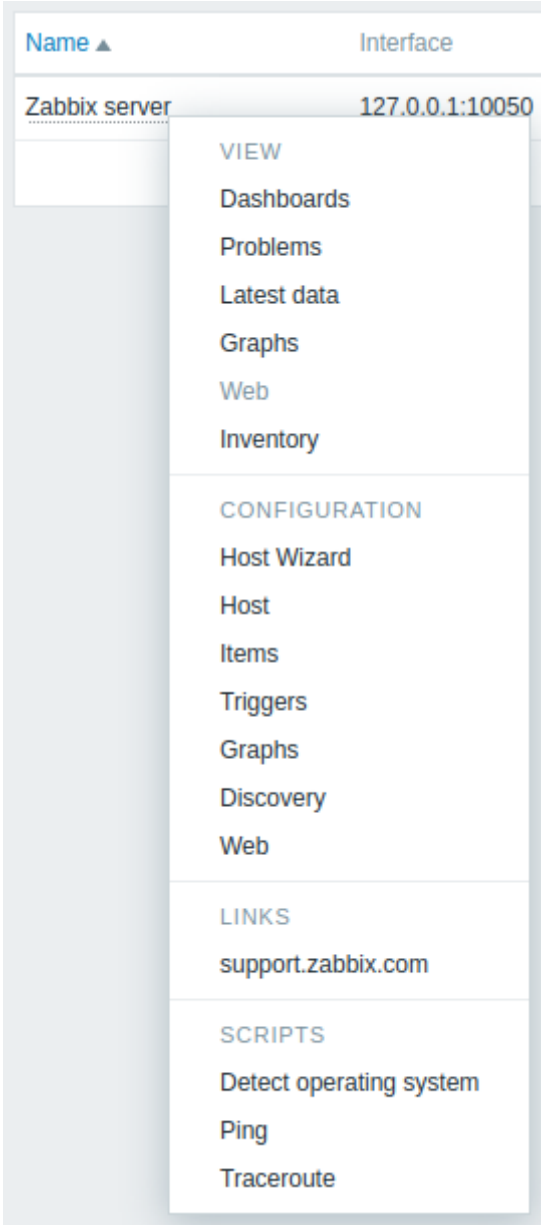
2 Menu hosta

Przegląd

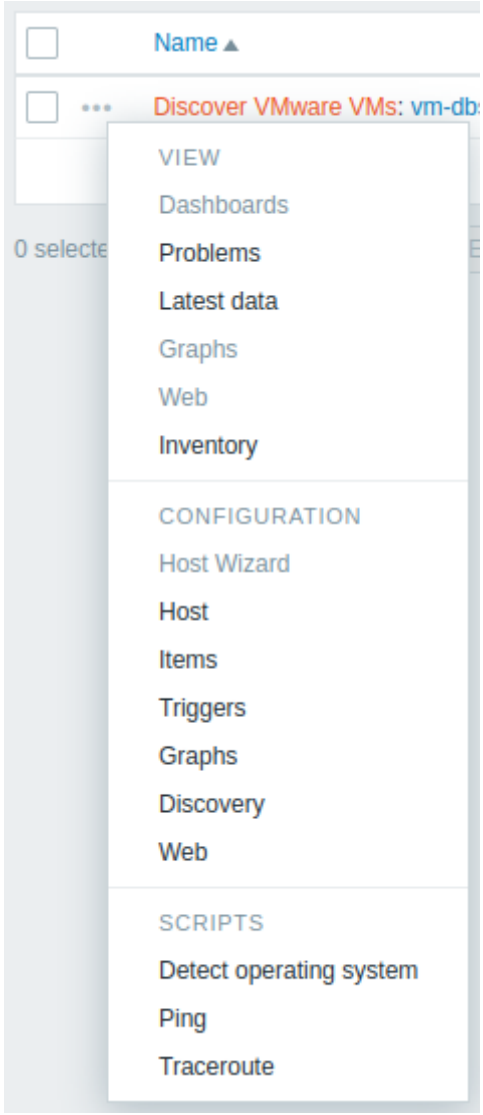
Menu hosta zawiera skróty do działań lub sekcji frontendu, które są często wymagane dla hosta.

Menu hosta można otworzyć, klikając nazwę hosta lub ikonę trzech kropek, w zależności od sekcji frontendu, na przykład:

Monitoring → Najnowsze dane



Zbieranie danych → Hosty



Zawartość

Menu kontekstowe hosta ma cztery sekcje: *View*, *Configuration*, *Links* i *Scripts*. Dla encji, które nie są skonfigurowane, linki są wyłączone i wyświetlane w szarym kolorze. Sekcje *Scripts* i *Links* są wyświetlane, jeśli odpowiadające im encje są skonfigurowane.

Sekcja *View* zawiera linki do:

- **Dashboards** - otwiera widżety i wykresy.
- **Problems** - otwiera sekcję *Problems* z listą nierozwiązanych problemów powiązanego wyzwalacza.
- **Latest data** - otwiera sekcję *Latest data* z listą wszystkich najnowszych danych bieżącego hosta.
- **Graphs** - otwiera proste wykresy bieżącego hosta.
- **Web** - otwiera link do skonfigurowanych scenariuszy web.
- **Inventory** - otwiera link do inwentarza bieżącego hosta.

Sekcja *Configuration* zawiera linki do:

- **Host Wizard** - otwiera Host Wizard dla bieżącego hosta (wyłączone dla wykrytych hostów).
- **Host** - formularz konfiguracji bieżącego hosta.
- **Items** - lista pozycji bieżącego hosta.
- **Triggers** - lista wyzwalaczy bieżącego hosta.
- **Graphs** - proste wykresy bieżącego hosta.
- **Discovery** - lista reguł niskopoziomowego wykrywania bieżącego hosta.
- **Web** - lista scenariuszy web bieżącego hosta.

Note:
Należy pamiętać, że sekcja konfiguracji jest dostępna tylko dla użytkowników Admin i Super admin.

Sekcja *Links* zawiera linki do:

- dostępu do skonfigurowanego trigger URL.
- dostępu do niestandardowych linków skonfigurowanych w Global scripts (z zakresem *Manual host action* i typem 'URL').

Sekcja *Scripts* umożliwia wykonywanie global scripts skonfigurowanych dla bieżącego hosta. Aby były dostępne w menu hosta, skrypty te muszą mieć zdefiniowany zakres jako *Manual host action*.

Obsługiwane lokalizacje

Menu hosta jest dostępne po kliknięciu nazwy hosta w różnych sekcjach frontend, na przykład:

- Widżety pulpitów, takie jak Problemy, Najpopularniejsze pozycje, Przegląd wyzwalaczy itp.
- Monitorowanie → Problemy
- Monitorowanie → Problemy → Szczegóły zdarzenia
- Monitorowanie → Hosty
- Monitorowanie → Hosty → Monitorowanie WWW
- Monitorowanie → Najnowsze dane
- Monitorowanie → Mapy
- Inwentaryzacja → Hosty
- Raporty → 100 najpopularniejszych wyzwalaczy

Menu hosta jest dostępne po kliknięciu ikony trzech kropek w sekcji Zbieranie danych → Hosty.

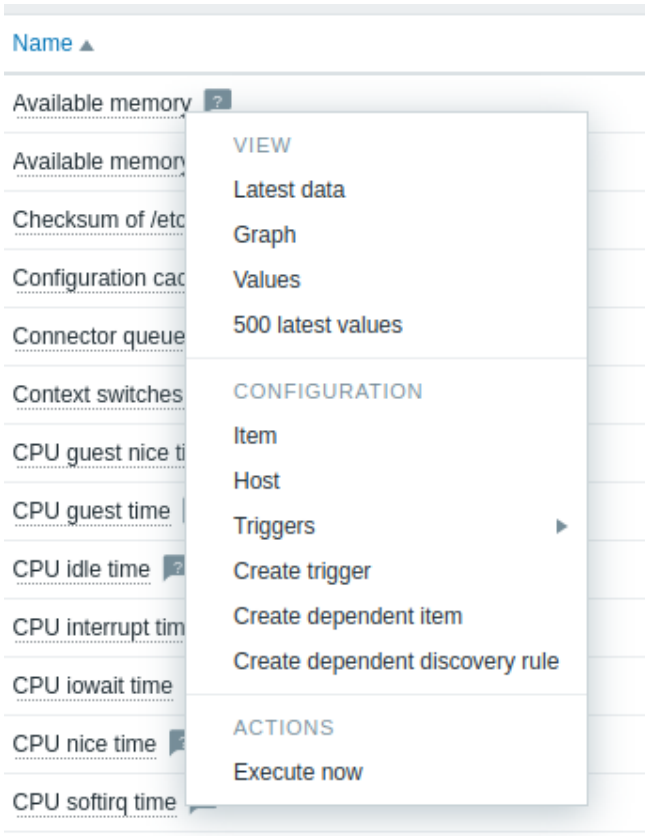
3 Menu pozycji

Przegląd

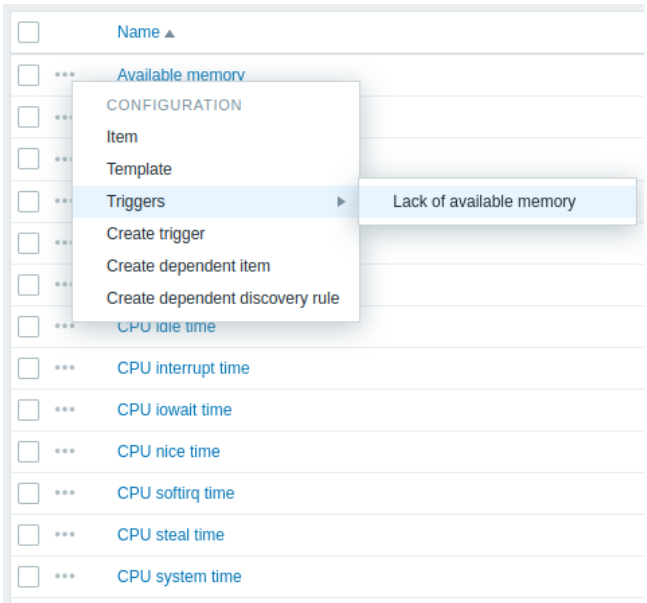
Menu pozycji zawiera skróty do działań lub sekcji frontend, które są często wymagane dla pozycji.

Menu pozycji można otworzyć, klikając nazwę pozycji lub ikonę z trzema kropkami, w zależności od sekcji frontend, na przykład:

Monitoring → Najnowsze dane



Zbieranie danych → Szablony → Pozycje



Zawartość

Menu pozycji ma trzy sekcje: *Widok*, *Konfiguracja* i *Działania*.

Sekcja *Widok* zawiera następujące opcje:

- **Najnowsze dane** - otwiera sekcję *Najnowsze dane* przefiltrowaną według bieżącego hosta i pozycji;
- **Wykres** - otwiera prosty wykres bieżącej pozycji;
- **Wartości** - otwiera listę wszystkich wartości odebranych dla bieżącej pozycji w ciągu ostatnich 60 minut;
- **500 najnowszych wartości** - otwiera listę 500 najnowszych wartości dla bieżącej pozycji.

Sekcja *Konfiguracja* (dostępna tylko dla użytkowników typu *Admin* i *Super admin*) zawiera następujące opcje:

- **Pozycja** - otwiera formularz konfiguracji pozycji bieżącej pozycji;
- **Szablon** - otwiera formularz konfiguracji szablonu szablonu, do którego należy pozycja (dostępne tylko wtedy, gdy menu pozycji jest otwierane z *Zbieranie danych* → *Szablony* → *Pozycje*);
- **Host** - otwiera formularz konfiguracji hosta hosta, do którego należy pozycja;
- **Wyzwalacze** - po najechaniu kursorem otwiera listę wyzwalaczy pozycji, jeśli istnieją; kliknięcie wyzwalacza otwiera jego formularz konfiguracji wyzwalacza;
- **Utwórz wyzwalacz** - otwiera formularz konfiguracji wyzwalacza w celu utworzenia wyzwalacza dla tej pozycji;
- **Utwórz pozycję zależną** - otwiera formularz konfiguracji pozycji w celu utworzenia pozycji zależnej, z bieżącą pozycją jako pozycją nadrzędną;
- **Utwórz zależną regułę wykrywania** - otwiera formularz konfiguracji reguły wykrywania w celu utworzenia zależnej reguły wykrywania, z bieżącą pozycją jako pozycją nadrzędną.

Sekcja *Działania* zawiera następującą opcję:

- **Wykonaj teraz** - natychmiast wykonuje sprawdzenie w celu pobrania nowej wartości pozycji.

Obsługiwane lokalizacje

Menu pozycji jest dostępne po kliknięciu nazwy pozycji w różnych sekcjach frontend, na przykład:

- Monitoring → Najnowsze dane
- Zbieranie danych → Hosts → Pozycje
- Zbieranie danych → Hosts → Reguły wykrywania → Prototypy pozycji

Menu pozycji jest dostępne po kliknięciu wartości pozycji w widżecie pulpitu Top items.

2 Sekcje frontend

Struktura menu Menu frontendu Zabbix ma następującą strukturę:

- Pulpity
- Monitorowanie
  - Problemy
  - Hosty
  - Najnowsze dane
  - Mapy
  - Odkrywanie
- Usługi
  - Usługi
  - SLA
  - Raport SLA
- Inwentaryzacja
  - Przegląd
  - Hosty
- Raporty
  - Informacje o systemie
  - Raporty cykliczne
  - Raport dostępności
  - 100 najważniejszych wyzwalaczy
  - Dziennik audytu
  - Dziennik działań
  - Powiadomienia
- Zbieranie danych
  - Grupy szablonów
  - Grupy hostów

    - Szablony
    - Hosty
    - Konserwacja
    - Korelacja zdarzeń
    - Odkrywanie
- Alerty
    - Działania
        * Działania wyzwalaczy
        * Działania usług
        * Działania odkrywania
        * Działania autorejestracji
        * Działania wewnętrzne
    - Typy mediów
    - Skrypty
- Użytkownicy
    - Grupy użytkowników
    - Role użytkowników
    - Użytkownicy
    - Tokeny API
    - Uwierzytelnianie
- Administracja
    - Ogólne
        * GUI
        * Autorejestracja
        * Limity czasu
        * Obrazy
        * Mapowanie ikon
        * Wyrażenia regularne
        * Opcje wyświetlania wyzwalaczy
        * Mapy geograficzne
        * Moduły
        * Konektory
        * Inne
    - Dziennik audytu
    - Porządkowanie
    - Grupy proxy
    - Proxy
    - Makra
    - Kolejka
        * Przegląd kolejki
        * Przegląd kolejki według proxy
        * Szczegóły kolejki

1 Pulpity

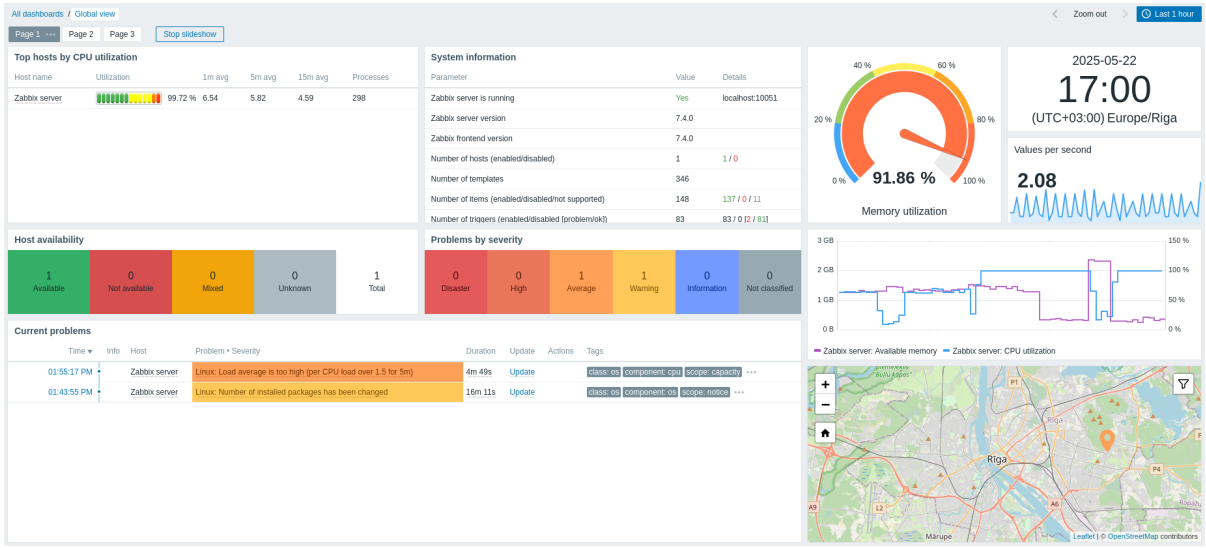
Przegląd

Sekcja *Dashboards* została zaprojektowana do wyświetlania podsumowań wszystkich ważnych informacji w **dashboardzie**.

Chociaż jednocześnie może być wyświetlany tylko jeden dashboard, można skonfigurować kilka dashboardów. Każdy dashboard może zawierać jedną lub kilka stron, które mogą być rotowane w pokazie slajdów.

Strona dashboardu składa się z widżetów, a każdy widżet jest przeznaczony do wyświetlania informacji określonego rodzaju i z określonego źródła, którym może być podsumowanie, mapa, wykres, zegar itp.

Dostęp do hostów w widżetach zależy od uprawnień hosta.



Strony i widżety są dodawane do dashboardu oraz edytowane w trybie edycji dashboardu. Strony można przeglądać i rotować w trybie wyświetlania dashboardu.

Okres czasu wyświetlany w widżetach wykresów jest kontrolowany przez selektor *Time period* znajdujący się nad widżetami. Etykieta selektora *Time period*, znajdująca się po prawej stronie, wyświetla aktualnie wybrany okres czasu. Kliknięcie etykiety karty umożliwia rozwijanie i zwijanie selektora *Time period*.

Rozmiar dashboardu

Minimalna szerokość dashboardu wynosi 1200 pikseli. Dashboard nie zmniejszy się poniżej tej szerokości; zamiast tego zostanie wyświetlony poziomy pasek przewijania, jeśli okno przeglądarki jest mniejsze.

Maksymalna szerokość dashboardu to szerokość okna przeglądarki. Widgety dashboardu rozciągają się poziomo, aby dopasować się do okna. Jednocześnie widget dashboardu nie może zostać rozciągnięty poziomo poza granice okna.

W poziomie dashboard składa się z 72 kolumn o zawsze równej szerokości, które dynamicznie się rozciągają/kurczą (ale łącznie nie do mniej niż 1200 pikseli).

W pionie dashboard może zawierać maksymalnie 64 wiersze; każdy wiersz ma stałą wysokość 70 pikseli.

Widget może więc mieć maksymalnie 72 kolumny szerokości i 64 wiersze wysokości.

Wyświetlanie dashboardów

Aby wyświetlić wszystkie skonfigurowane dashboardy, kliknij *All dashboards* bezpośrednio pod tytułem sekcji.



Dashboardy są wyświetlane z etykietą udostępniania:

- *My* - oznacza dashboard należący do bieżącego użytkownika
- *Shared* - oznacza publiczny dashboard lub prywatny dashboard udostępniony dowolnemu użytkownikowi albo grupie użytkowników

Należy pamiętać, że etykiety udostępniania są wyświetlane tylko dla dashboardów należących do bieżącego użytkownika. Dashboardy należące do innych użytkowników są wyświetlane bez etykiet udostępniania.

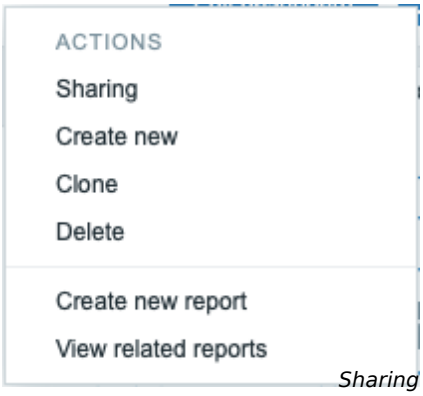
Filtr znajdujący się po prawej stronie nad listą umożliwia filtrowanie dashboardów według nazwy oraz według tych utworzonych przez bieżącego użytkownika.

Aby usunąć jeden lub kilka dashboardów, zaznacz pola wyboru odpowiednich dashboardów i kliknij *Delete* pod listą.

Wyświetlanie dashboardu

Aby wyświetlić pojedynczy dashboard, kliknij jego nazwę na liście dashboardów.

Podczas **wyświetlania** dashboardu dostępne są następujące opcje:

Edit dashboard	Przełącz do trybu **edycji** dashboardu. Tryb edycji jest również otwierany podczas tworzenia nowego dashboardu oraz po kliknięciu przycisku edycji widżetu.
	Otwórz menu akcji (zobacz opisy akcji poniżej).
	*Sharing* - edytuj ustawienia udostępniania dla dashboardu. *Create new* - utwórz nowy dashboard. *Clone* - utwórz nowy dashboard, kopiując właściwości istniejącego. Najpierw zostaniesz poproszony o podanie parametrów dashboardu. Następnie nowy dashboard otworzy się w trybie edycji ze wszystkimi widżetami oryginalnego dashboardu. *Delete* - usuń dashboard. *Create new report* - otwórz okno podręczne z formularzem konfiguracji raportu. Opcja jest wyłączona, jeśli użytkownik nie ma uprawnień do zarządzania raportami cyklicznymi. *View related reports* - otwórz okno podręczne z listą istniejących raportów opartych na bieżącym dashboardzie. Opcja jest wyłączona, jeśli nie ma powiązanych raportów lub użytkownik nie ma uprawnień do wyświetlania raportów cyklicznych.
	Wyświetlaj tylko zawartość strony (tryb kiosk). Do trybu kiosk można też uzyskać dostęp za pomocą następujących parametrów URL: `/zabbix.php?action=dashboard.view&kiosk=1`. Aby wyjść do trybu normalnego: `/zabbix.php?action=dashboard.view&kiosk=0`.

Udostępnianie

Pulpity mogą być publiczne lub prywatne.

Publiczne pulpity są widoczne dla wszystkich użytkowników. Prywatne pulpity są widoczne tylko dla ich właściciela oraz dla użytkowników/grup użytkowników dodanych w preferencjach udostępniania.

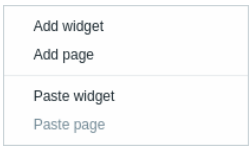
Aby zmienić status udostępniania pulpitu, kliknij opcję *Udostępnianie* w menu akcji podczas wyświetlania pojedynczego pulpitu:

Parameter	Description
Type	Wybierz typ pulpitu: **Private** - pulpit jest widoczny tylko dla wybranych grup użytkowników i użytkowników. **Public** - pulpit jest widoczny dla wszystkich.
List of user group shares	Wybierz grupy użytkowników, dla których pulpit jest dostępny. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.
List of user shares	Wybierz użytkowników, dla których pulpit jest dostępny. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.

Zobacz Uprawnienia do pulpitów, aby dowiedzieć się, w jaki sposób opcje udostępniania ograniczają działania dostępne dla użytkowników.

Edycja pulpitu

Podczas **edycji** pulpitu dostępne są następujące opcje:

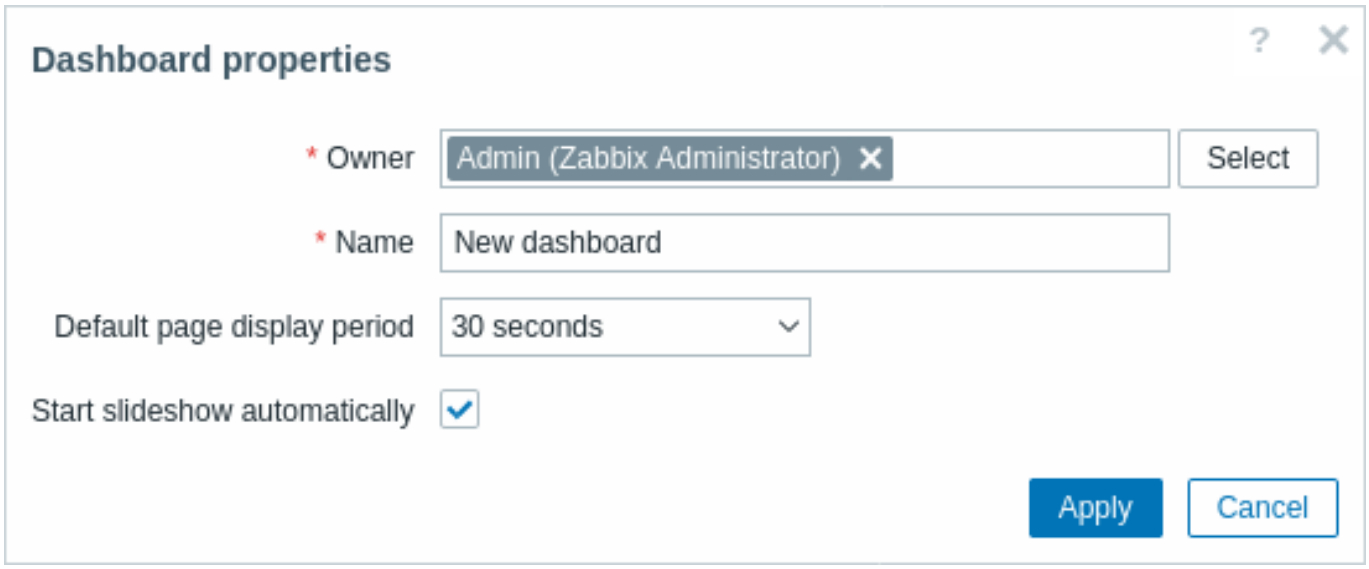
	Edytuj ogólne parametry pulpitu. Dodaj nowy widżet. Kliknięcie przycisku strzałki otworzy menu akcji (zobacz opisy akcji poniżej).
	*Dodaj widżet* - dodaj nowy widżet. *Dodaj stronę* - dodaj nową stronę. *Wklej widżet* - wklej skopiowany widżet. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnego widżetu. Jednocześnie można skopiować tylko jeden element (widżet lub stronę). *Wklej stronę* - wklej skopiowaną stronę. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnej strony.
	Zapisz zmiany pulpitu.
	Anuluj zmiany pulpitu.

Tworzenie dashboardu

Nowy dashboard można utworzyć na dwa sposoby:

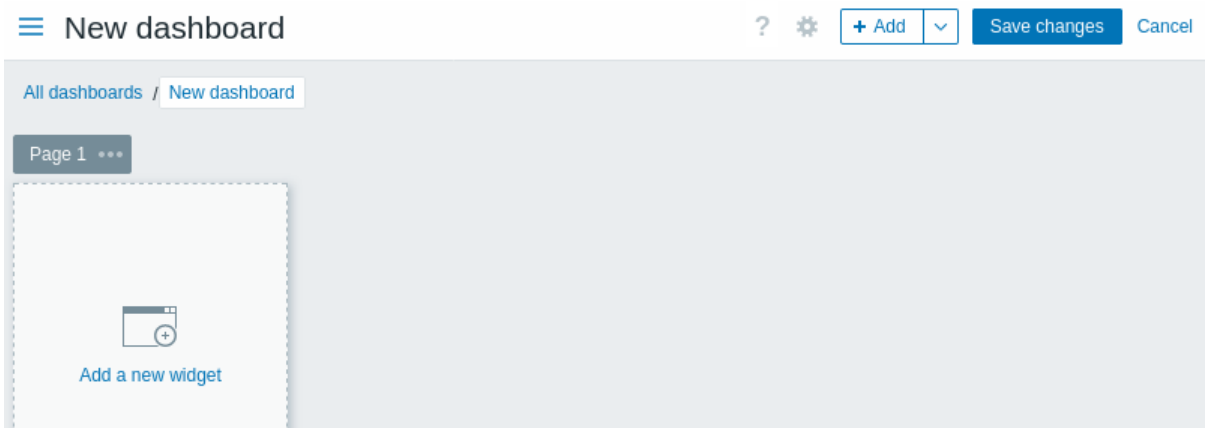
- Kliknij *Create dashboard*, podczas przeglądania wszystkich dashboardów
- Wybierz *Create new* z menu akcji, podczas przeglądania pojedynczego dashboardu

Najpierw zostaniesz poproszony o podanie ogólnych parametrów dashboardu:



Parameter	Description
Owner	Wybierz użytkownika systemowego, który będzie właścicielem dashboardu.
Name	Wprowadź nazwę dashboardu.
Default page display period	Wybierz okres, przez jaki strona dashboardu ma być wyświetlana przed przejściem do następnej strony w slideshow.
Start slideshow automatically	Zaznacz to pole wyboru, aby automatycznie uruchamiać slideshow, gdy istnieje więcej niż jedna strona dashboardu.

Po kliknięciu *Apply* zostanie otwarty pusty dashboard:



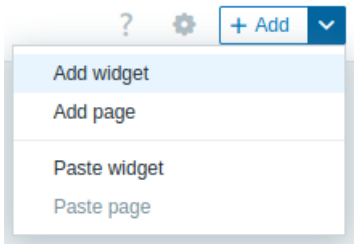
Aby wypełnić dashboard, możesz dodać widżety i strony.

Kliknij przycisk *Save changes*, aby zapisać dashboard. Jeśli klikniesz *Cancel*, dashboard nie zostanie utworzony.

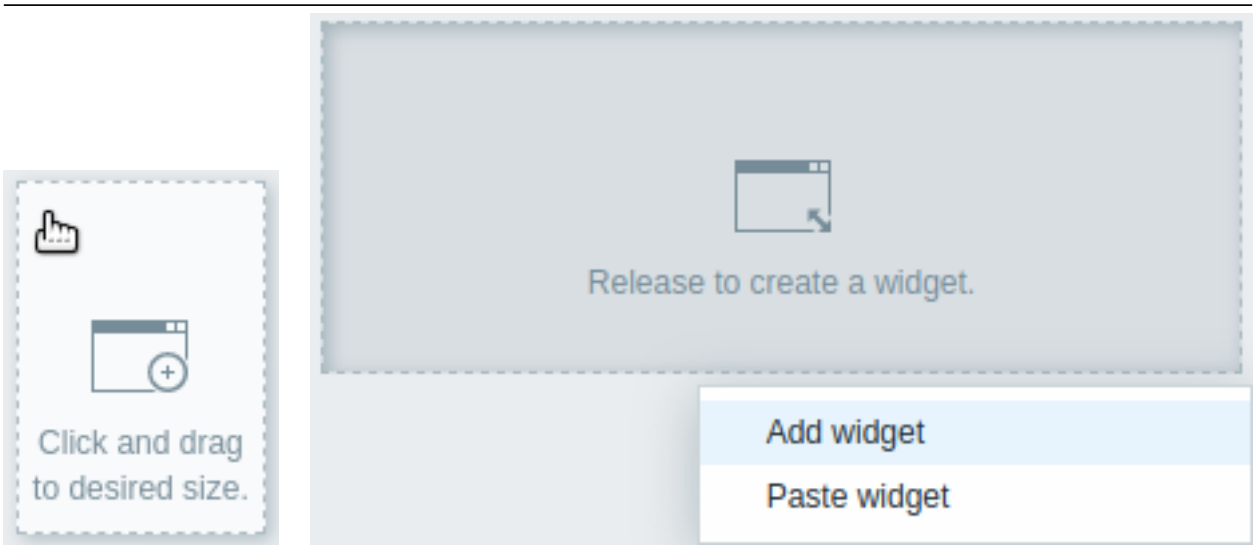
Dodawanie widżetów

Do dashboardu można dodać różne widżety (Action log, Clock, Discovery status itp.). Widżety można dodawać na dwa sposoby:

- Kliknij przycisk *Add* lub strzałkę obok niego, a następnie wybierz *Add widget* z menu akcji. Po skonfigurowaniu widżet zostanie dodany w domyślnym rozmiarze i umieszczony za wszystkimi istniejącymi widżetami.

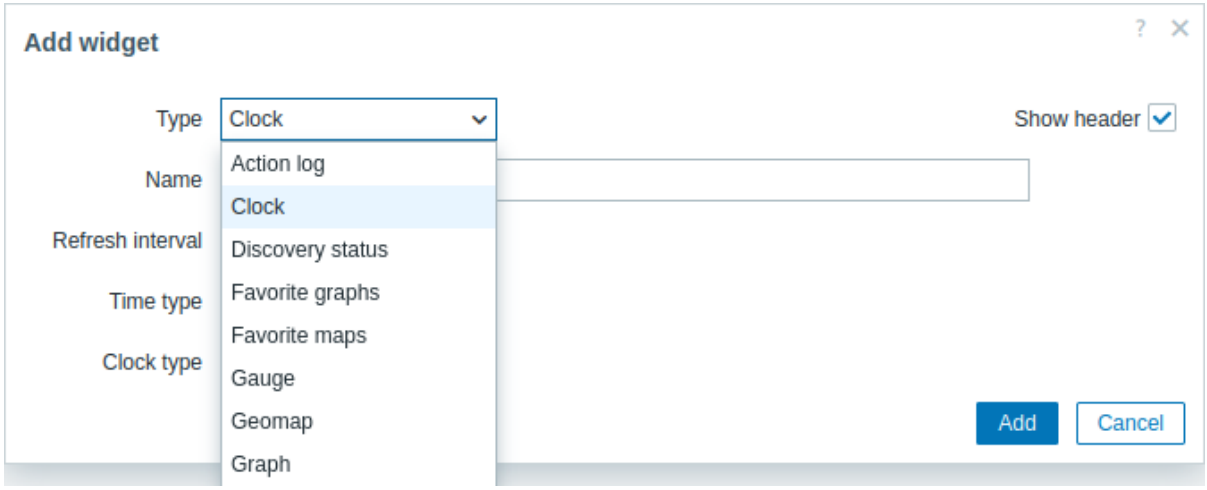


- Przesuń mysz na puste miejsce na dashboardzie. Zauważysz pojawienie się elementu zastępczego, a następnie kliknij go, aby otworzyć formularz konfiguracji widżetu. Po skonfigurowaniu widżet zostanie dodany w domyślnym rozmiarze lub zostanie zmieniony jego rozmiar, aby dopasować go do dostępnej przestrzeni. Alternatywnie przeciągnij element zastępczy, aby określić rozmiar widżetu, i zwolnij go, aby otworzyć formularz konfiguracji widżetu (jeśli widżet został skopiowany do schowka, najpierw zostaniesz poproszony o wybór między *Add widget* a *Paste widget*).



W formularzu konfiguracji widżetu:

1. Wybierz typ widżetu.
2. Skonfiguruj parametry widżetu.
3. Kliknij *Add*, aby dodać widżet do dashboardu.



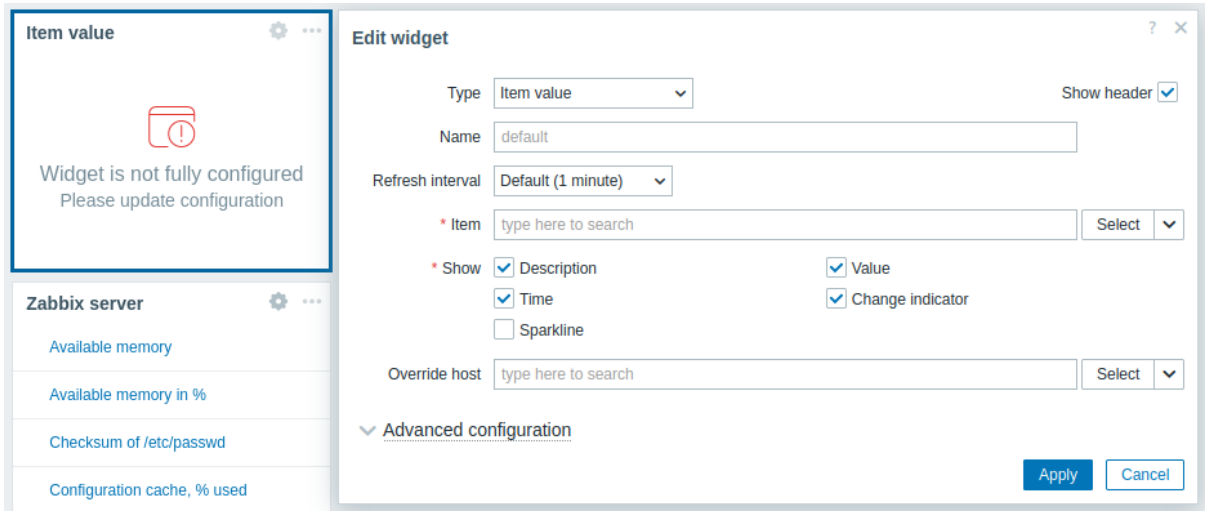
Edytowanie widżetów

Widżety można zmieniać rozmiar i położenie w trybie edycji dashboardu, przeciągając ich pasek tytułu. Możesz także zmienić położenie formularza konfiguracji widżetu, przeciągając jego pasek tytułu.

Każdy widżet zawiera elementy sterujące w prawym górnym rogu:

-  - edytuj widżet
- - otwórz menu widżetu

Podczas edycji widżet jest podświetlany, a zmiany są podglądane w czasie rzeczywistym. Jeśli wymagane parametry widżetu nie są skonfigurowane, widżet przechodzi w stan *Widget is not fully configured*.



Po zakończeniu edycji kliknij *Apply* w widżecie, a następnie *Save changes* w prawym górnym rogu dashboardu, aby zastosować zmiany.

Kopiowanie i wklejanie widżetów

Możesz kopiować i wklejać widżety pulpitu, aby szybko tworzyć nowe widżety z taką samą konfiguracją. Widżety można kopiować i wklejać w obrębie tego samego pulpitu lub między pulpitami otwartymi w różnych kartach.

Aby skopiować widżet, użyj menu widżetu. Aby wkleić skopiowany widżet (dostępne w trybie edycji pulpitu):

- Kliknij strzałkę obok przycisku *Dodaj* i wybierz *Wklej widżet*.
- Alternatywnie kliknij pusty obszar na pulpicie i wybierz *Wklej widżet*.

Możesz również wkleić skopiowany widżet na istniejący widżet, używając opcji *Wklej* w jego menu widżetu.

Tworzenie pokazu slajdów

Pokaz slajdów uruchomi się automatycznie, jeśli dashboard zawiera dwie lub więcej stron (zobacz Dodawanie stron) oraz jeśli spełniony jest jeden z poniższych warunków:

- opcja *Automatycznie uruchom pokaz slajdów* jest zaznaczona w właściwościach dashboardu
- adres URL dashboardu zawiera parametr `slideshow=1`

Strony są przełączane zgodnie z interwałami podanymi we właściwościach dashboardu i poszczególnych stron. Kliknij:

- *Zatrzymaj pokaz slajdów* - aby zatrzymać pokaz slajdów
- *Uruchom pokaz slajdów* - aby uruchomić pokaz slajdów



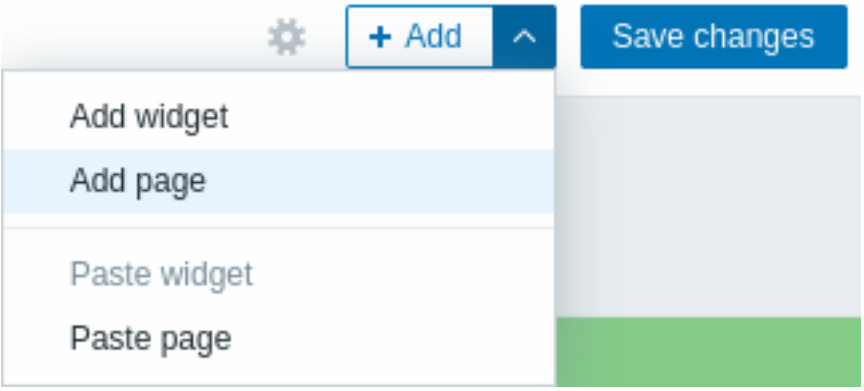
Elementy sterujące związane z pokazem slajdów są również dostępne w trybie kiosk (w którym wyświetlana jest tylko zawartość strony):

-  - zatrzymaj pokaz slajdów
-  - uruchom pokaz slajdów
-  - wróć o jedną stronę
-  - przejdź do następnej strony

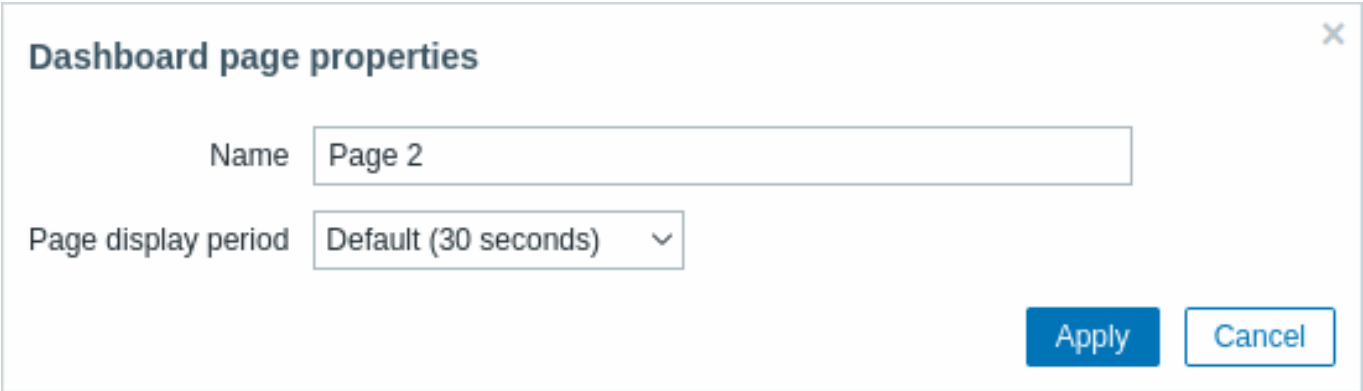
Dodawanie stron

Aby dodać nową stronę do dashboardu:

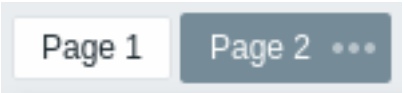
- Upewnij się, że dashboard jest w trybie edycji
- Kliknij strzałkę obok przycisku *Add* i wybierz opcję *Add page*



- Wypełnij ogólne parametry strony i kliknij *Apply*. Jeśli pozostawisz pole nazwy puste, strona zostanie dodana pod nazwą Page `N`, gdzie 'N' jest kolejnym numerem strony. Okres wyświetlania strony pozwala dostosować, jak długo strona jest wyświetlana w pokazie slajdów.



Zostanie dodana nowa strona, oznaczona nową kartą (*Page 2*).



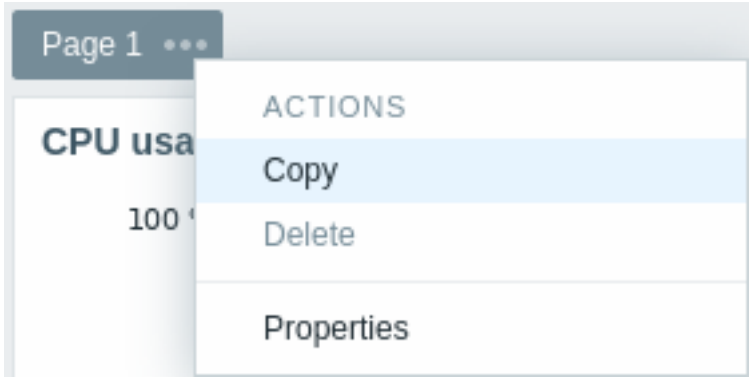
Strony można zmieniać kolejnością, przeciągając i upuszczając karty stron. Zmiana kolejności zachowuje oryginalne nazewnictwo stron. Zawsze można przejść do każdej strony, klikając jej kartę.

Gdy zostanie dodana nowa strona, jest ona pusta. Możesz dodać do niej widżety, jak opisano powyżej.

Kopiowanie/wklejanie stron

Strony dashboardu można kopiować i wklejać, co pozwala utworzyć nową stronę z właściwościami istniejącej. Można je wklejać w obrębie tego samego dashboardu lub do innego dashboardu.

Aby wkleić istniejącą stronę do dashboardu, najpierw skopiuj ją, używając menu strony:

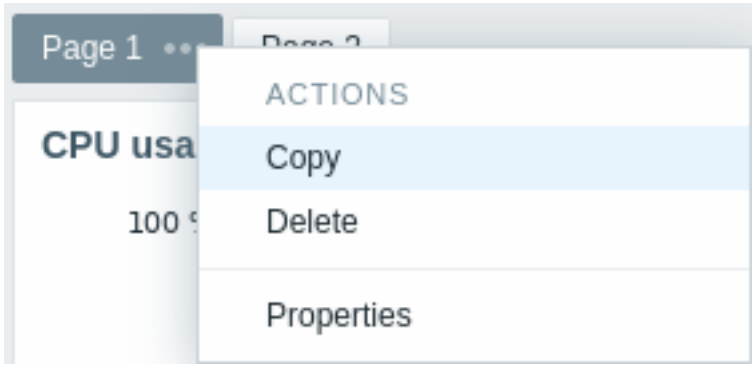


Aby wkleić skopiowaną stronę:

- Upewnij się, że dashboard jest w trybie edycji
- Kliknij strzałkę obok przycisku *Add* i wybierz opcję *Paste page*

Menu strony

Menu strony można otworzyć, klikając trzy kropki obok nazwy strony:

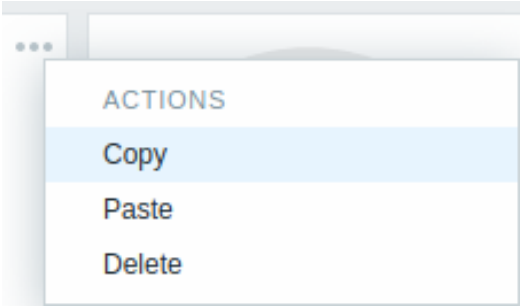


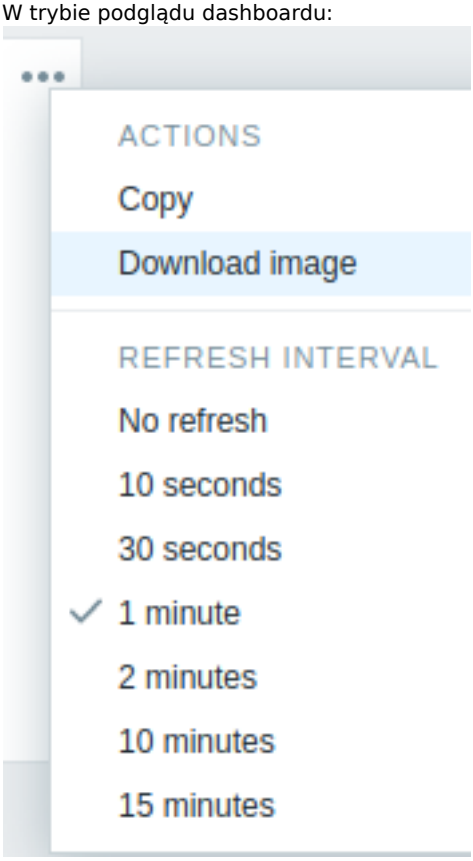
Zawiera ono następujące opcje:

- *Copy* - skopiuj stronę
- *Delete* - usuń stronę (strony można usuwać tylko w trybie edycji dashboardu)
- *Properties* - dostosuj parametry strony (nazwę oraz okres wyświetlania strony w pokazie slajdów)

Menu widgetu

Menu widgetu zawiera różne opcje w zależności od tego, czy dashboard jest w trybie edycji, czy podglądu:

Menu widgetu	Opcje
W trybie edycji dashboardu: 	*Copy* - skopiuj widget. *Paste* - wklej skopiowany widget w miejsce tego widgetu. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnego widgetu. *Delete* - usuń widget.

Menu widgetu	Opcje
W trybie podglądu dashboardu: 	*Copy* - skopiuj widget. *Download image* - pobierz widget jako obraz PNG. Ta opcja jest dostępna dla widgetów obsługujących tę funkcję (zobacz strony poszczególnych widgetów). W zależności od ustawień przeglądarki obraz jest automatycznie zapisywany w folderze Pobrane albo otwiera się okno dialogowe zapisu, umożliwiające wybranie lokalizacji. *Refresh interval* - wybierz częstotliwość odświeżania zawartości widgetu.

Uprawnienia do pulpitów

Uprawnienia do pulpitów dla zwykłych użytkowników oraz użytkowników typu „Admin” (pod warunkiem, że dostęp do pulpitów oraz opcja *Create and edit dashboards* są włączone dla roli użytkownika, która jest im przypisana) są ograniczone w sposób opisany poniżej.

- Mogą wyświetlać i klonować publiczny pulpit, nawet jeśli w ustawieniach udostępniania pulpitu nie dodano żadnych użytkowników ani grup użytkowników.
- Mogą wyświetlać i klonować prywatny pulpit, jeśli mają do niego co najmniej uprawnienia *Read* ustawione za pomocą ustawień udostępniania.
- Mogą edytować i usuwać pulpit tylko wtedy, gdy mają do niego uprawnienia *Read-write* ustawione za pomocą ustawień udostępniania.
- Nie mogą zmienić właściciela pulpitu.

1 Widżety dashboardu

Przegląd

Ta sekcja zawiera szczegóły parametrów wspólnych dla widżetów dashboard.

Parametry wspólne

Poniższe parametry są wspólne dla każdego widżetu:

Nazwa	Wprowadź nazwę widżetu.

Interwał odświeżania	Skonfiguruj domyślny interwał odświeżania. Domyślne interwały odświeżania dla widżetów mieszczą się w zakresie od *Brak odświeżania* do *15 minut*, w zależności od typu widżetu. Na przykład: - *Brak odświeżania* dla widżetu *URL*; - *1 minuta* dla widżetu *Action log*; - *15 minut* dla widżetu *Clock*. Interwały odświeżania można ustawić jako wartość domyślną dla wszystkich użytkowników. Przełącz dashboard w tryb edycji, kliknij przycisk edytuj widżet i wybierz żądany interwał odświeżania z listy rozwijanej. Każdy użytkownik może również ustawić własny interwał odświeżania widżetu. W trybie wyświetlania dashboardu kliknij przycisk z trzema kropkami na widżecie i wybierz żądany interwał odświeżania z listy rozwijanej. Należy pamiętać, że unikalny interwał odświeżania użytkownika ma pierwszeństwo przed ustawieniem widżetu i jest zachowywany nawet po zmianie ustawienia widżetu.
Pokaż nagłówek	Zaznacz pole wyboru, aby na stałe wyświetlać nagłówek widżetu. Gdy pole nie jest zaznaczone, nagłówek jest ukryty, aby oszczędzać miejsce, i staje się widoczny tylko po najechaniu myszą na widżet (zarówno w trybie wyświetlania, jak i edycji). Nagłówek jest również częściowo widoczny podczas przeciągania widżetu w nowe miejsce.

Parametry szczegółowe

Aby zobaczyć szczegółowe parametry dla każdego widżetu, przejdź do stron poszczególnych widżetów:

- Action log
- Clock
- Discovery status
- Favorite graphs
- Favorite maps
- Gauge
- Geomap
- Graph
- Graph (classic)
- Graph prototype
- Honeycomb
- Host availability
- Host card
- Host navigator
- Item card
- Item history
- Item navigator
- Item value
- Map
- Map navigation tree
- Pie chart
- Problem hosts
- Problems
- Problems by severity
- SLA report
- System information
- Top hosts
- Top items
- Top triggers
- Trigger overview
- URL
- Web monitoring

Parametry dynamiczne

Wiele widżetów ma parametry, które umożliwiają im współdzielenie danych konfiguracyjnych z innymi widżetami lub pulpitem.

Grupy hostów, hosty, pozycja, lista pozycji

Parametry *Host groups*, *Hosts*, *Item* i *Item list* umożliwiają wybór odpowiednio tych encji lub źródła danych zawierającego grupy hostów, hosty albo pozycje, dla których widżet może wyświetlać dane.

W przypadku parametrów *Host groups*, *Item* i *Item list* źródłem danych może być zgodny widżet z tego samego dashboardu.

W przypadku parametru *Hosts* źródłem danych może być zgodny widżet z tego samego dashboardu lub sam dashboard.

Note:
Widżet *Map* może również przekazywać dane grup hostów i hostów do zgodnych widżetów. Więcej informacji można znaleźć w sekcji Zachowanie widżetu.

Nadpisz host

Parametr *Override host* umożliwia wybranie źródła danych zawierającego host, dla którego widżet może wyświetlać dane. Źródłem danych może być zgodny widżet z tego samego dashboardu lub sam dashboard.

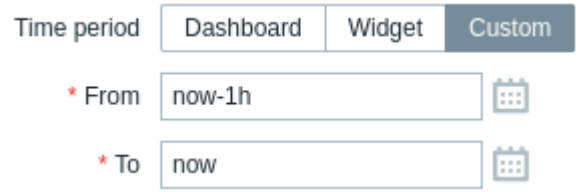


- Aby określić zgodny widżet, wpisz jego nazwę i wybierz go. Alternatywnie kliknij przycisk *Select* (lub przycisk listy rozwijanej, a następnie "Widget"), aby otworzyć okno podręczne z dostępnymi widżetami.
- Aby określić dashboard, kliknij przycisk listy rozwijanej, a następnie "Dashboard". Po zapisaniu dashboardu pole *Host* (do wybierania hostów) pojawi się u góry dashboardu.



Okres czasu

Parametr *Okres czasu* umożliwia wybranie źródła danych zawierającego okres czasu, dla którego widżet może wyświetlać dane. Źródłem danych może być zgodny widżet z tego samego dashboardu, sam dashboard lub okres czasu skonfigurowany w samym widżecie.



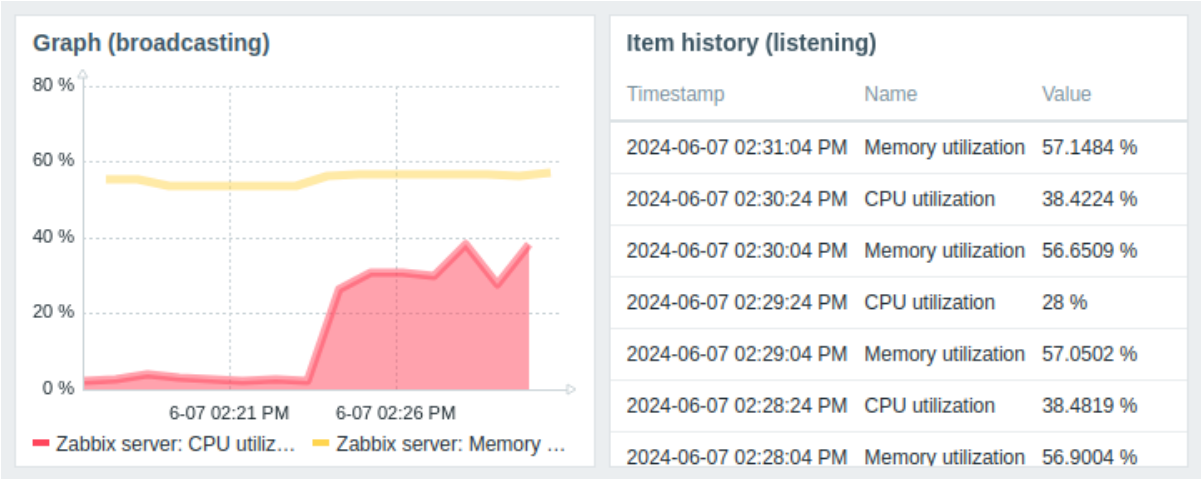
- Aby określić zgodny widżet, ustaw *Okres czasu* na "Widget", wprowadź jego nazwę i wybierz go. Alternatywnie kliknij przycisk *Wybierz*, aby otworzyć okno podręczne z dostępnymi widżetami.
- Aby określić dashboard, ustaw *Okres czasu* na "Dashboard". Po zapisaniu dashboardu na górze dashboardu pojawi się *selektor Okres czasu*.
- Aby skonfigurować okres czasu w samym widżecie, ustaw *Okres czasu* na "Custom" i wprowadź lub wybierz początek oraz koniec okresu czasu.

Note:
Niezależnie od konfiguracji *Okres czasu* widżetu, zgodne widżety mogą używać go jako źródła danych dla okresu czasu.

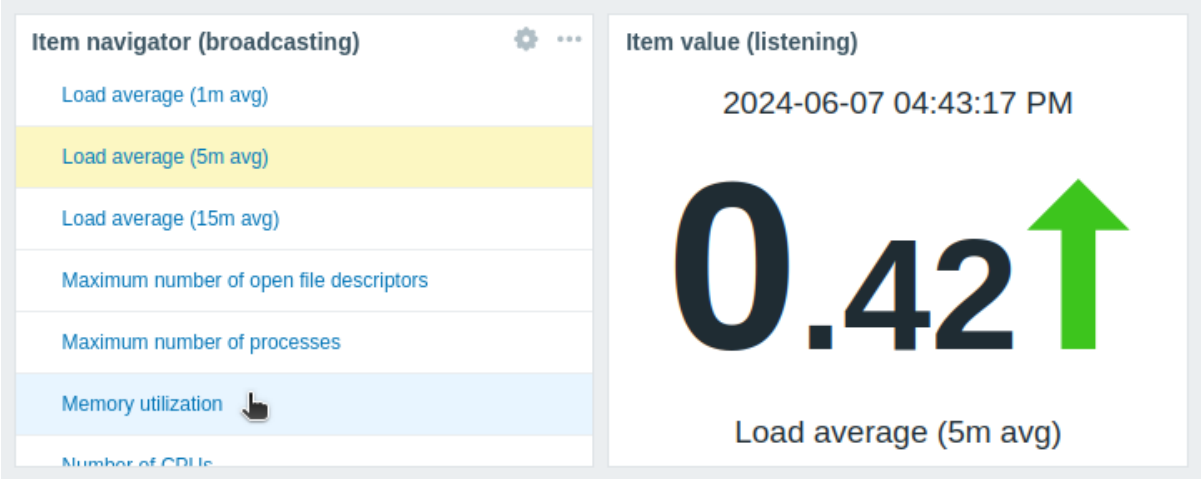
Zachowanie widgetów

Widgety różnią się sposobem, w jaki **rozgłaszają** dane do innych widgetów.

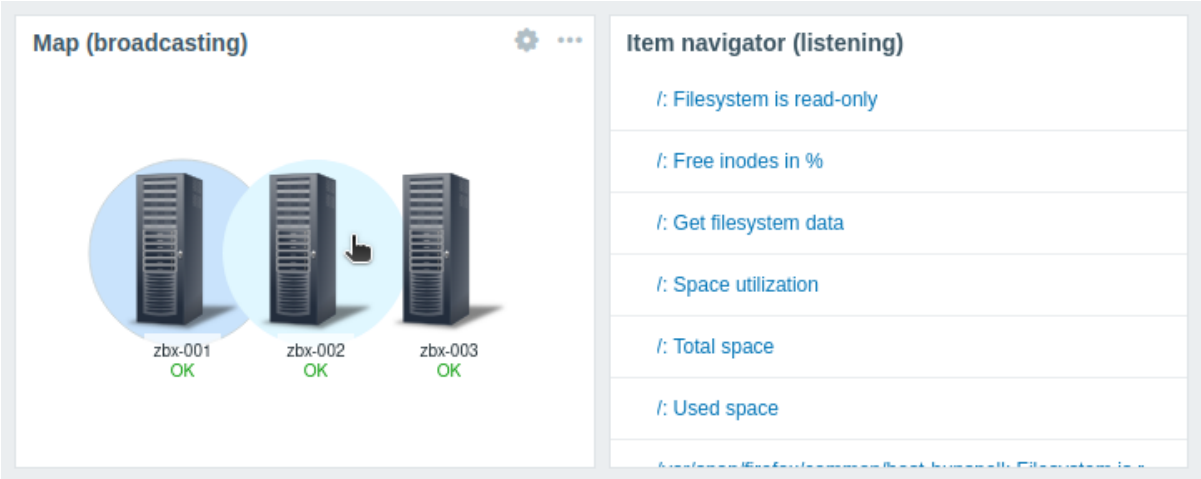
Wszystkie widgety zdolne do rozgłaszania danych zaczynają to robić automatycznie po utworzeniu. Na przykład widget *Graph* natychmiast rozgłasza dane o okresie czasu do widgetów nasłuchujących.



Widgety obsługujące wybór encji (np. *Item navigator*) automatycznie rozgłaszają dane dla pierwszej dostępnej encji. Na przykład widget *Item navigator* rozgłasza dane dla pierwszej pozycji na swojej liście pozycji. Gdy zostanie wybrana inna pozycja, rozgłasza dane dla tej pozycji. Po najechaniu myszą pozycja jest podświetlana na jasnoniebiesko; po zaznaczeniu jest podświetlana na żółto.

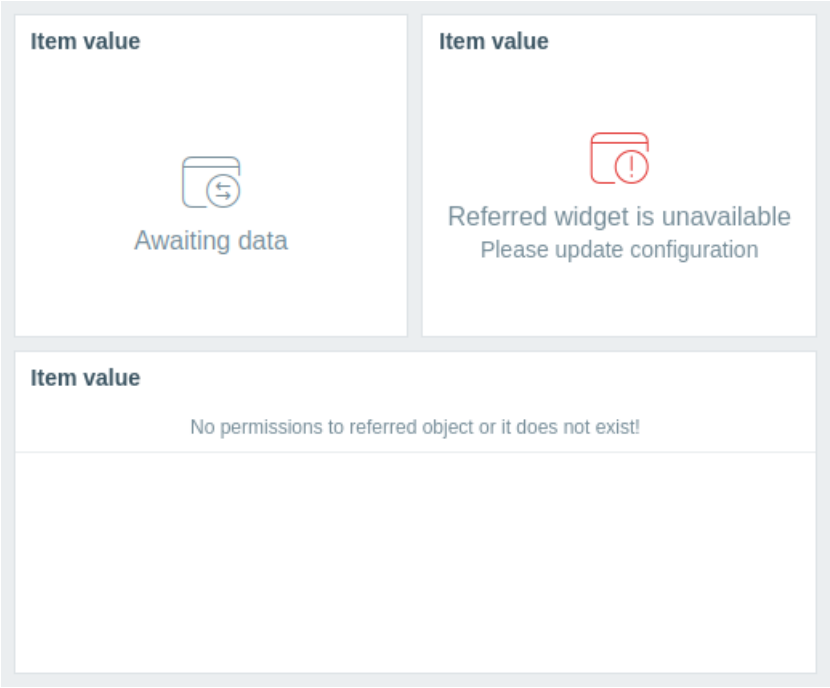


Widget *Map* również automatycznie rozgłasza dane dla pierwszego dostępnego elementu (np. host), który znajduje się najbliżej lewego górnego rogu widoku widgetu (w przypadku widgetu *Geomap* — najbliżej środka widoku widgetu). Gdy zostanie wybrany inny element, rozgłasza dane dla tego elementu. Po najechaniu myszą element jest podświetlany na jasnoniebiesko; po zaznaczeniu jest podświetlany na ciemnoniebiesko.



Widgety mają również określone zachowanie podczas **nasłuchiwania** danych z innych widgetów:

- Jeśli widget źródłowy danych nie rozgłasza danych, widget nasłuchujący przechodzi do stanu *Awaiting data*.
- Jeśli widget źródłowy danych został usunięty, zastąpiony niezgodnym widgetem lub przeniesiony na inną stronę dashboardu, widget nasłuchujący przechodzi do stanu *Referred widget is unavailable*.
- Jeśli określony host w źródle danych (widget lub dashboard) nie zawiera encji skonfigurowanej w widgetcie nasłuchującym (pozycja, wykres, mapa itp.) lub jeśli użytkownik nie ma uprawnień do dostępu do hosta, widget nasłuchujący wyświetla następujący komunikat: *"No permissions to referred object or it does not exist!"*



Informacje o możliwościach rozgłaszania i nasłuchiwania dla każdego widgetu znajdują się w sekcji Zgodność widgetów.

Zgodność widgetów

Niektóre widgety mogą przekazywać dane konfiguracyjne do innych widgetów, niektóre mogą nasłuchiwać danych, a niektóre mogą robić jedno i drugie. Na przykład:

- Widget *Action log* może pobierać tylko dane dotyczące okresu czasu z widgetów *Graph*, *Graph (classic)* i *Graph prototype*.
- Widget *Geomap* może przekazywać dane hosta do widgetów, które nasłuchują takich danych (*Honeycomb*, *Top items* itp.), a także może nasłuchiwać danych o grupach hostów i hostach z widgetów, które je przekazują (*Honeycomb*, *Problem hosts* itp.).
- Widget *Clock* nie może ani przekazywać, ani nasłuchiwać żadnych danych.

Poniższa tabela przedstawia możliwości przekazywania i nasłuchiwania dla każdego widgetu.

Widget	Przekazuje	Nasłuchuje
Action log	-	Okres czasu
Clock	-	-
Discovery status	-	-
Favorite graphs	-	-
Favorite maps	-	-
Gauge	-	Hosty, pozycje
Geomap	Hosty	Grupy hostów, Hosty
Graph	Okres czasu	Hosty, pozycje, Okres czasu
Graph (classic)	Okres czasu	Hosty, pozycje, Grafy, Okres czasu
Graph prototype	Okres czasu	Hosty, Okres czasu
Honeycomb	Hosty, pozycje	Grupy hostów, Hosty
Host availability	-	Grupy hostów
Host card	-	Hosty
Host navigator	Hosty	Grupy hostów
Item card	-	Hosty, pozycje, Okres czasu
Item history	pozycje	Hosty, Okres czasu
Item navigator	pozycje	Grupy hostów, Hosty
Item value	-	Hosty, pozycje, Okres czasu
Map	Grupy hostów, Hosty	Mapy
Map navigation tree	Mapy	-
Pie chart	-	pozycje, Okres czasu
Problem hosts	Grupy hostów	Grupy hostów, Hosty
Problems	Zdarzenia	Grupy hostów, Hosty
Problems by severity	Grupy hostów	Grupy hostów, Hosty
SLA report	-	-
System information	-	-

Widget	Przekazuje	Nasłuchuje
Top hosts	Hosty	Grupy hostów, Hosty
Top items	-	Grupy hostów, Hosty
Top triggers	-	Okres czasu
Trigger overview	-	Grupy hostów, Hosty
URL	-	Hosty
Web monitoring	Grupy hostów	Grupy hostów, Hosty

1 Dziennik akcji

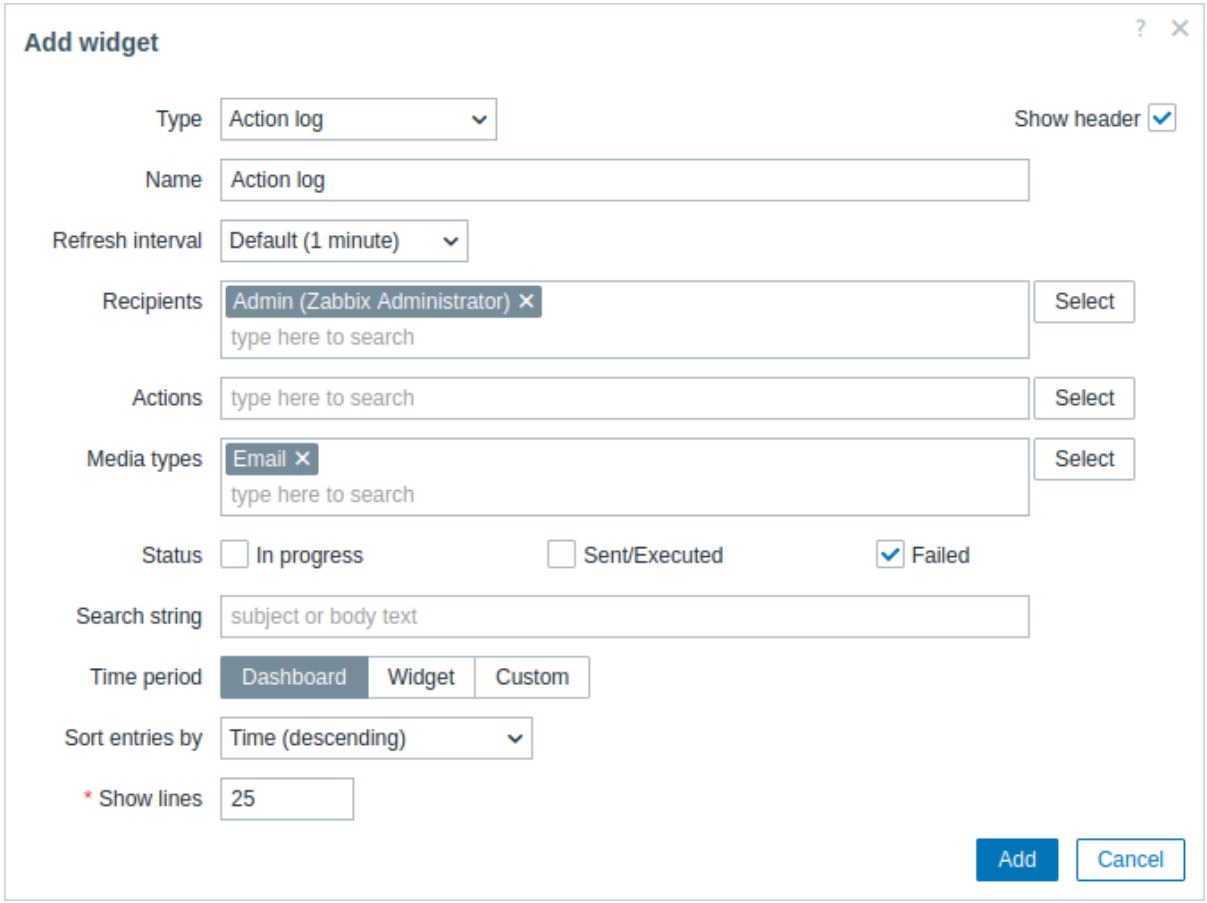
Przegląd

W widżecie dziennika działań można wyświetlać szczegóły operacji działań (powiadomienia, zdalne polecenia). Odtwarza on informacje z *Raporty* → *Dziennik działań*.

Można wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować, wybierz *Action log* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Recipients	Filtruj wpisy według odbiorców. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy odbiorcy zostanie wyświetlona lista rozwijana pasujących odbiorców. Jeśli nie zostaną wybrani żadni odbiorcy, zostaną wyświetlone szczegóły operacji akcji dla wszystkich odbiorców.
Actions	Filtruj wpisy według akcji. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy akcji zostanie wyświetlona lista rozwijana pasujących akcji. Jeśli nie zostaną wybrane żadne akcje, zostaną wyświetlone szczegóły operacji akcji dla wszystkich akcji.

Media types	Filtruj wpisy według typów mediów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy typu medium zostanie wyświetlona lista rozwijana pasujących typów mediów. Jeśli nie zostaną wybrane żadne typy mediów, zostaną wyświetlone szczegóły operacji akcji dla wszystkich typów mediów.
Status	Zaznacz pole wyboru, aby filtrować wpisy według odpowiedniego statusu: **In progress** - wyświetlane są operacje akcji, które są w toku; **Sent/Executed** - wyświetlane są operacje akcji, które wysłały powiadomienie lub zostały wykonane; **Failed** - wyświetlane są operacje akcji, które zakończyły się niepowodzeniem.
Search string	Filtruj wpisy według treści wiadomości/zdalnego polecenia. Jeśli wpiszesz tutaj ciąg znaków, zostaną wyświetlone tylko te operacje akcji, których wiadomość/zdalne polecenie zawiera wpisany ciąg. Makra nie są rozwijane.
Time period	Filtruj wpisy według okresu czasu. Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Time period* jako źródło danych; **Widget** - ustaw zgodny widżet określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli ustawione, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu myszą.
Widget	Wprowadź lub wybierz zgodny widżet jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* ma ustawioną wartość "Widget".
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma ustawioną wartość "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma ustawioną wartość "Custom".
Sort entries by	Sortuj wpisy według: **Time** (malejąco lub rosnąco); **Type** (malejąco lub rosnąco); **Status** (malejąco lub rosnąco); **Recipient** (malejąco lub rosnąco).
Show lines	Ustaw, ile wierszy dziennika akcji będzie wyświetlanych w widżecie.

2 Zegar

Przegląd

W widżecie zegara można wyświetlać czas lokalny, serwera lub określonego hosta.

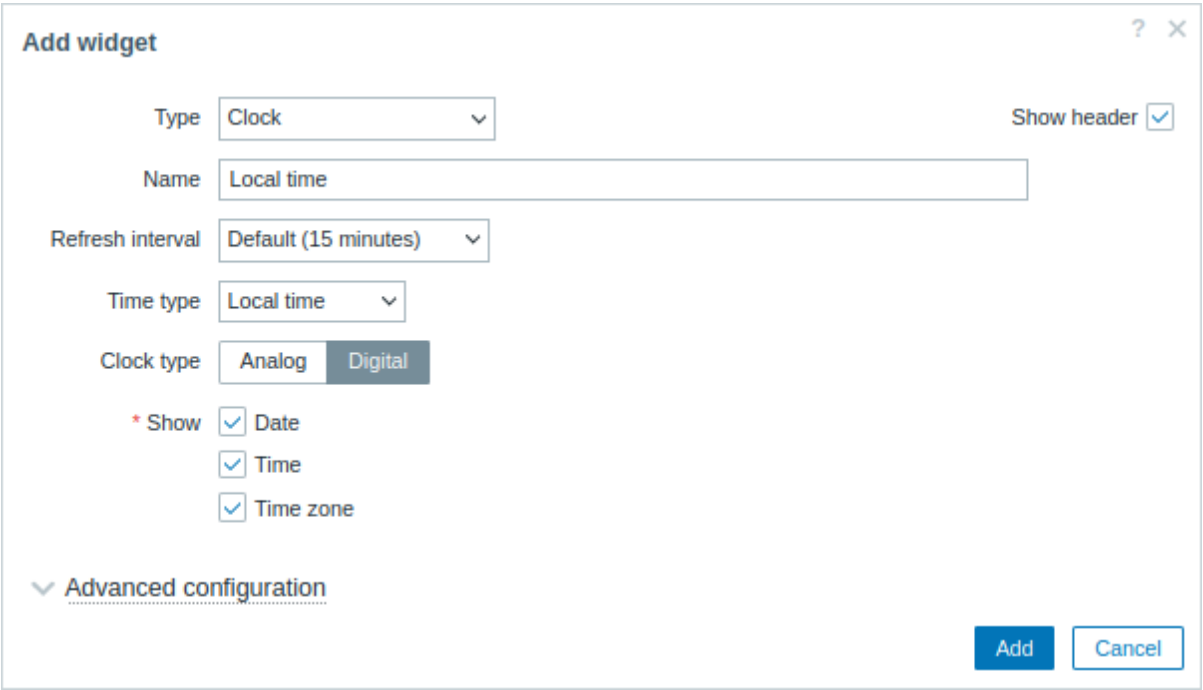
Można wyświetlać zarówno zegary analogowe, jak i cyfrowe:





Konfiguracja

Aby skonfigurować, wybierz *Clock* jako typ:



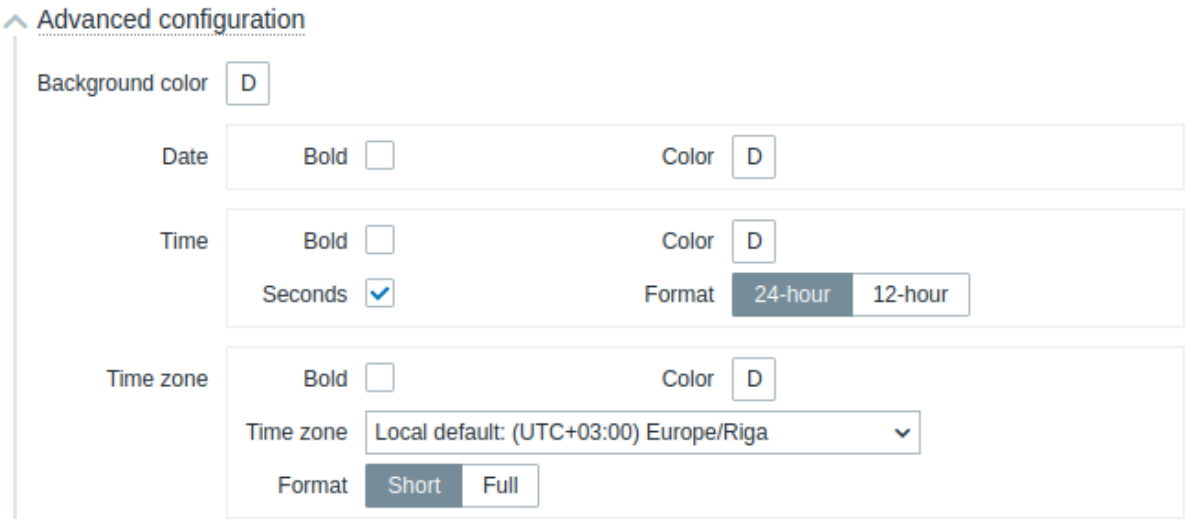
Oprócz parametrów, które są wspólne dla wszystkich widżetów, możesz ustawić następujące opcje specyficzne:

Time type	Wybierz czas lokalny, serwera lub określonego hosta. Czas serwera będzie identyczny z czasem ustawionym globalnie lub dla użytkownika Zabbix.
Item	Wybierz pozycję do wyświetlania czasu. Aby wyświetlić czas hosta, użyj pozycji `system.localtime[local]` item. Ta pozycja musi istnieć na hoście. To pole jest dostępne tylko wtedy, gdy wybrano *Host time*.
Clock type	Wybierz typ zegara: **Analog** - zegar analogowy **Digital** - zegar cyfrowy
Show	Wybierz jednostki informacji (data, czas, strefa czasowa) do wyświetlenia na zegarze cyfrowym. Rozmiar jednostki jest dynamiczny i wypełnia całą dostępną przestrzeń widżetu na podstawie wyświetlanych jednostek, rozmiaru widżetu i skalowania wyświetlania. To pole jest dostępne tylko wtedy, gdy w polu *Clock type* wybrano ”Digital”; należy wybrać co najmniej jeden typ jednostki informacji.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje zaawansowanej konfiguracji dla zegara cyfrowego. Ta sekcja jest dostępna tylko wtedy, gdy w polu *Clock type* wybrano ”Digital”.

Konfiguracja zaawansowana

Zaawansowane opcje konfiguracji są dostępne w zwijanej sekcji *Konfiguracja zaawansowana* i tylko dla tych elementów, które są wybrane w polu *Pokaż* (zobacz wyżej).

Dodatkowo konfiguracja zaawansowana umożliwia zmianę koloru tła dla całego widżetu.



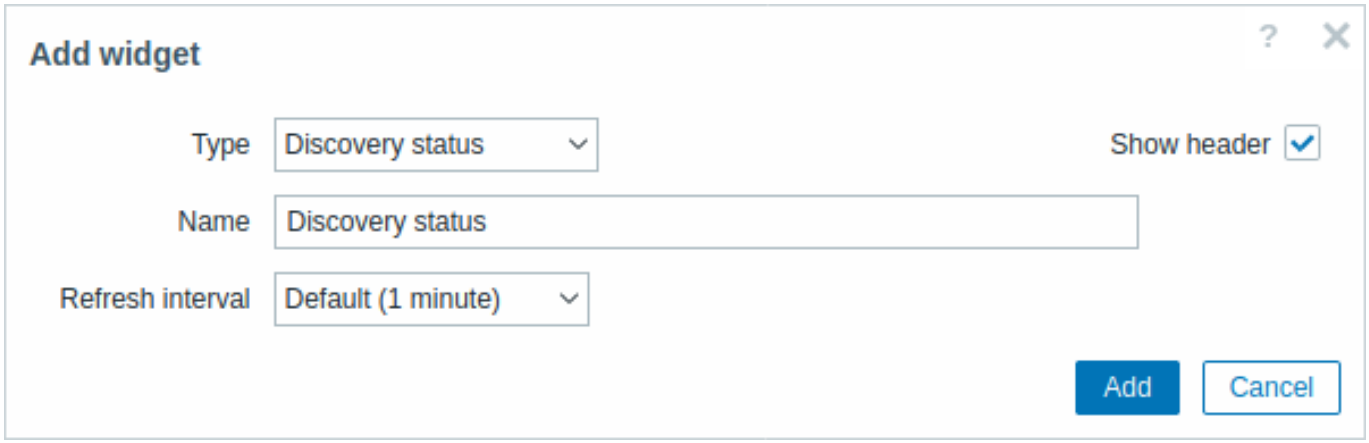
Kolor tła	Wybierz kolor tła z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Data	
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać datę pogrubioną czcionką.
Kolor	Wybierz kolor daty z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Czas	
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać czas pogrubioną czcionką.
Kolor	Wybierz kolor czasu z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Sekundy	Zaznacz pole wyboru, aby wyświetlać sekundy. W przeciwnym razie będą wyświetlane tylko godziny i minuty.
Format	Wybierz wyświetlanie czasu w formacie 24-godzinnym lub 12-godzinnym.
Strefa czasowa	
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać strefę czasową pogrubioną czcionką.
Kolor	Wybierz kolor strefy czasowej z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Strefa czasowa	Wybierz strefę czasową.
Format	Wybierz wyświetlanie strefy czasowej w formacie skróconym (np. `New York`) lub pełnym (np. `(UTC-04:00) America/New York`).

3 Status wykrywania

Przegląd

Ten widżet wyświetla podsumowanie stanu aktywnych reguł wykrywania sieci.



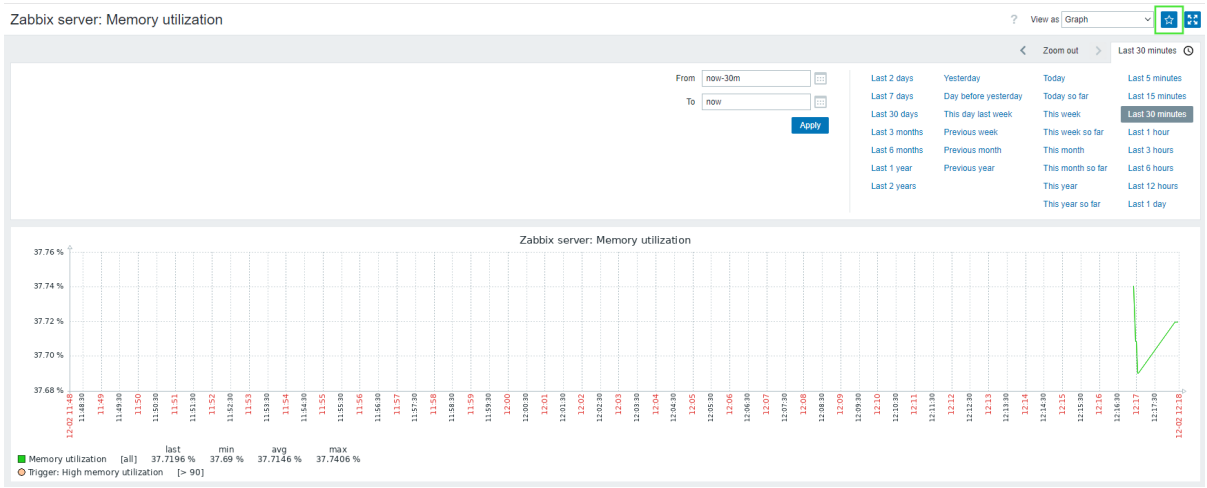
Wszystkie parametry konfiguracji są wspólne dla wszystkich widżetów.

4 Ulubione wykresy

Przegląd

Ten widżet zawiera skróty do najczęściej potrzebnych wykresów, posortowane alfabetycznie.

Lista skrótów jest uzupełniana, gdy wyświetlasz wykres w Monitoring -> Latest data -> Graphs, a następnie klikasz jego przycisk  *Dodaj do ulubionych*.



Wszystkie parametry konfiguracji są wspólne dla wszystkich widżetów.

5 Ulubione mapy

Przegląd

Ten widżet zawiera skróty do najczęściej potrzebnych map, posortowane alfabetycznie.

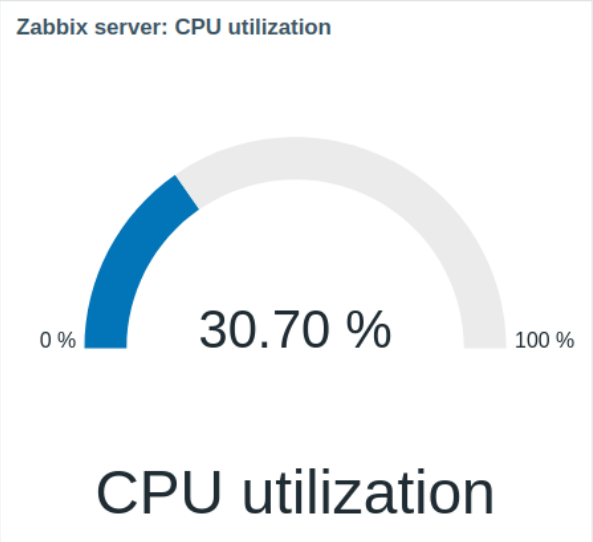
Lista skrótów jest uzupełniana, gdy wyświetlisz mapę, a następnie klikniesz jej przycisk *Dodaj do ulubionych*.

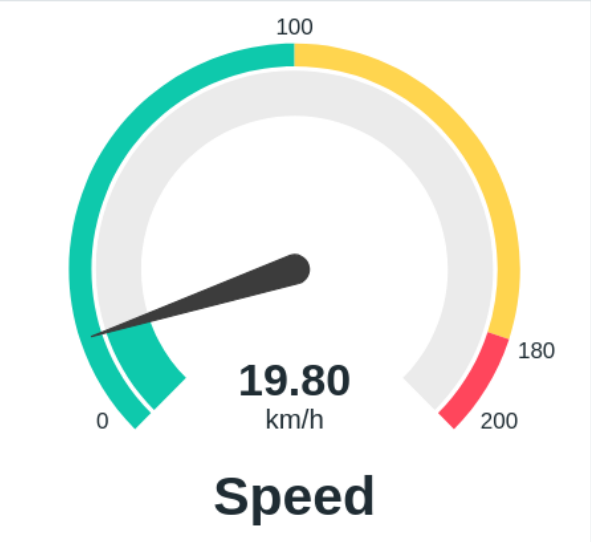
Wszystkie parametry konfiguracji są wspólne dla wszystkich widżetów.

6 Wskaźnik

Przegląd

Widżet gauge wyświetla wartość pojedynczej pozycji w formie wskaźnika.

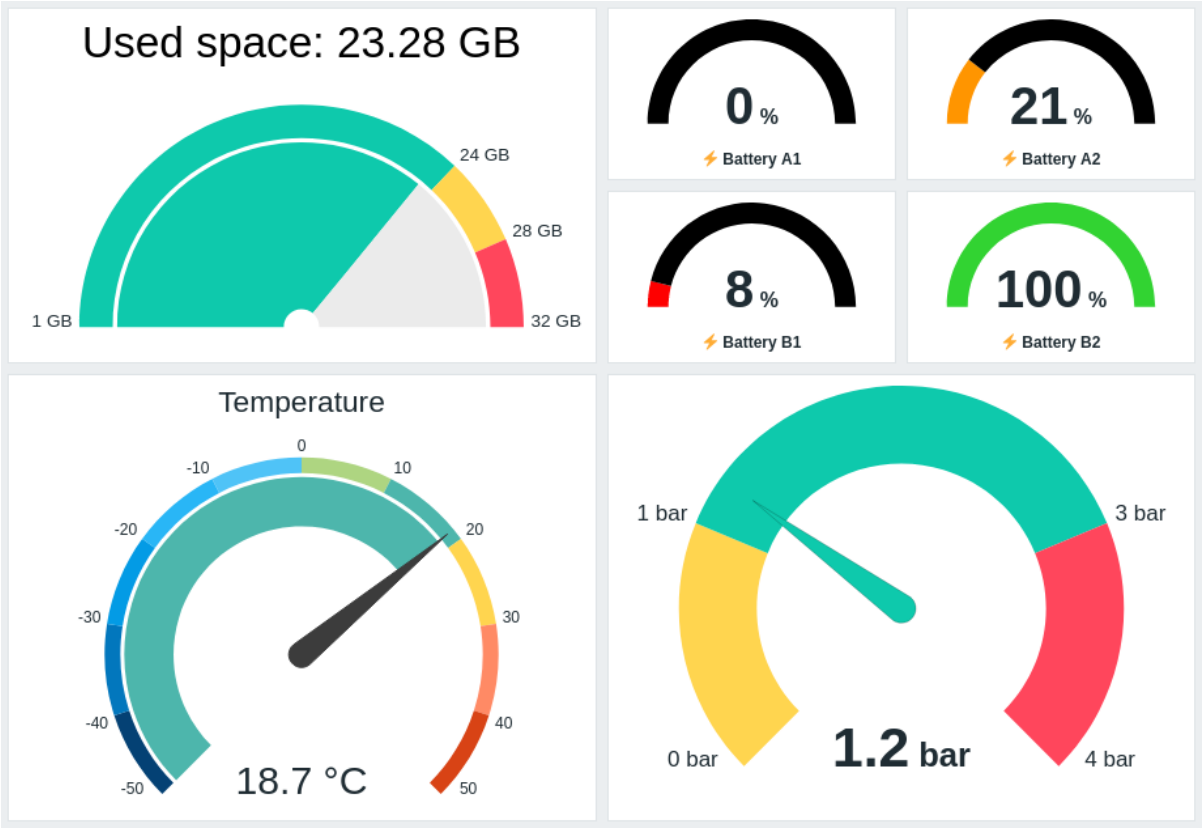




Po skonfigurowaniu widżet może składać się z następujących elementów:

- Opis pozycji (na przykład ”CPU utilization”, ”Speed”)
- Wartość pozycji (na przykład ”30.70”, ”19.80”)
- Jednostki wartości pozycji (na przykład ”%”, ”km/h”)
- Skala wskaźnika (na przykład ”0%/100%”, ”0/100/180/200”)
- Łuk wskaźnika (łuk wartości wskaźnika i łuk progów wskaźnika)
- Wskazówka wskaźnika

Widżet można wizualnie dostosować za pomocą opcji zaawansowanej konfiguracji, aby uzyskać szeroką gamę stylów wizualnych:



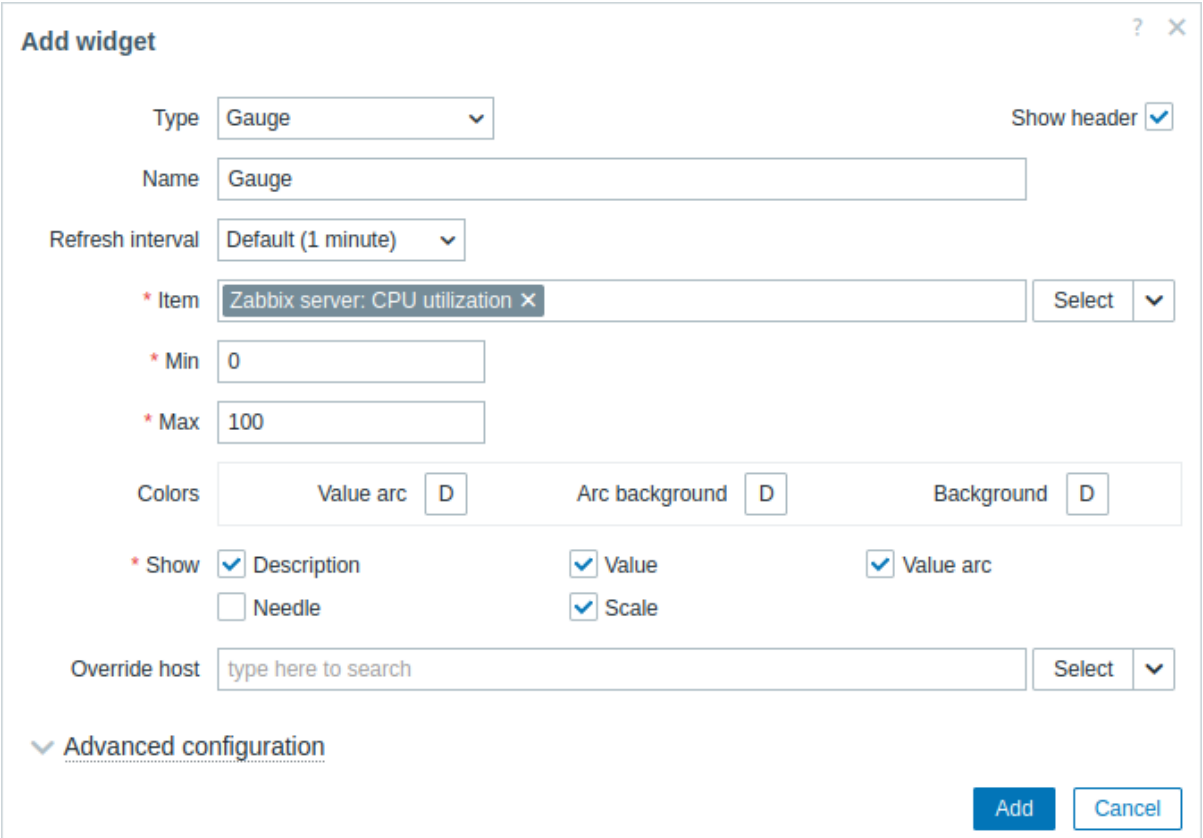
Widżet gauge może wyświetlać tylko wartości liczbowe. Wyświetlanie wartości binarnych nie jest obsługiwane.

Kliknięcie widżetu gauge prowadzi do wykresu dla tej pozycji.

Informacje wyświetlane w widżecie *Gauge* można pobrać jako obraz PNG, wybierając opcję *Download image* w menu widżetu.

Konfiguracja

Aby skonfigurować, wybierz *Gauge* jako typ:

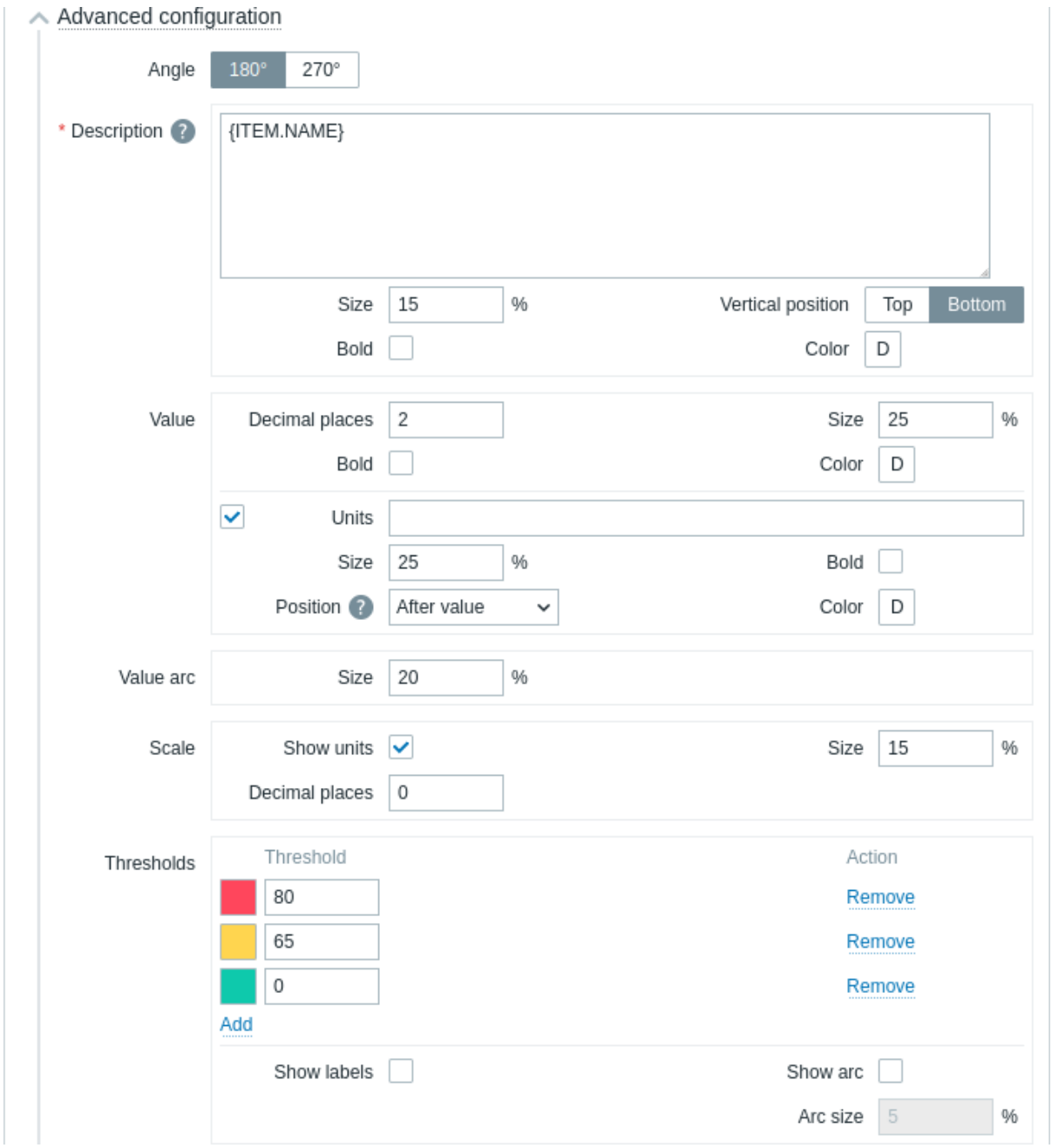


Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz zgodny widżet jako źródło danych dla pozycji. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy pozycji spowoduje wyświetlenie listy rozwijanej pasujących pozycji. Pamiętaj, że możesz wybrać tylko pozycje zwracające dane numeryczne.
Min	Wprowadź minimalną wartość wskaźnika. Obsługiwane są przyrostki (na przykład ”1d”, ”2w”, ”4K”, ”8G”). Obsługiwane są mapowania wartości.
Max	Wprowadź maksymalną wartość wskaźnika. Obsługiwane są przyrostki (na przykład ”1d”, ”2w”, ”4K”, ”8G”). Obsługiwane są mapowania wartości.
Colors	Wybierz kolor z selektora kolorów: **Value arc** - wybierz kolor łuku wartości wskaźnika; **Arc background** - wybierz kolor tła łuku wartości wskaźnika i łuku progów wskaźnika; **Background** - wybierz kolor tła widżetu. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Jeśli ustawiono *Thresholds*, domyślny kolor dla **Value arc** zależy od koloru progu. Aby powrócić do koloru domyślnego, kliknij przycisk *Use default* w selektorze kolorów.
Show	Zaznacz pole wyboru, aby wyświetlić odpowiedni element wskaźnika - opis, wartość, łuk wartości, wskazówkę, skalę (wartość minimalną i maksymalną wskaźnika na początku i końcu łuku wskaźnika). Odznacz, aby ukryć. Należy wybrać co najmniej jeden element. Pamiętaj, że wskazówka wskaźnika i skala mogą być wyświetlane, jeśli wyświetlany jest łuk wartości wskaźnika lub łuk progów wskaźnika (zobacz opcje zaawansowanej konfiguracji). Pamiętaj również, że jeśli wskazówka wskaźnika jest wyświetlana, wartość jest umieszczana pod wskazówką; jeśli wskazówka jest ukryta, wartość jest wyrównana do dolnej części łuku wskaźnika.
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardu.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje zaawansowanej konfiguracji. W tym miejscu możesz również dostosować elementy wskaźnika wybrane w polu *Show*.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*:



Kąt	Wybierz kąt wskaźnika (180° lub 270°).
Opis	
Opis	Wprowadź opis pozycji. Ten opis może zastąpić domyślną nazwę pozycji. Obsługiwane są opisy wielowierszowe. Możliwe jest połączenie tekstu i obsługiwanych makr. Obsługiwane są {HOST.*}, {ITEM.*}, {INVENTORY.*} oraz makra użytkownika.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla opisu pozycji (w procentach, względem całkowitej wysokości widżetu).
Pozycja pionowa	Wybierz pionowe położenie opisu pozycji (u góry lub u dołu, względem łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać opis pozycji pogrubioną czcionką.
Kolor	Wybierz kolor opisu pozycji z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Wartość	
Miejsca dziesiętne	Wprowadź liczbę miejsc dziesiętnych wyświetlanych przy wartości. Ta opcja ma wpływ tylko na pozycje zwracające dane numeryczne (float).
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla wartości (w procentach, względem wysokości łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać wartość pogrubioną czcionką.
Kolor	Wybierz kolor wartości z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Jednostki	
Jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji. Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostki ustawione w konfiguracji pozycji.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla jednostek pozycji (w procentach, względem wysokości łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać jednostki pozycji pogrubioną czcionką.
Pozycja	Wybierz położenie jednostek pozycji (nad, pod, przed lub za, względem wartości pozycji). Ta opcja jest ignorowana dla następujących jednostek związanych z czasem: unixtime, uptime, s.
Kolor	Wybierz kolor jednostek pozycji z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Łuk wartości	
Rozmiar łuku	Wprowadź wysokość rozmiaru łuku wartości wskaźnika (w procentach, względem promienia łuku wskaźnika).
Wskazówka	
Kolor	Wybierz kolor wskazówki wskaźnika z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Jeśli ustawiono *Progi*, domyślny kolor wskazówki zależy od koloru progu. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Skala	
Pokaż jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki przy minimalnej i maksymalnej wartości wskaźnika.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla minimalnej i maksymalnej wartości wskaźnika (w procentach, względem wysokości łuku wskaźnika).
Miejsca dziesiętne	Wprowadź liczbę miejsc dziesiętnych wyświetlanych przy minimalnej i maksymalnej wartości wskaźnika. Ta opcja ma wpływ tylko na pozycje zwracające dane numeryczne (float).
Progi	
Progi	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość numeryczną. Lista progów zostanie posortowana rosnąco podczas zapisywania. Należy pamiętać, że kolory skonfigurowane jako progi będą wyświetlane poprawnie tylko dla pozycji numerycznych. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Pokaż etykiety	Zaznacz pole wyboru, aby wyświetlać wartości progów jako etykiety na skali wskaźnika.
Pokaż łuk	Zaznacz pole wyboru, aby wyświetlać łuk progów wskaźnika.
Rozmiar łuku	Wprowadź wysokość rozmiaru łuku progów wskaźnika (w procentach, względem promienia łuku wskaźnika).

7 Geomap

Przegląd

Widżet geomapy wyświetla hosty jako znaczniki na mapie geograficznej, korzystając z otwartoźródłowej biblioteki JavaScript do interaktywnych map Leaflet.

> **Note:**
> Zabbix oferuje wielu predefiniowanych dostawców usług kafelków mapy oraz możliwość dodania niestandardowego dostawcy usług kafelków, a nawet hostowania własnych kafelków (konfigurowalne w sekcji menu *Administration* > *General* > *Geographical maps* menu section).

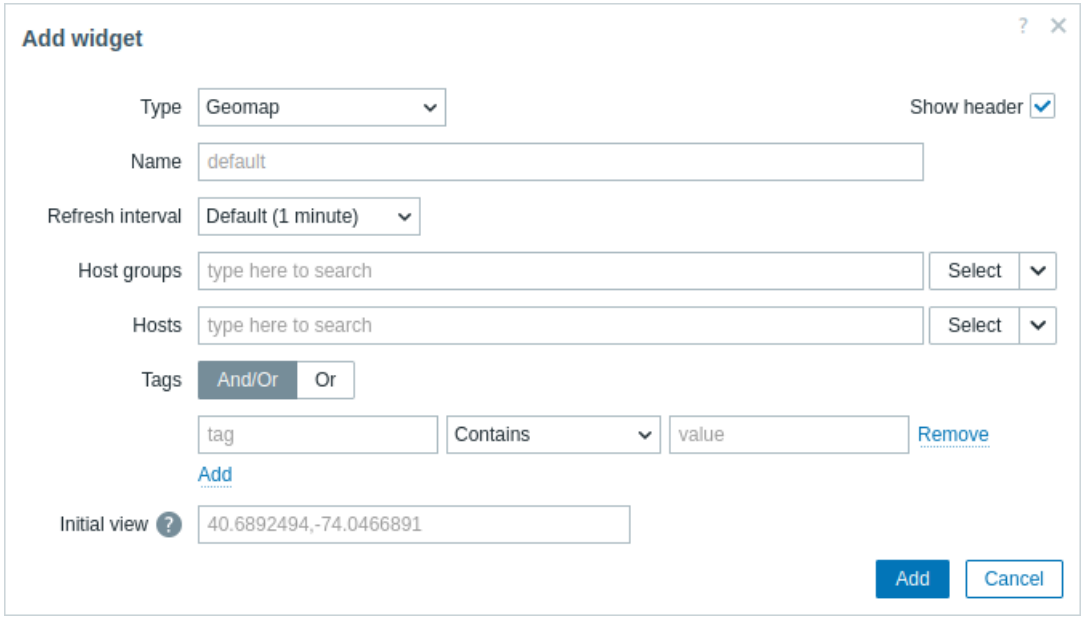
Domyślnie widżet wyświetla wszystkie włączone hosty, które mają zdefiniowane prawidłowe współrzędne w ich inwentarzu. Można skonfigurować filtrowanie hostów w parametrach widżetu.

Prawidłowe współrzędne hosta to:

- Szerokość geograficzna: od -90 do 90 (może być liczbą całkowitą lub zmiennoprzecinkową)
- Długość geograficzna: od -180 do 180 (może być liczbą całkowitą lub zmiennoprzecinkową)

Konfiguracja

Aby dodać widżet, wybierz *Geomap* jako typ.



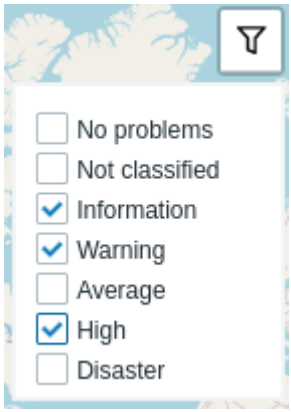
Oprócz parametrów, które są wspólne dla wszystkich widżetów, można ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, które mają być wyświetlane na mapie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Jeśli w polach *Host groups* i *Hosts* nic nie zostanie wybrane, zostaną wyświetlone wszystkie hosty z prawidłowymi współrzędnymi. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.
Hosts	Wybierz hosty, które mają być wyświetlane na mapie. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli w polach *Host groups* i *Hosts* nic nie zostanie wybrane, zostaną wyświetlone wszystkie hosty z prawidłowymi współrzędnymi. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.
Tags	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu zostaną pogrupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.

Initial view	Współrzędne środka oddzielone przecinkami oraz opcjonalny poziom powiększenia, które mają być wyświetlane podczas początkowego ładowania widżetu w formacie `<latitude>,<longitude>,<zoom>` Jeśli określono początkowy poziom powiększenia, widżet Geomap zostanie załadowany z podanym poziomem powiększenia. W przeciwnym razie początkowy poziom powiększenia jest obliczany jako połowa maksymalnego poziomu powiększenia dla danego dostawcy kafelków. Początkowy widok jest ignorowany, jeśli ustawiono widok domyślny (patrz poniżej). Przykłady: 40.6892494,-74.0466891 40.6892494,-122.0466891

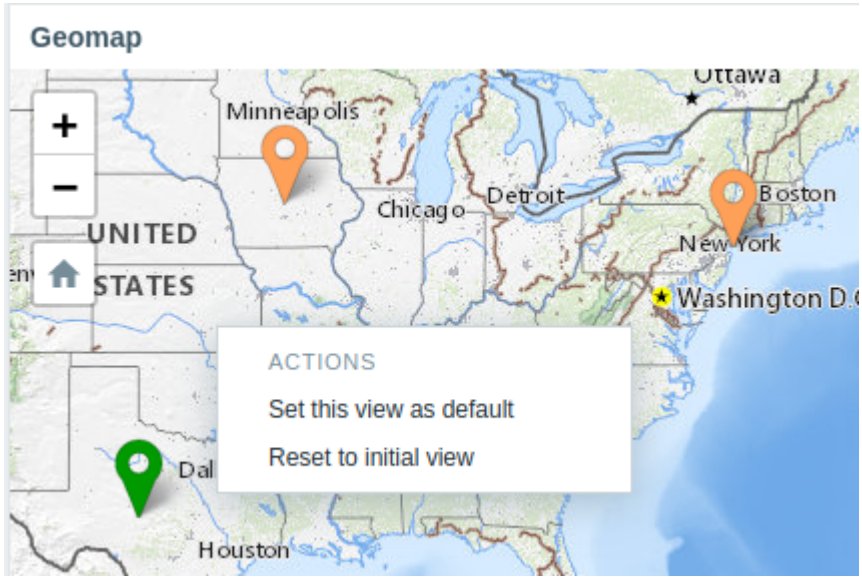
Markery hostów wyświetlane na mapie mają kolor najbardziej poważnego problemu hosta, a kolor zielony, jeśli host nie ma żadnych problemów. Kliknięcie markera hosta umożliwia wyświetlenie widocznej nazwy hosta oraz liczby nierozwiązanych problemów pogrupowanych według ważności. Kliknięcie widocznej nazwy otworzy menu hosta.

Hosty wyświetlane na mapie można filtrować według ważności problemu. Naciśnij ikonę filtra w prawym górnym rogu widżetu i zaznacz wymagane ważności.



Można powiększać i pomniejszać mapę za pomocą przycisków plus i minus w lewym górnym rogu widżetu albo za pomocą kółka przewijania myszy lub touchpada. Aby ustawić bieżący widok jako domyślny, kliknij prawym przyciskiem myszy w dowolnym miejscu na mapie i wybierz *Ustaw ten widok jako domyślny*. To ustawienie zastąpi parametr widżetu *Początkowy widok* dla bieżącego użytkownika. Aby cofnąć to działanie, ponownie kliknij prawym przyciskiem myszy w dowolnym miejscu na mapie i wybierz *Przywróć początkowy widok*.

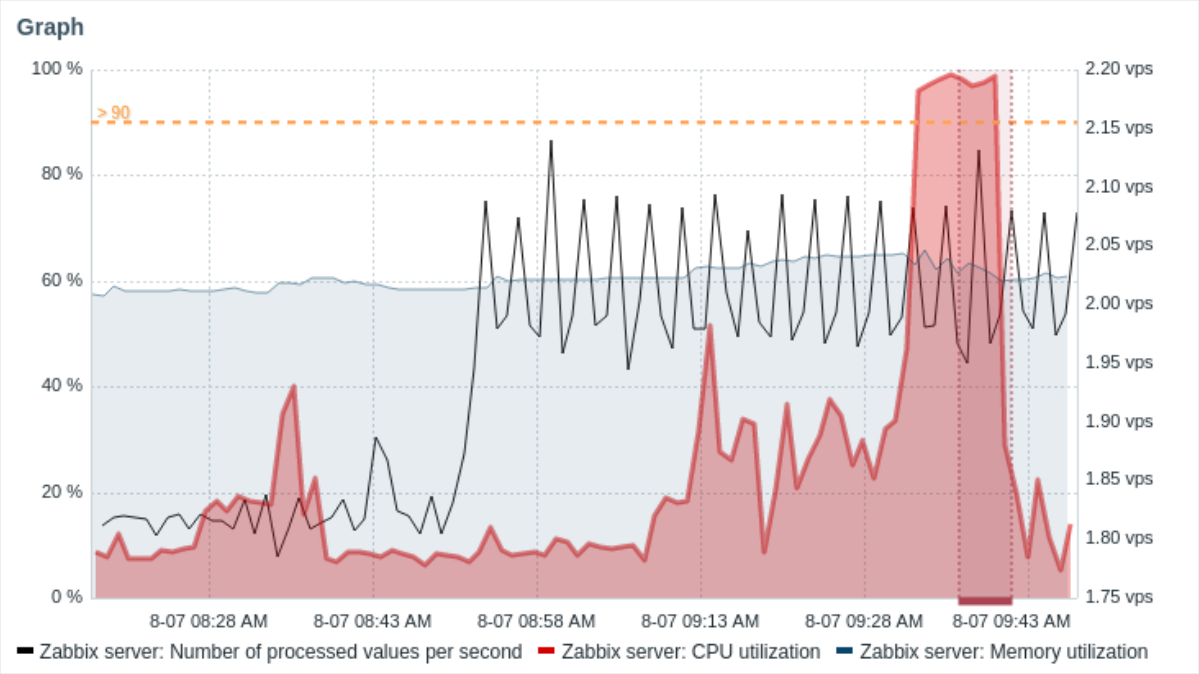
Gdy ustawiony jest *Początkowy widok* lub *Domyślny widok*, możesz w dowolnym momencie wrócić do tego widoku, klikając ikonę domu po lewej stronie.



8 Graph

Przegląd

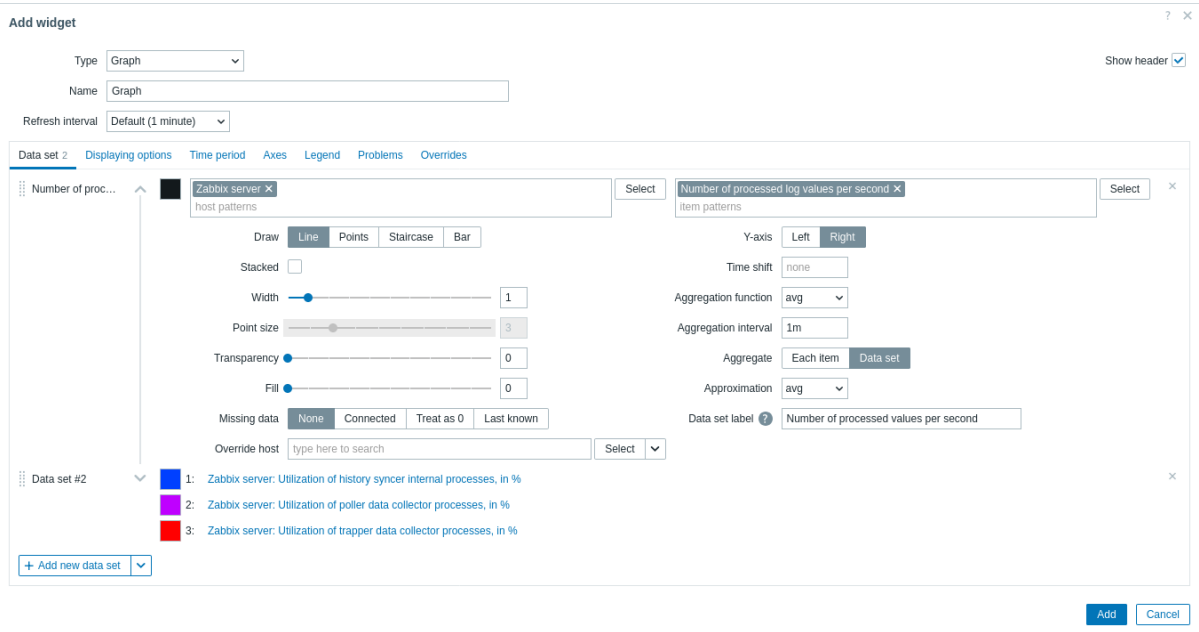
Widżet wykresu zapewnia nowoczesny i wszechstronny sposób wizualizacji danych zbieranych przez Zabbix przy użyciu techniki rysowania obrazu wektorowego. Ten widżet wykresu jest obsługiwany od Zabbix 4.0. Należy pamiętać, że widżet wykresu obsługiwany przed Zabbix 4.0 nadal może być używany jako Graph (classic). Więcej informacji można znaleźć również w sekcji *Dodawanie widżetów* na stronie *Dashboards*.



Informacje wyświetlane w widżecie *Graph* można pobrać jako obraz PNG, wybierając opcję *Download image* w menu widżetu.

Konfiguracja

Aby skonfigurować, wybierz *Graph* jako typ:



Zestaw danych

Karta **Zestaw danych** umożliwia wybór danych dla wykresu przez dodawanie zestawów danych. Można dodać dwa typy zestawów danych:

- *Wzorce pozycji* - wyświetlane są dane z pasujących pozycji. Możesz wybrać jeden kolor bazowy lub wskazać wiersz palety, aby przypisać różne kolory do każdej dopasowanej pozycji.
- *Lista pozycji* - wyświetlane są dane z wybranych pozycji. Możesz indywidualnie wybrać kolor każdej pozycji za pomocą selektora.

Domyślnie dodawany jest zestaw danych typu *Wzorce pozycji*.

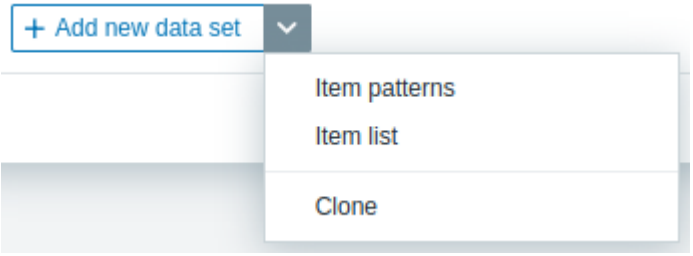
Zestaw danych	Dla zestawu danych **Wzorce pozycji**: Wybierz lub wprowadź wzorce hosta i pozycji; na wykresie zostaną wyświetlone dane pozycji pasujących do tych wzorców; można wyświetlić do 50 pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub więcej znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Symbol wieloznaczny jest zawsze interpretowany, więc nie można na przykład dodać osobno pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (na przykład *item2*, *item3*). Określenie wzorców hosta i pozycji jest obowiązkowe dla zestawów danych typu ”Wzorce pozycji”. Zobacz też: Szczegóły konfiguracji zestawu danych. Dla zestawu danych **Lista pozycji**: Wybierz pozycje dla wykresu, klikając przycisk *Dodaj pozycję*. Możesz także wybrać zgodne widżety jako źródło danych dla pozycji, klikając przycisk *Dodaj widżet*. Określenie pozycji lub widżetów jest obowiązkowe dla zestawów danych typu ”Lista pozycji”. Zobacz też: Szczegóły konfiguracji zestawu danych. Uwaga: dozwolone są tylko numeryczne typy pozycji. Podczas konfigurowania widżetu na panelu szablonu parametr określający wzorce hostów nie jest dostępny, a parametr określający listę pozycji pozwala wybrać tylko pozycje skonfigurowane w szablonie.
Rysuj	Wybierz typ rysowania metryki. Dostępne typy rysowania: *Line* (ustawiony domyślnie), *Points*, *Staircase* i *Bar*. Uwaga: jeśli na wykresie liniowym/schodkowym znajduje się tylko jeden punkt danych, jest on rysowany jako punkt niezależnie od typu rysowania. Rozmiar punktu jest obliczany na podstawie szerokości linii, ale nie może być mniejszy niż 3 piksele, nawet jeśli szerokość linii jest mniejsza.
Skumulowane	Zaznacz pole wyboru, aby wyświetlać dane jako skumulowane (z wypełnionymi obszarami). Ta opcja jest wyłączona, gdy wybrany jest typ rysowania *Points*.
Szerokość	Ustaw szerokość linii. Ta opcja jest dostępna, gdy wybrany jest typ rysowania *Line* lub *Staircase*.
Rozmiar punktu	Ustaw rozmiar punktu. Ta opcja jest dostępna, gdy wybrany jest typ rysowania *Points*.
Przezroczystość	Ustaw poziom przezroczystości.
Wypełnienie	Ustaw poziom wypełnienia. Ta opcja jest dostępna, gdy wybrany jest typ rysowania *Line* lub *Staircase*.
Brakujące dane	Wybierz opcję wyświetlania brakujących danych: **None** - przerwa pozostaje pusta; **Connected** - dwa wartości brzegowe są łączone; **Treat as 0** - brakujące dane są wyświetlane jako wartości 0; **Last known** - brakujące dane są wyświetlane z taką samą wartością jak ostatnia znana wartość; nie dotyczy typu rysowania *Points* i *Bar*.
Zastąp host	Wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.
Oś Y	Wybierz stronę wykresu, po której będzie wyświetlana oś Y.
Przesunięcie czasu	W razie potrzeby określ przesunięcie czasu. W tym polu możesz używać sufiksów czasu. Dozwolone są wartości ujemne.

Funkcja agregacji	Określ, której funkcji agregacji użyć dla każdej pozycji lub dla całego zestawu danych w wybranym *Interwale agregacji*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **sum** - wyświetl sumę wartości; **count** - wyświetl liczbę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość. Jeśli wybrano **not used**, wyświetlane są wszystkie wartości (bez agregacji). Tylko dane numeryczne mogą być wyświetlane dla *min*, *max*, *avg* i *sum*. Dla *count* dane nienumeryczne zostaną przekształcone na numeryczne. Zobacz też: Agregacja na wykresach.
Interwał agregacji	Określ interwał do agregowania wartości. W tym polu możesz używać sufiksów czasu. Wartość numeryczna bez sufiksu będzie traktowana jako sekundy. Uwaga: jeśli widżet jest skonfigurowany do wyświetlania danych historycznych na podstawie trendów (*History data selection* jest ustawione na *Trends* lub *Auto*), zaleca się użycie interwału agregacji będącego wielokrotnością 1 godziny (np. 3600, 60m, 1h, 3h itd.). Trendy przechowują godzinowe wartości zagregowane, więc użycie interwału agregacji, który nie jest wielokrotnością 1 godziny (np. 100s, 7min, 15min, 90min itd.), może prowadzić do wyników trudnych do interpretacji.
Agreguj	Określ, czy agregować: **Each item** - każda pozycja w zestawie danych będzie agregowana i wyświetlana osobno; **Data set** - wszystkie pozycje zestawu danych zostaną zagregowane i wyświetlone jako jedna wartość.
Przybliżenie	Określ, jaką wartość wyświetlać, gdy dla jednego pionowego piksela wykresu istnieje więcej niż jedna wartość: **all** - wyświetl najmniejszą, największą i średnią wartość; **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią. To ustawienie jest przydatne podczas wyświetlania wykresu dla dużego przedziału czasu z częstym interwałem aktualizacji (na przykład rok wartości zbieranych co 10 minut).
Etykieta zestawu danych	Określ etykietę zestawu danych wyświetlaną w konfiguracji wykresu *Zestaw danych* oraz w *Legendzie* wykresu (dla agregowanych zestawów danych). Wszystkie zestawy danych są numerowane, w tym te z określoną *Etykietą zestawu danych*. Jeśli nie określono etykiety, zestaw danych zostanie automatycznie oznaczony zgodnie ze swoim numerem (np. ”Data set #2”, ”Data set #3” itd.). Numeracja zestawów danych jest przeliczana po zmianie kolejności/przeciąganiu zestawów danych. Zbyt długie etykiety zestawów danych zostaną skrócone, aby zmieściły się w miejscu wyświetlania (np. ”Number of proc...”).

Szczegóły konfiguracji zestawu danych

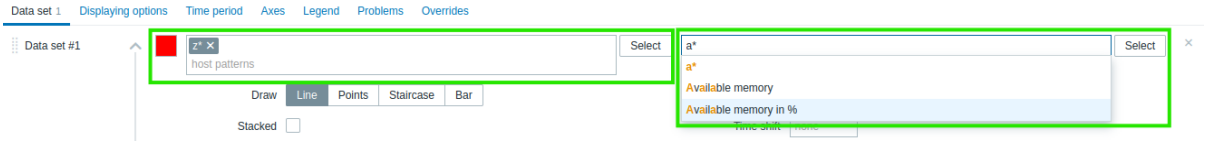
Istniejące zestawy danych są wyświetlane na liście. Możesz:

- Kliknąć ikonę przenoszenia i przeciągnąć zestaw danych w nowe miejsce na liście.
- Kliknąć ikonę rozwijania, aby rozwinąć szczegóły zestawu danych. Po rozwinięciu ikona ta zmienia się w ikonę zwijania.
- Kliknąć ikonę koloru, aby otworzyć selektor. Możesz wprowadzić kod szesnastkowy, wybrać jednolity próbnik albo przejść do karty *Palette* i wybrać wiersz predefiniowanych kolorów. Wybrany kolor jest stosowany bezpośrednio dla zestawów danych *Item list* lub jako baza dla generowanych odcieni w *Item patterns*. Użyj klawisza Tab, aby przechodzić między kontrolkami okna dialogowego, klawiszy strzałek, aby nawigować między próbnikami lub wierszami palety, Enter, aby wybrać, oraz Esc, aby anulować.
- Kliknąć przycisk *Add new data set*, aby dodać pusty zestaw danych umożliwiający wybór wzorców hosta i pozycji. Jeśli klikniesz ikonę skierowaną w dół obok przycisku *Add new data set*, pojawi się menu rozwijane, które umożliwia dodanie nowego zestawu danych *Item patterns* lub *Item list* albo sklonowanie aktualnie otwartego zestawu danych (*Clone*). Jeśli wszystkie zestawy danych są zwinięte, opcja *Clone* jest niedostępna.

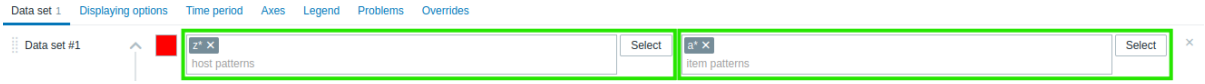


Zestaw danych **Wzorce pozycji** zawiera pola *Wzorce hostów* i *Wzorce pozycji*, które rozpoznają zarówno pełne nazwy, jak i wzorce zawierające symbol wieloznaczny (*). Ta funkcjonalność pozwala wybrać wszystkie nazwy hostów i nazwy pozycji zawierające wybrany wzorzec. Podczas wpisywania nazwy pozycji lub wzorca pozycji w polu *Wzorce pozycji* na liście rozwijanej wyświetlane są tylko pozycje należące do wybranych nazw hostów.

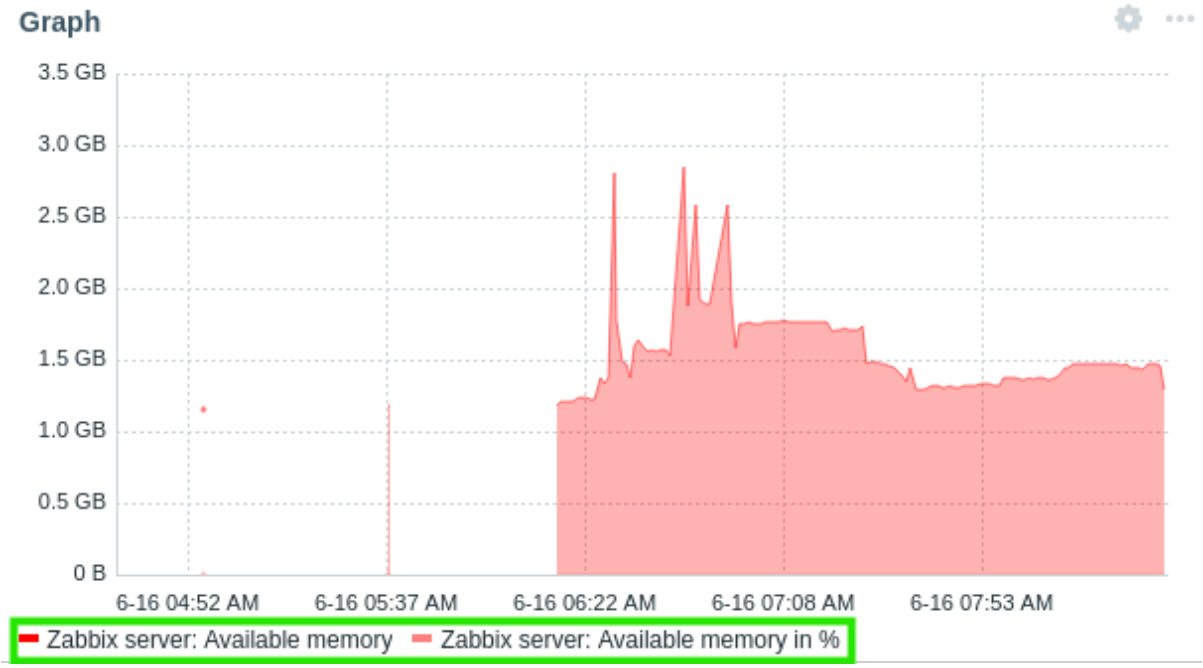
Na przykład po wpisaniu wzorca **z*** w polu *Wzorce hostów* lista rozwijana wyświetla wszystkie nazwy hostów zawierające ten wzorzec: **z*, Zabbix server, Zabbix proxy**. Po naciśnięciu *Enter* wzorzec zostaje zaakceptowany i jest wyświetlany jako **z***. Podobnie po wpisaniu wzorca **a*** w polu *Wzorce pozycji* lista rozwijana wyświetla wszystkie nazwy pozycji zawierające ten wzorzec: **a*, Available memory, Available memory in %**.



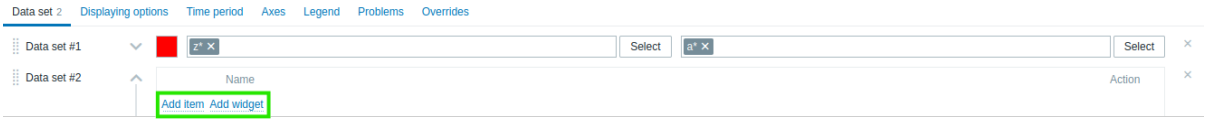
Po naciśnięciu *Enter* wzorzec zostaje zaakceptowany i wyświetlony jako **a***.



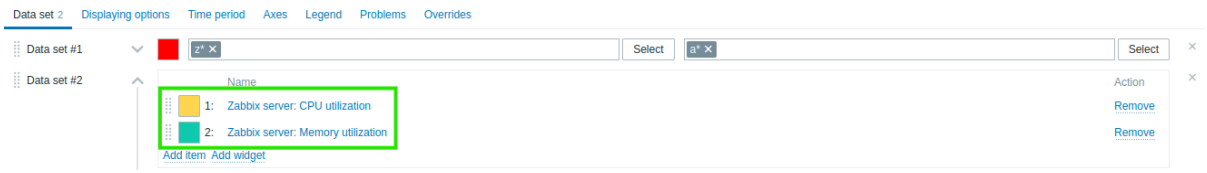
Wykres wyświetla następnie wszystkie pozycje należące do wybranych nazw hostów.



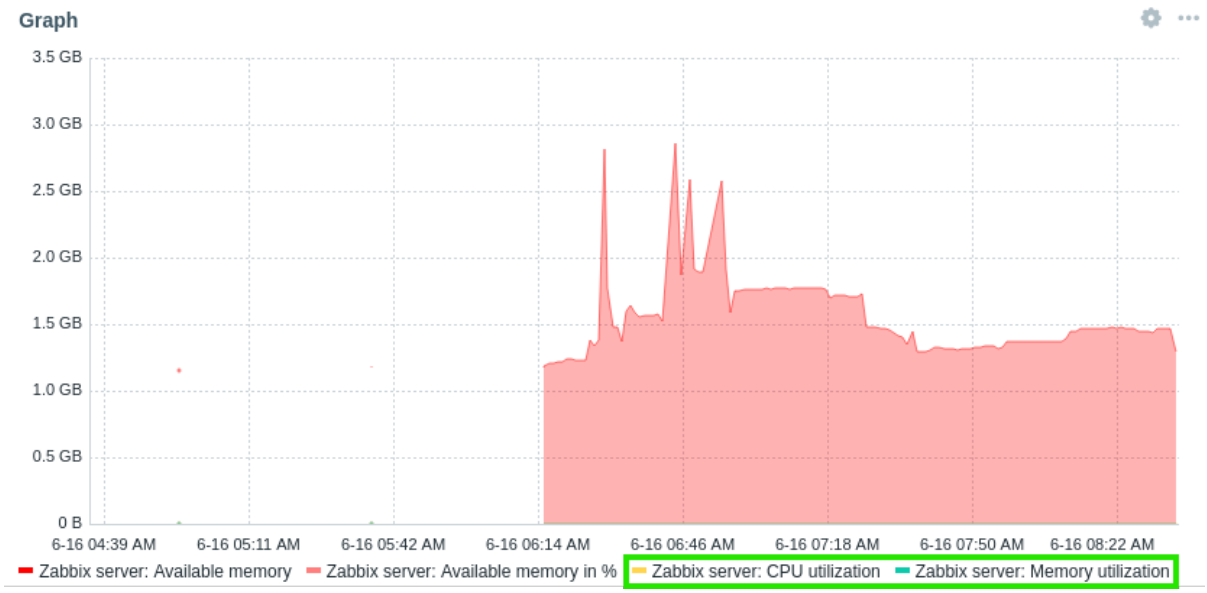
Zestaw danych **Item list** zawiera przycisk *Add item*, który umożliwia dodawanie pozycji wyświetlanych na wykresie. Możesz również dodać zgodne widżety jako źródło danych dla pozycji, klikając przycisk *Add widget*.



Na przykład kliknięcie przycisku *Add item* otwiera okno podręczne zawierające parametr *Host*. Po wybraniu hosta wszystkie jego pozycje dostępne do wyboru są wyświetlane na liście.



Po wybraniu jednej lub kilku pozycji będą one wyświetlane na liście pozycji zestawu danych oraz na wykresie.

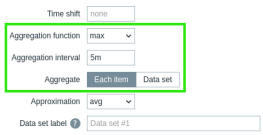


Agregacja na wykresach

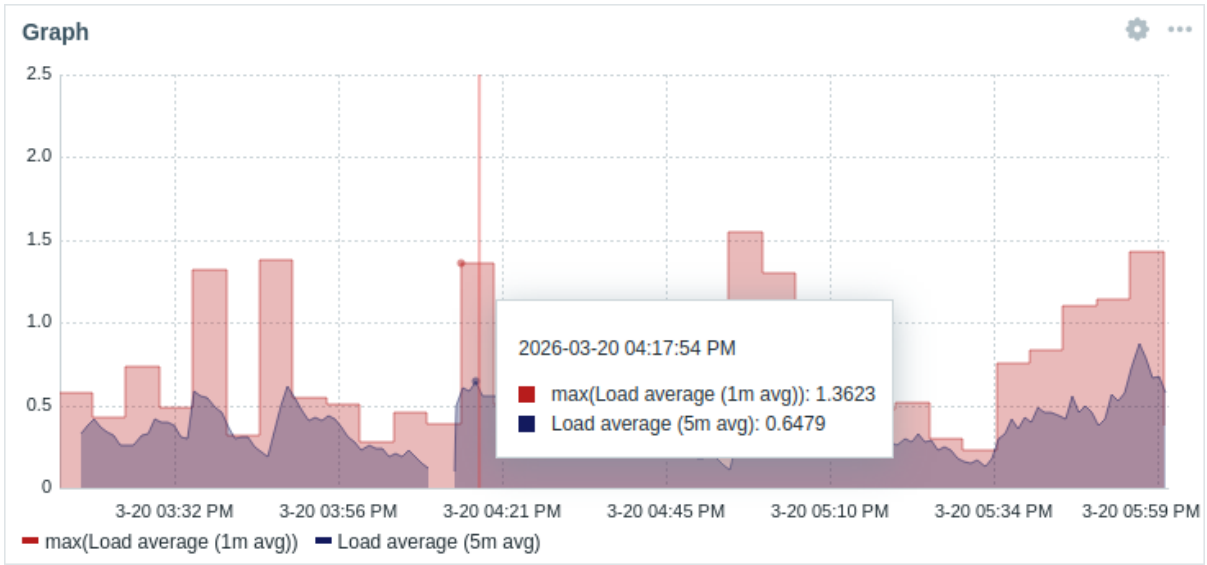
Agregacja umożliwia wyświetlanie wartości zagregowanej (minimum, maksimum, średnia itp.) w określonym przedziale czasu (5 minut, godzina, dzień itp.) zamiast wszystkich pojedynczych wartości. W obrębie zestawu danych agregację można zastosować osobno do każdej pozycji albo do całego zestawu danych.

Przykładowe zastosowania funkcji agregacji: - Średnia liczba żądań do serwera Nginx na dzień. - Najmniejsza ilość wolnego miejsca na dysku w klastrach na tydzień.

W poniższym przykładzie do pozycji zastosowano funkcję agregacji ”max” z interwałem agregacji wynoszącym pięć minut:



Przykład pokazuje maksymalne obciążenie CPU w ciągu pięciu minut (zagregowane z jednominutowych wartości średnich) w porównaniu ze średnim obciążeniem CPU w ciągu pięciu minut.

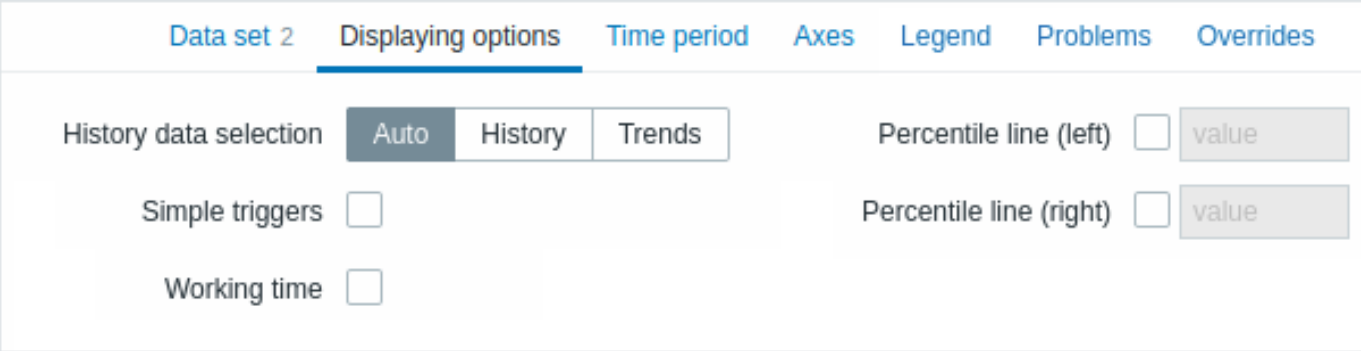


Gdy dla legendy wybrana jest opcja *Pokaż funkcję agregacji*, pozycje są wyświetlane w nawiasach, z prefiksem funkcji agregacji w legendzie i podpowiedzi. Jeśli dla wykresu określono *Etykietę zestawu danych*, etykieta jest wyświetlana w nawiasach, z prefiksem

funkcji agregacji.

Wyświetlanie opcji

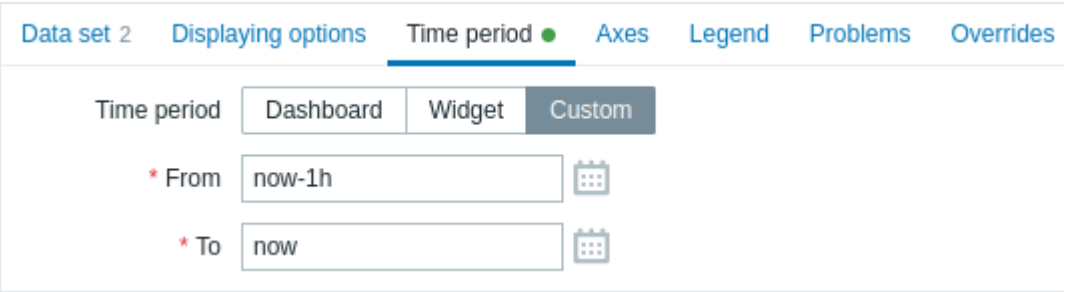
Karta **Wyświetlanie opcji** umożliwia zdefiniowanie wyboru danych historycznych:



Wybór danych historycznych	Ustaw źródło danych wykresu: **Auto** - dane są pobierane zgodnie z klasycznym algorytmem wykresu (domyślnie); **History** - dane z historii; **Trends** - dane z trendów.
Proste wyzwalacze	Zaznacz pole wyboru, aby wyświetlić progi wyzwalaczy dla prostych wyzwalaczy. Progi będą rysowane jako linie przerywane w kolorze odpowiadającym ważności wyzwalacza. Prosty wyzwalacz to wyzwalacz z jedną funkcją (tylko `last`, `max`, `min`, `avg`) dla jednej pozycji w wyrażeniu. Można narysować maksymalnie trzy wyzwalacze. Należy pamiętać, że wyzwalacz musi mieścić się w rysowanym zakresie, aby był widoczny.
Czas pracy	Zaznacz pole wyboru, aby wyświetlić czas pracy na wykresie. Czas pracy (dni robocze) jest wyświetlany na wykresach jako białe tło, natomiast czas poza pracą jest wyświetlany na szaro (przy domyślnym motywie frontend *Original blue*).
Linia percentyla (lewa)	Zaznacz pole wyboru i wprowadź wartość percentyla, aby wyświetlić określony percentyl jako linię na lewej osi Y wykresu. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie znajdować się na poziomie, poniżej którego leży 95 procent wartości.
Linia percentyla (prawa)	Zaznacz pole wyboru i wprowadź wartość percentyla, aby wyświetlić określony percentyl jako linię na prawej osi Y wykresu. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie znajdować się na poziomie, poniżej którego leży 95 procent wartości.

Okres czasu

Karta **Okres czasu** umożliwia ustawienie okresu czasu, dla którego dane mają być wyświetlane na wykresie:

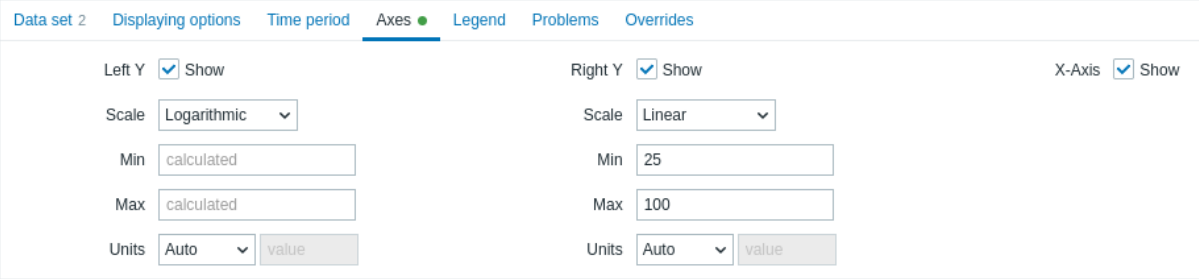


Okres czasu	Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Okres czasu* jako źródło danych; **Widget** - ustaw zgodny widget określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Widget”.

From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na "Custom".

Osie

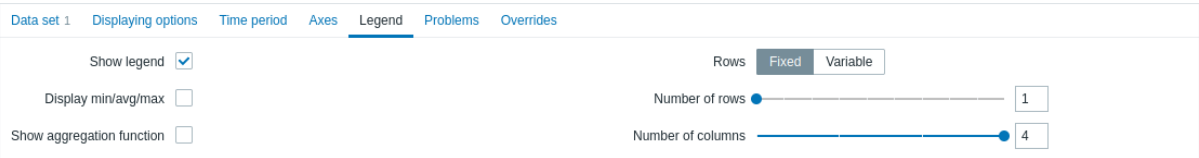
Zakładka **Osie** umożliwia dostosowanie sposobu wyświetlania osi:



Lewa Y	Zaznacz to pole wyboru, aby lewa oś Y była widoczna. Pole wyboru może być nieaktywne, jeśli opcja nie została zaznaczona ani w zakładce *Zbiór danych*, ani w zakładce *Nadpisania*.
Prawa Y	Zaznacz to pole wyboru, aby prawa oś Y była widoczna. Pole wyboru może być nieaktywne, jeśli opcja nie została zaznaczona ani w zakładce *Zbiór danych*, ani w zakładce *Nadpisania*.
Oś X	Odznacz to pole wyboru, aby ukryć oś X (domyślnie zaznaczone).
Skala	Wybierz skalę wartości osi wykresu z listy rozwijanej: **Liniowa** - wartości osi rosną o stałą wartość (np. 10, 20, 30), odpowiednia dla danych, które zmieniają się stopniowo lub obejmują mały albo średni zakres; **Logarytmiczna** - wartości osi rosną wykładniczo (np. 10, 100, 1000), odpowiednia dla danych, które zmieniają się szybko lub obejmują duży zakres.
Min	Ustaw minimalną wartość odpowiedniej osi. Określana jest minimalna wartość widocznego zakresu osi Y.
Max	Ustaw maksymalną wartość odpowiedniej osi. Określana jest maksymalna wartość widocznego zakresu osi Y.
Jednostki	Wybierz jednostkę dla wartości osi wykresu z listy rozwijanej: **Auto** - wartości osi są wyświetlane z użyciem jednostki pierwszej pozycji w zbiorze danych; **Statyczne** - wartości osi są wyświetlane z użyciem jednostki określonej w polu wejściowym *value*; jeśli pole pozostanie puste, wyświetlane będą tylko wartości liczbowe.

Legenda

Zakładka **Legenda** umożliwia dostosowanie legendy wykresu:

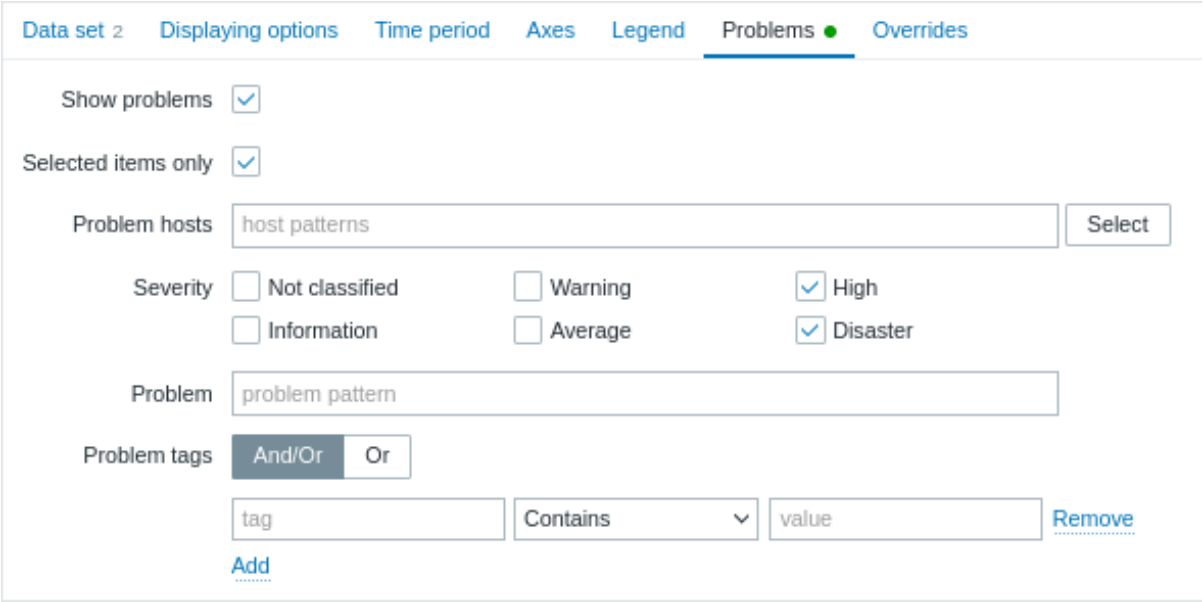


Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresie (domyślnie zaznaczone).
Wyświetl min/śr./maks.	Zaznacz to pole wyboru, aby wyświetlić w legendzie minimalne, średnie i maksymalne wartości pozycji.
Pokaż funkcję agregacji	Zaznacz to pole wyboru, aby wyświetlić w legendzie funkcję agregacji.
Wiersze	Wybierz tryb wyświetlania wierszy legendy: **Stałe** — liczba wyświetlanych wierszy jest określana przez wartość parametru *Liczba wierszy*; **Zmienne** — liczba wyświetlanych wierszy jest określana przez liczbę skonfigurowanych pozycji, bez przekraczania wartości parametru *Maksymalna liczba wierszy*.

Liczba wierszy/ Maksymalna liczba wierszy	Jeśli dla parametru *Wiersze* ustawiono wartość „Stałe”, ustaw liczbę wyświetlanych wierszy legendy (1-10). Jeśli dla parametru *Wiersze* ustawiono wartość „Zmienne”, ustaw maksymalną liczbę wyświetlanych wierszy legendy (1-10).
Liczba kolumn	Ustaw liczbę wyświetlanych kolumn legendy (1-4). Ten parametr jest dostępny, jeśli opcja *Wyświetl min/śr./maks.* jest odznaczona.

Problemy

Karta **Problems** umożliwia dostosowanie wyświetlania problemów:

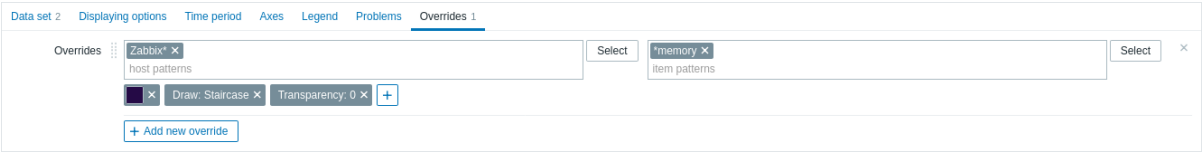


Show problems	Zaznacz to pole wyboru, aby włączyć wyświetlanie problemów na wykresie (domyślnie niezaznaczone, czyli wyłączone).
Selected items only	Zaznacz to pole wyboru, aby na wykresie wyświetlać tylko problemy dla wybranych pozycji.
Problem hosts	Wybierz hosty z problemami, które mają być wyświetlane na wykresie. Można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub więcej znaków). Aby określić wzorzec wieloznaczny, po prostu wpisz ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania zwróć uwagę, że wszystkie pasujące hosty są wyświetlane na liście rozwijanej. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Severity	Zaznacz poziomy ważności problemów, aby filtrować problemy wyświetlane na wykresie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlone zostaną wszystkie problemy.
Problem	Określ nazwę problemu, która ma być wyświetlana na wykresie.

Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek.

Nadpisywanie

Karta **Nadpisywanie** umożliwia dodawanie niestandardowych nadpisań dla zestawów danych:



Nadpisania są przydatne, gdy dla zestawu danych wybrano kilka pozycji za pomocą symbolu wieloznacznego ∗ i chcesz zmienić sposób, w jaki pozycje są domyślnie wyświetlane (np. domyślny kolor bazowy lub dowolną inną właściwość).

Istniejące nadpisania, jeśli są dostępne, są wyświetlane na liście. Aby dodać nowe nadpisanie:

- Kliknij przycisk 

- Wybierz hosty i pozycje dla nadpisania. Alternatywnie możesz wprowadzić wzorce hostów i pozycji. Można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec wieloznaczny, po prostu wpisz ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania zwróć uwagę, że wszystkie pasujące hosty są wyświetlane na liście rozwijanej. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie jest możliwe dodanie na przykład pozycji o nazwie ”item*” pojedynczo, jeśli istnieją inne pasujące pozycje (np. item2, item3). Parametry wzorca hosta i wzorca pozycji są obowiązkowe. Parametr służący do określania wzorców hostów nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu. Parametr służący do określania listy pozycji umożliwia wybranie tylko pozycji skonfigurowanych w szablonie podczas konfigurowania widżetu na pulpicie szablonu.
- Kliknij 

, aby wybrać parametry nadpisania. Należy wybrać co najmniej jeden parametr nadpisania. Opisy parametrów znajdują się w karcie *Zestaw danych* powyżej.

9 Graph (classic)

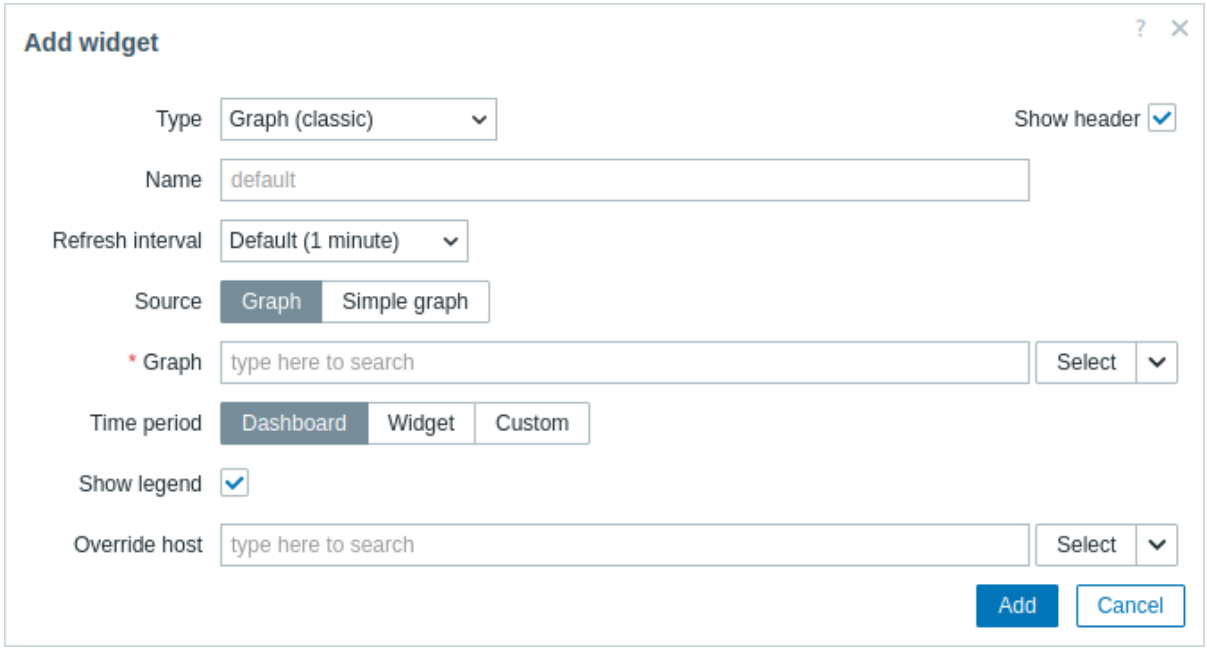
Przegląd

W widżecie klasycznego wykresu można wyświetlić pojedynczy wykres niestandardowy lub prosty wykres.

Informacje wyświetlane w widżecie *Wykres (klasyczny)* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu widżetu.

Konfiguracja

Aby skonfigurować, wybierz *Graph (classic)* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Source	Wybierz typ wykresu: **Graph** - custom graph; **Simple graph** - simple graph.
Graph	Wybierz niestandardowy wykres do wyświetlenia. Alternatywnie wybierz zgodny widżet jako źródło danych dla wykresów. Ten parametr jest dostępny, jeśli *Source* ma wartość "Graph".
Item	Wybierz pozycję do wyświetlenia na prostym wykresie. Alternatywnie wybierz zgodny widżet jako źródło danych dla pozycji. Ten parametr jest dostępny, jeśli *Source* ma wartość "Simple graph".
Time period	Ustaw okres czasu, dla którego mają być wyświetlane dane na wykresie. Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Time period* jako źródło danych; **Widget** - ustaw zgodny widżet określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem.
Widget	Wprowadź lub wybierz zgodny widżet jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* ma wartość "Widget".
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".
Show legend	Odznacz to pole wyboru, aby ukryć legendę na wykresie (domyślnie zaznaczone).
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.

Legenda wykresu składa się z trzech sekcji:

- Pozycje i ich zagregowane wartości
- Percentyle (jeśli skonfigurowano)
- Wyzwalacze (jeśli jakiekolwiek są powiązane z wyświetlanymi pozycjami)

Jeśli wysokość wykresu w obrębie widżetu jest niewystarczająca, legenda może nie być wyświetlana lub może być wyświetlana tylko częściowo. Najpierw ukrywane są wyzwalacze i percentyle, a następnie legenda pozycji. Aby wyświetlić pełną legendę, zwiększ pionowy rozmiar widżetu.

Note:
Można wyświetlić nie więcej niż 3 linie wyzwalaczy. Jeśli jest więcej wyzwalaczy, do wyświetlenia priorytetowo traktowane są wyzwalacze o niższym poziomie istotności.

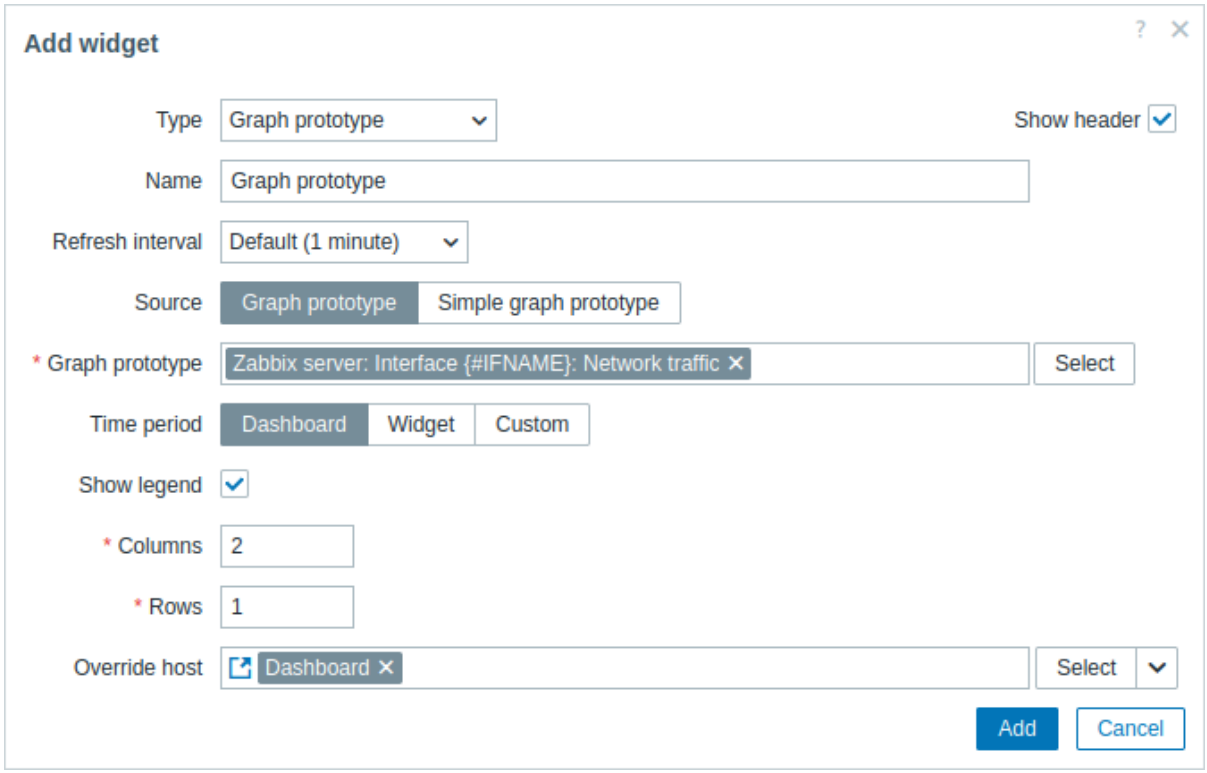
10 Graph prototype

Przegląd

W widżecie prototypu wykresu można wyświetlić siatkę wykresów utworzonych na podstawie prototypu wykresu lub prototypu pozycji przez wykrywanie niskiego poziomu.

Konfiguracja

Aby skonfigurować, wybierz *Graph prototype* jako typ widżetu:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Source	Wybierz źródło wykresów: **Graph prototype** lub **Simple graph prototype**.
Graph prototype	Wybierz prototyp wykresu, aby wyświetlić wykresy wykryte przez prototyp wykresu. Ten parametr jest dostępny, jeśli *Source* ma wartość "Graph prototype".
Item prototype	Wybierz prototyp pozycji, aby wyświetlić proste wykresy dla pozycji wykrytych przez prototyp pozycji. Ten parametr jest dostępny, jeśli *Source* ma wartość "Simple graph prototype".
Time period	Ustaw okres czasu, dla którego mają być wyświetlane dane na wykresach. Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Time period* jako źródło danych; **Widget** - ustaw zgodny widżet określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli ustawione, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem myszy.
Widget	Wprowadź lub wybierz zgodny widżet jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* ma wartość "Widget".
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".

Show legend	Odznacz to pole wyboru, aby ukryć legendę na wykresach (domyślnie zaznaczone).
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Columns	Wprowadź liczbę kolumn wykresów do wyświetlenia w widżecie prototypu wykresu.
Rows	Wprowadź liczbę wierszy wykresów do wyświetlenia w widżecie prototypu wykresu.

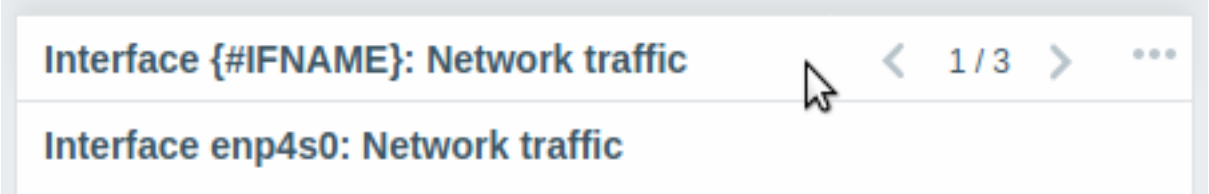
Legenda wykresu składa się z trzech sekcji:

- Pozycje i ich zagregowane wartości
- Percentyle (jeśli skonfigurowano)
- Wyzwalacze (jeśli są powiązane z wyświetlanymi pozycjami)

Jeśli wysokość wykresu w widżecie jest niewystarczająca, legenda może nie być wyświetlana lub może być wyświetlana tylko częściowo. Najpierw ukrywane są wyzwalacze i percentyle, a następnie legenda pozycji. Aby wyświetlić pełną legendę, zwiększ pionowy rozmiar widżetu.

Note:
Można wyświetlić nie więcej niż 3 linie wyzwalaczy. Jeśli wyzwalaczy jest więcej, do wyświetlenia są priorytetowo wybierane wyzwalacze o niższym poziomie ważności.

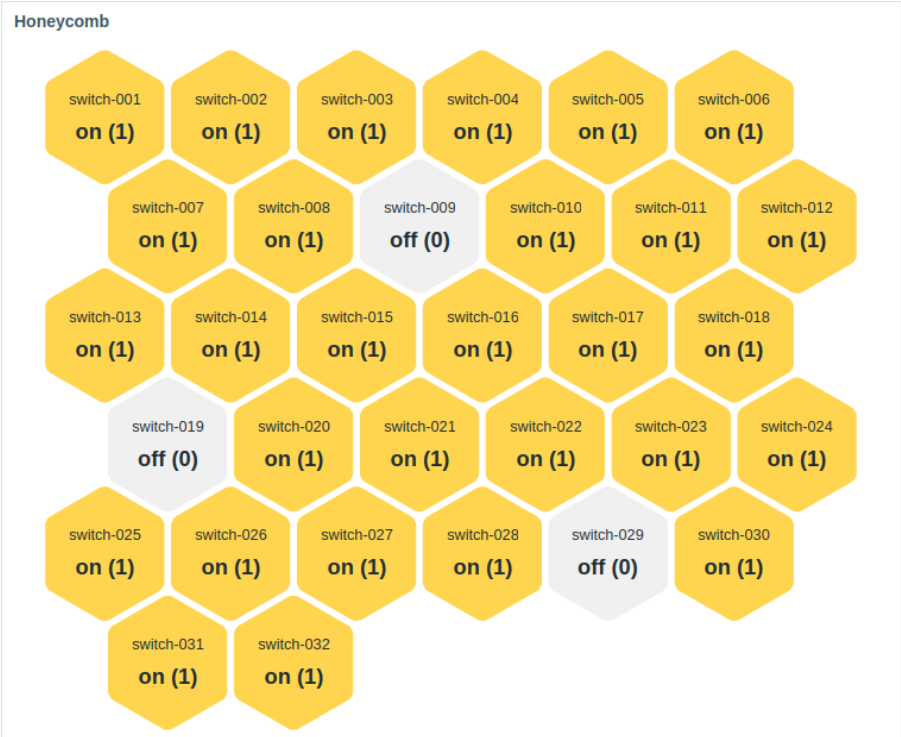
Chociaż parametry *Columns* i *Rows* umożliwiają dopasowanie więcej niż jednego wykresu w widżecie, nadal może istnieć więcej wykrytych wykresów niż kolumn/wierszy w widżecie. W takim przypadku w widżecie staje się dostępne stronicowanie, a wysuwany nagłówek umożliwia przełączanie między stronami za pomocą strzałek w lewo i w prawo:



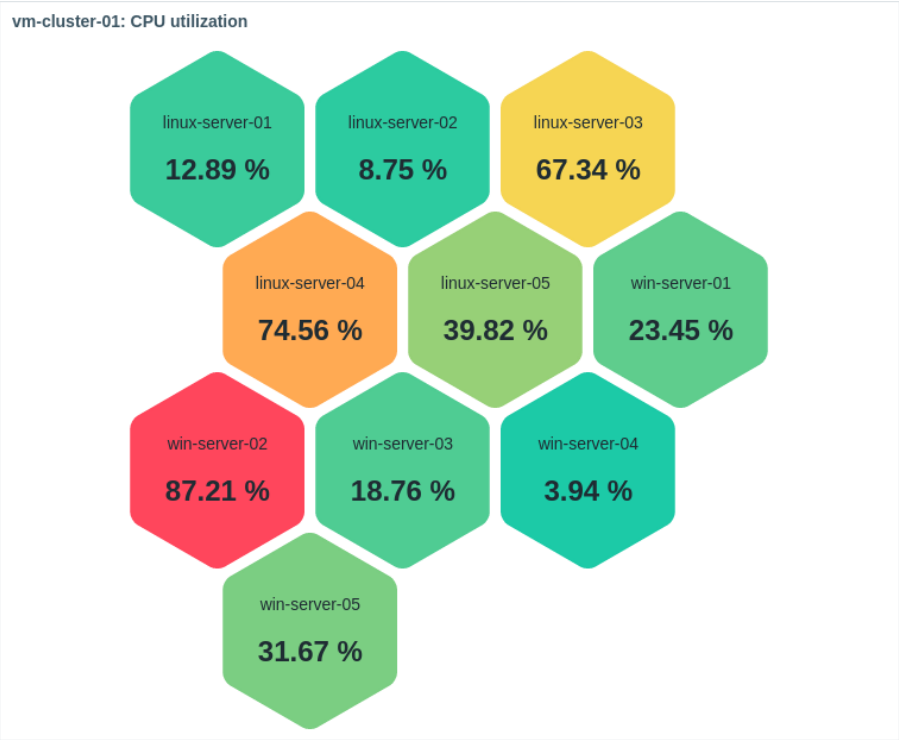
11 Honeycomb

Omówienie

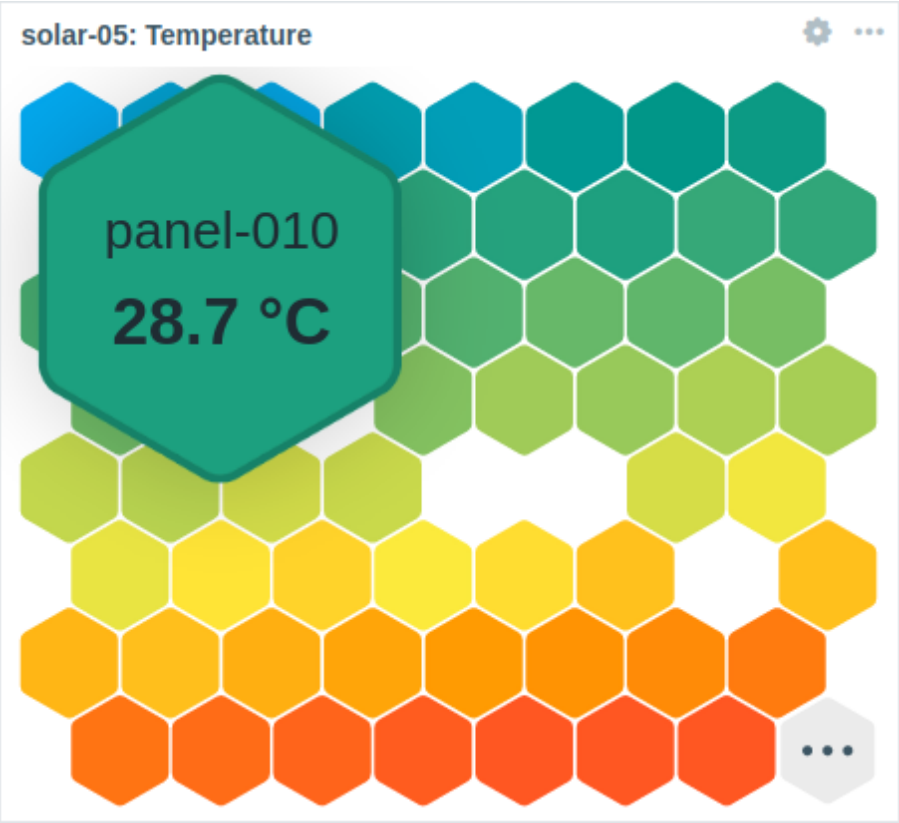
Widżet honeycomb oferuje dynamiczny przegląd monitorowanej infrastruktury sieciowej i zasobów, w którym grupy hostów, takie jak maszyny wirtualne i urządzenia sieciowe, wraz z odpowiadającymi im pozycjami, są wizualnie ևեղկայացուցիչ jako interaktywne sześciokątne komórki.



Widżet można wizualnie dostroić za pomocą opcji zaawansowanej konfiguracji, aby uzyskać szeroką gamę stylów wizualnych.



Po najechaniu kursorem aktywna komórka honeycomb powiększa się, aby poprawić jej widoczność. Kliknięcie komórki podświetla jej obramowanie, dopóki nie zostanie wybrana inna komórka.



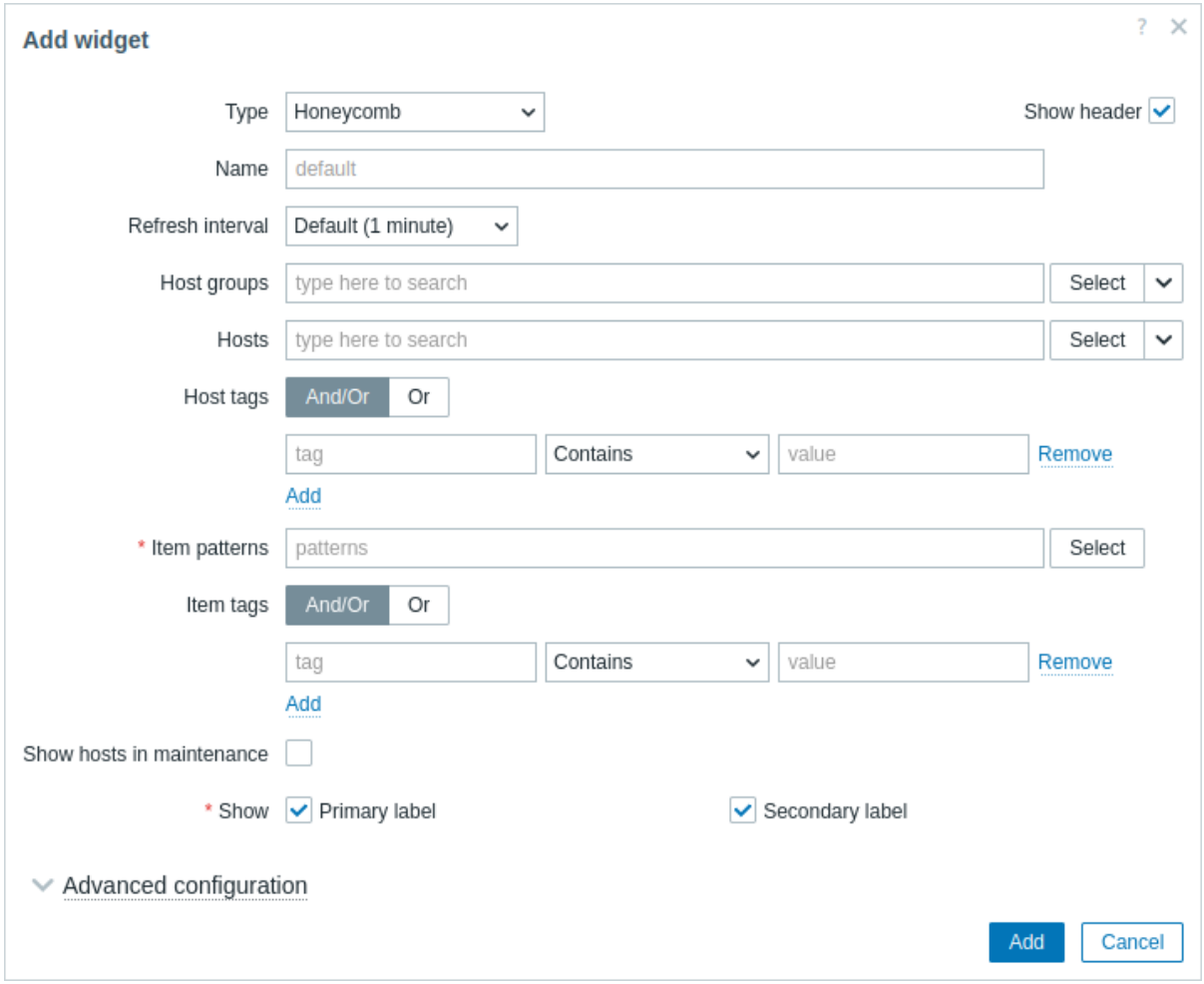
Liczba wyświetlanych komórek honeycomb jest ograniczona przez rozmiar widżetu i minimalny rozmiar komórki (32 piksele). Jeśli wszystkie komórki nie mieszczą się w widżecie, jako ostatnia komórka wyświetlane są wielokropki.

Widżet można zmienić rozmiar, aby zmieścić więcej komórek. Podczas zmiany rozmiaru rozmiar i położenie komórek honeycomb są dynamicznie dostosowywane. Każdy wiersz w honeycomb zachowuje jednakową liczbę komórek, z wyjątkiem ostatniego wiersza, jeśli łączna liczba komórek nie jest podzielna przez liczbę komórek w wierszu.

Informacje wyświetlane w widżecie *Honeycomb* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu widżetu.

Konfiguracja

Aby skonfigurować, wybierz *Honeycomb* jako typ:



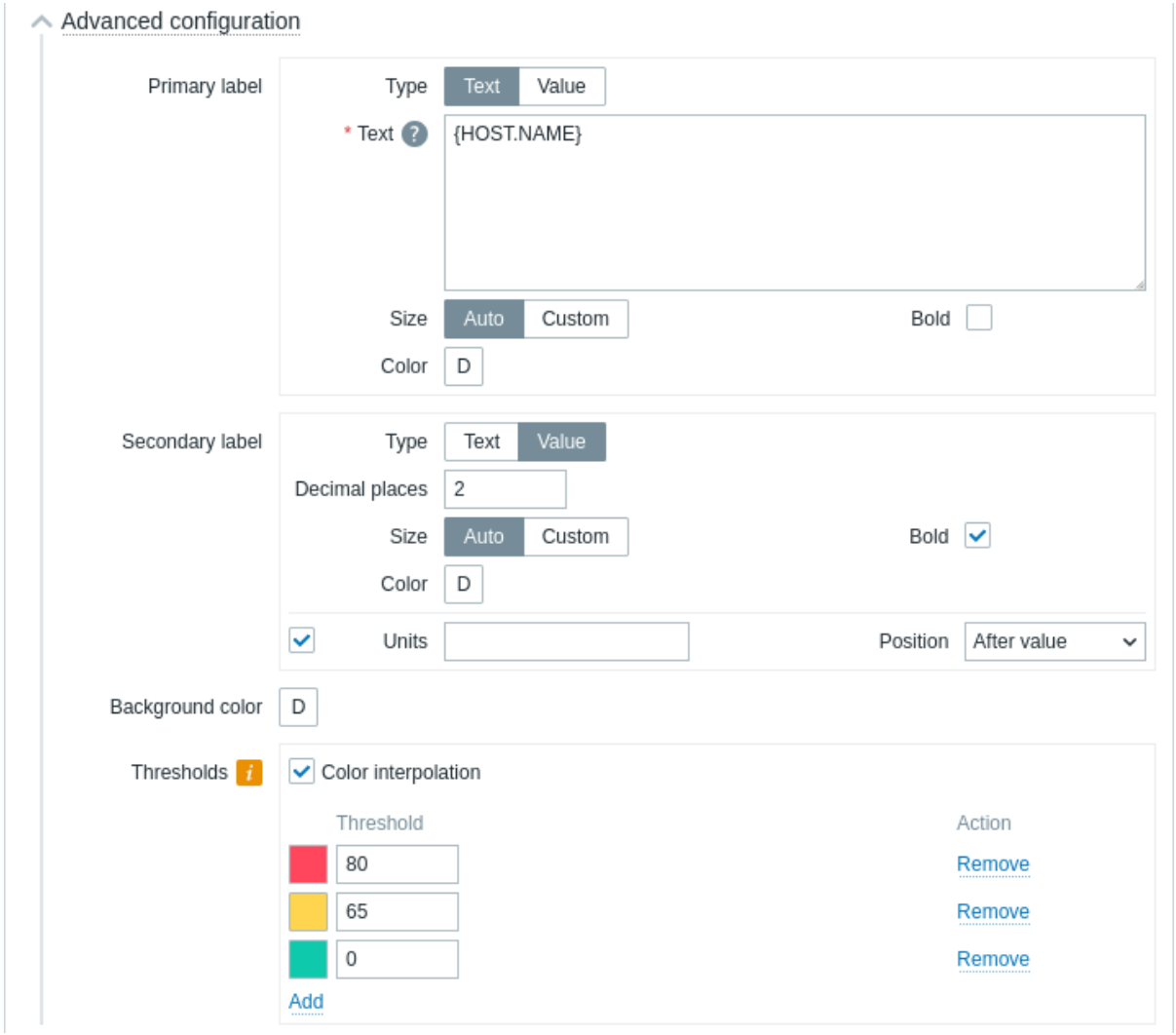
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Wybór nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli wszystkie grupy hostów zawierające hosty z pozycjami pasującymi do wybranego *Wzorca pozycji* (patrz poniżej). Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonie pulpitu.
Hosty	Wybierz hosty. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli wszystkie hosty z pozycjami pasującymi do wybranego *Wzorca pozycji* (patrz poniżej). Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonie pulpitu.

Tagi hostów	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Można uwzględniać oraz wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy tagów i wartości (uwzględnia wielkość liter); **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie fragmentu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy tagów i wartości (uwzględnia wielkość liter); **Nie zawiera** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie fragmentu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu zostaną zgrupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonie pulpitu.
Wzorce pozycji	Wprowadź wzorce pozycji lub wybierz istniejące pozycje jako wzorce pozycji. Dane z pozycji pasujących do wprowadzonych lub wybranych wzorców będą wyświetlane na honeycomb. Parametr *Wzorce pozycji* jest obowiązkowy. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci pozycje pasujące do zera lub więcej znaków; `Zabbix*` zwróci pozycje zaczynające się od ”Zabbix”). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania zostanie wyświetlona lista rozwijana z pasującymi pozycjami, ograniczona do tych należących do wybranych *Hostów* lub hostów w wybranych *Grupach hostów*, jeśli takie zostały wybrane. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać na przykład pozycji o nazwie *item** indywidualnie, jeśli istnieją inne pasujące pozycje (np. item2, item3). Podczas konfigurowania widżetu na szablonie pulpitu ten parametr umożliwia wybór tylko pozycji skonfigurowanych na szablonie.
Tagi pozycji	Określ tagi, aby ograniczyć liczbę pozycji wyświetlanych w widżecie. Więcej informacji można znaleźć powyżej w sekcji *Tagi hostów*.
Pokaż hosty w utrzymaniu	Zaznacz to pole wyboru, aby wyświetlać hosty w utrzymaniu (w takim przypadku obok nazwy hosta zostanie wyświetlona ikona utrzymania). Ten parametr jest oznaczony jako *Pokaż dane w utrzymaniu* podczas konfigurowania widżetu na szablonie pulpitu.
Pokaż	Zaznacz to pole wyboru, aby wyświetlić odpowiedni element komórki honeycomb - etykietę główną, etykietę pomocniczą. Co najmniej jeden element musi być wybrany.
Zaawansowana konfiguracja	Kliknij nagłówek *Zaawansowana konfiguracja*, aby wyświetlić opcje zaawansowanej konfiguracji.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana* i tylko dla elementów wybranych w polu *Pokaż* (zobacz wyżej) oraz dla koloru tła lub progów komórek plastra miodu.



Etykieta podstawowa/drugorzędna

Typ — Wybierz typ etykiety:
Tekst — etykieta będzie wyświetlać tekst określony dla parametru *Tekst*;
Wartość — etykieta będzie wyświetlać wartość pozycji z liczbą miejsc po przecinku określoną dla parametru *Miejsca po przecinku*.

Tekst — Wprowadź tekst etykiety. Ten tekst może zastąpić domyślną nazwę pozycji.
Obsługiwany jest tekst wielowierszowy. Możliwe jest połączenie tekstu i obsługiwanych makr.
Obsługiwane są `{HOST.\*}`, `{ITEM.\*}`, `{INVENTORY.\*}` oraz makra użytkownika.
Komórki plastra miodu są porządkowane alfabetycznie według nazwy hosta, a w obrębie każdego hosta — według nazwy pozycji.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Tekst*.

Miejsca po przecinku — Wprowadź liczbę miejsc po przecinku wyświetlanych wraz z wartością.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość*, i ma wpływ tylko na pozycje zwracające dane numeryczne (float).

Rozmiar — Wybierz rozmiar etykiety:
Automatycznie — użyj automatycznie dopasowanego rozmiaru etykiety;
Niestandardowy — wprowadź niestandardowy rozmiar etykiety (w procentach, względem rozmiaru komórki plastra miodu).
Zwróć uwagę, że etykiety, które nie mieszczą się w rozmiarze komórki plastra miodu, są obcinane.

Pogrubienie — Zaznacz pole wyboru, aby wyświetlać etykiety pogrubioną czcionką.

Kolor — Wybierz kolor etykiety z próbnika kolorów.
„D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Jednostki

Jednostki — Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji.
Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostki ustawione w konfiguracji pozycji.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość*.

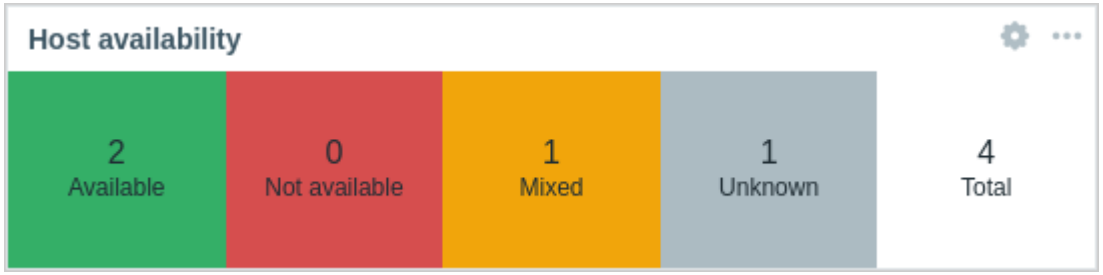
Pozycja	Wybierz położenie jednostek pozycji (przed lub po wartości pozycji). Ten parametr jest ignorowany dla następujących jednostek związanych z czasem: unixtime, uptime, s. Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość* i wybrano opcję *Jednostki*.
Kolor tła	
Kolor tła	Wybierz kolor tła komórek plastra miodu z próbnika kolorów. „D" oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Progi	
Interpolacja kolorów	Zaznacz pole wyboru, aby włączyć płynne przejścia między kolorami progów dla komórek plastra miodu. Ten parametr jest dostępny, jeśli ustawiono co najmniej dwa progi.
Próg	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość liczbową. Lista progów zostanie posortowana rosnąco podczas zapisywania. Zwróć uwagę, że kolory skonfigurowane jako progi będą wyświetlane poprawnie tylko dla pozycji numerycznych. Obsługiwane są sufiksy (na przykład „1d", „2w", „4K", „8G"). Obsługiwane są mapowania wartości.

12 Dostępność hosta

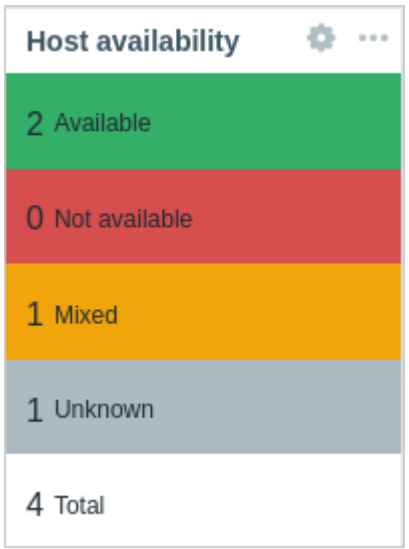
Omówienie

W widżecie dostępności hosta statystyki wysokiego poziomu dotyczące dostępności hostów są wyświetlane w kolorowych kolumnach/wierszach, w zależności od wybranego układu.

Wyświetlanie poziome (kolumny):



Wyświetlanie pionowe (wiersze):



Dostępność hostów w każdej kolumnie/wierszu jest liczona w następujący sposób:

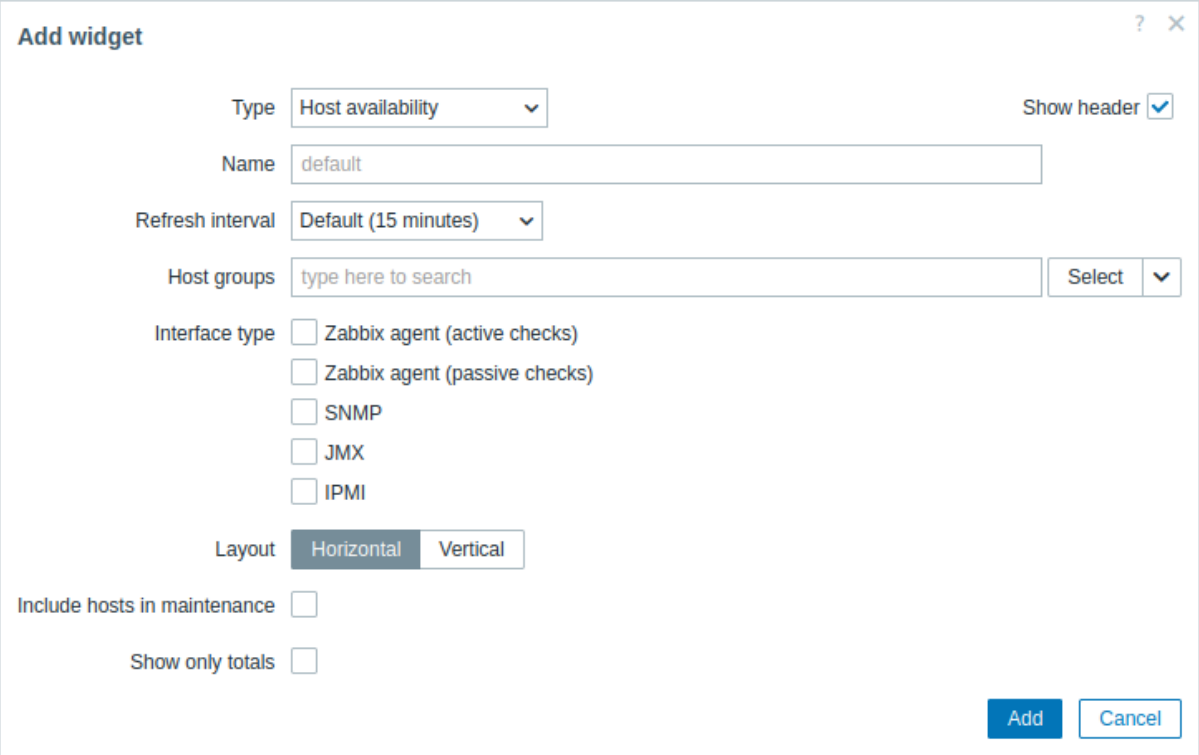
- *Dostępne* - hosty ze wszystkimi interfejsami dostępnymi
- *Niedostępne* - hosty ze wszystkimi interfejsami niedostępnymi

- *Mieszane* - hosty z co najmniej jednym interfejsem niedostępnym i co najmniej jednym dostępnym lub nieznanym; pozostałe mogą mieć dowolną wartość, w tym ”unknown”
- *Nieznane* - hosty z co najmniej jednym interfejsem nieznanym (żaden niedostępny)
- *Łącznie* - suma wszystkich hostów

Note:
W przypadku Zabbix agent (active checks) komórka *Mieszane* zawsze będzie pusta, ponieważ ten typ pozycji nie może mieć wielu interfejsów.

Konfiguracja

Aby skonfigurować, wybierz *Host availability* jako typ:



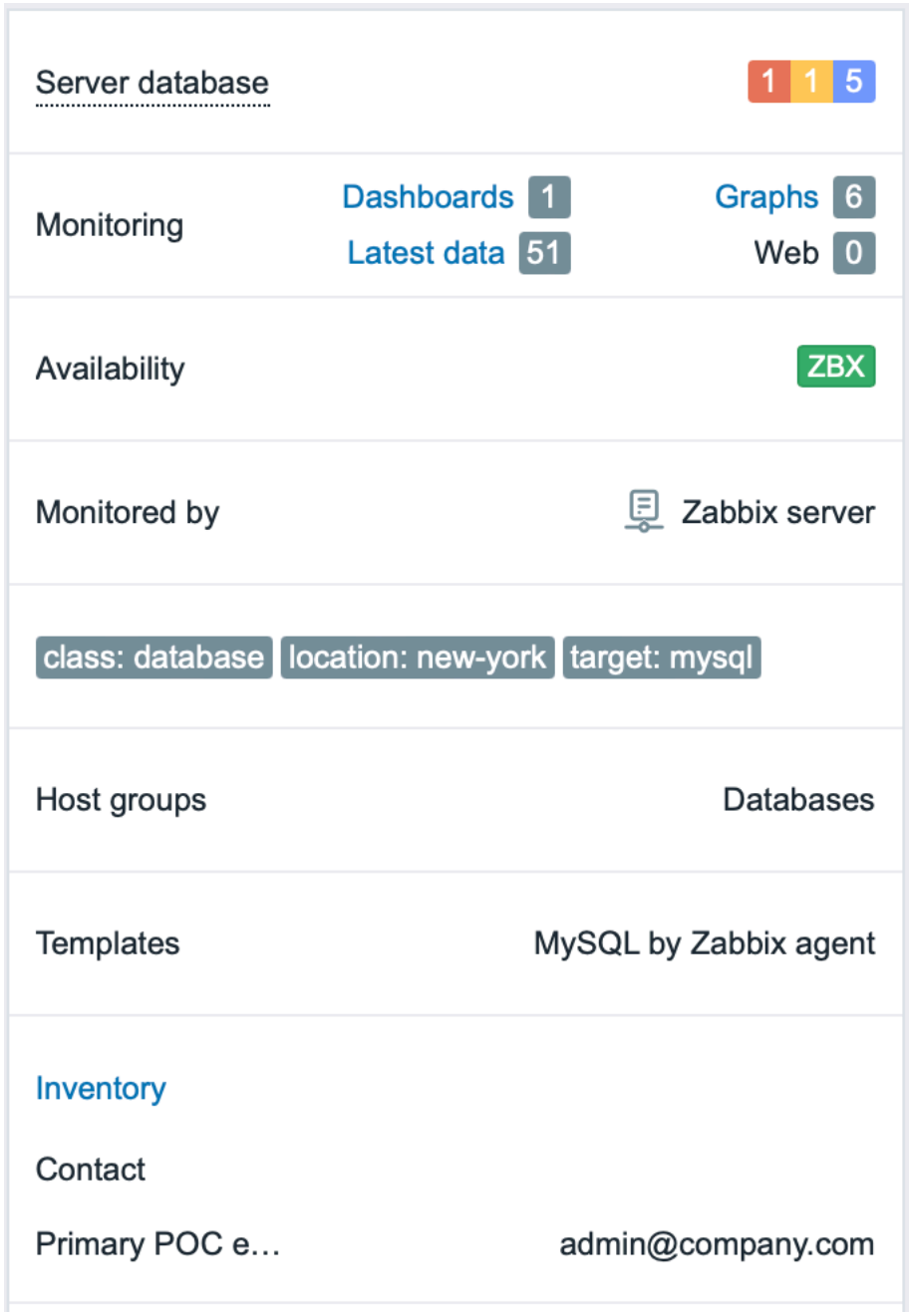
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Interface type	Wybierz, dla których interfejsów hostów mają być wyświetlane dane o dostępności. Domyślnie wyświetlana jest dostępność wszystkich interfejsów, jeśli nic nie zostanie wybrane.
Layout	Wybierz układ poziomy (kolumny) lub pionowy (wiersze).
Include hosts in maintenance	Uwzględnij w statystykach hosty znajdujące się w konserwacji. Ten parametr ma etykietę *Show data in maintenance* podczas konfigurowania widżetu na szablonowym dashboardzie.
Show only totals	Jeśli zaznaczone, wyświetlana jest łączna liczba hostów bez podziału według interfejsów. Ta opcja jest wyłączona, jeśli wybrano tylko jeden interfejs.

13 Karta hosta

Przegląd

Widżet karty hosta zapewnia kompleksowe i konfigurowalne informacje o pojedynczym hoście. Host można określić w konfiguracji widżetu, wybrać z innego zgodnego widżetu lub znaleźć za pomocą selektora hostów na poziomie pulpitu.

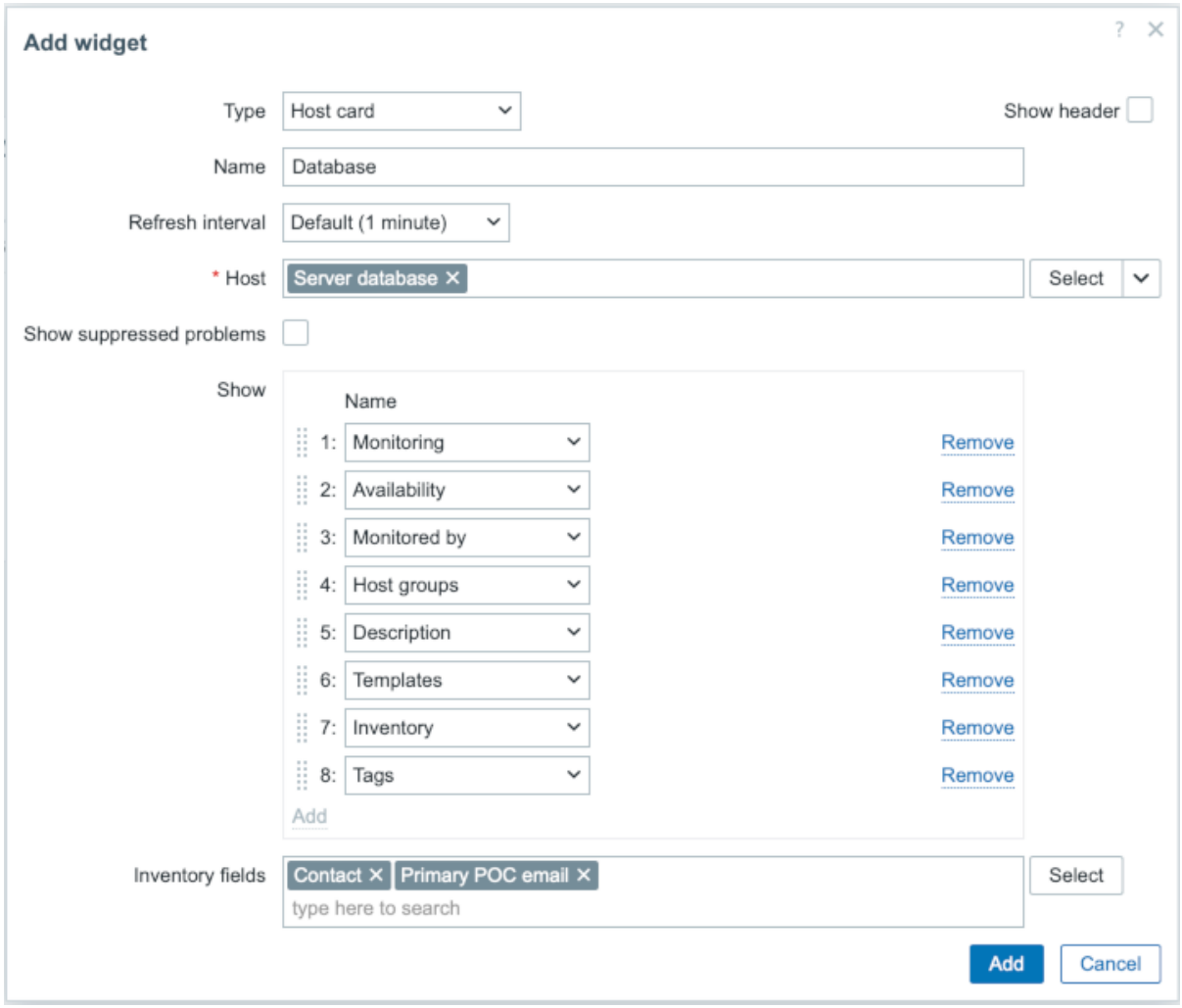


Górna część widżetu zawiera nazwę hosta oraz liczbę problemów hosta pogrupowanych według ważności. Jeśli host jest wyłączony, obok nazwy hosta zostanie wyświetlony tekst *Disabled*. Jeśli host znajduje się w trybie konserwacji, obok nazwy hosta zostanie wyświetlona ikona konserwacji. Pozostałe sekcje widżetu można dostosować zgodnie z potrzebami.

Ten widżet automatycznie dostosowuje układ do szerokości. Aby rozmieścić sekcje widżetu w kilku kolumnach, należy rozszerzyć widżet w poziomie.

Konfiguracja

Aby skonfigurować, wybierz *Host card* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące parametry specyficzne:

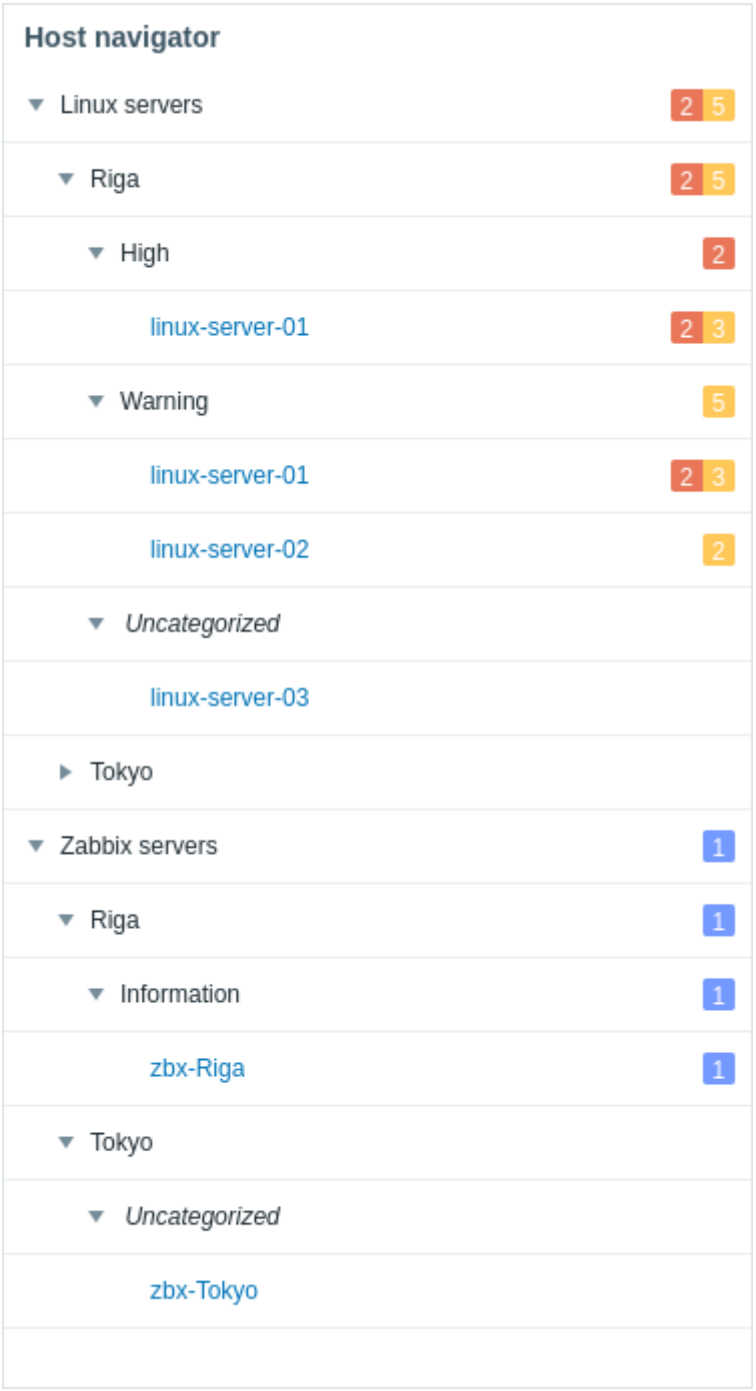
Host	Wybierz host. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych. To pole podczas wpisywania nazwy wyświetla pasujące sugestie. Możesz też nacisnąć przycisk Select, aby wybrać ręcznie. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Show suppressed problems	Zaznacz to pole wyboru, aby uwzględnić wyciszone problemy w obliczaniu problemów.
Show	Skonfiguruj do ośmiu sekcji wyświetlanych przez widżet. Sekcje można zmieniać kolejnością metodą przeciągnij i upuść.
Host groups	Grupy hostów, do których należy host.
Description	Opis hosta.
Monitoring	Szybkie łącza do przechodzenia do pulpitów, najnowszych danych, wykresów i scenariuszy web hosta. Liczba obok każdego łącza wskazuje liczbę obiektów.
Availability	Dostępność hosta według interfejsu.
Monitored by	Serwer Zabbix, proxy lub grupa proxy, przez które host jest monitorowany.
Templates	Szablony powiązane z hostem.
Inventory	Wyświetl pola inwentarza hosta. Jeśli zostanie wybrane, pojawi się parametr *Inventory fields*, umożliwiający określenie, które pola mają być wyświetlane.
Tags	Tagi hosta.

Inventory fields	Określ pola inwentarza do wyświetlenia. To pole podczas wpisywania nazwy wyświetla pasujące sugestie. Alternatywnie możesz nacisnąć przycisk Select, aby wybrać ręcznie. Jeśli nie zostaną określone żadne pola inwentarza, wyświetlone zostaną wszystkie wypełnione pola inwentarza. Ten parametr staje się dostępny tylko wtedy, gdy w sekcji *Show* zostanie wybrane *Inventory*.

14 Nawigator hostów

Przegląd

Widżet nawigatora hostów wyświetla hosty na podstawie różnych opcji filtrowania i grupowania.



Widżet umożliwia również sterowanie informacjami wyświetlanymi w innych widżetach na podstawie wybranego hosta.

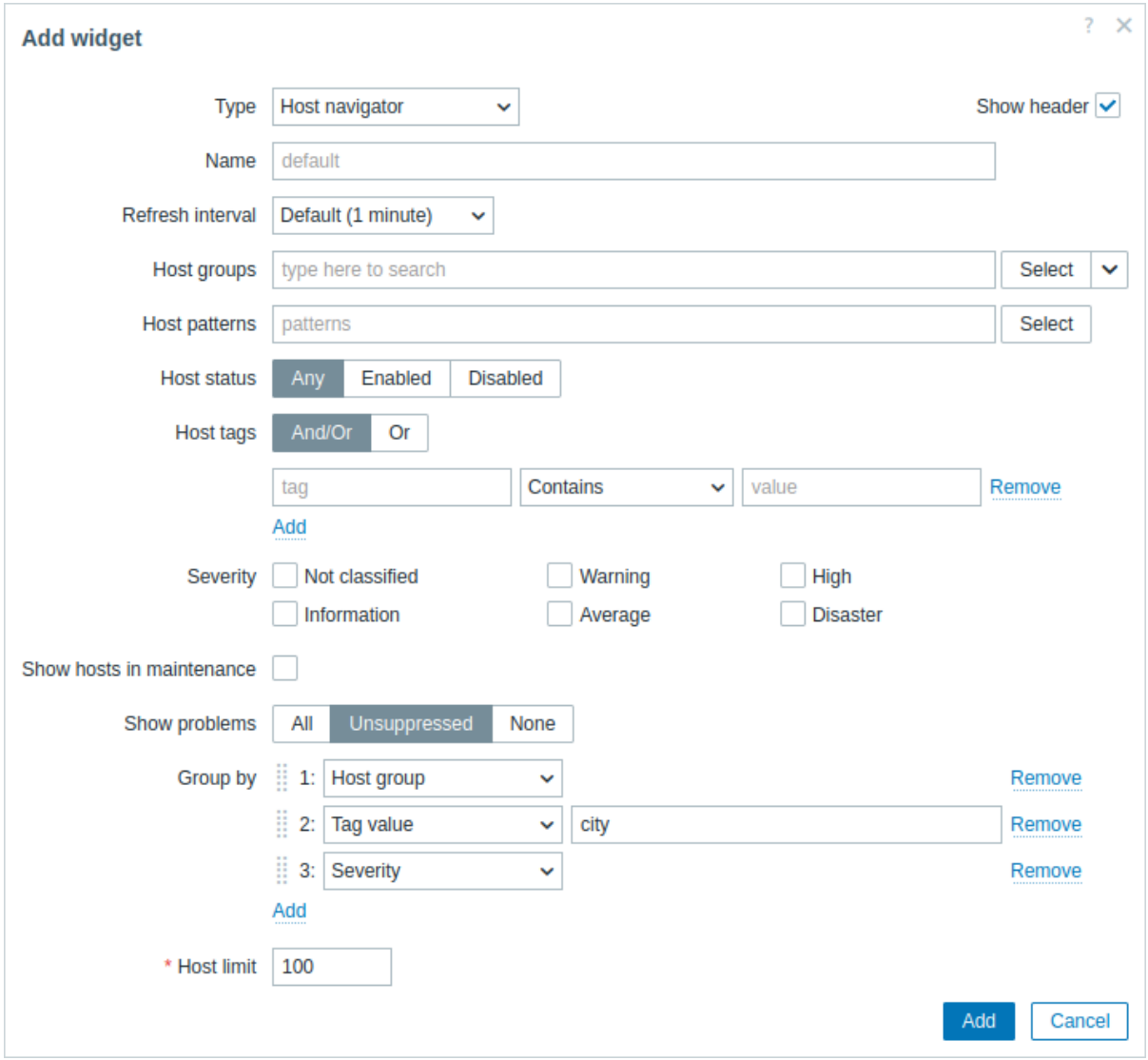


Grupy, według których są organizowane hosty, można rozwijać lub zwijać.

W przypadku grup, problemów i hostów w konserwacji dodatkowe szczegóły są dostępne w podpowiedziach wyświetlanych po najechaniu myszą.

Konfiguracja

Aby skonfigurować, wybierz *Host navigator* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Wybranie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli wszystkie hosty ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Host patterns	Wprowadź wzorce hostów lub wybierz istniejące hosty jako wzorce hostów. Hosty pasujące do określonych wzorców będą wyświetlane w host navigator. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli wszystkie hosty. Do wyboru można używać wzorców wieloznacznych (na przykład * zwróci hosty pasujące do zera lub większej liczby znaków; `Zabbix*` zwróci hosty zaczynające się od "Zabbix"). Aby określić wzorzec wieloznaczny, wpisz ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania lista rozwijana pokaże pasujące hosty ograniczone do tych należących do hostów w wybranych *Host groups*, jeśli takie istnieją.
Host status	Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie. Filtruj hosty do wyświetlenia na podstawie ich statusu (dowolny, włączony, wyłączony). Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.

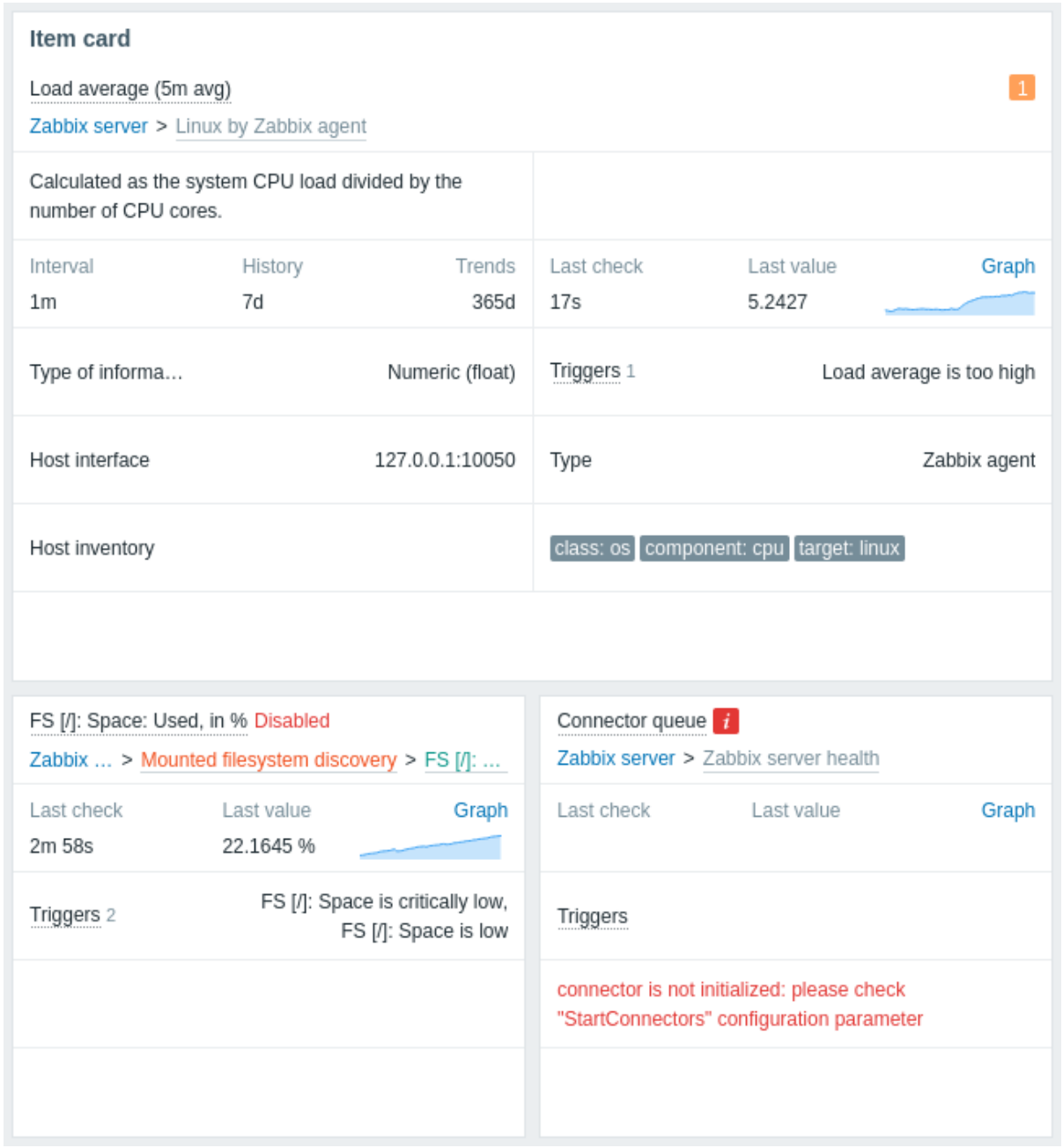
Host tags	Określ tagi, aby filtrować hosty wyświetlane w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Severity	Zaznacz poziomy problemów, aby filtrować hosty z problemami wyświetlane w widżecie. Jeśli nie zaznaczono żadnych poziomów, wyświetlone zostaną wszystkie hosty ze wszystkimi problemami.
Show hosts in maintenance	Zaznacz to pole wyboru, aby wyświetlać hosty w utrzymaniu (w takim przypadku obok nazwy hosta zostanie wyświetlona ikona utrzymania). Ten parametr ma etykietę *Show data in maintenance* podczas konfigurowania widżetu na szablonowym dashboardzie.
Show problems	Filtruj, które problemy mają być wyświetlane wraz z hostami w widżecie, na podstawie ich statusu (wszystkie, bez wstrzymanych, brak).
Group by	Dodaj atrybut grupowania, według którego będą grupowane wybrane hosty: **Host group** - grupuj hosty według ich grup hostów; **Tag value** - wprowadź nazwę tagu, aby grupować hosty według wartości tego tagu (na przykład wpisz ”city”, aby grupować hosty według wartości ”Riga”, ”Tokyo” itd.); **Severity** - grupuj hosty według poziomów ich problemów. Jeśli *Show problems* jest skonfigurowane tak, aby wyświetlać problemy, są one wyświetlane w następujący sposób: - dla każdej grupy poziomu wyświetlana jest tylko odpowiednia liczba problemów; - dla każdego hosta wyświetlane są wszystkie jego liczby problemów. Zwróć uwagę, że hosty będą grupowane tylko według tych poziomów, które zaznaczono w parametrze *Severity*; jeśli nie zaznaczono żadnych poziomów, wszystkie hosty będą grupowane według wszystkich poziomów. Atrybuty grupowania można zmieniać kolejnością, przeciągając je w górę lub w dół za uchwyt przed nazwą grupy. Zwróć uwagę, że kolejność atrybutów grupowania określa kolejność zagnieżdżania grup. Na przykład określenie wielu nazw tagów (1: color, 2: city) spowoduje, że hosty będą grupowane według koloru (red, blue itd.), a następnie według miasta (Riga, Tokyo itd.). Host może być wyświetlany w wielu grupach w zależności od skonfigurowanych atrybutów grupowania (na przykład podczas grupowania według grup hostów, jeśli host należy do wielu grup hostów). Kliknięcie takich hostów zaznacza je i podświetla we wszystkich grupach. Hosty, które nie pasują do skonfigurowanych atrybutów grupowania, są wyświetlane w grupie *Uncategorized*. Można określić maksymalnie 10 atrybutów grupowania i wszystkie muszą być unikalne. Jeśli nie określono żadnych atrybutów grupowania, hosty nie będą grupowane.

Host limit	Wprowadź maksymalną liczbę hostów do wyświetlenia. Możliwe wartości mieszczą się w zakresie 1-9999. Gdy dostępnych jest więcej hostów do wyświetlenia niż ustawiony limit, pod wyświetlonymi hostami pojawia się odpowiedni komunikat (na przykład ”Wyświetlanych jest 100 z ponad 100 hostów”). Zwróć uwagę, że skonfigurowany limit hostów wpływa również na wyświetlanie skonfigurowanych grup; na przykład jeśli limit hostów jest ustawiony na 100, a hosty są grupowane według wartości tagów (więcej niż 200), w widżecie zostanie wyświetlonych tylko pierwszych 100 wartości tagów wraz z odpowiadającymi im hostami. Na ten parametr nie wpływa parametr *Limit for search and filter results* w *Administration → General → GUI*. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.

15 Karta pozycji

Przegląd

Widżet karty pozycji zapewnia kompleksowe i konfigurowalne informacje o pojedynczej pozycji.

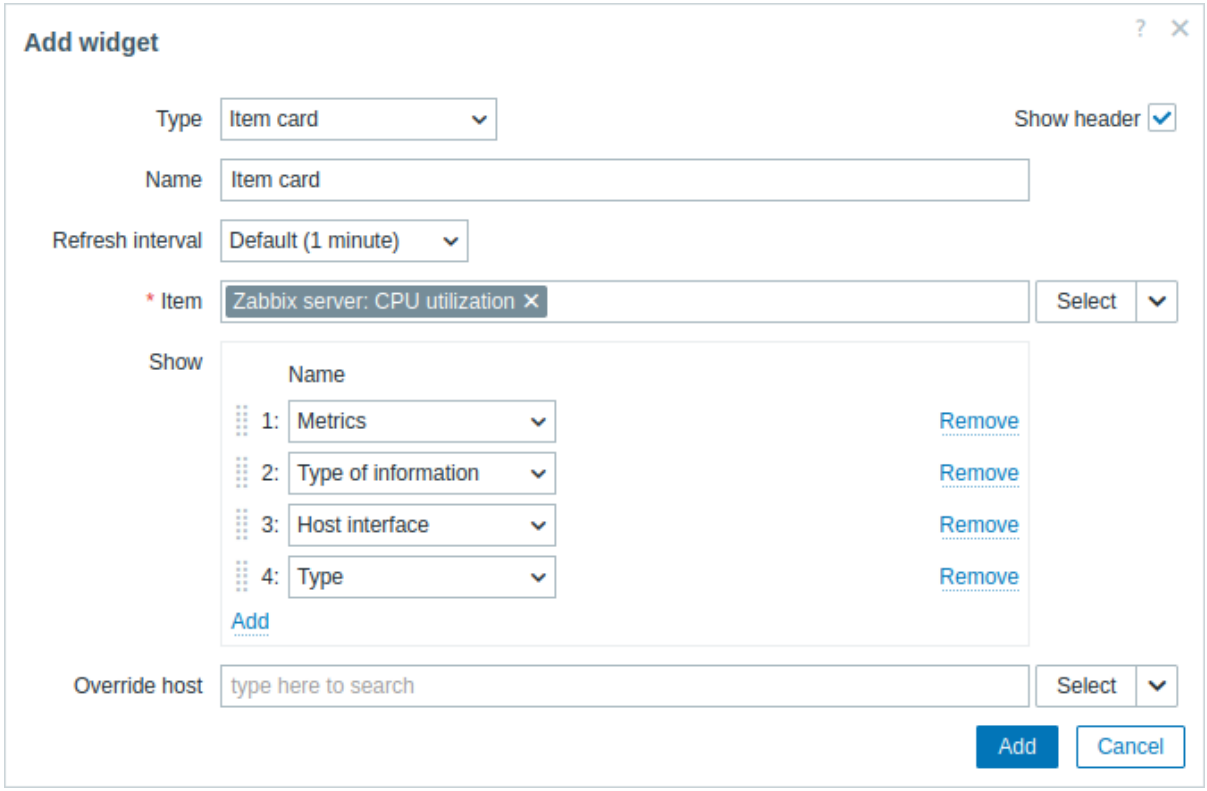


Pozycję można ustawić bezpośrednio w konfiguracji widżetu lub wybrać z kompatybilnego widżetu na pulpicie.

Widżet może wyświetlać wiele sekcji, z których każda pokazuje inny typ informacji (opis pozycji, najnowsze dane itp.). Układ sekcji automatycznie dostosowuje się do szerokości widżetu. Po rozszerzeniu w poziomie sekcje są rozmieszczane w kilku kolumnach, uporządkowanych od lewej do prawej w każdym wierszu.

Konfiguracja

Aby skonfigurować, wybierz *Item card* jako typ:



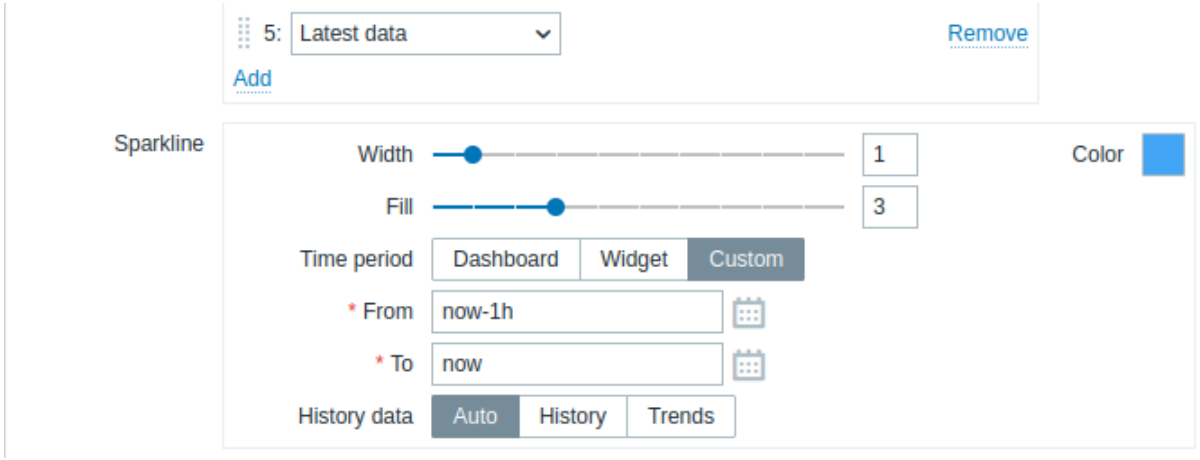
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz zgodny widżet jako źródło danych dla pozycji. Pasujące sugestie pojawiają się podczas wpisywania. Górna sekcja widżetu zawsze wyświetla **nazwę pozycji** (kliknięcie jej otwiera menu pozycji): - Jeśli pozycja jest wyłączona, obok nazwy pozycji wyświetlana jest czerwona etykieta *Disabled*. - W przypadku błędów wyświetlana jest kwadratowa ikona z literą "i". W przypadku problemów wyświetlana jest kwadratowa ikona z liczbą problemów dla każdego poziomu ważności problemu. Najechanie kursorem na ikonę otwiera okno podręczne ze szczegółami. - Na szablon dashboard wyświetlana jest nazwa techniczna pozycji (zamiast nazwy widocznej). Górna sekcja widżetu wyświetla również **ścieżkę pozycji** — jej host oraz, jeśli ma zastosowanie, szablon, regułę low-level discovery (LLD) lub element nadrzędny (kliknięcie encji otwiera jej konfigurację). Ścieżka pozycji jest wyświetlana tylko użytkownikom mającym uprawnienia do odpowiednich encji.
Show	Dodaj sekcje do wyświetlenia w widżecie. Sekcje można zmieniać kolejnością metodą przeciągnij i upuść.
Description	Wyświetla opis pozycji. Makra są rozwijane.
Error text	Wyświetla błąd pozycji (jeśli występuje).
Metrics	Wyświetla następujące informacje: - Interwał aktualizacji pozycji (z wyjątkiem pozycji trapper); - Konfigurację historii pozycji (jeśli istnieje); - Konfigurację trendów pozycji (jeśli istnieje). Makra są rozwijane. Jeśli pozycja używa własnych interwałów, obok interwału aktualizacji pozycji wyświetlana jest ikona pomocy. Najechanie kursorem na ikonę otwiera okno podręczne ze szczegółami własnego interwału.

Latest data	Wyświetla następujące informacje: - Czas od ostatniego sprawdzenia pozycji; - Ostatnia wartość pozycji; - Link do prostego wykresu/historii wartości pozycji (jeśli historia lub trendy są przechowywane); - Konfigurowalny wykres sparkline (jeśli pozycja jest liczbowa). W przypadku pozycji typu binarnego zamiast ostatniej wartości wyświetlana jest miniatura lub opcja *Show*. Najechanie kursorem na miniaturę otwiera okno podręczne z obrazem. Najechanie kursorem na *Show* otwiera okno podręczne z wartością pozycji (ciąg Base64). Na szablon dashboard zamiast ostatniej wartości pozycji wyświetlane jest *No data*.
Type of information	Wyświetla typ informacji pozycji.
Triggers	Wyświetla następujące informacje: - Etykietę *Triggers* z liczbą wyzwalaczy (najechanie kursorem na *Triggers* otwiera okno podręczne ze szczegółami wyzwalaczy); - Listę nazw wyzwalaczy (makra są rozwijane).
Host interface	Wyświetla interfejs hosta używany przez pozycję lub *No data*, jeśli pozycja nie ma interfejsu hosta (np. pozycje zależne, proste sprawdzenia). Na szablon dashboard wyświetlane jest *No data*.
Type	Wyświetla typ pozycji.
Host inventory	Wyświetla pole inwentarza hosta, które jest wypełniane wartością pozycji.
Tags	Wyświetla tagi pozycji.
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboard.

Sparkline

Opcje konfiguracji wykresu sparkline są dostępne tylko wtedy, gdy sekcja *Latest data* zostanie dodana do widgetu.



Width	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie wpisując wartość w zakresie od 0 do 10.
Color	Wybierz kolor linii i wypełnienia.
Fill	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie wpisując wartość w zakresie od 0 do 10.
Time period	Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Time period* jako źródło danych; **Widget** - ustaw zgodny widget określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu myszą.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na ”Widget”.
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na ”Custom”.

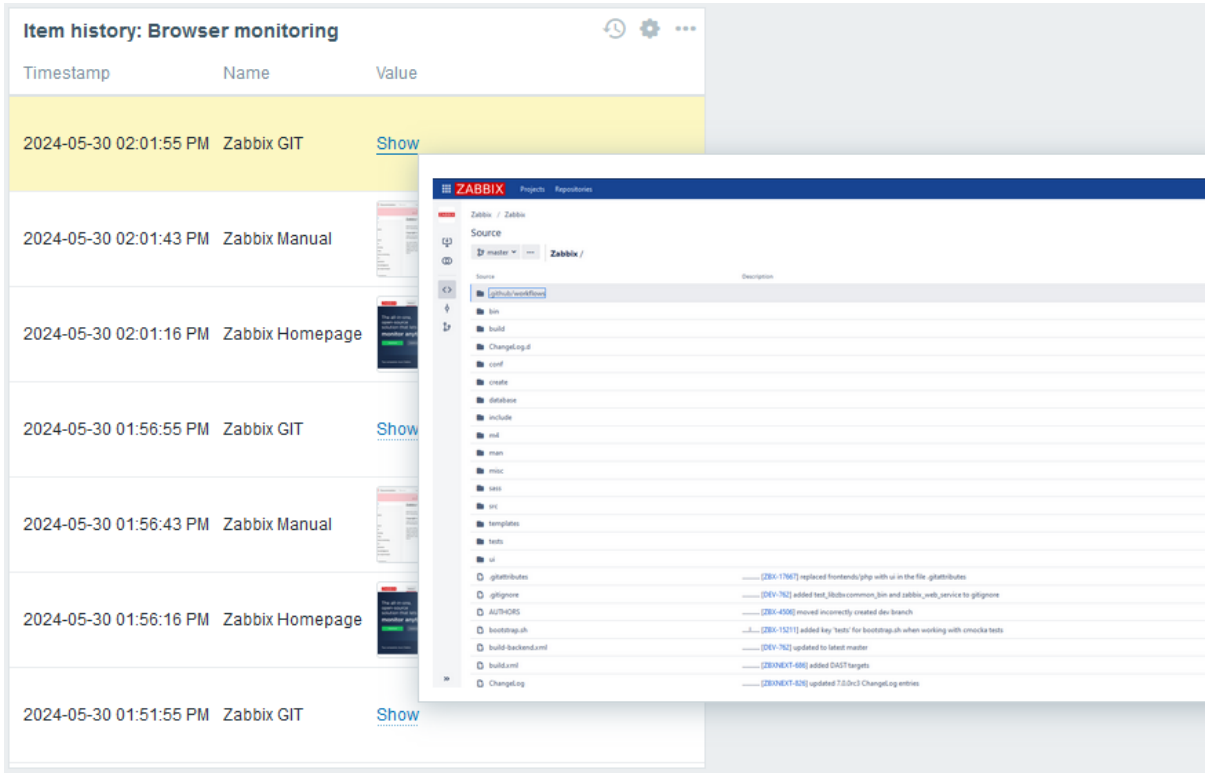
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Custom".
History data	Pobieraj dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobieraj dane historyczne; **Trends** - pobieraj dane trendów. Ten parametr dotyczy tylko danych liczbowych. Dane nieliczbowe będą zawsze pobierane z historii.

16 Historia pozycji

Przegląd

Widżet historii pozycji wyświetla najnowsze dane dla różnych typów pozycji (liczbowych, znakowych, logów, tekstowych i binarnych) w formie tabeli. Może również pokazywać paski postępu, obrazy dla binarnych typów danych (przydatne dla pozycji przeglądarki) oraz wyróżniać wartości (przydatne dla monitorowania plików log).

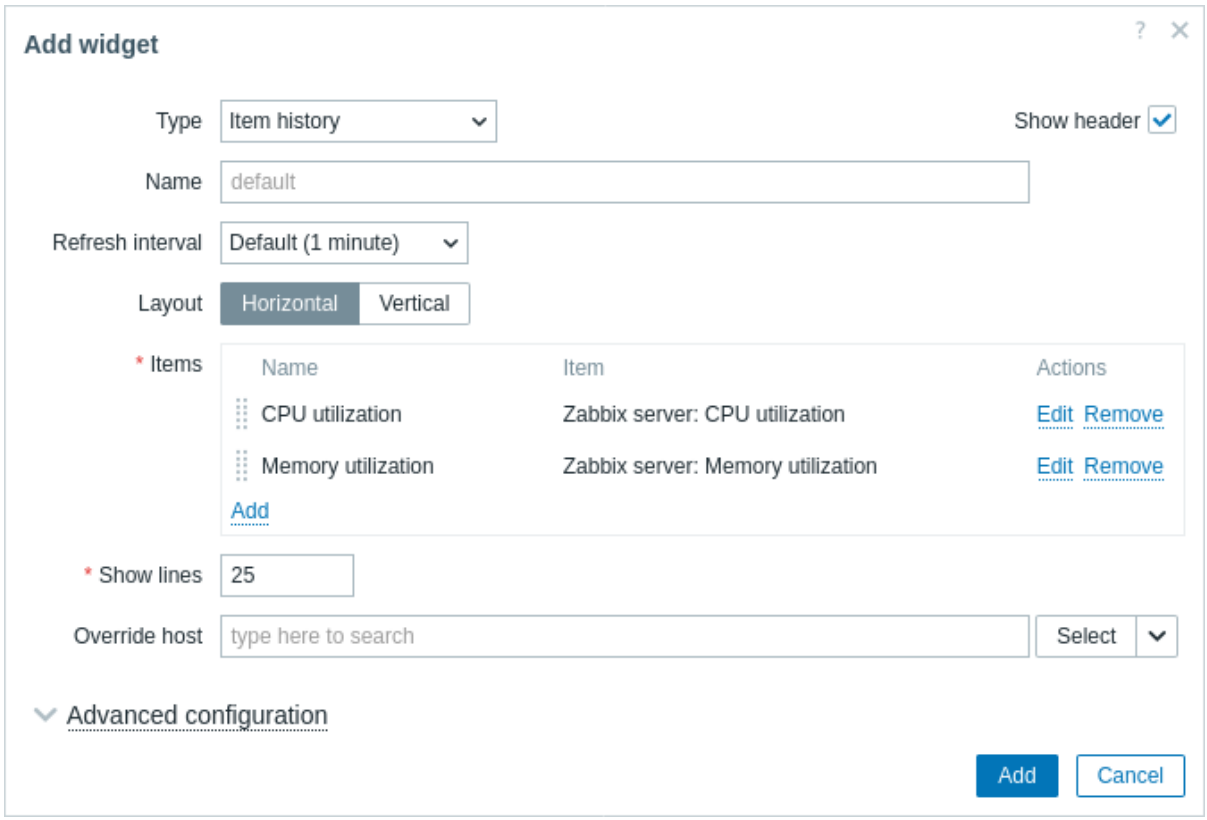




Można wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować, wybierz *Item history* jako typ:



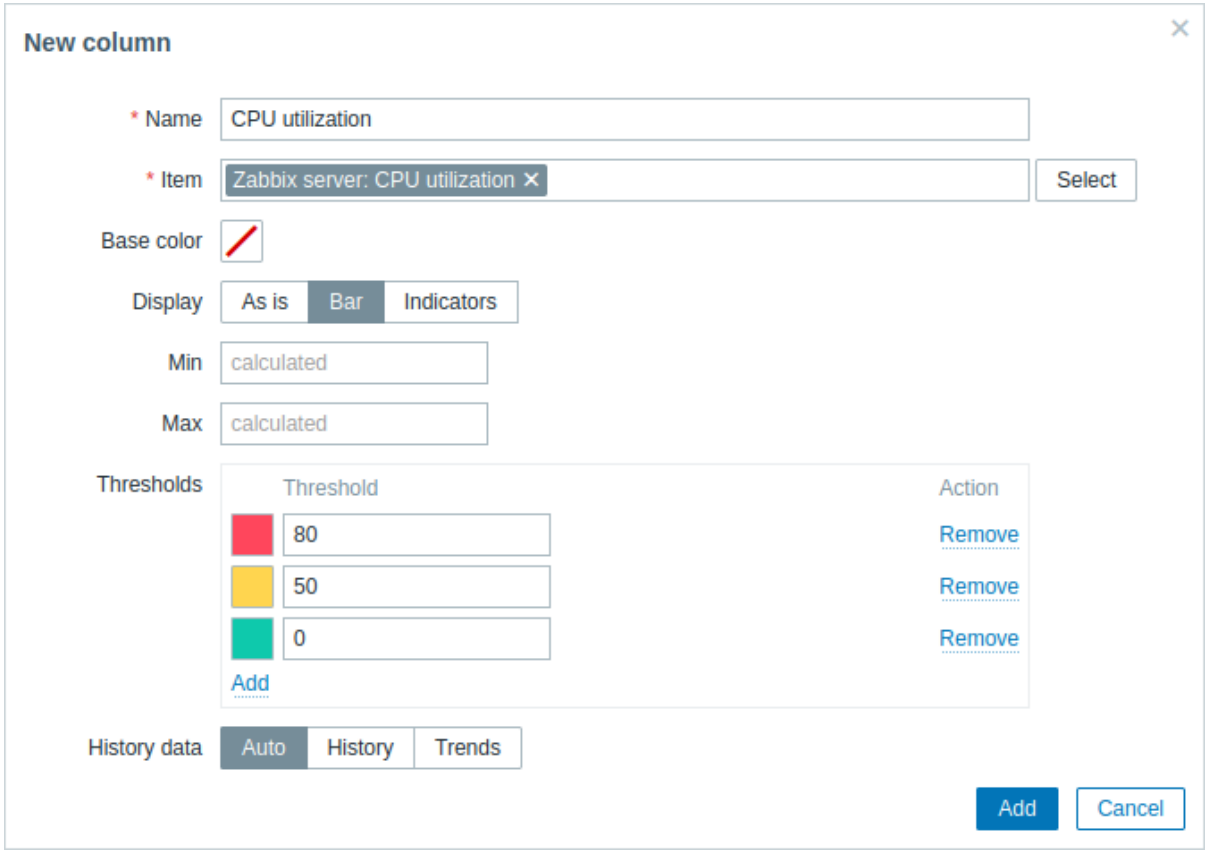
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Layout	Wybierz opcję układu dla kolumn pozycji: **Horizontal** - pozycje będą wyświetlane poziomo, a wartości pionowo; **Vertical** - pozycje będą wyświetlane pionowo, a wartości poziomo.

Items	Dodaj kolumny pozycji do wyświetlenia. Kolejność pozycji określa kolejność ich wyświetlania. Pozycje można zmieniać kolejnością, przeciągając je w górę lub w dół za uchwyt przed nazwą pozycji.
Show lines	Określ liczbę wierszy wartości pozycji do wyświetlenia.
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje zaawansowanej konfiguracji.

Konfiguracja kolumn

Aby skonfigurować kolumny pozycji, kliknij *Dodaj* w parametrze *Pozycje*:



Wspólne parametry kolumn:

Nazwa	Wprowadź nazwę kolumny. Jeśli pozostawisz puste, zostanie użyta nazwa pozycji z parametru *Pozycja*.
Pozycja	Wybierz pozycję. Zwróć uwagę, że parametry konfiguracji kolumny różnią się w zależności od typu informacji wybranej pozycji; więcej informacji znajdziesz w opisach poszczególnych parametrów poniżej. Podczas konfigurowania widżetu na pulpicie szablonu można wybrać tylko pozycje skonfigurowane w szablonie.
Kolor bazowy	Wybierz kolor tła kolumny lub kolor wypełnienia, jeśli *Wyświetlanie* jest ustawione na „Słupek” lub „Wskaźniki”. Zwróć uwagę, że kolor bazowy może zostać nadpisany przez kolory progów lub wyróżnienia.

Parametry kolumny specyficzne dla pozycji typu numerycznego:

Wyświetlanie	Wybierz sposób wyświetlania wartości pozycji: **As is** - jako zwykły tekst; **Bar** - jako jednolity pasek wypełniony kolorem; **Indicators** - jako segmentowany pasek wypełniony kolorem.

Min	Wprowadź wartość minimalną dla paska/wskaźników. Jeśli pole pozostanie puste, widget użyje minimalnej wartości pozycji. Ten parametr jest dostępny tylko wtedy, gdy opcja *Wyświetlanie* jest ustawiona na „Bar” lub „Indicators”.
Max	Wprowadź wartość maksymalną dla paska/wskaźników. Jeśli pole pozostanie puste, widget użyje maksymalnej wartości pozycji. Ten parametr jest dostępny tylko wtedy, gdy opcja *Wyświetlanie* jest ustawiona na „Bar” lub „Indicators”.
Progi	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość numeryczną. Lista progów zostanie posortowana rosnąco podczas zapisywania. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Dane historyczne	Wybierz, czy pobierać dane z historii, czy z trendów: **Auto** - wybór automatyczny; **History** - pobierz dane historyczne; **Trends** - pobierz dane trendów.

Parametry kolumn specyficzne dla pozycji typu character, text i log:

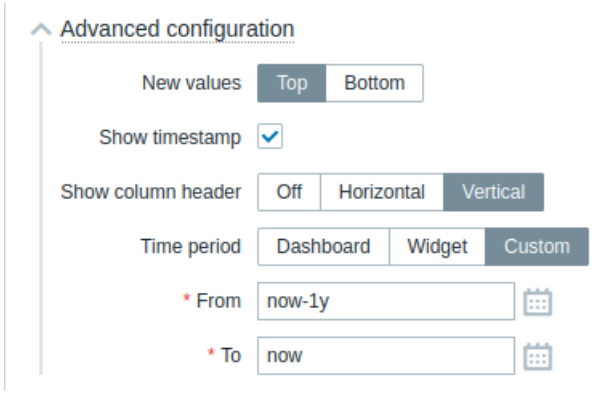
Highlights	Kliknij *Add*, aby dodać wyróżnienie, wybierz kolor wyróżnienia z selektora kolorów i określ wyrażenie regularne. Wybrany kolor zostanie użyty jako kolor tła dla wartości pozycji, w których określone wyrażenie regularne pasuje do tekstu.
Display	Wybierz sposób wyświetlania wartości pozycji: **As is** - dokładnie tak, jak została odebrana. Podziały wierszy są zachowywane, jeśli wartość zawiera znaki końca wiersza. Zawijanie słów jest stosowane, jeśli *Layout* jest ustawione na ”Vertical” i skonfigurowano więcej niż jedną kolumnę; **HTML** - jako tekst sformatowany w HTML; **Single line** - jako pojedyncza linia, skrócona do określonej długości (1-500 znaków). Po najechaniu kursorem lub kliknięciu skróconej wartości otwiera się okno podręczne z pełną wartością.
Use monospace font	Zaznacz to pole wyboru, aby wyświetlać wartość pozycji czcionką monospace (domyślnie niezaznaczone).
Display local time	Zaznacz to pole wyboru, aby wyświetlać czas lokalny zamiast znacznika czasu w kolumnie znacznika czasu. Pamiętaj, że pole wyboru *Show timestamp* w advanced configuration również musi być zaznaczone. Ten parametr jest dostępny tylko dla pozycji typu log.

Parametry kolumny specyficzne dla pozycji typu binarnego:

Pokaż miniaturę	Zaznacz to pole wyboru, aby wyświetlać miniaturę dla binariów obrazów lub opcję „Pokaż” dla binariów niebędących obrazami. Odznacz to pole wyboru, aby wyświetlać opcję „Pokaż” dla wszystkich wartości binarnych pozycji. Najechanie kursorem na opcję „Pokaż” lub kliknięcie jej otwiera wyskakujące okno z wartością pozycji (obrazem lub ciągiem Base64). Jeśli wartość pozycji jest pustym ciągiem, wyświetlana jest opcja „Pokaż”; najechanie kursorem na nią lub kliknięcie jej otwiera wyskakujące okno zawierające „Pusty ciąg”.

Zaawansowana konfiguracja

Opcje zaawansowanej konfiguracji są dostępne w zwijanej sekcji *Zaawansowana konfiguracja*:

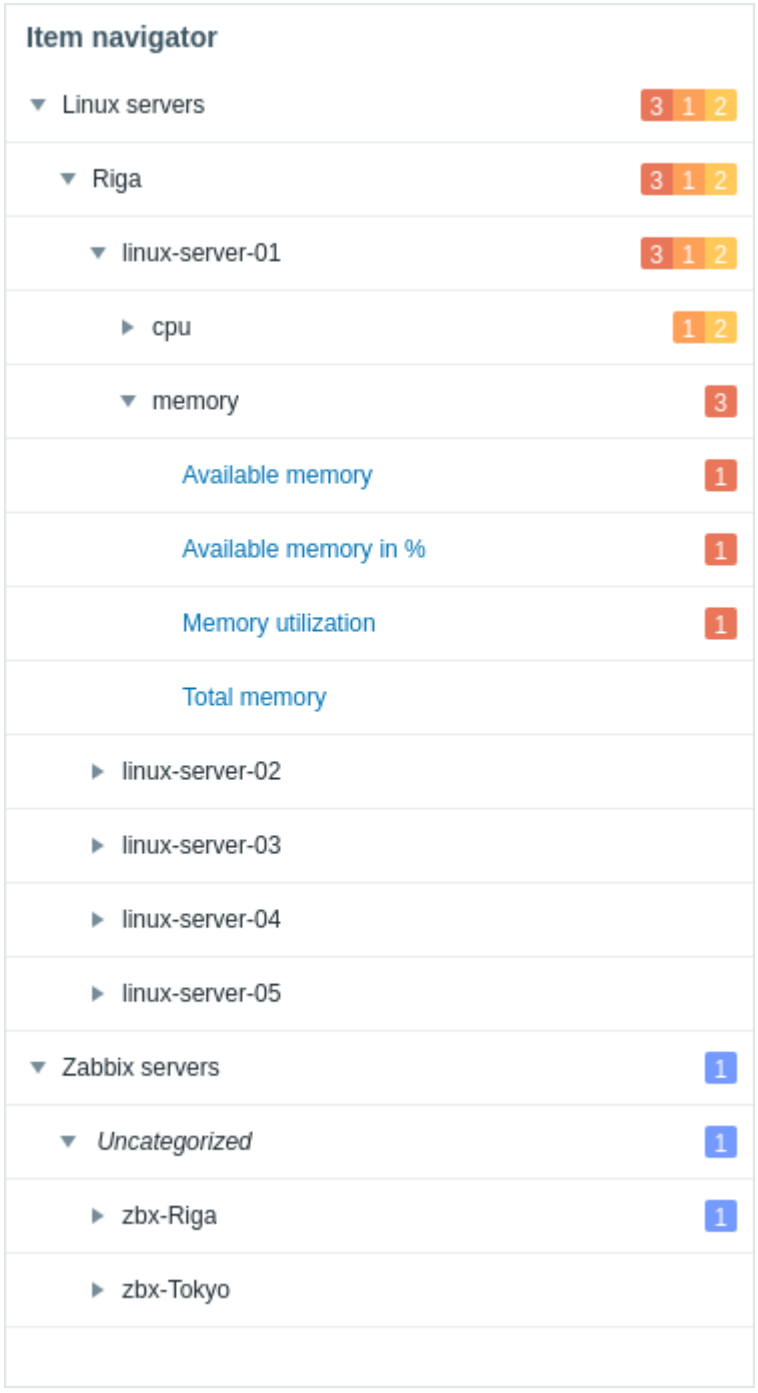


Nowe wartości	Wybierz, gdzie mają być dodawane nowe wartości pozycji: **Góra** - na górze kolumn; **Dół** - na dole kolumn.
Pokaż znacznik czasu	Zaznacz to pole wyboru, aby wyświetlić kolumnę znacznika czasu (domyślnie niezaznaczone).
Pokaż nagłówek kolumny	Wybierz orientację nagłówka kolumny: **Wył.** - ukryj nagłówek; **Poziomo** - wyświetl nagłówek poziomo; **Pionowo** - wyświetl nagłówek pionowo.
Okres czasu	Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Okres czasu* jako źródło danych; **Widget** - ustaw zgodny widget określony w parametrze *Widget* jako źródło danych; **Niestandardowy** - ustaw okres czasu określony w parametrach *Od* i *Do* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu myszą.
Widget	Wprowadź lub wybierz zgodny widget (*Graph*, *Graph (classic)*, *Graph prototype*) jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Widget”.
Od	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itp.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itp.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Niestandardowy”.

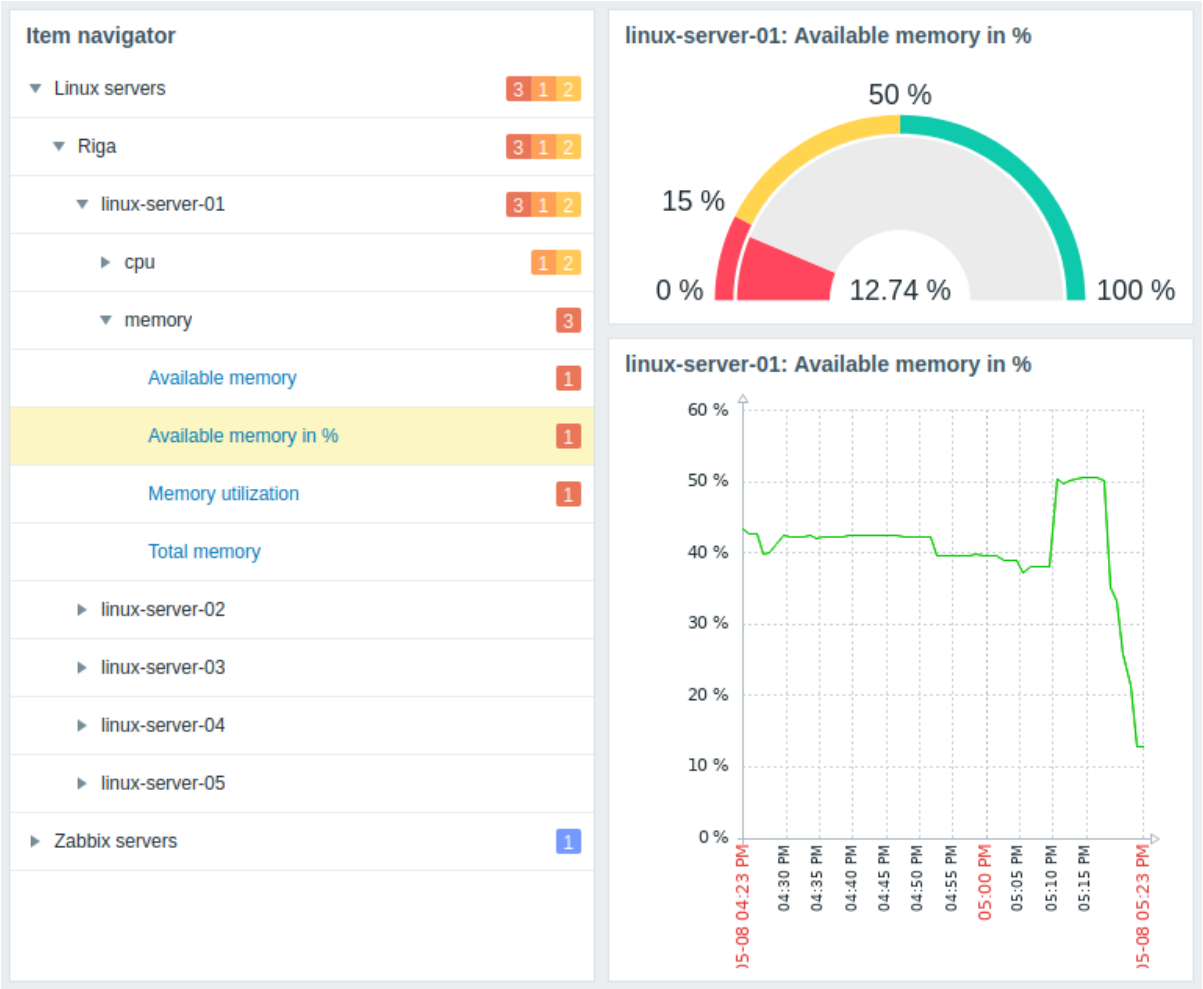
17 Nawigator pozycji

Przegląd

Widżet nawigatora pozycji wyświetla pozycje na podstawie różnych opcji filtrowania i grupowania.



Widżet umożliwia również sterowanie informacjami wyświetlanymi w innych widżetach na podstawie wybranej pozycji.

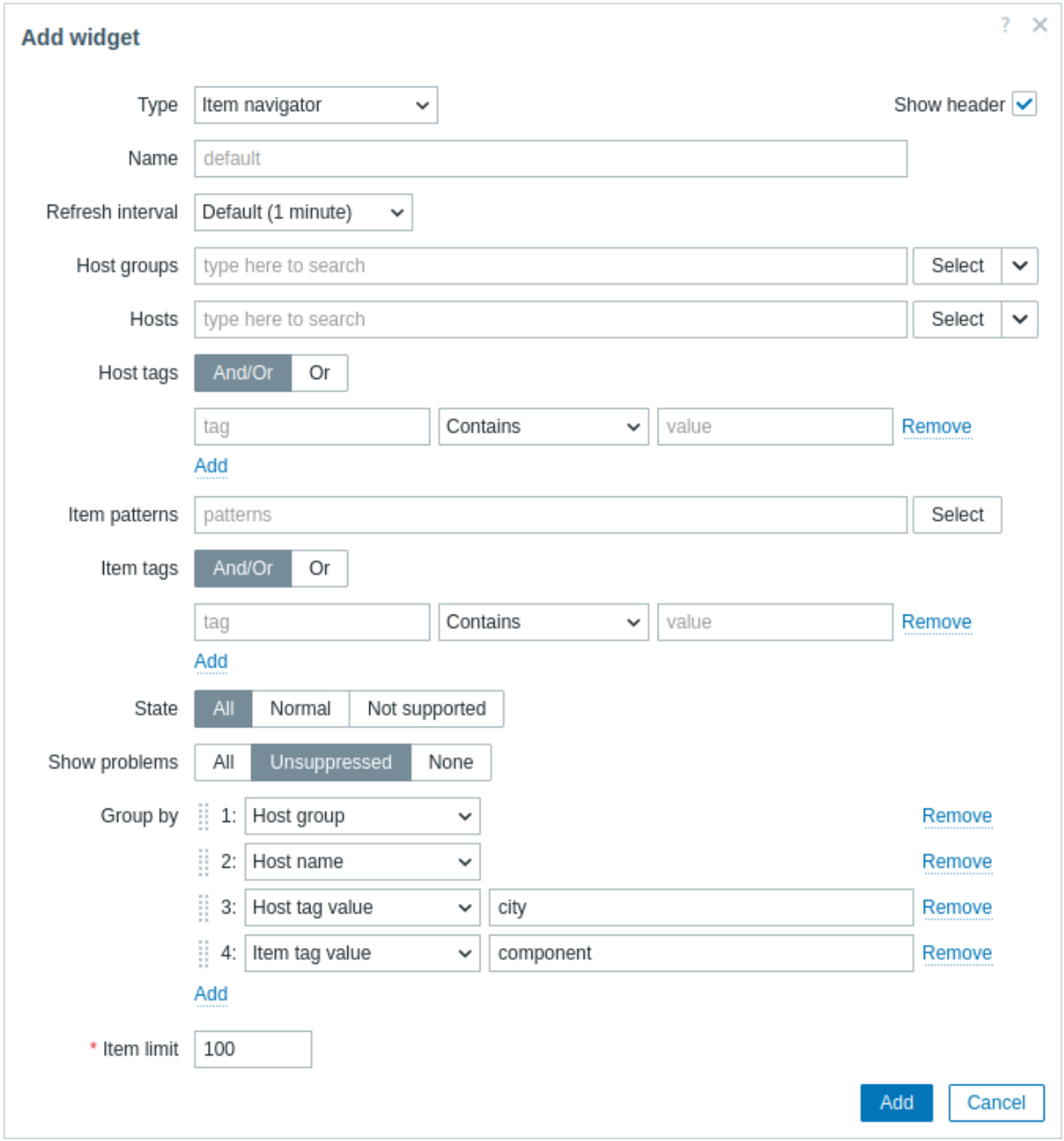


Grupy, według których są organizowane pozycje, można rozwijać lub zwijać.

W przypadku grup i problemów dodatkowe szczegóły są dostępne w podpowiedziach wyświetlanych po najechaniu myszą.

Konfiguracja

Aby skonfigurować, wybierz *Item navigator* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Wybranie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli pozycje należące do wszystkich hostów ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.
Hosts	Wybierz hosty. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli pozycje należące do wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.

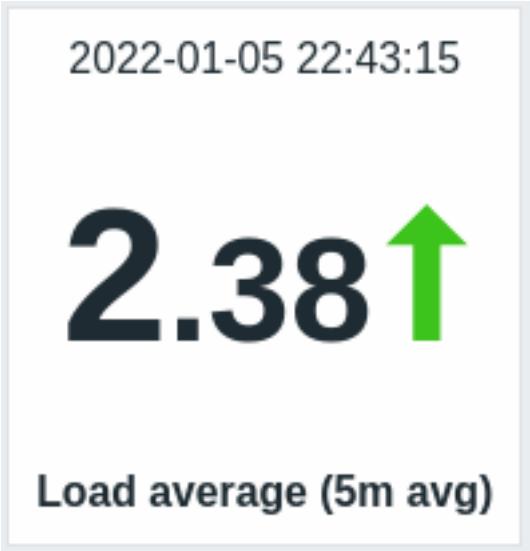
Host tags	Określ tagi hostów, aby filtrować pozycje wyświetlane w widżecie. Można uwzględniać oraz wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.
Item patterns	Wprowadź wzorce pozycji lub wybierz istniejące pozycje jako wzorce pozycji. Pozycje pasujące do określonych wzorców będą wyświetlane w nawigatorze pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci pozycje pasujące do zera lub większej liczby znaków; `Zabbix*` zwróci pozycje zaczynające się od ”Zabbix”). Aby określić wzorzec wieloznaczny, wpisz ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania lista rozwijana pokaże pasujące pozycje ograniczone do tych należących do wybranych *Hosts* lub hostów w wybranych *Host groups*, jeśli takie istnieją. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać na przykład pozycji o nazwie *item** indywidualnie, jeśli istnieją inne pasujące pozycje (np. item2, item3). Podczas konfigurowania widżetu na panelu szablonu ten parametr umożliwia wybór tylko pozycji skonfigurowanych na szablonie.
Item tags	Określ tagi pozycji, aby filtrować pozycje wyświetlane w widżecie. Można uwzględniać oraz wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek.
State	Filtruj, które pozycje mają być wyświetlane na podstawie ich stanu (all, normal, not supported).
Show problems	Filtruj, które problemy mają być wyświetlane wraz z pozycjami na podstawie ich statusu (all, unsuppressed, none).

Group by	Dodaj atrybut grupowania, według którego będą grupowane pozycje: **Host group** - grupuj pozycje według grup hostów, do których należą ich hosty; **Host name** - grupuj pozycje według ich hostów; **Host tag value** - wprowadź nazwę tagu, aby grupować pozycje według wartości tego tagu hosta (na przykład wpisz ”city”, aby grupować według wartości ”Riga”, ”Tokyo” itd.); **Item tag value** - wprowadź nazwę tagu, aby grupować pozycje według wartości tego tagu pozycji (na przykład wpisz ”component”, aby grupować pozycje według wartości ”cpu”, ”memory” itd.). Atrybuty grupowania można zmieniać kolejnością, przeciągając je w górę lub w dół za uchwyt przed nazwą grupy. Zwróć uwagę, że kolejność atrybutów grupowania określa kolejność zagnieżdżania grup. Na przykład podanie wielu nazw tagów hosta (1: color, 2: city) spowoduje grupowanie pozycji według koloru (red, blue itd.), a następnie według miasta (Riga, Tokyo itd.). Pozycja może być wyświetlana w wielu grupach w zależności od skonfigurowanych atrybutów grupowania (na przykład podczas grupowania według grup hostów, gdy host pozycji należy do wielu grup hostów). Kliknięcie takich pozycji zaznacza je i podświetla we wszystkich grupach. Pozycje, które nie pasują do skonfigurowanych atrybutów grupowania, są wyświetlane w grupie *Uncategorized*. Jeśli *Show problems* jest skonfigurowane tak, aby wyświetlać problemy, są one wyświetlane w następujący sposób: - dla każdej grupy wyświetlana jest łączna liczba problemów wszystkich podrzędnych pozycji; - dla każdej pozycji wyświetlana jest tylko liczba jej problemów. Można określić maksymalnie 10 atrybutów grupowania i wszystkie muszą być unikalne. Jeśli nie określono żadnych atrybutów grupowania, pozycje nie będą grupowane.
Item limit	Wprowadź maksymalną liczbę pozycji do wyświetlenia. Dozwolony zakres wartości to 1-9999. Jeśli dostępnych jest więcej pozycji do wyświetlenia niż ustawiony limit, pod wyświetlanymi pozycjami zostanie pokazany odpowiedni komunikat (na przykład ”Wyświetlono 100 z ponad 100 pozycji”). Zwróć uwagę, że skonfigurowany limit pozycji wpływa również na wyświetlanie skonfigurowanych grup; na przykład jeśli limit pozycji jest ustawiony na 100, a pozycje są grupowane według hostów (z których każdy zawiera 200 pozycji), w widżecie zostanie wyświetlony tylko pierwszy host ze 100 pozycjami. Na ten parametr nie wpływa parametr *Limit for search and filter results* w *Administration > General > GUI*.

18 Wartość pozycji

Omówienie

Ten widżet jest przydatny do wyróżnionego wyświetlania wartości pojedynczej pozycji. Może to być zarówno najnowsza wartość, jak i wartość zagregowana dla pewnego okresu w przeszłości.



Oprócz samej wartości można, jeśli chcesz, wyświetlić dodatkowe elementy:

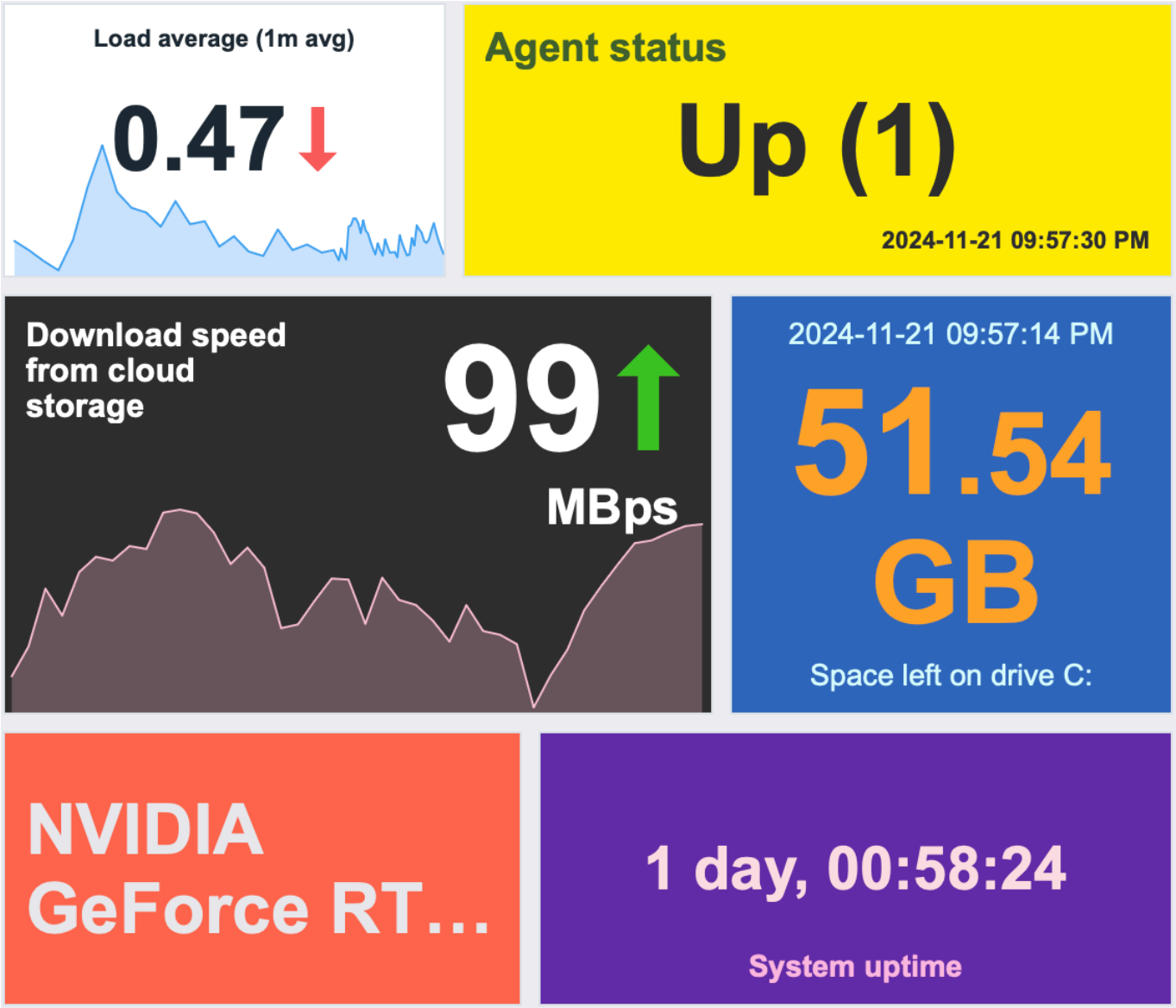
- Czas metryki
- Opis pozycji
- Wskaźnik zmiany wartości
- Kolor tła dla wartości
- Jednostka pozycji
- Wykres sparkline dla wartości z określonego przedziału czasu

Widżet może wyświetlać wartości liczbowe i tekstowe. Wyświetlanie wartości binarnych nie jest obsługiwane. Wartości tekstowe są wyświetlane w jednej linii i w razie potrzeby skracane. Jeśli nie ma wartości dla pozycji, wyświetlane jest „Brak danych”.

Wskaźnik zmiany zawsze porównuje z tym samym okresem w przeszłości. Na przykład najnowsza wartość zostanie porównana z poprzednią wartością, a najnowszy miesiąc z miesiącem wcześniejszym. Zwróć uwagę, że poprzedni okres dla agregacji jest obliczany jako przedział czasu o tej samej długości co oryginalny, z czasem zakończenia bezpośrednio przed czasem rozpoczęcia oryginalnego okresu.

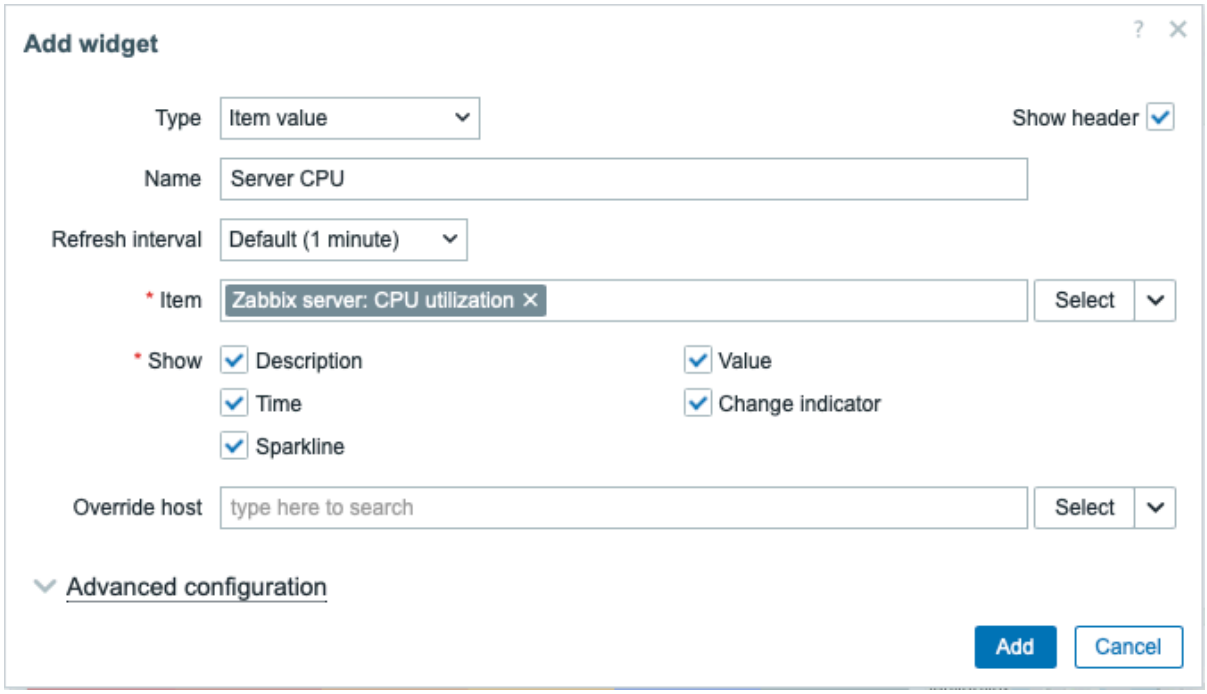
Kliknięcie wartości prowadzi do grafu ad hoc dla pozycji liczbowych lub do najnowszych danych dla pozycji tekstowych.

Widżet oraz wszystkie jego elementy można wizualnie dostroić za pomocą opcji zaawansowanej konfiguracji, co pozwala tworzyć szeroką gamę stylów wizualnych:



Konfiguracja

Aby skonfigurować, wybierz *Item value* jako typ widżetu:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz zgodny widżet jako źródło danych dla pozycji.
Show	Zaznacz pole wyboru, aby wyświetlić odpowiedni element: opis, wartość, czas, wskaźnik zmiany, sparkline. Odznacz, aby ukryć. Należy wybrać co najmniej jeden element.
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje zaawansowanej konfiguracji.

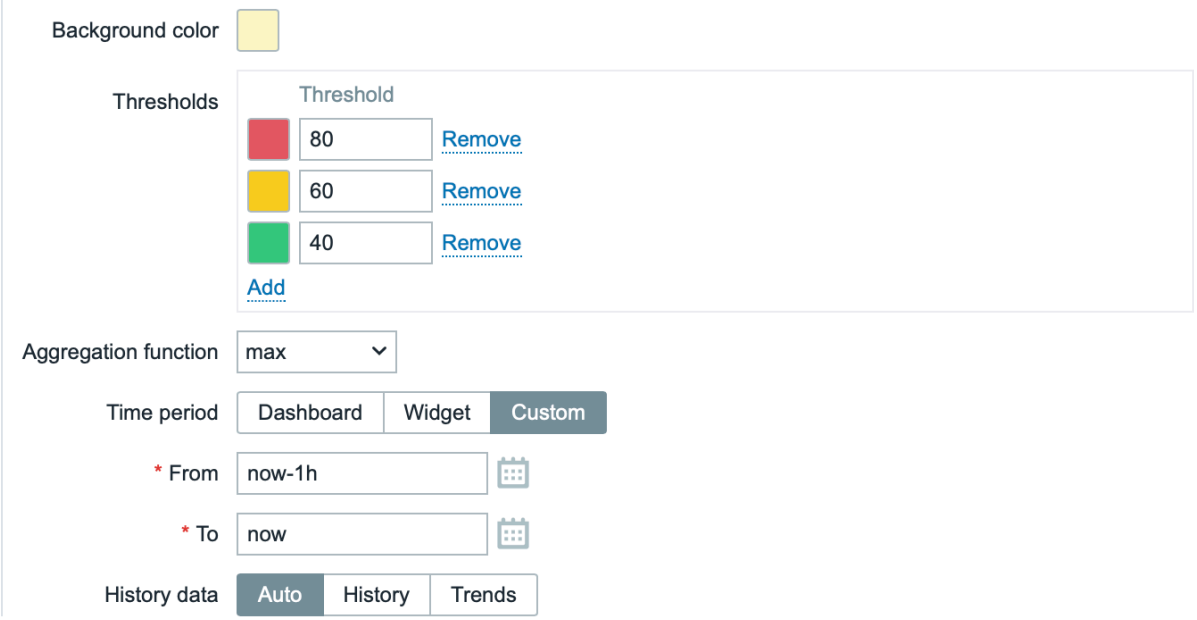
Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*, i tylko dla tych elementów, które są wybrane w polu *Pokaż* (zobacz powyżej).

Należy pamiętać, że wiele elementów nie może zajmować tej samej przestrzeni; jeśli zostaną umieszczone w tej samej przestrzeni, zostanie wyświetlony komunikat o błędzie.

Ogólne parametry widżetu

Te parametry określają kolor tła (statyczny lub dynamiczny) dla całego widżetu oraz funkcję agregacji do wyświetlania wartości.

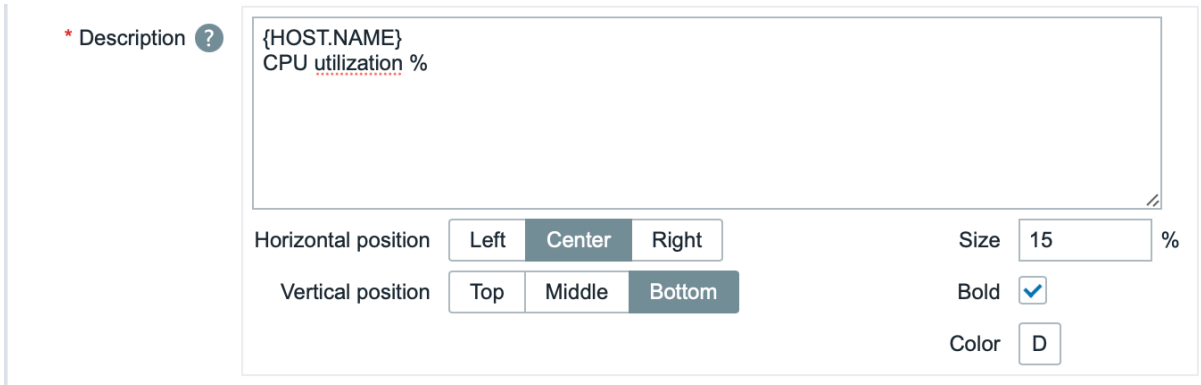


Kolor tła	Wybierz kolor tła dla całego widżetu z selektora kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
Progi	Skonfiguruj dynamiczny kolor tła dla całego widżetu. Kliknij *Dodaj*, aby dodać próg, wybierz kolor tła z selektora kolorów i określ wartość liczbową. Gdy wartość pozycji będzie równa wartości progu lub ją przekroczy, kolor tła zostanie zmieniony. Lista zostanie po zapisaniu posortowana rosnąco. Zwróć uwagę, że dynamiczny kolor tła będzie wyświetlany poprawnie tylko dla pozycji liczbowych.
Funkcja agregacji	Określ, której funkcji agregacji użyć w wybranym *Okresie czasu*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **count** - wyświetl liczbę wartości; **sum** - wyświetl sumę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość. Jeśli wybrano **nie używaj**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja pozwala wyświetlać wartość zagregowaną dla wybranego interwału (5 minut, godzina, dzień) zamiast najnowszej wartości. Tylko dane liczbowe mogą być wyświetlane dla *min*, *max*, *avg* i *sum*. Dla *count* dane nieliczbowe zostaną przekształcone na liczbowe.

Okres czasu	Określ okres czasu używany do agregowania wartości: **Dashboard** - użyj okresu czasu dashboardu; **Widget** - użyj okresu czasu określonego widżetu; **Custom** - użyj niestandardowego okresu czasu. Ten parametr jest ukryty, gdy *Funkcja agregacji* jest ustawiona na ”nie używaj”.
Widżet	Wybierz widżet. Ten parametr będzie wyświetlany tylko wtedy, gdy *Okres czasu* jest ustawiony na ”Widget”.
Od	Wybierz okres czasu od (wartość domyślna `now-1h`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy *Okres czasu* jest ustawiony na ”Custom”.
Do	Wybierz okres czasu do (wartość domyślna `now`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy *Okres czasu* jest ustawiony na ”Custom”.
Dane historyczne	Pobieraj dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobieraj dane historyczne; **Trends** - pobieraj dane trendów. To ustawienie ma zastosowanie tylko do danych liczbowych. Dane nieliczbowe będą zawsze pobierane z historii.

Opis

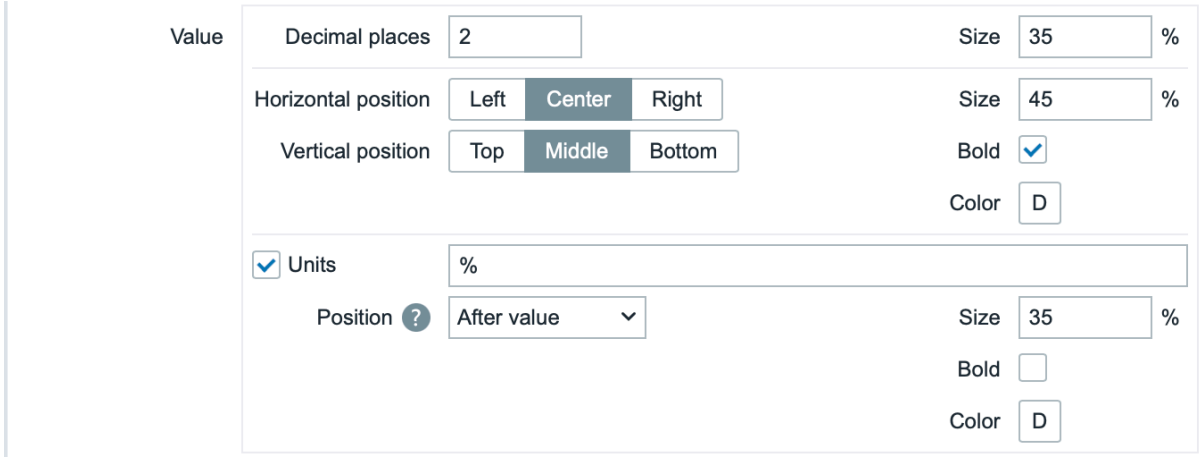
Te parametry określają, w jaki sposób ma być wyświetlany opis pozycji.



Opis	Wprowadź opis pozycji. Ten opis może zastąpić domyślną nazwę pozycji. Obsługiwane są opisy wielowierszowe. Możliwe jest połączenie tekstu i obsługiwanych makr. Obsługiwane są {HOST.*}, {ITEM.*}, {INVENTORY.*} oraz makra użytkownika.
Pozycja pozioma	Wybierz poziome położenie opisu pozycji — z lewej, z prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie opisu pozycji — u góry, u dołu lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla opisu pozycji (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać opis pozycji pogrubioną czcionką.
Kolor	Wybierz kolor opisu pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Wartość

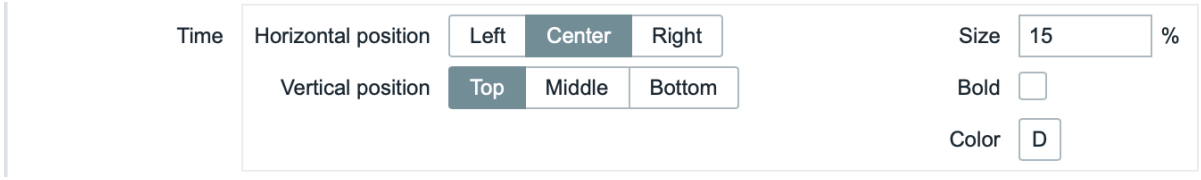
Te parametry określają, jak powinna być wyświetlana wartość pozycji.



Miejsca dziesiętne	Wybierz, ile miejsc dziesiętnych ma być wyświetlanych wraz z wartością. Ta wartość wpływa tylko na pozycje typu float. Dla pozycji z jednostkami ustawionymi na „s”, jeśli ta wartość jest większa od 0 (domyślnie 2), widżet zaokrągla najbardziej znaczącą jednostkę czasu i wyświetla wartość liczbową (np. „10.43m”) zamiast konwertować wartość na pełny ciąg czasu. Po ustawieniu na 0 sekundy są konwertowane na czytelny dla człowieka ciąg czasu (na przykład „4h 56m 30s”).
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla miejsc dziesiętnych (w procentach względem całkowitej wysokości widżetu).
Pozycja pozioma	Wybierz poziome położenie wartości pozycji — z lewej, z prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie wartości pozycji — u góry, na dole lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla wartości pozycji (w procentach względem całkowitej wysokości widżetu). Zwróć uwagę, że rozmiar wartości pozycji ma priorytet; pozostałe elementy muszą ustąpić jej miejsca. Jednak w przypadku wskaźnika zmiany, jeśli wartość jest zbyt duża, zostanie przycięta, aby pokazać wskaźnik zmiany.
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać wartość pozycji pogrubioną czcionką.
Kolor	Wybierz kolor wartości pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Jednostki	
Jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji. Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostkę z konfiguracji pozycji.
Pozycja	Wybierz położenie jednostki pozycji — nad, pod, przed lub za wartością.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla jednostki pozycji (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać jednostkę pozycji pogrubioną czcionką.
Kolor	Wybierz kolor jednostki pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Czas

Te parametry określają, w jaki sposób ma być wyświetlany czas (wartość zegara z historii pozycji).



Pozycja pozioma	Wybierz poziome położenie czasu — po lewej, po prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie czasu — u góry, na dole lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla czasu (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać czas pogrubioną czcionką.

Kolor	Wybierz kolor czasu z próbnika kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Wskaźnik zmiany

Ta sekcja umożliwia wybranie koloru wskaźników zmiany za pomocą selektora kolorów.



Wskaźniki zmiany są następujące:

- ↑ - wartość pozycji wzrosła (dla pozycji numerycznych)
- ↓ - wartość pozycji spadła (dla pozycji numerycznych)
- ↕ - wartość pozycji uległa zmianie (dla pozycji tekstowych i pozycji z mapowaniem wartości)

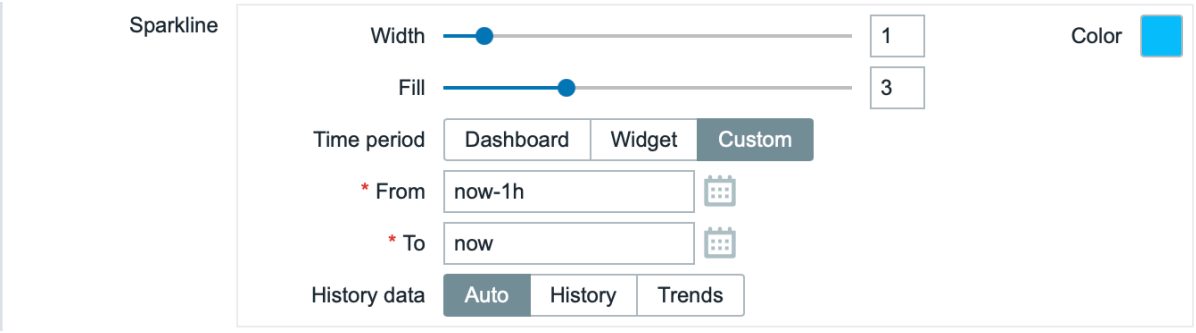
D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.

Pionowy rozmiar wskaźnika zmiany jest równy rozmiarowi wartości (części całkowitej wartości dla pozycji numerycznych).

Należy pamiętać, że wskaźniki wzrostu i spadku nie są wyświetlane przy pojedynczej wartości.

Sparkline

Te parametry określają, w jaki sposób ma być wyświetlany wykres sparkline.



Width	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie wpisując wartość w zakresie od 0 do 10.
Color	Wybierz kolor linii i wypełnienia.
Fill	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie wpisując wartość w zakresie od 0 do 10.
Time period	Określ przedział czasu, z którego wartości mają zostać uwzględnione w wykresie sparkline: **Dashboard** - użyj przedziału czasu dashboardu; **Widget** - użyj przedziału czasu określonego widgetu; **Custom** - użyj niestandardowego przedziału czasu.
Widget	Wybierz widget. Ten parametr będzie wyświetlany tylko wtedy, gdy *Time period* jest ustawiony na "Widget".
From	Wybierz początek przedziału czasu (wartość domyślna `now-1h`). Zobacz składnia czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy *Time period* jest ustawiony na "Custom".
To	Wybierz koniec przedziału czasu (wartość domyślna `now`). Zobacz składnia czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy *Time period* jest ustawiony na "Custom".
History data	Pobierz dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobierz dane historyczne; **Trends** - pobierz dane trendów.

19 Mapa

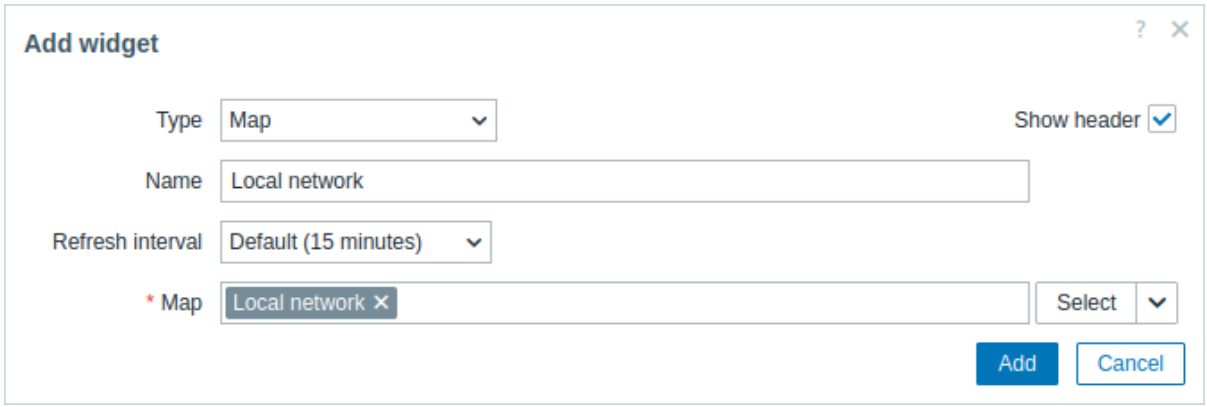
Przegląd

W widżecie mapy możesz wyświetlić:

- pojedynczą skonfigurowaną mapę sieci;
- jedną ze skonfigurowanych map sieci w drzewie nawigacji map (po kliknięciu nazwy mapy w drzewie).

Konfiguracja

Aby skonfigurować, wybierz *Mapa* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów możesz ustawić następujące opcje specyficzne:

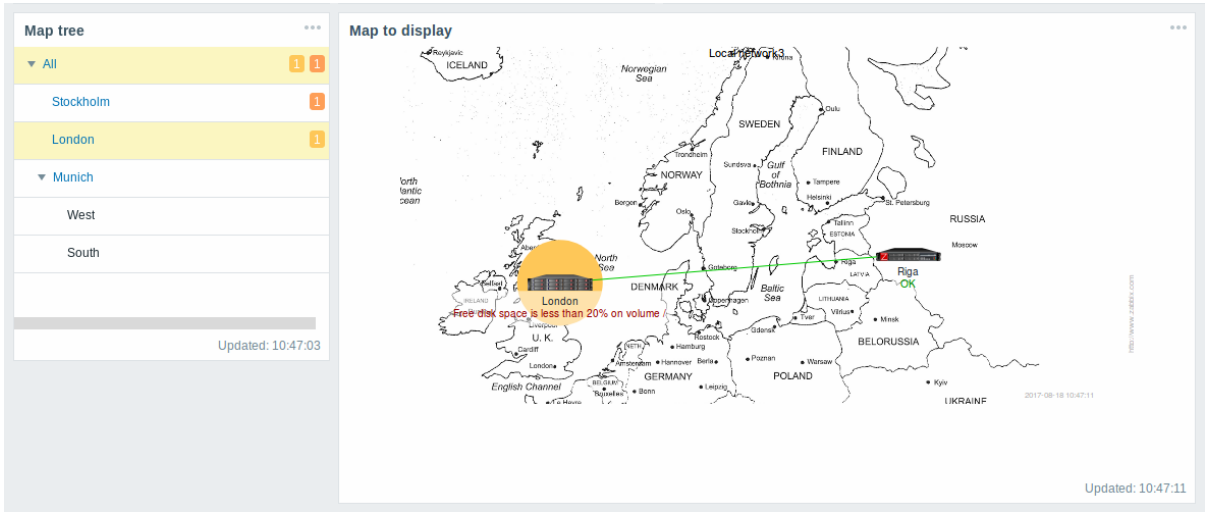
Mapa	Ustaw mapę do wyświetlenia. Alternatywnie wybierz zgodny widget jako źródło danych dla wyświetlanej mapy. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy lub widgetu zostanie wyświetlona rozwijana lista pasujących map lub widgetów.

20 Drzewo nawigacji mapy

Omówienie

Ten widżet umożliwia budowanie hierarchii istniejących map, a także wyświetlanie statystyk problemów dla każdej dołączonej mapy i grupy map.

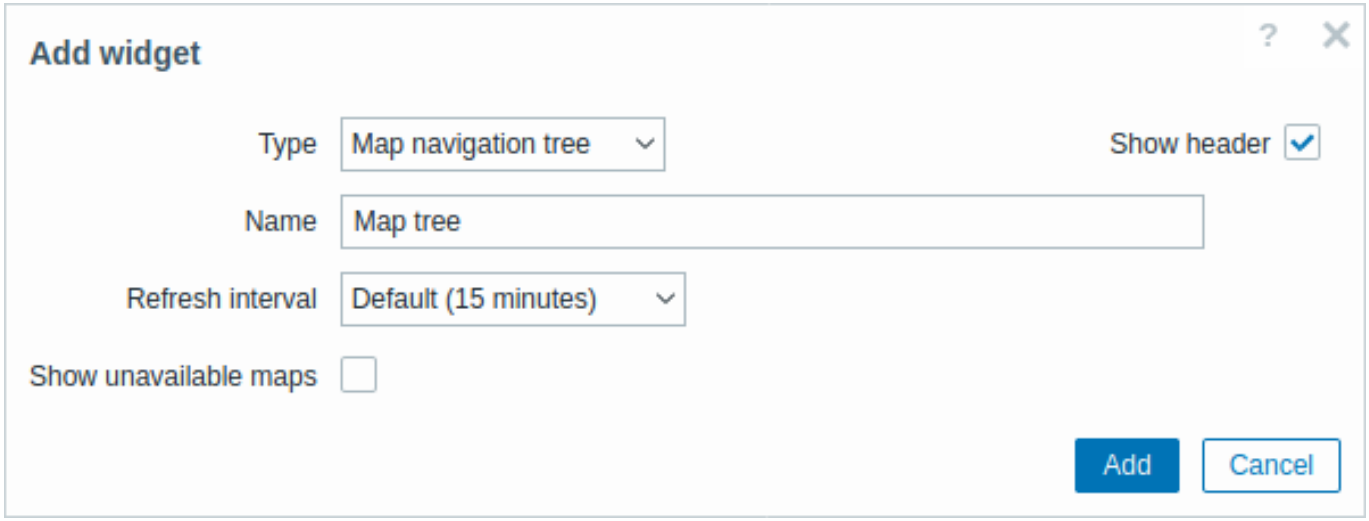
Staje się jeszcze bardziej użyteczny, jeśli połączysz widżet *Map* z drzewem nawigacji. W takim przypadku kliknięcie nazwy mapy w drzewie nawigacji powoduje wyświetlenie całej mapy w widżecie *Map*.



Statystyki dla mapy najwyższego poziomu w hierarchii wyświetlają sumę problemów wszystkich podmap oraz własnych problemów.

Konfiguracja

Aby skonfigurować widżet, wybierz *Map navigation tree* jako typ:

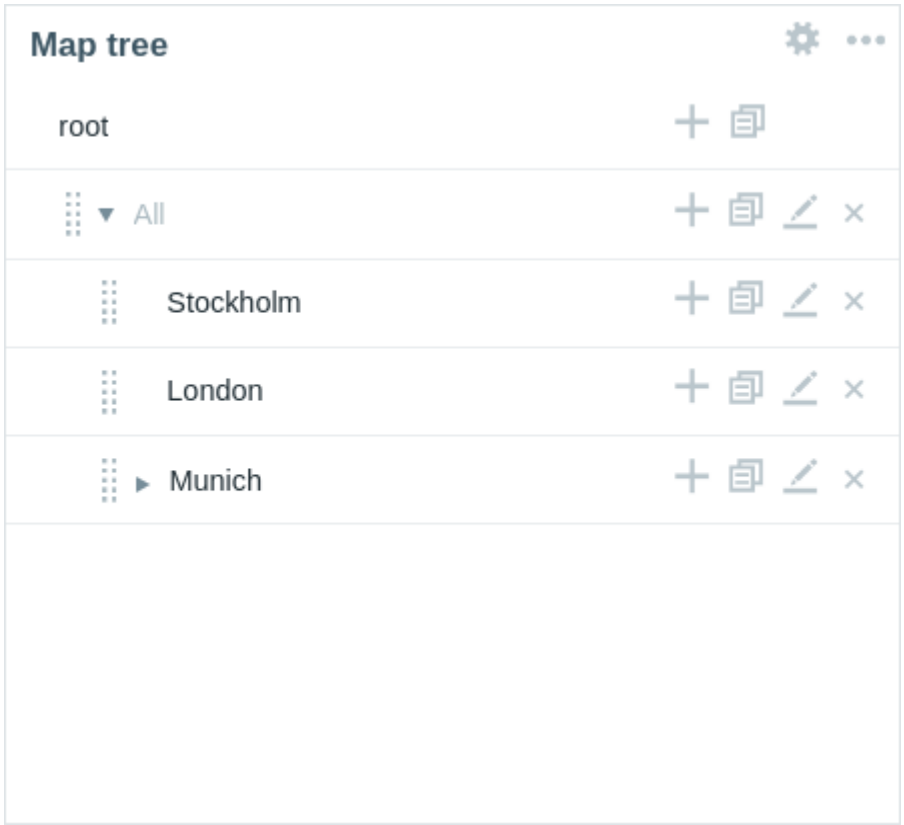


Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Pokaż niedostępne mapy	Zaznacz to pole wyboru, aby wyświetlać mapy, do których użytkownik nie ma uprawnień odczytu. Niedostępne mapy w drzewie nawigacji będą wyświetlane z wyszarzoną ikoną. Zwróć uwagę, że jeśli to pole wyboru jest zaznaczone, dostępne submaps są wyświetlane nawet wtedy, gdy mapa poziomu nadrzędnego jest niedostępna. Jeśli pole nie jest zaznaczone, dostępne submaps niedostępnej mapy nadrzędnej nie będą wyświetlane w ogóle. Liczba problemów jest obliczana na podstawie dostępnych map i dostępnych elementów mapy.

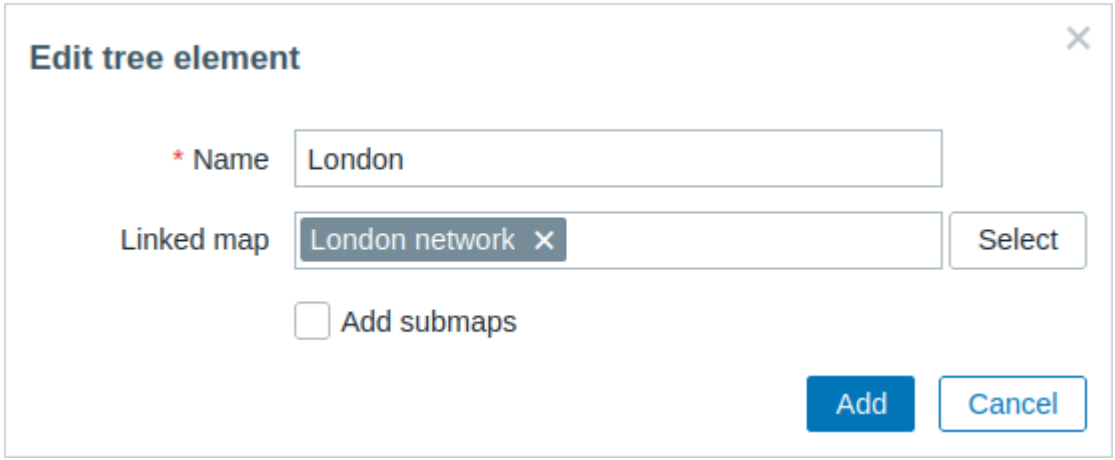
Elementy drzewa nawigacji są wyświetlane na liście. Możesz:

- przeciągnąć element (wraz z jego elementami podrzędnymi) w nowe miejsce na liście;
- rozwinąć lub zwinąć element, aby wyświetlić lub ukryć jego elementy podrzędne;
- dodać element podrzędny (z podlinkowaną mapą lub bez niej) do elementu;
- dodać wiele elementów podrzędnych (z podlinkowanymi mapami) do elementu;
- edytować element;
- usunąć element (wraz z jego elementami podrzędnymi).



Konfiguracja elementu

Aby skonfigurować element drzewa nawigacji, dodaj nowy element lub edytuj istniejący element.



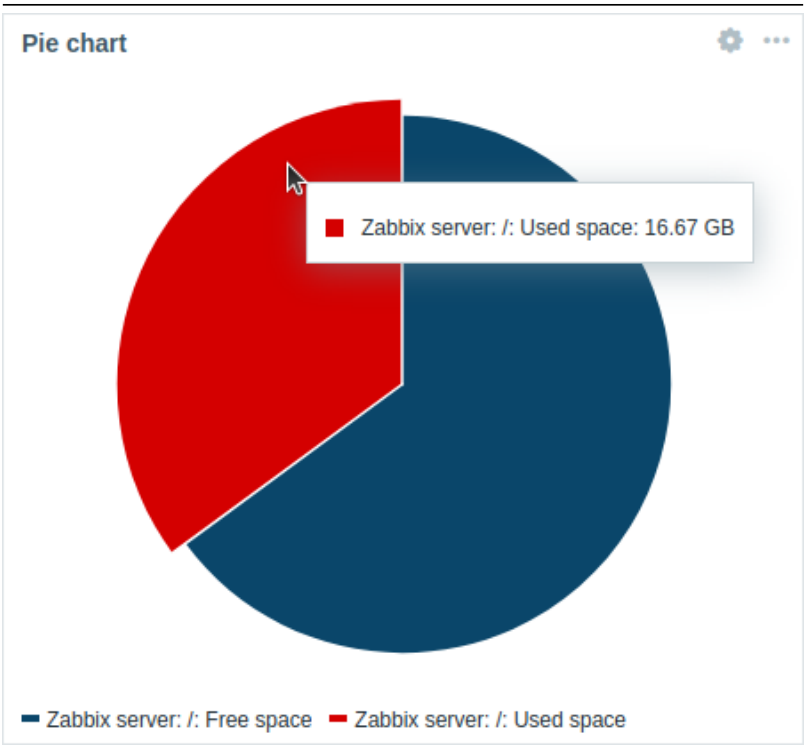
Dostępne są następujące parametry konfiguracji elementu drzewa nawigacji:

Nazwa	Wprowadź nazwę elementu drzewa nawigacji.
Powiązana mapa	Wybierz mapę, którą chcesz powiązać z elementem drzewa nawigacji. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy zostanie wyświetlona rozwijana lista pasujących map.
Dodaj podmapy	Zaznacz to pole wyboru, aby dodać podmapy powiązanej mapy jako elementy podrzędne elementu drzewa nawigacji.

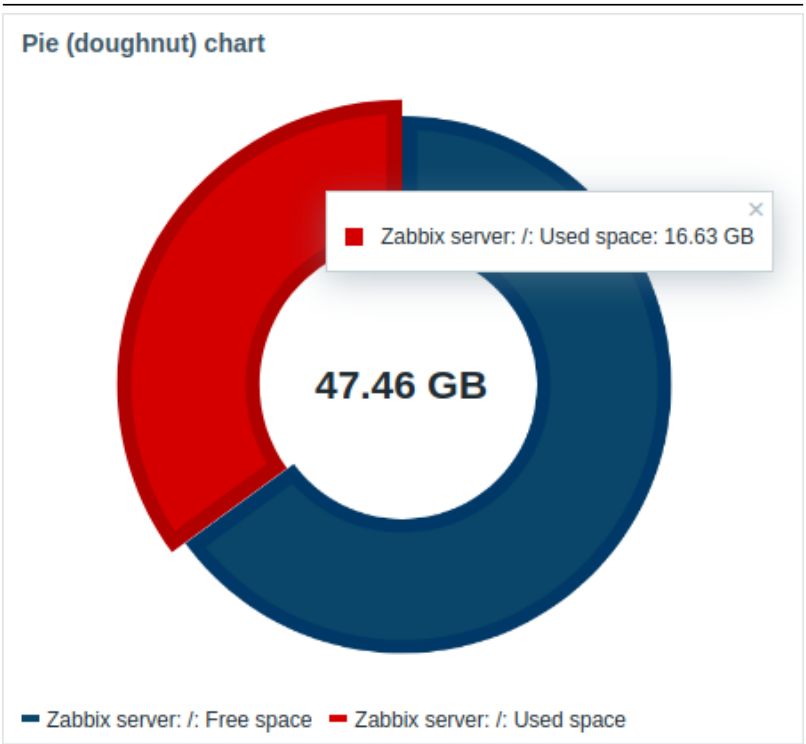
21 Wykres kołowy

Omówienie

Widżet wykresu kołowego umożliwia wyświetlanie wartości wybranych pozycji jako wykresu **kołowego** lub **pierścieniowego**.



Wykres kołowy.



Wykres pierścieniowy.

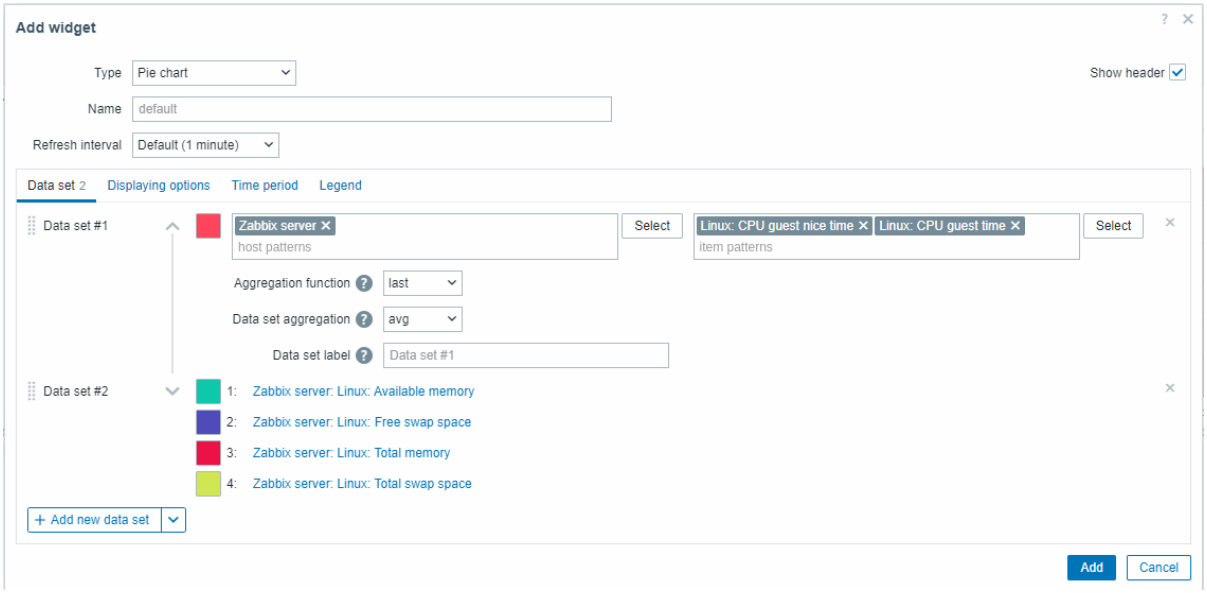
Po najechaniu kursorem na sektor, sektor znajdujący się w fokusie powiększa się na zewnątrz, a legenda dla tego sektora jest wyświetlana. Kliknięcie sektora znajdującego się w fokusie powoduje, że efekt wysunięcia pozostaje na stałe, aż do zamknięcia za pomocą „x".

Wykresy są rysowane przy użyciu techniki wektorowego tworzenia obrazów.

Informacje wyświetlane w widżecie *Wykres kołowy* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu widżetu.

Konfiguracja

Aby skonfigurować, wybierz *Pie chart* jako typ:



Zestaw danych

Karta **Zestaw danych** umożliwia wybór danych dla wykresu kołowego przez dodawanie zestawów danych. Można dodać dwa typy zestawów danych:

- *Wzorce pozycji* - wyświetlane są dane z pasujących pozycji. Możesz wybrać jeden kolor bazowy lub wskazać wiersz palety, aby przypisać różne kolory do każdej dopasowanej pozycji.
- *Lista pozycji* - wyświetlane są dane z wybranych pozycji. Możesz osobno wybrać kolor każdej pozycji za pomocą selektora.

Domyślnie dodawany jest zestaw danych typu *Wzorce pozycji*.

Zestaw danych	Dla zestawu danych **Wzorce pozycji**: Wybierz lub wprowadź wzorce hosta i pozycji; na wykresie kołowym zostaną wyświetlone dane pozycji pasujących do wprowadzonych wzorców; można wyświetlić maksymalnie 50 pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub więcej znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Symbol wieloznaczny jest zawsze interpretowany, więc nie można na przykład dodać osobno pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (na przykład *item2*, *item3*). Określenie wzorców hosta i pozycji jest obowiązkowe dla zestawów danych typu ”Wzorce pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Dla zestawu danych **Lista pozycji**: Wybierz pozycje dla wykresu kołowego, klikając przycisk *Dodaj pozycję*. Lista rozwijana *Typ* obok nazwy pozycji umożliwia wybór typu wyświetlania dla każdej pozycji: **Normalny** - wartość pozycji jest reprezentowana proporcjonalnie na wykresie kołowym (domyślnie); **Całkowity** - wartość pozycji zajmuje cały wykres kołowy. Należy pamiętać, że na jednym wykresie kołowym może istnieć tylko jedna pozycja typu ”Całkowity” i zostanie ona umieszczona jako pierwsza w legendzie wykresu kołowego. Jeśli pozycja zostanie ustawiona jako ”Całkowity”, parametr *Agregacja zestawu danych* (patrz poniżej) zostanie wyłączony i ustawiony na ”nieużywany”. Możesz także wybrać zgodne widżety jako źródło danych dla pozycji, klikając przycisk *Dodaj widżet*. Te same opcje mają zastosowanie do widżetów, co do pojedynczych pozycji. Określenie pozycji lub widżetów jest obowiązkowe dla zestawów danych typu ”Lista pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Pamiętaj, że dozwolone są tylko numeryczne typy pozycji. Podczas konfigurowania widżetu na pulpicie szablonu, parametr określający wzorce hostów nie jest dostępny, a parametr umożliwiający określenie listy pozycji pozwala wybrać tylko pozycje skonfigurowane w szablonie.
Funkcja agregacji	Określ, której funkcji agregacji użyć dla każdej pozycji w zestawie danych w wybranym *Okresie czasu*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **sum** - wyświetl sumę wartości; **count** - wyświetl liczbę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość (domyślnie). Agregacja umożliwia wyświetlenie wartości zagregowanej dla wybranego interwału (5 minut, godzina, dzień). Tylko dane numeryczne mogą być wyświetlane dla *min*, *max*, *avg* i *sum*. Dla *count* dane nienumeryczne zostaną przekształcone na numeryczne.
Agregacja zestawu danych	Określ, której funkcji agregacji użyć dla całego zestawu danych: **nieużywany** - brak agregacji, pozycje są wyświetlane osobno (domyślnie); **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **sum** - wyświetl sumę wartości; **count** - wyświetl liczbę wartości. Agregacja umożliwia wyświetlenie wartości zagregowanej dla interwału (5 minut, godzina, dzień) wybranego na karcie *Okres czasu* lub używanego dla całego pulpitu.

Etykieta zestawu danych	Określ własną etykietę dla zestawu danych. Etykieta jest wyświetlana w konfiguracji zestawu danych oraz w legendzie wykresu kołowego (dla zestawów danych agregowanych). Wszystkie zestawy danych są numerowane, w tym te z określoną *Etykietą zestawu danych*. Jeśli nie zostanie określona etykieta, zestaw danych zostanie automatycznie oznaczony zgodnie ze swoim numerem (np. ”Zestaw danych #2”, ”Zestaw danych #3” itd.). Numeracja zestawów danych jest przeliczana ponownie po zmianie kolejności/przeciągnięciu zestawów danych. Zbyt długie etykiety zestawów danych zostaną skrócone, aby zmieściły się w miejscu wyświetlania (np. ”Number of proc...”).

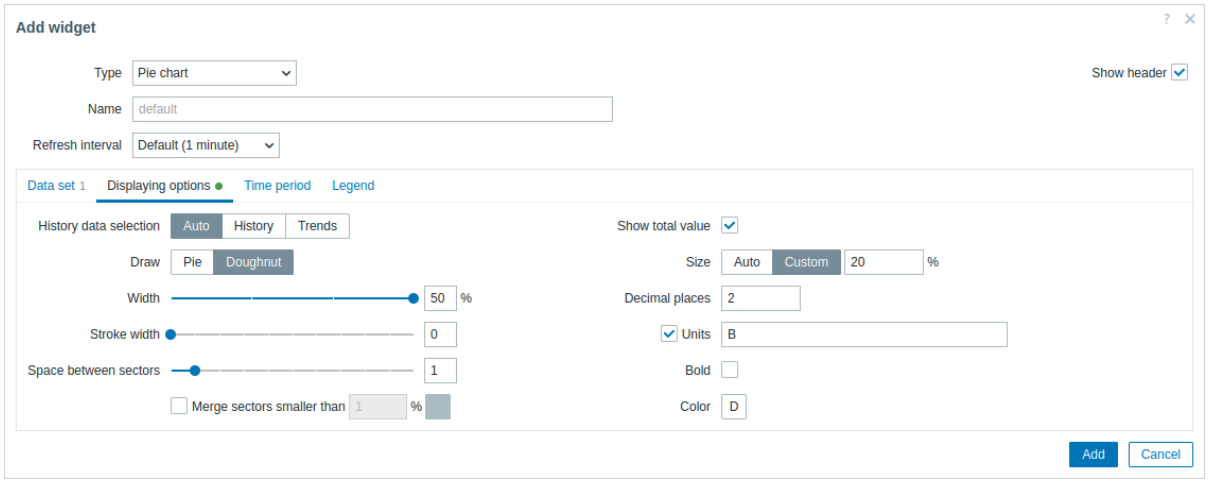
Szczegóły konfiguracji zestawu danych

Istniejące zestawy danych są wyświetlane na liście. Możesz zmieniać ich kolejność, rozwijać/zwijać, zmieniać kolory oraz klonować te zestawy danych.

Więcej informacji można znaleźć w szczegółach konfiguracji zestawu danych w widżecie *Graph*. Te szczegóły mają również zastosowanie do widżetu *Pie chart*.

Opcje wyświetlania

Zakładka **Opcje wyświetlania** umożliwia zdefiniowanie opcji wyboru danych historycznych i wizualizacji dla wykresu kołowego:

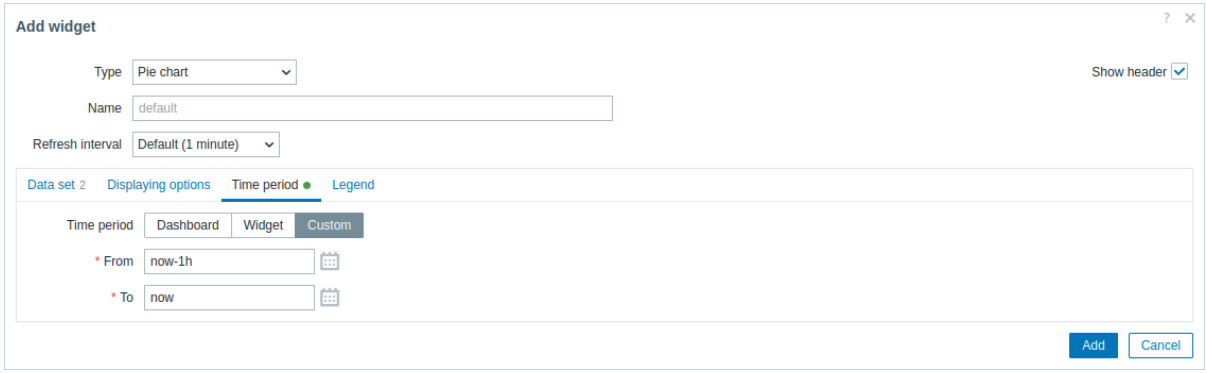


Wybór danych historycznych	Wybierz źródło danych: **Auto** - dane są pobierane zgodnie z klasycznym algorytmem (domyślnie); **Historia** - dane z historii; **Trendy** - dane z trendów.
Rysuj	Wybierz styl wizualizacji wykresu kołowego: **Kołowy** - pełny wykres kołowy (sektory zajmują 100% promienia); **Pierścieniowy** - wykres kołowy z pustym okręgiem pośrodku (sektory zajmują do 50% promienia).
Odstęp między sektorami	Wybierz rozmiar odstępu (w jednostkach od 0 do 10) między sektorami (domyślnie „1”).
Scal sektory mniejsze niż N%	Zaznacz pole wyboru, aby scalać sektory mniejsze niż N%. Jeśli opcja jest włączona, wybierz kolor dla scalonych sektorów oraz próg procentowy (N) dla scalania małych sektorów.
Styl rysowania: ***Pierścieniowy***	
Szerokość	Wybierz szerokość wykresu pierścieniowego: 20, 30, 40 lub 50% (domyślnie) promienia.

Szerokość obramowania	Wybierz szerokość obramowania sektora wykresu pierścieniowego (0-10).
Pokaż wartość całkowitą	Zaznacz pole wyboru, aby wyświetlić wartość całkowitą pośrodku wykresu pierścieniowego.
Rozmiar	Wybierz opcję rozmiaru dla wartości całkowitej: **Auto** - rozmiar tekstu jest dobierany automatycznie, aby czytelnie zmieścił się w środku wykresu pierścieniowego; **Własny** - określ rozmiar tekstu jako procent wysokości całkowitej widżetu.
Miejsca po przecinku	Określ liczbę miejsc po przecinku dla wartości całkowitej (0-6).
Jednostki	Określ jednostki dla wartości całkowitej.
Pogrubienie	Zaznacz pole wyboru, aby wyświetlić wartość całkowitą pogrubioną czcionką.
Kolor	Wybierz kolor dla wartości całkowitej.

Okres czasu

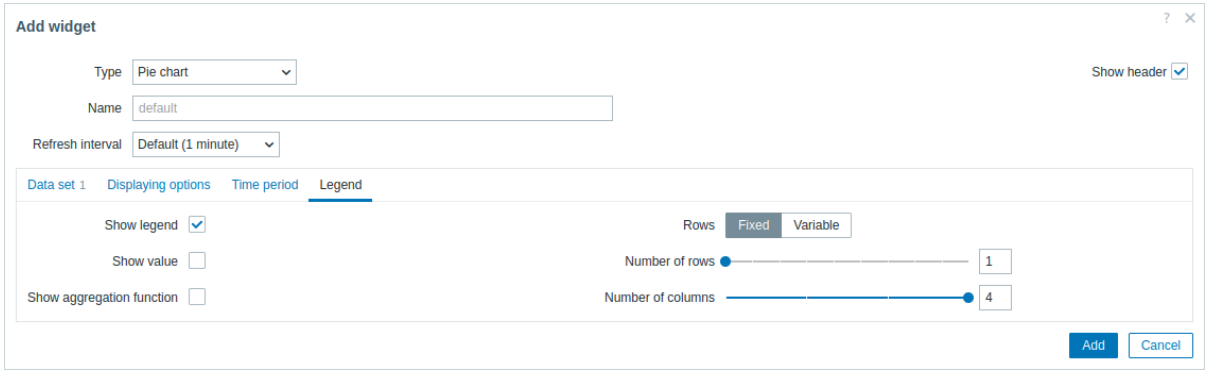
Karta **Okres czasu** umożliwia ustawienie niestandardowego okresu czasu dla ustawień agregacji wykresu kołowego:



Okres czasu	Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Okres czasu* jako źródło danych; **Widget** - ustaw zgodny widget określony w parametrze *Widget* jako źródło danych; **Niestandardowy** - ustaw okres czasu określony w parametrach *Od* i *Do* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu myszą.
Widget	Wprowadź lub wybierz zgodny widget (*Graph*, *Graph (classic)*, *Graph prototype*) jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Widget”.
Od	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na ”Niestandardowy”.

Legenda

Zakładka **Legenda** umożliwia dostosowanie legendy wykresu kołowego:



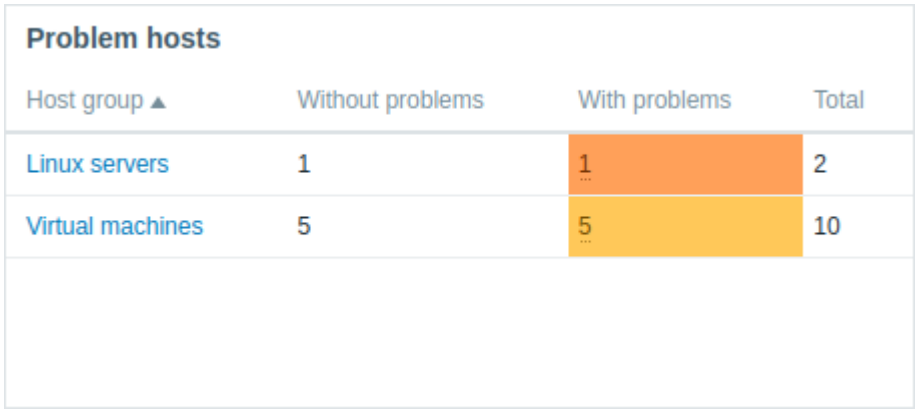
Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresie kołowym (domyślnie zaznaczone).
Pokaż wartość	Zaznacz to pole wyboru, aby wyświetlić wartość pozycji w legendzie.
Pokaż funkcję agregacji	Zaznacz to pole wyboru, aby wyświetlić funkcję agregacji w legendzie.
Wiersze	Wybierz tryb wyświetlania wierszy legendy: **Stałe** — liczba wyświetlanych wierszy jest określana przez wartość parametru *Liczba wierszy*; **Zmienne** — liczba wyświetlanych wierszy jest określana przez liczbę skonfigurowanych pozycji, bez przekraczania wartości parametru *Maksymalna liczba wierszy*.
Liczba wierszy/ Maksymalna liczba wierszy	Jeśli dla parametru *Wiersze* ustawiono wartość „Stałe”, określ liczbę wyświetlanych wierszy legendy (1-10). Jeśli dla parametru *Wiersze* ustawiono wartość „Zmienne”, określ maksymalną liczbę wyświetlanych wierszy legendy (1-10).
Liczba kolumn	Określ liczbę wyświetlanych kolumn legendy (1-4). Ten parametr jest dostępny, jeśli opcja *Pokaż wartość* jest odznaczona.

22 Problematyczne hosty

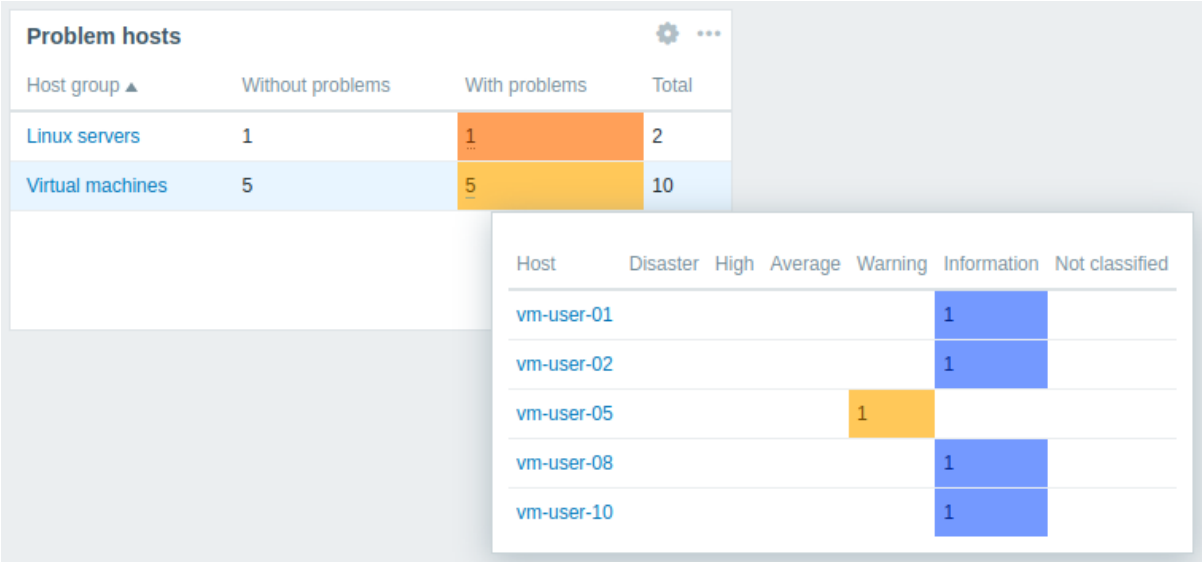
Przegląd

W widżecie hostów z problemami można wyświetlać liczbę problemów według grup hostów oraz najwyższy poziom ważności problemu w grupie.

Liczba problemów jest wyświetlana tylko dla problemów przyczynowych.

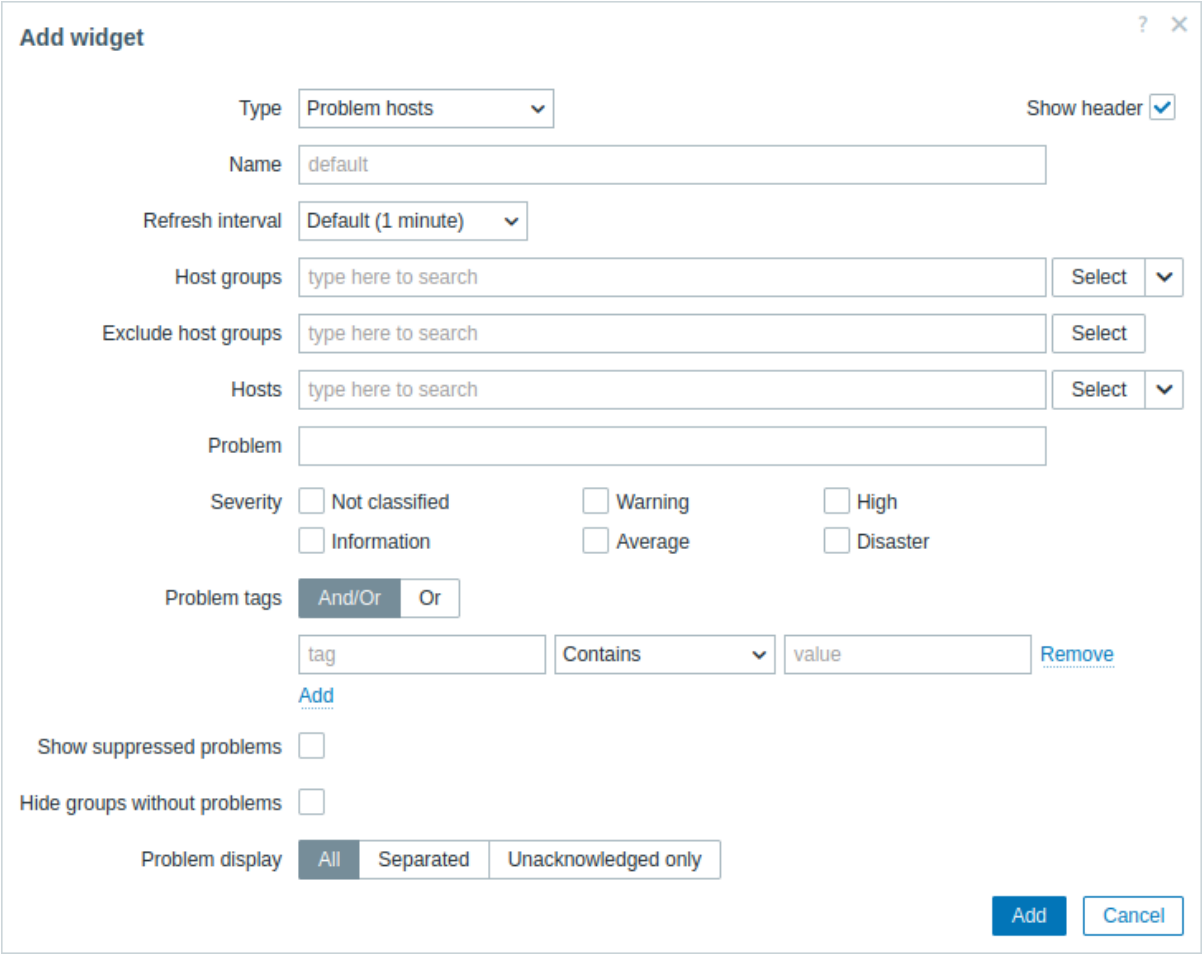


Po najechaniu kursorem na liczbę problemów wyświetlana jest lista hostów z problemami:



Konfiguracja

Aby skonfigurować, wybierz *Problem hosts* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów, możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wprowadzone żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Exclude host groups	Wybierz grupy hostów, które mają zostać ukryte w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą należeć do Grupy A, a hosty 002, 003 również do Grupy B. Jeśli jednocześnie wybierzemy *show* dla Grupy A i *exclude* dla Grupy B, na pulpicie zostaną wyświetlone tylko dane hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosts	Wybierz hosty, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie zostaną wprowadzone żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Problem	Możesz ograniczyć liczbę hostów z problemami wyświetlanych według nazwy problemu. Jeśli wpiszesz tutaj ciąg znaków, wyświetlone zostaną tylko te hosty, które mają problemy, których nazwa zawiera podany ciąg. Makra nie są rozwijane.
Severity	Zaznacz poziomy ważności problemów, aby filtrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlone zostaną wszystkie problemy.
Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby tłumione (niewyświetlane) z powodu konserwacji hosta.
Hide groups without problems	Zaznacz opcję *Hide groups without problems*, aby ukryć w widżecie dane z grup hostów bez problemów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Problem display	Wyświetl liczbę problemów jako: **All** - zostanie wyświetlona pełna liczba problemów; **Separated** - liczba niepotwierdzonych problemów zostanie wyświetlona oddzielnie jako część całkowitej liczby problemów; **Unacknowledged only** - zostanie wyświetlona tylko liczba niepotwierdzonych problemów.

23 Problemy

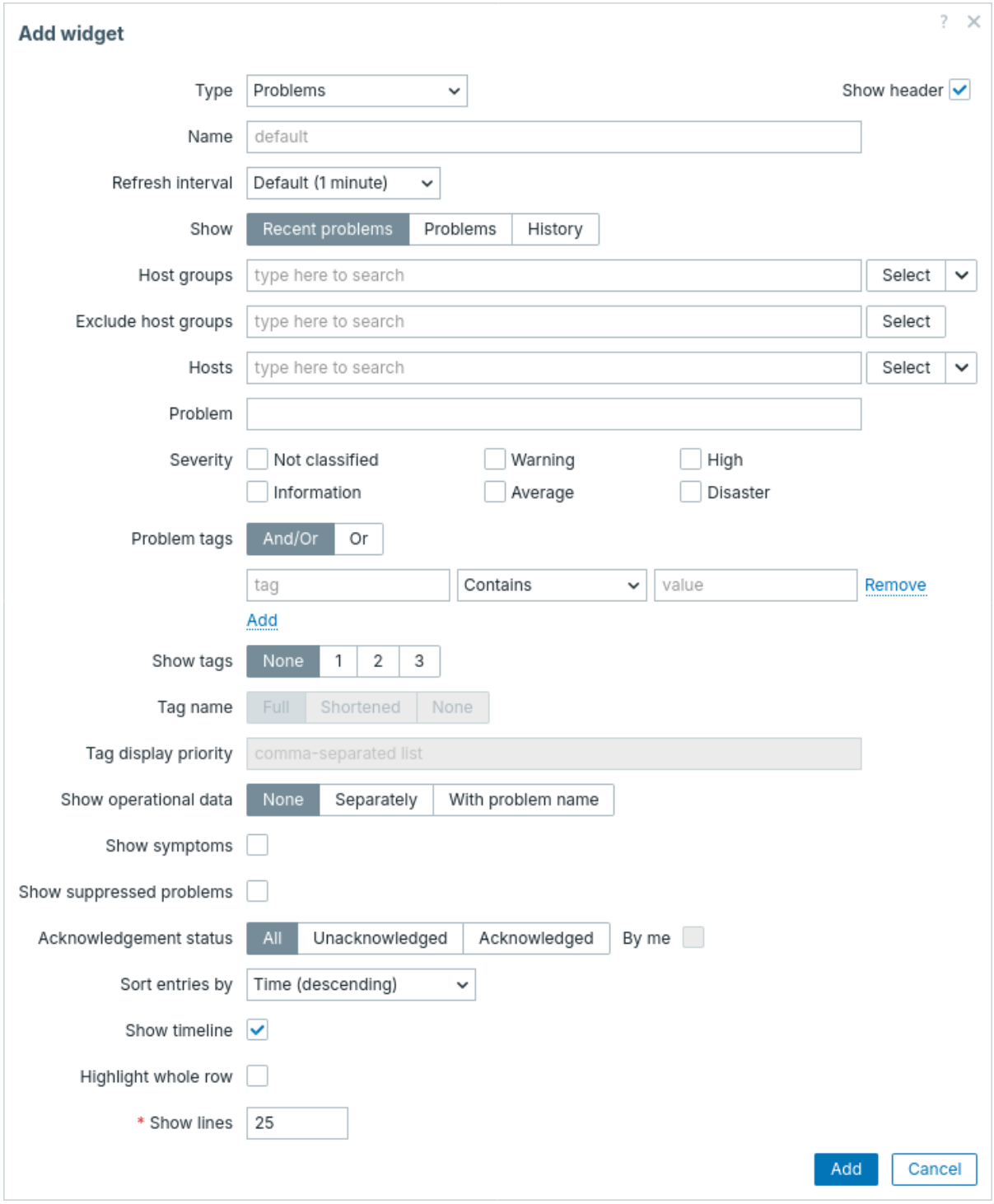
Przegląd

W tym widżecie można wyświetlać bieżące problemy. Informacje w tym widżecie są podobne do *Monitoring* > *Problemy*.

Można wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować, wybierz *Problems* jako typ:



Możesz ograniczyć liczbę problemów wyświetlanych w widżecie na różne sposoby - według statusu problemu, nazwy problemu, ważności, grupy hostów, hosta, tagu zdarzenia, statusu potwierdzenia itd.

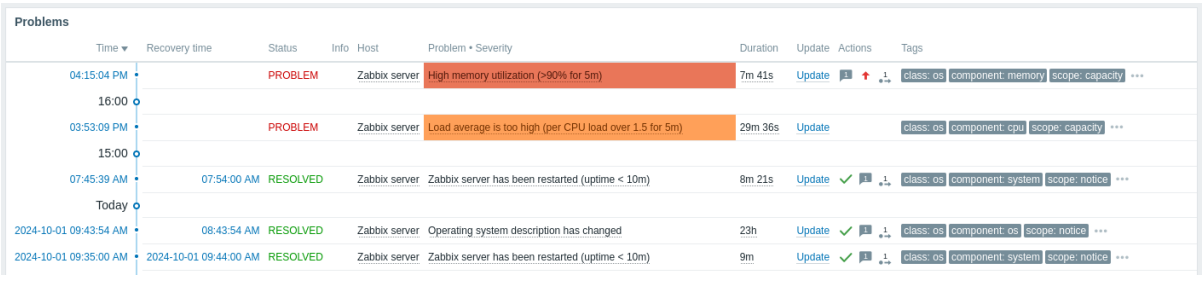
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Show	Filtruj według statusu problemu: **Recent problems** - wyświetlane są nierozwiązane i niedawno rozwiązane problemy (domyślnie); **Problems** - wyświetlane są nierozwiązane problemy; **History** - wyświetlana jest historia wszystkich zdarzeń.

Host groups	Wybierz grupy hostów, których problemy mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Problemy z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną podane żadne grupy hostów, będą wyświetlane problemy ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Exclude host groups	Wybierz grupy hostów, których problemy mają być ukrywane w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Problemy z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą znajdować się w Grupie A, a hosty 002, 003 również w Grupie B. Jeśli wybierzemy jednocześnie *show* Grupę A i *exclude* Grupę B, w widżecie będą wyświetlane tylko problemy z hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts	Wybierz hosty, których problemy mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana pasujących hostów. Jeśli nie zostaną podane żadne hosty, będą wyświetlane problemy ze wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Problem	Możesz ograniczyć liczbę wyświetlanych problemów według ich nazwy. Jeśli wpiszesz tutaj ciąg znaków, zostaną wyświetlone tylko te problemy, których nazwa zawiera wpisany ciąg. Makra nie są rozwijane.
Severity	Zaznacz ważności problemów, aby filtrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczona żadna ważność, wyświetlane będą wszystkie problemy.
Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wpisany ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wpisany ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek. Po zastosowaniu filtra tagi określone tutaj będą wyświetlane jako pierwsze wraz z problemem, chyba że zostanie to zastąpione przez listę *Tag display priority* (patrz poniżej).
Show tags	Wybierz liczbę wyświetlanych tagów: **None** - brak kolumny *Tags*; **1** - kolumna *Tags* zawiera jeden tag; **2** - kolumna *Tags* zawiera dwa tagi; **3** - kolumna *Tags* zawiera trzy tagi. Aby zobaczyć wszystkie tagi problemu, najedź kursorem myszy na ikonę trzech kropek.

Tag name	Wybierz tryb wyświetlania nazw tagów: **Full** - nazwy i wartości tagów są wyświetlane w pełnej postaci; **Shortened** - nazwy tagów są skrócone do 3 znaków, ale wartości tagów są wyświetlane w pełnej postaci; **None** - wyświetlane są tylko wartości tagów; bez nazw.
Tag display priority	Wprowadź priorytet wyświetlania tagów dla problemu jako listę tagów rozdzielonych przecinkami. Należy używać tylko nazw tagów, bez wartości. Przykład: `Services,Applications,Application` Tagi z tej listy będą zawsze wyświetlane jako pierwsze, zastępując naturalną kolejność alfabetyczną.
Show operational data	Wybierz tryb wyświetlania danych operacyjnych: **None** - dane operacyjne nie są wyświetlane; **Separately** - dane operacyjne są wyświetlane w osobnej kolumnie; **With problem name** - dane operacyjne są dołączane do nazwy problemu, z użyciem nawiasów dla danych operacyjnych.
Show symptoms	Zaznacz pole wyboru, aby wyświetlać w osobnym wierszu problemy sklasyfikowane jako symptomy.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby tłumione (niewyświetlane) z powodu konserwacji hosta lub pojedynczego tłumienia problemów.
Acknowledgement status	Filtruj, aby wyświetlać wszystkie problemy, tylko problemy niepotwierdzone lub tylko problemy potwierdzone. Zaznacz dodatkowe pole wyboru, aby odfiltrować problemy kiedykolwiek przez Ciebie potwierdzone.
Sort entries by	Sortuj wpisy według: **Time** (malejąco lub rosnąco); **Severity** (malejąco lub rosnąco); **Problem name** (malejąco lub rosnąco); **Host** (malejąco lub rosnąco). Sortowanie wpisów według **Host** (malejąco lub rosnąco) nie jest dostępne podczas konfigurowania widżetu na szablonowym dashboardzie.
Show timeline	Zaznacz pole wyboru, aby wyświetlić wizualną oś czasu.
Highlight whole row	Zaznacz pole wyboru, aby podświetlić cały wiersz dla nierozwiązanych problemów. Do podświetlenia używany jest kolor ważności problemu. *Highlight whole row* nie jest dostępne w motywach o wysokim kontraście. Ta opcja jest obsługiwana od Zabbix 7.4.3.
Show lines	Określ liczbę wierszy problemów do wyświetlenia.

Korzystanie z widżetu

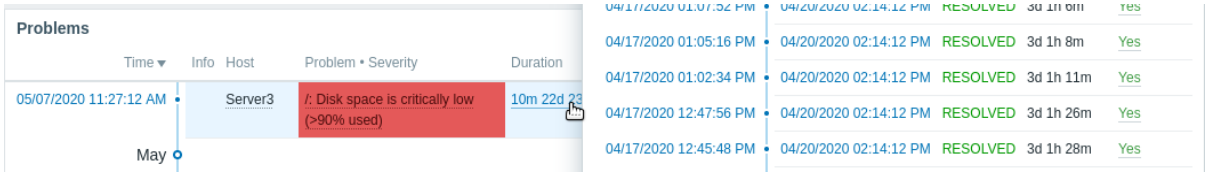


Widżet Problemów oferuje szybki dostęp do dodatkowych informacji:

- Kliknij datę problemu (w kolumnie Czas) lub datę odzyskania (w kolumnie Czas odzyskania), aby wyświetlić szczegóły zdarzenia.
- Jeśli pole Informacje nie jest puste, najedź kursorem na wyświetlaną ikonę, aby zobaczyć dodatkowe szczegóły.
- Kliknij nazwę hosta, aby otworzyć menu hosta.
- Kliknij nazwę problemu, aby otworzyć menu zdarzenia.
- Najedź kursorem na czas trwania problemu lub kliknij go, aby wyświetlić wyskakujące okno zdarzenia problemu.
- Naciśnięcie przycisku Aktualizuj otwiera okno Aktualizacja problemu.
- Najedź kursorem na szarą ikonę strzałki w kolumnie Akcje lub kliknij ją, aby wyświetlić listę wykonanych akcji.

Wyskakujące okno zdarzenia problemu

Wyskakujące okno zdarzenia problemu zawiera listę zdarzeń problemu dla tego wyzwalacza oraz, jeśli zostały zdefiniowane, opis wyzwalacza i klikalny adres URL.



Aby wyświetlić wyskakujące okno zdarzenia problemu:

- Najedź kursorem myszy na czas trwania problemu w kolumnie *Czas trwania* widżetu *Problemy*. Okno zniknie po odsunięciu kursora od czasu trwania.
- Kliknij czas trwania w kolumnie *Czas trwania* widżetu *Problemy*. Okno zniknie dopiero po ponownym kliknięciu czasu trwania.

24 Problemy według ważności

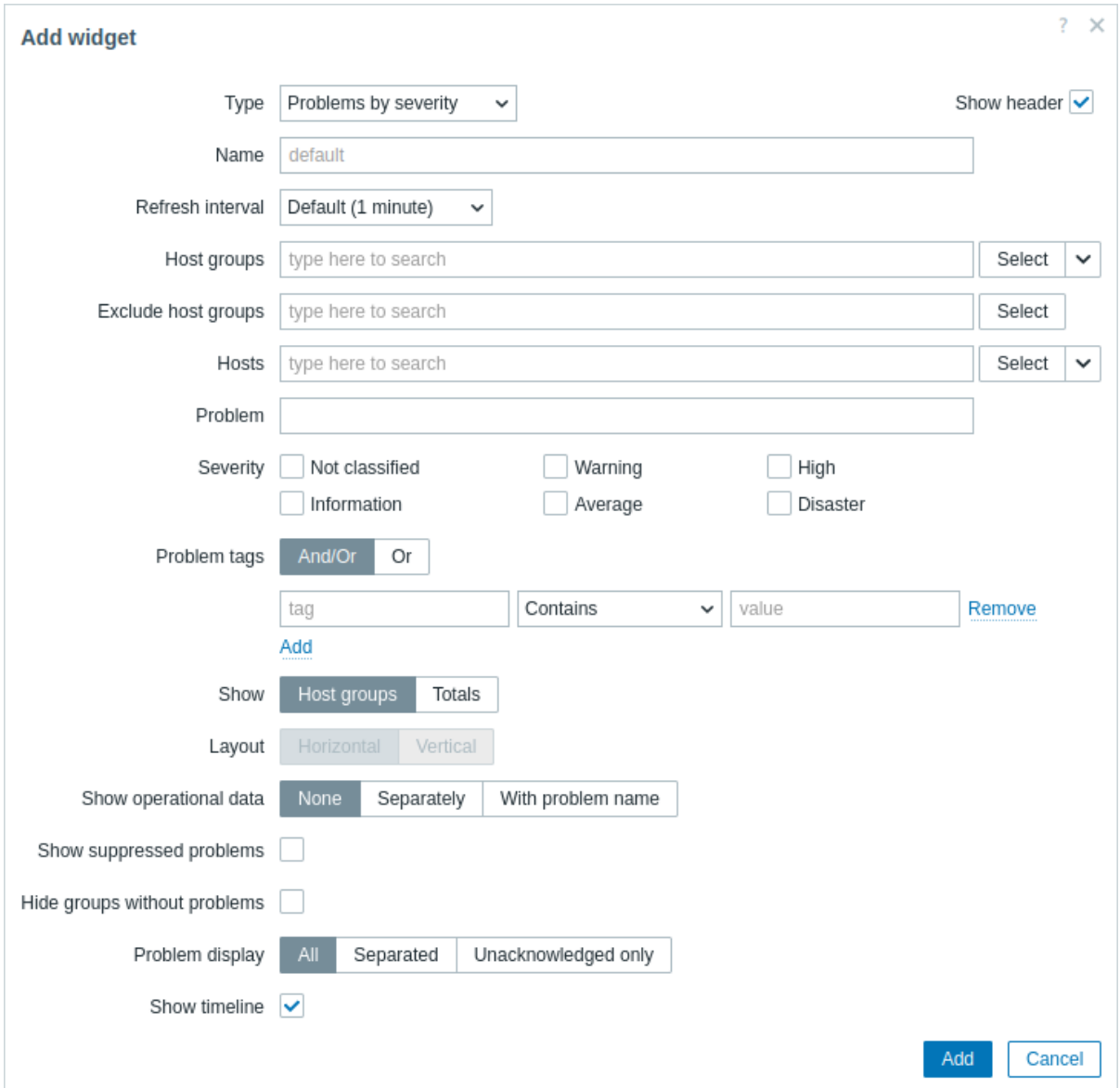
Przegląd

W tym widżecie można wyświetlić liczbę problemów według poziomu ważności. Można ograniczyć, które hosty i wyzwalacze są wyświetlane w widżecie, oraz określić, w jaki sposób wyświetlana jest liczba problemów.

Liczba problemów jest wyświetlana tylko dla problemów przyczynowych.

Konfiguracja

Aby skonfigurować widget, wybierz *Problems by severity* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wprowadzone żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Wyklucz grupy hostów	Wybierz grupy hostów, które mają zostać ukryte w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą należeć do Grupy A, a hosty 002, 003 również do Grupy B. Jeśli jednocześnie wybierzemy *pokazywanie* Grupy A i *wykluczenie* Grupy B, na pulpicie zostaną wyświetlone tylko dane hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Hosty	Wybierz hosty, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie zostaną wprowadzone żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Problem	Możesz ograniczyć liczbę hostów z problemami wyświetlanych na podstawie nazwy problemu. Jeśli wpiszesz tutaj ciąg znaków, wyświetlane będą tylko te hosty, których problemy mają w nazwie podany ciąg. Makra nie są rozwijane.
Ważność	Zaznacz poziomy ważności problemów, aby filtrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlane będą wszystkie problemy.
Tagi problemów	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem *Or*; **Or** - wystarczy spełnienie jednego warunku.
Pokaż	Wybierz opcję wyświetlania: **Host groups** - wyświetl problemy według grup hostów; **Totals** - wyświetl łączną liczbę problemów dla wszystkich wybranych grup hostów w kolorowych blokach odpowiadających ważności problemu. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu, a wyświetlana będzie tylko łączna liczba problemów.
Układ	Wybierz opcję układu: **Horizontal** - kolorowe bloki z sumami będą wyświetlane poziomo; **Vertical** - kolorowe bloki z sumami będą wyświetlane pionowo. Ten parametr jest dostępny, jeśli *Pokaż* jest ustawione na ”Totals”.

Pokaż dane operacyjne	Wyświetl dane operacyjne (zobacz opis *Operational data* w *Monitoring* > *Problems*) jako: **None** - dane operacyjne nie będą wyświetlane **Separately** - dane operacyjne będą wyświetlane w osobnym wierszu **With problem name** - dane operacyjne będą wyświetlane razem z nazwą problemu.
Pokaż wyciszone problemy	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby wyciszone (niepokazywane) z powodu konserwacji hosta.
Ukryj grupy bez problemów	Zaznacz opcję *Ukryj grupy bez problemów*, aby ukryć w widżecie dane z grup hostów bez problemów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Wyświetlanie problemów	Wyświetl liczbę problemów jako: **All** - zostanie wyświetlona pełna liczba problemów; **Separated** - liczba niepotwierdzonych problemów będzie wyświetlana oddzielnie jako część całkowitej liczby problemów; **Unacknowledged only** - wyświetlana będzie tylko liczba niepotwierdzonych problemów.
Pokaż oś czasu	Zaznacz pole wyboru, aby wyświetlić wizualną oś czasu.

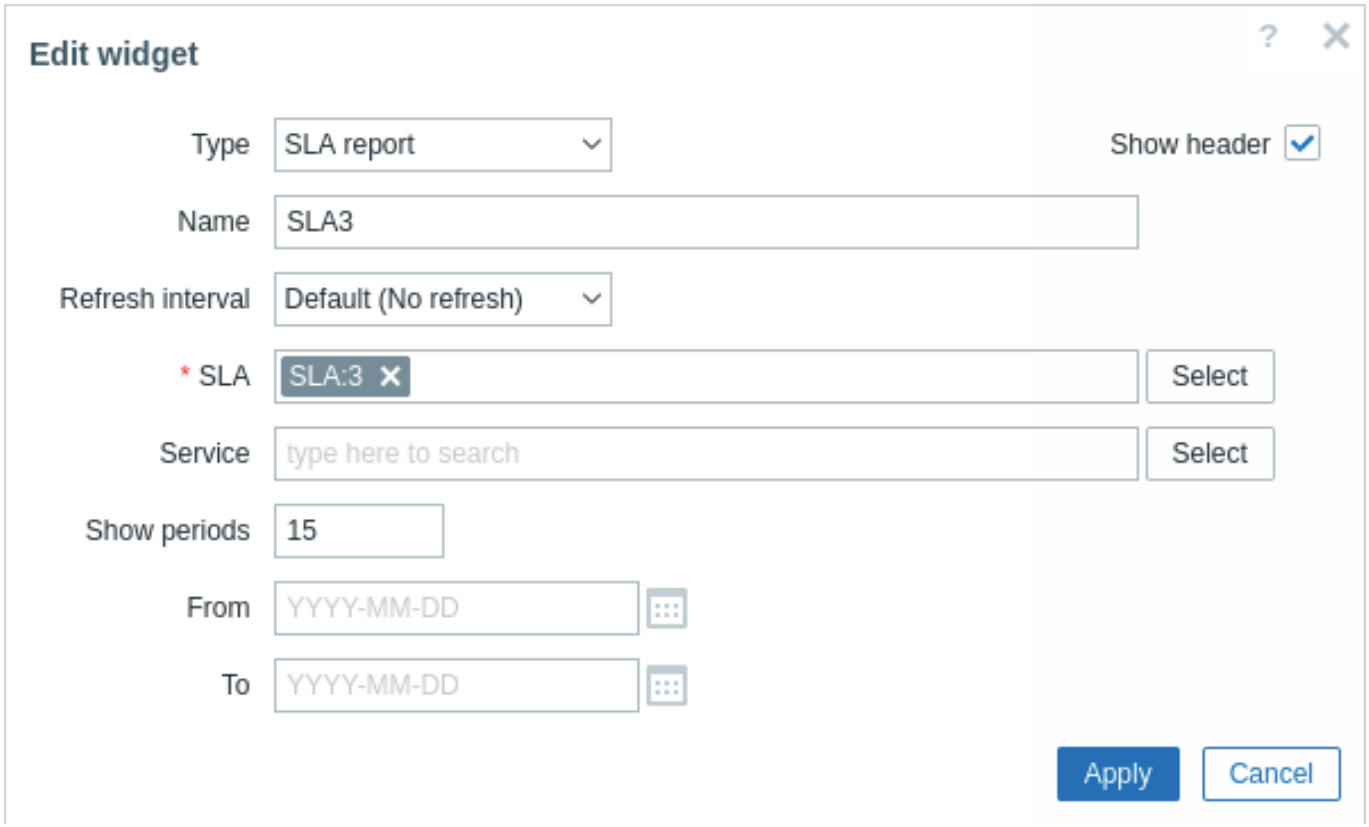
25 Raport SLA

Przegląd

Ten widżet jest przydatny do wyświetlania raportów SLA. Pod względem funkcjonalnym jest podobny do sekcji *Usługi -> Raport SLA*.

Konfiguracja

Aby skonfigurować, wybierz *SLA report* jako typ:



Oprócz parametrów, które są wspólne dla wszystkich widżetów, możesz ustawić następujące opcje specyficzne:

SLA	Wybierz SLA dla raportu.
Service	Wybierz usługę dla raportu.
Show periods	Ustaw, ile okresów będzie wyświetlanych w widżecie (domyślnie 20, maksymalnie 100).
From	Wybierz datę początkową raportu. Obsługiwane są daty względne: `now`, `now/d`, `now/w-1w` itd.; obsługiwane modyfikatory daty: `d`, `w`, `M`, `y`.

To	Wybierz datę końcową raportu. Obsługiwane są daty względne: `now`, `now/d`, `now/w-1w` itd.; obsługiwane modyfikatory daty: `d`, `w`, `M`, `y`.

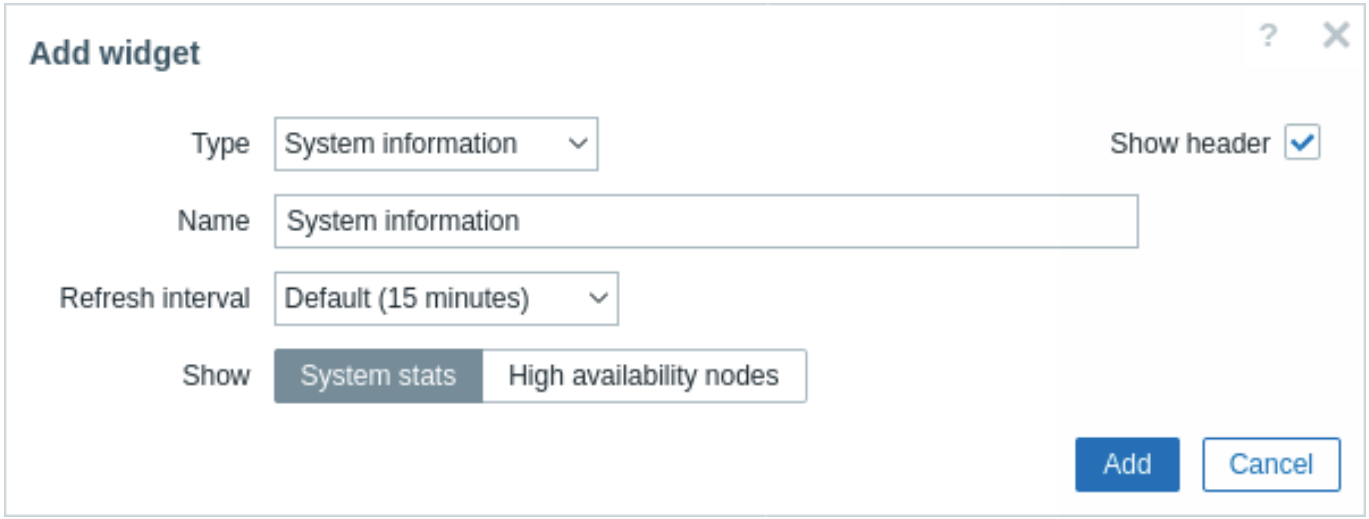
26 Informacje o systemie

Przegląd

Ten widżet wyświetla te same informacje co w *Reports* → *System information*, jednak pojedynczy widżet dashboardu może jednocześnie wyświetlać tylko statystyki systemu albo węzły wysokiej dostępności (nie oba naraz).

Konfiguracja

Aby skonfigurować, wybierz *Informacje o systemie* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

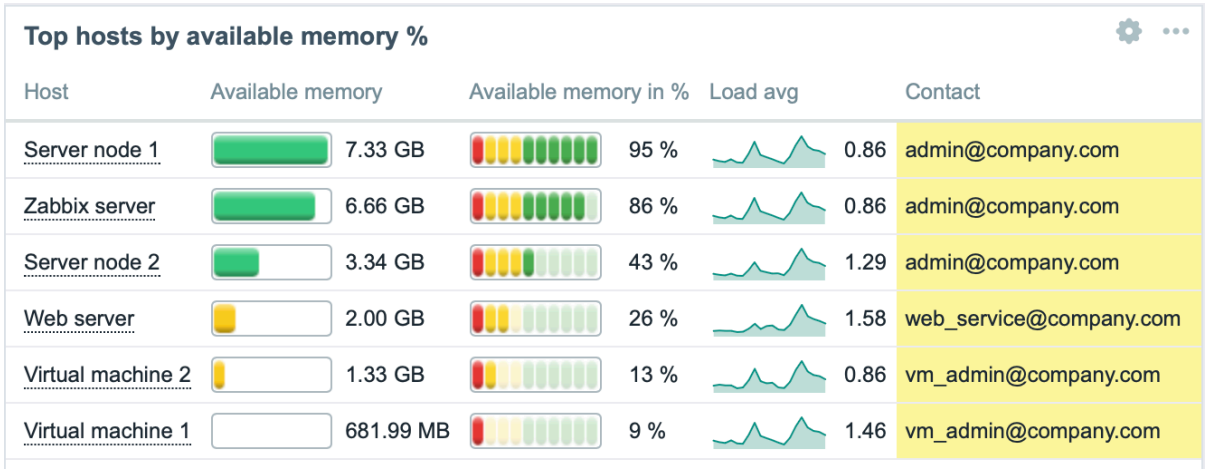
Pokaż	Wybierz, co wyświetlać: **Statystyki systemu** — wyświetla podsumowanie kluczowych danych serwera Zabbix i systemu; **Węzły wysokiej dostępności** — wyświetla status węzłów wysokiej dostępności (jeśli klaster wysokiej dostępności jest włączony).
Pokaż szczegóły sprawdzania aktualizacji oprogramowania	Zaznacz pole wyboru, aby wyświetlać szczegóły sprawdzania aktualizacji oprogramowania Zabbix. Ta opcja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix i w polu *Pokaż* wybrano opcję „Statystyki systemu”.

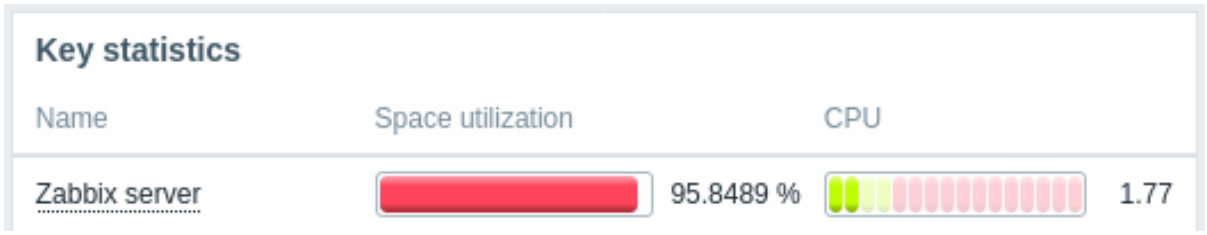
27 Najlepsze hosty

Omówienie

Widżet Top hosts umożliwia tworzenie niestandardowych tabel do wyświetlania danych dla wielu hostów i może być wykorzystywany do tworzenia raportów typu *Top N* oraz raportów z paskami postępu, przydatnych przy planowaniu pojemności. Typ wyświetlanych danych można dostosować i mogą one obejmować użycie CPU, dostępność pamięci lub dowolne inne zebrane wartości w formacie liczbowym, tekstowym lub binarnym, a także statyczne ciągi tekstowe. Dane mogą być prezentowane w różnych formatach, takich jak paski, wskaźniki lub wykresy sparkline.

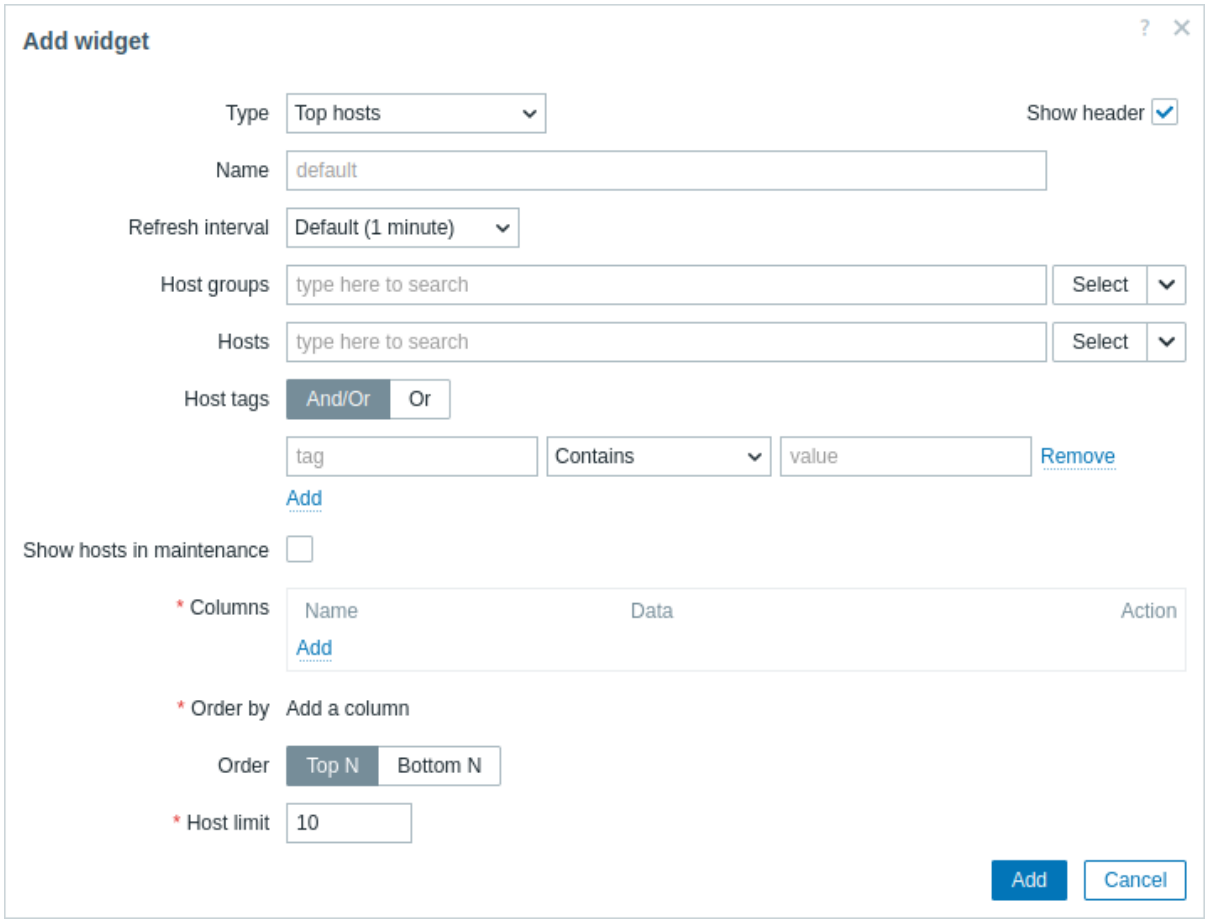
Maksymalna liczba hostów, które można wyświetlić, wynosi 1000.





Konfiguracja

Aby skonfigurować, wybierz *Top hosts* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów można ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

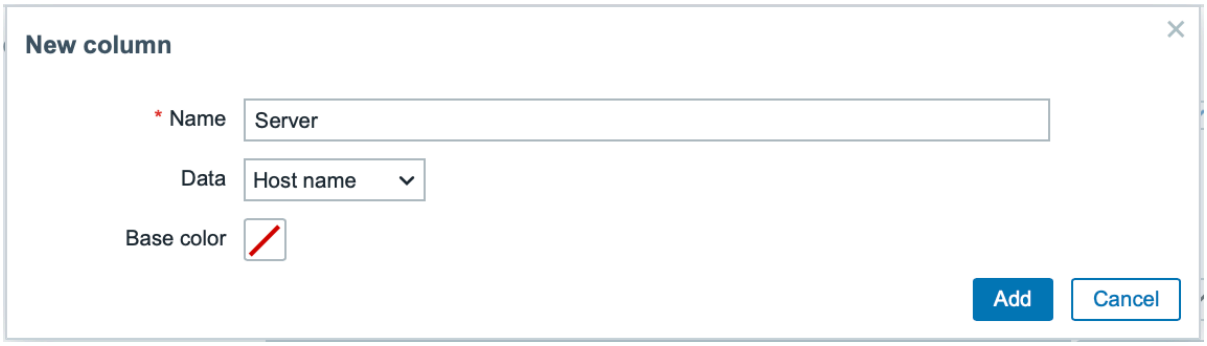
Hosts	Wybierz hosty, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Host tags	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Show hosts in maintenance	Zaznacz to pole wyboru, aby wyświetlać również hosty w utrzymaniu (w takim przypadku obok nazwy hosta zostanie pokazana ikona utrzymania). Domyślnie niezaznaczone.
Columns	Dodaj kolumny danych do wyświetlenia. Kolejność kolumn określa ich wyświetlanie od lewej do prawej. Kolumny można zmieniać miejscami, przeciągając je w górę i w dół za uchwyt znajdujący się przed nazwą kolumny.
Order by	Określ kolumnę ze zdefiniowanej listy *Columns*, która ma być używana do sortowania *Top N* lub *Bottom N*.
Order	Określ kolejność wierszy: **Top N** - malejąco zgodnie z zagregowaną wartością *Order by*; **Bottom N** - rosnąco zgodnie z zagregowaną wartością *Order by*.
Host limit	Liczba wierszy hostów do wyświetlenia (1-1000). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Konfiguracja kolumn

Można dodać kolumny trzech typów danych: nazwa hosta, wartość pozycji lub tekst. Lista dostępnych parametrów kolumny zależy od typu danych kolumny, a w przypadku typu wartości pozycji także od formatu wartości.

Kolumna Nazwa hosta

Kolumna Nazwa hosta służy do wyświetlania nazwy hosta.



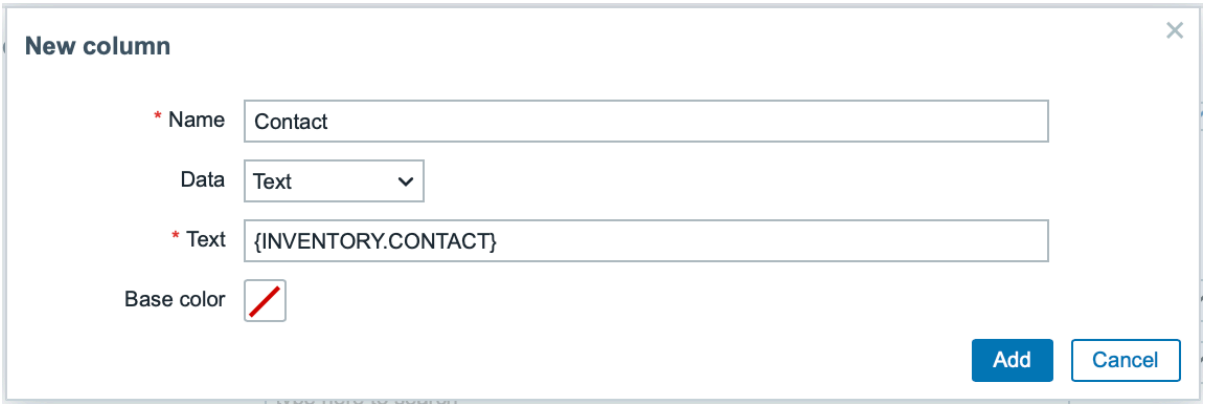
Obsługiwane parametry:

Name	Nazwa kolumny.
Data	Typ danych wyświetlanych w kolumnie; wybierz **Nazwa hosta**.

Base color	Kolor tła kolumny.

Kolumna tekstowa

Kolumna tekstowa służy do wyświetlania dowolnego określonego ciągu tekstowego.

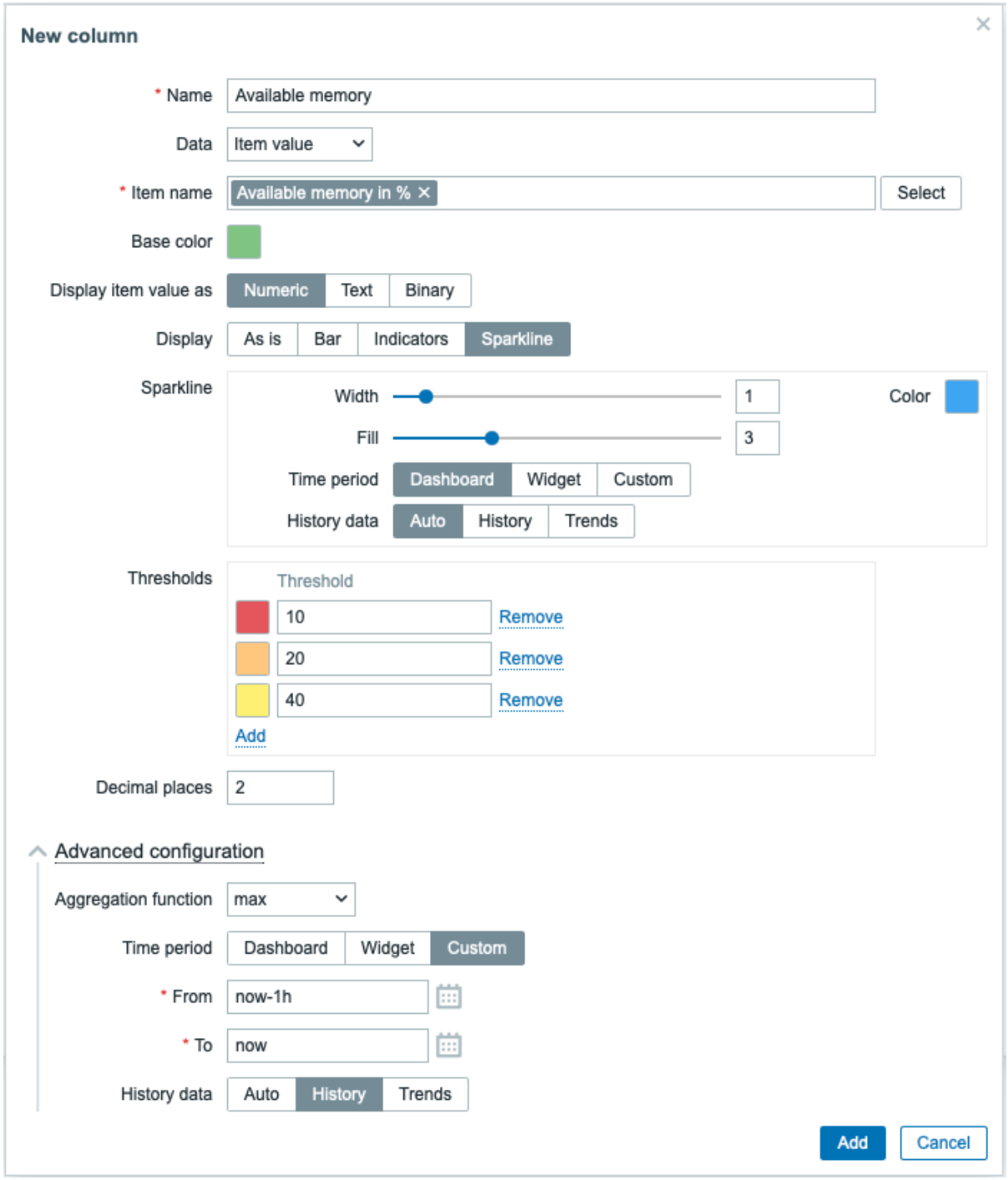


Obsługiwane parametry:

Name	Nazwa kolumny.
Data	Typ danych do wyświetlenia w kolumnie, wybierz **Text**.
Base color	Kolor tła kolumny.
Text	Wprowadź ciąg do wyświetlenia. Może zawierać host i inventory makra.

Kolumna wartości pozycji

Kolumna wartości pozycji służy do wyświetlania wartości określonej pozycji; obsługuje wiele formatów i opcji wyświetlania wartości.



Obsługiwane parametry:

Nazwa	Nazwa kolumny.
Dane	Typ danych do wyświetlenia w kolumnie, wybierz **Item value**.
Nazwa pozycji	Wybierz pozycję; jej nazwa jest używana do dopasowywania i wyświetlania pozycji o tej samej nazwie na wszystkich wybranych hostach. Jeśli host ma wiele takich pozycji, widżet wyświetla pozycję, której klucz jest pierwszy alfabetycznie (np. *CPU utilization* z kluczem `proc.cpu.util` zamiast `system.cpu.util`). Podczas konfigurowania widżetu na panelu szablonu można wybrać tylko pozycje skonfigurowane w szablonie.
Kolor bazowy	Kolor tła kolumny; kolor wypełnienia, jeśli dane *Item value* są wyświetlane jako pasek/wskaźniki. Domyślny kolor można zastąpić kolorem niestandardowym, jeśli wartość pozycji przekroczy jeden z określonych progów.
Wyświetl wartość pozycji jako	Format wyświetlania wartości pozycji: **Numeric**, **Text** lub **Binary**. Wybrana opcja określa, które dodatkowe parametry będą dostępne. Zapoznaj się z listą parametrów dla każdego formatu poniżej.
Konfiguracja zaawansowana	

Funkcja agregacji	Określ, której funkcji agregacji użyć w ramach wybranego *Okresu czasu*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **count** - wyświetl liczbę wartości; **sum** - wyświetl sumę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość. Jeśli wybrano **not used**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja pozwala wyświetlać wartość zagregowaną dla wybranego interwału (5 minut, godzina, dzień) zamiast najnowszej wartości. Tylko dane numeryczne mogą być wyświetlane dla *min*, *max*, *avg* i *sum*. Dla *count* dane nienumeryczne zostaną przekształcone na numeryczne.
Okres czasu	Określ okres czasu używany do agregowania wartości: **Dashboard** - użyj okresu czasu dashboardu; **Widget** - użyj okresu czasu określonego widżetu; **Custom** - użyj niestandardowego okresu czasu. Ten parametr nie będzie wyświetlany, jeśli dla *Funkcja agregacji* ustawiono wartość ”not used”.
Widget	Wybierz widżet. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość ”Widget”.
Od	Wybierz okres czasu od (wartość domyślna `now-1h`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość ”Custom”.
Do	Wybierz okres czasu do (wartość domyślna `now`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość ”Custom”.
Dane historyczne	Pobierz dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobierz dane historyczne; **Trends** - pobierz dane trendów. Ten parametr jest dostępny tylko dla numerycznych wartości pozycji. Dla wartości tekstowych i binarnych dane będą zawsze pobierane z historii.

Parametry numerycznej wartości pozycji:

Wyświetlanie	Określ, jak wartość ma być wyświetlana: **As is** - jako zwykły tekst; **Bar** - jako jednolity, wypełniony kolorem pasek; **Indicators** - jako segmentowany, wypełniony kolorem pasek; **Sparkline** - miniwykres liniowy.
Progi	Określ wartości progowe, po których kolor tła/wypełnienia powinien się zmieniać. Lista zostanie po zapisaniu posortowana rosnąco. W przypadku sparkline progi są stosowane tylko do ostatniej wartości pozycji.
Liczba miejsc po przecinku	Określ, ile miejsc po przecinku ma być wyświetlanych przy wartości.
Bar/Indicators	
Min	Wartość minimalna.
Max	Wartość maksymalna.
Sparkline	
Szerokość	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie wpisując wartość z zakresu od 0 do 10.
Kolor	Wybierz kolor linii i wypełnienia.
Wypełnienie	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie wpisując wartość z zakresu od 0 do 10.
Okres czasu	Określ okres czasu, z którego wartości mają zostać uwzględnione na wykresie sparkline: **Dashboard** - użyj okresu czasu dashboardu; **Widget** - użyj okresu czasu określonego widżetu; **Custom** - użyj niestandardowego okresu czasu. Należy pamiętać, że okres czasu wpływa tylko na wykres sparkline. Wartość wyświetlana obok sparkline reprezentuje ostatnią wartość pozycji, a nie ostatnią wartość z wybranego okresu czasu.

Widget	Wybierz widżet. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość "Widget".
Od	Wybierz okres czasu od (wartość domyślna `now-1h`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość "Custom".
Do	Wybierz okres czasu do (wartość domyślna `now`). Zobacz składnię czasu względnego. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Okres czasu* ustawiono wartość "Custom".
Dane historyczne	Pobierz dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobierz dane historyczne; **Trends** - pobierz dane trendów.

Parametry tekstowej wartości pozycji:

Wyróżnienia	Określ wyrażenia regularne, po dopasowaniu których kolor tła/wypełnienia powinien się zmienić.

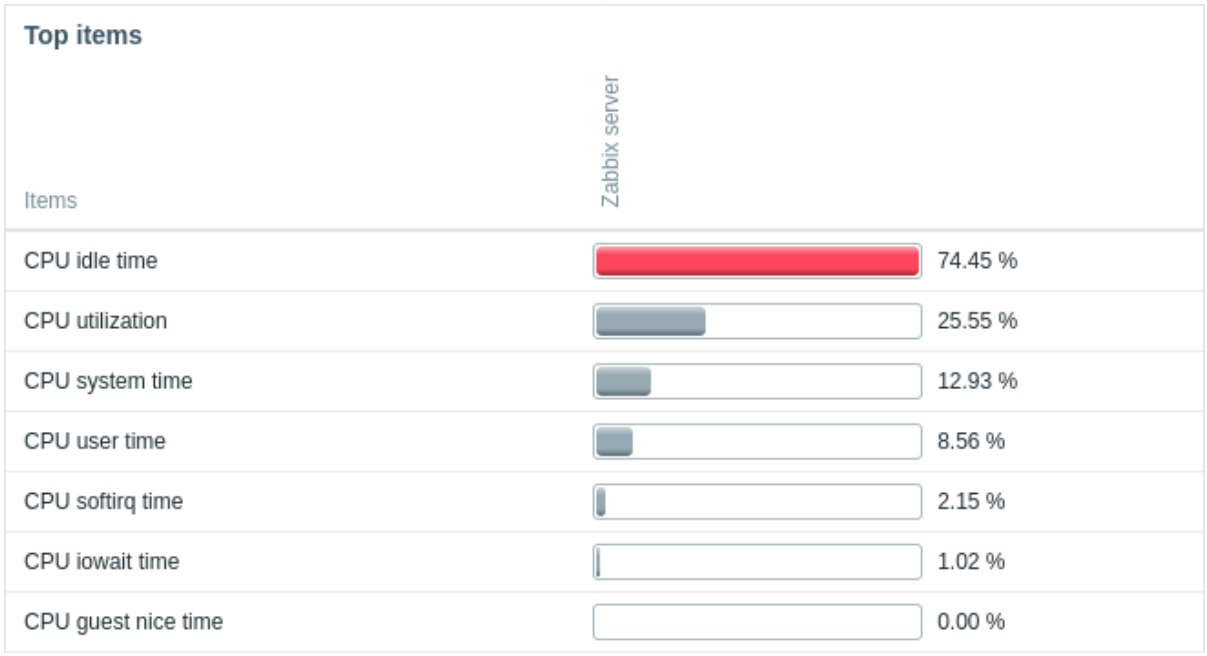
Parametry binarnej wartości pozycji:

Pokaż miniaturę	Określ, czy utworzyć i wyświetlić miniaturę obrazu zawierającego dane binarne, czy wyświetlić hiperłącze *Show* prowadzące do obrazu w pełnym rozmiarze w kolumnie wartości.

28 Najważniejsze pozycje

Omówienie

W widżecie *Top items* można wyświetlać najwyższe (lub najniższe) wartości dla grupy pozycji.



Można wybierać pozycje bezpośrednio lub według wzorca, a także dostosować kolejność kolumn i wyróżnianie.

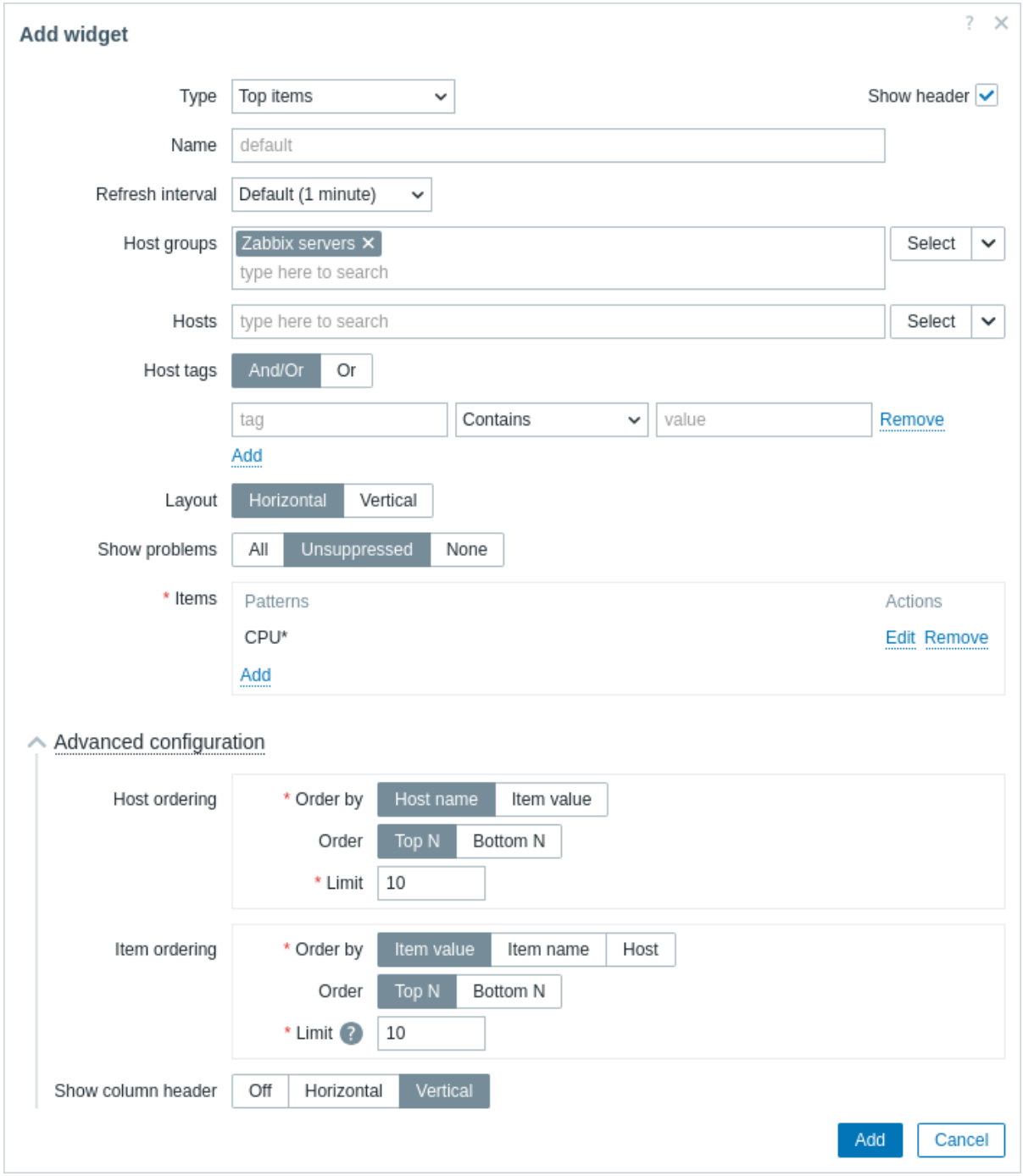
Kolor pozycji z problemem jest oparty na kolorze ważności problemu, który można dostosować na ekranie aktualizacji problemu.

Domyślnie wyświetlane są tylko wartości mieszczące się w ostatnich 24 godzinach. To ograniczenie wprowadzono w celu poprawy czasu początkowego ładowania dużych stron z najnowszymi danymi. Jest ono konfigurowalne w *Administracja* → *Ogólne* → *GUI*, przy użyciu opcji *Maksymalny okres wyświetlania historii*.

Kliknięcie wartości pozycji otwiera menu pozycji.

Konfiguracja

Aby skonfigurować, wybierz *Top items* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

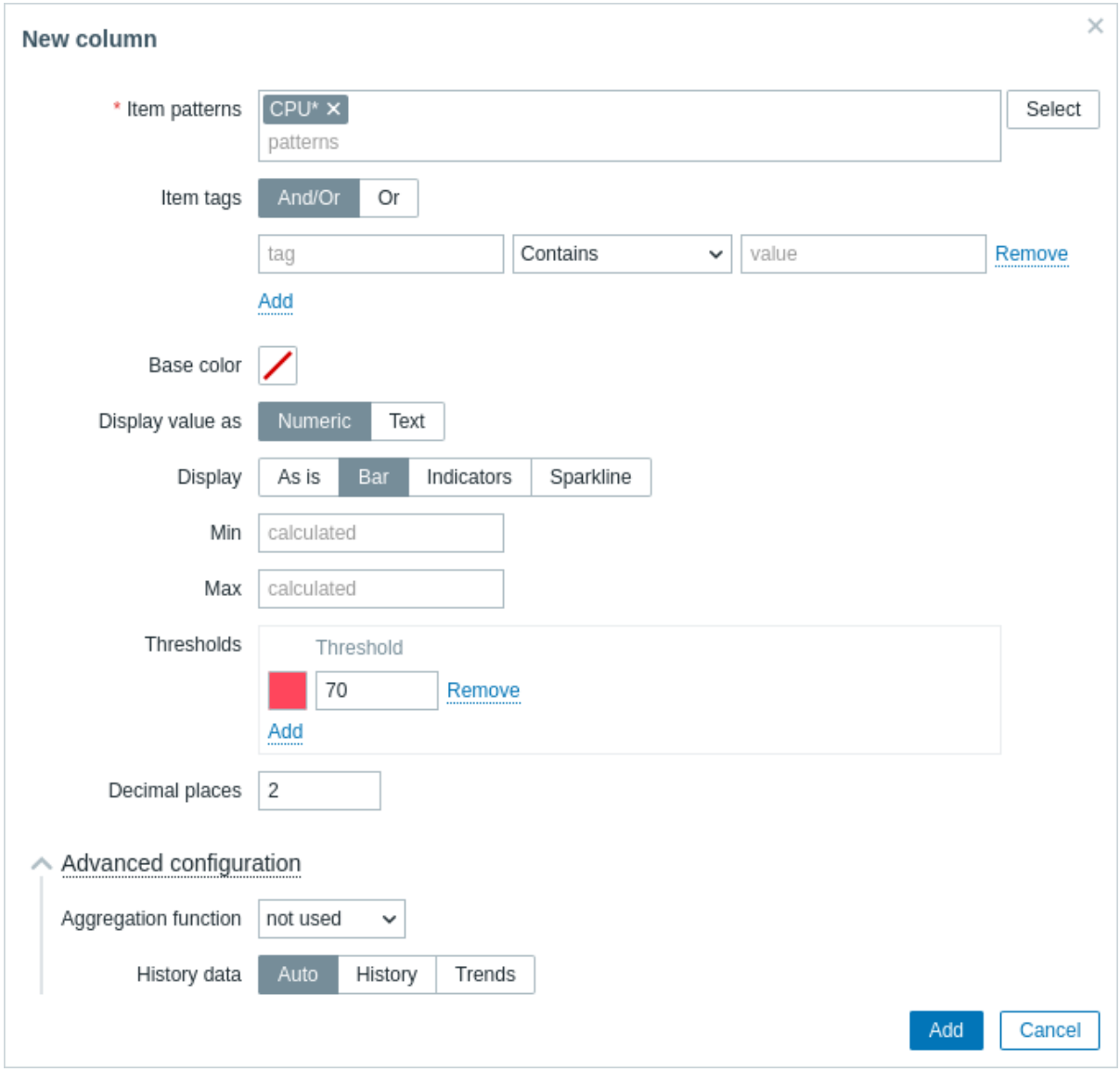
Host groups	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Hosts	Wybierz hosty. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.

Host tags		Określ tagi, aby ograniczyć liczbę danych hostów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg (dopasowanie fragmentu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg (dopasowanie fragmentu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek.
Layout		Wybierz opcję układu: **Horizontal** - nazwy hostów będą wyświetlane po lewej; **Vertical** - nazwy hostów będą wyświetlane u góry.
Show problems		Wybierz sposób wyświetlania problemów: **All** - wyświetlaj wszystkie problemy; **Unsuppressed** - wyświetlaj tylko problemy, które nie są wyciszone z powodu konserwacji hosta; **None** - nie wyświetlaj problemów w ogóle. *Uwaga* gdy reprezentowany jest stan problemu, zastępuje on ustawienia stylu wyświetlania z konfiguracji kolumny pozycja.
Items		Dodaj wzorce pozycji lub konkretne pozycje do wyświetlenia (zobacz Konfiguracja kolumny).
Advanced configuration		
Host ordering		Wybierz opcje sortowania dla kolumny/wiersza hosta.
	Order by	Sortuj hosty według: **Host name** - hosty będą sortowane według nazwy hosta; **Item value** - hosty będą sortowane według wartości wybranych pozycji.
	Order	Wybierz, czy wyświetlać najwyższe, czy najniższe wartości: **TopN** - N najwyższych wartości; **BottomN** - N najniższych wartości. Wartość N jest wybierana w polu *Limit*.
	Limit	Wprowadź limit wyświetlanych hostów (zakres 1-1000; domyślnie 10). Ta wartość stanie się wartością N w polu *Order*.
Item ordering		Wybierz opcje sortowania dla kolumny/wiersza pozycji.
	Order by	Sortuj pozycje według: **Item value** - pozycje będą sortowane według wartości pozycji; **Item name** - pozycje będą sortowane według nazwy pozycji; **Host** - pozycje będą sortowane według wybranego wzorca hosta.
	Order	Wybierz, czy wyświetlać najwyższe, czy najniższe wartości: **TopN** - N najwyższych wartości; **BottomN** - N najniższych wartości. Wartość N jest wybierana w polu *Limit*.

Limit	Wprowadź limit wyświetlanych pozycji (zakres 1-1000; domyślnie 10). Ta wartość stanie się wartością N w polu *Order*.
Show column header	Wybierz opcje wyświetlania nagłówka kolumny: **Off** - nie wyświetlaj nagłówka kolumny; **Horizontal** - wyświetlaj tekst poziomo w nagłówku; **Vertical** - wyświetlaj tekst pionowo w nagłówku.

Konfiguracja kolumn

Aby skonfigurować kolumny pozycji, kliknij *Dodaj* w parametrze *Pozycje*:



Wspólne parametry kolumn:

Wzorce pozycji	Określ jeden lub kilka wzorców pozycji, używając znaku wieloznacznego. Alternatywnie wybierz pozycje. Podczas konfigurowania widżetu na pulpicie szablonu można wybrać tylko pozycje skonfigurowane w szablonie.

Tagi pozycji	Określ tagi, aby ograniczyć liczbę danych pozycji wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze rozróżnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, w których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, w których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Kolor bazowy	Wybierz kolor tła kolumny lub kolor wypełnienia, jeśli dla parametru *Wyświetlanie* ustawiono wartość „Słupek” lub „Wskaźniki”. Zwróć uwagę, że kolor bazowy może zostać zastąpiony przez kolory progów lub wyróżnienia.
Wyświetlaj wartość jako	Format wyświetlania wartości pozycji: **Liczbowy** lub **Tekst**. Wybrana opcja określa, które dodatkowe parametry będą dostępne. Szczegóły znajdują się na poniższej liście parametrów dla każdego formatu.

Parametry kolumn dla wartości numerycznych:

Display	Określ, w jaki sposób wartość ma być wyświetlana: **As is** - jako zwykły tekst; **Bar** - jako jednolity pasek wypełniony kolorem; **Indicators** - jako segmentowany pasek wypełniony kolorem; **Sparkline** - miniwykres liniowy.
Min	Minimalna wartość dla wyświetlania w formie paska/wskaźników.
Max	Maksymalna wartość dla wyświetlania w formie paska/wskaźników.
Width	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie wpisując wartość z zakresu od 0 do 10. Ten parametr dotyczy tylko wyświetlania sparkline.
Fill	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie wpisując wartość z zakresu od 0 do 10. Ten parametr dotyczy tylko wyświetlania sparkline.
Color	Wybierz kolor linii i wypełnienia. Ten parametr dotyczy tylko wyświetlania sparkline.
Zobacz *Advanced configuration* w celu uzyskania opisu pól związanych z okresem czasu i wyborem danych historycznych.	
Thresholds	Określ wartości progowe, przy których kolor tła/wypełnienia ma się zmieniać. Lista zostanie po zapisaniu posortowana rosnąco. W przypadku sparkline progi są stosowane tylko do ostatniej wartości pozycja.
Decimal places ***Advanced configuration***	Określ liczbę miejsc po przecinku wyświetlanych przy wartości.

Aggregation function	Określ, której funkcji agregacji użyć w wybranym *Time period*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **count** - wyświetl liczbę wartości; **sum** - wyświetl sumę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość. Jeśli wybrano **not used**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja pozwala wyświetlać wartość zagregowaną dla wybranego interwału (5 minut, godzina, dzień) zamiast najnowszej wartości. Tylko dane numeryczne mogą być wyświetlane dla *min*, *max*, *avg* i *sum*. Dla *count* dane nienumeryczne zostaną przekształcone na numeryczne.
Time period	Określ okres czasu używany do agregowania wartości: **Dashboard** - użyj okresu czasu dashboardu; **Widget** - użyj okresu czasu określonego widgetu; **Custom** - użyj niestandardowego okresu czasu. Ten parametr nie będzie wyświetlany, jeśli dla *Aggregation function* ustawiono ”not used”.
Widget	Wybierz widget. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Time period* ustawiono wartość ”Widget”.
From	Wybierz początek okresu czasu (wartość domyślna `now-1h`). Zobacz relative time syntax. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Time period* ustawiono wartość ”Custom”.
To	Wybierz koniec okresu czasu (wartość domyślna `now`). Zobacz relative time syntax. Ten parametr będzie wyświetlany tylko wtedy, gdy dla *Time period* ustawiono wartość ”Custom”.
History data	Pobieraj dane z historii lub trendów: **Auto** - automatyczny wybór; **History** - pobieraj dane historyczne; **Trends** - pobieraj dane trendów. Ten parametr jest dostępny tylko dla numerycznych wartości pozycja. Dla wartości tekstowych dane będą zawsze pobierane z historii.

Parametry kolumn dla wartości tekstowych:

Highlights	Określ wyrażenia regularne, po dopasowaniu których kolor tła/wypełnienia ma się zmieniać.

29 Najważniejsze wyzwalacze

Omówienie

W widżecie *Top triggers* można zobaczyć wyzwalacze z największą liczbą problemów.

Top triggers

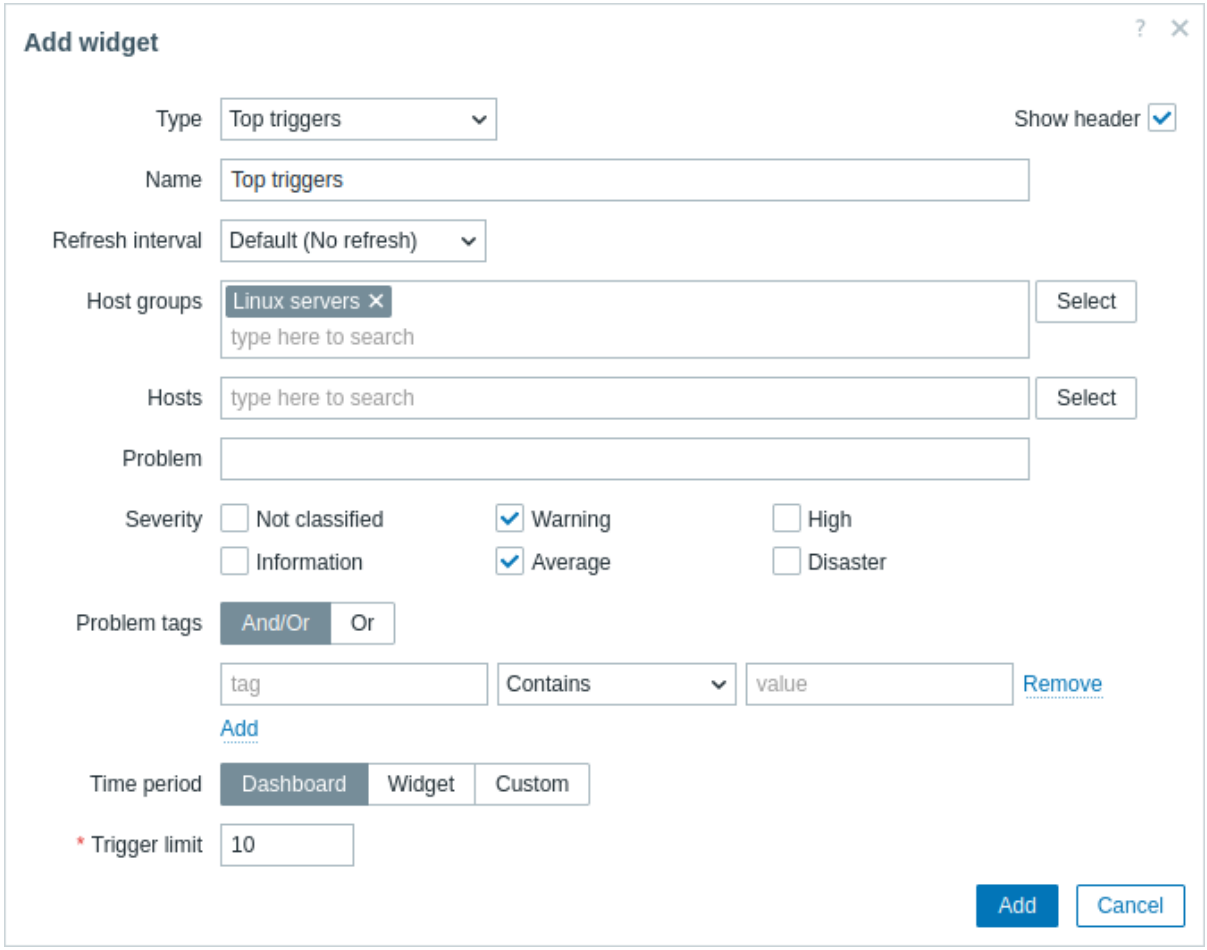
Host	Trigger	Severity	Number of problems
Zabbix server	Interface enp0s3: Link down	Average	2
Zabbix server	Load average is too high	Average	2
Zabbix server	Zabbix agent is not available	Average	2
Zabbix server	Zabbix server: More than 100 items having missing data for more than 10 minutes	Warning	2
Zabbix server	Zabbix server: Utilization of escalator processes is high	Average	2

Maksymalna liczba wyzwalaczy, które mogą być wyświetlone, wynosi 1000. Podczas przeglądania widżetu na pulpicie nawigacyjnym można wybrać okres czasu, dla którego mają być wyświetlane dane.

Informacje o najważniejszych wyzwalaczach są również dostępne w sekcji menu *Reports* → *Top 100 triggers*.

Konfiguracja

Aby skonfigurować, wybierz *Top triggers* jako typ widżetu:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, dla których mają być wyświetlane problemy w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona lista rozwijana pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Problemy z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wpisane żadne grupy hostów, będą wyświetlane problemy ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosts	Wybierz hosty, dla których mają być wyświetlane problemy w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana pasujących hostów. Jeśli nie zostaną wpisane żadne hosty, będą wyświetlane problemy ze wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Problem	Możesz wyświetlać wyłącznie wyzwalacze dla określonych problemów. W tym celu wpisz ciąg znaków, który ma zostać dopasowany do nazwy problemu. Makra nie są rozwijane.
Severity	Zaznacz poziomy ważności wyzwalaczy, aby filtrować wyzwalacze wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, będą wyświetlane wszystkie wyzwalacze.

Problem tags	Określ tagi problemów, które mają być wyświetlane w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek.
Time period	Wybierz źródło danych dla okresu czasu: **Dashboard** - ustaw selektor *Time period* jako źródło danych; **Widget** - ustaw zgodny widżet określony w parametrze *Widget* jako źródło danych; **Custom** - ustaw okres czasu określony w parametrach *From* i *To* jako źródło danych; jeśli zostanie ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu myszą.
Widget	Wpisz lub wybierz zgodny widżet (*Graph*, *Graph (classic)*, *Graph prototype*) jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* ma wartość "Widget".
From	Wpisz lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".
To	Wpisz lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w`, itd.). Ten parametr jest dostępny, jeśli *Time period* ma wartość "Custom".
Trigger limit	Ustaw liczbę wyzwalaczy do wyświetlenia. Zakres możliwych wartości: 1-1000.

30 Przegląd wyzwalaczy

Przegląd

W widżecie przeglądu wyzwalaczy można wyświetlać stany wyzwalaczy dla grupy hostów.

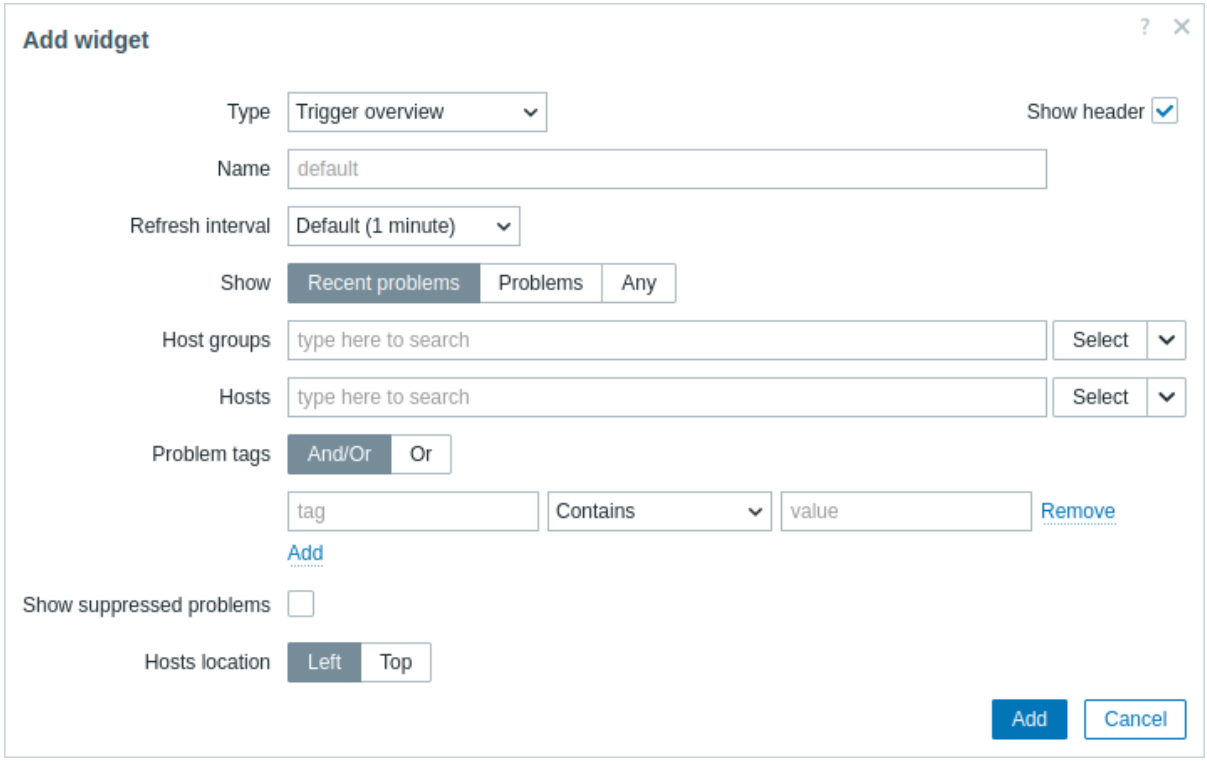
- Stany wyzwalaczy są wyświetlane jako kolorowe bloki (kolor bloków dla wyzwalaczy PROBLEM zależy od koloru ważności problemu, który można dostosować na ekranie aktualizacji problemu). Należy pamiętać, że ostatnie zmiany stanu wyzwalacza (w ciągu ostatnich 2 minut) będą wyświetlane jako migające bloki.
- Szare strzałki w górę i w dół wskazują wyzwalacze, które mają zależności. Po najechaniu myszą są wyświetlane szczegóły zależności.
- Ikona pola wyboru wskazuje potwierdzone problemy. Aby ta ikona była wyświetlana, wszystkie problemy lub rozwiązane problemy danego wyzwalacza muszą zostać potwierdzone.

Kliknięcie bloku wyzwalacza udostępnia zależne od kontekstu łącza do zdarzeń problemów wyzwalacza, ekranu potwierdzania problemu, konfiguracji wyzwalacza, adresu URL wyzwalacza lub prostej listy wykresów/ostatnich wartości.

Należy pamiętać, że domyślnie wyświetlanych jest 50 rekordów (można to skonfigurować w *Administracja* → *Ogólne* → *GUI*, za pomocą opcji *Maksymalna liczba kolumn i wierszy w tabelach przeglądu*). Jeśli istnieje więcej rekordów niż skonfigurowano do wyświetlenia, na dole tabeli zostanie wyświetlony komunikat z prośbą o podanie bardziej szczegółowych kryteriów filtrowania. Nie ma stronicowania. Należy pamiętać, że to ograniczenie jest stosowane najpierw, przed jakimkolwiek dalszym filtrowaniem danych, na przykład według tagów.

Konfiguracja

Aby skonfigurować widget, wybierz *Przegląd wyzwalaczy* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Pokaż	Filtruj wyzwalacze według stanu wyzwalacza: **Ostatnie problemy** - *(domyślnie)* pokazuj wyzwalacze, które niedawno były lub nadal są w stanie PROBLEM (rozwiązane i nierozwiązane); **Problemy** - pokazuj wyzwalacze, które są w stanie PROBLEM (nierozwiązane); **Dowolne** - pokazuj wszystkie wyzwalacze.
Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Hosty	Wybierz hosty. Alternatywnie wybierz zgodny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.

Tagi problemów	Określ tagi do filtrowania wyzwalaczy wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. **Uwaga:** Jeśli parametr *Pokaż* jest ustawiony na „Dowolne”, zostaną wyświetlone wszystkie wyzwalacze, nawet jeśli określono tagi. Jednak chociaż ostatnie zmiany stanu wyzwalacza (wyświetlane jako migające bloki) będą aktualizowane dla wszystkich wyzwalaczy, szczegóły stanu wyzwalacza (kolor ważności problemu oraz informacja, czy problem został potwierdzony) będą aktualizowane tylko dla wyzwalaczy pasujących do określonych tagów. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Pokaż ukryte problemy	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niepokazywane) z powodu konserwacji hosta.
Układ	Wybierz opcję układu: **Poziomy** - nazwy hostów będą wyświetlane po lewej stronie; **Pionowy** - nazwy hostów będą wyświetlane u góry.

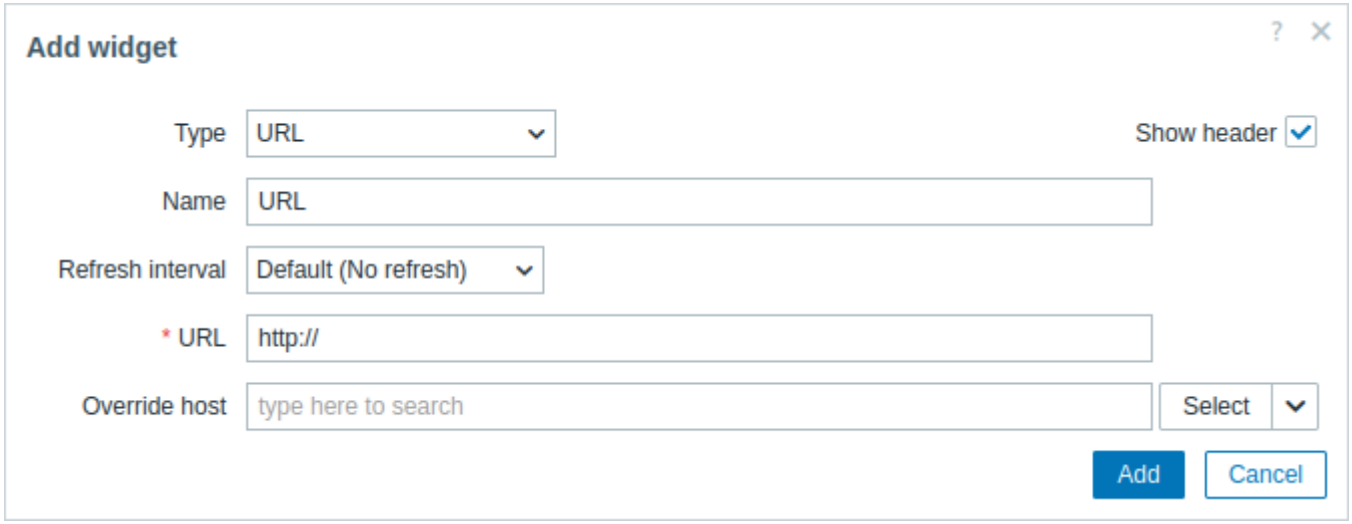
31 URL

Przegląd

Ten widżet wyświetla zawartość pobraną z określonego adresu URL.

Konfiguracja

Aby skonfigurować, wybierz *URL* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

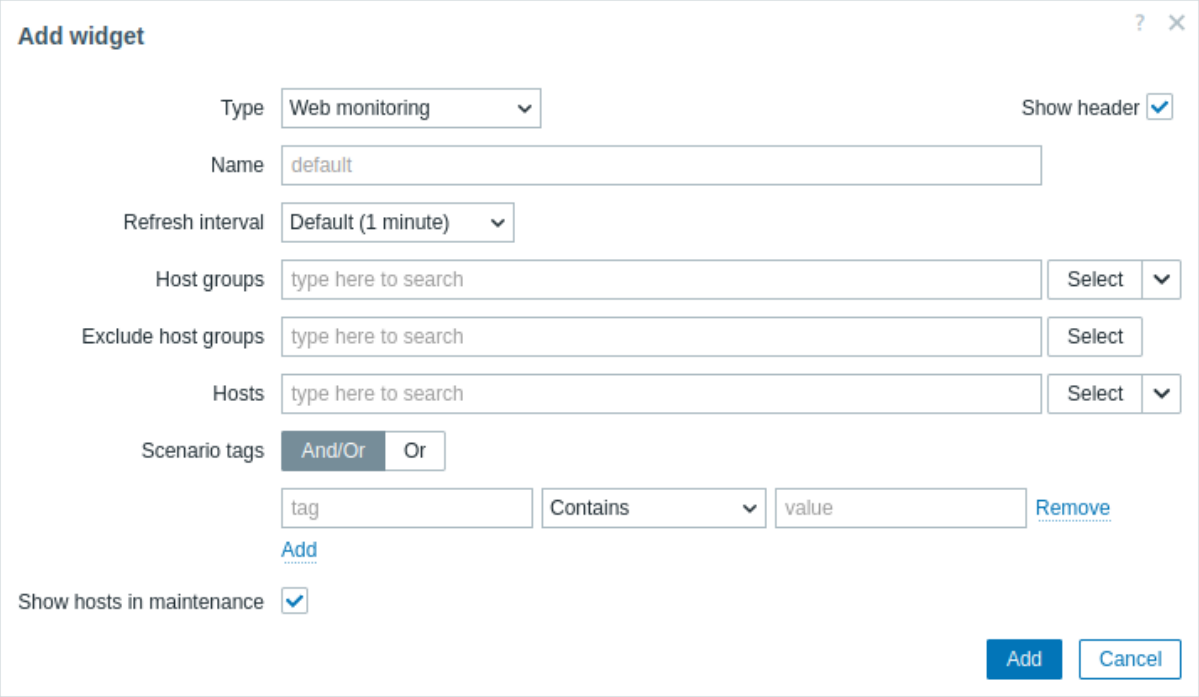
URL	Wprowadź URL do wyświetlenia (do 2048 znaków). Zewnętrzne URL-e muszą zaczynać się od `http://` lub `https://`. Wewnętrzne URL-e obsługują ścieżki względne (na przykład `zabbix.php?action=report.status`). Obsługiwane są makra {HOST.*}.
Override host	Wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.

Attention:
Przeglądarki mogą nie wczytać strony HTTP skonfigurowanej w widżecie, jeśli frontend Zabbixa jest dostępny przez HTTPS.

Przegląd

Ten widżet wyświetla podsumowanie stanu aktywnych scenariuszy monitorowania WWW. Szczegółowe informacje można znaleźć w sekcji Widżet monitorowania WWW.

Konfiguracja



Note:
W przypadkach, gdy użytkownik nie ma uprawnień do dostępu do niektórych elementów widżetu, nazwa tego elementu będzie wyświetlana jako *Inaccessible* podczas konfiguracji widżetu. W rezultacie zamiast „rzeczywistej" nazwy elementu pojawią się *Inaccessible Item, Inaccessible Host, Inaccessible Group, Inaccessible Map* oraz *Inaccessible Graph*.

Oprócz parametrów wspólnych dla wszystkich widżetów można ustawić następujące opcje specyficzne:

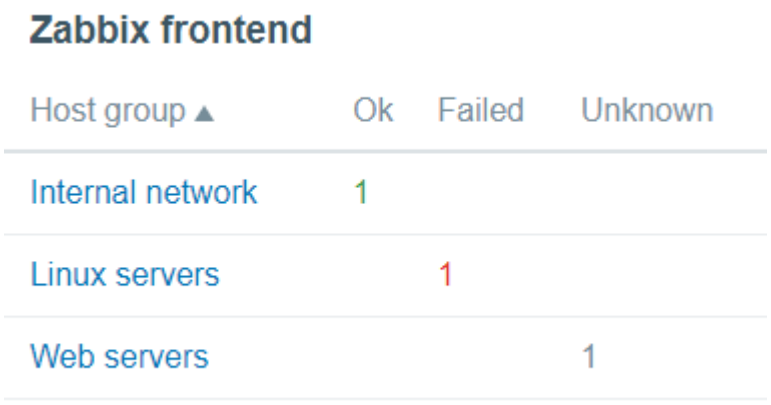
Host groups	Wybierz grupy hostów do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną podane żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.

Exclude host groups	Wybierz grupy hostów, które mają być ukryte w widżecie. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Dane hostów z tych grup hostów nie będą wyświetlane. Na przykład hosty 001, 002, 003 mogą znajdować się w Grupie A, a hosty 002, 003 również w Grupie B. Jeśli jednocześnie wybierzemy *show* Grupę A i *exclude* Grupę B, na panelu zostaną wyświetlone tylko dane z hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.
Hosts	Wybierz hosty do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet lub panel jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli nie zostaną podane żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widżetu na panelu szablonu.
Scenario tags	Określ tagi, aby ograniczyć liczbę scenariuszy web wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane według warunku *Or*; **Or** - wystarczy spełnienie jednego warunku.
Show hosts in maintenance	Uwzględnij hosty znajdujące się w trybie konserwacji w statystykach. Ten parametr jest oznaczony jako *Show data in maintenance* podczas konfigurowania widżetu na panelu szablonu.

Widget monitorowania WWW

Po zakończeniu konfiguracji możesz chcieć zobaczyć widget wraz z danymi, które wyświetla. Aby to zrobić, przejdź do *Dashboards* i kliknij nazwę dashboardu, na którym utworzono widget.

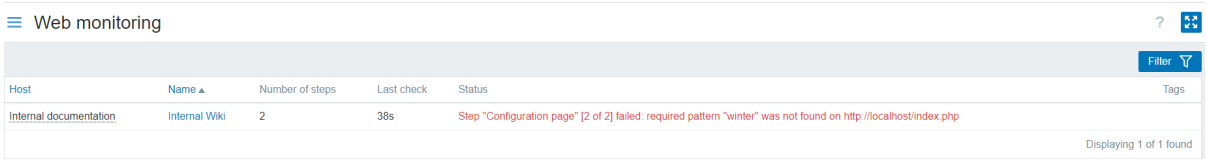
W tym przykładzie widać widget o nazwie ”Zabbix frontend”, który wyświetla stan monitorowania WWW dla trzech grup hostów: ”Internal network”, ”Linux servers” i ”Web servers.”



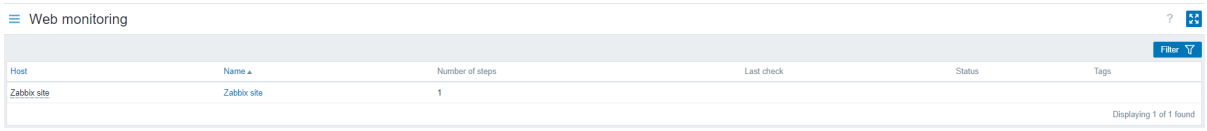
Widget monitorowania WWW wyświetla następujące informacje:

- nazwę widgetu; poniżej znajdują się cztery kolumny:
  - *Host group* - wyświetla listę grup hostów, które zawierają hosty z skonfigurowanymi scenariuszami WWW;

- *Ok* - wyświetla liczbę scenariuszy WWW (na zielono), gdy spełnione są dwa warunki:
  - Zabbix zebrał najnowsze dane dla scenariusza/scenariuszy WWW;
  - wszystkie kroki skonfigurowane w scenariuszu WWW mają status "Ok" *Status*.
- *Failed* - wyświetla liczbę scenariuszy WWW (na czerwono), w których wystąpiły nieudane kroki:
  - kliknij nazwę hosta, a otworzy się nowe okno; kolumna *Status* zawiera szczegółowe informacje (na czerwono) o kroku, na którym Zabbix nie zdołał zebrać danych; ponadto
  - podaje wskazówkę dotyczącą parametru, który należy poprawić w formularzu konfiguracji.



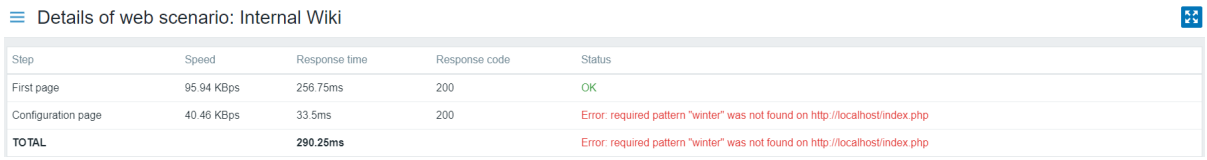
- *Unknown* - wyświetla liczbę scenariuszy WWW (na szaro), dla których Zabbix nie zebrał danych ani nie ma informacji o nieudanych krokach.



Wyświetlanie statusu i danych

Klikalne łącza w widżecie umożliwiają łatwą nawigację i szybkie uzyskanie pełnych informacji o każdym scenariuszu WWW. Aby wyświetlić:

- Status scenariusza WWW, kliknij nazwę grupy hostów.
- bardziej szczegółowe statystyki, kliknij nazwę scenariusza. W tym przykładzie jest to "Zabbix frontend".
- szczegóły w przypadku statusu *Failed*, kliknij nazwę grupy hostów; w otwartym oknie kliknij nazwę scenariusza WWW w kolumnie *Name*; zostaną wyświetlone bardziej szczegółowe informacje o skonfigurowanych krokach, dla których Zabbix nie zdołał zebrać danych.



Teraz możesz wrócić do formularza konfiguracji scenariusza WWW i poprawić ustawienia.

Aby wyświetlić szczegóły w przypadku statusu *Unknown*, możesz powtórzyć te same kroki, jak opisano dla *Failed*.

Attention:
Przy pierwszym wystąpieniu monitorowania scenariusz WWW jest zawsze wyświetlany w stanie *Unknown*, który zaraz po pierwszym sprawdzeniu zmienia się na stan *Failed* lub *Ok*. W przypadku, gdy host jest monitorowany przez proxy, zmiana statusu następuje zgodnie z częstotliwością zbierania danych skonfigurowaną na proxy.

2 Monitorowanie

Przegląd

Menu Monitoring służy do wyświetlania danych. Niezależnie od tego, jakie informacje Zabbix jest skonfigurowany do zbierania, wizualizacji i reagowania na nie, będą one wyświetlane w różnych sekcjach menu Monitoring.

Przyciski trybu widoku

Następujące przyciski znajdujące się w prawym górnym rogu są wspólne dla każdej sekcji:

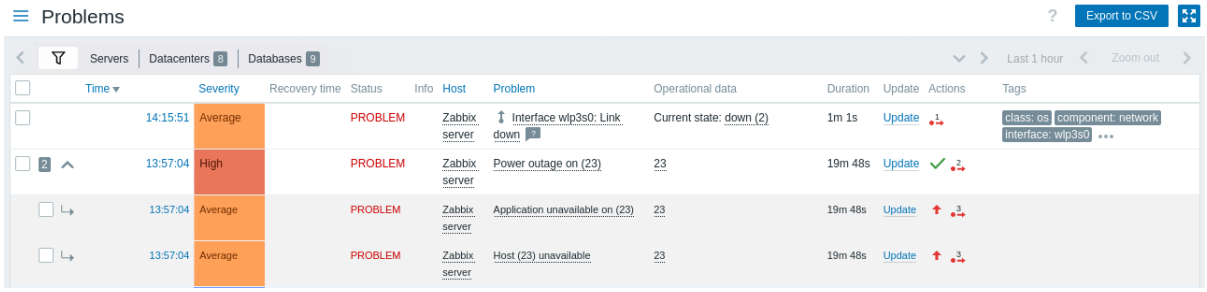
	Wyświetl stronę w trybie kiosku. W tym trybie wyświetlana jest tylko zawartość strony. Aby opuścić tryb kiosku, przesuń kursor myszy, aż pojawi się przycisk wyjścia  , a następnie kliknij go. Zostaniesz przeniesiony z powrotem do trybu normalnego.

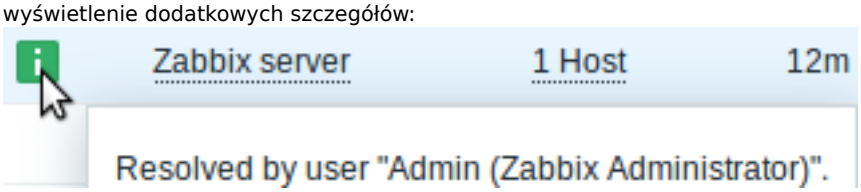
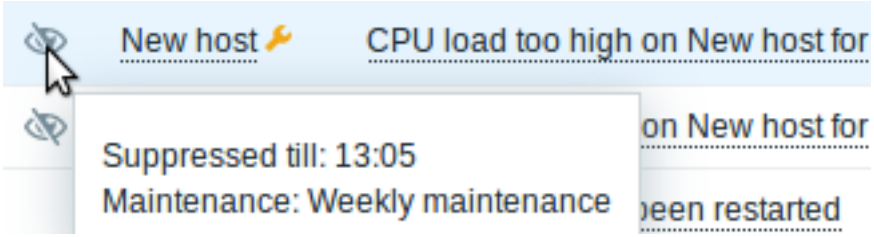
1 Problemy

Przegląd

W *Monitoring* > *Problems* możesz zobaczyć, jakie problemy obecnie występują. Problemami są te wyzwalacze, które znajdują się w stanie „Problem”.

Domyślnie wszystkie nowe problemy są klasyfikowane jako problemy przyczynowe. Możliwe jest ręczne przeklasyfikowanie wybranych problemów na problem objawowy problemu przyczynowego. Więcej informacji znajduje się w sekcji zdarzenia przyczynowe i objawowe.



Kolumna	Opis
Pole wyboru	Wyświetlane są pola wyboru do zaznaczania problemów. Ikony obok pól wyboru mają następujące znaczenie:  - liczba zdarzeń objawowych dla problemu przyczynowego;  - rozwiń, aby wyświetlić zdarzenia objawowe;  - zwiń, aby ukryć zdarzenia objawowe;  - to jest zdarzenie objawowe.
Czas	Wyświetlany jest czas rozpoczęcia problemu.
Ważność	Wyświetlana jest ważność problemu. Ważność problemu jest pierwotnie oparta na ważności bazowego wyzwalacza problemu, jednak po wystąpieniu zdarzenia może zostać zaktualizowana za pomocą ekranu *Aktualizacja problemu*. Kolor ważności problemu jest używany jako tło komórki przez czas trwania problemu.
Czas rozwiązania	Wyświetlany jest czas rozwiązania problemu.
Status	Wyświetlany jest status problemu: **Problem** - nierozwiązany problem **Resolved** - niedawno rozwiązany problem. Możesz ukryć niedawno rozwiązane problemy za pomocą filtra. Nowe i niedawno rozwiązane problemy migają przez 2 minuty. Rozwiązane problemy są wyświetlane łącznie przez 5 minut. Obie te wartości można skonfigurować w *Administracja* > *Ogólne* > *Opcje wyświetlania wyzwalaczy*.
Informacje	Wyświetlana jest zielona ikona informacji, jeśli problem został zamknięty przez globalną korelację lub ręcznie podczas aktualizacji problemu. Najechanie kursorem na ikonę spowoduje wyświetlenie dodatkowych szczegółów:  Poniższa ikona jest wyświetlana, jeśli pokazywany jest problem z wyciszeniem (zobacz opcję *Pokaż wyciszone problemy* w filtrze). Najechanie kursorem na ikonę spowoduje wyświetlenie dodatkowych szczegółów: 

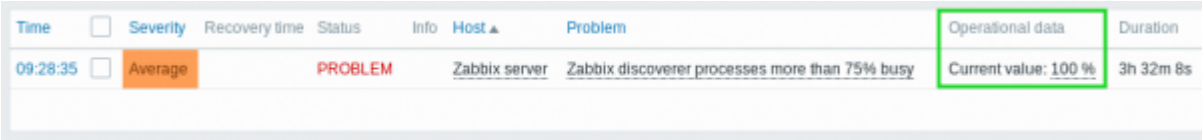
Kolumna	Opis
Host	Wyświetlany jest host problemu. Kliknięcie nazwy hosta otwiera menu hosta.
Problem	Wyświetlana jest nazwa problemu. Nazwa problemu jest oparta na nazwie bazowego wyzwalacza problemu. Makra w nazwie wyzwalacza są rozwijane w momencie wystąpienia problemu i rozwinięte wartości nie są już później aktualizowane. *Uwaga*: możliwe jest dołączenie do nazwy problemu danych operacyjnych pokazujących niektóre najnowsze wartości pozycji. Kliknięcie nazwy problemu otwiera menu zdarzenia. Najechanie kursorem na ikonę po nazwie problemu spowoduje wyświetlenie opisu wyzwalacza (dla tych problemów, które go mają).
Dane operacyjne	Wyświetlane są dane operacyjne zawierające najnowsze wartości pozycji. Dane operacyjne mogą być kombinacją tekstu i makr wartości pozycji, jeśli zostały skonfigurowane na poziomie wyzwalacza. Jeśli dane operacyjne nie są skonfigurowane na poziomie wyzwalacza, wyświetlane są najnowsze wartości wszystkich pozycji z wyrażenia. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze dla opcji *Pokaż dane operacyjne* wybrano *Oddzielnie*.
Czas trwania	Wyświetlany jest czas trwania problemu. Zobacz także: Ujemny czas trwania problemu
Aktualizacja	Kliknij odnośnik *Aktualizacja*, aby przejść do ekranu aktualizacji problemu, gdzie można wykonać różne działania dotyczące problemu, w tym dodać komentarz i potwierdzić problem.
Działania	Historia aktywności dotyczących problemu jest wyświetlana za pomocą symbolicznych ikon: - problem został potwierdzony. Ta ikona jest zawsze wyświetlana jako pierwsza. - dodano komentarze. Wyświetlana jest również liczba komentarzy. - ważność problemu została zwiększona (np. Information > Warning). - ważność problemu została zmniejszona (np. Warning > Information). - ważność problemu została zmieniona, ale wróciła do pierwotnego poziomu (np. Warning > Information > Warning). - problem został wyciszony. - wyciszenie problemu zostało usunięte. - wykonano działania. Wyświetlana jest również liczba działań. - wykonano działania, co najmniej jedno jest w toku. Wyświetlana jest również liczba działań. - wykonano działania, co najmniej jedno zakończyło się niepowodzeniem. Wyświetlana jest również liczba działań. Po najechaniu kursorem na ikony wyświetlane są wyskakujące okna ze szczegółami aktywności. Zobacz wyświetlanie szczegółów, aby dowiedzieć się więcej o ikonach używanych w wyskakującym oknie dla wykonanych działań.
Tagi	Wyświetlane są tagi (jeśli istnieją). Dodatkowo mogą być również wyświetlane tagi z zewnętrznego systemu zgłoszeń (zobacz opcję *Process tags* podczas konfigurowania webhooków).

Dane operacyjne problemów

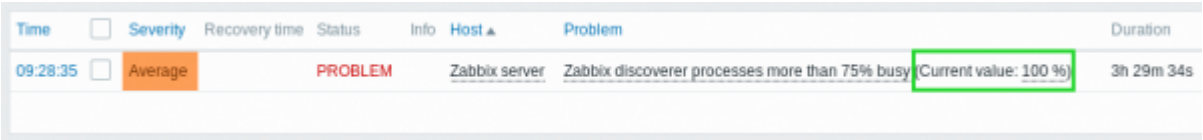
Możliwe jest wyświetlanie danych operacyjnych dla bieżących problemów, tj. najnowszych wartości pozycji zamiast wartości pozycji z momentu wystąpienia problemu.

Wyświetlanie danych operacyjnych można skonfigurować w filtrze *Monitoring* > *Problemy* lub w konfiguracji odpowiedniego widżetu dashboardu, wybierając jedną z trzech opcji:

- *Brak* - nie są wyświetlane żadne dane operacyjne
- *Osobno* - dane operacyjne są wyświetlane w osobnej kolumnie



- *Z nazwą problemu* - dane operacyjne są dołączane do nazwy problemu i umieszczane w nawiasach. Dane operacyjne są dołączane do nazwy problemu tylko wtedy, gdy pole *Dane operacyjne* nie jest puste w konfiguracji wyzwalacza.



Zawartość danych operacyjnych można skonfigurować dla każdego wyzwalacza w polu *Dane operacyjne*. Pole to akceptuje dowolny ciąg znaków z makrami, a przede wszystkim makro `{ITEM.LASTVALUE<1-9>}`.

`{ITEM.LASTVALUE<1-9>}` w tym polu zawsze zostanie rozwinięte do najnowszych wartości pozycji w wyrażeniu wyzwalacza. `{ITEM.VALUE<1-9>}` w tym polu zostanie rozwinięte do wartości pozycji z momentu zmiany stanu wyzwalacza (tj. przejścia w stan problemu, przejścia w stan OK, zamknięcia ręcznego przez użytkownika lub zamknięcia przez korelację).

Należy pamiętać, że ręczne zamknięcie problemu nie powoduje utworzenia nowej wartości, więc rozwinięta wartość `{ITEM.LASTVALUE<1-9>}` lub `{ITEM.VALUE<1-9>}` nadal będzie pokazywać wartość z czasu wystąpienia problemu.

`{ITEM.LASTVALUE<1-9>}` lub `{ITEM.VALUE<1-9>}` zostanie rozwinięte do *UNKNOWN* jeśli najnowsza wartość historyczna została zebrana ponad czas *Maksymalny okres wyświetlania historii* temu (zobacz *Administracja > Ogólne*).

Ujemny czas trwania problemu

W niektórych typowych sytuacjach rzeczywiście możliwe jest wystąpienie ujemnego czasu trwania problemu, tj. gdy czas rozwiązania problemu jest wcześniejszy niż czas utworzenia problemu, np.:

- Jeśli jakiś host jest monitorowany przez proxy i wystąpi błąd sieci, w wyniku którego przez pewien czas nie są odbierane dane z proxy, wyzwalacz nodata(/host/key) zostanie uruchomiony przez serwer. Po przywróceniu połączenia serwer otrzyma dane pozycji z proxy z czasem z przeszłości. Następnie problem nodata(/host/key) zostanie rozwiązany i będzie miał ujemny czas trwania problemu;
- Gdy dane pozycji rozwiązujące zdarzenie problemu są wysyłane przez Zabbix sender i zawierają znacznik czasu wcześniejszy niż czas utworzenia problemu, również zostanie wyświetlony ujemny czas trwania problemu.

Note:
Ujemny czas trwania problemu nie wpływa w żaden sposób na obliczanie SLA ani raport dostępności dla danego wyzwalacza; nie skraca ani nie wydłuża czasu trwania problemu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Masowa aktualizacja* — aktualizuj wybrane problemy, przechodząc do ekranu aktualizacji problemu

Aby skorzystać z tej opcji, zaznacz pola wyboru przy odpowiednich problemach, a następnie kliknij przycisk *Masowa aktualizacja*.

Przyciski

Przycisk po prawej stronie oferuje następującą opcję:

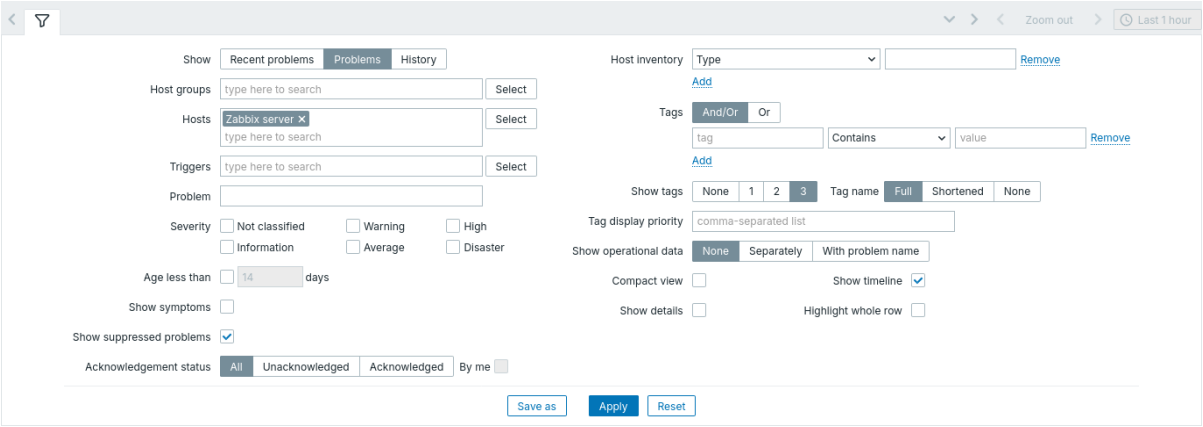
Export to CSV	Eksportuj zawartość ze wszystkich stron do pliku CSV.

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko problemy, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Filtr znajduje się nad tabelą. Ulubione ustawienia filtra można zapisać jako karty, a następnie szybko otwierać, klikając karty nad filtrem.



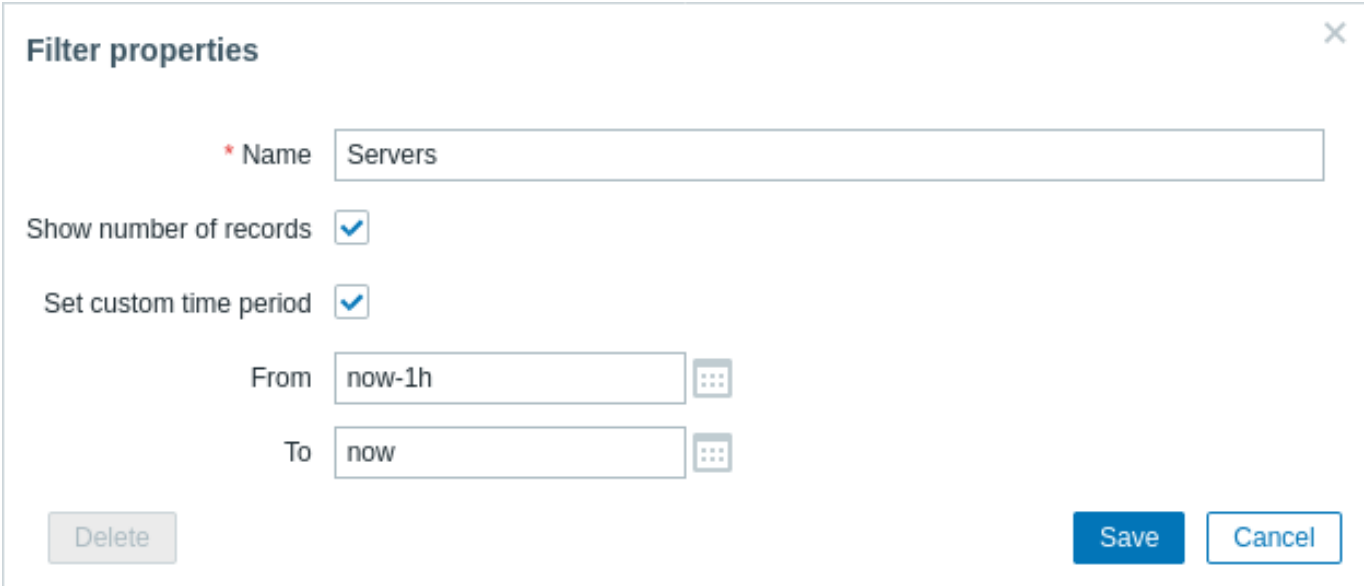
Parameter	Description
Show	Filtruj według statusu problemu: **Recent problems** - wyświetlane są nierozwiązane i ostatnio rozwiązane problemy (domyślnie) **Problems** - wyświetlane są nierozwiązane problemy **History** - wyświetlana jest historia wszystkich zdarzeń
Host groups	Filtruj według jednej lub wielu grup hostów. Określenie nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów.
Hosts	Filtruj według jednego lub wielu hostów.
Triggers	Filtruj według jednego lub wielu wyzwalaczy.
Problem	Filtruj według nazwy problemu.
Severity	Filtruj według ważności wyzwalacza (problemu).
Age less than	Filtruj według wieku problemu.
Show symptoms	Zaznacz pole wyboru, aby wyświetlać w osobnym wierszu problemy sklasyfikowane jako symptomy.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta lub pojedynczego tłumienia problemów.
Acknowledgement status	Filtruj, aby wyświetlać wszystkie problemy, tylko niepotwierdzone problemy lub tylko potwierdzone problemy. Zaznacz dodatkowe pole wyboru, aby odfiltrować te problemy, które kiedykolwiek zostały przez Ciebie potwierdzone.
Host inventory	Filtruj według typu i wartości inwentarza.
Tags	Filtruj według nazwy i wartości tagu zdarzenia. Możliwe jest uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżniana wielkość liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżniana wielkość liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek Po zastosowaniu filtra tagi określone tutaj będą wyświetlane jako pierwsze wraz z problemem, chyba że zostanie to zastąpione przez listę *Tag display priority* (patrz poniżej).
Show tags	Wybierz liczbę wyświetlanych tagów: **None** - brak kolumny *Tags* w *Monitoring* > *Problems* **1** - kolumna *Tags* zawiera jeden tag **2** - kolumna *Tags* zawiera dwa tagi **3** - kolumna *Tags* zawiera trzy tagi Aby zobaczyć wszystkie tagi dla problemu, najedź kursorem myszy na ikonę trzech kropek.

Parameter	Description
Tag name	Wybierz tryb wyświetlania nazw tagów: **Full** - nazwy i wartości tagów są wyświetlane w pełnej formie **Shortened** - nazwy tagów są skracane do 3 znaków; wartości tagów są wyświetlane w pełnej formie **None** - wyświetlane są tylko wartości tagów; bez nazw
Tag display priority	Wprowadź priorytet wyświetlania tagów dla problemu jako listę tagów rozdzieloną przecinkami (na przykład: `customer,scope,component`). Należy używać wyłącznie nazw tagów, bez wartości. Tagi z tej listy będą zawsze wyświetlane jako pierwsze, zastępując naturalne sortowanie alfabetyczne.
Show operational data	Wybierz tryb wyświetlania danych operacyjnych: **None** - dane operacyjne nie są wyświetlane **Separately** - dane operacyjne są wyświetlane w osobnej kolumnie **With problem name** - dane operacyjne są dołączane do nazwy problemu, z użyciem nawiasów dla danych operacyjnych
Compact view	Zaznacz pole wyboru, aby włączyć widok kompaktowy.
Show details	Zaznacz pole wyboru, aby wyświetlać podstawowe wyrażenia wyzwalaczy problemów. Wyłączone, jeśli zaznaczono pole wyboru *Compact view*.
Show timeline	Zaznacz pole wyboru, aby wyświetlać wizualną oś czasu i grupowanie. Wyłączone, jeśli zaznaczono pole wyboru *Compact view*.
Highlight whole row	Zaznacz pole wyboru, aby podświetlać cały wiersz dla nierozwiązanych problemów. Do podświetlania używany jest kolor ważności problemu. *Highlight whole row* nie jest dostępne w motywach o wysokim kontraście. W wersjach Zabbixa wcześniejszych niż 7.4.3 ta opcja jest obsługiwana tylko wtedy, gdy w standardowych niebieskim i ciemnym motywie zaznaczono pole wyboru *Compact view*.

Karty ulubionych filtrów

Często używane zestawy parametrów filtra można zapisać na kartach.

Aby zapisać nowy zestaw parametrów filtra, otwórz kartę główną i skonfiguruj ustawienia filtra, a następnie naciśnij przycisk *Save as*. W nowym wyskakującym oknie zdefiniuj *Właściwości filtra*.

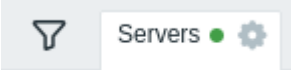


Parametr	Opis
Name	Nazwa filtra wyświetlana na liście kart.
Show number of records	Zaznacz, jeśli chcesz, aby liczba problemów była wyświetlana obok nazwy karty.
Override time period selector	Zaznacz, aby ustawić określony domyślny okres czasu dla tego zestawu filtrów. Jeśli opcja jest ustawiona, okres czasu dla tej karty będzie można zmienić tylko przez aktualizację ustawień filtra. Dla kart bez niestandardowego okresu czasu zakres czasu można zmienić, naciskając przycisk selektora czasu w prawym górnym rogu (nazwa przycisku zależy od wybranego przedziału czasu: This week, Last 30 minutes, Yesterday itd.). Ta opcja jest dostępna tylko dla filtrów w *Monitoring* > *Problems*.

Parametr	Opis
From/To	Początek i koniec okresu czasu w składni czasu bezwzględnego (`Y-m-d H:i:s`) lub względnego (`now-1d`). Dostępne, jeśli zaznaczono *Set custom time period*.

Po zapisaniu filtr jest tworzony jako nazwana karta filtra i natychmiast aktywowany.

Aby edytować właściwości istniejącego filtra, naciśnij symbol koła zębatego obok nazwy aktywnej karty.



Uwagi:

- Aby ukryć obszar filtra, kliknij nazwę bieżącej karty. Kliknij ponownie nazwę aktywnej karty, aby ponownie otworzyć obszar filtra.
- Obsługiwana jest nawigacja za pomocą klawiatury: użyj strzałek, aby przełączać się między kartami, naciśnij *Enter*, aby otworzyć.
- Przyciski w lewo/w prawo nad filtrem mogą być używane do przełączania między zapisanymi filtrami. Alternatywnie przycisk skierowany w dół otwiera menu rozwijane ze wszystkimi zapisanymi filtrami i możesz kliknąć ten, którego potrzebujesz.
- Karty filtrów można przestawiać metodą przeciągnij i upuść.
- Jeśli ustawienia zapisanego filtra zostały zmienione (ale nie zapisane), po nazwie filtra wyświetlana jest zielona kropka. Aby zaktualizować filtr zgodnie z nowymi ustawieniami, kliknij przycisk *Update*, który jest wyświetlany zamiast przycisku *Save as*.
- Bieżące ustawienia filtra są zapamiętywane w profilu użytkownika. Gdy użytkownik ponownie otworzy stronę, ustawienia filtra pozostaną takie same.

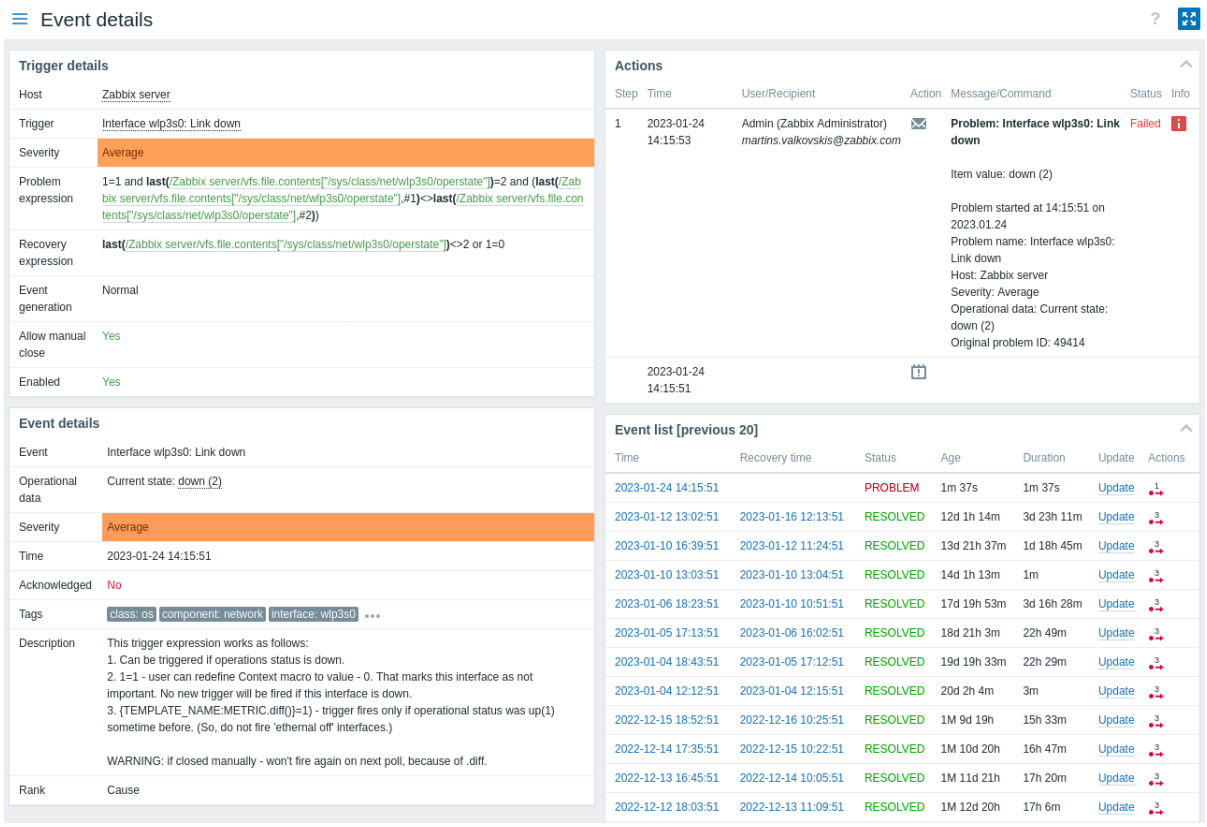
Note:
Aby udostępnić filtry, skopiuj adres URL aktywnego filtra i wyślij go innym. Po otwarciu tego adresu URL inni użytkownicy będą mogli zapisać ten zestaw parametrów jako trwały filtr na swoim koncie Zabbix.
 Zobacz także: Parametry strony.

Przyciski filtra

Apply	Zastosuj określone kryteria filtrowania (bez zapisywania).
Reset	Zresetuj bieżący filtr i przywróć zapisane parametry bieżącej karty. Na karcie głównej spowoduje to wyczyszczenie filtra.
Save as	Zapisz bieżące parametry filtra na nowej karcie. Dostępne tylko na karcie głównej.
Update	Zastąp parametry karty aktualnie określonymi parametrami. Niedostępne na karcie głównej.

Wyświetlanie szczegółów

Czasy rozpoczęcia problemu i przywrócenia w *Monitoring* > *Problems* są linkami. Kliknięcie ich otwiera więcej szczegółów zdarzenia.



Należy pamiętać, że ważność problemu może różnić się dla wyzwalacza i zdarzenia problemu — jeśli została zaktualizowana dla zdarzenia problemu przy użyciu ekranu *Update problem*.

Na liście działań używane są następujące ikony do oznaczenia typu aktywności:

- - wygenerowano zdarzenie problemu.
- - wiadomość została wysłana.
- - zdarzenie problemu zostało potwierdzone.
- - cofnięto potwierdzenie zdarzenia problemu.
- - dodano komentarz.
- - ważność problemu została zwiększona (np. Information > Warning).
- - ważność problemu została zmniejszona (np. Warning > Information).
- - ważność problemu została zmieniona, ale wróciła do pierwotnego poziomu (np. Warning > Information > Warning).
- - wykonano zdalne polecenie.
- - zdarzenie problemu zostało przywrócone.
- - problem został zamknięty ręcznie.
- - problem został wyciszony.
- - wyłączono wyciszenie problemu.
- - problem został przekształcony w problem objawowy.
- - problem został przekształcony w problem przyczynowy.

1 Problemy przyczyn i objawów

Przegląd

Domyślnie wszystkie nowe problemy są klasyfikowane jako problemy przyczynowe. Możliwe jest ręczne przeklasyfikowanie

określonych problemów na problemy objawowe problemu przyczynowego.

Na przykład awaria zasilania może być rzeczywistą przyczyną źródłową, przez którą jakiś host jest nieosiągalny lub jakaś usługa nie działa. W takim przypadku problemy „host jest nieosiągalny" i „usługa nie działa" muszą zostać sklasyfikowane jako problemy objawowe problemu „awaria zasilania" — problemu przyczynowego.

Hierarchia przyczyna-objaw obsługuje tylko dwa poziomy. Problemu, który jest już objawem, nie można przypisać „podrzędnych" problemów objawowych; wszelkie problemy przypisane jako objawy do problemu objawowego staną się objawami tego samego problemu przyczynowego.

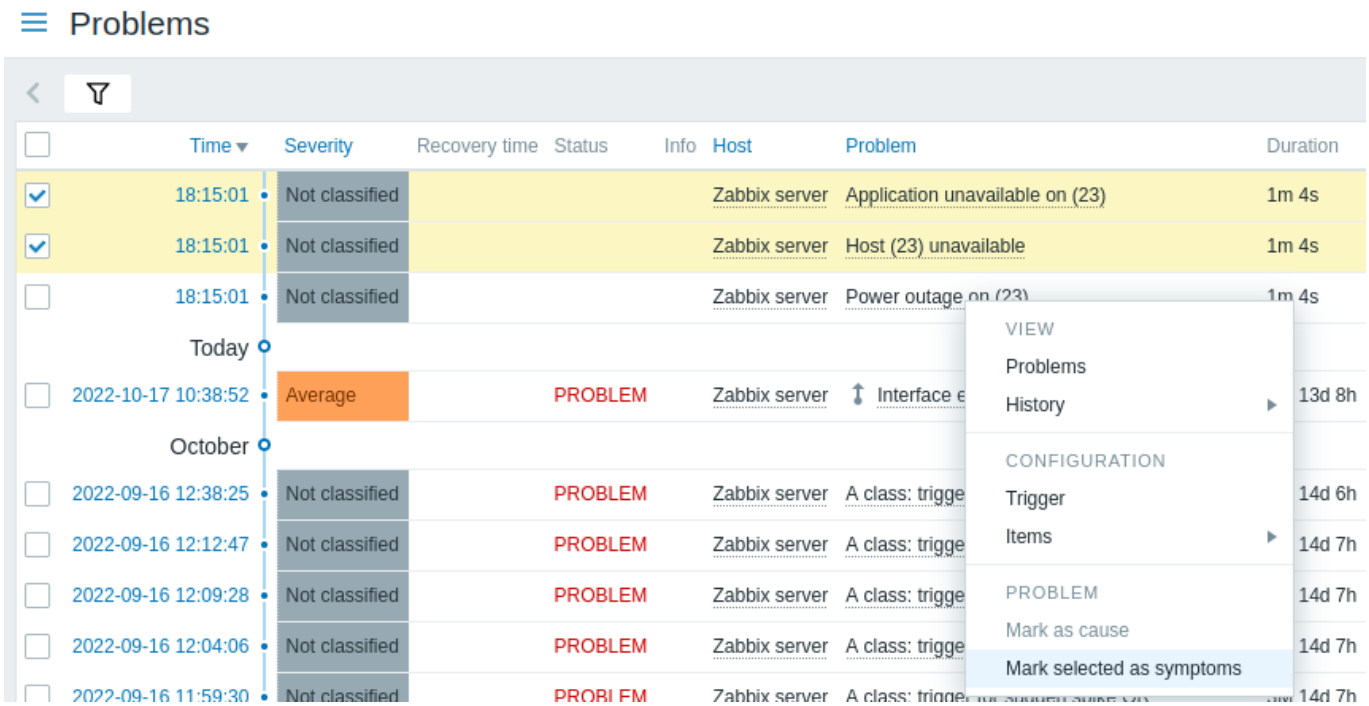
W sumach problemów na mapach, w widżetach pulpitu, takich jak *Problemy według ważności* lub *Hosty problemowe* itd., uwzględniane są tylko problemy przyczynowe. Jednak klasyfikacja problemów nie wpływa na usługi.

Problem objawowy może być powiązany tylko z jednym problemem przyczynowym. Problemy objawowe nie są rozwiązywane automatycznie, jeśli problem przyczynowy zostanie rozwiązany lub zamknięty.

Konfiguracja

Aby przeklasyfikować problem na problem będący symptomem, najpierw wybierz go na liście problemów. Można wybrać jeden lub kilka problemów.

Następnie przejdź do problemu-przyczyny i w jego menu kontekstowym kliknij opcję *Oznacz wybrane jako symptomy*.



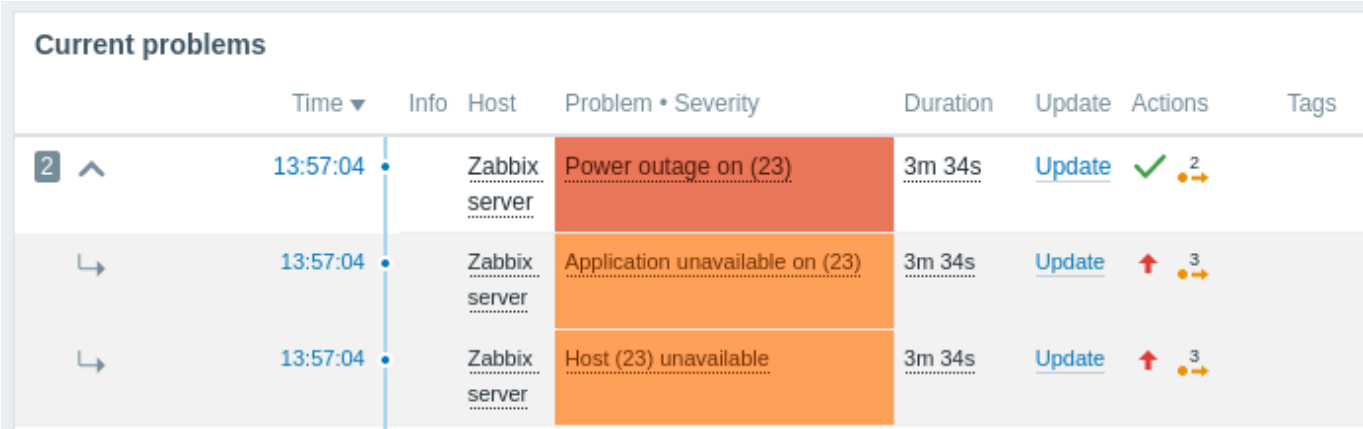
Po tym wybrane problemy zostaną zaktualizowane przez serwer i staną się problemami będącymi symptomami problemu-przyczyny.

Podczas aktualizacji statusu problemu jest on wyświetlany na jeden z dwóch sposobów:

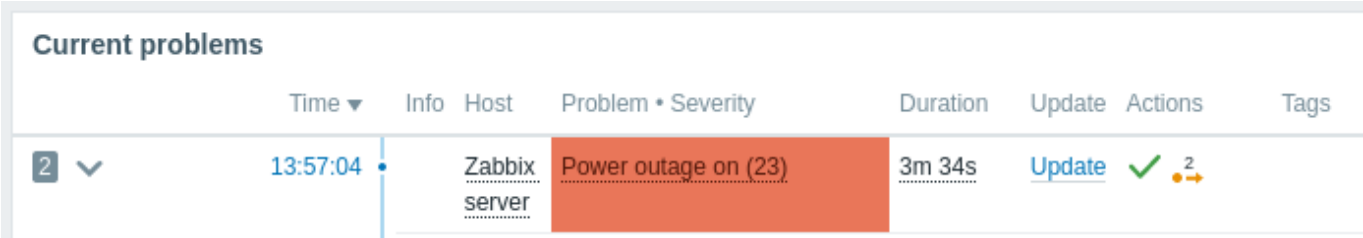
- W kolumnie Status wyświetlany jest migający status „UPDATING";
- Migająca ikona lub w kolumnie Info (ma to zastosowanie, jeśli w filtrze wybrano tylko *Problemy*, a zatem kolumna Status nie jest wyświetlana).

Wyświetlanie

Problemy symptomów są wyświetlane poniżej problemu przyczyny i odpowiednio oznaczone w *Monitoring -> Problems* (oraz w widżecie pulpitu *Problems*) — za pomocą ikony, mniejszej czcionki i innego tła.



W widoku zwiniętym widoczny jest tylko problem przyczyny; istnienie problemów symptomów jest wskazywane przez liczbę na początku wiersza oraz ikonę rozwijania widoku.



Możliwe jest również dodatkowe wyświetlanie problemów symptomów normalną czcionką i w osobnym wierszu. W tym celu wybierz *Show symptoms* w ustawieniach filtra lub konfiguracji widżetu.

Przywracanie do problemu przyczyny

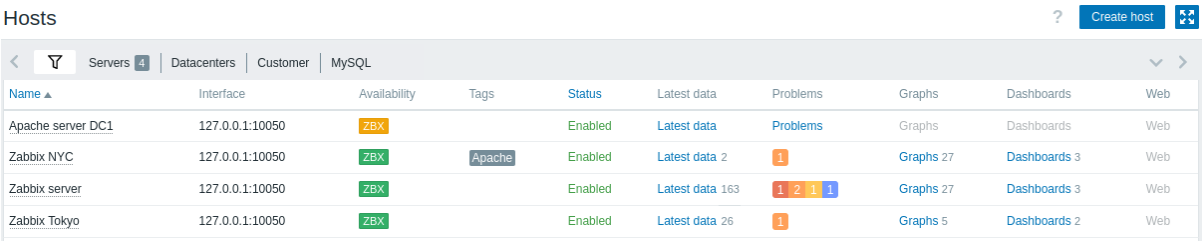
Problem objawu można przywrócić do problemu przyczyny. Aby to zrobić, należy:

- kliknąć opcję *Oznacz jako przyczynę* w menu kontekstowym problemu objawu;
- zaznaczyć opcję *Przekształć w przyczynę* na ekranie aktualizacji problemu i kliknąć *Aktualizuj* (ta opcja zadziała również, jeśli wybrano kilka problemów).

2 Hosty

Przegląd

Sekcja *Monitoring → Hosts* wyświetla pełną listę monitorowanych hostów ze szczegółowymi informacjami o interfejsie hosta, dostępności, tagach, bieżących problemach, statusie (włączony/wyłączony) oraz linkami umożliwiającymi łatwe przechodzenie do najnowszych danych hosta, historii problemów, wykresów, dashboardów i scenariuszy web.



Column	Description
Name	Widoczna nazwa hosta. Kliknięcie nazwy otwiera menu hosta. Pomarańczowa ikona klucza po nazwie oznacza, że ten host znajduje się w trybie konserwacji. Kliknij nagłówek kolumny, aby sortować hosty według nazwy rosnąco lub malejąco.
Interface	Wyświetlany jest główny interfejs hosta.

Column	Description
Availability	Wyświetlana jest dostępność hosta dla skonfigurowanego interfejsu. Ikony dostępności reprezentują bieżący status interfejsu hosta na serwerze Zabbix. Dlatego jeśli wyłączysz hosta we frontendzie, jego dostępność zostanie zaktualizowana po zsynchronizowaniu przez serwer Zabbix zmian konfiguracji. Podobnie, jeśli włączysz hosta, jego dostępność zostanie zaktualizowana po zsynchronizowaniu przez serwer Zabbix zmian konfiguracji i odpytywaniu hosta. Ikony dostępności reprezentują tylko te typy interfejsów (Agent, SNMP, IPMI, JMX), które są skonfigurowane. Po najechaniu kursorem na ikonę wyświetlane jest okno podręczne z listą wszystkich interfejsów tego samego typu wraz ze szczegółami, statusem i błędami. Dla interfejsu Agent okno podręczne wyświetla interfejsy (pasywne) oraz aktywne sprawdzenia. Jeśli host ma tylko aktywne sprawdzenia, ikona interfejsu Agent jest wyświetlana nawet wtedy, gdy host nie ma skonfigurowanego interfejsu Agent. Kolumna jest pusta dla hostów bez interfejsów. Status pojedynczego interfejsu hosta jest określany na podstawie połączenia między włączoną pozycją korzystającą z tego interfejsu a hostem. Status może być następujący: **Available** - połączenie z hostem zakończyło się powodzeniem; **Not available** - połączenie z hostem nie powiodło się (przekroczenie limitu czasu, problemy z zaporą sieciową itp.); **Unknown** - nie podjęto próby połączenia z hostem lub wynik jest nieznany. Dodatkowe informacje o tym, jak serwer Zabbix określa status interfejsu, można znaleźć w sekcjach Unknown interface status oraz Unreachable/unavailable host interface settings. Status wszystkich interfejsów hosta jednego typu (Agent, SNMP, IPMI, JMX) jest określany na podstawie tych interfejsów, które są używane przez co najmniej jedną włączoną pozycję. Status jest oznaczany kolorem ikony: **Green** - wszystkie interfejsy są dostępne; **Yellow** - co najmniej jeden interfejs jest niedostępny, a co najmniej jeden jest dostępny lub nieznany; **Red** - wszystkie interfejsy są niedostępne; **Gray** - co najmniej jeden interfejs jest nieznany, ale żaden nie jest niedostępny. **Dostępność aktywnych sprawdzeń.** Jeśli na hoście włączono co najmniej jedno aktywne sprawdzenie, aktywne sprawdzenia również wpływają na łączną dostępność interfejsu Agent, jak opisano powyżej. Aby określić dostępność aktywnych sprawdzeń, komunikaty heartbeat są wysyłane w wątku aktywnych sprawdzeń agenta. Częstotliwość komunikatów heartbeat jest □□□□□□□owana przez parametr `HeartbeatFrequency` w konfiguracji Zabbix agent lub agent 2 (domyślnie 60 sekund, zakres 0-3600). Aktywne sprawdzenia są uznawane za niedostępne, gdy heartbeat aktywnego sprawdzenia jest starszy niż 2 x HeartbeatFrequency sekund. **Uwaga:** Agenty Zabbix starsze niż wersja 6.2.x nie wysyłają heartbeatów aktywnych sprawdzeń, więc dostępność ich hostów pozostaje nieznana.
Tags	Tagi hosta oraz wszystkich powiązanych szablonów, z nierozwiniętymi makrami.
Status	Status hosta - *Enabled* lub *Disabled*. Kliknij nagłówek kolumny, aby sortować hosty według statusu rosnąco lub malejąco.
Latest data	Kliknięcie linku otworzy stronę *Monitoring* **-** *Latest data* z wszystkimi najnowszymi danymi zebranymi z hosta. Liczba pozycji z najnowszymi danymi jest wyświetlana na szaro.
Problems	Liczba otwartych problemów hosta posortowanych według ważności. Kolor kwadratu wskazuje ważność problemu. Liczba na kwadracie oznacza liczbę problemów dla danej ważności. Kliknięcie ikony otworzy stronę *Monitoring* **-** *Problems* dla bieżącego hosta. Jeśli host nie ma problemów, wyświetlany jest link do sekcji *Problems* dla tego hosta w formie tekstu. Użyj filtra, aby wybrać, czy uwzględniać problemy wyciszone (domyślnie nie są uwzględniane).
Graphs	Kliknięcie linku wyświetli wykresy skonfigurowane dla hosta. Liczba wykresów jest wyświetlana na szaro. Jeśli host nie ma wykresów, link jest wyłączony (szary tekst) i nie jest wyświetlana żadna liczba.

Column	Description
Dashboards	Kliknięcie linku wyświetli dashboardy skonfigurowane dla hosta. Liczba dashboardów jest wyświetlana na szaro. Jeśli host nie ma dashboardów, link jest wyłączony (szary tekst) i nie jest wyświetlana żadna liczba.
Web	Kliknięcie linku wyświetli scenariusze web skonfigurowane dla hosta. Liczba scenariuszy web jest wyświetlana na szaro. Jeśli host nie ma scenariuszy web, link jest wyłączony (szary tekst) i nie jest wyświetlana żadna liczba.

Przyciski

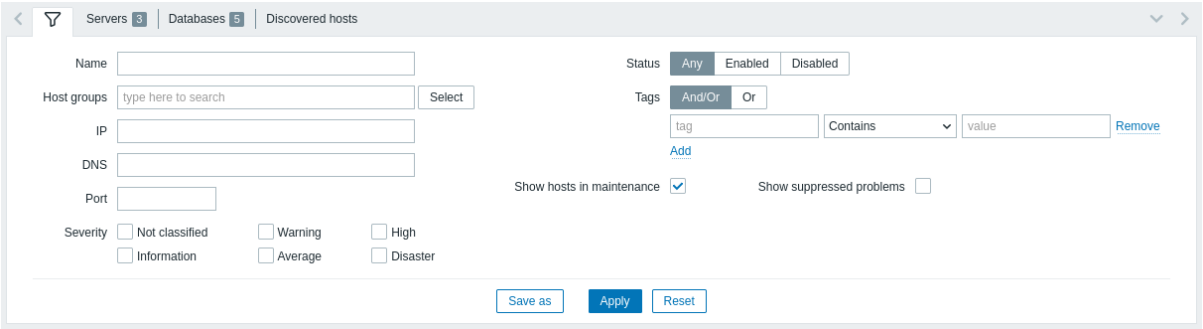
Utwórz host umożliwia utworzenie nowego hosta. Ten przycisk jest dostępny tylko dla użytkowników typu Admin i Super Admin.

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko hosty, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Filtr znajduje się nad tabelą. Możliwe jest filtrowanie hostów według nazwy, grupy hostów, IP lub DNS, portu interfejsu, tagów, ważności problemu, statusu (włączony/wyłączony/dowolny); możesz także wybrać, czy wyświetlać stłumione problemy oraz hosty, które są obecnie w konserwacji.



Parametr	Opis
Name	Filtruj według widocznej nazwy hosta.
Host groups	Filtruj według jednej lub większej liczby grup hostów. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów.
IP	Filtruj według adresu IP.
DNS	Filtruj według nazwy DNS.
Port	Filtruj według numeru portu.
Severity	Filtruj według ważności problemu. Domyślnie wyświetlane są problemy o wszystkich poziomach ważności. Problemy są wyświetlane, jeśli nie są stłumione.
Status	Filtruj według statusu hosta.

Parametr	Opis
Tags	Filtruj według nazwy i wartości tagu hosta. Hosty mogą być filtrowane według tagów na poziomie hosta, jak również tagów ze wszystkich podłączonych szablonów, w tym szablonów zagnieżdżonych. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane operatorem Or **Or** - wystarczy, że jeden warunek zostanie spełniony
Show hosts in maintenance	Zaznacz pole wyboru, aby wyświetlić hosty będące w konserwacji (domyślnie są wyświetlane).
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlić problemy, które w przeciwnym razie byłyby stłumione (niepokazywane) z powodu konserwacji hosta lub pojedynczego problem suppression.

Zapisywanie filtra

Ulubione ustawienia filtra można zapisać jako zakładki, a następnie szybko uzyskać do nich dostęp, klikając odpowiednią zakładkę nad filtrem.

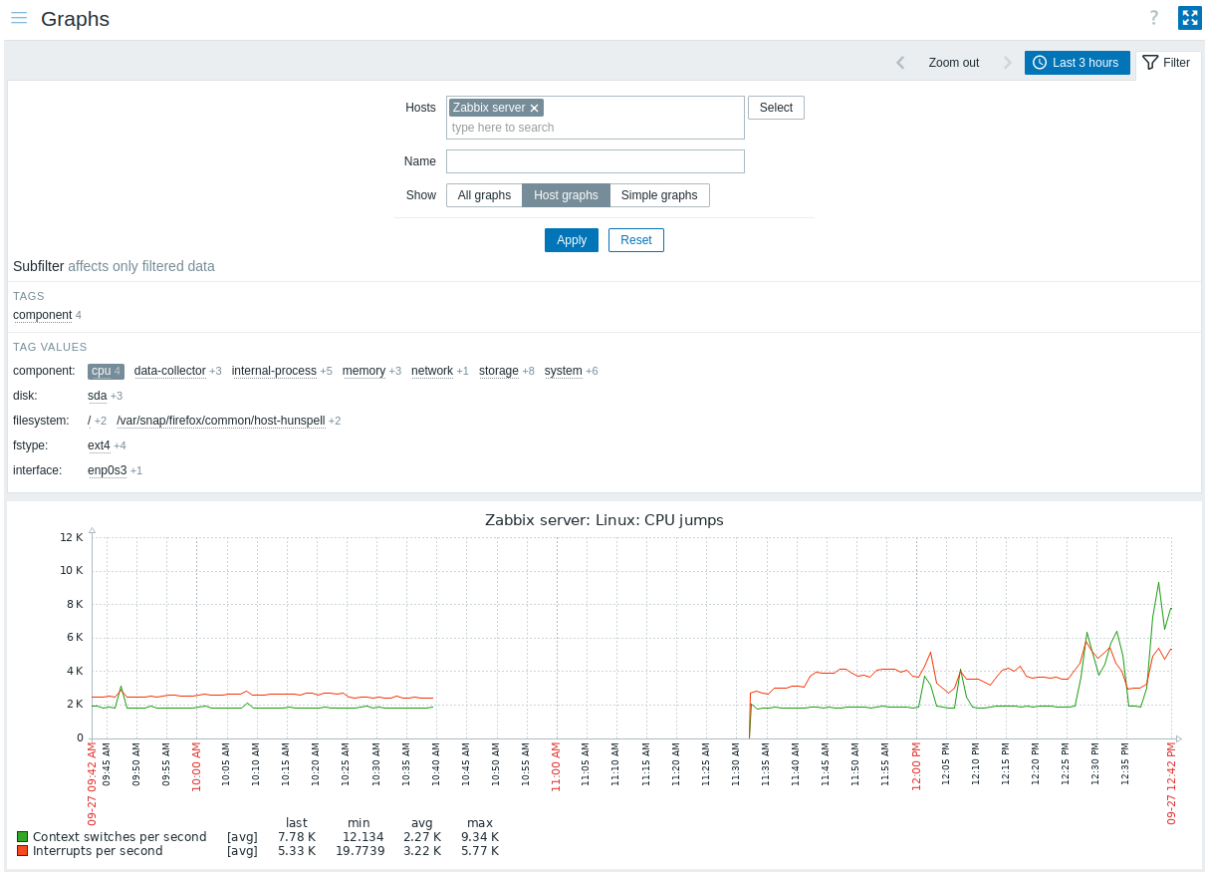
Zobacz więcej szczegółów o zapisywaniu filtrów.

1 Wykresy

Przegląd

Wykresy hosta są dostępne w *Monitoring* → *Hosts* po kliknięciu Wykresy dla odpowiedniego hosta.

Można wyświetlić dowolny wykres niestandardowy, który został skonfigurowany dla hosta, a także dowolny wykres prosty.



Wykresy są sortowane według:

- nazwy wykresu (wykresy niestandardowe)
- nazwy pozycji (wykresy proste)

Wykresy dla wyłączonych hostów są również dostępne.

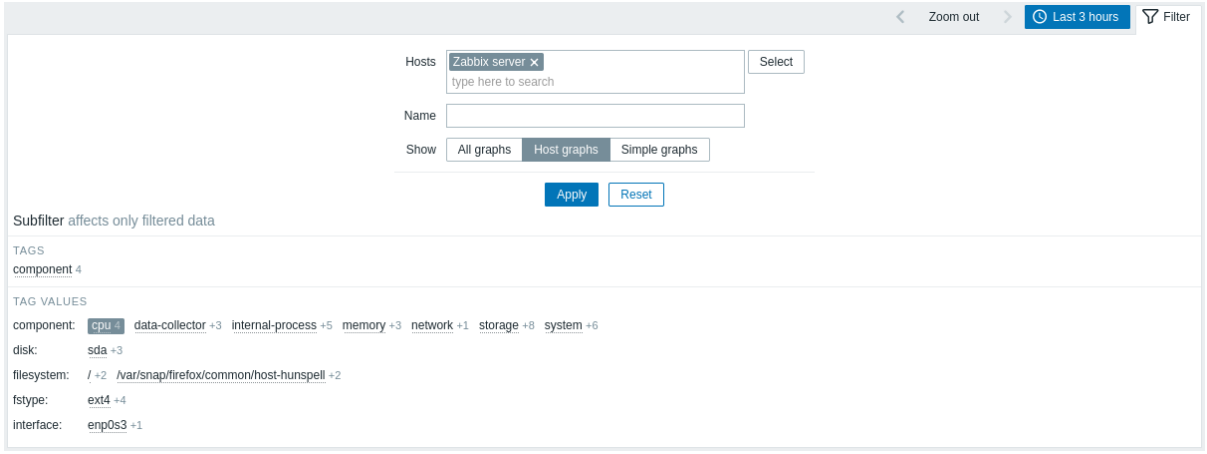
Selektor przedziału czasu

Zwróć uwagę na selektor przedziału czasu nad wykresem. Umożliwia on wybór często wymaganych przedziałów jednym kliknięciem myszy.

Zobacz także: *Selektor przedziału czasu*

Używanie filtra

Aby wyświetlić określony wykres, wybierz go w filtrze. Filtr umożliwia określenie hosta, nazwy wykresu oraz opcji *Pokaż* (wszystkie/wykresy hosta/proste wykresy).



Jeśli w filtrze nie wybrano hosta, żadne wykresy nie są wyświetlane.

Używanie podfiltra

Podfiltr umożliwia dalszą modyfikację filtrowania z filtra głównego.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych wykresów. Wykresy są powiązane wspólną jednostką — nazwą tagu lub jego wartością. Po kliknięciu nazwy/wartości tagu zostaje ona podświetlona szarym tłem, a wykresy są natychmiast filtrowane (bez potrzeby klikania *Apply* w filtrze głównym). Kliknięcie innej nazwy/wartości tagu dodaje ją do przefiltrowanych wyników. Ponowne kliknięcie nazwy/wartości tagu usuwa filtrowanie.

Podfiltry są generowane na podstawie przefiltrowanych danych, których liczba jest ograniczona do 1000 rekordów. Jeśli chcesz zobaczyć więcej rekordów w podfiltrze, musisz zwiększyć wartość parametru *Limit for search and filter results* (w *Administration -> General -> GUI*).

W przeciwieństwie do filtra głównego podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zapewnić aktualne informacje o dostępnych opcjach filtrowania i licznikach ich wystąpień.

Liczba wyświetlanych jednostek jest ograniczona poziomo do 100. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie jest ona klikalna. Listy pionowe (takie jak tagi i ich wartości) są ograniczone do 20 wpisów. Jeśli jest ich więcej, wyświetlana jest ikona trzech kropek; nie jest ona klikalna.

Liczba obok każdej klikalnej jednostki wskazuje liczbę wykresów, które znajdują się w wynikach filtra głównego.

Po wybraniu jednej jednostki liczby przy pozostałych dostępnych jednostkach są wyświetlane ze znakiem plus, wskazującym, ile wykresów można dodać do bieżącego wyboru.

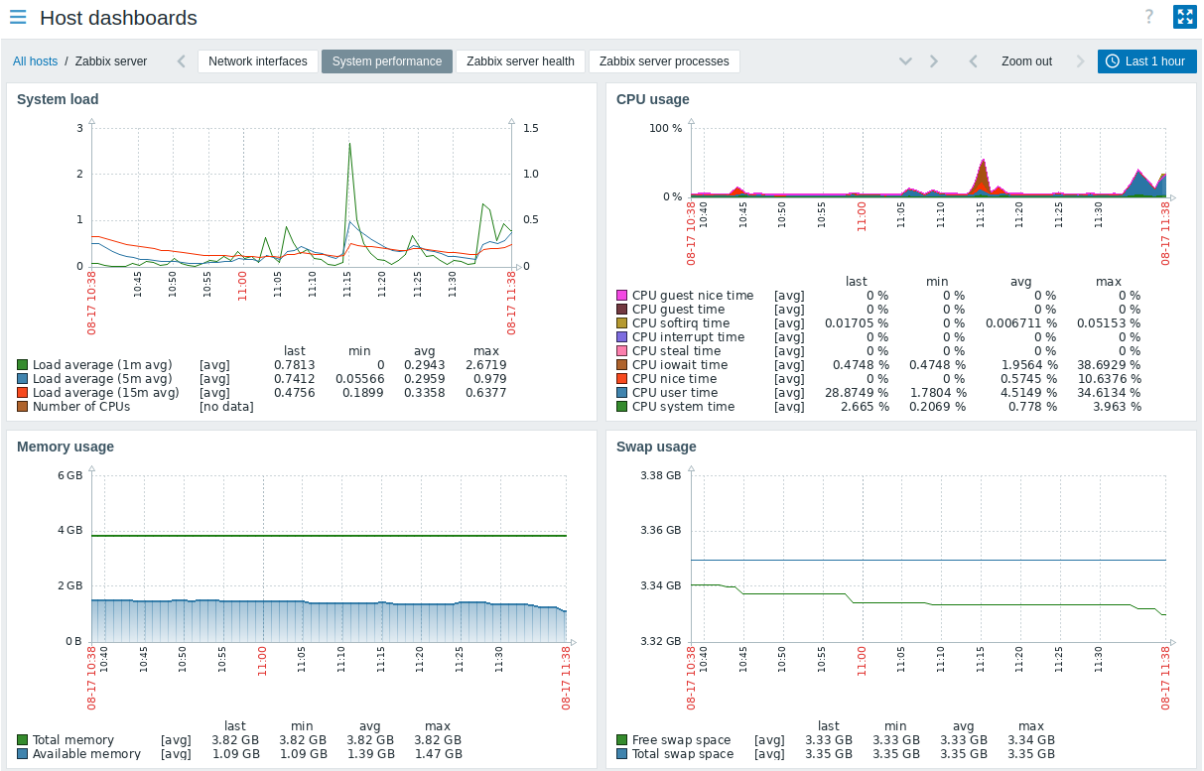
Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, są opisane na stronie Monitoring.

2 Pulpity hosta

Przegląd

Dashboardy hostów wyglądają podobnie do dashboardów globalnych; jednak dashboardy hostów nie mają właściciela i wyświetlają dane tylko dla wybranego hosta.



Podczas przeglądania dashboardów hostów można przełączać się między skonfigurowanymi dashboardami, klikając:

- karty dashboardów;
- przyciski strzałek pod nagłówkiem;
- przycisk strzałki pod nagłówkiem, który wyświetli pełną listę dostępnych dashboardów hostów.

Aby przejść do sekcji *Monitoring* → *Hosts*, kliknij łącze nawigacyjne *All hosts* pod nagłówkiem w lewym górnym rogu.

Konfiguracja

Pulpity hostów są konfigurowane na poziomie szablonu. Po podłączeniu szablonu do hosta dla tego hosta generowane są pulpity hosta. Należy pamiętać, że pulpitów hosta *nie można* konfigurować w sekcji *Pulpity*, która jest zarezerwowana dla pulpitów globalnych.

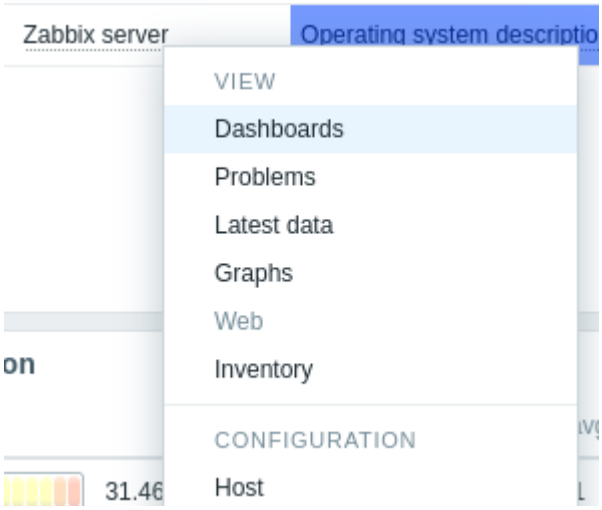
Widżety pulpitów hosta również można konfigurować wyłącznie na poziomie szablonu, z wyjątkiem zmiany interwału odświeżania. Ponadto widżety pulpitów hosta można kopiować wyłącznie do innych pulpitów hosta w obrębie tego samego szablonu. Należy pamiętać, że widżetów z pulpitów globalnych *nie można* kopiować do pulpitów hosta.

Note:
Pulpity hosta przed Zabbix 5.2 były ekranami hosta. Podczas importowania starszego szablonu zawierającego ekrany import ekranów zostanie zignorowany.

Dostęp

Do dashboardów hosta można uzyskać dostęp:

- po wyszukaniu nazwy hosta w wyszukiwaniu globalnym (kliknij link *Dashboards* widoczny w wynikach wyszukiwania);
- po kliknięciu nazwy hosta w *Inwentaryzacja* → *Hosty* (kliknij link *Dashboards* dostępny w podglądzie hosta);
- z menu hosta, klikając *Dashboards*.



Należy pamiętać, że dashboardy hosta *nie mogą* być dostępne bezpośrednio w sekcji *Dashboards*, która jest przeznaczona dla dashboardów globalnych.

3 Scenariusze WWW

Omówienie

Informacje o scenariuszu web hosta można uzyskać w sekcji *Monitoring* → *Hosts*, klikając Web dla odpowiedniego hosta.



Kliknięcie nazwy hosta powoduje wyświetlenie menu hosta. Dane wyłączonych hostów są również dostępne. Nazwa wyłączonego hosta jest wyświetlana na czerwono.

Maksymalna liczba scenariuszy wyświetlanych na stronie zależy od ustawienia profilu użytkownika *Rows per page* setting.

Domyślnie wyświetlane są tylko wartości mieszczące się w ostatnich 24 godzinach. Ograniczenie to wprowadzono w celu poprawy czasu początkowego ładowania dużych stron z najnowszymi danymi. Można wydłużyć ten okres, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration* → *General* → *GUI*.

Nazwa scenariusza jest linkiem do bardziej szczegółowych statystyk na jego temat:



Używanie filtra

Strona wyświetla listę wszystkich scenariuszy WWW wybranego hosta. Aby wyświetlić scenariusze WWW dla innego hosta lub grupy hostów bez powrotu do strony *Monitoring* → *Hosts*, wybierz tego hosta lub grupę w filtrze. Możesz również filtrować scenariusze na podstawie tagów.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

3 Najnowsze dane

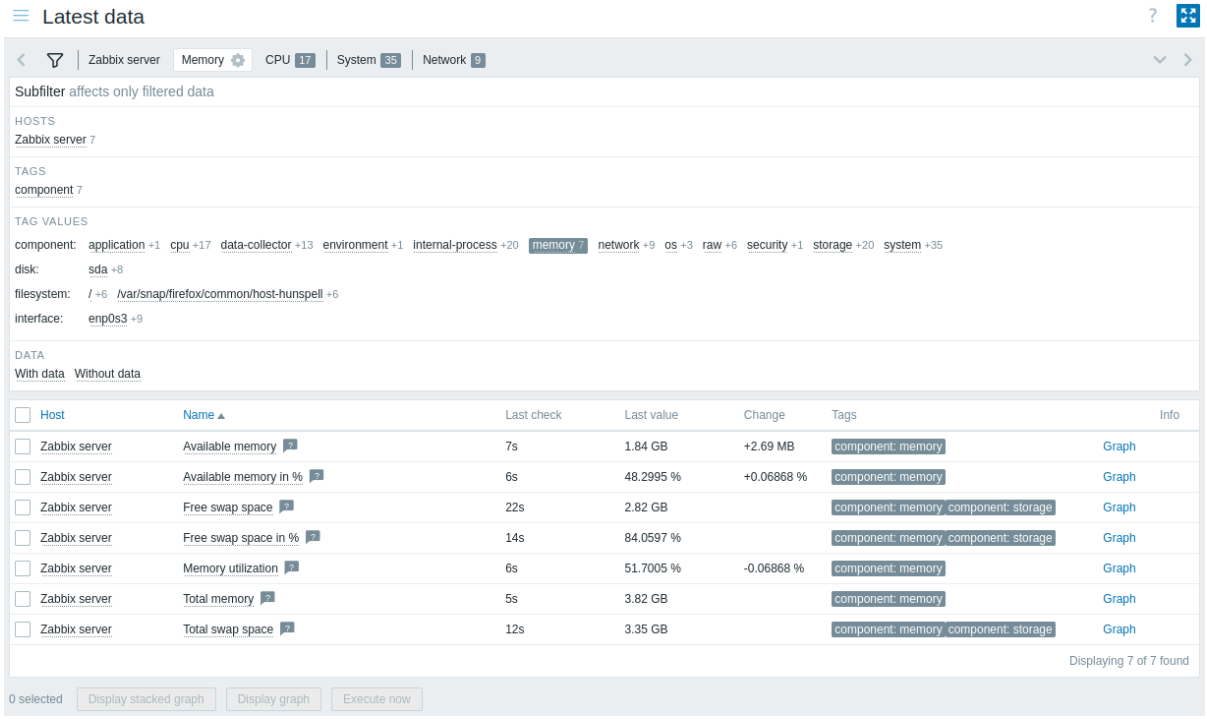
Omówienie

Sekcja *Monitoring* → *Latest data* wyświetla najnowsze wartości zebrane przez pozycje.

Ta sekcja zawiera następujące elementy:

- Filter
- Subfilter
- Lista pozycji

Note:
Subfilter i lista pozycji są wyświetlane tylko wtedy, gdy filtr jest ustawiony i istnieją wyniki do wyświetlenia.



Column	Description
Host	Nazwa hosta, do którego należy pozycja. Kliknięcie nazwy powoduje wyświetlenie menu kontekstowego hosta. Jeśli host jest w trybie konserwacji, po nazwie hosta wyświetlana jest pomarańczowa ikona klucza . Jeśli host jest wyłączony, jego nazwa jest wyświetlana na czerwono. Należy pamiętać, że dane wyłączonych hostów (w tym wykresy i listy wartości pozycji) są dostępne w sekcji *Latest data*.
Name	Nazwa pozycji. Kliknięcie nazwy powoduje wyświetlenie menu pozycji. Obok nazwy pozycji dla wszystkich pozycji, które mają opis, wyświetlana jest ikona znaku zapytania . Najedź kursorem na ikonę, aby wyświetlić podpowiedź z opisem pozycji.
Last check	Czas, jaki upłynął od ostatniego sprawdzenia pozycji.
Last value	Najnowsza wartość dla pozycji. Wartości są wyświetlane z zastosowaną konwersją jednostek i mapowaniem wartości. Najedź kursorem na wartość, aby wyświetlić surowe dane (przycięte do 8192 znaków). Wartości binarne są wyświetlane jako symbol zastępczy (*binary value*) zamiast rzeczywistej wartości. Domyślnie wyświetlane są tylko wartości odebrane w ciągu ostatnich 24 godzin. To ograniczenie poprawia czas początkowego ładowania dużych stron z najnowszymi danymi; aby je rozszerzyć, zmień wartość parametru *Max history display period* w *Administration → General → GUI*.
Change	Różnica między poprzednią wartością a najnowszą wartością. W przypadku pozycji z częstotliwością aktualizacji wynoszącą 1 dzień lub więcej, wartość zmiany nigdy nie będzie wyświetlana (przy ustawieniach domyślnych). W takim przypadku ostatnia wartość nie będzie w ogóle wyświetlana, jeśli została odebrana ponad 24 godziny temu.
Tags	Tagi powiązane z pozycją. Tagi na liście pozycji są klikalne. Kliknięcie tagu włącza go w subfilter, powodując wyświetlanie na liście pozycji tylko tych pozycji, które zawierają ten tag (oraz inne tagi wcześniej wybrane w subfilter). Należy pamiętać, że po przefiltrowaniu pozycji w ten sposób tagi na liście pozycji nie są już klikalne. Dalsze modyfikacje na podstawie tagów (na przykład usuwanie tagów lub określanie innych filtrów) muszą być wykonywane w subfilter.
Graph/History	Odnośnik do prostego wykresu/historii wartości pozycji.
Info	Dodatkowe informacje o pozycji. Jeśli pozycja zawiera błędy (na przykład stała się nieobsługiwana), wyświetlana jest ikona informacji . Najedź kursorem na ikonę, aby zobaczyć szczegóły.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, opisano na stronie Monitoring.

Działania masowe

Przyciski pod listą oferują działania masowe dla jednej lub kilku wybranych pozycji:

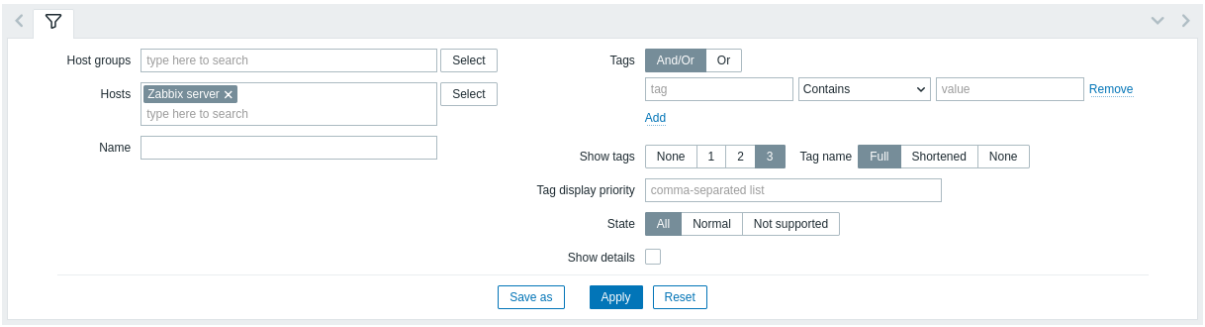
- *Wyświetl skumulowany wykres* - wyświetl skumulowany wykres ad hoc.
- *Wyświetl wykres* - wyświetl prosty wykres ad hoc.
- *Wykonaj teraz* - natychmiast wykonaj sprawdzenie nowych wartości pozycji. Obsługiwane tylko dla sprawdzeń **pasywnych** (zobacz więcej szczegółów). Ta opcja jest dostępna tylko dla hostów z dostępem do odczytu i zapisu. Dostęp do tej opcji dla hostów z uprawnieniami tylko do odczytu zależy od opcji roli użytkownika o nazwie *Wywołaj "Wykonaj teraz" na hostach tylko do odczytu*.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich pozycjach, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te pozycje, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Ikona filtra  znajduje się nad listą pozycji i podfiltrem. Kliknij ją, aby rozwinąć filtr.



Filtr umożliwia zawężenie listy pozycji według grupy hostów, hosta, nazwy pozycji, tagu, stanu i innych ustawień. Określenie nadrzędnej grupy hostów w filtrze powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. Szczegółowe informacje o filtrowaniu według tagów można znaleźć w *Monitoring* → *Problems*.

Opcja filtra *Show details* pozwala rozszerzyć informacje wyświetlane dla pozycji. Zaznacz ją, aby wyświetlić takie szczegóły, jak interwał odświeżania pozycji, ustawienia historii i trendów, typ pozycji oraz błędy pozycji (fine/unsupported).

Zapisywanie filtra

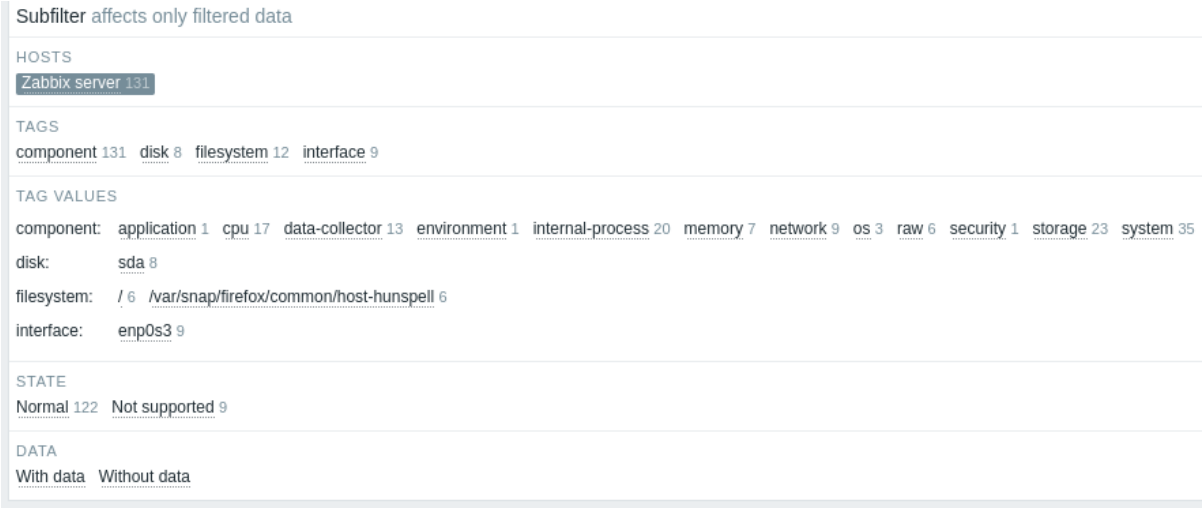
Ustawienia ulubionego filtra można zapisać jako karty, a następnie szybko uzyskać do nich dostęp, klikając odpowiednią kartę nad filtrem.

Więcej szczegółów można znaleźć w sekcji zapisywania filtrów.

Używanie podfiltra

Podfiltr umożliwia dalszą modyfikację filtrowania z filtra głównego.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych pozycji. Pozycje są powiązane wspólną jednostką — hostem, nazwą lub wartością tagu, stanem pozycji albo statusem danych. Po kliknięciu jednostki zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (bez konieczności kliknięcia *Apply* w filtrze głównym). Kliknięcie innej jednostki dodaje ją do filtrowanych wyników. Ponowne kliknięcie jednostki usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, których liczba jest ograniczona do 1000 rekordów. Jeśli jest 20 hostów, z których każdy ma 100 pozycji (czyli łącznie 2000 rekordów), w podfiltrze będzie widoczna tylko połowa hostów. Jeśli chcesz zobaczyć więcej rekordów w podfiltrze, musisz zwiększyć wartość parametru *Limit for search and filter results* (w *Administration -> General -> GUI*).

W przeciwieństwie do filtra głównego, podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierał aktualne informacje o dostępnych opcjach filtrowania i licznikach.

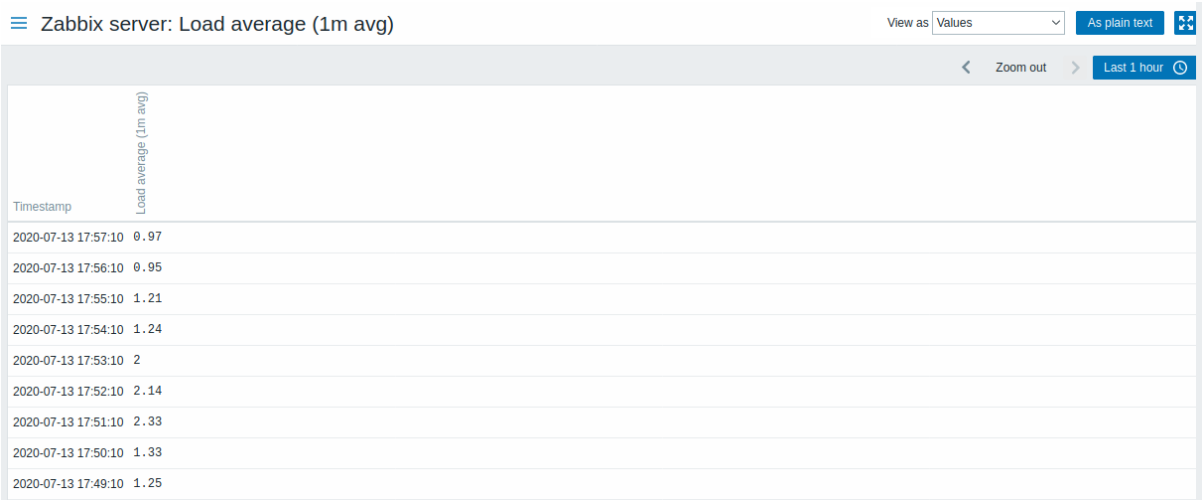
Dla każdej grupy jednostek (hosty, tagi, wartości tagów itd.) wyświetlanych jest maksymalnie 10 wierszy jednostek. Jeśli jednostek jest więcej, tę listę można rozwinąć, aby wyświetlić maksymalnie 1000 wpisów (wartość SUBFILTER_VALUES_PER_GROUP w frontend definitions), klikając ikonę trzech kropek ••• na końcu listy. W przypadku *Tag values* listę można rozwinąć, aby wyświetlić maksymalnie 200 nazw tagów wraz z odpowiadającymi im wartościami. Pamiętaj, że po pełnym rozwinięciu listy nie można jej zwinąć.

Liczba obok każdej klikalnej jednostki wskazuje liczbę pozycji w niej zgrupowanych (na podstawie wyników filtra głównego). Po kliknięciu jednostki liczby przy innych dostępnych jednostkach są wyświetlane ze znakiem plus, wskazującym, ile pozycji można dodać do bieżącego wyboru. Jednostki bez pozycji nie są wyświetlane, chyba że zostały wcześniej wybrane w podfiltrze.

Wykresy i historia

Kolumna *Graph/History* na liście pozycji oferuje następujące odnośniki:

- **History** — dla wszystkich tekstowych pozycji, prowadzi do list (*Values/500 latest values*) wyświetlających historię poprzednich wartości pozycji.
- **Graph** — dla wszystkich numerycznych pozycji, prowadzi do prostego wykresu. Zwróć uwagę, że po wyświetleniu wykresu lista rozwijana w prawym górnym rogu umożliwia również przełączenie do *Values/500 latest values*.



Wartości wyświetlane na tej liście są surowe, tzn. nie jest stosowane żadne przetwarzanie końcowe.

Note:
Łączna liczba wyświetlanych wartości jest określana przez parametr *Limit for search and filter results*, ustawiany w *Administration → General → GUI*.

4 Mapy

Przegląd

W sekcji *Monitoring* → *Mapy* można konfigurować, zarządzać i wyświetlać mapy sieci.

Po otwarciu tej sekcji zobaczysz albo ostatnią mapę, do której uzyskano dostęp, albo listę wszystkich map, do których masz dostęp.

Wszystkie mapy mogą być publiczne lub prywatne. Mapy publiczne są dostępne dla wszystkich użytkowników, natomiast mapy prywatne są dostępne tylko dla ich właściciela oraz użytkowników, którym mapa została udostępniona.

Lista map



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa mapy. Kliknij nazwę, aby wyświetlić mapę.
Szerokość	Wyświetlana jest szerokość mapy.
Wysokość	Wyświetlana jest wysokość mapy.
Działania	Dostępne są dwa działania: **Właściwości** - ustaw ogólne właściwości mapy **Edytuj** - przejdź do siatki w celu dodawania elementów mapy

Aby skonfigurować nową mapę, kliknij przycisk *Utwórz mapę* w prawym górnym rogu. Aby zaimportować mapę z pliku YAML, XML lub JSON, kliknij przycisk *Importuj* w prawym górnym rogu. Użytkownik importujący mapę zostanie ustawiony jako jej właściciel.

Dwa przyciski pod listą oferują opcje edycji masowej:

- *Eksportuj* - eksportuj mapy do pliku YAML, XML lub JSON
- *Usuń* - usuń mapy

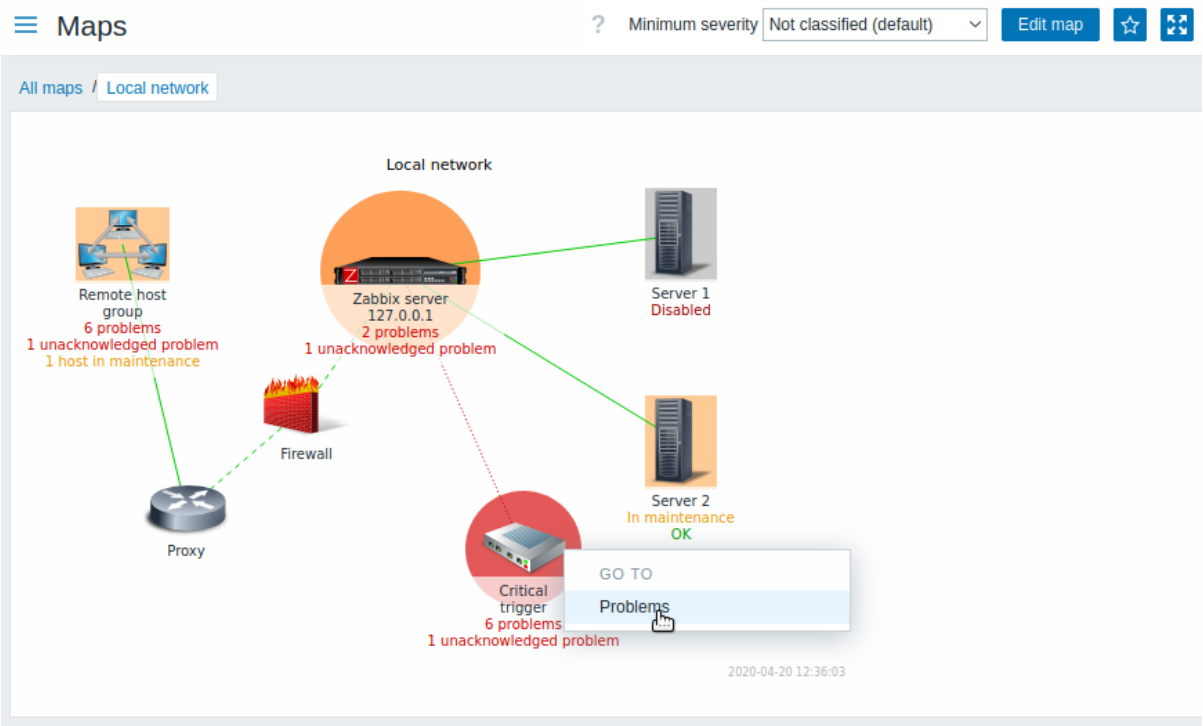
Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich mapach, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko mapy, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Wyświetlanie map

Aby wyświetlić mapę, kliknij jej nazwę na liście wszystkich map.



Możesz użyć listy rozwijanej na pasku tytułu mapy, aby wybrać najniższy poziom ważności problemów wyzwalaczy, które mają być wyświetlane. Poziom ważności oznaczony jako *domyślny* to poziom ustawiony w konfiguracji mapy. Jeśli mapa zawiera podmapę, przejście do podmapy zachowa poziom ważności mapy wyższego poziomu (z wyjątkiem sytuacji, gdy jest to *Niesklasyfikowane*; w takim przypadku nie zostanie on przekazany do podmapy).

Podświetlanie ikon

Jeśli element mapy ma status problemu, jest podświetlany okrągłym kółkiem. Kolor wypełnienia kółka odpowiada kolorowi ważności problemu. Tylko problemy o ważności równej lub wyższej od wybranego poziomu ważności będą wyświetlane dla elementu. Jeśli wszystkie problemy zostały potwierdzone, wokół kółka wyświetlana jest gruba zielona ramka.

Dodatkowo:

- host w maintenance jest podświetlany pomarańczowym, wypełnionym kwadratem. Należy pamiętać, że podświetlanie maintenance ma priorytet nad podświetlaniem według ważności problemu, jeśli elementem mapy jest host.
- wyłączony host (niemonitorowany) jest podświetlany szarym, wypełnionym kwadratem.

Podświetlanie jest wyświetlane, jeśli pole wyboru *Podświetlanie ikon* jest zaznaczone w konfiguracji mapy.

Oznaczenia ostatnich zmian

Czerwone trójkąty skierowane do wewnątrz wokół elementu wskazują na niedawną zmianę stanu wyzwalacza — taką, która nastąpiła w ciągu ostatnich 30 minut. Trójkąty te są wyświetlane, jeśli zaznaczono pole wyboru *Oznacz elementy przy zmianie stanu wyzwalacza* w konfiguracji mapy.

Linki

Kliknięcie elementu mapy otwiera menu z dostępnymi linkami. Kliknięcie nazwy hosta powoduje wyświetlenie menu hosta.

Przyciski

Przyciski po prawej stronie oferują następujące opcje:

Edit map	Przejdź do edycji zawartości mapy.
☆	Dodaj mapę do widżetu ulubionych w Dashboardach.
★	Mapa znajduje się w widżecie ulubionych w Dashboardach. Kliknij, aby usunąć mapę z widżetu ulubionych.

Przyciski trybu widoku wspólne dla wszystkich sekcji są opisane na stronie Monitoring.

Czytelne podsumowanie na mapach

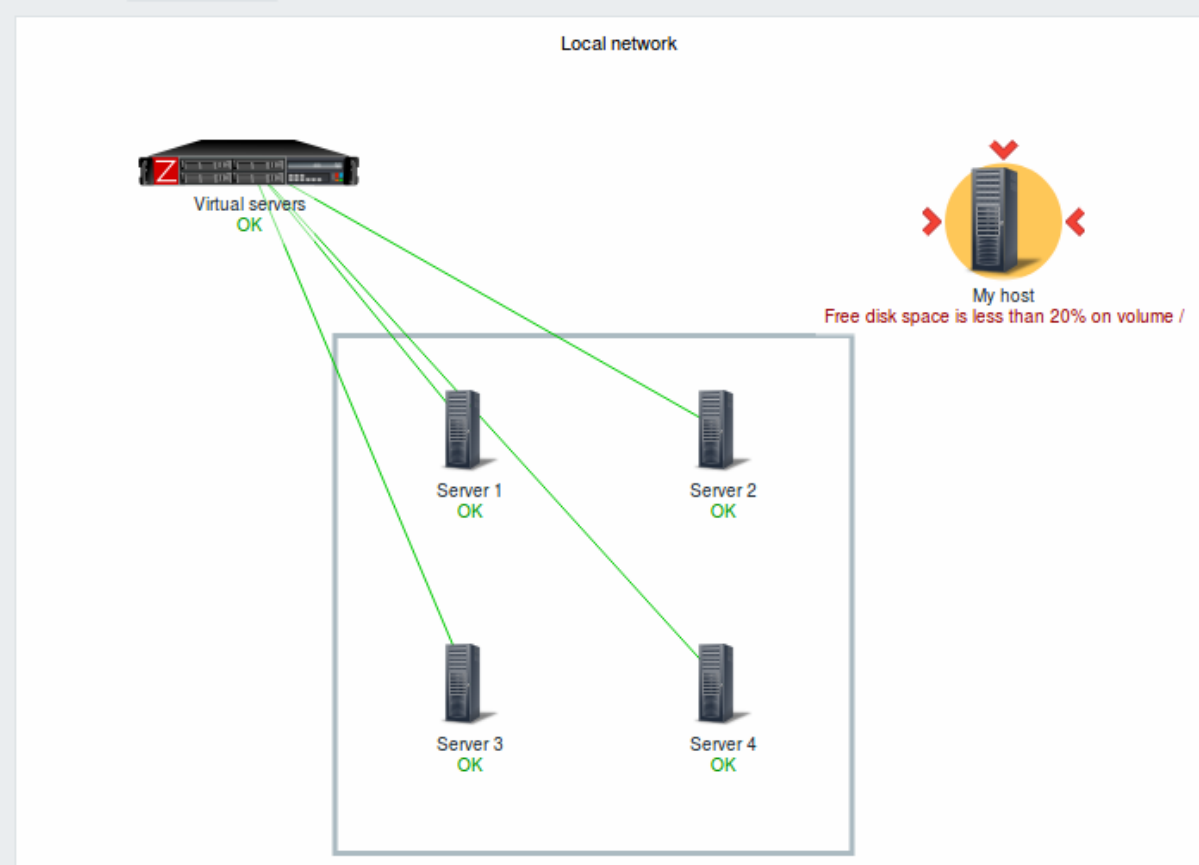
Dostępna jest ukryta właściwość „aria-label”, która umożliwia odczytywanie informacji o mapie za pomocą czytnika ekranu. Dostępny jest zarówno ogólny opis mapy, jak i opis poszczególnych elementów, w następującym formacie:

- dla opisu mapy: `<Nazwa mapy>, <* z * elementów w stanie problemu>, <* problemów łącznie>.`
- dla opisu jednego elementu z jednym problemem: `<Typ elementu>, Status <Status elementu>, <Nazwa elementu>, <Opis problemu>.`
- dla opisu jednego elementu z wieloma problemami: `<Typ elementu>, Status <Status elementu>, <Nazwa elementu>, <* problemów>.`
- dla opisu jednego elementu bez problemów: `<Typ elementu>, Status <Status elementu>, <Nazwa elementu>.`

Na przykład dostępny jest następujący opis:

```
'Sieć lokalna, 1 z 6 elementów w stanie problemu, 1 problem łącznie. Host, Status problem, Mój host, Wolne
```

dla następującej mapy:



Odwoływanie się do mapy sieciowej

Do map sieciowych można odwoływać się za pomocą parametrów GET `sysmapid` i `mapname`. Na przykład,

```
http://zabbix/zabbix/zabbix.php?action=map.view&mapname=Local%20network
```

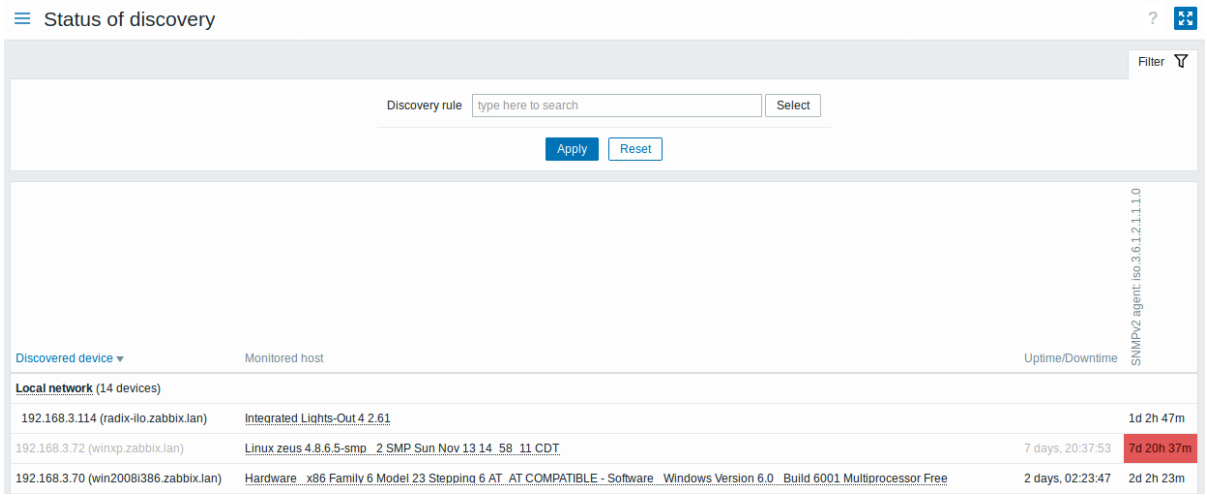
otworzy mapę o tej nazwie (Local network).

Jeśli określono zarówno `sysmapid` (ID mapy), jak i `mapname` (nazwę mapy), `mapname` ma wyższy priorytet.

5 Wykrywanie

Przegląd

W sekcji *Monitoring* → *Discovery* wyświetlane są wyniki odkrywania sieci. Odkryte urządzenia są sortowane według reguły odkrywania.



Wyświetlane dane:

Kolumna	Opis
Odkryte urządzenie	Odkryte urządzenia są wyświetlane na liście, pogrupowane według reguły odkrywania. Kliknięcie reguły odkrywania otwiera menu reguły zawierające odnośnik do formularza konfiguracji reguły odkrywania.
Monitorowany host	Jeśli urządzenie jest już monitorowane, w tej kolumnie zostanie wyświetlona nazwa hosta. Kliknięcie nazwy hosta otwiera menu hosta.
Czas działania/Czas niedostępności	W tej kolumnie wyświetlany jest czas, przez jaki urządzenie pozostaje wykryte, lub czas od jego utraty po wcześniejszym odkryciu.
Sprawdzenie odkrywania	Wyświetlany jest stan poszczególnej usługi (sprawdzenia odkrywania) dla każdego odkrytego urządzenia. Czerwona komórka oznacza, że usługa jest niedostępna. Czas działania lub czas niedostępności usługi jest podany w komórce. Ta kolumna jest wyświetlana tylko wtedy, gdy usługa została znaleziona na co najmniej jednym odkrytym urządzeniu.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Jeśli w filtrze nic nie wybrano, wyświetlane są wszystkie włączone reguły wykrywania. Aby wybrać do wyświetlenia konkretną regułę wykrywania, zacznij wpisywać jej nazwę w filtrze. Wszystkie pasujące włączone reguły wykrywania zostaną wyświetlone na liście do wyboru. Można wybrać więcej niż jedną regułę wykrywania.

3 Usługi

Przegląd

Menu Usługi służy do funkcji monitorowania usług w Zabbix.

1 Usługi

Przegląd

W tej sekcji można zobaczyć ogólny status wszystkich usług, które zostały skonfigurowane w Zabbix, na podstawie Twojej infrastruktury.

Usługa może być hierarchią składającą się z kilku poziomów innych usług, nazywanych usługami ”dziecięcymi”, które wpływają na ogólny status usługi (zobacz także przegląd funkcjonalności monitorowania usług).

Główne kategorie statusu usługi to *OK* lub *Problem*, gdzie status *Problem* jest wyrażany przez odpowiednią nazwę i kolor ważności problemu.

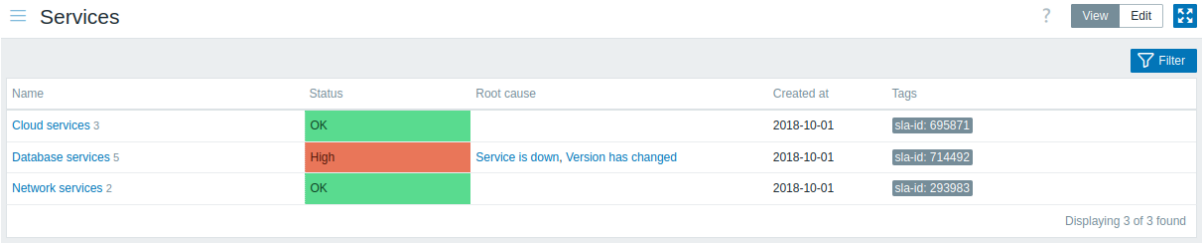
Podczas gdy tryb widoku umożliwia monitorowanie usług wraz z ich statusem i innymi szczegółami, w tej sekcji można również skonfigurować hierarchię usług (dodawać/edytować usługi, usługi dziecięce), przełączając się do trybu edycji.

Aby przełączyć się z trybu widoku do trybu edycji (i z powrotem), kliknij odpowiedni przycisk w prawym górnym rogu:

- View - wyświetl usługi
- Edit - dodawaj/edytuj usługi oraz usługi dziecięce

Należy pamiętać, że dostęp do edycji zależy od ustawień roli użytkownika.

Wyświetlanie usług



Wyświetlana jest lista istniejących usług.

Wyświetlane dane:

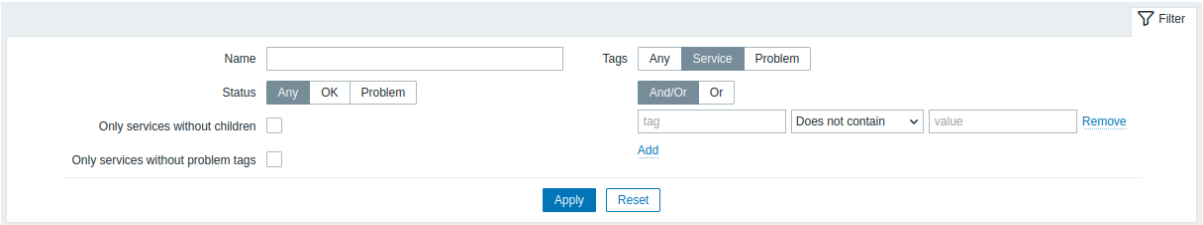
Parameter	Description
Name	Nazwa usługi. Nazwa usługi jest linkiem do szczegółów usługi. Liczba po nazwie wskazuje, ile usług podrzędnych ma dana usługa.
Status	Status usługi: **OK** - brak problemów **<problem color and severity>** - wskazuje problem i jego ważność. W przypadku wielu problemów wyświetlany jest kolor i ważność najbardziej krytycznego problemu.
Root cause	Wyświetlane są podstawowe problemy, które bezpośrednio lub pośrednio wpływają na status usługi. Te same problemy są wyświetlane zgodnie z wynikiem zwracanym przez makro {SERVICE.ROOTCAUSE}. Kliknij nazwę problemu, aby zobaczyć więcej szczegółów w *Monitoring* → *Problemy*. Problemy, które nie wpływają na status usługi, nie są uwzględniane na liście.
Created at	Wyświetlany jest czas utworzenia usługi.
Tags	Wyświetlane są tagi usługi. Tagi są używane do identyfikacji usługi w akcjach usługi oraz w SLA.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko usługi, które Cię interesują.



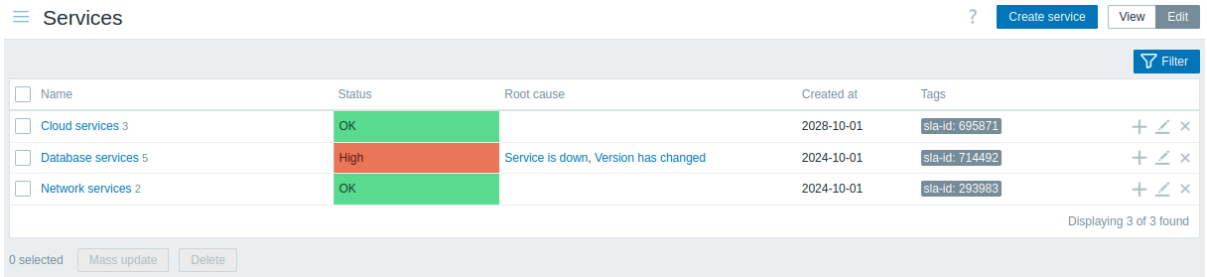
Parameter	Description
Name	Filtruj według nazwy usługi.
Status	Filtruj według statusu usługi.
Only services without children	Zaznacz pole wyboru, aby wyświetlić tylko usługi bez usług podrzędnych. Ten parametr jest dostępny tylko w trybie edycji usług.

Parameter	Description
Only services without problem tags *Tags*	Zaznacz pole wyboru, aby wyświetlić tylko usługi bez tagów problemów. Ten parametr jest dostępny tylko w trybie edycji usług. Filtruj według nazwy i wartości tagu usługi albo nazwy i wartości tagu problemu usługi (w trybie edycji usług). Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu zostaną zgrupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek.

Edycja usług

Kliknij przycisk *Edytuj*, aby przejść do trybu edycji. W trybie edycji lista jest uzupełniona o pola wyboru przed wpisami, a także o następujące dodatkowe opcje:

-  - dodaj usługę podrzędną do tej usługi
-  - edytuj tę usługę
-  - usuń tę usługę

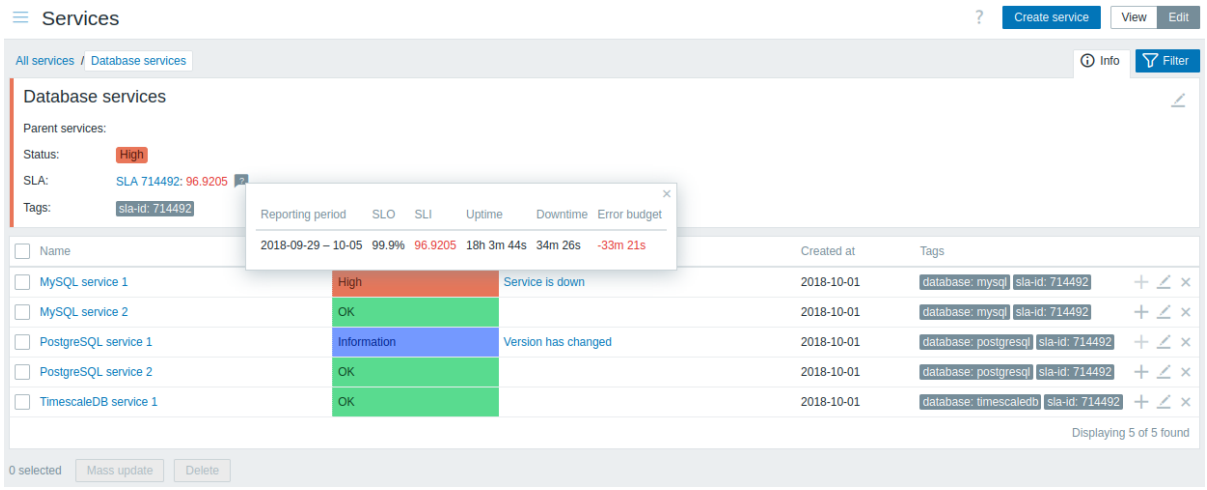


Aby skonfigurować nową usługę, kliknij przycisk *Utwórz usługę* w prawym górnym rogu.

Szczegóły usługi

Aby uzyskać dostęp do szczegółów usługi, kliknij nazwę usługi. Aby wrócić do listy wszystkich usług, kliknij *Wszystkie usługi*.

Szczegóły usługi obejmują pole informacyjne oraz listę usług podrzędnych.



Aby uzyskać dostęp do pola informacyjnego, kliknij kartę *Info*. Pole informacyjne zawiera następujące wpisy:

- Nazwy usług nadrzędnych (jeśli istnieją)
- Bieżący status tej usługi
- Bieżące SLA tej usługi w formacie `nazwa SLA:wskaźnik poziomu usługi`. „Nazwa SLA" jest również łączem do raportu SLA dla tej usługi. Jeśli umieścisz kursor myszy nad polem informacyjnym obok wskaźnika poziomu usługi (SLI), zostanie wyświetlona wyskakująca lista informacyjna ze szczegółami SLI. Wskaźnik poziomu usługi wyświetla bieżący poziom usługi w procentach.
- Tagi usługi

Pole informacyjne zawiera również łącze do konfiguracji usługi.

Aby użyć filtra dla usług podrzędnych, kliknij kartę *Filtr*.

W trybie edycji lista usług podrzędnych jest uzupełniona o dodatkowe opcje edycji:

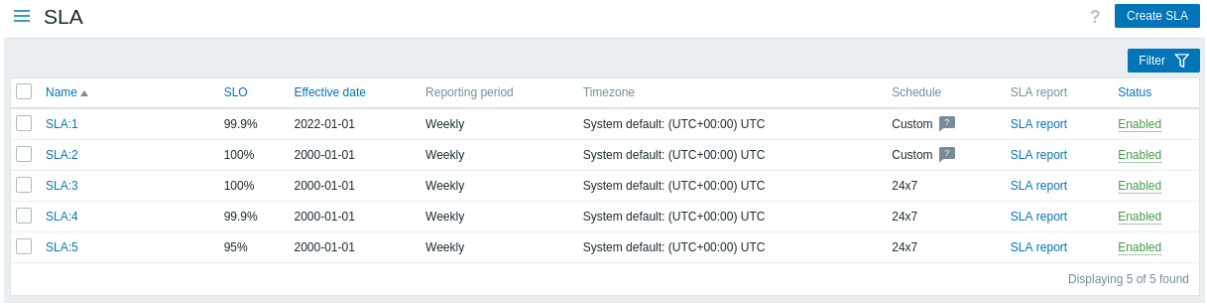
-  - dodaj usługę podrzędną do tej usługi
-  - edytuj tę usługę
-  - usuń tę usługę

2 SLA

Przegląd

Ta sekcja umożliwia przeglądanie i konfigurowanie SLA.

SLA



Wyświetlana jest lista skonfigurowanych SLA. *Uwaga*: wyświetlane będą tylko SLA powiązane z usługami dostępnymi dla użytkownika (tylko do odczytu, chyba że dla roli użytkownika włączono opcję *Manage SLA*).

Wyświetlane dane:

Parametr	Opis
Name	Wyświetlana jest nazwa SLA. Nazwa jest linkiem do konfiguracji SLA.
SLO	Wyświetlany jest cel poziomu usługi (SLO).
Effective date	Wyświetlana jest data rozpoczęcia obliczania SLA.

Parametr	Opis
Reporting period	Wyświetlany jest okres używany w raporcie SLA — *dzienny*, *tygodniowy*, *miesięczny*, *kwartalny* lub *roczny*.
Time zone	Wyświetlana jest strefa czasowa SLA.
Schedule	Wyświetlany jest harmonogram SLA — 24x7 lub niestandardowy.
SLA report	Kliknij link, aby zobaczyć raport SLA dla tego SLA.
Status	Wyświetlany jest status SLA — włączone lub wyłączone.

3 Raport SLA

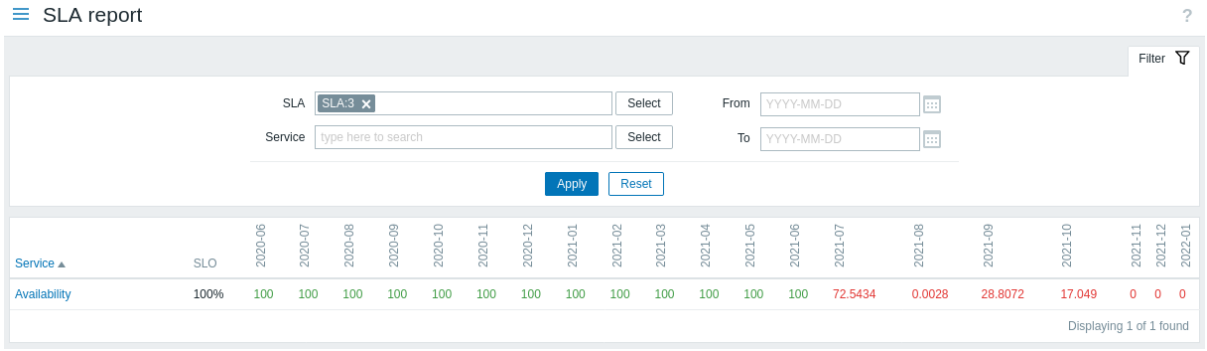
Przegląd

Ta sekcja umożliwia przeglądanie raportów SLA na podstawie kryteriów wybranych w filtrze.

Raporty SLA mogą być również wyświetlane jako widget pulpitu.

Raport

Filtr umożliwia wybór raportu na podstawie nazwy SLA, a także nazwy usługi. Możliwe jest również ograniczenie wyświetlanego okresu.

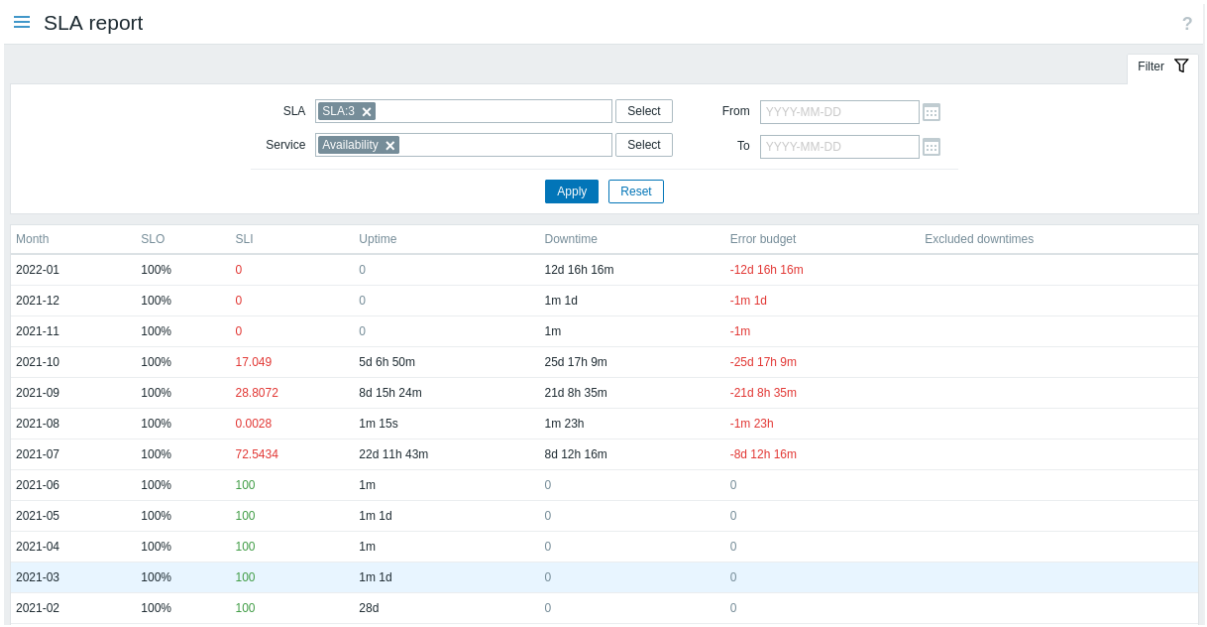


Każda kolumna (okres) wyświetla SLI dla tego okresu. Wartości SLI, które naruszają ustawiony SLO, są wyróżnione na czerwono.

W raporcie wyświetlanych jest 20 okresów. Maksymalnie można wyświetlić 100 okresów, jeśli określono zarówno datę *Od*, jak i datę *Do*.

Szczegóły raportu

Jeśli klikniesz nazwę usługi w raporcie, możesz przejść do innego raportu, który wyświetla bardziej szczegółowy widok.



Pamiętaj, że ujemny czas trwania problemu nie wpływa na obliczanie ani raportowanie SLA.

4 Inwentaryzacja

Przegląd

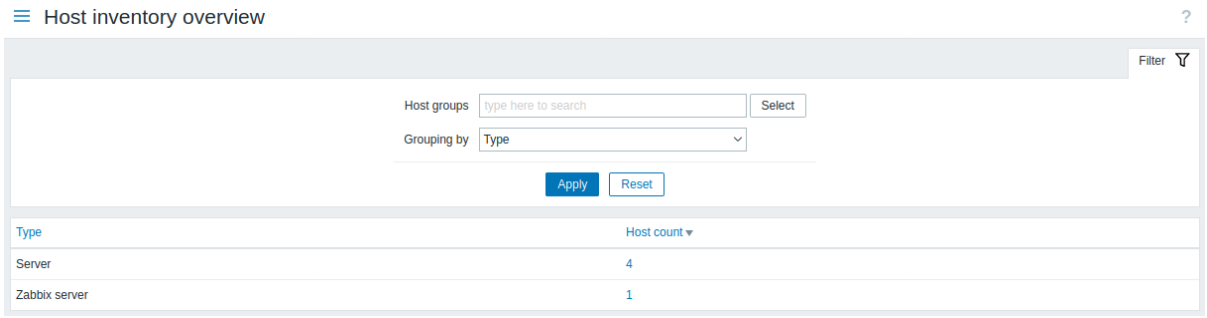
Menu Inwentaryzacja zawiera sekcje zapewniające przegląd danych inwentaryzacyjnych hostów według wybranego parametru, a także możliwość wyświetlania szczegółów inwentaryzacyjnych hosta.

1 Przegląd

Przegląd

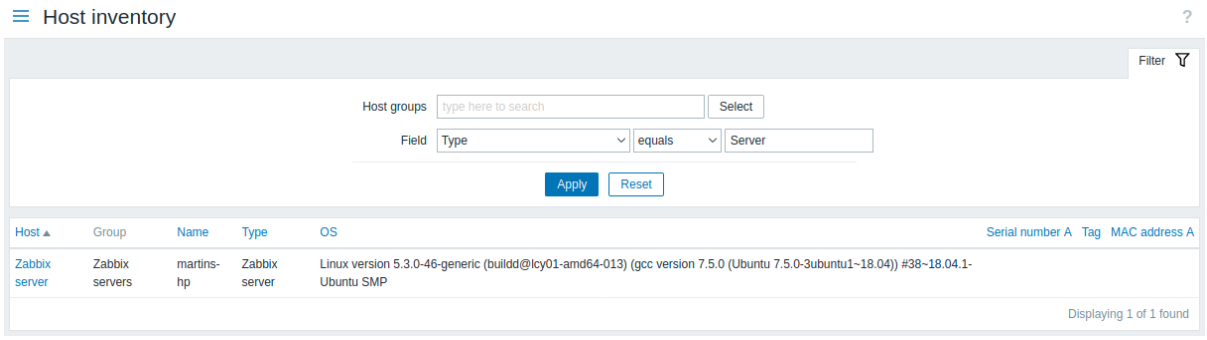
Sekcja *Inwentaryzacja* → *Przegląd* udostępnia sposoby uzyskania przeglądu danych inwentaryzacji hostów.

Aby wyświetlić przegląd, wybierz grupy hostów (lub żadną) oraz pole inwentaryzacji, według którego mają być wyświetlane dane. Zostanie wyświetlona liczba hostów odpowiadających każdej wartości wybranego pola.



Kompletność przeglądu zależy od tego, jak dużo informacji inwentaryzacyjnych jest utrzymywanych dla hostów.

Liczby w kolumnie *Liczba hostów* są odnośnikami; prowadzą do przefiltrowania tych hostów w tabeli *Inwentaryzacja hostów*.

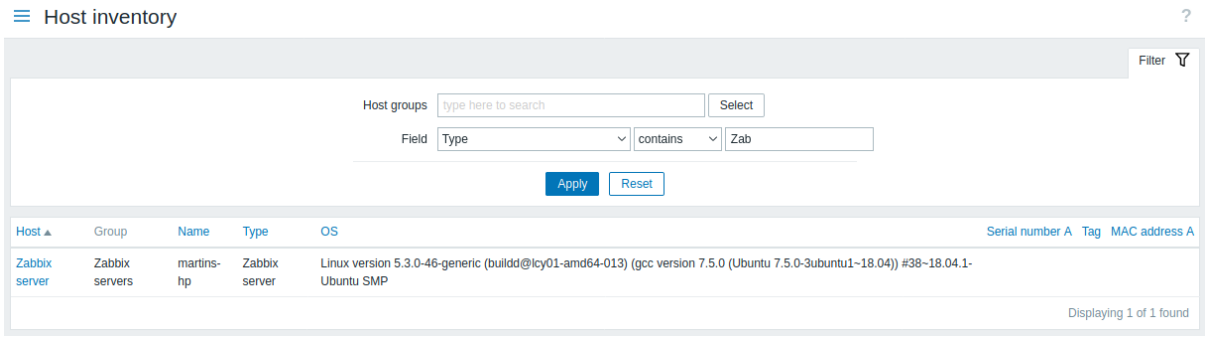


2 Hosty

Przegląd

W sekcji *Inwentaryzacja* → *Hosty* wyświetlane są dane inwentaryzacyjne hostów.

Możesz filtrować hosty według grup(y) hostów oraz według dowolnego pola inwentaryzacji, aby wyświetlić tylko te hosty, które Cię interesują.

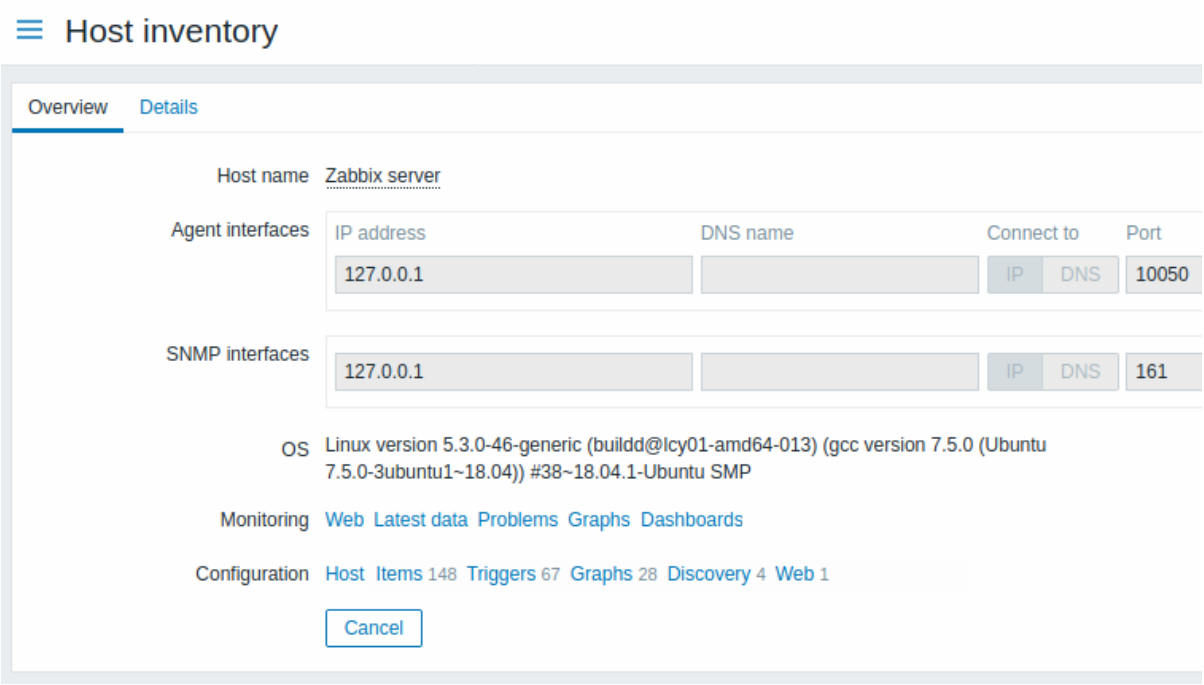


Aby wyświetlić pełną inwentaryzację wszystkich hostów, naciśnij przycisk „Reset”.

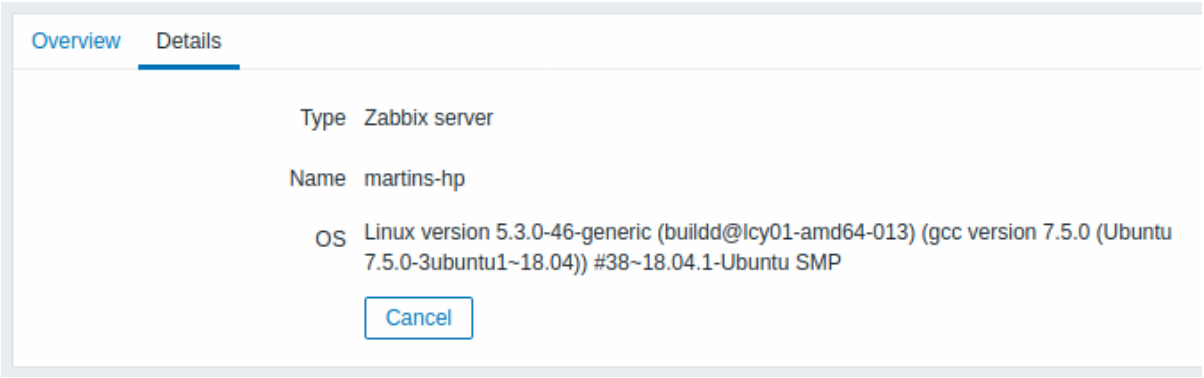
Chociaż w tabeli wyświetlane są tylko niektóre kluczowe pola inwentaryzacji, możesz również zobaczyć wszystkie dostępne informacje inwentaryzacyjne dla danego hosta. Aby to zrobić, kliknij nazwę hosta w pierwszej kolumnie.

Szczegóły inwentarza

Zakładka **Przegląd** zawiera ogólne informacje o hoście, najnowsze dane monitoringu oraz opcje konfiguracji hosta:



Zakładka **Szczegóły** zawiera wszystkie dostępne szczegóły inwentarza dla hosta:



Kompletność danych inwentarza zależy od tego, ile informacji inwentarzowych jest utrzymywanych dla hosta. Jeśli żadne informacje nie są utrzymywane, zakładka *Szczegóły* jest wyłączona.

5 Raporty

Przegląd

Menu Raporty zawiera kilka sekcji z różnorodnymi, predefiniowanymi i dostosowywanymi przez użytkownika raportami, które koncentrują się na wyświetlaniu przeglądu takich parametrów jak informacje o systemie, wyzwalacze i zebrane dane.

1 Informacje o systemie

Przegląd

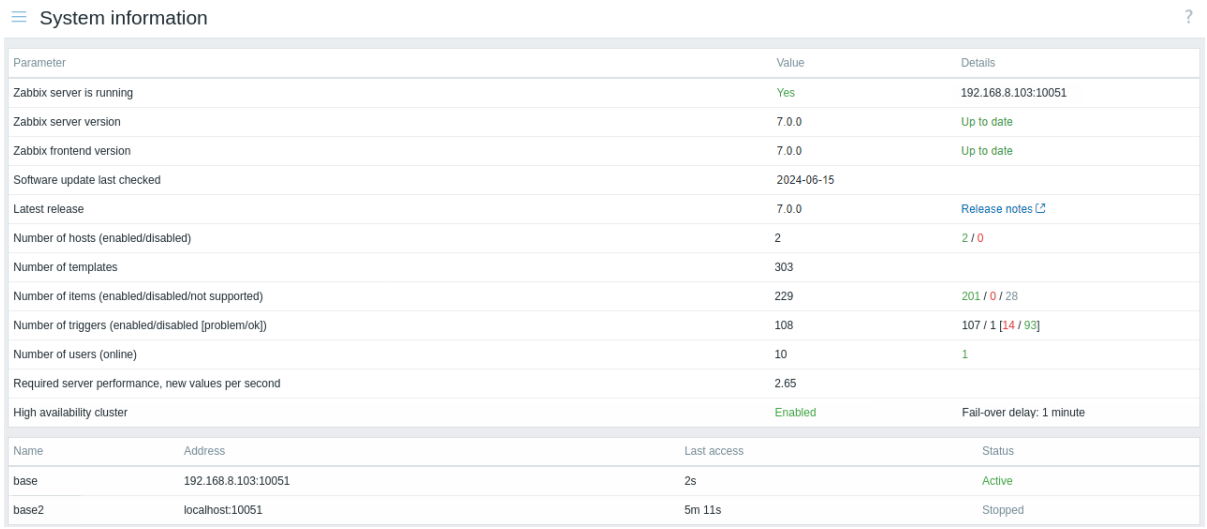
W *Raporty → Informacje o systemie* wyświetlane jest podsumowanie kluczowych danych serwera Zabbix i systemu. Dane systemowe są zbierane przy użyciu pozycji wewnętrznych.

Należy pamiętać, że w konfiguracji wysokiej dostępności możliwe jest przekierowanie źródła informacji o systemie (instancji serwera). Aby to zrobić, edytuj plik *zabbix.conf.php* — odkomentuj i ustaw `$ZBX_SERVER` lub jednocześnie `$ZBX_SERVER` i `$ZBX_SERVER_PORT` na serwer inny niż ten aktualnie pokazany jako aktywny. Pamiętaj, że przy ustawieniu tylko `$ZBX_SERVER` zostanie użyta domyślna wartość (10051) dla `$ZBX_SERVER_PORT`.

Po włączeniu konfiguracji wysokiej dostępności poniżej statystyk systemowych wyświetlany jest osobny blok ze szczegółami węzłów wysokiej dostępności. Ten blok jest widoczny tylko dla użytkowników Zabbix *Super Admin*.

Informacje o systemie są również dostępne jako widget pulpitu.

Statystyki systemu



Wyświetlane dane:

Parametr	Wartość	Szczegóły
Serwer Zabbix działa	Status serwera Zabbix: **Tak** - serwer działa **Nie** - serwer nie działa *Uwaga:* Aby wyświetlić pozostałe informacje, frontend WWW wymaga działającego serwera oraz co najmniej jednego uruchomionego procesu trapper na serwerze (parametr StartTrappers w pliku zabbix_server.conf > 0).	Lokalizacja i port serwera Zabbix.
Wersja serwera Zabbix	Wyświetlany jest numer bieżącej wersji serwera. *Uwaga:* Jest wyświetlany tylko wtedy, gdy serwer Zabbix działa.	Wyświetlany jest status wersji serwera: **Aktualna** - używana jest najnowsza wersja; **Dostępna nowa aktualizacja** - dostępna jest nowsza wersja; **Nieaktualna** - okres pełnego wsparcia dla tej wersji wygasł. Ta informacja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak danych o bieżącej wersji.
Wersja frontend Zabbix	Wyświetlany jest numer wersji frontend Zabbix.	Wyświetlany jest status wersji frontend Zabbix: **Aktualna** - używana jest najnowsza wersja; **Dostępna nowa aktualizacja** - dostępna jest nowsza wersja; **Nieaktualna** - okres pełnego wsparcia dla tej wersji wygasł. Ta informacja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak danych o bieżącej wersji.
Ostatnie sprawdzenie aktualizacji oprogramowania	Wyświetlana jest data ostatniego sprawdzenia aktualizacji oprogramowania Zabbix. Ta informacja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix.	

Parametr	Wartość	Szczegóły
Najnowsze wydanie	Wyświetlany jest numer nowszego wydania (jeśli jest dostępne) dla bieżącej wersji Zabbix. Ta informacja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak danych o bieżącej wersji.	Wyświetlany jest link do informacji o wydaniu najnowszego dostępnego wydania Zabbix.
Liczba hostów	Wyświetlana jest łączna liczba skonfigurowanych hostów.	Liczba monitorowanych/niemonitorowanych hostów.
Liczba szablonów	Wyświetlana jest łączna liczba szablonów.	
Liczba pozycji	Wyświetlana jest łączna liczba pozycji.	Liczba monitorowanych/wyłączonych/nieobsługiwanych pozycji na poziomie hosta. Pozycje na wyłączonych hostach są liczone jako wyłączone.
Liczba wyzwalaczy	Wyświetlana jest łączna liczba wyzwalaczy.	Liczba włączonych/wyłączonych wyzwalaczy na poziomie hosta; podział włączonych wyzwalaczy według stanów „Problem”/„OK”. Wyzwalacze wymienione w stanie „OK” obejmują wyzwalacze ze statusem „Unknown”. Wyzwalacze zależne od wyłączonych pozycji lub przypisane do wyłączonych hostów są liczone jako wyłączone.
Liczba użytkowników	Wyświetlana jest łączna liczba skonfigurowanych użytkowników.	Liczba użytkowników online.
Wymagana wydajność serwera, nowe wartości na sekundę	Wyświetlana jest oczekiwana liczba nowych wartości przetwarzanych przez serwer Zabbix na sekundę.	*Wymagana wydajność serwera* jest wartością szacunkową i może być użyteczna jako wskazówka. Aby uzyskać dokładne liczby przetwarzanych wartości, użyj wewnętrznej pozycji `zabbix[wcache,values,all]`. W obliczeniach uwzględniane są włączone pozycje z monitorowanych hostów. Pozycje logów są liczone jako jedna wartość na interwał aktualizacji pozycji. Uwzględniane są wartości z regularnych interwałów; wartości z interwałów elastycznych i harmonogramowanych nie są. Obliczenia nie są korygowane podczas okresu utrzymania „nodata”. Pozycje trapper nie są liczone.
Globalne skrypty na serwerze Zabbix	W tym polu zostanie wyświetlone **Disabled**, jeśli globalne skrypty są wyłączone na serwerze Zabbix przez ustawienie EnableGlobalScripts=0 w konfiguracji serwera.	
Klaster wysokiej dostępności	Status klastra wysokiej dostępności dla serwera Zabbix: **Disabled** - serwer samodzielny **Enabled** - istnieje co najmniej jeden węzeł wysokiej dostępności	Jeśli jest włączony, wyświetlane jest opóźnienie przełączenia awaryjnego.

Informacje o systemie wyświetlą również komunikat o błędzie w następujących warunkach:

- Używana baza danych nie ma wymaganego zestawu znaków ani reguł sortowania (UTF-8).
- Wersja bazy danych jest niższa lub wyższa niż obsługiwany zakres (dostępne tylko dla użytkowników z typem roli *Super admin role*).
- Housekeeping dla TimescaleDB jest niepoprawnie skonfigurowany (tabele historii lub trendów zawierają skompresowane chunki, ale opcje *Override item history period* lub *Override item trend period* są wyłączone).

Węzły wysokiej dostępności

Jeśli klaster wysokiej dostępności jest włączony, wyświetlany jest dodatkowy blok danych ze statusem każdego węzła wysokiej dostępności.



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	16 2021-10-20 17:58:47	Standby

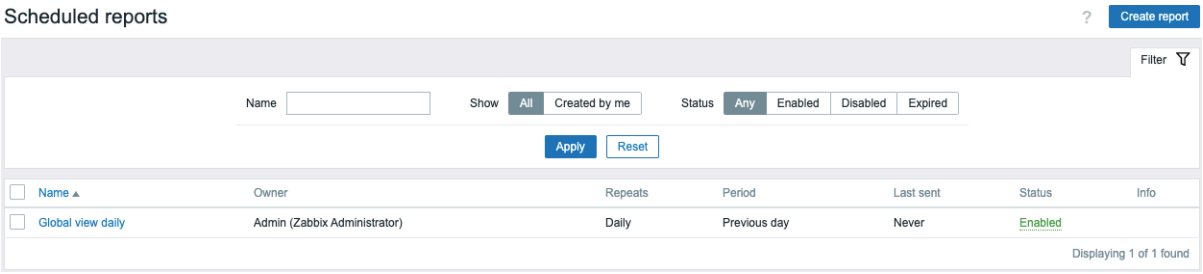
Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa węzła, zgodnie z definicją w konfiguracji serwera.
Adres	Adres IP węzła i port.
Ostatni dostęp	Czas ostatniego dostępu do węzła. Po najechaniu kursorem na komórkę wyświetlany jest znacznik czasu ostatniego dostępu w długim formacie.
Status	Status węzła. Wiersze tabeli są sortowane według tych statusów w następującej kolejności priorytetów: **Aktywny** - węzeł działa i pracuje **Niedostępny** - węzeł nie był widziany dłużej niż wynosi opóźnienie przełączenia awaryjnego (warto ustalić przyczynę) **Zatrzymany** - węzeł został zatrzymany lub nie mógł się uruchomić (warto go uruchomić lub usunąć) **W gotowości** - węzeł działa i oczekuje

2 Zaplanowane raporty

Przegląd

W *Raporty* → *Raporty cykliczne* użytkownicy z odpowiednimi uprawnieniami mogą skonfigurować cykliczne generowanie wersji PDF pulpitów, które będą wysyłane pocztą elektroniczną do określonych odbiorców.



Ekran otwierający wyświetla informacje o raportach cyklicznych, które można filtrować w celu ułatwienia nawigacji — zobacz sekcję Korzystanie z filtra poniżej.

Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa raportu. Kliknięcie jej otwiera formularz konfiguracji raportu.
Właściciel	Użytkownik, który utworzył raport.
Powtarzanie	Częstotliwość generowania raportu (codziennie/co tydzień/co miesiąc/co rok).
Okres	Przedział czasu, dla którego przygotowywany jest raport (poprzedni dzień, tydzień, miesiąc lub rok).
Ostatnio wysłano	Data i godzina wysłania ostatniego raportu.

Kolumna	Opis
Status	Bieżący status raportu (włączony/wyłączony/wygasły). Użytkownicy z odpowiednimi uprawnieniami mogą zmienić status, klikając go — z „Włączony” na „Wyłączony” (i z powrotem) oraz z „Wygasły” na „Wyłączony” (i z powrotem). Dla użytkowników bez wystarczających uprawnień status nie jest klikalny.
Informacje	Wyświetla ikony informacyjne: Czerwona ikona wskazuje, że generowanie raportu nie powiodło się; po najechaniu kursorem zostanie wyświetlona etykieta z informacją o błędzie. Żółta ikona wskazuje, że raport został wygenerowany, ale wysłanie do części (lub wszystkich) odbiorców nie powiodło się albo że raport wygasł; po najechaniu kursorem zostanie wyświetlona etykieta z dodatkowymi informacjami.

Używanie filtra

Możesz użyć filtra, aby zawęzić listę raportów. Dla lepszej wydajności wyszukiwania dane są przeszukiwane przy nierozwiniętych makrach.

Dostępne są następujące opcje filtrowania:

- *Nazwa* - dozwolone jest częściowe dopasowanie nazwy
- *Pokaż* - utworzone przez bieżącego użytkownika lub wszystkie raporty
- *Status* - wybierz spośród „Dowolny” (pokaż wszystkie raporty), „Włączony”, „Wyłączony” lub „Wygasły”

Filtr znajduje się poniżej nazwy sekcji *Raporty cykliczne*. Można go rozwinąć i zwinąć, klikając kartę *Filtr* w prawym górnym rogu.

Aktualizacja masowa

Czasami możesz chcieć usunąć lub zmienić status wielu raportów jednocześnie. Zamiast otwierać każdy raport osobno do edycji, możesz w tym celu użyć funkcji aktualizacji masowej.

Aby masowo zaktualizować raporty, wykonaj następujące czynności:

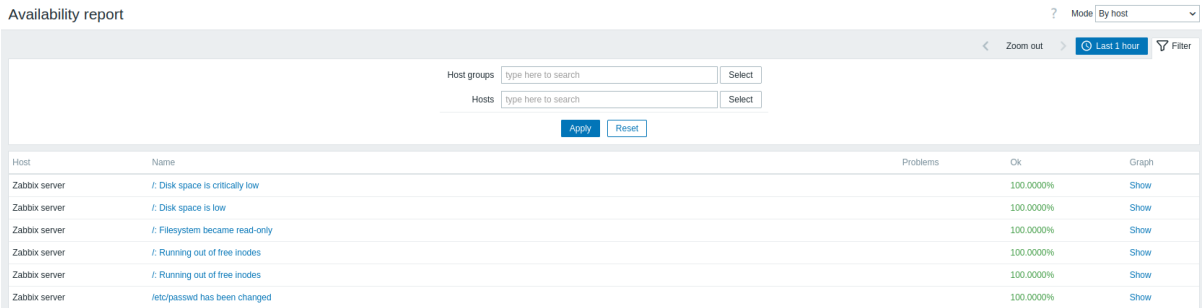
- Zaznacz pola wyboru raportów, które chcesz zaktualizować, na liście
- Kliknij odpowiedni przycisk pod listą, aby wprowadzić zmiany (*Włącz*, *Wyłącz* lub *Usuń*)

3 Raport dostępności

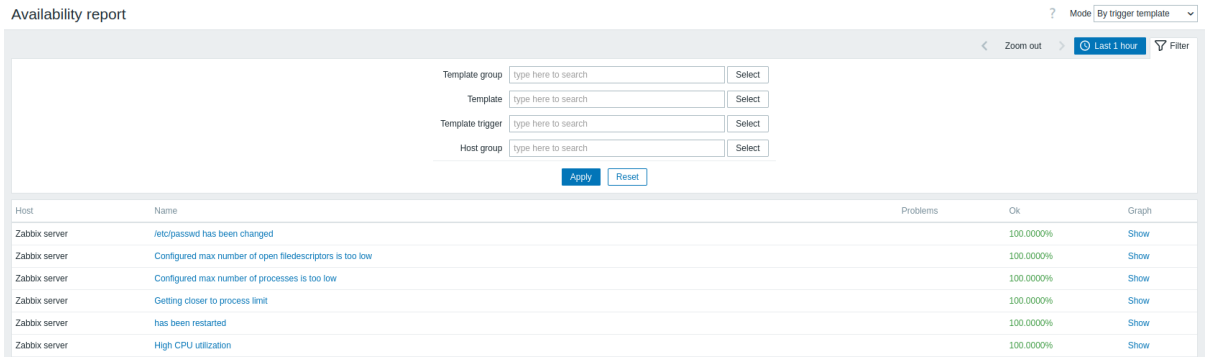
Przegląd

W *Raporty* > *Raport dostępności* możesz zobaczyć, przez jaki procent czasu każdy wyzwalacz znajdował się w stanie problem/OK.

Dla każdego stanu wyświetlany jest procent czasu, co ułatwia określenie dostępności różnych elementów w systemie.



Z listy rozwijanej w prawym górnym rogu możesz wybrać tryb wyboru — czy wyświetlać wyzwalacze według hostów, czy według wyzwalaczy należących do szablonu.



Nazwa wyzwalacza jest łączem do najnowszych zdarzeń tego wyzwalacza.

Używanie filtra

Filtr może pomóc zawęzić liczbę wyświetlanych hostów i/lub wyzwalaczy. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Filtr znajduje się poniżej nazwy sekcji *Raport dostępności*. Można go otworzyć i zwinąć, klikając kartę *Filtr* po prawej stronie.

Filtrowanie według szablonu wyzwalacza

W trybie *Według szablonu wyzwalacza* wyniki można filtrować według jednego lub kilku parametrów wymienionych poniżej.

Parametr	Opis
Grupa szablonów	Filtruj hosty według wyzwalaczy dziedziczonych z szablonów należących do wybranej grupy szablonów. Określenie nadrzędnej grupy szablonów powoduje niejawne wybranie wszystkich zagnieżdżonych grup szablonów.
Szablon	Filtruj hosty według wyzwalaczy dziedziczonych z wybranego szablonu, w tym z szablonów zagnieżdżonych. Jeśli zagnieżdżony szablon ma własne wyzwalacze, te wyzwalacze nie będą wyświetlane.
Wyzwalacz szablonu	Filtruj hosty według wybranego wyzwalacza. Inne wyzwalacze przefiltrowanych hostów nie będą wyświetlane.
Grupa hostów	Filtruj hosty należące do wybranej grupy hostów.

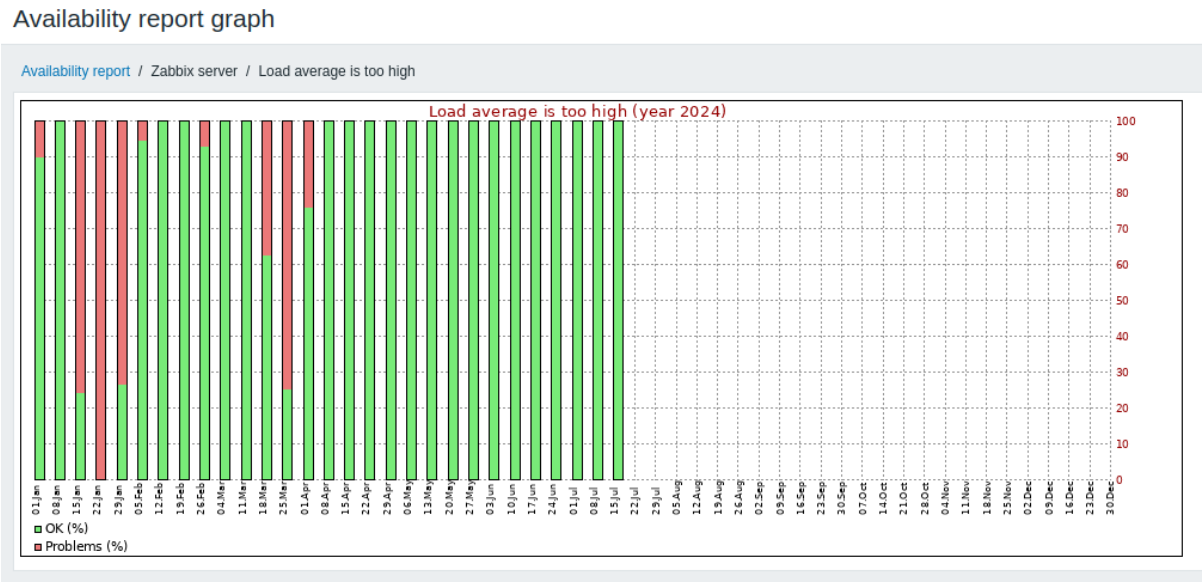
Filtrowanie według hosta

W trybie *Według hosta* wyniki mogą być filtrowane według hosta lub grupy hostów. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów.

Selektor okresu czasu

Selektor okresu czasu umożliwia wybór często używanych okresów czasu jednym kliknięciem. Selektor można rozwinąć i zwinąć, klikając kartę *Okres czasu* obok filtra.

Kliknięcie *Pokaż* w kolumnie *Wykres* wyświetla informacje o dostępności na wykresie słupkowym, gdzie każdy słupek reprezentuje miniony tydzień bieżącego roku.



Zielony kolor słupka oznacza czas OK, a czerwony — czas problemu.

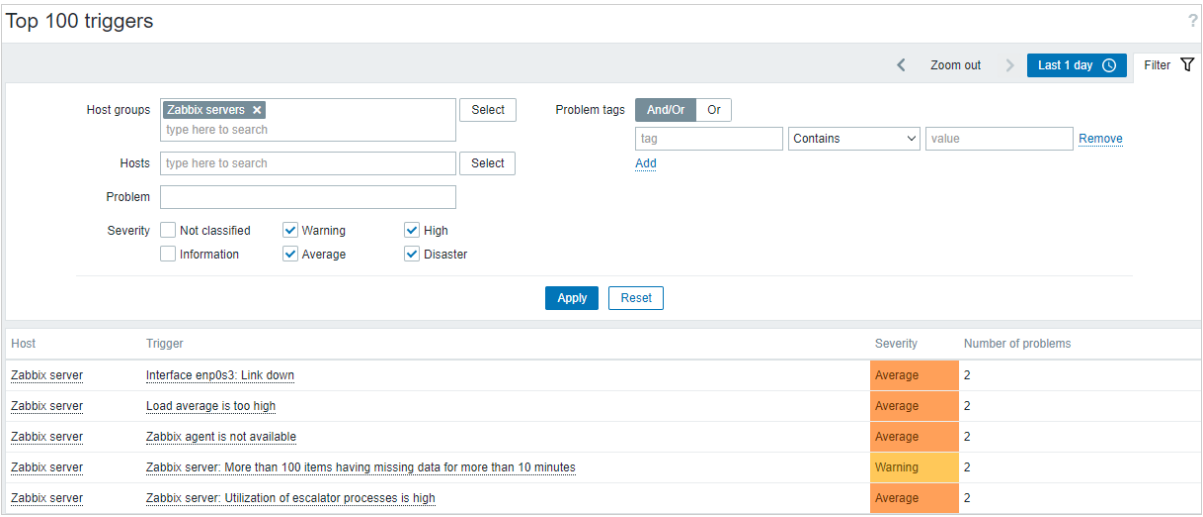
Wpływ okresów konserwacji

Konserwacja nie wyklucza automatycznie czasu z raportu dostępności. Tylko konserwacja skonfigurowana jako *Brak zbierania danych* zatrzymuje odpytywanie (więc nie są generowane żadne problemy), a zatem pozostawia procent dostępności bez zmian dla wyzwalaczy, których to dotyczy.

4 Top 100 wyzwalaczy

Przegląd

W *Reports* → *Top 100 triggers* można zobaczyć wyzwalacze z największą liczbą problemów wykrytych w wybranym okresie.



Zarówno wpisy w kolumnie host, jak i trigger są linkami, które oferują kilka przydatnych opcji:

- dla host - kliknięcie nazwy hosta wyświetla menu hosta
- dla trigger - kliknięcie nazwy wyzwalacza wyświetla linki do najnowszych zdarzeń, prostego wykresu dla każdej pozycji wyzwalacza oraz formularzy konfiguracji samego wyzwalacza i każdej pozycji wyzwalacza

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlać wyzwalacze według grupy hostów, host, nazwy problemu, tagów lub ważności wyzwalacza. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Filtr znajduje się poniżej nazwy sekcji *Top 100 triggers*. Można go rozwinąć i zwinąć, klikając kartę *Filter* po prawej stronie.

Selektor okresu czasu

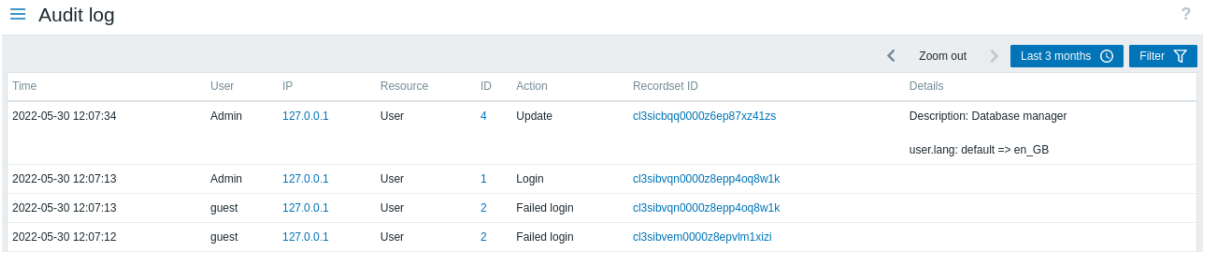
Selektor *Time period* umożliwia wybór często używanych okresów jednym kliknięciem myszy. Selektor *Time period* można rozwinąć i zwinąć, klikając kartę *Time period* obok filtra.

5 Dziennik audytu

Przegląd

W sekcji *Raporty* → *Dziennik audytu* można przeglądać wpisy dotyczące aktywności użytkowników i systemu.

Note:
Aby wpisy audytu były zbierane i wyświetlane, w sekcji *Administracja* → *Dziennik audytu* musi być zaznaczone pole wyboru *Włącz rejestrowanie audytu*. Jeśli to ustawienie nie jest włączone, historia aktywności nie będzie zapisywana w bazie danych i nie będzie wyświetlana w dzienniku audytu.



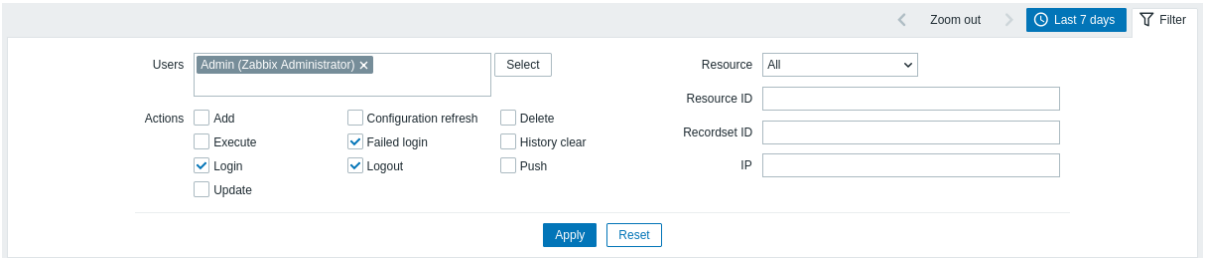
Dziennik audytu wyświetla następujące dane:

Kolumna	Opis
Czas	Znacznik czasu wpisu audytu.
Użytkownik	Użytkownik, który wykonał działanie.
IP	Adres IP, z którego zainicjowano działanie. Kliknięcie hiperłącza spowoduje filtrowanie wpisów dziennika audytu według tego adresu IP.
Zasób	Typ zasobu, którego dotyczy działanie (*Token API*, *Akcja*, *Uwierzytelnianie*, *Autorejestracja* itd.).
ID	ID zasobu, którego dotyczy działanie. Kliknięcie hiperłącza spowoduje filtrowanie wpisów dziennika audytu według ID tego zasobu.
Działanie	Typ działania (*Dodanie*, *Odświeżenie konfiguracji*, *Usunięcie*, *Wykonanie*, *Nieudane logowanie*, *Wyczyszczenie historii*, *Logowanie*, *Wylogowanie*, *Push*, *Aktualizacja*).
ID zestawu rekordów	Wspólne ID dla wszystkich wpisów dziennika audytu utworzonych w wyniku tej samej operacji. Na przykład podczas podłączania szablonu do hosta tworzony jest osobny wpis dziennika audytu dla każdej dziedziczonej encji szablonu (pozycja, wyzwalacz itd.) — wszystkie te wpisy będą miały to samo *ID zestawu rekordów*. Kliknięcie hiperłącza spowoduje filtrowanie wpisów dziennika audytu według tego *ID zestawu rekordów*.
Szczegóły	Opis zasobu oraz szczegółowe informacje o wykonanym działaniu. Jeśli wpis zawiera więcej niż dwa wiersze, zostanie wyświetlony dodatkowy odnośnik *Szczegóły*. Kliknij ten odnośnik, aby wyświetlić pełną listę zmian.

Note:
Gdy pozycja trapper lub pozycja HTTP agent (z włączonym przechwytywaniem) otrzyma dane, wpis w dzienniku audytu zostanie dodany tylko wtedy, gdy dane zostały wysłane przy użyciu metody API `history.push`, a nie narzędzia Zabbix sender.

Korzystanie z filtra

Filtr znajduje się poniżej paska *Dziennik audytu*. Można go otworzyć i zwinąć, klikając kartę *Filtr* w prawym górnym rogu.



Za pomocą filtra można zawęzić rekordy według użytkownika, zasobu, którego dotyczy operacja, identyfikatora zasobu, wykonanej operacji (*ID zestawu rekordów*) oraz adresu IP. W zależności od zasobu w filtrze można wybrać jedną lub więcej określonych akcji.

Aby zapewnić lepszą wydajność wyszukiwania, wszystkie dane są przeszukiwane z nierozwiniętymi makrami.

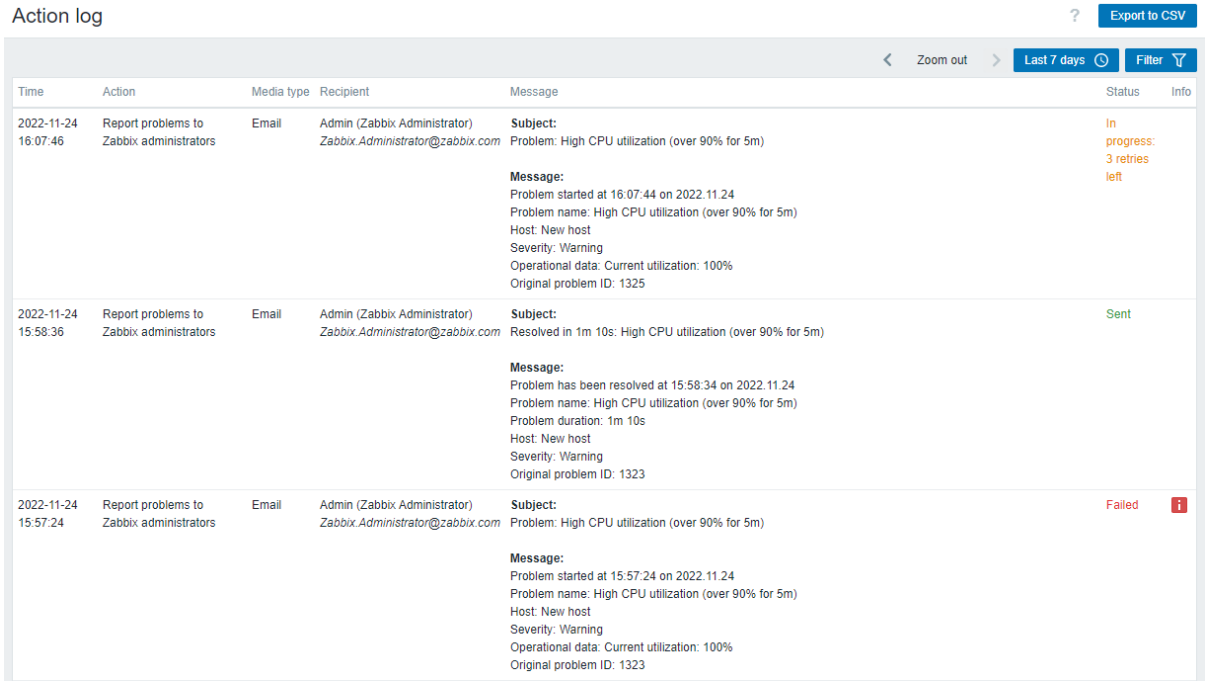
Selektor okresu czasu

Selektor okresu czasu umożliwia wybór często używanych okresów jednym kliknięciem myszy. Selektor *okresu czasu* można rozwinąć i zwinąć, klikając kartę *Okres czasu* w prawym górnym rogu.

6 Dziennik akcji

Przegląd

W sekcji *Reports* → *Action log* użytkownicy mogą wyświetlać szczegóły operacji (powiadomienia, zdalne polecenia) wykonanych w ramach akcji.



Wyświetlane dane:

Column	Description
Time	Znacznik czasu operacji.
Action	Nazwa akcji powodującej wykonanie operacji.
Media type	Typ medium (np. Email, Jira itp.) użyty do wysłania powiadomienia. W przypadku operacji, które wykonały zdalne polecenia, ta kolumna będzie pusta.
Recipient	Informacje o odbiorcy powiadomienia - nazwa użytkownika, imię i nazwisko (w nawiasach) oraz dodatkowe informacje zależnie od typu medium (email, nazwa użytkownika itp.). W przypadku operacji, które wykonały zdalne polecenia, ta kolumna będzie pusta.
Message	Treść wiadomości, zdalnego polecenia lub nazwa skryptu globalnego. Zdalne polecenie jest oddzielone od hosta docelowego symbolem dwukropka: `<host>:<command>`. Na przykład, jeśli zdalne polecenie zostało wykonane na serwerze Zabbix, informacja będzie miała następujący format: `Zabbix server:<command>`.
Status	Status operacji: *In progress* - operacja wysyłania powiadomienia jest w toku (wyświetlana jest również pozostała liczba prób, jakie serwer podejmie, aby wysłać powiadomienie) *Sent* - powiadomienie zostało wysłane *Executed* - zdalne polecenie zostało wykonane *Failed* - operacja nie została zakończona
Info	Informacje o błędzie (jeśli wystąpił) dotyczące wykonania operacji.

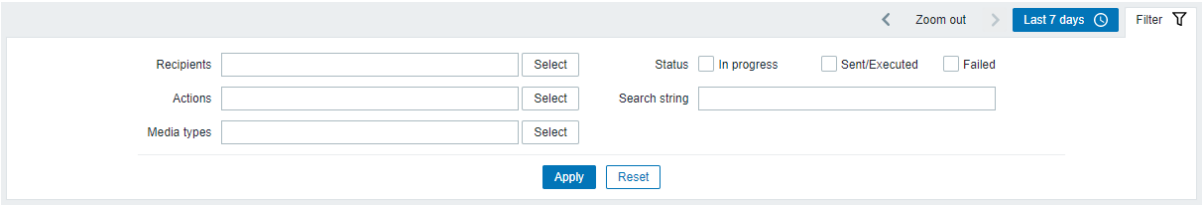
Przyciski

Przycisk w prawym górnym rogu strony oferuje następującą opcję:

Export to CSV	Eksportuj rekordy dziennika akcji z bieżącej strony do pliku CSV. Jeśli zastosowano filtr, wyeksportowane zostaną tylko przefiltrowane rekordy. W wyeksportowanym pliku CSV kolumny „Recipient" i „Message" są podzielone na kilka kolumn — „Recipient's Zabbix username", „Recipient's name", „Recipient's surname", „Recipient" oraz „Subject", „Message", „Command".

Korzystanie z filtra

Filtr znajduje się poniżej paska *Dziennik akcji*. Można go rozwinąć i zwinąć, klikając kartę *Filtr* w prawym górnym rogu strony.



Za pomocą filtra można zawęzić rekordy według odbiorców powiadomień, akcji, typów mediów, statusu lub treści wiadomości/zdalnego polecenia (*Szukany ciąg*). Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Selektor okresu czasu

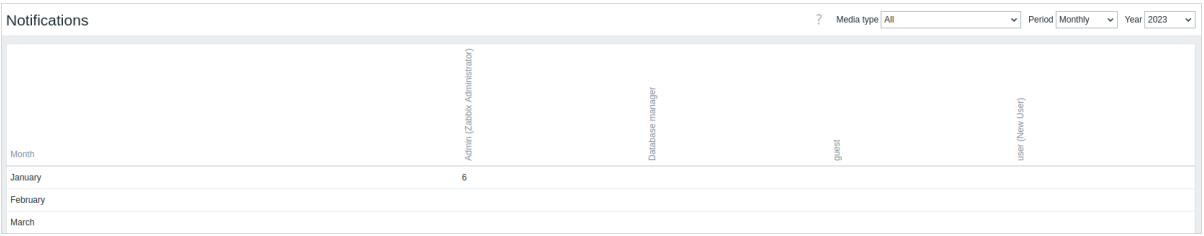
Selektor okresu czasu umożliwia wybór często używanych okresów jednym kliknięciem myszy. Selektor *okresu czasu* można rozwinąć i zwinąć, klikając kartę *Okres czasu* obok filtra.

7 Powiadomienia

Przegląd

W sekcji *Raporty* → *Powiadomienia* wyświetlany jest raport dotyczący liczby powiadomień wysłanych do każdego użytkownika.

Z list rozwijanych w prawym górnym rogu można wybrać typ nośnika (lub wszystkie), okres (dane dla każdego dnia/tygodnia/miesiąca/roku) oraz rok dla wysłanych powiadomień.



Każda kolumna wyświetla sumy dla jednego użytkownika systemu.

6 Zbieranie danych

Przegląd

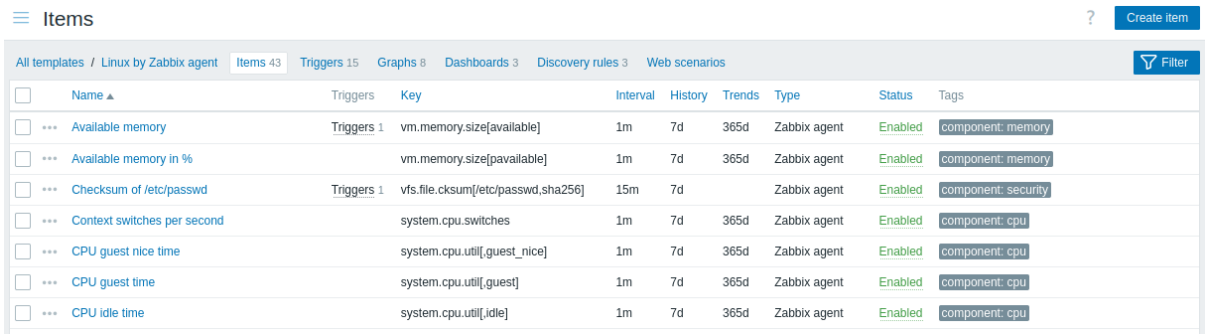
To menu zawiera sekcje związane z konfiguracją zbierania danych.

1 Pozycje

Przegląd

Lista pozycji dla szablonu jest dostępna w *Zbieranie danych* → *Szablony* po kliknięciu *Pozycje* dla odpowiedniego szablonu.

Zostanie wyświetlona lista istniejących pozycji.



Wyświetlane dane:

Kolumna	Opis
Menu pozycji	Kliknij ikonę z trzema kropkami, aby otworzyć menu pozycji.
Szablon	Szablon, do którego należy pozycja. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa pozycji jest wyświetlana jako niebieski link do szczegółów pozycji. Kliknięcie linku z nazwą pozycji otwiera formularz konfiguracji pozycji. Jeśli pozycja jest dziedziczona z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą pozycji jako szary link. Kliknięcie linku szablonu otwiera listę pozycji na poziomie tego szablonu.
Wyzwalacze	Po najechaniu kursorem na Wyzwalacze zostanie wyświetlone okno informacyjne pokazujące wyzwalacze powiązane z pozycją. Liczba wyzwalaczy jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania.
Historia	Wyświetlana jest liczba dni, przez które będzie przechowywana historia danych pozycji.
Trendy	Wyświetlana jest liczba dni, przez które będzie przechowywana historia trendów pozycji.
Typ	Wyświetlany jest typ pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itd.).
Status	Wyświetlany jest status pozycji — *Włączony* lub *Wyłączony*. Klikając status, możesz go zmienić — z Włączony na Wyłączony (i odwrotnie).
Tagi	Wyświetlane są tagi pozycji. Można wyświetlić maksymalnie trzy tagi (pary nazwa:wartość). Jeśli tagów jest więcej, wyświetlany jest link „...", który po najechaniu kursorem pozwala zobaczyć wszystkie tagi.

Aby skonfigurować nową pozycję, kliknij przycisk *Utwórz pozycję* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

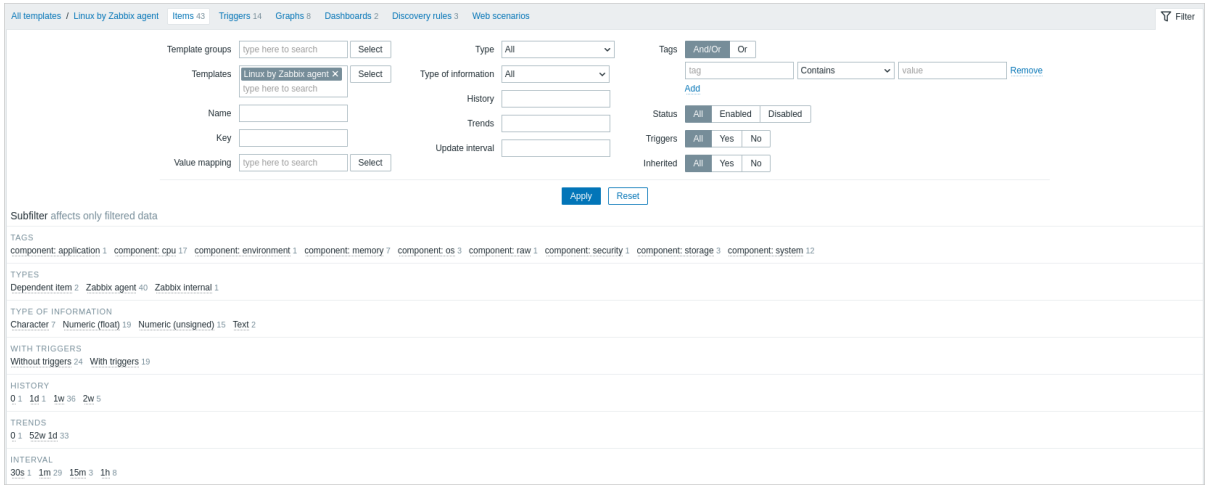
- *Włącz* — zmień status pozycji na *Włączony*.
- *Wyłącz* — zmień status pozycji na *Wyłączony*.
- *Kopiuj* — skopiuj pozycje do innych hostów lub szablonów.
- *Masowa aktualizacja* — zaktualizuj kilka właściwości dla wielu pozycji jednocześnie.
- *Usuń* — usuń pozycje.

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich pozycjach, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Lista pozycji może zawierać wiele pozycji. Korzystając z filtra, możesz odfiltrować część z nich, aby szybko znaleźć szukane pozycje. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania.

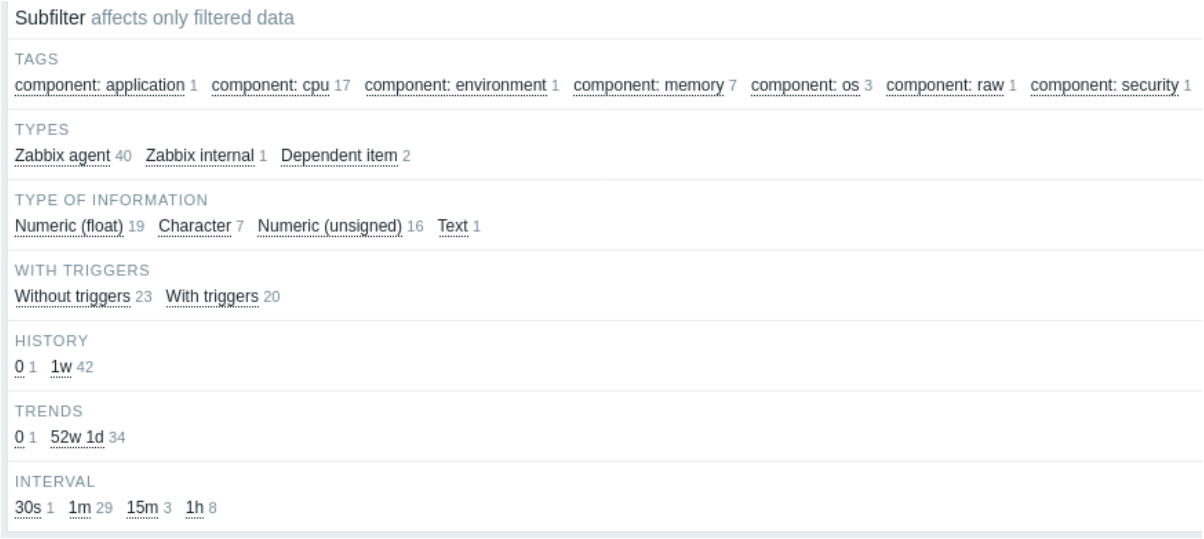


Parametr	Opis
Template groups	Filtruj według jednej lub większej liczby grup szablonów. Określenie nadrzędnej grupy szablonów powoduje domyślnie wybranie wszystkich grup zagnieżdżonych.
Templates	Filtruj według jednego lub większej liczby szablonów.
Name	Filtruj według nazwy pozycji.
Key	Filtruj według klucza pozycji.
Value mapping	Filtruj według używanej mapy wartości. Ten parametr nie jest wyświetlany, jeśli opcja *Templates* jest pusta.
Type	Filtruj według typu pozycji (agent Zabbix, agent SNMP itp.).
Type of information	Filtruj według typu informacji (bezznakowa wartość numeryczna, liczba zmiennoprzecinkowa itp.).
History	Filtruj według czasu przechowywania historii pozycji.
Trends	Filtruj według czasu przechowywania trendów pozycji.
Update interval	Filtruj według interwału aktualizacji pozycji.
Tags	Określ tagi, aby ograniczyć liczbę wyświetlanych pozycji. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter) Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane operatorem Or **Or** - wystarczy spełnienie jednego warunku
Status	Filtruj według statusu pozycji — *Enabled* lub *Disabled*.
Triggers	Filtruj pozycje z wyzwalaczami (lub bez nich).
Inherited	Filtruj pozycje dziedziczone (lub niedziedziczone) z podłączonych szablonów.

Korzystanie z podfiltra

Podfiltr pozwala dalej modyfikować filtrowanie z filtra głównego.

Zawiera klikalne odnośniki umożliwiające szybki dostęp do powiązanych pozycji. Pozycje są powiązane wspólną jednostką, taką jak tag, typ pozycji, interwał aktualizacji pozycji itp. Po kliknięciu jednostki zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (bez potrzeby klikania przycisku *Zastosuj* w filtrze głównym). Kliknięcie innej jednostki dodaje ją do filtrowanych wyników. Ponowne kliknięcie jednostki usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, których liczba jest ograniczona do 1000 rekordów. Jeśli chcesz zobaczyć więcej rekordów w podfiltrze, musisz zwiększyć wartość parametru *Limit for search and filter results* (w *Administration -> General -> GUI*).

W przeciwieństwie do filtra głównego, podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierał aktualne informacje o dostępnych opcjach filtrowania i licznikach.

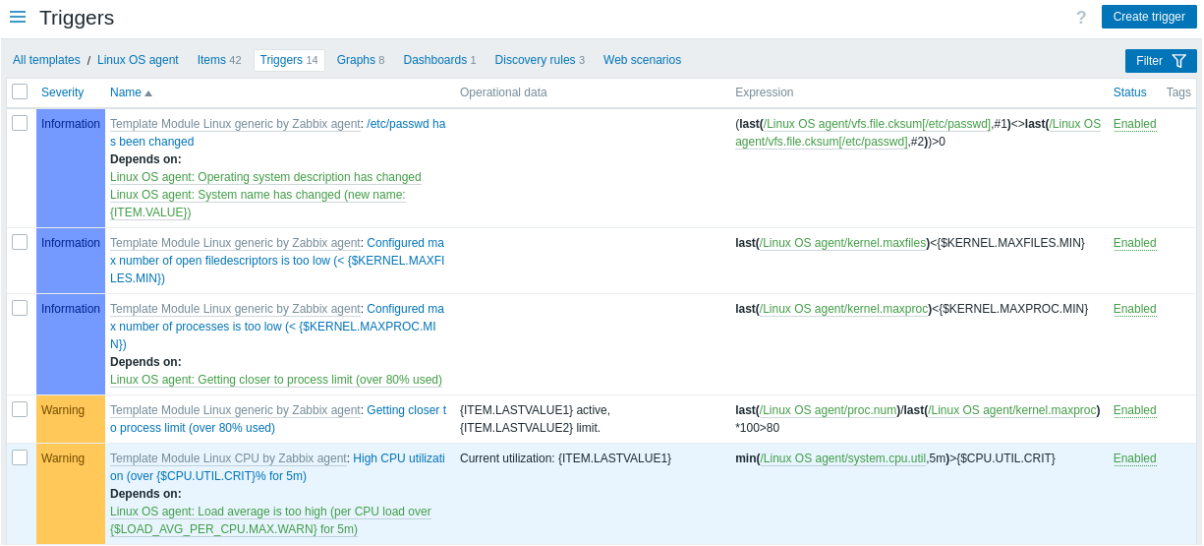
Liczba wyświetlanych jednostek jest ograniczona poziomo do 100. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie można jej kliknąć.

Liczba obok każdej klikalnej jednostki wskazuje liczbę pozycji w niej zgrupowanych (na podstawie wyników filtra głównego). Po kliknięciu jednostki liczby przy innych dostępnych jednostkach są wyświetlane ze znakiem plus, wskazującym, ile pozycji można dodać do bieżącego wyboru. Jednostki bez pozycji nie są wyświetlane, chyba że zostały wcześniej wybrane w podfiltrze.

2 Wyzwalacze

Przegląd

Lista wyzwalaczy dla szablonu jest dostępna w *Zbieranie danych* → *Szablony* po kliknięciu *Wyzwalacze* dla odpowiedniego szablonu.



Wyświetlane dane:

Kolumna	Opis
Istotność	Istotność wyzwalacza jest wyświetlana zarówno nazwą, jak i kolorem tła komórki.

Kolumna	Opis
Szablon	Szablon, do którego należy wyzwalacz. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa wyzwalacza jest wyświetlana jako niebieski link do szczegółów wyzwalacza. Kliknięcie linku z nazwą wyzwalacza otwiera formularz konfiguracji wyzwalacza. Jeśli wyzwalacz jest dziedziczony z innego szablonu, przed nazwą wyzwalacza wyświetlana jest nazwa szablonu jako szary link. Kliknięcie linku szablonu otwiera listę wyzwalaczy na poziomie tego szablonu.
Dane operacyjne	Definicja danych operacyjnych wyzwalacza, zawierająca dowolne ciągi znaków i makra, które będą dynamicznie rozwijane w *Monitorowanie* → *Problemy*.
Wyrażenie	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia szablon-pozycja jest wyświetlana jako link prowadzący do formularza konfiguracji pozycji.
Status	Wyświetlany jest status wyzwalacza — *Włączony* lub *Wyłączony*. Klikając status, można go zmienić — z Włączony na Wyłączony (i z powrotem).
Tagi	Jeśli wyzwalacz zawiera tagi, w tej kolumnie wyświetlane są nazwa i wartość tagu.

Aby skonfigurować nowy wyzwalacz, kliknij przycisk *Utwórz wyzwalacz* w prawym górnym rogu.

Opcje edycji zbiorczej

Przyciski poniżej listy oferują kilka opcji edycji zbiorczej:

- *Enable* - zmień status wyzwalacza na *Enabled*
- *Disable* - zmień status wyzwalacza na *Disabled*
- *Copy* - skopiuj wyzwalacze do innych hostów lub szablonów
- *Mass update* - zaktualizuj kilka właściwości dla wielu wyzwalaczy jednocześnie
- *Delete* - usuń wyzwalacze

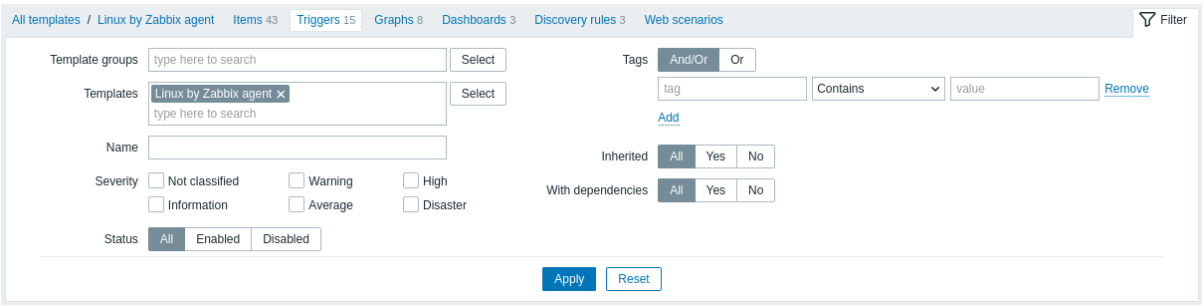
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wyzwalaczami, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko wyzwalacze, które Cię interesują.

Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



Parameter	Description
Template groups	Filtruj według jednej lub kilku grup szablonów. Określenie nadrzędnej grupy szablonów automatycznie wybiera wszystkie zagnieżdżone grupy.
Templates	Filtruj według jednego lub kilku szablonów. Jeśli powyżej wybrano już grupy szablonów, wybór szablonów jest ograniczony do tych grup.
Name	Filtruj według nazwy wyzwalacza.
Severity	Wybierz, aby filtrować według jednej lub kilku istotności wyzwalacza.
Status	Filtruj według statusu wyzwalacza.

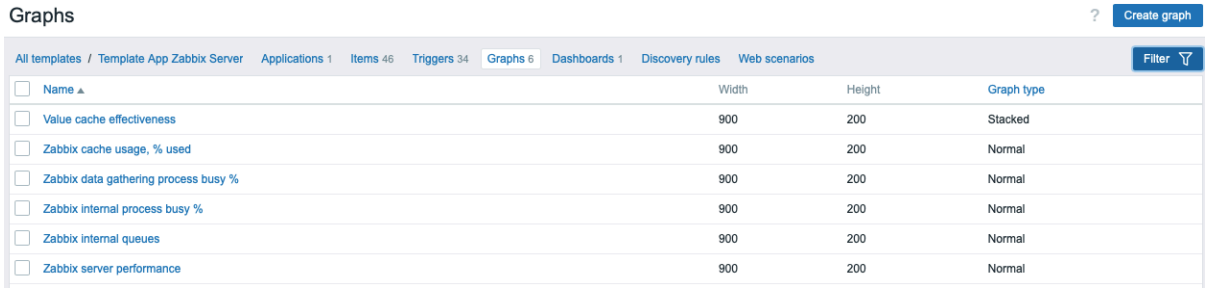
Parameter	Description
Tags	Filtruj według nazwy i wartości tagu wyzwalacza. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony jest jeden warunek Makra i funkcje makr są obsługiwane w polach nazwy tagu i wartości tagu.
Inherited	Filtruj wyzwalacze dziedziczone (lub niedziedziczone) z powiązanych szablonów.
With dependencies	Filtruj wyzwalacze z zależnościami (lub bez zależności).

3 Wykresy

Przegląd

Lista własnych wykresów dla szablonu jest dostępna w *Zbieranie danych* → *Szablony* po kliknięciu *Wykresy* dla odpowiedniego szablonu.

Wyświetlana jest lista istniejących wykresów.



Wyświetlane dane:

Kolumna	Opis
Szablon	Szablon, do którego należy wykres. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa własnego wykresu, wyświetlana jako niebieski link do szczegółów wykresu. Kliknięcie linku z nazwą wykresu otwiera formularz konfiguracji wykresu. Jeśli wykres jest dziedziczony z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary link. Kliknięcie linku szablonu otwiera listę wykresów na poziomie tego szablonu.
Szerokość	Wyświetlana jest szerokość wykresu.
Wysokość	Wyświetlana jest wysokość wykresu.
Typ wykresu	Wyświetlany jest typ wykresu — *Normalny*, *Skumulowany*, *Kołowy* lub *Rozsunięty*.

Aby skonfigurować nowy wykres, kliknij przycisk *Utwórz wykres* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Kopiuj* - kopiowanie wykresów do innych hostów lub szablonów
- *Usuń* - usuwanie wykresów

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich wykresach, a następnie kliknij wymagany przycisk.

Używanie filtra

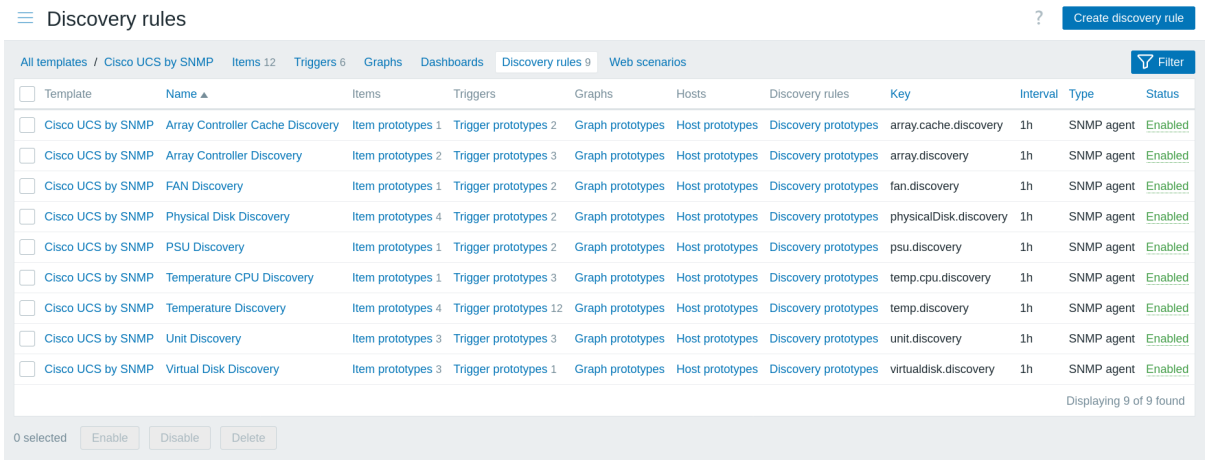
Możesz filtrować wykresy według grupy szablonów i szablonu. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

4 Reguły wykrywania

Przegląd

Lista reguł wykrywania niskiego poziomu dla szablonu jest dostępna w *Zbieranie danych* → *Szablony* po kliknięciu *Wykrywanie* dla odpowiedniego szablonu.

Wyświetlana jest lista istniejących reguł wykrywania niskiego poziomu. Możliwe jest również wyświetlenie wszystkich reguł wykrywania niezależnie od szablonu lub wszystkich reguł wykrywania dla określonej grupy szablonów poprzez zmianę ustawień filtra.



Wyświetlane dane:

Kolumna	Opis
Szablon	Szablon, do którego należy reguła wykrywania. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu.
Nazwa	Nazwa reguły, wyświetlana jako niebieski link. Kliknięcie nazwy reguły otwiera formularz konfiguracji reguły wykrywania niskiego poziomu. Jeśli reguła wykrywania jest dziedziczona z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą reguły jako szary link. Kliknięcie linku szablonu otwiera listę reguł wykrywania na poziomie tego szablonu.
Pozycje	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Wyzwalacze	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Wykresy	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosty	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji używany do wykrywania.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania.
Typ	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itp.).
Status	Wyświetlany jest status reguły wykrywania — *Włączony* lub *Wyłączony*. Klikając status, można go zmienić — z Włączony na Wyłączony (i z powrotem).

Aby skonfigurować nową regułę wykrywania niskiego poziomu, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

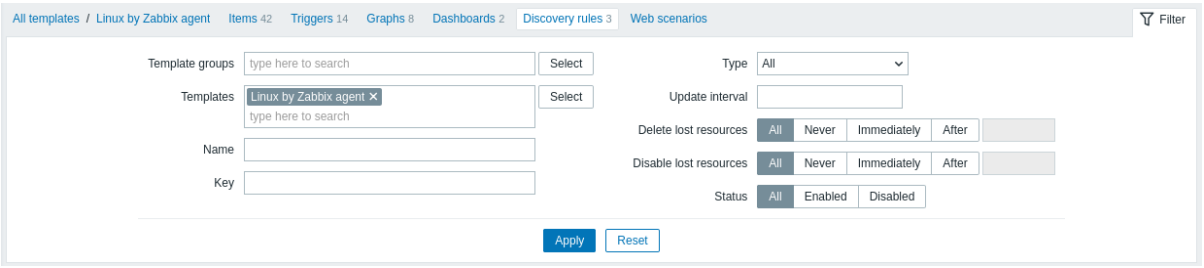
- *Włącz* - zmień status reguły wykrywania niskiego poziomu na *Włączony*
- *Wyłącz* - zmień status reguły wykrywania niskiego poziomu na *Wyłączony*
- *Usuń* - usuń reguły wykrywania niskiego poziomu

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich regułach wykrywania, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania, takie jak szablon, nazwa reguły wykrywania, klucz pozycji, typ pozycji itp.



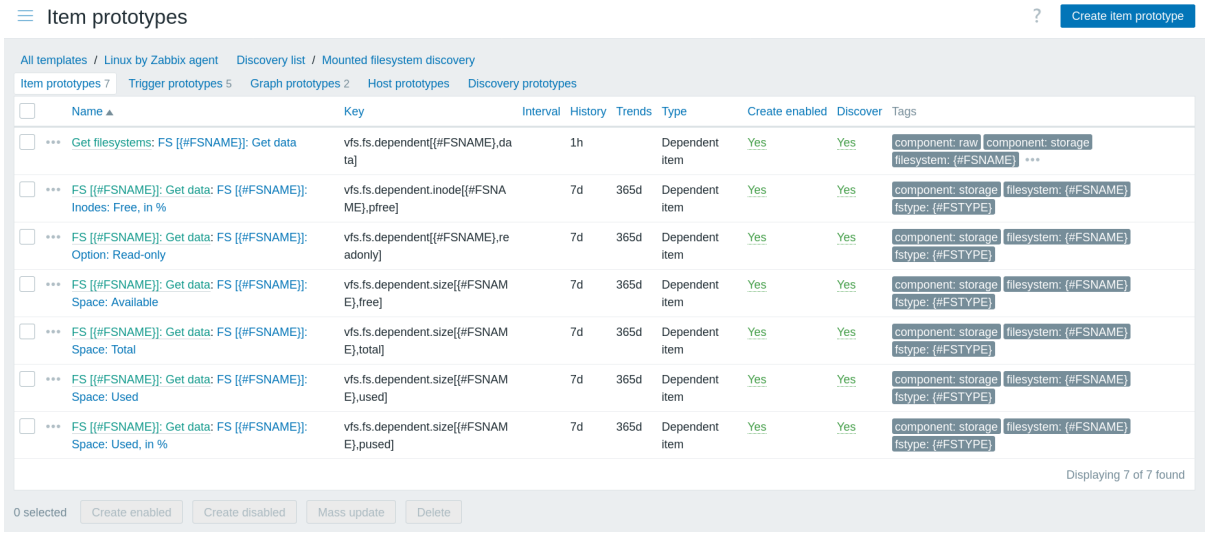
Parametr	Opis
Grupy szablonów	Filtruj według jednej lub większej liczby grup szablonów. Określenie nadrzędnej grupy szablonów powoduje domyślnie wybranie wszystkich grup zagnieżdżonych.
Szablony	Filtruj według jednego lub większej liczby szablonów.
Nazwa	Filtruj według nazwy reguły wykrywania.
Klucz	Filtruj według klucza pozycji wykrywania.
Typ	Filtruj według typu pozycji wykrywania.
Interwał aktualizacji	Filtruj według interwału aktualizacji. Niedostępne dla pozycji typu Zabbix trapper i pozycji zależnych.
Usuwanie utraconych zasobów	Filtruj według okresu *Usuwanie utraconych zasobów*.
Wyłączanie utraconych zasobów	Filtruj według okresu *Wyłączanie utraconych zasobów*.
Status	Filtruj według statusu reguły wykrywania (Wszystkie/Włączone/Wyłączone).

1 Prototypy pozycji

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy pozycji reguły wykrywania niskiego poziomu w szablonie.

Jeśli szablon jest podłączony do hosta, prototypy pozycji staną się podstawą do tworzenia rzeczywistych pozycji hosta podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Wyświetlana jest nazwa prototypu pozycji jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu pozycji. Jeśli prototyp pozycji należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą prototypu pozycji jako szary link. Kliknięcie linku szablonu otwiera listę prototypów pozycji na poziomie podłączonego szablonu. Jeśli prototyp pozycji jest typu „Pozycja zależna”, nazwa pozycji nadrzędnej jest wyświetlana przed nazwą prototypu pozycji na zielono (tak jak w przypadku prototypów pozycji na poziomie hosta).
Klucz	Wyświetlany jest klucz prototypu pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ prototypu pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itp.).
Utwórz włączone	Utwórz pozycję na podstawie tego prototypu jako: **Tak** - włączona **Nie** - wyłączona. Możesz przełączać między „Tak” i „Nie”, klikając je.
Odkrywaj	Odkrywaj pozycję na podstawie tego prototypu: **Tak** - odkrywaj **Nie** - nie odkrywaj. Możesz przełączać między „Tak” i „Nie”, klikając je.
Tagi	Wyświetlane są tagi prototypu pozycji.

Aby skonfigurować nowy prototyp pozycji, kliknij przycisk *Utwórz prototyp pozycji* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te pozycje jako *Włączone*
- *Utwórz wyłączone* - utwórz te pozycje jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy pozycji
- *Usuń* - usuń te prototypy pozycji

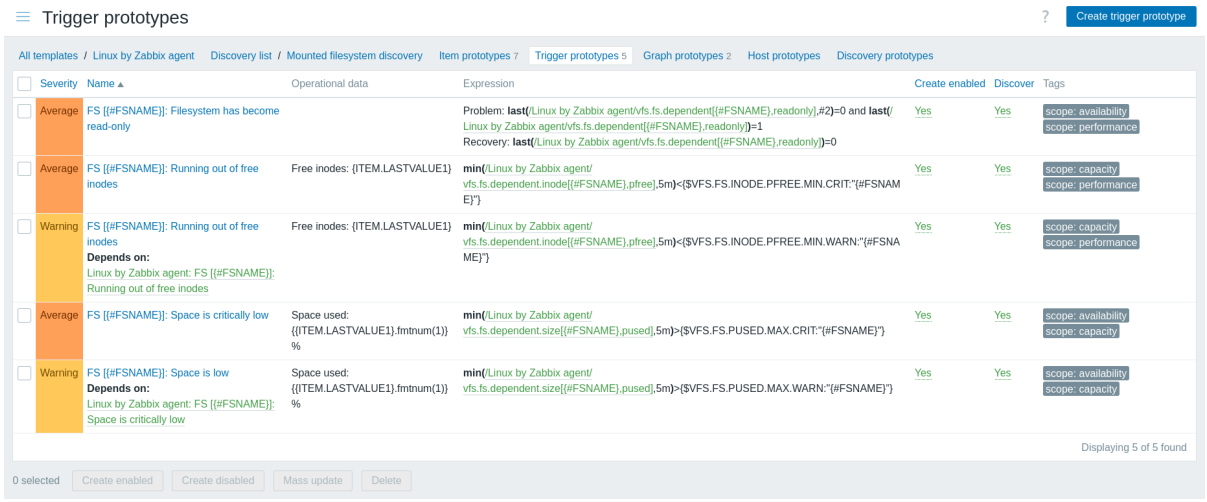
Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach pozycji, a następnie kliknij wymagany przycisk.

2 Prototypy wyzwalaczy

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wyzwalaczy reguły wykrywania niskiego poziomu w szablonie.

Jeśli szablon jest podłączony do hosta, prototypy wyzwalaczy staną się podstawą do tworzenia rzeczywistych wyzwalaczy hosta podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Column	Description
Severity	Poziom ważności wyzwalacza jest wyświetlany zarówno nazwą, jak i kolorem tła komórki.
Name	Nazwa prototypu wyzwalacza, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wyzwalacza. Jeśli prototyp wyzwalacza należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wyzwalaczy na poziomie podłączonego szablonu.
Expression	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia odnosząca się do szablonu i pozycji jest wyświetlana jako link prowadzący do formularza konfiguracji pozycji.
Operational data	Wyświetlany jest format danych operacyjnych wyzwalacza, zawierający dowolne ciągi znaków i makra, które będą dynamicznie rozwijane w *Monitoring* → *Problems*.
Create enabled	Utwórz wyzwalacz na podstawie tego prototypu jako: **Yes** - włączony **No** - wyłączony. Możesz przełączać między „Yes” i „No”, klikając je.
Discover	Wykrywaj wyzwalacz na podstawie tego prototypu: **Yes** - wykrywaj **No** - nie wykrywaj. Możesz przełączać między „Yes” i „No”, klikając je.
Tags	Wyświetlane są tagi prototypu wyzwalacza.

Aby skonfigurować nowy prototyp wyzwalacza, kliknij przycisk *Create trigger prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te wyzwalacze jako *Włączone*
- *Utwórz wyłączone* - utwórz te wyzwalacze jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy wyzwalaczy
- *Usuń* - usuń te prototypy wyzwalaczy

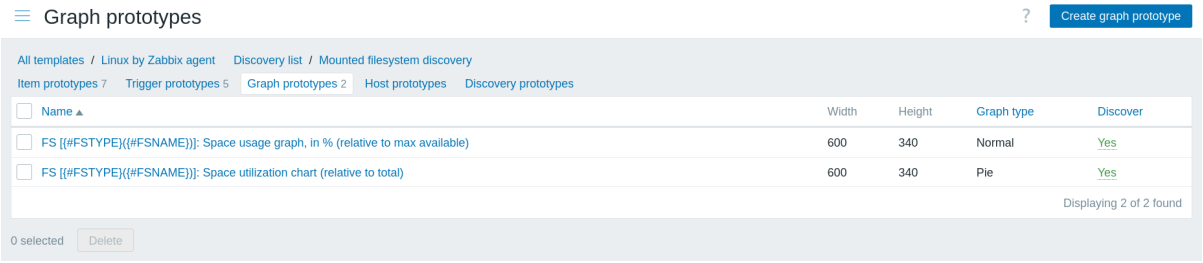
Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wyzwalaczy, a następnie kliknij wymagany przycisk.

3 Prototypy wykresów

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wykresów reguły wykrywania niskiego poziomu w szablonie.

Jeśli szablon jest podłączony do hosta, prototypy wykresów staną się podstawą do tworzenia rzeczywistych wykresów hosta podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wykresu, wyświetlana jako niebieski odnośnik. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykresu. Jeśli prototyp wykresu należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę prototypów wykresów na poziomie podłączonego szablonu.
Szerokość	Wyświetlana jest szerokość prototypu wykresu.
Wysokość	Wyświetlana jest wysokość prototypu wykresu.
Typ	Wyświetlany jest typ prototypu wykresu — *Normalny*, *Skumulowany*, *Kołowy* lub *Rozsunięty*.
Odkrywaj	Odkrywanie wykresu na podstawie tego prototypu: **Tak** - odkrywaj **Nie** - nie odkrywaj. Możesz przełączać między „Tak” i „Nie”, klikając te wartości.

Aby skonfigurować nowy prototyp wykresu, kliknij przycisk *Utwórz prototyp wykresu* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Usuń* - usuń te prototypy wykresów

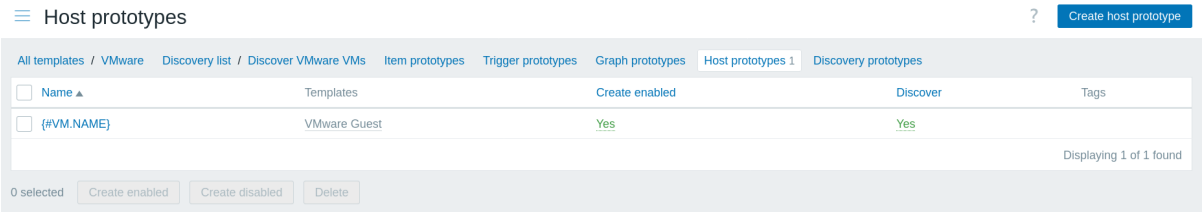
Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wykresów, a następnie kliknij wymagany przycisk.

4 Prototypy hostów

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy hostów reguły wykrywania niskiego poziomu w szablonie.

Jeśli szablon jest podłączony do hosta, prototypy hostów staną się podstawą do tworzenia rzeczywistych hostów podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu hosta, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu hosta. Jeśli prototyp hosta należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów hostów na poziomie podłączonego szablonu.
Szablony	Wyświetlane są szablony prototypu hosta.
Tworzenie włączone	Utwórz host na podstawie tego prototypu jako: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać między „Tak” i „Nie”, klikając te wartości.

Kolumna	Opis
Wykrywaj	Wykrywaj host na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać między „Tak” i „Nie”, klikając te wartości.
Tagi	Wyświetlane są tagi prototypu hosta.

Aby skonfigurować nowy prototyp hosta, kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Create enabled* — utwórz te hosty jako *Enabled*
- *Create disabled* — utwórz te hosty jako *Disabled*
- *Delete* — usuń te prototypy hostów

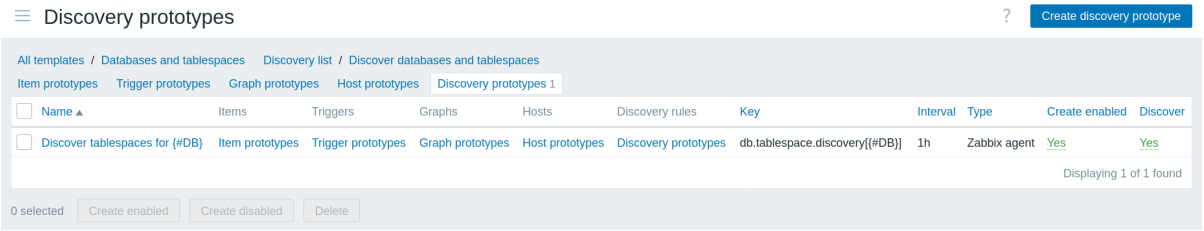
Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich prototypach hostów, a następnie kliknij wymagany przycisk.

5 Prototypy wykrywania

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wykrywania reguły wykrywania niskiego poziomu w szablonie.

Jeśli szablon jest podłączony do hosta, prototypy wykrywania staną się podstawą do tworzenia rzeczywistych prototypów wykrywania nadrzędnej reguły wykrywania.



Wyświetlane dane:

Column	Description
Name	Nazwa prototypu wykrywania, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykrywania. Jeśli prototyp wykrywania należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą prototypu wykrywania jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykrywania na poziomie podłączonego szablonu.
Items	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Triggers	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Graphs	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosts	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Discovery rules	Wyświetlany jest link do listy prototypów wykrywania. Liczba istniejących prototypów wykrywania jest wyświetlana na szaro.
Key	Wyświetlany jest klucz pozycji używany do wykrywania.
Interval	Wyświetlana jest częstotliwość wykonywania wykrywania.
Type	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itp.).
Create enabled	Wyświetlany jest status tworzenia prototypu wykrywania — tworzenie włączone (*Yes*) lub tworzenie wyłączone (*No*). Klikając status, można go zmienić — z Yes na No (i z powrotem).
Discover	Wyświetlany jest status wykrywania prototypu wykrywania — wykrywaj (*Yes*) lub nie wykrywaj (*No*). Klikając status, można go zmienić — z Yes na No (i z powrotem).

Aby skonfigurować nowy prototyp wykrywania, kliknij przycisk *Create discovery prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te prototypy wykrywania jako *Włączone*
- *Utwórz wyłączone* - utwórz te prototypy wykrywania jako *Wyłączone*
- *Usuń* - usuń te prototypy wykrywania

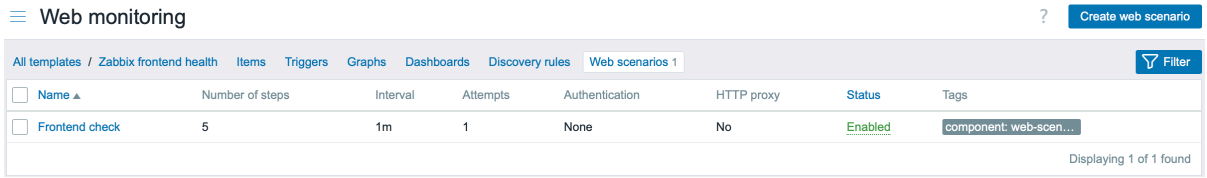
Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wykrywania, a następnie kliknij wymagany przycisk.

5 Scenariusze WWW

Przegląd

Do listy scenariuszy WWW dla szablonu można przejść z poziomu *Data collection* → *Templates*, klikając *Web* przy odpowiednim szablonie.

Zostanie wyświetlona lista istniejących scenariuszy WWW.



Wyświetlane dane:

Kolumna	Opis
Name	Nazwa scenariusza WWW. Kliknięcie nazwy scenariusza WWW otwiera formularz konfiguracji scenariusza WWW. Jeśli scenariusz WWW jest dziedziczony z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą scenariusza WWW jako szare łącze. Kliknięcie łącza szablonu otwiera listę scenariuszy WWW na poziomie tego szablonu.
Number of steps	Liczba kroków zawartych w scenariuszu.
Update interval	Jak często scenariusz jest wykonywany.
Attempts	Ile prób wykonania kroków scenariusza WWW jest podejmowanych.
Authentication	Wyświetlana jest metoda uwierzytelniania — Basic, NTLM, Kerberos, Digest lub None.
HTTP proxy	Wyświetla HTTP proxy lub „No”, jeśli nie jest używany.
Status	Wyświetlany jest status scenariusza WWW — *Enabled* lub *Disabled*. Kliknięcie statusu pozwala go zmienić.
Tags	Wyświetlane są tagi scenariusza WWW. Można wyświetlić maksymalnie trzy tagi (pary nazwa:wartość). Jeśli tagów jest więcej, wyświetlane jest łącze „...”, które po najechaniu kursorem pozwala zobaczyć wszystkie tagi.

Aby skonfigurować nowy scenariusz WWW, kliknij przycisk *Create web scenario* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

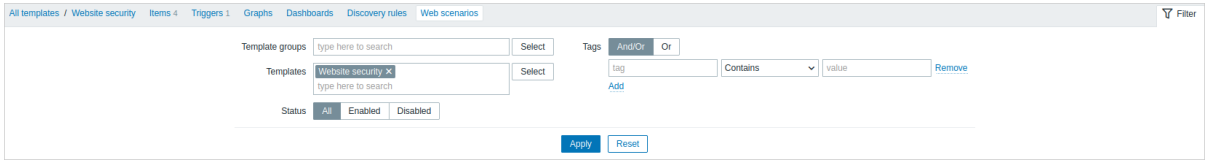
- *Włącz* - zmień status scenariusza na *Włączony*
- *Wyłącz* - zmień status scenariusza na *Wyłączony*
- *Usuń* - usuń scenariusze web

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich scenariuszach web, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko scenariusze, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą scenariuszy WWW. Po jego kliknięciu pojawi się filtr, za pomocą którego można filtrować scenariusze według grupy szablonów, szablonu, statusu i tagów.

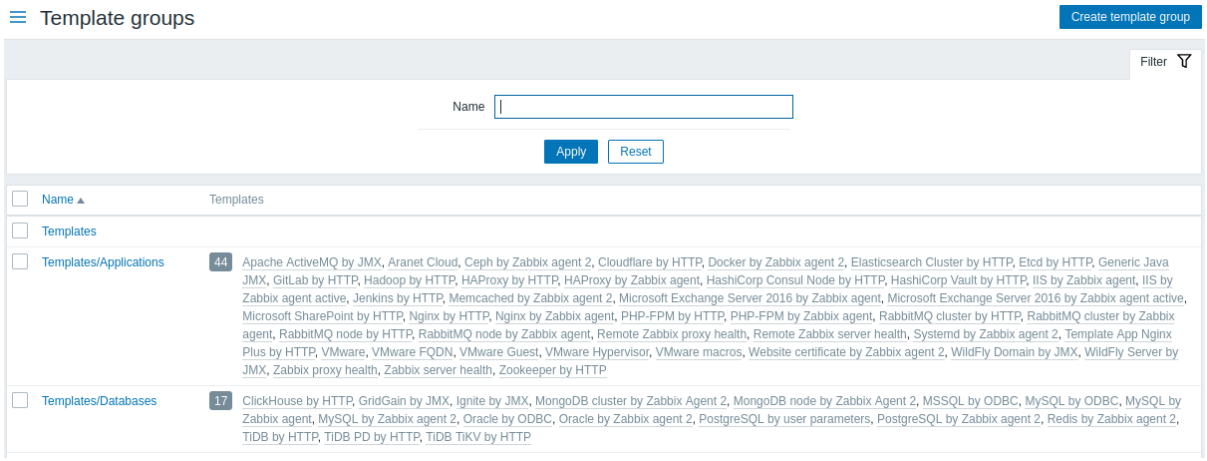


1 Grupy szablonów

Przegląd

W sekcji *Zbieranie danych* → *Grupy szablonów* użytkownicy mogą konfigurować i zarządzać grupami szablonów.

Wyświetlana jest lista istniejących grup szablonów wraz z ich szczegółami. Można wyszukiwać i filtrować grupy szablonów według nazwy.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy szablonów. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy.
Szablony	Liczba szablonów w grupie (wyświetlana na szaro), po której następuje lista członków grupy. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Kliknięcie liczby otwiera listę szablonów w tej grupie.

Opcje masowej edycji

Aby usunąć jednocześnie kilka grup szablonów, zaznacz pola wyboru przy odpowiednich grupach, a następnie kliknij przycisk Delete poniżej listy.

Używanie filtra

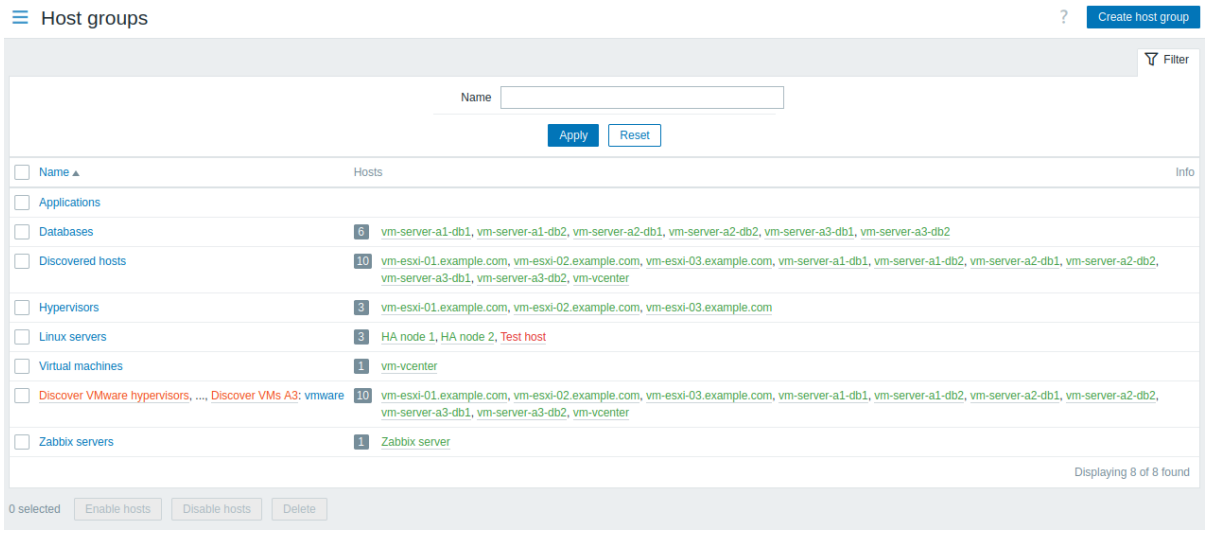
Możesz użyć filtra, aby wyświetlić tylko te grupy szablonów, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane przy nierozwiniętych makrach.

2 Grupy hostów

Przegląd

W sekcji *Zbieranie danych* → *Grupy hostów* użytkownicy mogą konfigurować i zarządzać grupami hostów.

Wyświetlana jest lista istniejących grup hostów wraz z ich szczegółami. Można wyszukiwać i filtrować grupy hostów według nazwy.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy hostów. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy. Wykryte grupy hostów są wyświetlane z nazwami reguł wykrywania niskiego poziomu jako prefiksem. Kliknięcie nazwy reguły LLD otwiera formularz konfiguracji prototypu hosta. Należy pamiętać, że wykryte grupy hostów są usuwane, gdy wszystkie reguły LLD, które je wykryły, zostaną usunięte.
Hosty	Liczba hostów w grupie (wyświetlana na szaro), po której następuje lista członków grupy. Kliknięcie nazwy hosta otworzy formularz konfiguracji hosta. Kliknięcie liczby spowoduje odfiltrowanie na pełnej liście hostów tych, które należą do grupy.
Informacje	Wyświetlane są informacje o błędach (jeśli występują) dotyczące grupy hostów.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Włącz hosty* — zmienia status wszystkich hostów w grupie na „Monitorowany”
- *Wyłącz hosty* — zmienia status wszystkich hostów w grupie na „Niemonitorowany”
- *Usuń* — usuwa grupy hostów

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich grupach hostów, a następnie kliknij wymagany przycisk.

Używanie filtra

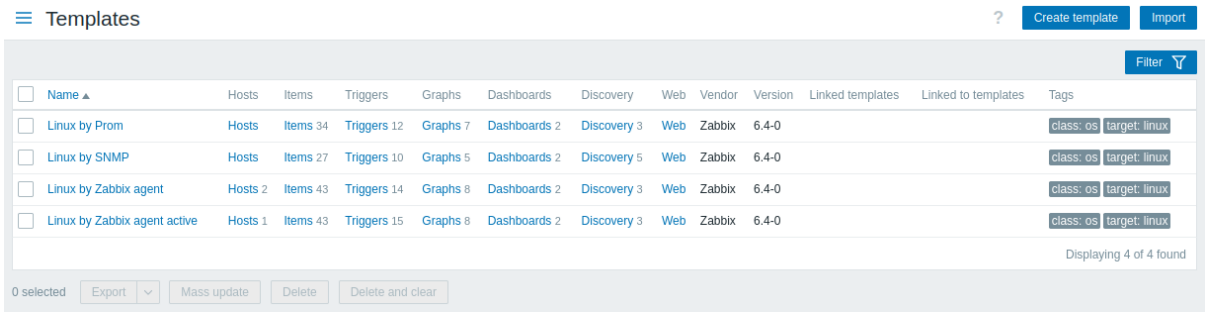
Możesz użyć filtra, aby wyświetlić tylko te grupy hostów, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

3 Szablony

Omówienie

W sekcji *Data collection* → *Templates* użytkownicy mogą konfigurować i zarządzać szablonami.

Wyświetlana jest lista istniejących szablonów wraz z ich szczegółami.



Wyświetlane dane:

Column	Description
Name	Nazwa szablonu. Kliknięcie nazwy szablonu otwiera formularz konfiguracji.
Hosts	Liczba edytowalnych hostów, z którymi szablon jest powiązany; hosty tylko do odczytu nie są uwzględniane. Kliknięcie *Hosts* otworzy listę hostów z filtrem obejmującym wyłącznie te hosty, które są powiązane z szablonem.
Entities (Items, Triggers, Graphs, Dashboards, Discovery, Web)	Liczba odpowiednich encji w szablonie (wyświetlana na szaro). Kliknięcie nazwy encji spowoduje odfiltrowanie w całej liście tej encji tych elementów, które należą do szablonu.
Linked templates	Szablony powiązane z szablonem.
Linked to templates	Szablony, z którymi szablon jest powiązany.
Vendor, *Version*	Dostawca i wersja szablonu; wyświetlane, jeśli konfiguracja szablonu zawiera takie informacje, i tylko dla szablonów dostarczanych wraz z produktem, zaimportowanych szablonów lub szablonów zmodyfikowanych za pomocą Template API. W przypadku szablonów dostarczanych wraz z produktem wersja jest wyświetlana w następującej postaci: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład: 7.0-0, 7.0-3, 7.4-0, 7.4-3.
Tags	Tagi szablonu, z nierozwiniętymi makrami.

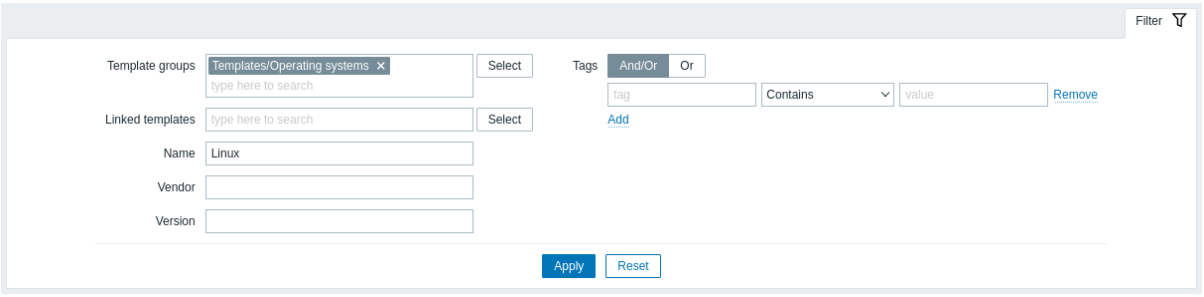
Aby skonfigurować nowy szablon, kliknij przycisk *Create template* w prawym górnym rogu.

Aby zaimportować szablon z pliku YAML, XML lub JSON, kliknij przycisk *Import* w prawym górnym rogu.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te szablony, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Łącze *Filtr* jest dostępne poniżej przycisków *Utwórz szablon* i *Importuj*. Po jego kliknięciu dostępny staje się filtr, w którym można filtrować szablony według grup szablonów, podłączonych szablonów, nazwy i tagów.



Parametr	Opis
Grupy szablonów	Filtruj według jednej lub większej liczby grup szablonów. Określenie nadrzędnej grupy szablonów powoduje domyślny wybór wszystkich grup zagnieżdżonych.
Podłączone szablony	Filtruj według bezpośrednio podłączonych szablonów.
Nazwa	Filtruj według nazwy szablonu.
Dostawca	Filtruj według dostawcy szablonu.
Wersja	Filtruj według wersji szablonu.

Parametr	Opis
Tagi	Filtruj według nazwy i wartości tagu szablonu. Filtrowanie jest możliwe tylko według tagów na poziomie szablonu (nie dziedziczonych). Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane przez warunek Or; **Or** - wystarczy, że jeden warunek zostanie spełniony.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

- *Eksport* — eksport szablonu do pliku YAML, XML lub JSON;
- *Masowa aktualizacja* — zaktualizuj kilka właściwości dla wielu szablonów jednocześnie;
- *Usuń* — usuń szablon, pozostawiając jego powiązane encje (pozycje, wyzwalacze itp.) na hostach;
- *Usuń i wyczyść* — usuń szablon i jego powiązane encje z hostów.

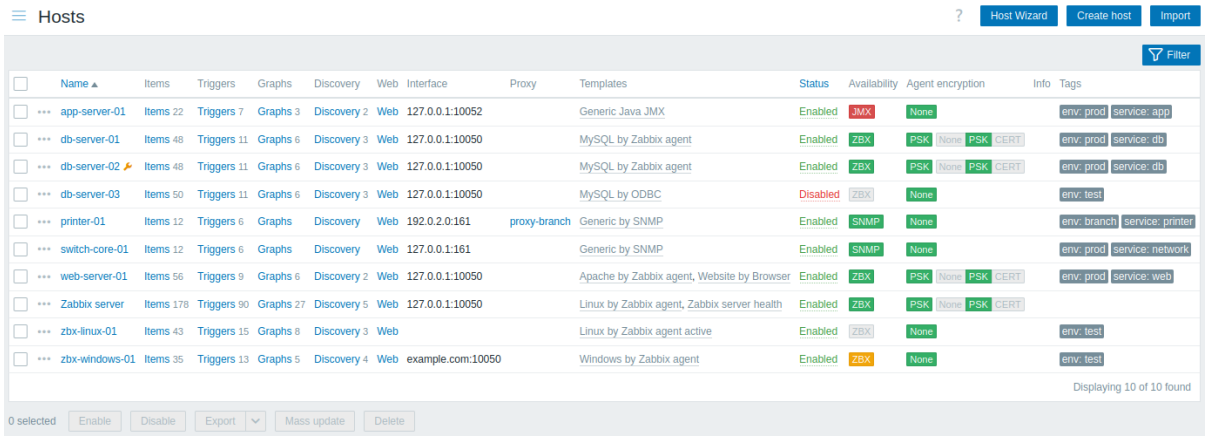
Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich szablonach, a następnie kliknij wymagany przycisk.

4 Hosty

Przegląd

W sekcji *Zbieranie danych* → *Hosty* użytkownicy mogą konfigurować i utrzymywać hosty.

Wyświetlana jest lista istniejących hostów wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Menu hosta	Kliknij ikonę trzech kropek, aby otworzyć menu hosta.
Nazwa	Nazwa hosta. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Pomarańczowa ikona klucza po nazwie hosta wskazuje, że ten host jest w trakcie konserwacji. Szczegóły konserwacji są wyświetlane po najechaniu kursorem myszy na ikonę.

Kolumna	Opis
Encje (Pozycje, Wyzwalacze, Wykresy, Odkrywanie, WWW)	Kliknięcie nazwy encji spowoduje wyświetlenie pozycji, wyzwalaczy itd. hosta. Liczba odpowiednich encji jest wyświetlana na szaro.
Interfejs	Wyświetlany jest główny interfejs hosta.
Proxy	Przypisane proxy są wyświetlane w tej kolumnie: <Nazwa proxy> - host jest monitorowany przez samodzielny proxy (nawet jeśli proxy jest częścią grupy proxy); <Nazwa grupy proxy: nazwa proxy> - host jest monitorowany przez grupę proxy, a serwer Zabbix przypisał proxy do monitorowania hosta; <Nazwa grupy proxy> - host jest monitorowany przez grupę proxy bez żadnych proxy lub jeśli serwer Zabbix nie przypisał proxy do monitorowania hosta; Brak - host nie jest monitorowany ani przez proxy, ani przez grupę proxy. Ta kolumna jest wyświetlana tylko wtedy, gdy opcja filtra *Monitorowany przez* jest ustawiona na „Dowolny”, „Proxy” lub „Grupa proxy”.
Szablony	Wyświetlane są szablony podłączone do hosta. Jeśli podłączony szablon zawiera inne szablony, są one wyświetlane w nawiasach, oddzielone przecinkami. Kliknięcie nazwy szablonu otworzy jego formularz konfiguracji.
Status	Wyświetlany jest status hosta — *Włączony* lub *Wyłączony*. Klikając status, można go zmienić ręcznie. Odnalezione hosty, które zostały utracone, są oznaczone ikoną informacji. Tekst podpowiedzi zawiera szczegóły dotyczące ich statusu.

Kolumna	Opis
Dostępność	Wyświetlana jest dostępność hosta dla każdego skonfigurowanego interfejsu. Ikony dostępności przedstawiają bieżący status interfejsu hosta na serwerze Zabbix. Dlatego jeśli wyłączysz host w frontend, jego dostępność zostanie zaktualizowana po zsynchronizowaniu zmian konfiguracji przez serwer Zabbix. Podobnie, jeśli włączysz host, jego dostępność zostanie zaktualizowana po zsynchronizowaniu zmian konfiguracji przez serwer Zabbix i odpytaniu hosta. Ikony dostępności przedstawiają tylko te typy interfejsów (Agent, SNMP, IPMI, JMX), które są skonfigurowane. Najechanie kursorem na ikonę powoduje wyświetlenie wyskakującego okna z listą wszystkich interfejsów tego samego typu wraz ze szczegółami, statusem i błędami. Dla interfejsu Agent wyskakujące okno wyświetla interfejsy (pasywne) oraz aktywne testy. Jeśli host ma tylko aktywne testy, ikona interfejsu Agent jest wyświetlana nawet wtedy, gdy host nie ma skonfigurowanego interfejsu Agent. Kolumna jest pusta dla hostów bez interfejsów. Status pojedynczego interfejsu hosta jest określany przez połączenie między włączoną pozycją używającą tego interfejsu a hostem. Status może być następujący: **Dostępny** - połączenie z hostem zakończyło się powodzeniem; **Niedostępny** - połączenie z hostem nie powiodło się (przekroczenie czasu, problemy z firewallem itp.); **Nieznany** - nie podjęto próby połączenia z hostem lub wynik jest nieznany. Aby uzyskać dodatkowe informacje o tym, jak serwer Zabbix określa status interfejsu, zobacz Nieznany status interfejsu oraz Ustawienia nieosiągalnego/niedostępnego interfejsu hosta. Status wszystkich interfejsów hosta jednego typu (Agent, SNMP, IPMI, JMX) jest określany przez te interfejsy, które są używane przez co najmniej jedną włączoną pozycję. Status jest wskazywany kolorem ikony: **Zielony** - wszystkie interfejsy są dostępne; **Żółty** - co najmniej jeden interfejs jest niedostępny, a co najmniej jeden jest dostępny lub nieznany; **Czerwony** - wszystkie interfejsy są niedostępne; **Szary** - co najmniej jeden interfejs jest nieznany, ale żaden nie jest niedostępny. **Dostępność aktywnych testów.** Jeśli na hoście jest włączony co najmniej jeden aktywny test, aktywne testy również wpływają na całkowitą dostępność interfejsu Agent, jak opisano powyżej. Aby określić dostępność aktywnych testów, komunikaty heartbeat są wysyłane w wątku aktywnych testów agenta. Częstotliwość komunikatów heartbeat jest kontrolowana przez parametr `HeartbeatFrequency` w konfiguracji Zabbix agent lub agent 2 (domyślnie 60 sekund, zakres 0-3600). Aktywne testy są uznawane za niedostępne, gdy heartbeat aktywnych testów jest starszy niż 2 x HeartbeatFrequency sekund. **Uwaga:** Agenty Zabbix starsze niż wersja 6.2.x nie wysyłają heartbeatów aktywnych testów, więc dostępność ich hostów pozostaje nieznana.
Szyfrowanie agenta	Wyświetlany jest status szyfrowania dla połączeń do hosta i z hosta: **Brak** - brak szyfrowania; **PSK** - użycie klucza współdzielonego; **Cert** - użycie certyfikatu.
Informacje	Wyświetlane są informacje o błędach dotyczących hosta (jeśli występują).
Tagi	Tagi hosta z nierozwiniętymi makrami.

Aby skonfigurować nowy host, kliknij przycisk *Host Wizard* lub *Utwórz host* w prawym górnym rogu. Aby zaimportować host z pliku YAML, XML lub JSON, kliknij przycisk *Importuj* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

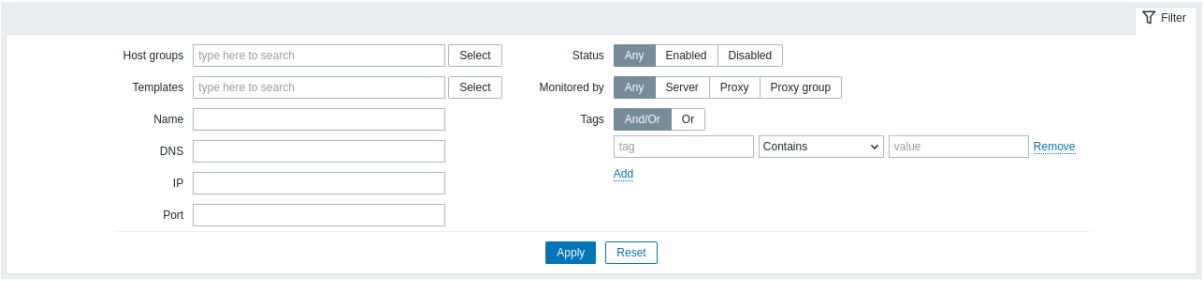
- *Włącz* — zmień status hosta na *Monitorowany*;
- *Wyłącz* — zmień status hosta na *Niemonitorowany*;
- *Eksportuj* — wyeksportuj hosty do pliku YAML, XML lub JSON;
- *Masowa aktualizacja* — zaktualizuj kilka właściwości dla wielu hostów jednocześnie;
- *Usuń* — usuń hosty.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich hostach, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko hosty, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania.



Parametr	Opis
Grupy hostów	Filtruj według jednej lub większej liczby grup hostów. Określenie nadrzędnej grupy hostów powoduje domyślnie wybranie wszystkich zagnieżdżonych grup hostów.
Szablony	Filtruj według podłączonych szablonów.
Nazwa	Filtruj według widocznej nazwy hosta.
DNS	Filtruj według nazwy DNS.
IP	Filtruj według adresu IP.
Port	Filtruj według numeru portu.
Status	Filtruj według statusu hosta.
Monitorowane przez	Filtruj hosty monitorowane przez serwer Zabbix, proxy lub grupę proxy.
Proxy	Filtruj hosty monitorowane przez określone tutaj proxy. To pole jest dostępne tylko wtedy, gdy w polu *Monitorowane przez* wybrano „Proxy”.
Grupy proxy	Filtruj hosty monitorowane przez określone tutaj grupy proxy. To pole jest dostępne tylko wtedy, gdy w polu *Monitorowane przez* wybrano „Grupa proxy”.
Tagi	Filtruj według nazwy i wartości tagu hosta. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** — uwzględnij określone nazwy tagów; **Równa się** — uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** — uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** — wyklucz określone nazwy tagów; **Nie równa się** — wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** — wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** — wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** — wystarczy spełnienie jednego warunku.

Nieznany status interfejsu

Serwer Zabbix wyświetla status „Unknown” dla interfejsu hosta (Agent, SNMP, IPMI, JMX) w następujących przypadkach:

- Host jest wyłączony.
- Host jest ustawiony do monitorowania przez proxy, inny proxy lub serwer, jeśli wcześniej był monitorowany przez proxy.
- Host jest monitorowany przez proxy, który wydaje się być offline (nie otrzymano żadnych aktualizacji z proxy podczas maksymalnego interwału heartbeat — 1 godzina).
- Wszystkie pozycje hosta z tym typem interfejsu są wyłączone.
- Nie skonfigurowano żadnych pollerów dla tego typu interfejsu (na przykład parametr konfiguracji serwera `StartAgentPollers` lub `StartSNMPPollers` jest ustawiony na 0).

Dostępność interfejsu jest ustawiana na „Unknown” po synchronizacji pamięci podręcznej konfiguracji serwera Zabbix.

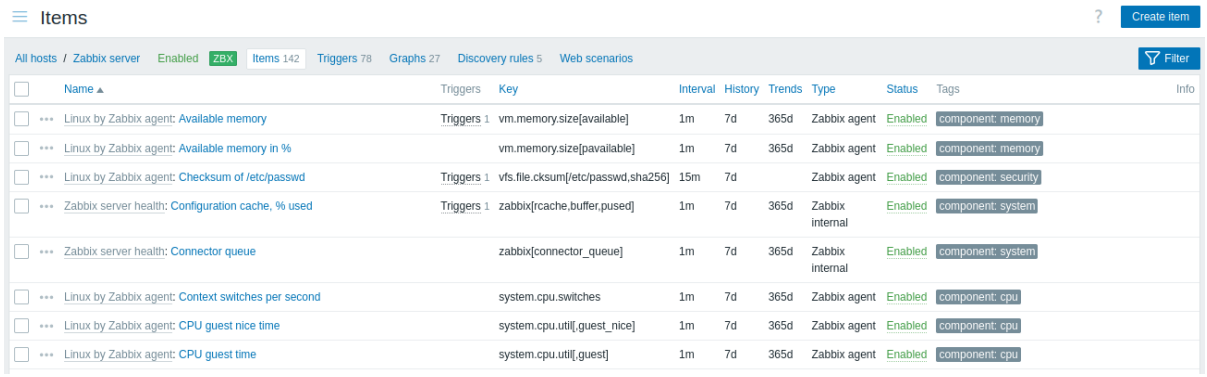
Dostępność interfejsu (Available/Not available) na hostach monitorowanych przez proxy jest przywracana po synchronizacji pamięci podręcznej konfiguracji proxy.

1 Pozycje

Przegląd

Do listy pozycji dla hosta można przejść z poziomu *Zbieranie danych* → *Hosty*, klikając *Pozycje* przy odpowiednim hoście.

Zostanie wyświetlona lista istniejących pozycji.



Wyświetlane dane:

Column	Description
Menu pozycji	Kliknij ikonę z trzema kropkami, aby otworzyć menu pozycji.
Host	Host pozycji. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele hostów lub nie wybrano żadnego hosta.
Nazwa	Nazwa pozycji jest wyświetlana jako niebieski odnośnik do szczegółów pozycji. Kliknięcie odnośnika z nazwą pozycji otwiera formularz konfiguracji pozycji. Jeśli pozycja hosta należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą pozycji jako szary odnośnik. Kliknięcie odnośnika szablonu otworzy listę pozycji na poziomie szablonu. Jeśli pozycja została utworzona z prototypu pozycji, jej nazwa jest poprzedzona nazwą reguły wykrywania niskiego poziomu, wyświetlaną na pomarańczowo. Kliknięcie nazwy reguły wykrywania otworzy listę prototypów pozycji.
Wyzwalacze	Po najechaniu kursorem na Wyzwalacze zostanie wyświetlone okno informacyjne pokazujące wyzwalacze powiązane z pozycją. Liczba wyzwalaczy jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania. *Uwaga*: pozycje pasywne mogą być również sprawdzane natychmiast po naciśnięciu przycisku *Wykonaj teraz* button.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itp.).
Status	Wyświetlany jest status pozycji — *Włączona*, *Wyłączona* lub *Nieobsługiwana*. Status można zmienić ręcznie, klikając go — z Włączona na Wyłączona (i z powrotem); z Nieobsługiwana na Wyłączona (i z powrotem). Odnalezione pozycje, które zostały utracone, są oznaczone ikoną informacji. Tekst podpowiedzi zawiera szczegóły dotyczące ich statusu.
Tagi	Wyświetlane są tagi pozycji. Można wyświetlić maksymalnie trzy tagi (pary nazwa:wartość). Jeśli tagów jest więcej, wyświetlany jest odnośnik „...”, który po najechaniu kursorem pozwala zobaczyć wszystkie tagi.
Informacje	Jeśli pozycja działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nową pozycję, kliknij przycisk *Utwórz pozycję* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

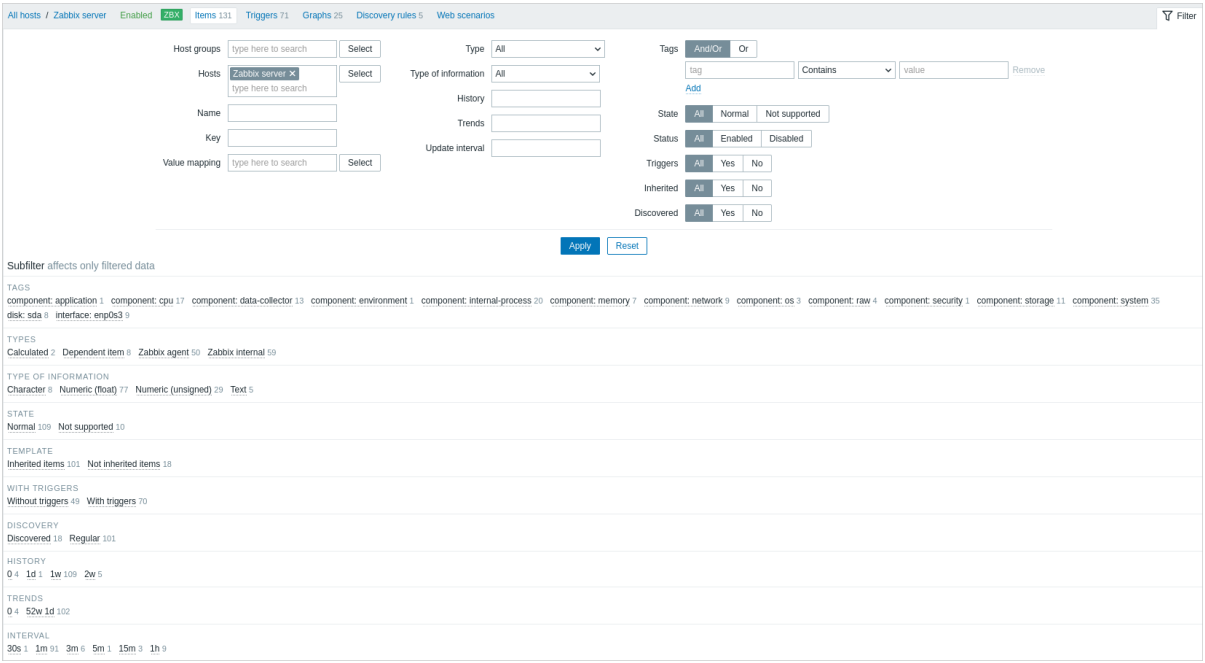
- *Włącz* - zmień status pozycji na *Włączony*
- *Wyłącz* - zmień status pozycji na *Wyłączony*
- *Wykonaj teraz* - natychmiast wykonaj sprawdzenie nowych wartości pozycji. Obsługiwane tylko dla sprawdzeń **pasywnych** (zobacz więcej szczegółów). Należy pamiętać, że podczas natychmiastowego sprawdzania wartości pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wartości nie będą odzwierciedlać bardzo niedawnych zmian w konfiguracji pozycji.
- *Wyczyść historię i trendy* - usuń dane historii i trendów dla pozycji.
- *Kopiuj* - skopiuj pozycje do innych hostów lub szablonów.
- *Masowa aktualizacja* - zaktualizuj kilka właściwości dla wielu pozycji jednocześnie.
- *Usuń* - usuń pozycje.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich pozycjach, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te pozycje, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania.



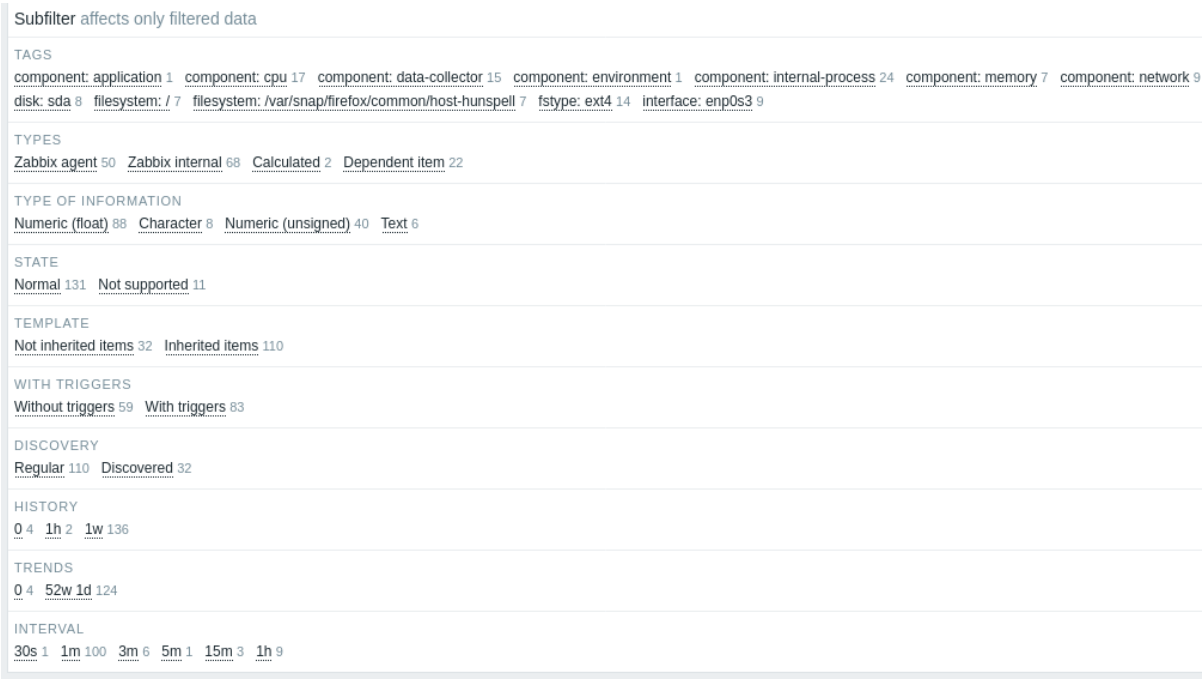
Parametr	Opis
Grupy hostów	Filtruj według jednej lub większej liczby grup hostów. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Nie można wybrać grup hostów zawierających wyłącznie szablony.
Hosty	Filtruj według jednego lub większej liczby hostów.
Nazwa	Filtruj według nazwy pozycji.
Klucz	Filtruj według klucza pozycji.
Mapowanie wartości	Filtruj według używanej mapy wartości. Ten parametr nie jest wyświetlany, jeśli opcja *Hosty* jest pusta.
Typ	Filtruj według typu pozycji (agent Zabbix, agent SNMP itp.).
Typ informacji	Filtruj według typu informacji (bezznakowa wartość liczbowa, liczba zmiennoprzecinkowa itp.).
Historia	Filtruj według czasu przechowywania historii pozycji.
Trendy	Filtruj według czasu przechowywania trendów pozycji.
Interwał aktualizacji	Filtruj według interwału aktualizacji pozycji.

Parametr	Opis
Tagi	Określ tagi, aby ograniczyć liczbę wyświetlanych pozycji. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) **Nie istnieje** - wyklucz określone nazwy tagów **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Nie zawiera** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter) Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or **Or** - wystarczy spełnienie jednego warunku
Stan	Filtruj według stanu pozycji — *Normalny* lub *Nieobsługiwany*.
Status	Filtruj według statusu pozycji — *Włączony* lub *Wyłączony*.
Wyzwalacze	Filtruj pozycje z wyzwalaczami (lub bez nich).
Dziedziczone	Filtruj pozycje dziedziczone (lub niedziedziczone) z szablonu.
Odkrywanie	Filtruj pozycje wykryte (lub niewykryte) przez wykrywanie niskiego poziomu.

Używanie podfiltra

Podfiltr pozwala dalej modyfikować filtrowanie z filtra głównego.

Zawiera klikalne odnośniki umożliwiające szybki dostęp do powiązanych pozycji. Pozycje są powiązane wspólną jednostką, taką jak tag, typ pozycji, stan pozycji, interwał aktualizacji pozycji itp. Po kliknięciu jednostki zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (bez potrzeby klikania *Zastosuj* w filtrze głównym). Kliknięcie innej jednostki dodaje ją do filtrowanych wyników. Ponowne kliknięcie jednostki usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, których liczba jest ograniczona do 1000 rekordów. Jeśli chcesz zobaczyć więcej rekordów w podfiltrze, musisz zwiększyć wartość parametru *Limit for search and filter results* (w *Administracja -> Ogólne -> GUI*).

W przeciwieństwie do filtra głównego, podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierał aktualne informacje o dostępnych opcjach filtrowania i licznikach.

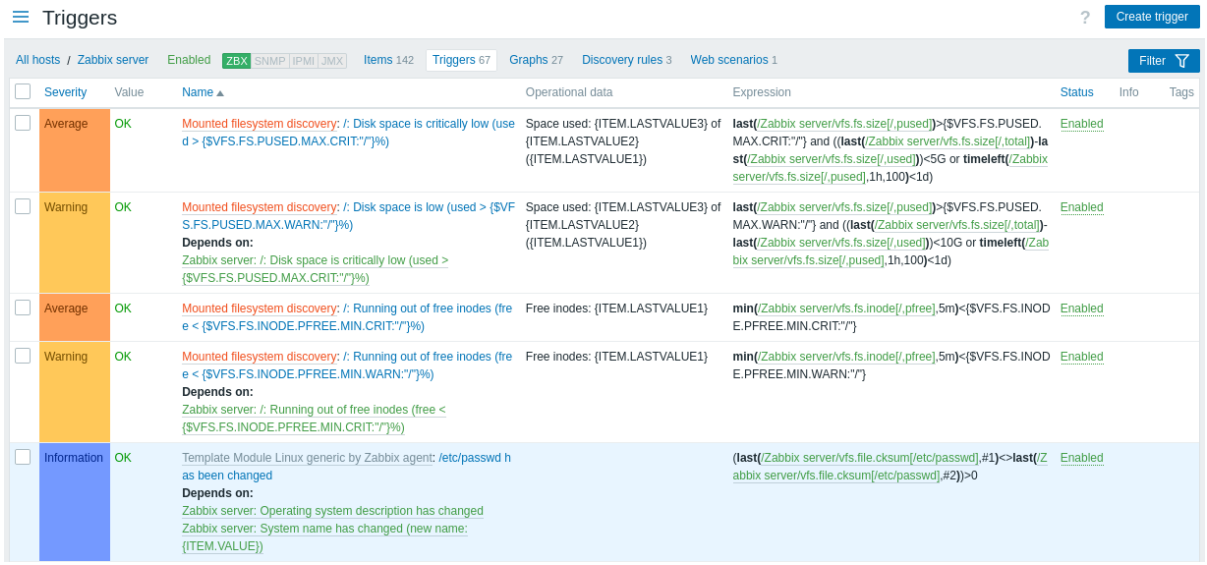
Liczba wyświetlanych jednostek jest ograniczona poziomo do 100. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie można jej kliknąć.

Liczba obok każdej klikalnej jednostki wskazuje liczbę pozycji w niej zgrupowanych (na podstawie wyników filtra głównego). Po kliknięciu jednostki liczby przy innych dostępnych jednostkach są wyświetlane ze znakiem plus, wskazującym, ile pozycji można dodać do bieżącego wyboru. Jednostki bez pozycji nie są wyświetlane, chyba że zostały wcześniej wybrane w podfiltrze.

2 Wyzwalacze

Przegląd

Listę wyzwalaczy dla hosta można otworzyć z poziomu *Zbieranie danych* → *Hosty* , klikając *Wyzwalacze* dla odpowiedniego hosta.



Wyświetlane dane:

Column	Description
Severity	Poziom ważności wyzwalacza jest wyświetlany zarówno nazwą, jak i kolorem tła komórki.
Value	Wyświetlana jest wartość wyzwalacza: **OK** - wyzwalacz jest w stanie OK **PROBLEM** - wyzwalacz jest w stanie Problem
Host	Host wyzwalacza. Kliknięcie nazwy hosta otwiera formularz konfiguracji. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze zaznaczono wiele hostów lub nie zaznaczono żadnego hosta.
Name	Nazwa wyzwalacza, wyświetlana jako niebieski link do szczegółów wyzwalacza. Kliknięcie linku z nazwą wyzwalacza otwiera formularz konfiguracji. Jeśli host, do którego należy wyzwalacz, jest częścią szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary link. Kliknięcie linku do szablonu otworzy listę wyzwalaczy na poziomie szablonu. Jeśli wyzwalacz został utworzony na podstawie prototypu wyzwalacza, jego nazwa jest poprzedzona nazwą reguły wykrywania niskiego poziomu, wyświetlaną na pomarańczowo. Kliknięcie nazwy reguły wykrywania otwiera listę prototypów wyzwalaczy.
Operational data	Definicja danych operacyjnych wyzwalacza, zawierająca dowolne ciągi znaków i makra, które zostaną dynamicznie rozwiązane w *Monitoring* → *Problemy*.
Expression	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia dotycząca hosta i pozycji jest wyświetlana jako link prowadzący do formularza konfiguracji pozycji.
Status	Wyświetlany jest status wyzwalacza - *Enabled*, *Disabled* lub *Unknown*. Klikając status, można go ręcznie zmienić - z Enabled na Disabled (i z powrotem); z Unknown na Disabled (i z powrotem). Problemy wyłączonego wyzwalacza nie są już wyświetlane we frontend, ale nie są usuwane. Wykryte wyzwalacze, które zostały utracone, są oznaczone ikoną informacyjną. Tekst podpowiedzi zawiera szczegóły dotyczące ich statusu.
Info	Jeśli wszystko działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą ”i”. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.
Tags	Jeśli wyzwalacz zawiera tagi, w tej kolumnie wyświetlane są nazwa i wartość tagu.

Aby skonfigurować nowy wyzwalacz, kliknij przycisk *Create trigger* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

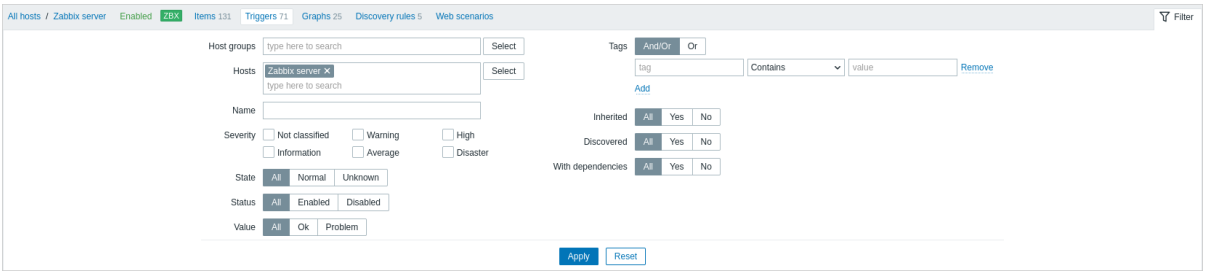
- *Włącz* - zmienia status wyzwalacza na *Włączony*.
- *Wyłącz* - zmienia status wyzwalacza na *Wyłączony*.
- *Kopiuj* - kopiuje wyzwalacze do innych hostów lub szablonów.
- *Masowa aktualizacja* - aktualizuje kilka właściwości dla wielu wyzwalaczy jednocześnie.
- *Usuń* - usuwa wyzwalacze.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wyzwalaczami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko wyzwalacze, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



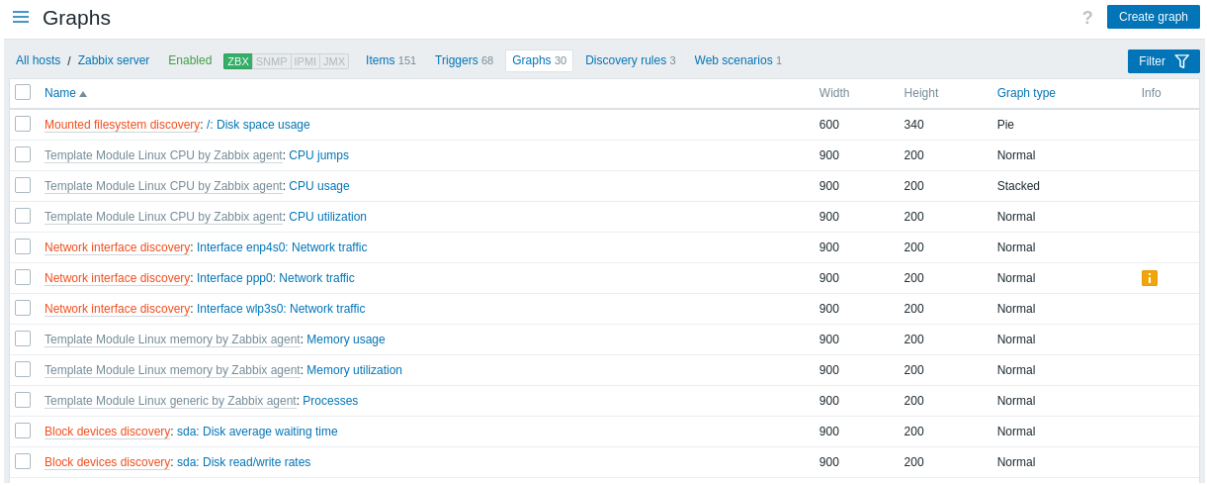
Parameter	Description
Host groups	Filtrowanie według jednej lub kilku grup hostów. Wskazanie nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów. Nie można wybrać grup hostów zawierających wyłącznie szablony.
Hosts	Filtrowanie według jednego lub kilku hostów. Jeśli powyżej wybrano już grupy hostów, wybór hostów jest ograniczony do tych grup.
Name	Filtrowanie według nazwy wyzwalacza.
Severity	Wybierz, aby filtrować według jednej lub kilku wartości ważności wyzwalacza.
State	Filtrowanie według stanu wyzwalacza.
Status	Filtrowanie według statusu wyzwalacza.
Value	Filtrowanie według wartości wyzwalacza.
Tags	Filtrowanie według nazwy i wartości tagu wyzwalacza. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek Obsługiwane są makra i funkcje makr zarówno w polach nazwy tagu, jak i wartości tagu.
Inherited	Filtrowanie wyzwalaczy odziedziczonych (lub nieodziedziczonych) z szablonu.
Discovered	Filtrowanie wyzwalaczy wykrytych (lub niewykrytych) przez wykrywanie niskiego poziomu.
With dependencies	Filtrowanie wyzwalaczy z zależnościami (lub bez zależności).

3 Wykresy

Przegląd

Do listy własnych wykresów dla hosta można przejść z poziomu *Zbieranie danych* → *Hosty*, klikając *Wykresy* przy odpowiednim hoście.

Wyświetlana jest lista istniejących wykresów.



Wyświetlane dane:

Kolumna	Opis
Host	Host wykresu. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele hostów lub nie wybrano żadnego hosta.
Nazwa	Nazwa własnego wykresu, wyświetlana jako niebieski link do szczegółów wykresu. Kliknięcie linku z nazwą wykresu otwiera formularz konfiguracji wykresu. Jeśli wykres hosta należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary link. Kliknięcie linku szablonu otworzy listę wykresów na poziomie szablonu. Jeśli wykres został utworzony z prototypu wykresu, jego nazwę poprzedza nazwa reguły wykrywania niskiego poziomu, wyświetlana na pomarańczowo. Kliknięcie nazwy reguły wykrywania otwiera listę prototypów wykresów.
Szerokość	Wyświetlana jest szerokość wykresu.
Wysokość	Wyświetlana jest wysokość wykresu.
Typ wykresu	Wyświetlany jest typ wykresu — *Normalny*, *Skumulowany*, *Kołowy* lub *Rozsunięty*.
Informacje	Jeśli wykres działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nowy wykres, kliknij przycisk *Utwórz wykres* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Kopiuj* — kopiowanie wykresów do innych hostów lub szablonów
- *Usuń* — usuwanie wykresów

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich wykresach, a następnie kliknij wymagany przycisk.

Używanie filtra

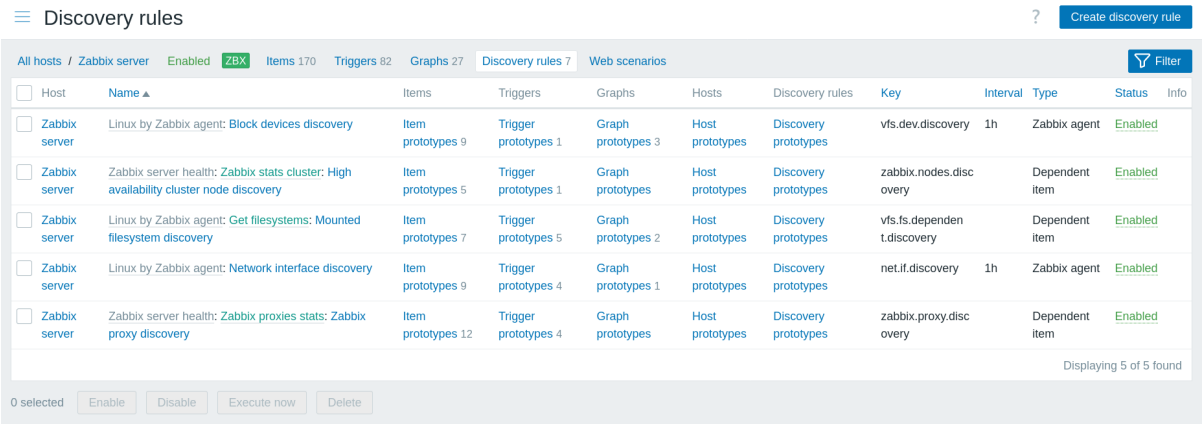
Możesz filtrować wykresy według grupy hostów i hosta. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

4 Reguły wykrywania

Przegląd

Lista reguł wykrywania niskiego poziomu dla hosta jest dostępna w *Zbieranie danych* → *Hosty* po kliknięciu *Wykrywanie* dla odpowiedniego hosta.

Wyświetlana jest lista istniejących reguł wykrywania niskiego poziomu. Możliwe jest również wyświetlenie wszystkich reguł wykrywania niezależnie od hosta lub wszystkich reguł wykrywania dla określonej grupy hostów poprzez zmianę ustawień filtra.



Wyświetlane dane:

Column	Description
Host	Wyświetlana jest widoczna nazwa hosta. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Jeśli widoczna nazwa hosta nie jest dostępna, wyświetlana jest techniczna nazwa hosta.
Name	Wyświetlana jest nazwa reguły jako niebieski link. Kliknięcie nazwy reguły otwiera formularz konfiguracji reguły wykrywania niskiego poziomu. Jeśli reguła wykrywania należy do szablonu, przed nazwą reguły wyświetlana jest nazwa szablonu jako szary link. Kliknięcie linku szablonu otwiera listę reguł na poziomie szablonu.
Items	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Triggers	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Graphs	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosts	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Key	Wyświetlany jest klucz pozycji używany do wykrywania.
Interval	Wyświetlana jest częstotliwość wykonywania wykrywania. *Uwaga*, wykrywanie można również wykonać natychmiast, naciskając przycisk *Wykonaj teraz* poniżej listy.
Type	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itp.).
Status	Wyświetlany jest status reguły wykrywania — *Włączona*, *Wyłączona* lub *Nieobsługiwana*. Kliknięcie statusu umożliwia jego zmianę — z Włączona na Wyłączona (i z powrotem) oraz z Nieobsługiwana na Wyłączona (i z powrotem).
Info	Jeśli wszystko działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i". Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nową regułę wykrywania niskiego poziomu, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

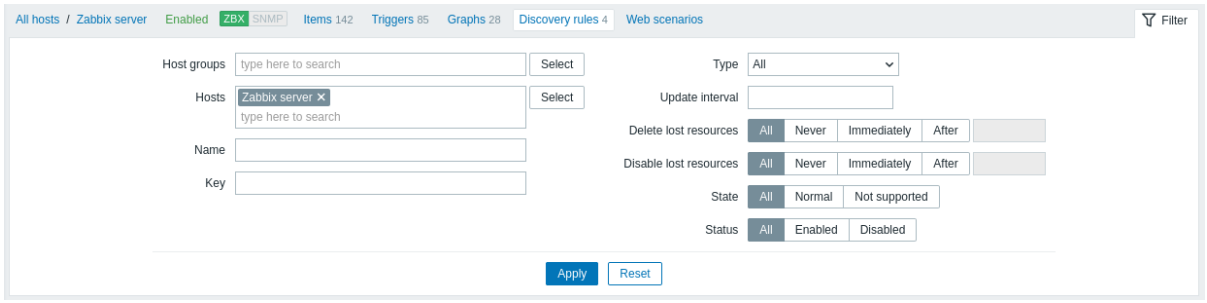
- *Włącz* - zmień status reguły wykrywania niskiego poziomu na *Włączony*.
- *Wyłącz* - zmień status reguły wykrywania niskiego poziomu na *Wyłączony*.
- *Wykonaj teraz* - wykonaj wykrywanie na podstawie reguł wykrywania natychmiast. Zobacz więcej szczegółów. Należy pamiętać, że podczas natychmiastowego wykonywania wykrywania pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wynik nie będzie odzwierciedlał bardzo niedawnych zmian w konfiguracji reguł wykrywania.
- *Usuń* - usuń reguły wykrywania niskiego poziomu.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich regułach wykrywania, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Łącze *Filtr* jest dostępne nad listą reguł wykrywania. Po jego kliknięciu dostępny staje się filtr, w którym można filtrować reguły wykrywania według grup hostów, hosta, nazwy, klucza pozycji, typu pozycji i innych parametrów.

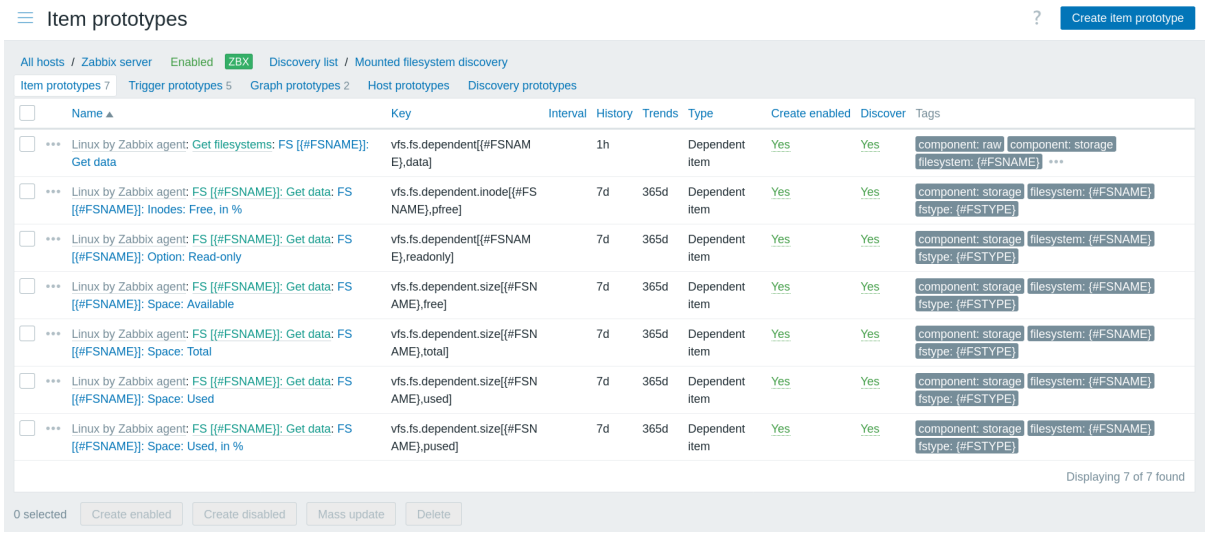


Parametr	Opis
Host groups	Filtruj według jednej lub większej liczby grup hostów. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów.
Hosts	Filtruj według jednego lub większej liczby hostów.
Name	Filtruj według nazwy reguły wykrywania.
Key	Filtruj według klucza pozycji wykrywania.
Type	Filtruj według typu pozycji wykrywania.
Update interval	Filtruj według interwału aktualizacji. Niedostępne dla pozycji typu Zabbix trapper i pozycji zależnych.
Delete lost resources	Filtruj według okresu *Delete lost resources*.
Disable lost resources	Filtruj według okresu *Disable lost resources*.
SNMP OID	Filtruj według SNMP OID. Dostępne tylko wtedy, gdy jako typ wybrano *SNMP agent*.
State	Filtruj według stanu reguły wykrywania (Wszystkie/Normalny/Nieobsługiwany).
Status	Filtruj według statusu reguły wykrywania (Wszystkie/Włączone/Wyłączone).

1 Prototypy pozycji

Przegląd

W tej sekcji wyświetlane są prototypy pozycji reguły wykrywania niskiego poziomu na hoście. Prototypy pozycji stanowią podstawę rzeczywistych pozycji hosta, które są tworzone podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Column	Description
Name	Nazwa prototypu pozycji, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu pozycji. Jeśli prototyp pozycji należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą reguły jako szary link. Kliknięcie linku szablonu otwiera listę prototypów pozycji na poziomie szablonu. Jeśli prototyp pozycji jest typu „Pozycja zależna”, nazwa pozycji nadrzędnej jest wyświetlana przed nazwą prototypu pozycji na zielono.
Key	Wyświetlany jest klucz prototypu pozycji.
Interval	Wyświetlana jest częstotliwość sprawdzania.
History	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trends	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Type	Wyświetlany jest typ prototypu pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itp.).
Create enabled	Utwórz pozycję na podstawie tego prototypu jako: **Yes** - włączoną **No** - wyłączoną. Możesz przełączać między „Yes” i „No”, klikając je.
Discover	Wykrywaj pozycję na podstawie tego prototypu: **Yes** - wykrywaj **No** - nie wykrywaj. Możesz przełączać między „Yes” i „No”, klikając je.
Tags	Wyświetlane są tagi prototypu pozycji.

Aby skonfigurować nowy prototyp pozycji, kliknij przycisk *Create item prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

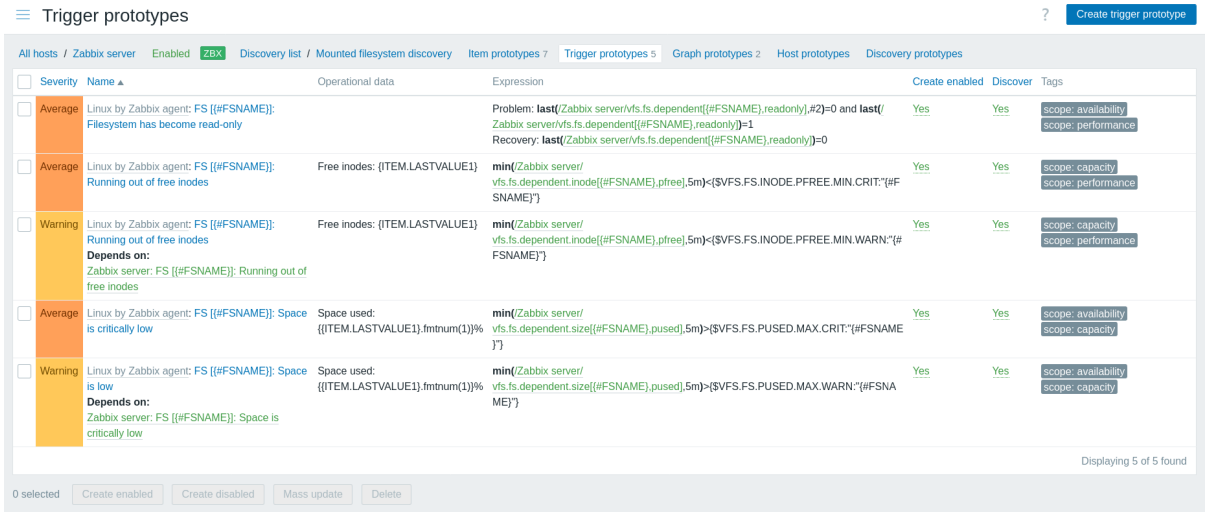
- *Utwórz włączone* - utwórz te prototypy pozycji jako *Włączone*
- *Utwórz wyłączone* - utwórz te prototypy pozycji jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy pozycji
- *Usuń* - usuń te prototypy pozycji

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach pozycji, a następnie kliknij wymagany przycisk.

2 Prototypy wyzwalaczy

Przegląd

W tej sekcji wyświetlane są prototypy wyzwalaczy reguły wykrywania niskiego poziomu na hoście. Prototypy wyzwalaczy stanowią podstawę rzeczywistych wyzwalaczy hosta, które są tworzone podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wyzwalacza, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wyzwalacza. Jeśli prototyp wyzwalacza należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wyzwalaczy na poziomie podłączonego szablonu.
Dane operacyjne	Wyświetlany jest format danych operacyjnych wyzwalacza, zawierający dowolne ciągi znaków i makra, które będą dynamicznie rozwijane w *Monitoring* → *Problemy*.
Twórz jako włączone	Utwórz wyzwalacz na podstawie tego prototypu jako: **Tak** - włączony **Nie** - wyłączony. Możesz przełączać między „Tak” i „Nie”, klikając je.
Odkrywaj	Odkrywaj wyzwalacz na podstawie tego prototypu: **Tak** - odkrywaj **Nie** - nie odkrywaj. Możesz przełączać między „Tak” i „Nie”, klikając je.
Tagi	Wyświetlane są tagi prototypu wyzwalacza.

Aby skonfigurować nowy prototyp wyzwalacza, kliknij przycisk *Utwórz prototyp wyzwalacza* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

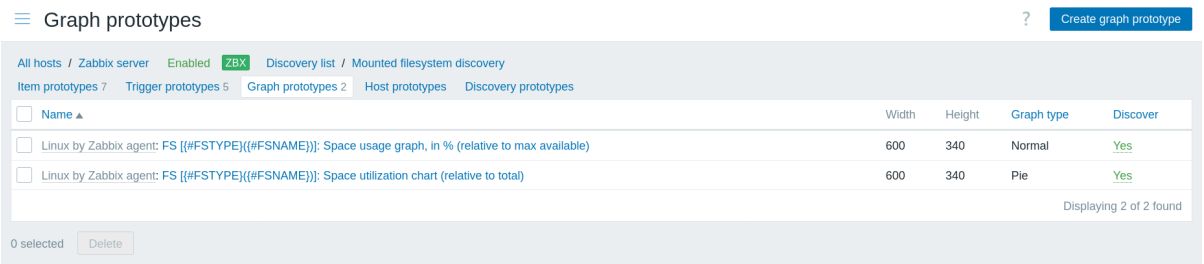
- *Create enabled* - utwórz te wyzwalacze jako *Enabled*
- *Create disabled* - utwórz te wyzwalacze jako *Disabled*
- *Mass update* - masowo zaktualizuj te prototypy wyzwalaczy
- *Delete* - usuń te prototypy wyzwalaczy

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wyzwalaczy, a następnie kliknij wymagany przycisk.

3 Prototypy wykresów

Przegląd

W tej sekcji wyświetlane są prototypy wykresów reguły wykrywania niskiego poziomu na hoście. Prototypy wykresów stanowią podstawę rzeczywistych wykresów hosta, które są tworzone podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wykresu, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykresu. Jeśli prototyp wykresu należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykresów na poziomie podłączonego szablonu.
Szerokość	Wyświetlana jest szerokość prototypu wykresu.
Wysokość	Wyświetlana jest wysokość prototypu wykresu.
Typ	Wyświetlany jest typ prototypu wykresu — *Normalny*, *Skumulowany*, *Kołowy* lub *Rozsunięty*.
Odkrywaj	Odkrywaj wykres na podstawie tego prototypu: **Tak** — odkrywaj **Nie** — nie odkrywaj. Możesz przełączać między „Tak” i „Nie”, klikając je.

Aby skonfigurować nowy prototyp wykresu, kliknij przycisk *Utwórz prototyp wykresu* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

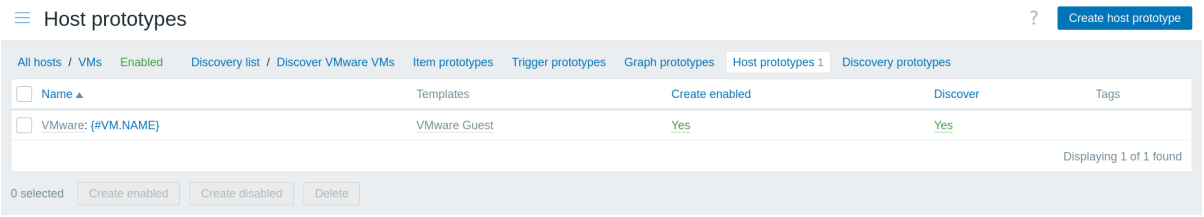
- *Usuń* - usuń te prototypy wykresów

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wykresów, a następnie kliknij wymagany przycisk.

4 Prototypy hostów

Przegląd

W tej sekcji wyświetlane są prototypy hostów reguły wykrywania niskiego poziomu na hoście. Prototypy hostów stanowią podstawę rzeczywistych hostów, które są tworzone podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu hosta, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu hosta. Jeśli prototyp hosta należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów hostów na poziomie podłączonego szablonu.
Szablony	Wyświetlane są szablony prototypu hosta.
Twórz włączone	Utwórz host na podstawie tego prototypu jako: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać między „Tak” i „Nie”, klikając je.
Odkrywaj	Odkrywaj host na podstawie tego prototypu: **Tak** - odkrywaj **Nie** - nie odkrywaj. Możesz przełączać między „Tak” i „Nie”, klikając je.
Tagi	Wyświetlane są tagi prototypu hosta.

Aby skonfigurować nowy prototyp hosta, kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

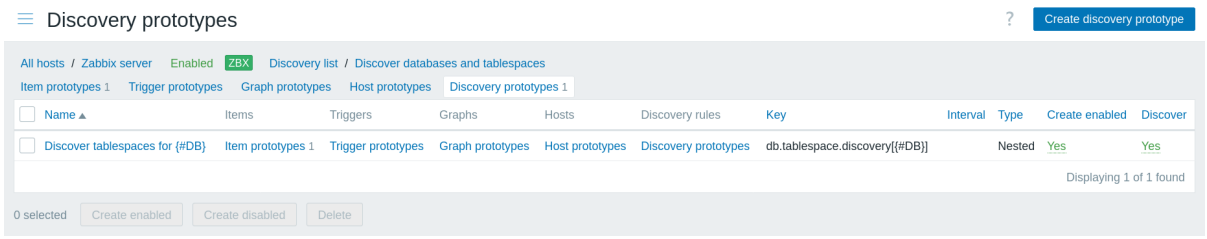
- *Utwórz włączone* - utwórz te hosty jako *Włączone*
- *Utwórz wyłączone* - utwórz te hosty jako *Wyłączone*
- *Usuń* - usuń te prototypy hostów

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach hostów, a następnie kliknij wymagany przycisk.

5 Prototypy wykrywania

Przegląd

W tej sekcji wyświetlane są prototypy wykrywania reguły wykrywania niskiego poziomu na hoście. Prototypy wykrywania są zagnieżdżonymi regułami wykrywania nadrzędnej reguły wykrywania.



Wyświetlane dane:

Column	Description
Name	Nazwa prototypu wykrywania, wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykrywania. Jeśli prototyp wykrywania należy do podłączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykrywania na poziomie podłączonego szablonu.
Items	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Triggers	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Graphs	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosts	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Discovery rules	Wyświetlany jest link do listy prototypów wykrywania. Liczba istniejących prototypów wykrywania jest wyświetlana na szaro.
Key	Wyświetlany jest klucz pozycji używany do wykrywania.
Interval	Wyświetlana jest częstotliwość wykonywania wykrywania. *Uwaga*: wykrywanie można również wykonać natychmiast, naciskając przycisk *Execute now* poniżej listy.
Type	Wyświetlany jest typ pozycji używany do wykrywania (Zabbix agent, SNMP agent itp.).
Create enabled	Wyświetlany jest status tworzenia prototypu wykrywania — tworzenie włączone (*Yes*) lub tworzenie wyłączone (*No*). Klikając status, można go zmienić — z Yes na No (i z powrotem).
Discover	Wyświetlany jest status wykrywania prototypu wykrywania — wykrywaj (*Yes*) lub nie wykrywaj (*No*). Klikając status, można go zmienić — z Yes na No (i z powrotem).

Aby skonfigurować nowy prototyp wykrywania, kliknij przycisk *Create discovery prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te prototypy wykrywania jako *Włączone*
- *Utwórz wyłączone* - utwórz te prototypy wykrywania jako *Wyłączone*
- *Usuń* - usuń te prototypy wykrywania

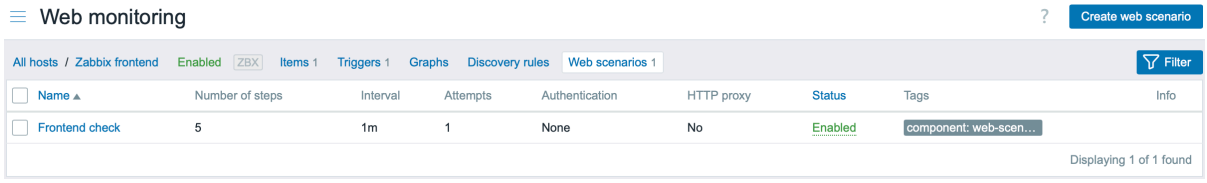
Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach wykrywania, a następnie kliknij wymagany przycisk.

5 Scenariusze webowe

Przegląd

Lista scenariuszy WWW dla hosta jest dostępna z poziomu *Zbieranie danych → Hosty*, po kliknięciu *WWW* dla odpowiedniego hosta.

Wyświetlana jest lista istniejących scenariuszy WWW.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa scenariusza WWW. Kliknięcie nazwy scenariusza WWW otwiera formularz konfiguracji scenariusza WWW. Jeśli scenariusz WWW hosta należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą scenariusza WWW jako szary odnośnik. Kliknięcie odnośnika szablonu otworzy listę scenariuszy WWW na poziomie szablonu.
Liczba kroków	Liczba kroków zawartych w scenariuszu.
Interwał aktualizacji	Jak często scenariusz jest wykonywany.
Próby	Liczba prób wykonywania kroków scenariusza WWW.
Uwierzytelnianie	Wyświetlana jest metoda uwierzytelniania — Basic, NTLM, Kerberos, Digest lub None.
HTTP proxy	Wyświetla HTTP proxy lub „Nie”, jeśli nie jest używany.
Status	Wyświetlany jest status scenariusza WWW — *Włączony* lub *Wyłączony*. Kliknięcie statusu pozwala go zmienić.
Tagi	Wyświetlane są tagi scenariusza WWW. Można wyświetlić maksymalnie trzy tagi (pary nazwa:wartość). Jeśli tagów jest więcej, wyświetlany jest odnośnik „...”, który po najechaniu kursorem pozwala zobaczyć wszystkie tagi.
Informacje	Jeśli wszystko działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nowy scenariusz WWW, kliknij przycisk *Utwórz scenariusz WWW* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

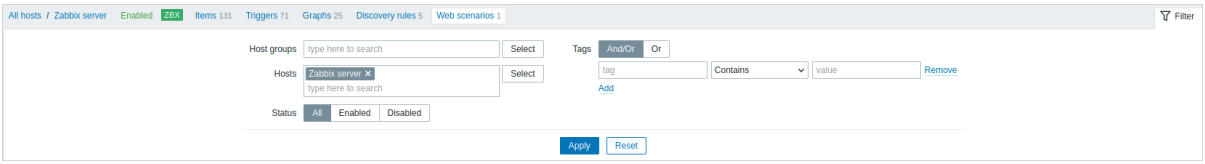
- *Włącz* - zmień status scenariusza na *Włączony*
- *Wyłącz* - zmień status scenariusza na *Wyłączony*
- *Wyczyść historię i trendy* - wyczyść dane historii i trendów dla scenariuszy
- *Usuń* - usuń scenariusze WWW

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich scenariuszach WWW, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te scenariusze, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Łącze *Filtr* jest dostępne nad listą scenariuszy WWW. Kliknięcie go otworzy filtr, w którym można określić żądane kryteria filtrowania.



Parametr	Opis
Grupy hostów	Filtruj według jednej lub większej liczby grup hostów. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów.
Hosty	Filtruj według jednego lub większej liczby hostów. Jeśli powyżej wybrano już grupy hostów, wybór hostów jest ograniczony do tych grup.
Status	Filtruj według statusu scenariusza WWW.

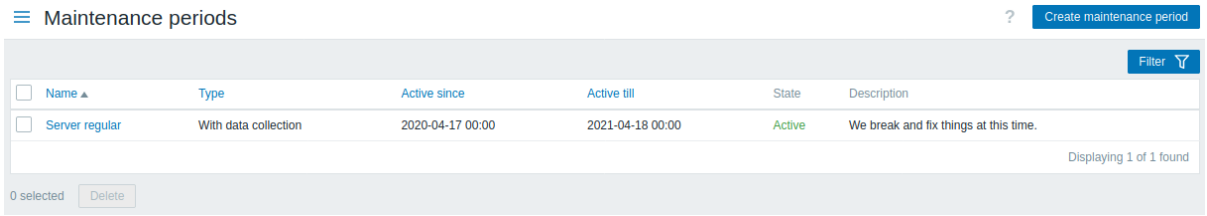
Parametr	Opis
Tagi	Filtruj według nazwy i wartości tagu scenariusza WWW. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** - wystarczy, że spełniony będzie jeden warunek.

5 Konserwacja

Przegląd

W sekcji *Zbieranie danych* → *Konserwacja* użytkownicy mogą konfigurować i utrzymywać okresy konserwacji dla hostów.

Wyświetlana jest lista istniejących okresów konserwacji wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa okresu konserwacji. Kliknięcie nazwy okresu konserwacji otwiera formularz konfiguracji okresu konserwacji.
Typ	Wyświetlany jest typ konserwacji: *Ze zbieraniem danych* lub *Bez zbierania danych*
Aktywny od	Data i godzina, od których wykonywanie okresów konserwacji staje się aktywne. Uwaga: Ten czas nie aktywuje okresu konserwacji; okresy konserwacji należy ustawić osobno.
Aktywny do	Data i godzina, o której wykonywanie okresów konserwacji przestaje być aktywne.
Stan	Stan okresu konserwacji: **Zbliża się** - wkrótce stanie się aktywny **Aktywny** - jest aktywny **Wygasły** - nie jest już aktywny
Opis	Wyświetlany jest opis okresu konserwacji.

Aby skonfigurować nowy okres konserwacji, kliknij przycisk *Utwórz okres konserwacji* w prawym górnym rogu.

Opcje masowej edycji

Przycisk poniżej listy udostępnia jedną opcję masowej edycji:

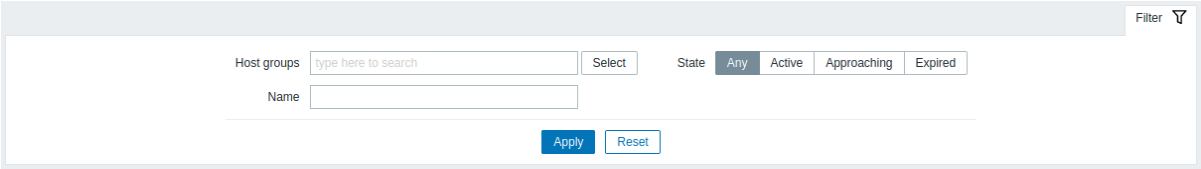
- *Usuń* — usuń okresy konserwacji

Aby użyć tej opcji, zaznacz pola wyboru przy odpowiednich okresach konserwacji i kliknij *Usuń*.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te okresy konserwacji, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane bez rozwijania makr.

Link *Filtr* jest dostępny nad listą okresów konserwacji. Po kliknięciu go dostępny staje się filtr, za pomocą którego można filtrować okresy konserwacji według grupy hostów, nazwy i stanu.



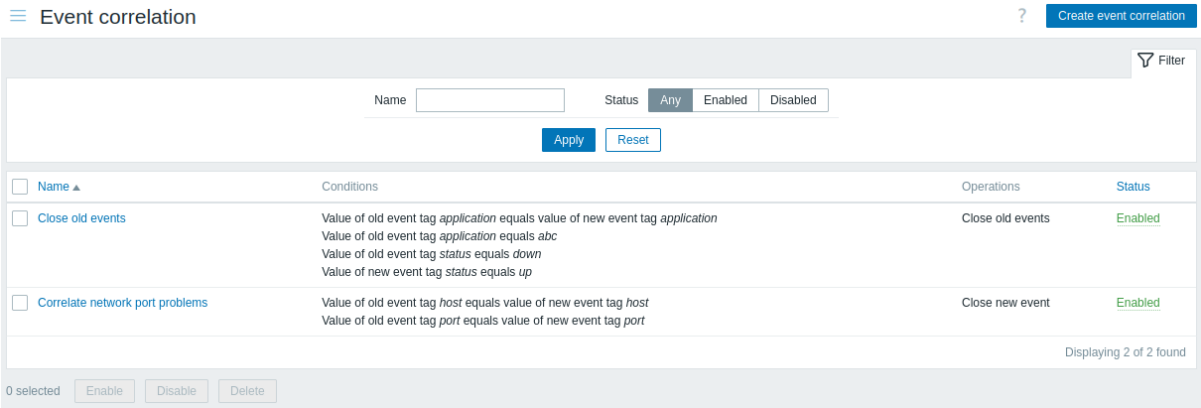
Obliczanie kolejek podczas konserwacji

Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły znajdują się w Obliczanie kolejek podczas konserwacji.

6 Korelacja zdarzeń

Przegląd

W sekcji *Zbieranie danych* → *Korelacja zdarzeń* użytkownicy mogą konfigurować i utrzymywać reguły globalnej korelacji dla zdarzeń Zabbix.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa reguły korelacji. Kliknięcie nazwy reguły korelacji otwiera formularz konfiguracji reguły.
Warunki	Wyświetlane są warunki reguły korelacji.
Operacje	Wyświetlane są operacje reguły korelacji.
Status	Wyświetlany jest status reguły korelacji — *Włączona* lub *Wyłączona*. Kliknięcie statusu umożliwia jego zmianę.

Aby skonfigurować nową regułę korelacji, kliknij przycisk *Utwórz korelację zdarzeń* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

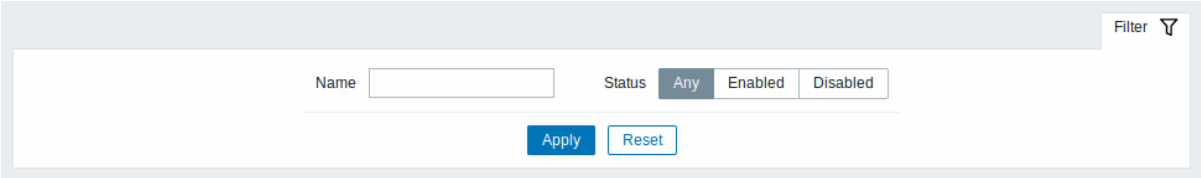
- *Włącz* - zmień stan reguły korelacji na *Włączony*
- *Wyłącz* - zmień stan reguły korelacji na *Wyłączony*
- *Usuń* - usuń reguły korelacji

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich regułach korelacji, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły korelacji, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą reguł korelacji. Po jego kliknięciu dostępny staje się filtr, za pomocą którego można filtrować reguły korelacji według nazwy i statusu.

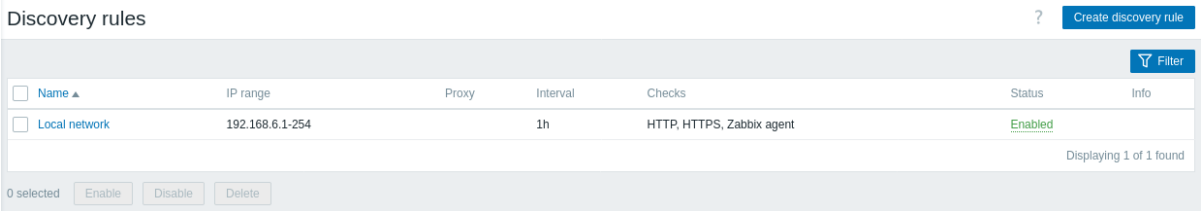


7 Wykrywanie

Przegląd

W sekcji *Zbieranie danych* → *Wykrywanie* użytkownicy mogą konfigurować i utrzymywać reguły wykrywania.

Wyświetlana jest lista istniejących reguł wykrywania wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa reguły wykrywania. Kliknięcie nazwy reguły wykrywania otwiera formularz konfiguracji reguły wykrywania.
Zakres IP	Wyświetlany jest zakres adresów IP używany do skanowania sieci.
Proxy	Wyświetlana jest nazwa proxy, jeśli wykrywanie jest wykonywane przez proxy.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania.
Kontrole	Wyświetlane są typy kontroli używane do wykrywania.
Status	Wyświetlany jest status reguły wykrywania — *Włączona* lub *Wyłączona*. Kliknięcie statusu umożliwia jego zmianę.
Informacja	Jeśli wszystko działa poprawnie, w tej kolumnie nic nie jest wyświetlane. W przypadku błędów wyświetlana jest czerwona ikona informacji z literą „i”. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nową regułę wykrywania, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

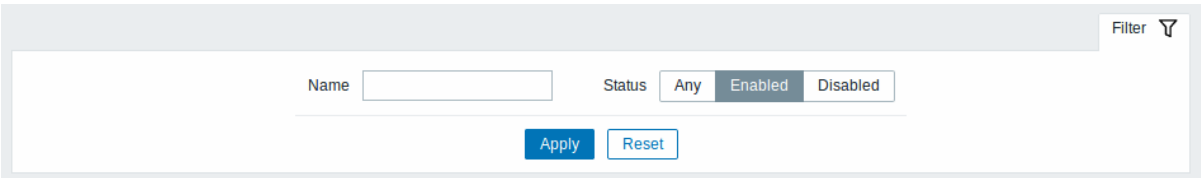
- *Włącz* - zmień status reguły wykrywania na *Włączony*
- *Wyłącz* - zmień status reguły wykrywania na *Wyłączony*
- *Usuń* - usuń reguły wykrywania

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich regułach wykrywania, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą reguł wykrywania. Po kliknięciu staje się dostępny filtr, za pomocą którego można filtrować reguły wykrywania według nazwy i statusu.



7 Alerty

Przegląd

To menu zawiera sekcje związane z konfiguracją alertów w Zabbix.

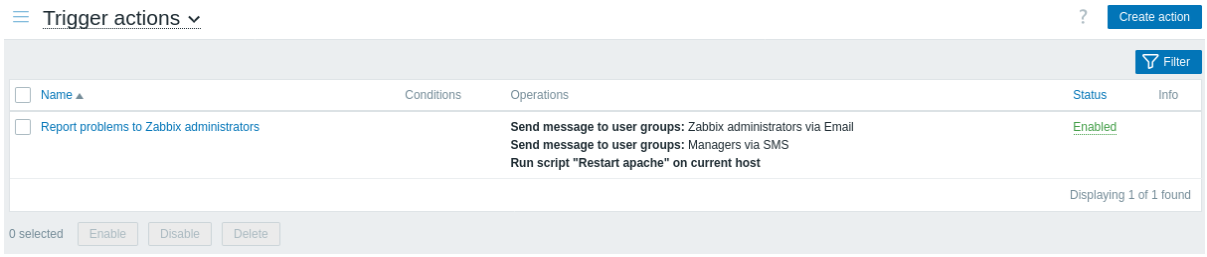
1 Akcje

Przegląd

W sekcji *Alerty* → *Akcje* użytkownicy mogą konfigurować i zarządzać akcjami.

Wyświetlane akcje to akcje przypisane do wybranego źródła zdarzeń (wyzwalacz, usługi, wykrywanie, autorejestracja, akcje wewnętrzne).

Aby przełączyć się na inne źródło zdarzeń, kliknij *Akcje* w sekcji menu *Alerty*. Możliwe jest również przełączanie się między źródłami za pomocą rozwijanej listy tytułu w lewym górnym rogu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa akcji. Kliknięcie nazwy akcji otwiera formularz konfiguracji akcji.
Warunki	Wyświetlane są warunki akcji.
Operacje	Wyświetlane są operacje akcji. Lista operacji wyświetla również typ nośnika (email, SMS lub skrypt) używanego do powiadomień, a także imię i nazwisko (w nawiasie po nazwie użytkownika) odbiorcy powiadomienia. Operacja akcji może być zarówno powiadomieniem, jak i zdalnym poleceniem, w zależności od wybranego typu operacji.
Status	Wyświetlany jest status akcji — *Włączona* lub *Wyłączona*. Kliknięcie statusu umożliwia jego zmianę. Zobacz sekcję Eskalacje, aby uzyskać więcej informacji o tym, co się stanie, jeśli akcja zostanie wyłączona podczas trwającej eskalacji.
Info	Jeśli wszystko działa poprawnie, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów, takich jak brakujące operacje akcji lub warunki po aktualizacji/wykrywaniu niskopoziomowym, wyświetlana jest ikona ostrzeżenia. Najedź kursorem na ikonę, aby zobaczyć podpowiedź z opisem błędu.

Aby skonfigurować nową akcję, kliknij przycisk *Utwórz akcję* w prawym górnym rogu.

Dla użytkowników bez uprawnień Super admin akcje są wyświetlane zgodnie z ustawieniami uprawnień. Oznacza to, że w niektórych przypadkach użytkownik bez uprawnień Super admin nie może wyświetlić pełnej listy akcji z powodu określonych ograniczeń uprawnień. Akcja jest wyświetlana użytkownikowi bez uprawnień Super admin, jeśli spełnione są następujące warunki:

- Użytkownik ma dostęp do odczytu i zapisu do grup hostów, hostów, szablonów oraz wyzwalaczy w warunkach akcji
- Użytkownik ma dostęp do odczytu i zapisu do grup hostów, hostów i szablonów w operacjach akcji, operacjach odzyskiwania i operacjach aktualizacji
- Użytkownik ma dostęp do odczytu do grup użytkowników i użytkowników w operacjach akcji, operacjach odzyskiwania i operacjach aktualizacji

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

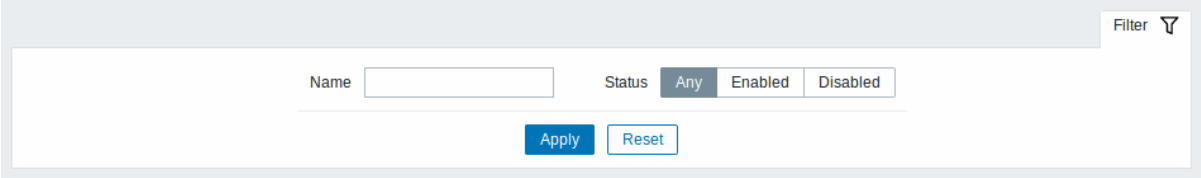
- *Włącz* - zmień status akcji na *Włączona*
- *Wyłącz* - zmień status akcji na *Wyłączona*
- *Usuń* - usuń akcje

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich akcjach, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te akcje, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Link *Filtr* jest dostępny nad listą akcji. Po kliknięciu udostępniony zostanie filtr, w którym można filtrować akcje według nazwy i statusu.



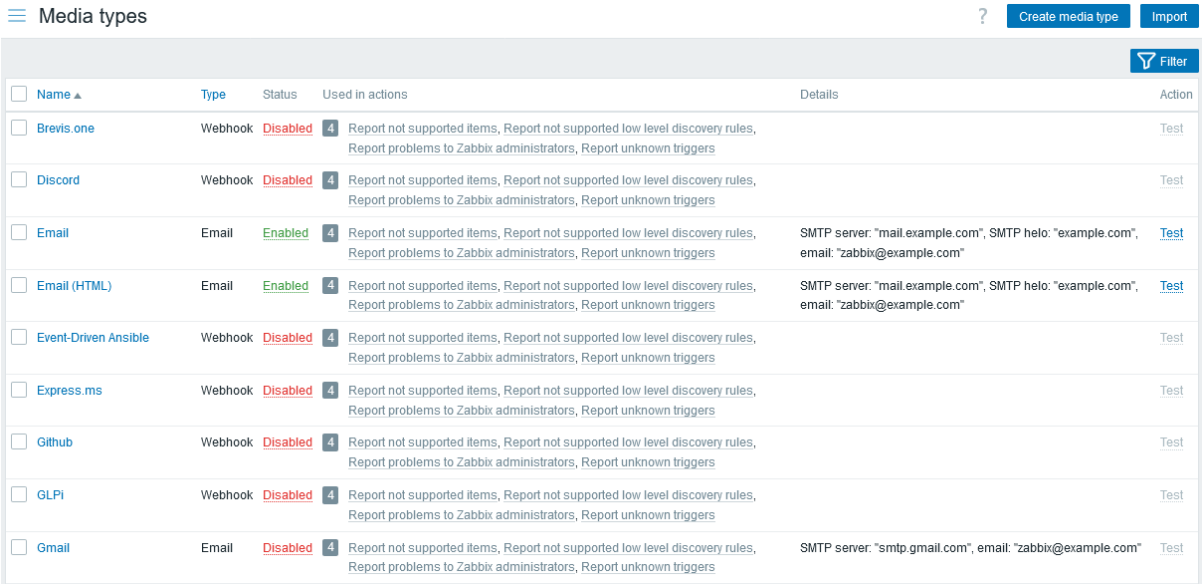
2 Typy mediów

Omówienie

W sekcji *Alerts* → *Media types* użytkownicy mogą konfigurować i zarządzać informacjami o typach mediów.

Informacje o typie medium zawierają ogólne instrukcje dotyczące używania medium jako kanału dostarczania powiadomień. Szczegółowe dane, takie jak poszczególne adresy e-mail, na które ma zostać wysłane powiadomienie, są przechowywane przy poszczególnych użytkownikach.

Wyświetlana jest lista istniejących typów mediów wraz z ich szczegółami.



Wyświetlane dane:

Column	Description
Name	Nazwa typu medium. Kliknięcie nazwy otwiera formularz konfiguracji typu medium.
Type	Wyświetlany jest typ medium (e-mail, SMS itp.).
Status	Wyświetlany jest status typu medium - *Enabled* lub *Disabled*. Klikając status, można go zmienić.
Used in actions	Wyświetlane są akcje, w których używany jest typ medium, poprzedzone łączną liczbą tych akcji. Kliknięcie nazwy akcji otwiera formularz konfiguracji akcji. Jeśli użytkownik nie ma uprawnień do akcji, nazwa nie jest klikalna.
Details	Wyświetlane są szczegółowe informacje o typie medium.
Actions	Dostępna jest następująca akcja: **Test** - kliknij, aby otworzyć formularz testowy, w którym można wprowadzić parametry typu medium (np. adres odbiorcy wraz z testowym tematem i treścią) oraz wysłać wiadomość testową, aby sprawdzić, czy skonfigurowany typ medium działa poprawnie. Zobacz także: testowanie typu medium dla Email, Webhook lub Script.

Aby skonfigurować nowy typ medium, kliknij przycisk *Create media type* w prawym górnym rogu.

Aby zaimportować typ medium, kliknij przycisk *Import* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

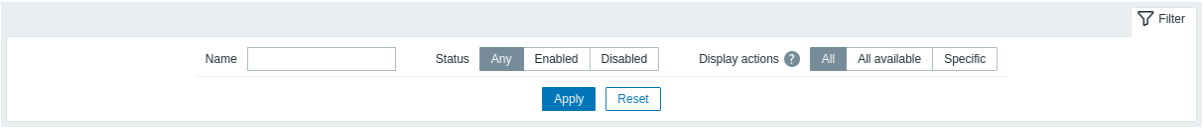
- *Włącz* - zmień status typu nośnika na *Włączony*
- *Wyłącz* - zmień status typu nośnika na *Wyłączony*
- *Eksportuj* - wyeksportuj typy nośników do pliku YAML, XML lub JSON
- *Usuń* - usuń typy nośników

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich typach nośników, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te typy mediów, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą typów mediów. Po jego kliknięciu dostępny staje się filtr, za pomocą którego można filtrować typy mediów według nazwy i statusu. Dodatkowo możesz użyć filtra, aby wyświetlać akcje w kolumnie *Używane w akcjach* na podstawie zakresu użycia ich typu mediów (zdefiniowanego przez parametr *Send to media type* w szczegółach operacji akcji).



3 Skrypty

Przegląd

W sekcji *Alerty* > *Skrypty* można konfigurować i utrzymywać globalne skrypty zdefiniowane przez użytkownika.

Globalne skrypty, w zależności od skonfigurowanego zakresu oraz uprawnień użytkownika, są dostępne do uruchomienia:

- z menu hosta w różnych miejscach frontendu (*Pulpit*, *Problemy*, *Najnowsze dane*, *Mapy* itp.)
- z menu zdarzenia
- mogą być uruchamiane jako operacja akcji

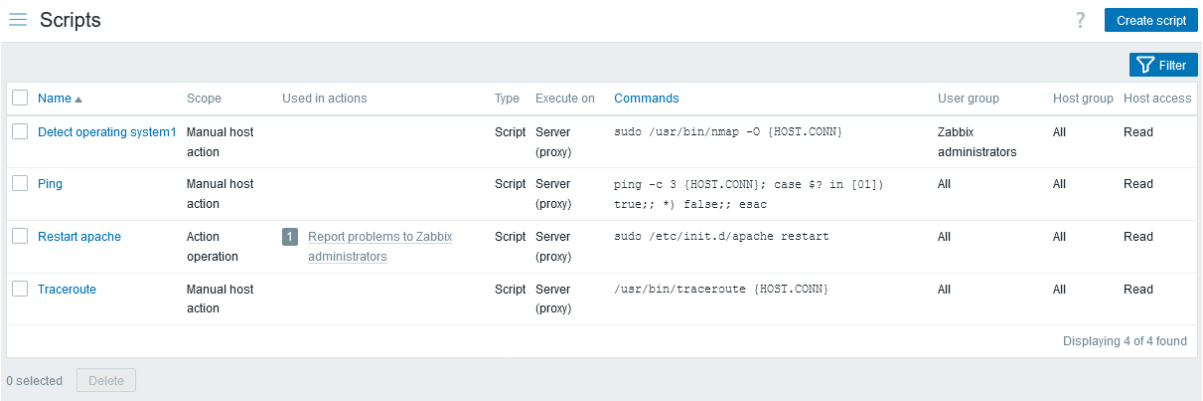
Skrypty są wykonywane na agencie Zabbix, serwerze Zabbix (proxy) lub tylko na serwerze Zabbix. Zobacz także Wykonywanie poleceń.

Zarówno na agencie Zabbix, jak i na proxy Zabbix zdalne skrypty są domyślnie wyłączone. Można je włączyć w następujący sposób:

- Dla zdalnych poleceń wykonywanych na agencie Zabbix:
  - dodając parametr AllowKey=system.run[<command>,*] dla każdego dozwolonego polecenia w konfiguracji agenta; * oznacza tryb wait i nowait;
- Dla zdalnych poleceń wykonywanych na proxy Zabbix:
  - **Ostrzeżenie: Nie jest wymagane włączanie zdalnych poleceń na proxy Zabbix, jeśli zdalne polecenia są wykonywane na agencie Zabbix monitorowanym przez proxy Zabbix.** Jeśli jednak wymagane jest wykonywanie zdalnych poleceń na proxy Zabbix, ustaw parametr *EnableRemoteCommands* na '1' w konfiguracji proxy.

Wykonywanie globalnych skryptów na serwerze Zabbix można wyłączyć, ustawiając `EnableGlobalScripts=0` w konfiguracji serwera. W nowych instalacjach, począwszy od Zabbix 7.0, wykonywanie globalnych skryptów na serwerze Zabbix jest domyślnie wyłączone.

Wyświetlana jest lista istniejących skryptów wraz z ich szczegółami.



Wyświetlane dane:

Column	Description
Name	Nazwa skryptu. Kliknięcie nazwy skryptu otwiera formularz konfiguracji skryptu.
Scope	Zakres skryptu — operacja akcji, ręczna akcja hosta lub ręczna akcja zdarzenia. To ustawienie określa, gdzie skrypt jest dostępny.
Used in actions	Wyświetlane są wszystkie akcje, w których używany jest skrypt, poprzedzone łączną liczbą tych akcji. Kliknięcie nazwy akcji otwiera formularz konfiguracji akcji. Jeśli użytkownik nie ma uprawnień do akcji, nazwa nie jest klikalna.
Type	Wyświetlany jest typ skryptu — polecenie *URL*, *Webhook*, *Script*, *SSH*, *Telnet* lub *IPMI*.
Execute on	Wyświetlana jest informacja, czy skrypt zostanie wykonany na agencie Zabbix, proxy Zabbix lub serwerze, czy tylko na serwerze Zabbix.
Commands	Wyświetlane są wszystkie polecenia, które mają zostać wykonane w ramach skryptu. Dla webhooków nic nie jest tutaj wyświetlane.
User group	Wyświetlana jest grupa użytkowników, dla której skrypt jest dostępny (lub *All* dla wszystkich grup użytkowników).
Host group	Wyświetlana jest grupa hostów, dla której skrypt jest dostępny (lub *All* dla wszystkich grup hostów).
Host access	Wyświetlany jest poziom uprawnień dla grupy hostów — *Read* lub *Write*. Tylko użytkownicy z wymaganym poziomem uprawnień będą mieli dostęp do wykonania skryptu.

Aby skonfigurować nowy skrypt, kliknij przycisk *Create script* w prawym górnym rogu.

Opcje masowej edycji

Przycisk pod listą oferuje jedną opcję masowej edycji:

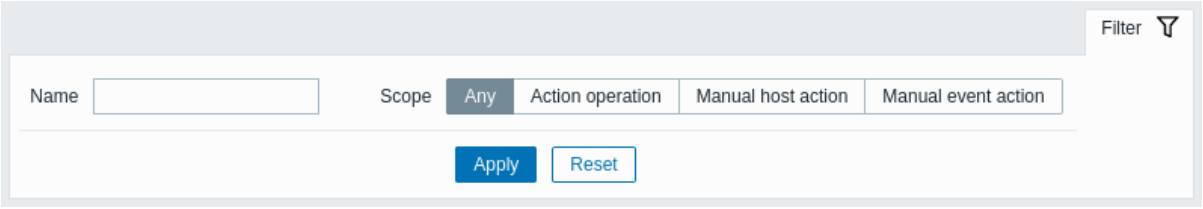
- *Usuń* - usuń skrypty

Aby użyć tej opcji, zaznacz pola wyboru przy odpowiednich skryptach i kliknij *Usuń*.

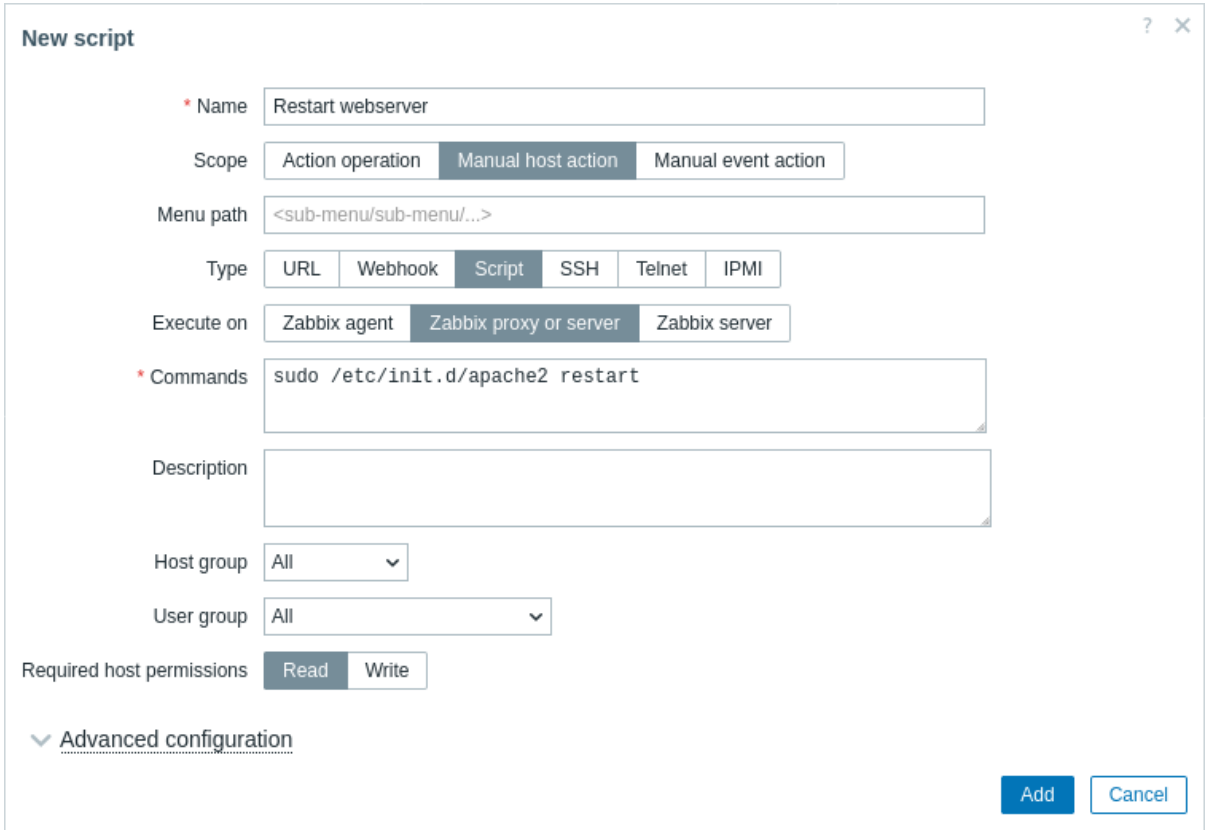
Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te skrypty, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą skryptów. Po jego kliknięciu pojawi się filtr, w którym można filtrować skrypty według nazwy i zakresu.



Konfigurowanie skryptu globalnego



Note:
Zaleca się używanie funkcji makr zamiast zwykłych makr w skryptach globalnych w celu zwiększenia bezpieczeństwa, ponieważ makra nie są automatycznie poprzedzane znakami ucieczki.

Atrybuty skryptu:

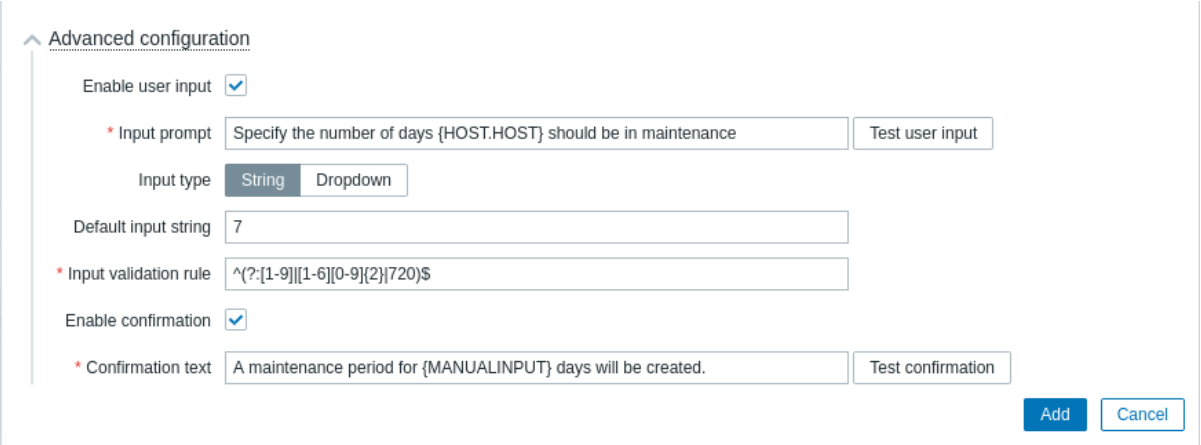
Parametr	Opis
Nazwa	Unikalna nazwa skryptu. Np. `Clear /tmp filesystem`
Zakres	Zakres skryptu — operacja akcji, ręczna akcja hosta lub ręczna akcja zdarzenia. To ustawienie określa, gdzie skrypt może być używany — odpowiednio w zdalnych poleceniach operacji akcji, z menu hosta lub z menu zdarzenia. Ustawienie zakresu na „Operacja akcji" powoduje, że skrypt jest dostępny dla wszystkich użytkowników mających dostęp do *Alerty* > *Akcje*. Jeśli skrypt jest faktycznie używany w akcji, jego zakresu nie można zmienić z „operacja akcji" na inny. **Obsługa makr** Zakres wpływa na zakres dostępnych makr. Na przykład makra związane z użytkownikiem ({USER.*}) są obsługiwane w skryptach, aby umożliwić przekazywanie informacji o użytkowniku, który uruchomił skrypt. Nie są one jednak obsługiwane, jeśli zakresem skryptu jest operacja akcji, ponieważ operacje akcji są wykonywane automatycznie. Makro {MANUALINPUT} pozwala określić dane wejściowe ręcznie w momencie wykonywania skryptu. Jest ono obsługiwane dla skryptów ręcznych akcji hosta i ręcznych akcji zdarzenia. Aby sprawdzić, które inne makra są obsługiwane, wyszukaj „Trigger-based notifications and commands/Trigger-based commands", „Manual host action scripts" oraz „Manual event action scripts" w tabeli obsługiwanych makr. Zwróć uwagę, że jeśli makro może zostać rozwinięte do wartości zawierającej spacje (na przykład nazwa hosta), należy pamiętać o użyciu cudzysłowów tam, gdzie to konieczne.
Ścieżka menu	Żądana ścieżka menu do skryptu. Na przykład `Default` lub `Default/` spowoduje wyświetlenie skryptu w odpowiednim katalogu. Menu mogą być zagnieżdżone, np. `Main menu/Sub menu1/Sub menu2`. Przy dostępie do skryptów przez menu hosta/zdarzenia w sekcjach monitorowania będą one uporządkowane zgodnie z podanymi katalogami. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Ręczna akcja hosta" lub „Ręczna akcja zdarzenia".

Parametr	Opis
Typ	Kliknij odpowiedni przycisk, aby wybrać typ skryptu: **URL**, **Webhook**, **Skrypt**, **SSH**, **Telnet** lub polecenie **IPMI**. Typ **URL** jest dostępny tylko wtedy, gdy jako *Zakres* wybrano „Ręczna akcja hosta” lub „Ręczna akcja zdarzenia”.
Typ skryptu: **URL**	
URL	Określ adres URL do szybkiego dostępu z menu hosta lub menu zdarzenia. Obsługiwane są makra oraz niestandardowe makra użytkownika. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Użyj makra {MANUALINPUT} w tym polu, aby móc określić dane wejściowe ręcznie w momencie wykonywania skryptu, na przykład: `http://{MANUALINPUT}/zabbix/zabbix.php?action=dashboard.view` Wartości makr nie mogą być kodowane w formacie URL.
Otwórz w nowym oknie	Określa, czy adres URL ma zostać otwarty w nowej karcie przeglądarki, czy w tej samej.
Typ skryptu: **Webhook**	
Parametry	Określ zmienne webhooka jako pary atrybut-wartość. Zobacz także: konfiguracja nośnika Webhook. Makra oraz niestandardowe makra użytkownika są obsługiwane w wartościach parametrów. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej).
Skrypt	Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Zobacz także: konfiguracja nośnika Webhook, Dodatkowe obiekty Javascript.
Limit czasu	Limit czasu wykonywania JavaScript (1-60s, domyślnie 30s). Obsługiwane są przyrostki czasu, np. 30s, 1m.
Typ skryptu: **Skrypt**	
Wykonaj na	Kliknij odpowiedni przycisk, aby wykonać skrypt powłoki na: **Zabbix agent** — skrypt zostanie wykonany przez Zabbix agent (jeśli pozycja system.run jest dozwolona) na hoście **Zabbix proxy lub serwer** — skrypt zostanie wykonany przez Zabbix proxy lub serwer — w zależności od tego, czy host jest monitorowany przez proxy czy serwer. Zostanie wykonany na proxy, jeśli jest to włączone przez parametr EnableRemoteCommands. Zostanie wykonany na serwerze, jeśli skrypty globalne są włączone przez parametr serwera EnableGlobalScripts. **Zabbix serwer** — skrypt zostanie wykonany tylko przez Zabbix serwer. Ta opcja nie będzie dostępna, jeśli skrypty globalne są wyłączone przez parametr serwera EnableGlobalScripts.
Polecenia	Wprowadź pełną ścieżkę do poleceń, które mają zostać wykonane w skrypcie. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Typ skryptu: **SSH**	
Metoda uwierzytelniania	Wybierz metodę uwierzytelniania — hasło lub klucz publiczny.
Nazwa użytkownika	Wprowadź nazwę użytkownika.
Hasło	Wprowadź hasło. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Hasło”.
Plik klucza publicznego	Wprowadź ścieżkę do pliku klucza publicznego. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Klucz publiczny”.
Plik klucza prywatnego	Wprowadź ścieżkę do pliku klucza prywatnego. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Klucz publiczny”.
Hasło klucza	Wprowadź hasło klucza. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Klucz publiczny”.
Port	Wprowadź port zdalnej usługi SSH na docelowym hoście, z którym połączy się Zabbix.
Polecenia	Wprowadź polecenia. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Typ skryptu: **Telnet**	
Nazwa użytkownika	Wprowadź nazwę użytkownika.
Hasło	Wprowadź hasło.
Port	Wprowadź port zdalnej usługi Telnet na docelowym hoście, z którym połączy się Zabbix.
Polecenia	Wprowadź polecenia. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.

Parametr	Opis
Typ skryptu: **IPMI**	
Polecenie	Wprowadź polecenie IPMI. Obsługa makr zależy od zakresu skryptu (patrz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Opis	Wprowadź opis skryptu.
Grupa hostów	Wybierz grupę hostów, dla której skrypt będzie dostępny (lub *Wszystkie* dla wszystkich grup hostów).
Grupa użytkowników	Wybierz grupę użytkowników, dla której skrypt będzie dostępny (lub *Wszystkie* dla wszystkich grup użytkowników). To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Ręczna akcja hosta” lub „Ręczna akcja zdarzenia”.
Wymagane uprawnienia do hosta	Wybierz poziom uprawnień dla grupy hostów — *Odczyt* lub *Zapis*. Tylko użytkownicy z wymaganym poziomem uprawnień będą mieli dostęp do wykonywania skryptu. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Ręczna akcja hosta” lub „Ręczna akcja zdarzenia”.
Konfiguracja zaawansowana	Kliknij nagłówek *Konfiguracja zaawansowana*, aby wyświetlić opcje konfiguracji zaawansowanej. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Ręczna akcja hosta” lub „Ręczna akcja zdarzenia”.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*:



Parametr	Opis
Włącz dane wejściowe użytkownika	Zaznacz pole wyboru, aby włączyć ręczne wprowadzanie danych przez użytkownika przed wykonaniem skryptu. Ręcznie wprowadzone dane zastąpią wartość makra {MANUALINPUT} w skrypcie. Zobacz także: Ręczne dane wejściowe użytkownika.
Monit wejściowy	Wprowadź własny tekst zachęcający do podania danych przez użytkownika. Ten tekst będzie wyświetlany nad polem wejściowym w wyskakującym oknie *Ręczne dane wejściowe*. Aby zobaczyć podgląd wyskakującego okna *Ręczne dane wejściowe*, kliknij *Testuj dane wejściowe użytkownika*. Podgląd umożliwia także sprawdzenie, czy ciąg wejściowy jest zgodny z regułą walidacji danych wejściowych (zobacz parametry poniżej). Obsługa makr i makr użytkownika zależy od zakresu skryptu (zobacz *Zakres* w ogólnych parametrach konfiguracji skryptu).
Typ danych wejściowych	Wybierz typ ręcznie wprowadzanych danych: **Ciąg znaków** — pojedynczy ciąg znaków; **Lista rozwijana** — wartość jest wybierana z wielu opcji listy rozwijanej.
Opcje listy rozwijanej	Wprowadź unikalne wartości dla listy rozwijanej danych wejściowych użytkownika jako listę rozdzielaną przecinkami. Aby uwzględnić pustą opcję na liście rozwijanej, dodaj dodatkowy przecinek na początku, w środku lub na końcu listy. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano „Lista rozwijana”.

Parametr	Opis
Domyślny ciąg wejściowy	Wprowadź domyślny ciąg dla danych wejściowych użytkownika (lub pozostaw puste). To pole będzie walidowane względem wyrażenia regularnego podanego w polu *Reguła walidacji danych wejściowych*. Wartość wprowadzona tutaj będzie domyślnie wyświetlana w wyskakującym oknie *Ręczne dane wejściowe*. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano „Ciąg znaków”.
Reguła walidacji danych wejściowych	Wprowadź wyrażenie regularne do walidacji ciągu danych wejściowych użytkownika. Obsługiwane są globalne wyrażenia regularne. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano „Ciąg znaków”.
Włącz potwierdzenie	Zaznacz pole wyboru, aby przed wykonaniem skryptu wyświetlić komunikat potwierdzający. Ta funkcja może być szczególnie przydatna w przypadku potencjalnie niebezpiecznych operacji (takich jak skrypt restartu) lub takich, które mogą zająć dużo czasu.
Tekst potwierdzenia	Wprowadź własny tekst potwierdzenia dla wyskakującego okna potwierdzenia włączonego za pomocą powyższego pola wyboru (na przykład *Zdalny system zostanie uruchomiony ponownie. Czy na pewno?*). Aby zobaczyć, jak tekst będzie wyglądał, kliknij *Testuj potwierdzenie* obok pola. Obsługiwane są makra oraz niestandardowe makra użytkownika. *Uwaga:* makra nie będą rozwijane podczas testowania komunikatu potwierdzenia.

Jeśli skonfigurowano zarówno ręczne dane wejściowe użytkownika, jak i komunikat potwierdzenia, zostaną one wyświetlone kolejno w następujących po sobie wyskakujących oknach.

Ręczne dane wejściowe użytkownika

Ręczne dane wejściowe użytkownika pozwalają podać własny parametr przy każdym uruchomieniu skryptu. Eliminuje to konieczność tworzenia wielu podobnych skryptów użytkownika różniących się tylko jednym parametrem.

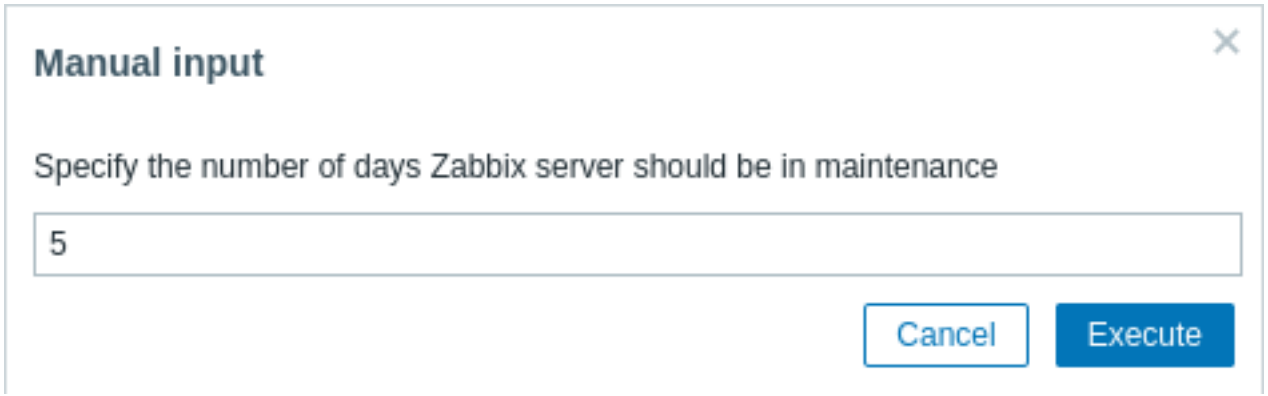
Na przykład podczas uruchamiania skryptu można chcieć podać inną liczbę całkowitą lub inny adres URL.

Aby włączyć ręczne dane wejściowe użytkownika:

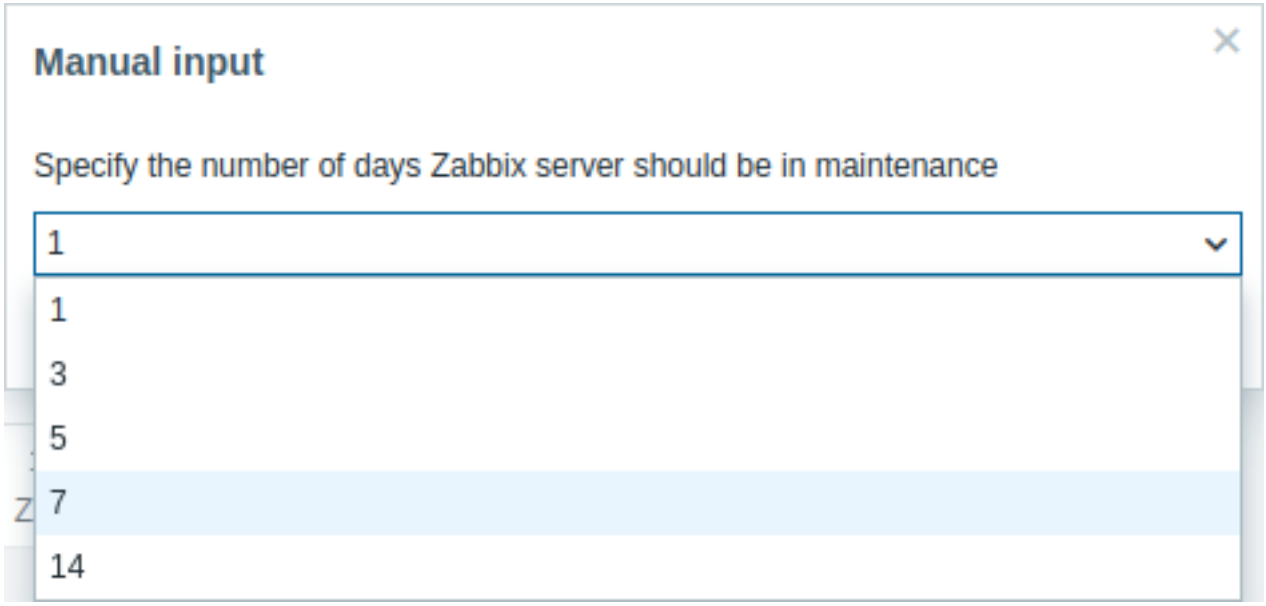
- użyj makra {MANUALINPUT} w skrypcie (polecenia, skrypt, parametr skryptu) tam, gdzie jest to wymagane; lub w polu URL w skryptach URL;
- w zaawansowanej konfiguracji skryptu włącz ręczne dane wejściowe użytkownika i skonfiguruj opcje wejściowe.

Po włączeniu danych wejściowych użytkownika, przed wykonaniem skryptu użytkownikowi zostanie wyświetlone wyskakujące okno *Ręczne dane wejściowe* z prośbą o podanie własnej wartości. Podana wartość zastąpi {MANUALINPUT} w skrypcie.

W zależności od konfiguracji użytkownik zostanie poproszony o wprowadzenie wartości tekstowej:



Lub o wybranie wartości z listy rozwijanej wcześniej zdefiniowanych opcji:



Ręczne dane wejściowe użytkownika są dostępne tylko dla skryptów, w których zakres to „Ręczna akcja hosta” lub „Ręczna akcja zdarzenia”.

Wykonywanie skryptu i wynik

Skrypty uruchamiane przez serwer Zabbix są wykonywane w kolejności opisanej na stronie wykonywania poleceń.

Wynik skryptu jest wyświetlany w oknie podręcznym, które pojawia się po uruchomieniu skryptu. Wartość zwracana przez skrypt to standardowe wyjście:

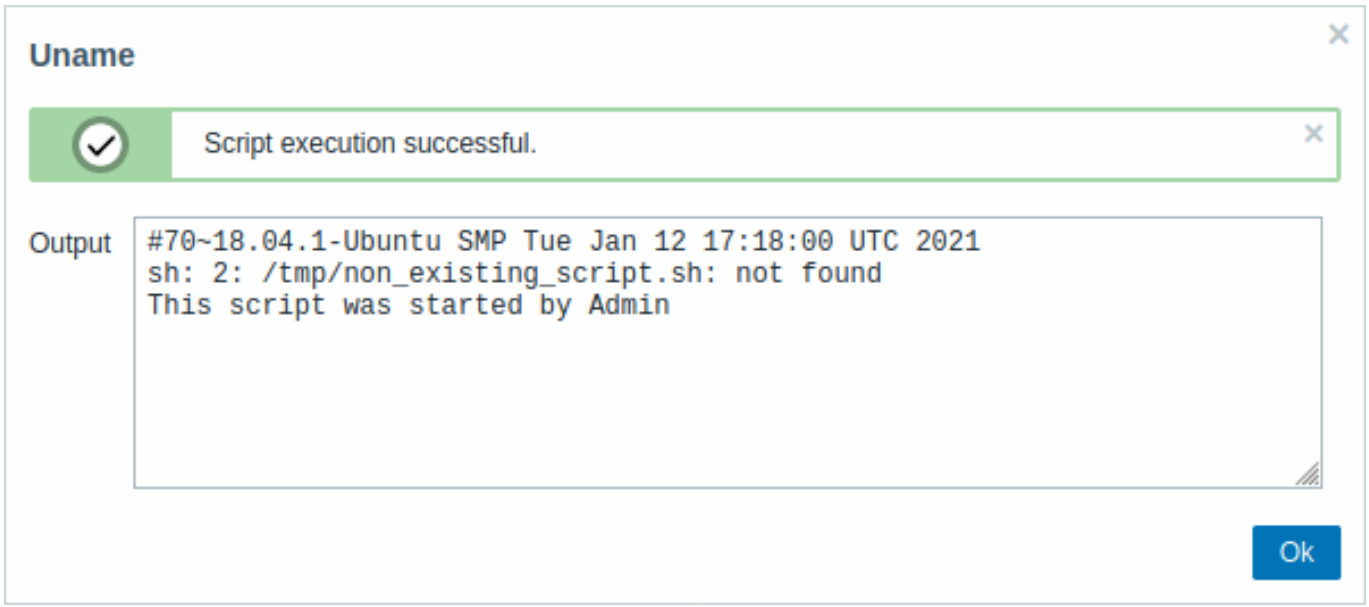
- Jeśli skrypt zakończy się pomyślnie (kod zakończenia 0), wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane).
- Jeśli skrypt zakończy się błędem (niezerowy kod zakończenia), wartość zwracana to standardowy błąd ograniczony do 2 KB.

Zabbix domyślnie nie przechowuje rozszerzonego wyjścia skryptu. Aby zachować pełne szczegóły wyjścia, można zaimplementować logowanie wewnątrz samego skryptu (np. przekierować wyjście do lokalnego pliku dziennika).

Należy pamiętać, że w przypadku skryptów wykonywanych na serwerze Zabbix lub proxy Zabbix obowiązują również limity bazy danych.

Poniżej znajduje się przykład skryptu i okna z wynikiem:

```
uname -v
/tmp/non_existing_script.sh
echo "Ten skrypt został uruchomiony przez {USER.USERNAME}"
```



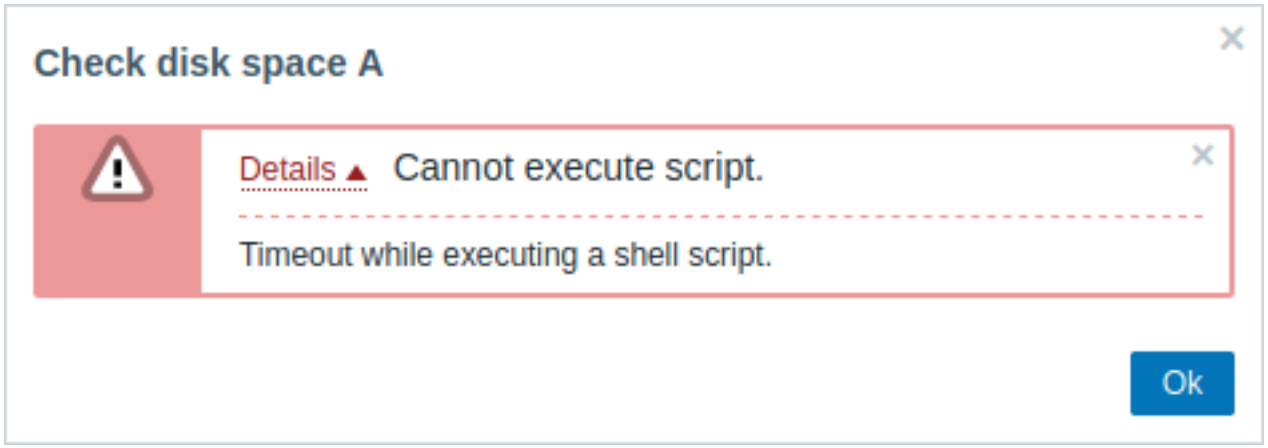
Wynik skryptu nie wyświetla samego skryptu.

Limit czasu skryptu w Zabbix agent

Możesz napotkać sytuację, w której podczas wykonywania skryptu wystąpi przekroczenie limitu czasu.

Poniżej przedstawiono przykład skryptu uruchamianego w Zabbix agent oraz okno z wynikiem:

```
sleep 5
df -h
```



Komunikat o błędzie w tym przypadku jest następujący:

```
Timeout while executing a shell script.
```

Aby uniknąć takich sytuacji, zaleca się zoptymalizowanie samego skryptu (w powyższym przykładzie „5") zamiast dostosowywania parametru `Timeout` w konfiguracji Zabbix agent oraz konfiguracji serwera Zabbix.

Jeśli parametr `Timeout` został zmieniony w konfiguracji Zabbix agent, pojawi się następujący komunikat o błędzie:

```
Get value from agent failed: ZBX_TCP_READ() timed out.
```

Oznacza to, że modyfikacja została wprowadzona w konfiguracji Zabbix agent, ale konieczna jest również zmiana parametru `Timeout` w konfiguracji serwera Zabbix.

Limit czasu skryptu na agentach aktywnych

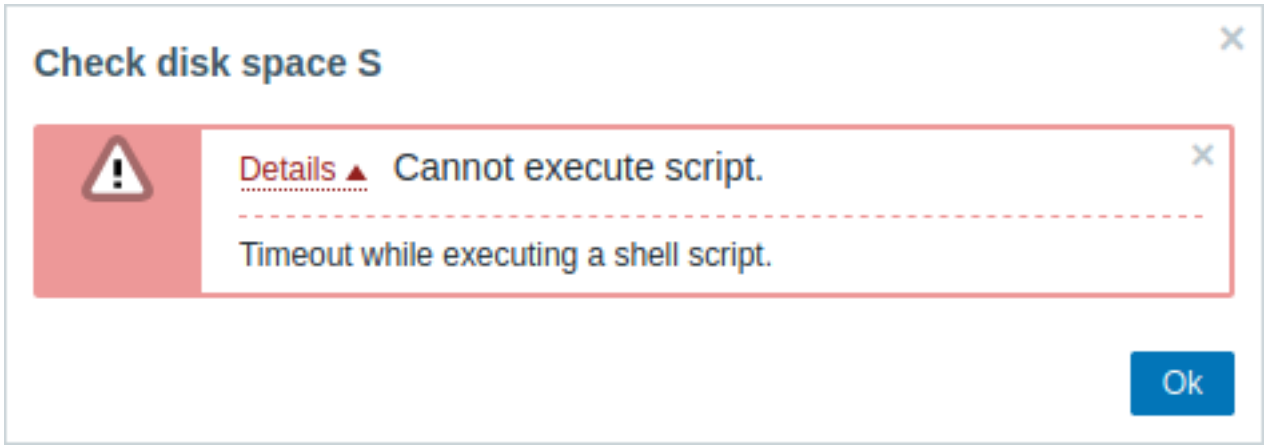
Możliwe jest wykonywanie skryptów tylko na agentach działających w trybie aktywnym. Gdy wykonanie skryptu zostanie wyzwolone przez operację akcji lub ręczne uruchomienie skryptu, polecenie jest dołączane do konfiguracji aktywnych kontroli i wykonywane, gdy aktywny agent je otrzyma.

Ręczne skrypty są wysyłane do aktywnego agenta wraz z limitem czasu serwera/proxy na wykonanie skryptu. Zwiększ domyślny limit czasu wykonywania skryptu dla serwera/proxy. Limit czasu musi być wyższy niż częstotliwość odświeżania aktywnych kontroli, w przeciwnym razie zostanie przekroczony, zanim aktywny agent otrzyma skrypt i będzie mógł zwrócić wynik.

Limit czasu skryptu na serwerze/proxy Zabbix

Poniżej przedstawiono przykład skryptu uruchamianego na serwerze Zabbix oraz okno z wynikiem:

```
sleep 11
df -h
```



Zaleca się również zoptymalizowanie samego skryptu (zamiast dostosowywania parametru `TrapperTimeout` do odpowiedniej wartości (w naszym przypadku > 11) poprzez modyfikację konfiguracji serwera Zabbix).

8 Użytkownicy

Przegląd

To menu zawiera sekcje związane z konfiguracją użytkowników w Zabbix. To menu jest dostępne tylko dla użytkowników typu SuperAdmin.

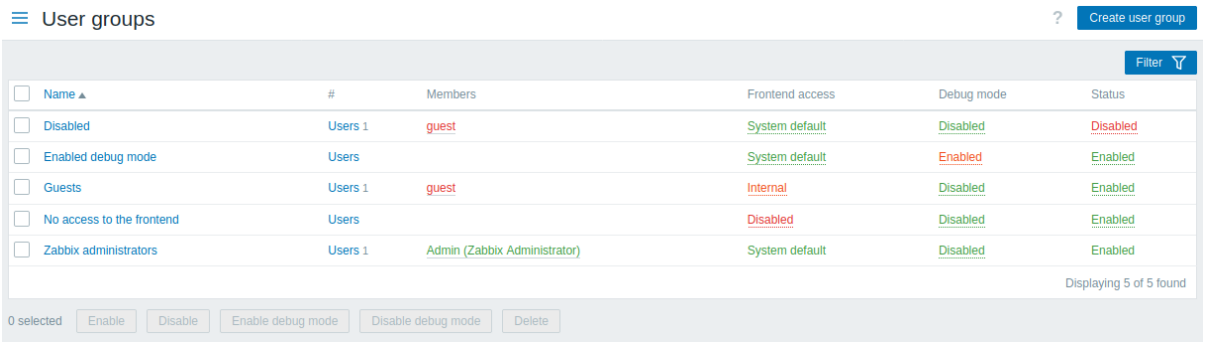
1 Grupy użytkowników

Przegląd

W sekcji *Użytkownicy* → *Grupy użytkowników* zarządza się grupami użytkowników systemu.

Grupy użytkowników

Wyświetlana jest lista istniejących grup użytkowników wraz z ich szczegółami.



Wyświetlane dane:

Column	Description
Name	Nazwa grupy użytkowników. Kliknięcie nazwy grupy użytkowników otwiera formularz konfiguracji grupy użytkowników.
#	Liczba użytkowników w grupie. Kliknięcie *Users* spowoduje wyświetlenie odpowiednich użytkowników odfiltrowanych na liście użytkowników.
Members	Nazwy użytkowników poszczególnych użytkowników w grupie użytkowników (z imieniem i nazwiskiem w nawiasach). Kliknięcie nazwy użytkownika otworzy formularz konfiguracji użytkownika. Użytkownicy z wyłączonych grup są wyświetlani na czerwono.
Frontend access	Wyświetlany jest poziom dostępu do frontend: **System default** - użytkownicy są uwierzytelniani przez Zabbix, LDAP lub HTTP (w zależności od metody authentication ustawionej globalnie); **Internal** - użytkownicy są uwierzytelniani przez Zabbix; ignorowane, jeśli uwierzytelnianie HTTP jest globalnym ustawieniem domyślnym; **LDAP** - użytkownicy są uwierzytelniani przez LDAP; ignorowane, jeśli uwierzytelnianie HTTP jest globalnym ustawieniem domyślnym; **Disabled** - dostęp do frontend Zabbix jest zabroniony dla tej grupy. Klikając bieżący poziom, można go zmienić.
Debug mode	Wyświetlany jest stan trybu debugowania - *Enabled* lub *Disabled*. Klikając stan, można go zmienić.
Status	Wyświetlany jest status grupy użytkowników - *Enabled* lub *Disabled*. Klikając status, można go zmienić.

Aby skonfigurować nową grupę użytkowników, kliknij przycisk *Create user group* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

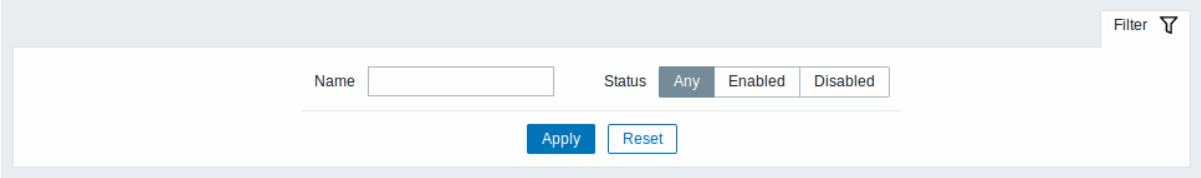
- *Włącz* — zmienia status grupy użytkowników na *Włączona*
- *Wyłącz* — zmienia status grupy użytkowników na *Wyłączona*
- *Włącz tryb debugowania* — włącza tryb debugowania dla grup użytkowników
- *Wyłącz tryb debugowania* — wyłącza tryb debugowania dla grup użytkowników
- *Usuń* — usuwa grupy użytkowników

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich grupach użytkowników, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te grupy użytkowników, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą grup użytkowników. Po jego kliknięciu pojawi się filtr, w którym można filtrować grupy użytkowników według nazwy i statusu.



2 Role użytkowników

Przegląd

W sekcji *Użytkownicy* > *Role użytkowników* można tworzyć role użytkowników.

Role użytkowników umożliwiają tworzenie szczegółowych uprawnień na podstawie początkowo wybranego typu użytkownika (*User*, *Admin*, *Super admin*).

Po wybraniu typu użytkownika wszystkie dostępne uprawnienia dla tego typu użytkownika są przyznawane (domyślnie zaznaczone).

Uprawnienia można jedynie odebrać z podzbioru dostępnego dla danego typu użytkownika; nie można ich rozszerzyć poza zakres dostępny dla tego typu użytkownika.

Pola wyboru dla niedostępnych uprawnień są wyszarzone; użytkownicy nie będą mogli uzyskać dostępu do elementu nawet przez wpisanie bezpośredniego adresu URL tego elementu w przeglądarce.

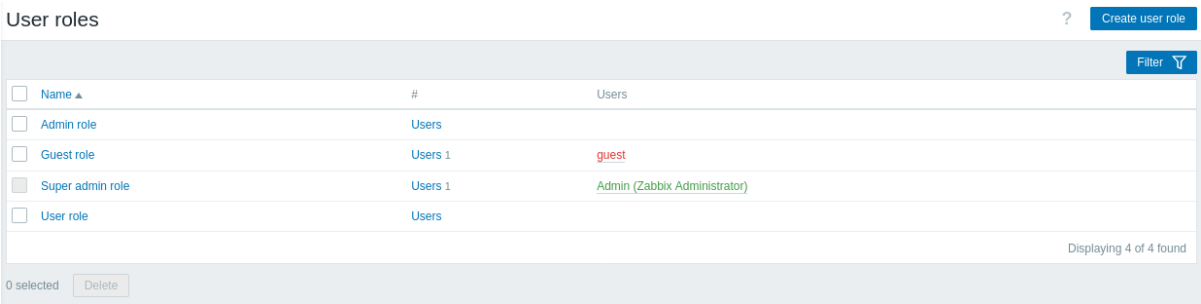
Note:
Ograniczenie dostępu do niektórych elementów interfejsu użytkownika zapobiega jedynie otwarciu tej strony — nie usuwa możliwości dostępu do danych bazowych w innych częściach interfejsu.

Role użytkowników można przypisywać użytkownikom systemu. Każdy użytkownik może mieć przypisaną tylko jedną rolę.

Domyślne role użytkowników

Domyślnie Zabbix jest skonfigurowany z czterema rolami użytkowników, które mają wstępnie zdefiniowany zestaw uprawnień:

- Rola Gość
- Rola Użytkownik
- Rola Administrator
- Rola Super administrator

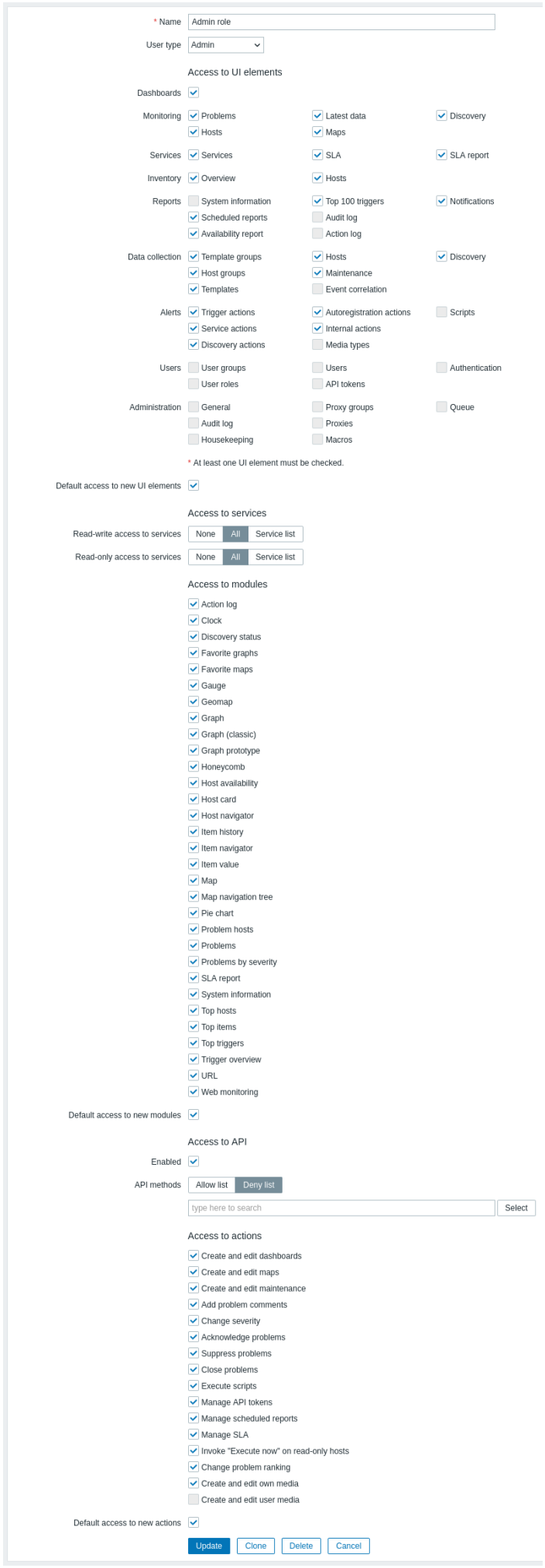


Są one oparte na głównych typach użytkowników w Zabbix. Wyświetlana jest lista wszystkich użytkowników przypisanych do danej roli. Użytkownicy należący do wyłączonych grup są oznaczeni na czerwono. *Rola Gość* jest rolą typu użytkownika z jedynymi uprawnieniami do przeglądania niektórych sekcji frontend.

Note:
Domyślna *Rola Super administrator* nie może być modyfikowana ani usunięta, ponieważ w Zabbix musi istnieć co najmniej jeden użytkownik Super administrator z nieograniczonymi uprawnieniami. Użytkownicy typu *Super administrator* mogą modyfikować ustawienia własnej roli, ale nie typ użytkownika.

Konfiguracja

Aby utworzyć nową rolę, kliknij przycisk *Create user role* w prawym górnym rogu. Aby zaktualizować istniejącą rolę, kliknij nazwę roli, aby otworzyć formularz konfiguracji.


* Name
Admin role
User type
Admin
Access to UI elements
Dashboards
Monitoring
Problems
Latest data
Discovery
Hosts
Maps
Services
Services
SLA
SLA report
Inventory
Overview
Hosts
Reports
System information
Top 100 triggers
Notifications
Scheduled reports
Audit log
Availability report
Action log
Data collection
Template groups
Hosts
Discovery
Host groups
Maintenance
Templates
Event correlation
Alerts
Trigger actions
Autoregistration actions
Scripts
Service actions
Internal actions
Discovery actions
Media types
Users
User groups
Users
Authentication
User roles
API tokens
Administration
General
Proxy groups
Queue
Audit log
Proxies
Housekeeping
Macros
* At least one UI element must be checked.
Default access to new UI elements
Access to services
Read-write access to services
None
All
Service list
Read-only access to services
None
All
Service list
Access to modules
Action log
Clock
Discovery status
Favorite graphs
Favorite maps
Gauge
Geomap
Graph
Graph (classic)
Graph prototype
Honeycomb
Host availability
Host card
Host navigator
Item history
Item navigator
Item value
Map
Map navigation tree
Pie chart
Problem hosts
Problems
Problems by severity
SLA report
System information
Top hosts
Top items
Top triggers
Trigger overview
URL
Web monitoring
Default access to new modules
Access to API
Enabled
API methods
Allow list
Deny list
type here to search
Select
Access to actions
Create and edit dashboards
Create and edit maps
Create and edit maintenance
Add problem comments
Change severity
Acknowledge problems
Suppress problems
Close problems
Execute scripts
Manage API tokens
Manage scheduled reports
Manage SLA
Invoke "Execute now" on read-only hosts
Change problem ranking
Create and edit own media
Create and edit user media
Default access to new actions
Update
Clone
Delete
Cancel

Wyświetlane są dostępne uprawnienia. Aby cofnąć określone uprawnienie, odznacz jego pole wyboru.

Dostępne uprawnienia wraz z wartościami domyślnymi dla każdej wstępnie skonfigurowanej roli użytkownika w Zabbix opisano poniżej.

Domyślne uprawnienia

Dostęp do elementów interfejsu użytkownika

Domyślny dostęp do sekcji menu zależy od typu użytkownika. Zobacz stronę Uprawnienia, aby uzyskać szczegóły.

Dostęp do innych opcji

Parametr	**Opis**	**Domyślne role użytkowników**			
		Rola Super admin	*Rola Admin*	*Rola User*	*Rola Guest*
Domyślny dostęp do nowych elementów interfejsu użytkownika	Ta opcja określa, jak nowe sekcje menu będą dostępne po aktualizacji Zabbix. Istniejące sekcje menu modułów pozostają bez zmian.	Tak	Tak	Tak	Tak
Dostęp do usług					
Dostęp do usług do odczytu i zapisu	Wybierz dostęp do usług do odczytu i zapisu: **Brak** - brak dostępu **Wszystkie** - dostęp do wszystkich usług w trybie odczytu i zapisu **Lista usług** - wybierz usługi z dostępem do odczytu i zapisu Dostęp do odczytu i zapisu, jeśli został przyznany, ma pierwszeństwo przed ustawieniami dostępu tylko do odczytu i jest dynamicznie dziedziczony przez usługi podrzędne.	Wszystkie	Wszystkie	Brak	Brak
Dostęp do usług do odczytu i zapisu z tagiem	Określ nazwę tagu oraz opcjonalnie jego wartość, aby dodatkowo przyznać dostęp do usług do odczytu i zapisu usługom pasującym do tagu. Ta opcja jest dostępna, jeśli w parametrze *Dostęp do usług do odczytu i zapisu* wybrano „Lista usług”. Dostęp do odczytu i zapisu, jeśli został przyznany, ma pierwszeństwo przed ustawieniami dostępu tylko do odczytu i jest dynamicznie dziedziczony przez usługi podrzędne.				

Dostęp do usług tylko do odczytu	Wybierz dostęp do usług tylko do odczytu: **Brak** - brak dostępu **Wszystkie** - dostęp do wszystkich usług tylko do odczytu **Lista usług** - wybierz usługi z dostępem tylko do odczytu Dostęp tylko do odczytu nie ma pierwszeństwa przed dostępem do odczytu i zapisu i jest dynamicznie dziedziczony przez usługi podrzędne.			Wszystkie	Wszystkie
Dostęp do usług tylko do odczytu z tagiem	Określ nazwę tagu oraz opcjonalnie jego wartość, aby dodatkowo przyznać dostęp tylko do odczytu usługom pasującym do tagu. Ta opcja jest dostępna, jeśli w parametrze *Dostęp do usług tylko do odczytu* wybrano „Lista usług”. Dostęp tylko do odczytu nie ma pierwszeństwa przed dostępem do odczytu i zapisu i jest dynamicznie dziedziczony przez usługi podrzędne.				
Dostęp do mod-ułów					
<Nazwa modułu>	Zezwalaj/zabraniaj dostępu do określonego modułu. W tej sekcji są wyświetlane tylko włączone moduły. Nie można przyznać ani ograniczyć dostępu do modułu, który jest obecnie wyłączony.	Tak	Tak	Tak	Tak
Domyślny dostęp do nowych modułów	Ta opcja określa, jak nowe moduły i widgety będą dostępne po aktualizacji Zabbix. Dotyczy to również modułów i widgetów dodanych w sekcji *Administration* > *General* > *Modules*.				
Dostęp do API					
Włączone Metody API	Włącz/wyłącz dostęp do API. Wybierz *Allow list*, aby zezwolić, albo *Deny list*, aby zabronić metod API określonych w polu wyszukiwania. Pamiętaj, że nie można zezwolić na niektóre metody API i jednocześnie zabronić innych. W polu wyszukiwania zacznij wpisywać nazwę metody, a następnie wybierz metodę z listy autouzupełniania. Możesz także nacisnąć przycisk Select i wybrać metody z pełnej listy dostępnej dla tego typu użytkownika. Pamiętaj, że jeśli określona akcja w bloku Dostęp do akcji jest odznaczona, użytkownicy nie będą mogli używać metod API powiązanych z tą akcją. Obsługiwane są symbole wieloznaczne. Przykłady: `dashboard.*` (wszystkie metody usługi API „dashboard.”) `*` (dowolna metoda), `*.export` (metody o nazwie „.export” ze wszystkich usług API). Jeśli nie określono żadnych metod, reguła *Allow/Deny list* zostanie zignorowana.	Tak	Tak	Tak	Nie
Dostęp do akcji					
Tworzenie i edycja pulpitów	Wyczyszczenie tego pola wyboru spowoduje również odebranie praw do używania metod API `.create`, `.update` i `.delete` dla odpowiednich elementów.	Tak	Tak	Tak	Nie

Tworzenie i edycja map			
Tworzenie i edycja okresów konserwacji			Nie
Dodawanie komentarzy do problemów Zmiana ważności Potwierdzanie problemów Tłumienie problemów Zamykanie problemów	Wyczyszczenie tego pola wyboru spowoduje również odebranie praw do wykonywania odpowiedniej akcji za pomocą metody API `event.acknowledge`.		Tak
Wykonywanie skryptów	Wyczyszczenie tego pola wyboru spowoduje również odebranie praw do używania metody API `script.execute`.		
Zarządzanie tokenami API	Wyczyszczenie tego pola wyboru spowoduje również odebranie praw do używania wszystkich metod API `token.*`.		
Zarządzanie zaplanowanymi raportami	Wyczyszczenie tego pola wyboru spowoduje również odebranie praw do używania wszystkich metod API `report.*`.		Nie
Zarządzanie SLA	Włącz/wyłącz prawa do zarządzania SLA.		
Wywoływanie „Wykonaj teraz” na hostach tylko do odczytu	Zezwalaj na używanie opcji „Wykonaj teraz” w najnowszych danych dla pozycji hostów tylko do odczytu.		Tak
Zmiana rankingu problemu	Zezwalaj na zmianę rankingu problemu z przyczyny na symptom i odwrotnie.		
Tworzenie i edycja własnych mediów	Zezwalaj na tworzenie/edycję własnych mediów.		
Tworzenie i edycja mediów użytkownika	Zezwalaj na tworzenie/edycję mediów dla użytkowników. Ta opcja jest dostępna tylko dla użytkowników Super admin.	Nie	Nie
Domyślny dostęp do nowych akcji	Ta opcja określa, jak nowe akcje będą dostępne po aktualizacji Zabbix.	Tak	Tak

Zobacz także:

- Konfigurowanie użytkownika

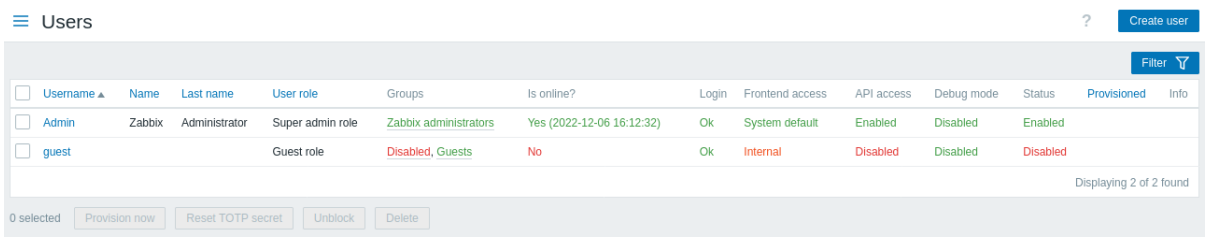
3 Użytkownicy

Przegląd

W sekcji *Użytkownicy* → *Użytkownicy* zarządza się użytkownikami systemu.

Użytkownicy

Wyświetlana jest lista istniejących użytkowników wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa użytkownika	Nazwa użytkownika używana do logowania do Zabbix. Kliknięcie nazwy użytkownika otwiera formularz konfiguracji użytkownika.
Imię	Imię użytkownika.
Nazwisko	Nazwisko użytkownika.
Rola użytkownika	Wyświetlana jest rola użytkownika.
Grupy	Wyświetlane są grupy, do których należy użytkownik. Kliknięcie nazwy grupy użytkowników otwiera formularz konfiguracji grupy użytkowników. Wyłączone grupy są wyświetlane na czerwono.
Czy online?	Wyświetlany jest status online użytkownika — *Tak* lub *Nie*. Czas ostatniej aktywności użytkownika jest wyświetlany w nawiasach.
Logowanie	Wyświetlany jest status logowania użytkownika — *Ok* lub *Zablokowany*. Użytkownik może zostać tymczasowo zablokowany po przekroczeniu liczby nieudanych prób logowania ustawionej w sekcji *Administracja → Ogólne → Inne* (domyślnie pięć). Klikając *Zablokowany*, można odblokować użytkownika.
Dostęp do frontend	Wyświetlany jest poziom dostępu do frontend — *Domyślny systemowy*, *Wewnętrzny*, *LDAP* lub *Wyłączony*, zależnie od ustawienia dla całej grupy użytkowników.
Dostęp do API	Wyświetlany jest status dostępu do API — *Włączony* lub *Wyłączony*, zależnie od ustawienia dla roli użytkownika.
Tryb debugowania	Wyświetlany jest status trybu debugowania — *Włączony* lub *Wyłączony*, zależnie od ustawienia dla całej grupy użytkowników.
Status	Wyświetlany jest status użytkownika — *Włączony* lub *Wyłączony*, zależnie od ustawienia dla całej grupy użytkowników.
Provisioned	Wyświetlana jest data ostatniego provisioningu użytkownika. Używane dla użytkowników utworzonych przez provisioning JIT z LDAP/SAML.
Info	Wyświetlane są informacje o błędach. Żółte ostrzeżenie jest wyświetlane dla użytkowników bez grup użytkowników. Czerwone ostrzeżenie jest wyświetlane dla użytkowników bez ról oraz dla użytkowników bez ról i grup użytkowników.

Aby skonfigurować nowego użytkownika, kliknij przycisk *Utwórz użytkownika* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

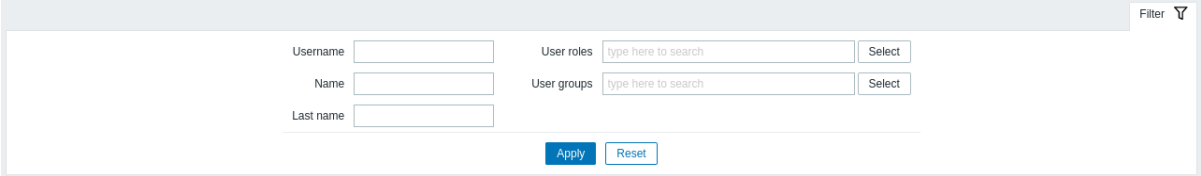
- *Provision now* - zaktualizuj informacje o użytkowniku z LDAP (ta opcja jest włączona tylko wtedy, gdy wybrano użytkownika LDAP)
- *Reset TOTP secret* - zresetuj sekrety TOTP użytkownika dla wszystkich metod TOTP i usuń sesję użytkownika (ta opcja jest włączona tylko wtedy, gdy włączono MFA; w przypadku użytkowników bez sekretów TOTP ich sesja nie zostanie usunięta)
- *Unblock* - ponownie włącz dostęp do systemu dla zablokowanych użytkowników
- *Delete* - usuń użytkowników

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich użytkownikach, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko użytkowników, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

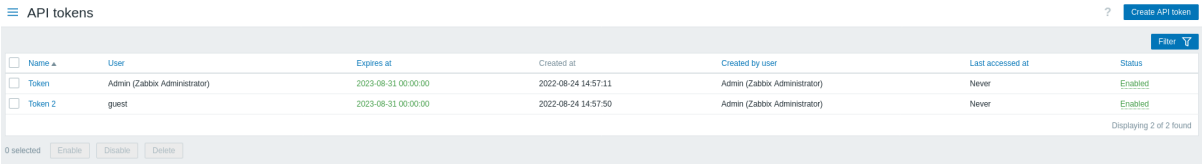
Łącze *Filtr* jest dostępne nad listą użytkowników. Po jego kliknięciu udostępniony zostanie filtr, za pomocą którego można filtrować użytkowników według nazwy użytkownika, imienia, nazwiska, roli użytkownika i grupy użytkowników.



4 Tokeny API

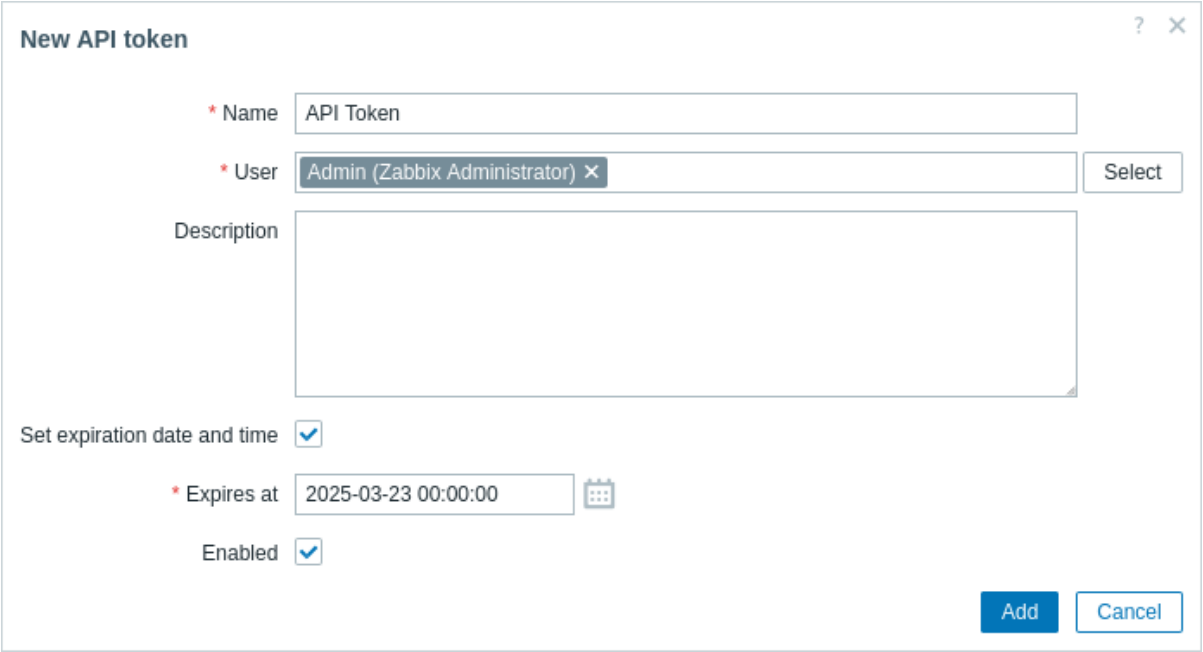
Przegląd

Ta sekcja umożliwia tworzenie tokenów API i zarządzanie nimi.



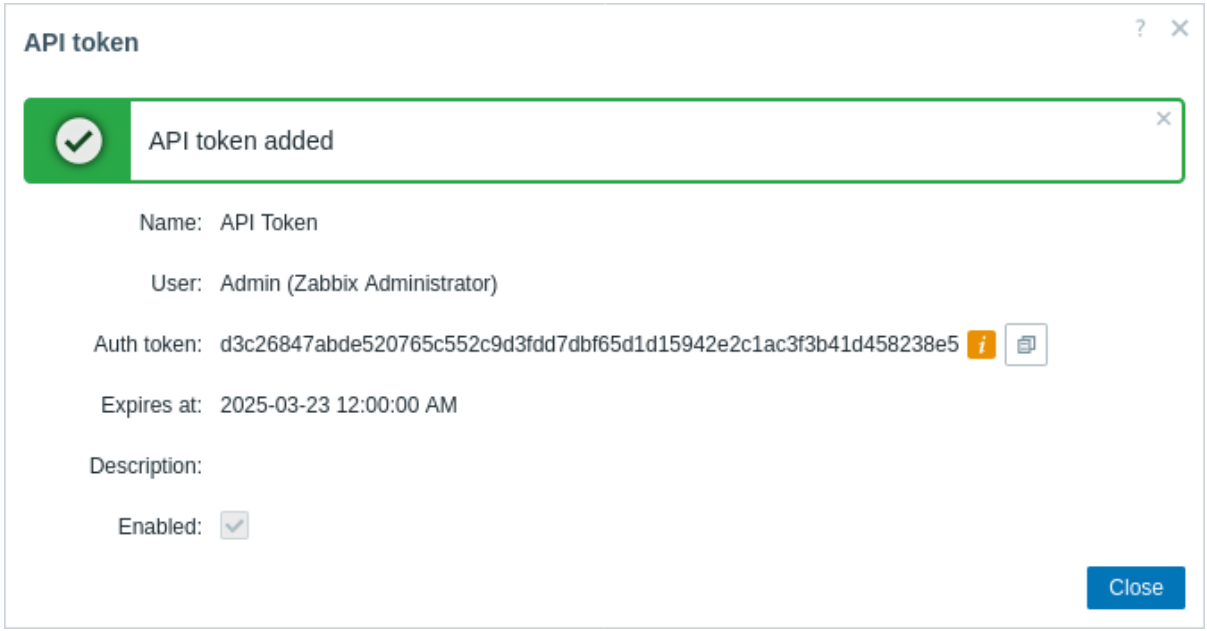
Tokeny API można filtrować według nazwy, użytkowników, do których tokeny są przypisane, daty wygaśnięcia, użytkowników, którzy utworzyli tokeny, lub statusu (włączony/wyłączony). Kliknij status tokenu na liście, aby szybko włączyć/wyłączyć token. Możesz także zbiorczo włączać/wyłączać tokeny, zaznaczając je na liście, a następnie klikając przyciski *Enable/Disable* pod listą.

Aby utworzyć nowy token, naciśnij przycisk *Create API token* w prawym górnym rogu, a następnie wypełnij wymagane pola na ekranie konfiguracji tokenu:



Parameter	Description
Name	Widoczna nazwa tokenu.
User	Użytkownik, do którego token ma zostać przypisany. Aby szybko wybrać użytkownika, zacznij wpisywać nazwę użytkownika, imię lub nazwisko, a następnie wybierz odpowiedniego użytkownika z listy autouzupełniania. Alternatywnie możesz nacisnąć przycisk *Select* i wybrać użytkownika z pełnej listy użytkowników. Token może być przypisany tylko do jednego użytkownika.
Description	Opcjonalny opis tokenu.
Set expiration date and time	Odznacz to pole wyboru, jeśli token nie powinien mieć daty wygaśnięcia.
Expiry date	Kliknij ikonę kalendarza, aby wybrać datę wygaśnięcia tokenu, lub wprowadź datę ręcznie w formacie YYYY-MM-DD hh:mm:ss
Enabled	Odznacz to pole wyboru, jeśli chcesz utworzyć token w stanie wyłączonym.

Naciśnij *Add*, aby utworzyć token.



Skopiuj wartość *Auth token* i zapisz ją w bezpiecznym miejscu **przed zamknięciem strony**, a następnie naciśnij Close. Token pojawi się na liście.

Warning:
Wartość *Auth token* nie może zostać później ponownie wyświetlona. Jest ona dostępna tylko bezpośrednio po utworzeniu tokenu. Jeśli utracisz zapisany token, konieczne będzie jego ponowne wygenerowanie, co spowoduje utworzenie nowego ciągu autoryzacyjnego.

Kliknij nazwę tokenu, aby edytować nazwę, opis, ustawienia daty wygaśnięcia lub status tokenu. Zwróć uwagę, że nie ma możliwości zmiany użytkownika, do którego token jest przypisany. Naciśnij przycisk *Update*, aby zapisać zmiany. Jeśli token został utracony lub ujawniony, możesz nacisnąć przycisk *Regenerate*, aby wygenerować nową wartość tokenu. Pojawi się okno dialogowe z prośbą o potwierdzenie tej operacji, ponieważ po jej wykonaniu wcześniej wygenerowany token stanie się nieważny.

Użytkownicy bez dostępu do sekcji menu *Administration* mogą przeglądać i modyfikować szczegóły tokenów przypisanych do nich w sekcji *User profile* → *API tokens* section tylko wtedy, gdy uprawnienie *Manage API tokens* jest dozwolone w uprawnieniach ich user role.

5 Uwierzytelnianie

Przegląd

Sekcja *Użytkownicy* → *Uwierzytelnianie* umożliwia określenie metody uwierzytelniania użytkowników dla Zabbix oraz wewnętrznych wymagań dotyczących haseł.

Dostępne metody uwierzytelniania to uwierzytelnianie wewnętrzne, HTTP, LDAP, SAML oraz MFA.

Domyślne uwierzytelnianie

Domyślnie Zabbix używa **wewnętrznego** uwierzytelniania Zabbix dla wszystkich użytkowników.

Można zmienić domyślną metodę uwierzytelniania na LDAP w całym systemie. Aby to zrobić, przejdź do zakładki *LDAP* i skonfiguruj parametry LDAP, a następnie wróć do zakładki *Authentication* i przełącz selektor *Default authentication* na LDAP.

Należy pamiętać, że metodę uwierzytelniania można precyzyjnie dostosować na poziomie grupy użytkowników. Nawet jeśli uwierzytelnianie LDAP jest ustawione globalnie, niektóre grupy użytkowników nadal mogą być uwierzytelniane przez Zabbix. Dla tych grup frontend access musi być ustawiony na Internal.

Możliwe jest również włączenie uwierzytelniania LDAP tylko dla określonych grup użytkowników, jeśli globalnie używane jest uwierzytelnianie wewnętrzne. W takim przypadku szczegóły uwierzytelniania LDAP mogą zostać określone i użyte dla konkretnych grup użytkowników, których frontend access musi być wtedy ustawiony na LDAP. Jeśli użytkownik należy do co najmniej jednej grupy użytkowników z uwierzytelnianiem LDAP, nie będzie mógł korzystać z metody uwierzytelniania wewnętrznego.

Metody uwierzytelniania HTTP, SAML 2.0 i MFA mogą być używane dodatkowo obok domyślnej metody uwierzytelniania.

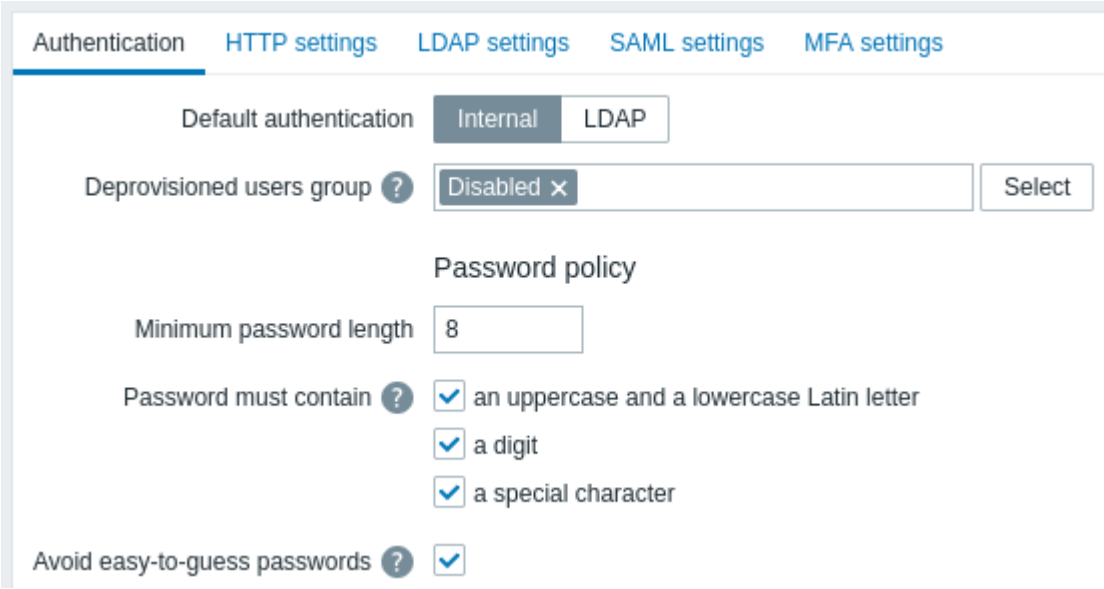
Zabbix obsługuje provisioning just-in-time (JIT), który umożliwia tworzenie kont użytkowników w Zabbix przy pierwszym uwierzytelnieniu użytkownika zewnętrznego oraz provisionowanie tych kont użytkowników. Provisioning JIT jest obsługiwany dla LDAP i SAML.

Zobacz także:

- Uwierzytelnianie HTTP
- Uwierzytelnianie LDAP
- Uwierzytelnianie SAML
- Uwierzytelnianie MFA

Konfiguracja

Zakładka *Authentication* umożliwia ustawienie domyślnej metody uwierzytelniania, określenie grupy dla użytkowników wycofanych z obsługi oraz ustawienie wymagań dotyczących złożoności haseł dla użytkowników Zabbix.



Parametry konfiguracji:

Parametr	Opis
Default authentication	Wybierz domyślną metodę uwierzytelniania w Zabbix — *Internal* lub *LDAP*.
Deprovisioned users group	Określ grupę użytkowników dla użytkowników wycofanych z obsługi. To ustawienie jest wymagane tylko w przypadku aprowizacji JIT, w odniesieniu do użytkowników utworzonych w Zabbix z systemów LDAP lub SAML, którzy nie muszą już być dalej aprowizowani. Należy wskazać wyłączoną grupę użytkowników.
Minimum password length	Domyślnie minimalna długość hasła jest ustawiona na 8. Obsługiwany zakres: 1-70. Należy pamiętać, że hasła dłuższe niż 72 znaki zostaną obcięte.
Password must contain	Zaznacz jedno lub kilka pól wyboru, aby wymagać użycia określonych znaków w haśle: - wielkiej i małej litery alfabetu łacińskiego - cyfry - znaku specjalnego Najedź kursorem na znak zapytania, aby zobaczyć podpowiedź z listą znaków dla każdej opcji.
Avoid easy-to-guess passwords	Jeśli opcja jest zaznaczona, hasło będzie sprawdzane pod kątem następujących wymagań: - nie może zawierać imienia, nazwiska ani nazwy użytkownika - nie może być jednym z popularnych haseł ani haseł specyficznych dla kontekstu. Lista popularnych i specyficznych dla kontekstu haseł jest generowana automatycznie na podstawie listy NCSC „Top 100k passwords”, listy SecLists „Top 1M passwords” oraz listy haseł specyficznych dla kontekstu Zabbix. Użytkownicy wewnętrzni nie będą mogli ustawiać haseł znajdujących się na tej liście, ponieważ takie hasła są uznawane za słabe ze względu na ich powszechne użycie.

Zmiany wymagań dotyczących złożoności haseł nie wpłyną na istniejące hasła użytkowników, ale jeśli istniejący użytkownik zdecyduje się zmienić hasło, nowe hasło będzie musiało spełniać bieżące wymagania. Podpowiedź z listą wymagań będzie wyświetlana obok pola *Password* w profilu użytkownika oraz w formularzu konfiguracji użytkownika dostępnym z menu *Users* → *Users*.

1 HTTP

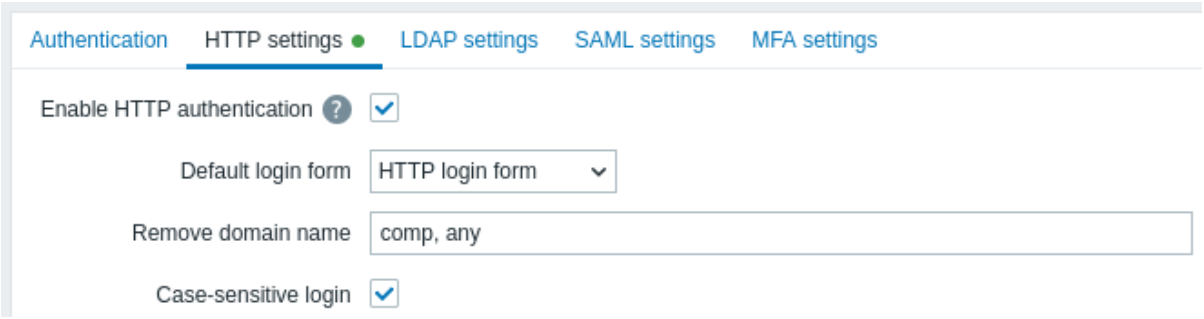
Przegląd

Uwierzytelnianie HTTP lub oparte na serwerze WWW authentication (na przykład: BasicAuthentication, NTLM/Kerberos) może być używane do sprawdzania nazw użytkowników i haseł. Należy jednak pamiętać, że użytkownik musi również istnieć w Zabbix, jednak jego hasło Zabbix nie będzie używane.

Attention:
Uwaga! Upewnij się, że uwierzytelnianie serwera WWW jest skonfigurowane i działa poprawnie przed jego włączeniem.

Uwierzytelnianie HTTP można wyłączyć we frontendzie, konfigurując odpowiednią opcję na karcie *HTTP settings* w sekcji *Users > Authentication*. Gdy uwierzytelnianie HTTP jest wyłączone, karta z opcjami uwierzytelniania HTTP nie będzie wyświetlana we frontendzie. Należy pamiętać, że ponowna instalacja frontendu (uruchomienie setup.php) zresetuje ustawienia uwierzytelniania, w tym konfigurację uwierzytelniania HTTP.

Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Włącz uwierzytelnianie HTTP	Zaznacz pole wyboru, aby włączyć uwierzytelnianie HTTP. Po najechaniu kursorem na zostanie wyświetlona podpowiedź z ostrzeżeniem, że w przypadku uwierzytelniania przez serwer WWW wszyscy użytkownicy (nawet z dostępem do frontend ustawionym na LDAP/Internal) będą uwierzytelniani przez serwer WWW, a nie przez Zabbix.
Domyślny formularz logowania	Określ, czy nieuwierzytelnieni użytkownicy mają być kierowani do: **Formularz logowania Zabbix** - standardowa strona logowania Zabbix. **Formularz logowania HTTP** - strona logowania HTTP. Zaleca się włączenie uwierzytelniania opartego na serwerze WWW tylko dla strony `index_http.php`. Jeśli *Domyślny formularz logowania* jest ustawiony na „Strona logowania HTTP", użytkownik zostanie zalogowany automatycznie, jeśli moduł uwierzytelniania serwera WWW ustawi prawidłowy login użytkownika w zmiennej `$_SERVER`. Obsługiwane klucze `$_SERVER` to `PHP_AUTH_USER`, `REMOTE_USER`, `AUTH_USER`.
Usuń nazwę domeny	Lista nazw domen rozdzielonych przecinkami, które powinny zostać usunięte z nazwy użytkownika. Na przykład `comp,any` — jeśli nazwa użytkownika to 'Admin@any' lub 'comp\Admin', użytkownik zostanie zalogowany jako 'Admin'; jeśli nazwa użytkownika to 'notacompany\Admin', logowanie zostanie odrzucone.
Logowanie z rozróżnianiem wielkości liter	Odznacz pole wyboru, aby wyłączyć logowanie z rozróżnianiem wielkości liter dla nazw użytkowników (domyślnie włączone). Wyłączenie logowania z rozróżnianiem wielkości liter pozwala na przykład zalogować się jako „admin", nawet jeśli użytkownik Zabbix to „Admin" lub „ADMIN". Pamiętaj, że jeśli logowanie z rozróżnianiem wielkości liter jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach użytkownika (np. Admin i admin), logowanie dla tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: „Authentication failed: supplied credentials are not unique."

Note:
W przypadku użytkowników wewnętrznych, którzy nie mogą zalogować się przy użyciu poświadczeń HTTP (gdy domyślnie ustawiony jest formularz logowania HTTP), co prowadzi do błędu 401, można dodać w dyrektywach podstawowego uwierzytelniania wiersz `ErrorDocument 401 /index.php?form=default`, który przekieruje do zwykłego formularza logowania Zabbix.

Przegląd

Zewnętrzne uwierzytelnianie LDAP może być używane do sprawdzania nazw użytkowników i haseł.

Uwierzytelnianie LDAP w Zabbix działa co najmniej z Microsoft Active Directory i OpenLDAP.

Jeśli skonfigurowano tylko logowanie LDAP, użytkownik musi również istnieć w Zabbix, jednak jego hasło Zabbix nie będzie używane. Jeśli uwierzytelnianie zakończy się powodzeniem, Zabbix dopasuje lokalną nazwę użytkownika do atrybutu nazwy użytkownika zwróconego przez LDAP.

Udostępnianie użytkowników

Możliwe jest skonfigurowanie JIT (just-in-time) **udostępniania użytkowników** dla użytkowników LDAP. W takim przypadku nie jest wymagane, aby użytkownik istniał już w Zabbix. Konto użytkownika może zostać utworzone, gdy użytkownik zaloguje się do Zabbix po raz pierwszy.

Gdy użytkownik LDAP wprowadza swój login LDAP i hasło, Zabbix sprawdza *domyślny* serwer LDAP, aby ustalić, czy taki użytkownik istnieje. Jeśli użytkownik istnieje i nie ma jeszcze konta w Zabbix, w Zabbix tworzony jest nowy użytkownik i użytkownik może się zalogować.

Użytkownik utworzony za pomocą udostępniania JIT jest powiązany z serwerem LDAP (katalogiem), który w momencie utworzenia jest ustawiony jako domyślny. Późniejsza zmiana domyślnego serwera LDAP nie zmienia ani nie aktualizuje serwera LDAP powiązanego z użytkownikami, którzy zostali już udostępnieni.

Attention:
Jeśli udostępnianie JIT jest włączone, na karcie *Authentication* należy określić grupę użytkowników dla użytkowników wycofanych z udostępniania.

Udostępnianie JIT umożliwia również aktualizowanie udostępnionych kont użytkowników na podstawie zmian w LDAP. Na przykład, jeśli użytkownik zostanie przeniesiony z jednej grupy LDAP do innej, użytkownik zostanie również przeniesiony z jednej grupy do innej w Zabbix; jeśli użytkownik zostanie usunięty z grupy LDAP, użytkownik zostanie również usunięty z grupy w Zabbix i, jeśli nie należy do żadnej innej grupy, zostanie dodany do grupy użytkowników dla użytkowników wycofanych z udostępniania. Udostępnione konta użytkowników są aktualizowane zgodnie ze skonfigurowanym okresem udostępniania lub gdy użytkownik loguje się do Zabbix.

Należy pamiętać, że udostępnianie w tle jest wykonywane przez frontend Zabbix, gdy użytkownik z niego korzysta lub ma co najmniej jedną otwartą stronę frontend w przeglądarce. Nie ma dedykowanych procesów działających w tle do udostępniania użytkowników.

LDAP obsługuje trzy sposoby wiązania z katalogiem na potrzeby uwierzytelniania i wyszukiwania:

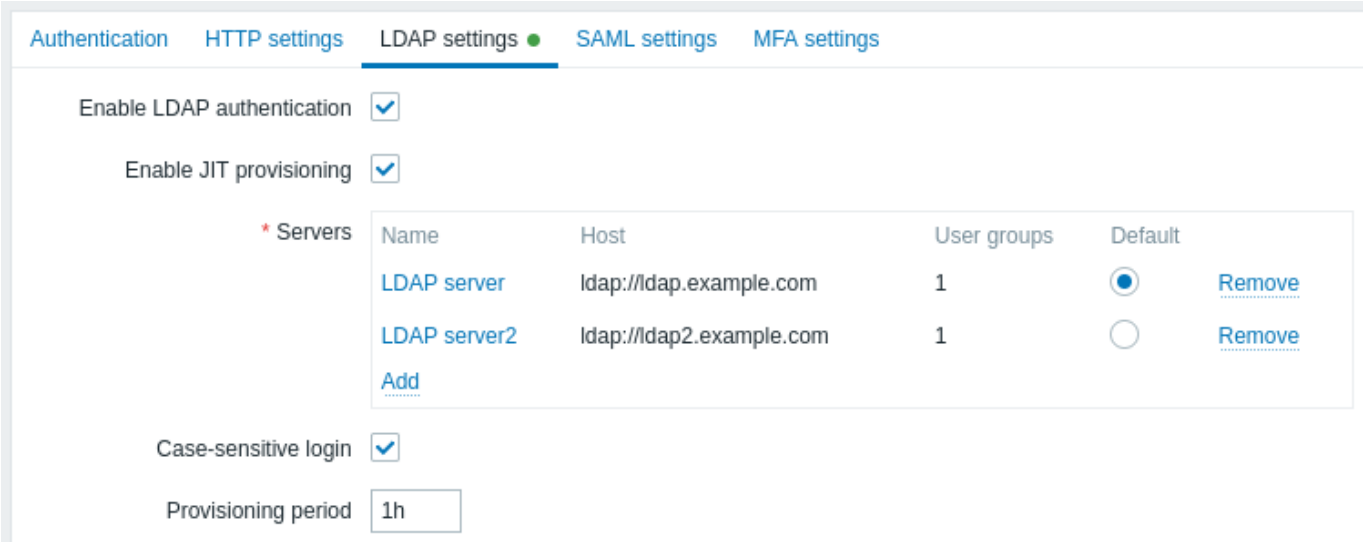
- Wiązanie anonimowe — nie podano *Bind DN* / *Bind password*, a serwer LDAP zezwala na anonimowe zapytania.
- Dedykowany użytkownik wiążący (konto usługi) — określone konto LDAP jest ustawione w *Bind DN* / *Bind password* i jest używane przez Zabbix do wyszukiwania oraz udostępniania. Jest to zalecana i najbardziej elastyczna opcja, ponieważ Zabbix może wykonywać wyszukiwania i udostępnianie w tle bez poświadczeń użytkownika końcowego.
- Bezpośrednie wiązanie użytkownika — Zabbix wykonuje wiązanie przy użyciu poświadczeń wprowadzonych przez użytkownika podczas logowania (bez skonfigurowanych *Bind DN* / *Bind password*); konfiguruje się to przez uwzględnienie symbolu zastępczego, takiego jak `uid=%{user}`, w Base DN. W tym trybie Zabbix ma dostęp do hasła użytkownika tylko podczas interaktywnego logowania. W rezultacie działania związane z udostępnianiem, które wymagają uwierzytelnienia w LDAP poza sesją logowania użytkownika (na przykład użycie przycisku *Provision now* lub uruchomienie udostępniania w tle, gdy użytkownik nie jest aktywnie zalogowany), nie mogą się uwierzytelnić i dlatego nie będą działać. Udostępnianie i aktualizacje w przypadku bezpośredniego wiązania użytkownika odbywają się wyłącznie w momencie logowania użytkownika.

Wiele serwerów

W razie potrzeby można zdefiniować kilka serwerów LDAP. Na przykład do uwierzytelniania innej grupy użytkowników można użyć innego serwera. Po skonfigurowaniu serwerów LDAP w konfiguracji grupy użytkowników będzie można wybrać wymagany serwer LDAP dla odpowiedniej grupy użytkowników.

Jeśli użytkownik należy do wielu grup użytkowników i wielu serwerów LDAP, do uwierzytelniania zostanie użyty pierwszy serwer z listy serwerów LDAP posortowanej według nazwy w porządku rosnącym.

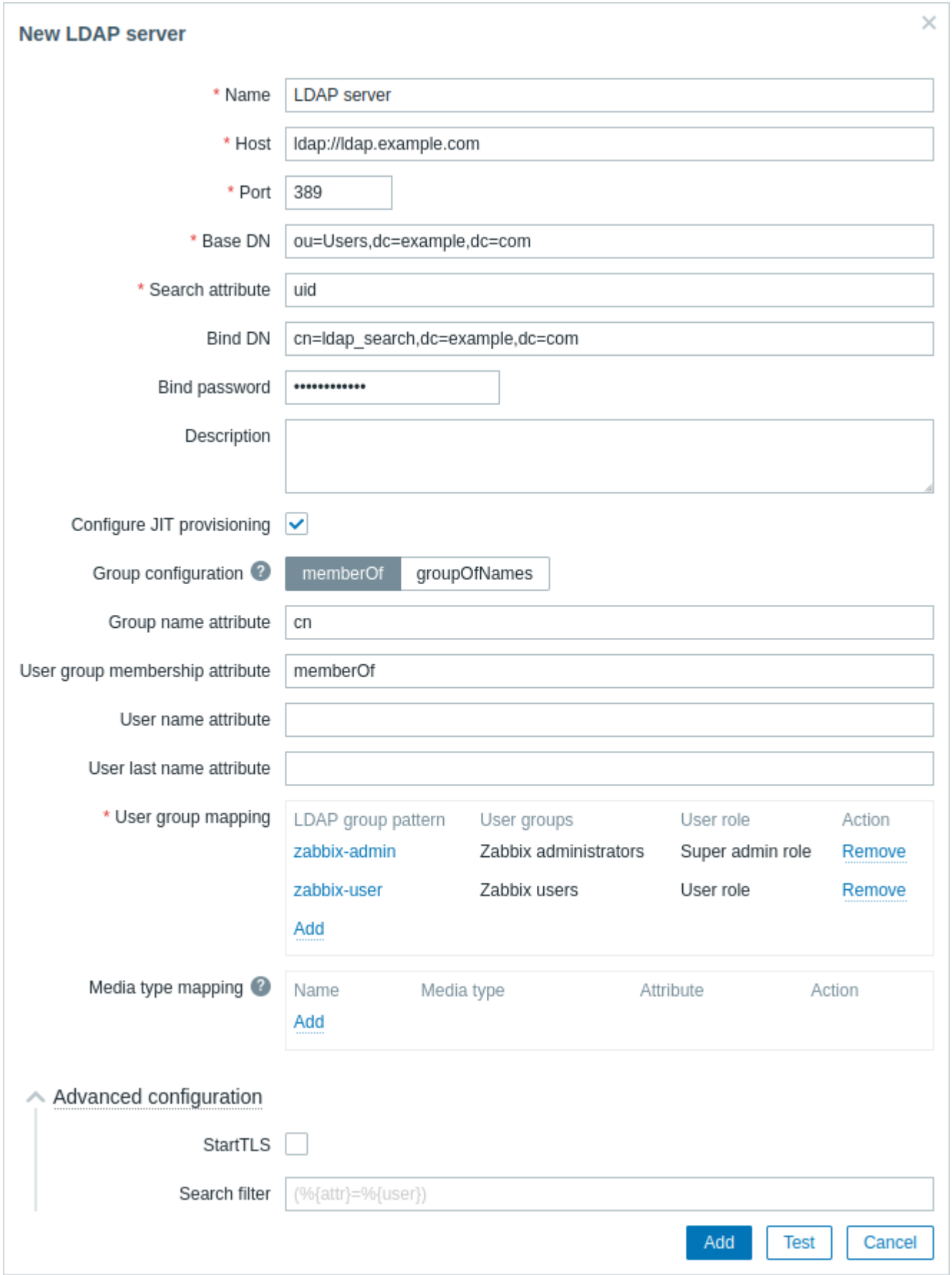
Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Enable LDAP authentication	Zaznacz pole wyboru, aby włączyć uwierzytelnianie LDAP.
Enable JIT provisioning	Zaznacz pole wyboru, aby włączyć aprowizację JIT.
Servers	Kliknij *Add*, aby skonfigurować serwer LDAP (zobacz konfiguracja serwera LDAP poniżej).
Case-sensitive login	Odznacz pole wyboru, aby wyłączyć rozróżnianie wielkości liter podczas logowania dla nazw użytkowników (domyślnie włączone). Wyłączenie rozróżniania wielkości liter podczas logowania pozwala na przykład zalogować się jako „admin”, nawet jeśli użytkownik Zabbix to „Admin” lub „ADMIN”. Pamiętaj, że jeśli rozróżnianie wielkości liter podczas logowania jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach użytkownika (np. Admin i admin), logowanie dla tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: „Authentication failed: supplied credentials are not unique.”
Provisioning period	Ustaw okres aprowizacji, tj. jak często zalogowany użytkownik będzie aprowizowany podczas pracy z frontend.

Konfiguracja serwera LDAP



Parametry konfiguracji serwera LDAP:

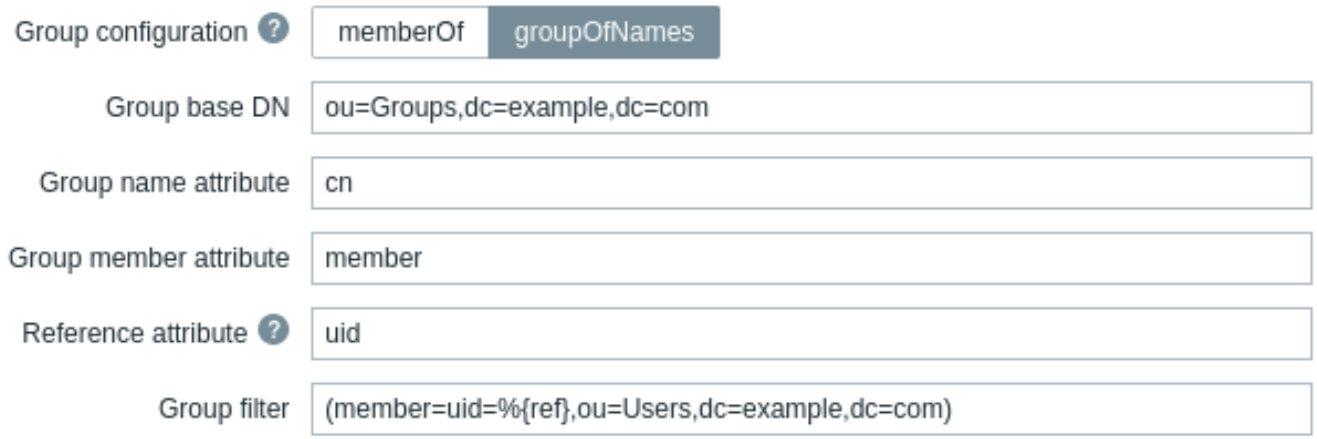
Parameter	Description
Name	Nazwa serwera LDAP w konfiguracji Zabbix.

Parameter	Description
Host	Nazwa hosta, IP lub URI serwera LDAP. Przykłady: ldap.example.com, 127.0.0.1, ldap://ldap.example.com W przypadku bezpiecznego serwera LDAP użyj protokołu *ldaps* i nazwy hosta. Przykład: ldaps://ldap.example.com W OpenLDAP 2.x.x i nowszych można użyć pełnego URI LDAP w postaci ldap://hostname:port lub ldaps://hostname:port.
Port	Port serwera LDAP. Domyślnie 389. Dla bezpiecznego połączenia LDAP numer portu to zwykle 636. Nie jest używany przy korzystaniu z pełnych URI LDAP.
Base DN	Ścieżka bazowa do kont użytkowników na serwerze LDAP: ou=Users,ou=system (dla OpenLDAP), DC=company,DC=com (dla Microsoft Active Directory) uid=%{user},dc=example,dc=com (dla bezpośredniego wiązania użytkownika, patrz uwaga poniżej)
Search attribute	Atrybut konta LDAP używany do wyszukiwania: uid (dla OpenLDAP), sAMAccountName (dla Microsoft Active Directory)
Bind DN	Konto LDAP do wiązania i wyszukiwania na serwerze LDAP, przykłady: uid=ldap_search,ou=system (dla OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (dla Microsoft Active Directory) Obsługiwane jest również anonimowe wiązanie. Należy pamiętać, że anonimowe wiązanie może potencjalnie udostępnić konfigurację domeny nieuprawnionym użytkownikom (informacje o użytkownikach, komputerach, serwerach, grupach, usługach itp.). Ze względów bezpieczeństwa wyłącz anonimowe wiązania na hostach LDAP i zamiast tego używaj uwierzytelnionego dostępu.
Bind password	Hasło LDAP konta używanego do wiązania i wyszukiwania na serwerze LDAP.
Description	Opis serwera LDAP.
Configure JIT provisioning	Zaznacz to pole wyboru, aby wyświetlić opcje związane z aprowizacją JIT.
Group configuration	Wybierz metodę konfiguracji grup: **memberOf** - przez wyszukiwanie użytkowników i atrybutu ich przynależności do grupy **groupOfNames** - przez wyszukiwanie grup za pomocą atrybutu member Należy pamiętać, że memberOf jest preferowane, ponieważ działa szybciej; użyj groupOfNames, jeśli Twój serwer LDAP nie obsługuje `memberOf` lub wymagane jest filtrowanie grup.
Group name attribute	Określ atrybut, z którego ma być pobierana nazwa grupy ze wszystkich obiektów w atrybucie `memberOf` (zobacz pole *User group membership attribute*) Nazwa grupy jest wymagana do mapowania grup użytkowników.
User group membership attribute	Określ atrybut zawierający informacje o grupach, do których należy użytkownik (np. `memberOf`). Na przykład atrybut memberOf może zawierać informacje takie jak: `memberOf=cn=zabbix-admin,ou=Groups,dc=example,dc=com` To pole jest dostępne tylko dla metody memberOf.
User name attribute	Określ atrybut zawierający imię użytkownika.
User last name attribute	Określ atrybut zawierający nazwisko użytkownika.
User group mapping	Mapuj wzorzec grupy użytkowników LDAP na grupę użytkowników Zabbix i rolę użytkownika. Jest to wymagane do określenia, jaką grupę/rolę otrzyma aprowizowany użytkownik w Zabbix. Kliknij *Add*, aby dodać mapowanie. Pole *LDAP group pattern* obsługuje symbole wieloznaczne. Nazwa grupy musi odpowiadać istniejącej grupie. Jeśli użytkownik LDAP pasuje do kilku grup użytkowników Zabbix, stanie się członkiem wszystkich tych grup. Jeśli użytkownik pasuje do kilku ról użytkownika Zabbix, otrzyma tę z nich, która ma najwyższy poziom uprawnień.
Media type mapping	Mapuj atrybuty LDAP użytkownika mediów (np. email) na media użytkownika Zabbix do wysyłania powiadomień (wartość atrybutu jest używana jako pole mediów *Send to*).
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji (patrz poniżej).
StartTLS	Zaznacz pole wyboru, aby użyć operacji StartTLS podczas łączenia z serwerem LDAP. Połączenie nie powiedzie się, jeśli serwer nie obsługuje StartTLS. StartTLS nie może być używany z serwerami korzystającymi z protokołu *ldaps*.

Parameter	Description
Search filter	Zdefiniuj własny ciąg podczas uwierzytelniania użytkownika w LDAP. Obsługiwane są następujące symbole zastępcze: `%{attr}` - nazwa atrybutu wyszukiwania (uid, sAMAccountName) `%{user}` - wartość nazwy użytkownika do uwierzytelnienia Na przykład, aby przeprowadzić wyszukiwanie z rozróżnianiem wielkości liter w środowisku LDAP lub Microsoft Active Directory niewrażliwym na wielkość liter, ciąg można zdefiniować następująco: `(%{attr}:caseExactMatch:=%{user})`. Jeśli filtr nie zostanie dostosowany, LDAP użyje domyślnego: `(%{attr}=%{user})`.

Note:
Aby skonfigurować serwer LDAP do **bezpośredniego wiązania użytkownika**, dodaj atrybut uid=%{user} do parametru *Base DN* (na przykład *uid=%{user},dc=example,dc=com*) i pozostaw parametry *BindDN* oraz *Bind password* puste. Podczas uwierzytelniania symbol zastępczy %{user} zostanie zastąpiony nazwą użytkownika wprowadzoną podczas logowania.
 Przy bezpośrednim wiązaniu użytkownika Zabbix ma dostęp do poświadczeń użytkownika tylko podczas interaktywnego logowania. Dlatego zadania aprowizacji działające bez interaktywnego logowania (na przykład przycisk *Provision now* lub metoda API `user.provision`) będą ignorować użytkowników uwierzytelniających się za pomocą bezpośredniego wiązania użytkownika, ponieważ Zabbix nie może wykonać wiązania w ich imieniu. Użyj anonimowego wiązania lub dedykowanego użytkownika bind (konta usługi), jeśli potrzebujesz, aby aprowizacja i wyszukiwania LDAP działały z frontend bez konieczności zalogowania użytkownika końcowego.

Następujące pola są specyficzne dla metody *Group configuration* „groupOfNames”:



Parameter	Description
Group base DN	Ścieżka bazowa do grup na serwerze LDAP.
Group name attribute	Określ atrybut, z którego ma być pobierana nazwa grupy w określonej ścieżce bazowej do grup. Nazwa grupy jest wymagana do mapowania grup użytkowników.
Group member attribute	Określ atrybut zawierający informacje o członkach grupy w LDAP (np. `member`).
Reference attribute	Określ atrybut referencyjny dla filtra grupy (zobacz pole *Group filter*). Następnie użyj `%{ref}` w filtrze grupy, aby pobrać wartości dla określonego tutaj atrybutu.
Group filter	Określ filtr służący do pobrania grupy, której członkiem jest użytkownik. Na przykład `(member=uid=%{ref},ou=Users,dc=example,dc=com)` dopasuje „User1”, jeśli atrybut member grupy ma wartość `uid=User1,ou=Users,dc=example,dc=com`, i zwróci grupę, której członkiem jest „User1”.

Warning:
W przypadku problemów z certyfikatami, aby bezpieczne połączenie LDAP (ldaps) działało, może być konieczne dodanie wiersza `TLS_REQCERT allow` do pliku konfiguracyjnego /etc/openldap/ldap.conf. Może to obniżyć bezpieczeństwo połączenia z katalogiem LDAP.

Note:
Zaleca się utworzenie osobnego konta LDAP (*Bind DN*) do wykonywania wiązania i wyszukiwania na serwerze LDAP z minimalnymi uprawnieniami w LDAP, zamiast używania rzeczywistych kont użytkowników (używanych do logowania do frontend Zabbix).
 Takie podejście zapewnia większe bezpieczeństwo i nie wymaga zmiany *Bind password*, gdy użytkownik zmieni własne hasło na serwerze LDAP.
 W powyższej tabeli jest to nazwa konta *ldap_search*.

Testowanie dostępu

Przycisk *Test* umożliwia przetestowanie dostępu użytkownika:

Parameter	Description
Login	Nazwa użytkownika LDAP do przetestowania (wstępnie wypełniona bieżącą nazwą użytkownika z frontend Zabbix). Ta nazwa użytkownika musi istnieć na serwerze LDAP. Zabbix nie aktywuje uwierzytelniania LDAP, jeśli nie będzie w stanie uwierzytelnić testowanego użytkownika.
User password	Hasło użytkownika LDAP do przetestowania.

3 SAML

Przegląd

Uwierzytelnianie SAML 2.0 authentication może być używane do logowania do Zabbix.

Jeśli skonfigurowano tylko logowanie SAML, użytkownik musi również istnieć w Zabbix, jednak jego hasło Zabbix nie będzie używane. Jeśli uwierzytelnienie zakończy się powodzeniem, Zabbix dopasuje lokalną nazwę użytkownika do atrybutu nazwy użytkownika zwróconego przez SAML.

Provisioning użytkowników

Możliwe jest skonfigurowanie JIT (just-in-time) **provisioning użytkowników** dla użytkowników SAML. W takim przypadku nie jest wymagane, aby użytkownik już istniał w Zabbix. Konto użytkownika może zostać utworzone w momencie, gdy użytkownik zaloguje się do Zabbix po raz pierwszy.

Attention:
Jeśli JIT provisioning jest włączony, w zakładce *Authentication* należy określić grupę użytkowników dla zdeprowizjonowanych użytkowników.

Oprócz JIT provisioning można również włączyć i skonfigurować SCIM (System for Cross-domain Identity Management) provisioning - *ciągłe* zarządzanie kontami użytkowników dla tych użytkowników, którzy zostali utworzeni przez provisioning użytkowników. SCIM provisioning wymaga tokenu API Zabbix (z uprawnieniami Super admin) do uwierzytelniania w Zabbix.

Na przykład, jeśli użytkownik zostanie przeniesiony z jednej grupy SAML do innej, zostanie również przeniesiony z jednej grupy do innej w Zabbix; jeśli użytkownik zostanie usunięty z grupy SAML, zostanie również usunięty z grupy w Zabbix i, jeśli nie należy do żadnej innej grupy, zostanie dodany do grupy użytkowników dla zdeprowizjonowanych użytkowników.

Jeśli SCIM jest włączony i skonfigurowany, użytkownik SAML zostanie aprowizowany w momencie logowania do Zabbix i będzie na bieżąco aktualizowany na podstawie zmian w SAML. Istniejący już użytkownicy SAML nie zostaną aprowizowani, a aktualizowani będą wyłącznie aprowizowani użytkownicy. Należy pamiętać, że podczas aprowizacji lub aktualizacji użytkownika zostaną dodane tylko prawidłowe media.

Jeśli SCIM nie jest włączony, użytkownik SAML zostanie aprowizowany (a później zaktualizowany) w momencie logowania do Zabbix.

Note:
Jeśli uwierzytelnianie SAML jest włączone, użytkownicy będą mogli wybrać między logowaniem lokalnym a logowaniem przez jednokrotne logowanie SAML. Jeśli używany jest JIT provisioning, możliwe jest tylko jednokrotne logowanie.

Konfiguracja dostawcy tożsamości

Aby współpracować z Zabbix, dostawca tożsamości SAML (onelogin.com, auth0.com, okta.com itp.) musi zostać skonfigurowany w następujący sposób:

- *Assertion Consumer URL* powinien być ustawiony na `<path_to_zabbix_ui>/index_sso.php?acs`
- *Single Logout URL* powinien być ustawiony na `<path_to_zabbix_ui>/index_sso.php?sls`

Przykłady `<path_to_zabbix_ui>`: `https://example.com/zabbix/ui`, `http://another.example.com/zabbix`, `http://<any_public_ip_address>/zabbix`

Konfiguracja Zabbix

> **Attention:**
> Wymagane jest zainstalowanie php-openssl, jeśli chcesz używać uwierzytelniania SAML we frontendzie.

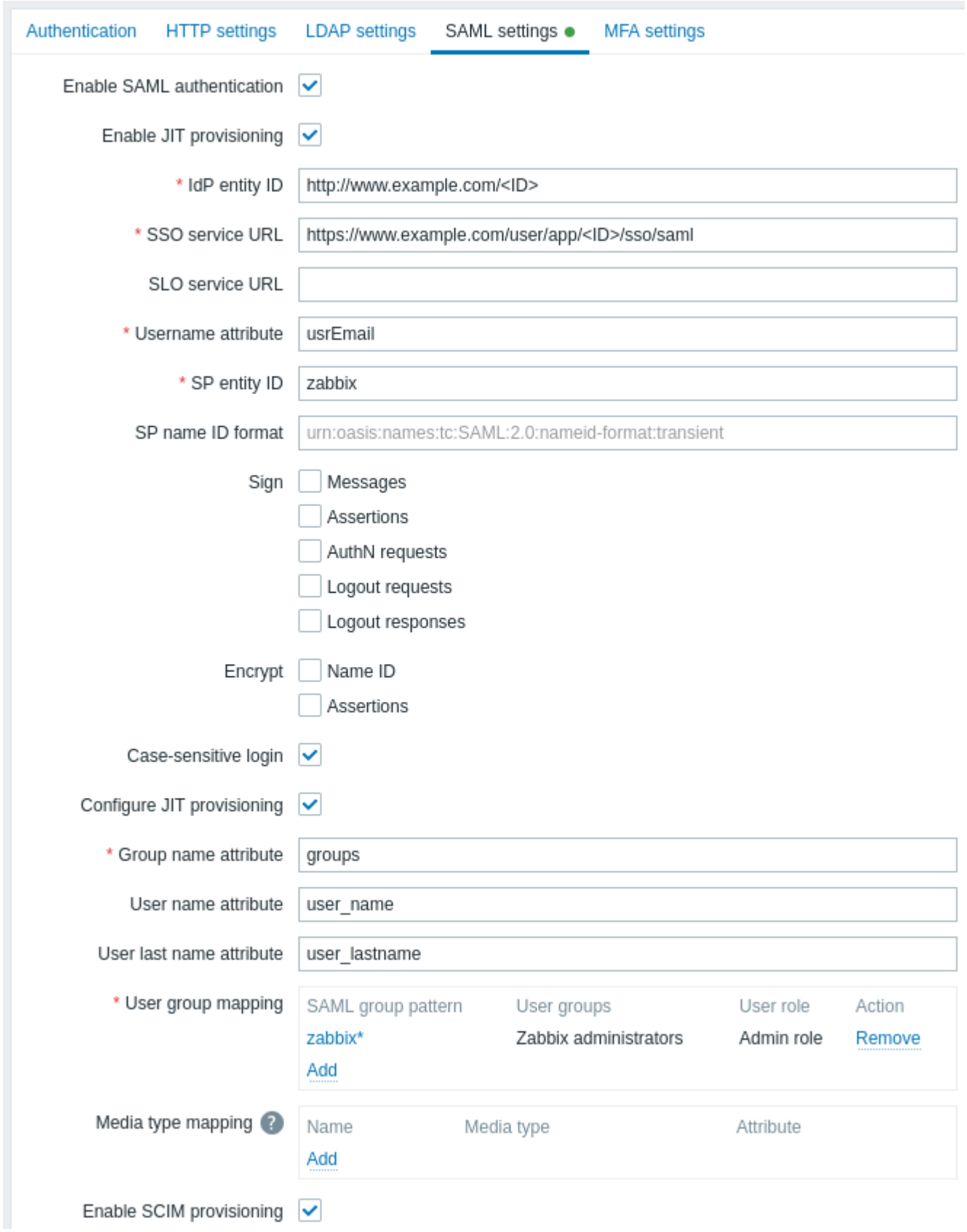
Aby używać uwierzytelniania SAML, Zabbix powinien być skonfigurowany w następujący sposób:

1. Klucz prywatny i certyfikat powinny być przechowywane w *ui/conf/certs/*, chyba że podano niestandardowe ścieżki w zabbix.conf.php.

Domyślnie Zabbix będzie szukać w następujących lokalizacjach:

- ui/conf/certs/sp.key - plik klucza prywatnego SP
- ui/conf/certs/sp.crt - plik certyfikatu SP
- ui/conf/certs/idp.crt - plik certyfikatu IDP

2. Wszystkie najważniejsze ustawienia można skonfigurować we frontendzie Zabbix. Możliwe jest jednak określenie dodatkowych ustawień w pliku konfiguracyjnym.



Parametry konfiguracji dostępne we frontendzie Zabbix:

Parameter	Description
Enable SAML authentication	Zaznacz pole wyboru, aby włączyć uwierzytelnianie SAML.
Enable JIT provisioning	Zaznacz pole wyboru, aby włączyć aprowizację użytkowników JIT.
IDP entity ID	Unikalny identyfikator encji w dostawcy tożsamości SAML.
SSO service URL	Adres URL, na który użytkownicy zostaną przekierowani podczas logowania.

Parameter	Description
SLO service URL	Adres URL, na który użytkownicy zostaną przekierowani podczas wylogowywania. Jeśli pozostanie puste, usługa SLO nie będzie używana.
Username attribute	Atrybut SAML używany jako nazwa użytkownika podczas logowania do Zabbix. Lista obsługiwanych wartości jest określana przez dostawcę tożsamości. Przykłady: 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	Unikalny identyfikator dostawcy usług (jeśli nie będzie zgodny, operacja zostanie odrzucona). Można określić adres URL lub dowolny ciąg danych.
SP name ID format	Zażądaj określonego formatu name ID w odpowiedzi. Przykłady: urn:oasis:names:tc:SAML:2.0:nameid-format:persistent urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified urn:oasis:names:tc:SAML:2.0:nameid-format:transient
Sign	Zaznacz pola wyboru, aby wybrać encje, dla których ma być włączony podpis SAML: *Messages* *Assertions* *AuthN requests* *Logout requests* *Logout responses*
Encrypt	Zaznacz pola wyboru, aby wybrać encje, dla których ma być włączone szyfrowanie SAML: *Name ID* *Assertions*
Case-sensitive login	Odznacz pole wyboru, aby wyłączyć rozróżnianie wielkości liter w nazwach użytkowników (domyślnie włączone). Wyłączenie rozróżniania wielkości liter pozwala na przykład zalogować się jako ”admin”, nawet jeśli użytkownik Zabbix to ”Admin” lub ”ADMIN”. Pamiętaj, że jeśli rozróżnianie wielkości liter jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach (np. Admin i admin), logowanie tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: ”Authentication failed: supplied credentials are not unique.”
Configure JIT provisioning	Zaznacz to pole wyboru, aby wyświetlić opcje związane z aprowizacją użytkowników JIT.
Group name attribute	Określ atrybut nazwy grupy dla aprowizacji użytkowników JIT.
User name attribute	Określ atrybut nazwy użytkownika dla aprowizacji użytkowników JIT.
User last name attribute	Określ atrybut nazwiska użytkownika dla aprowizacji użytkowników JIT.
User group mapping	Mapuj wzorzec grupy użytkowników SAML na grupę użytkowników i rolę użytkownika Zabbix. Jest to wymagane do określenia, jaką grupę/rolę otrzyma aprowizowany użytkownik w Zabbix. Kliknij *Add*, aby dodać mapowanie. Pole *SAML group pattern* obsługuje symbole wieloznaczne. Nazwa grupy musi odpowiadać istniejącej grupie. Jeśli użytkownik SAML pasuje do kilku grup użytkowników Zabbix, stanie się członkiem wszystkich z nich. Jeśli użytkownik pasuje do kilku ról użytkowników Zabbix, otrzyma najwyższy poziom uprawnień spośród nich.
Media type mapping	Mapuj atrybuty mediów SAML użytkownika (np. e-mail) na media użytkownika Zabbix do wysyłania powiadomień.
Enable SCIM provisioning	Zaznacz to pole wyboru, aby włączyć aprowizację SCIM 2.0.

Przykłady konfiguracji dostawców tożsamości SAML do logowania i aprowizacji użytkowników w Zabbix znajdziesz w:

- Microsoft Azure AD

- Okta
- Onelogin

Uwagi dotyczące provisioningu SCIM

W przypadku provisioningu SCIM należy określić ścieżkę do frontend Zabbix i dodać do niej `api_scim.php` po stronie dostawcy tożsamości, tj.:

`https://<path-to-zabbix-ui>/api_scim.php`

Atrybuty użytkownika używane w Zabbix (nazwa użytkownika, imię użytkownika, nazwisko użytkownika oraz atrybuty mediów) należy dodać jako atrybuty niestandardowe, a w razie potrzeby zewnętrzna przestrzeń nazw powinna być taka sama jak w schemacie użytkownika: `urn:ietf:params:scim:schemas:core:2.0:User`.

Ustawienia zaawansowane

Dodatkowe parametry SAML można skonfigurować w pliku konfiguracyjnym frontend Zabbix (*zabbix.conf.php*):

- $SSO['SP_KEY'] = '<ścieżka do pliku klucza prywatnego SP>';
- $SSO['SP_CERT'] = '<ścieżka do pliku certyfikatu SP>';
- $SSO['IDP_CERT'] = '<ścieżka do pliku certyfikatu IDP>';
- $SSO['SETTINGS']

Note:
Tablica `$SSO['SETTINGS']` musi mieć taką samą strukturę, jakiej oczekuje biblioteka *SAML PHP Toolkit* (dostarczana z Zabbix). Pełny opis dostępnych opcji konfiguracji znajduje się w oficjalnej dokumentacji biblioteki.

Tylko następujące opcje mogą być ustawiane w ramach $SSO['SETTINGS']:

- *strict*
- *baseurl*
- *compress*
- *contactPerson*
- *organization*
- *sp* (tylko opcje określone na tej liście)
  - *attributeConsumingService*
  - *x509certNew*
- *idp* (tylko opcje określone na tej liście)
  - *singleLogoutService* (tylko jedna opcja)
    - *responseUrl*
  - *certFingerprint*
  - *certFingerprintAlgorithm*
  - *x509certMulti*
- *security* (tylko opcje określone na tej liście)
  - *signMetadata*
  - *wantNameId*
  - *requestedAuthnContext*
  - *requestedAuthnContextComparison*
  - *wantXMLValidation*
  - *relaxDestinationValidation*
  - *destinationStrictlyMatches*
  - *rejectUnsolicitedResponsesWithInResponseTo*
  - *signatureAlgorithm*
  - *digestAlgorithm*
  - *lowercaseUrlencoding*

Wszystkie pozostałe opcje będą pobierane z bazy danych i nie mogą zostać nadpisane. Opcja *debug* zostanie zignorowana.

Dodatkowo, jeśli interfejs użytkownika Zabbix znajduje się za proxy lub load balancerem, można użyć niestandardowej opcji *use_proxy_headers*:

- *false* (domyślnie) - ignoruj opcję;
- *true* - używaj nagłówków HTTP X-Forwarded-* do budowania bazowego URL.

Jeśli do połączenia z instancją Zabbix używany jest load balancer, który korzysta z TLS/SSL, a Zabbix nie, należy określić parametry 'baseurl', 'strict' i 'use_proxy_headers' w następujący sposób:

```
$SSO['SETTINGS'] = [
    'strict' => false,
```

```
    'baseurl' => 'https://zabbix.example.com/zabbix/',
    'use_proxy_headers' => true
];
```

Przykład konfiguracji:

```
$SSO['SETTINGS'] = [
    'security' => [
        'signatureAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha384'
        'digestAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#sha384',
        // ...
    ],
    // ...
];
```

Konfiguracja frontend z Kerberos/ADFS

Plik konfiguracji frontend Zabbixa (*zabbix.conf.php*) można wykorzystać do skonfigurowania SSO z uwierzytelnianiem Kerberos i ADFS:

```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => [
            'urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos',
        ],
        'requestedAuthnContextComparison' => 'exact'
    ]
];
```

W takim przypadku w konfiguracji SAML w polu *SP name ID* ustaw:

`urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified`

4 MFA

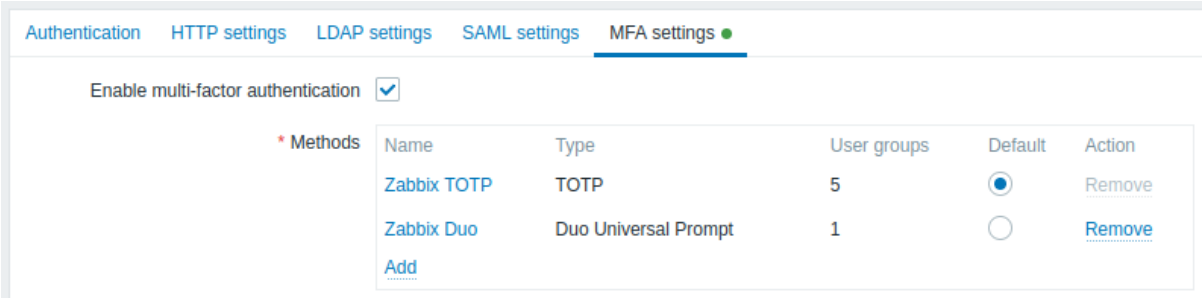
Przegląd

Uwierzytelnianie wieloskładnikowe authentication (MFA) może być używane do logowania do Zabbix, zapewniając dodatkową warstwę bezpieczeństwa poza samą nazwą użytkownika i hasłem.

W przypadku MFA użytkownik musi istnieć w Zabbix, musi podać poświadczenia Zabbix podczas logowania, a także musi potwierdzić swoją tożsamość w inny sposób, zazwyczaj za pomocą kodu wygenerowanego przez aplikację uwierzytelniającą na telefonie użytkownika.

Dostępnych jest wiele metod MFA, co pozwala użytkownikom wybrać opcję najlepiej odpowiadającą ich wymaganiom i preferencjom w zakresie bezpieczeństwa. Metody te to jednorazowe hasło czasowe (TOTP) oraz Duo Universal Prompt.

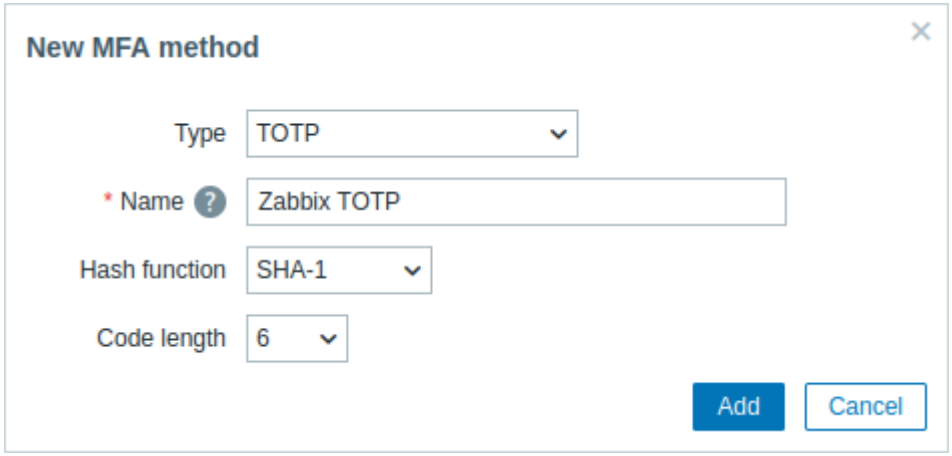
Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Włącz uwierzytelnianie wieloskładnikowe	Zaznacz pole wyboru, aby włączyć uwierzytelnianie wieloskładnikowe.
Metody	Kliknij *Dodaj*, aby skonfigurować metodę MFA (zobacz konfigurację metody poniżej).

Konfiguracja metody



Parametry konfiguracji metody:

Parameter	Description
Type	Wybierz typ metody MFA: **TOTP** — użyj aplikacji uwierzytelniającej do generowania jednorazowych haseł opartych na czasie; **Duo Universal Prompt** — użyj usługi uwierzytelniania Duo do zapewnienia uwierzytelniania wieloskładnikowego.
Name	Wprowadź nazwę wyświetlaną jako nazwa konta dla wszystkich użytkowników MFA w aplikacjach uwierzytelniających (na przykład „Zabbix”).
Hash function	Wybierz funkcję skrótu (SHA-1, SHA-256 lub SHA-512) do generowania kodów TOTP. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „TOTP”. Należy pamiętać, że wybór SHA-256 lub SHA-512 może znacznie ograniczyć kompatybilność, ponieważ wiele aplikacji obecnie nie obsługuje tych funkcji.
Code length	Wybierz długość kodu weryfikacyjnego (6 lub 8). Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „TOTP”.
API hostname	Wprowadź nazwę hosta API podaną przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.
Client ID	Wprowadź identyfikator klienta podany przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.
Client secret	Wprowadź klucz tajny klienta podany przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.

Przykłady konfiguracji

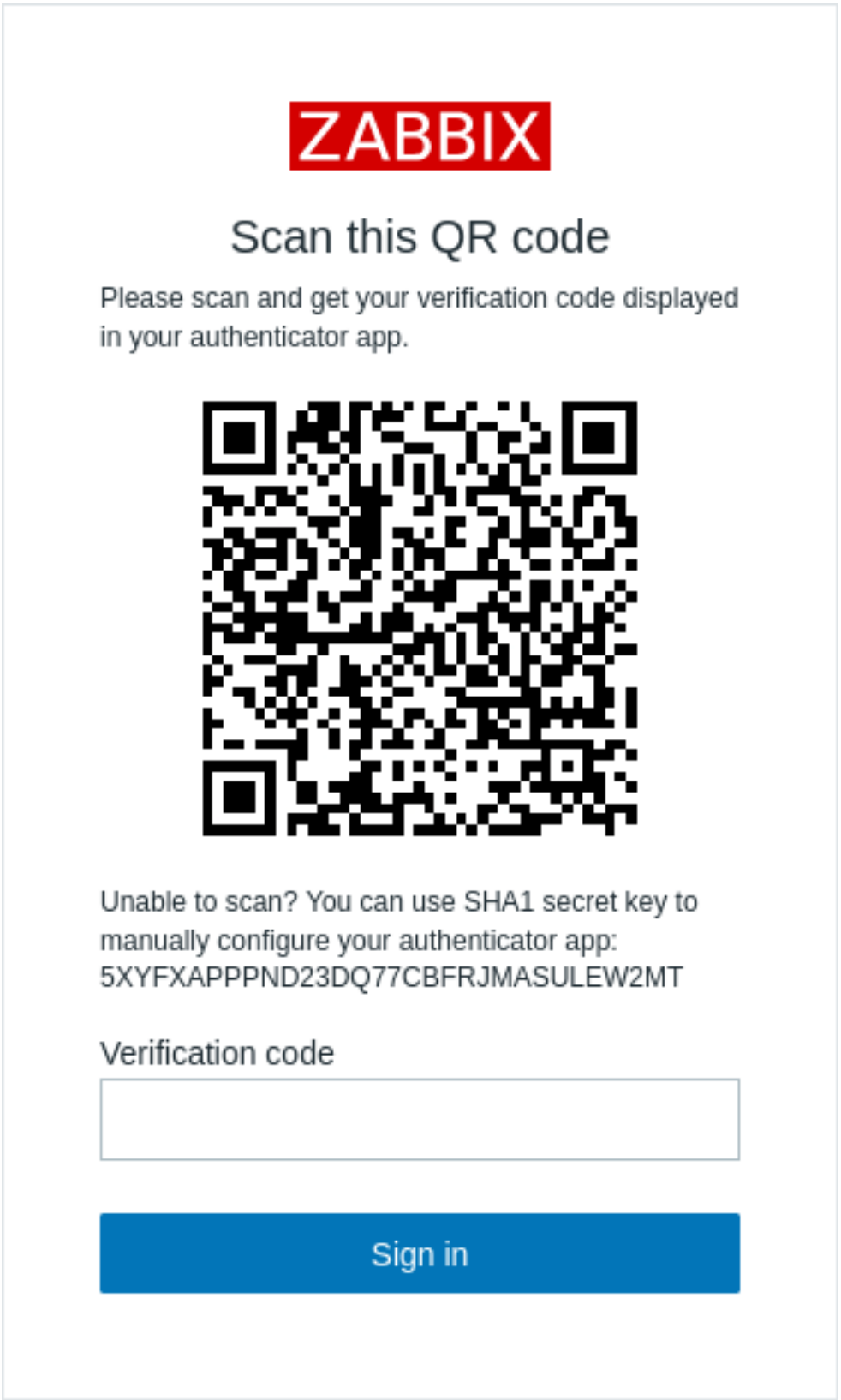
Ta sekcja zawiera przykłady konfiguracji MFA z użyciem Time-Based One-Time Password (TOTP) oraz Duo Universal Prompt.

TOTP

W przypadku TOTP użytkownicy muszą zweryfikować swoją tożsamość za pomocą aplikacji uwierzytelniającej (na przykład aplikacji Google Authenticator).

1. Przejdź do ustawień MFA w Zabbix w sekcji *Users* → *Authentication* i włącz uwierzytelnianie wieloskładnikowe.
2. Dodaj nową metodę MFA z następującą konfiguracją:
   - Typ: TOTP
   - Nazwa: Zabbix TOTP
   - Funkcja skrótu: SHA-1
   - Długość kodu: 6
3. Kliknij przycisk *Add*, a następnie przycisk *Update*.
4. Przejdź do *Users* → *User groups* i utwórz nową grupę użytkowników z następującą konfiguracją:
   - Nazwa grupy: grupa TOTP
   - Użytkownicy: Admin
   - Uwierzytelnianie wieloskładnikowe: Default (lub „Zabbix TOTP”, jeśli nie jest ustawione jako domyślne)

5. Wyloguj się z Zabbix i zaloguj się ponownie, używając swoich poświadczeń. Po pomyślnym zalogowaniu zostaniesz poproszony o zapisanie się do MFA; zostanie wyświetlony kod QR oraz tajny klucz.



6. Zeskanuj kod QR lub wprowadź tajny klucz do aplikacji Google Authenticator. Aplikacja wygeneruje kod weryfikacyjny, który należy wprowadzić, aby zakończyć proces logowania.

7. Przy kolejnych logowaniach pobierz kod weryfikacyjny z aplikacji Google Authenticator i wprowadź go podczas logowania.

Duo Universal Prompt

W przypadku Duo Universal Prompt użytkownicy muszą zweryfikować swoją tożsamość za pomocą aplikacji uwierzytelniającej Duo Mobile.

Attention:
Metoda MFA Duo Universal Prompt wymaga zainstalowania rozszerzenia php-curl, dostępu do Zabbix przez HTTPS oraz uprawnień do połączeń wychodzących do serwerów Duo. Ponadto, jeśli włączono Content Security Policy (CSP) na serwerze WWW, należy dodać „duo.com" do dyrektywy CSP w pliku konfiguracyjnym hosta wirtualnego.

1. Zarejestruj bezpłatne konto administratora Duo na stronie Duo Signup.
2. Otwórz panel administracyjny Duo, przejdź do *Applications* → *Protect an Application*, wyszukaj aplikację *Web SDK* i kliknij *Protect*.
3. Zanotuj dane uwierzytelniające (Client ID, Client secret, API hostname) wymagane do skonfigurowania metody MFA w Zabbix.
4. Przejdź do ustawień MFA w Zabbix w sekcji *Users* → *Authentication* i włącz uwierzytelnianie wieloskładnikowe.
5. Dodaj nową metodę MFA z następującą konfiguracją:
   - Typ: Duo Universal Prompt
   - Nazwa: Zabbix Duo
   - API hostname: (użyj API hostname z Duo)
   - Client ID: (użyj Client ID z Duo)
   - Client secret: (użyj Client secret z Duo)
6. Kliknij przycisk *Add*, a następnie przycisk *Update*.
7. Przejdź do *Users* → *User groups* i utwórz nową grupę użytkowników z następującą konfiguracją:
   - Nazwa grupy: Duo group
   - Użytkownicy: Admin
   - Uwierzytelnianie wieloskładnikowe: Default (lub „Zabbix Duo", jeśli nie jest ustawione jako domyślne)
8. Wyloguj się z Zabbix i zaloguj się ponownie, używając swoich danych uwierzytelniających. Po pomyślnym zalogowaniu pojawi się monit o zapisanie się do MFA i nastąpi przekierowanie do Duo. Dokończ konfigurację Duo i zweryfikuj użytkownika za pomocą aplikacji Duo na telefonie, aby się zalogować.
9. Przy kolejnych logowaniach użyj odpowiedniej metody MFA udostępnianej przez aplikację Duo (takiej jak pobranie kodu weryfikacyjnego, odpowiedź na powiadomienia push lub użycie kluczy sprzętowych) i wprowadź wymagane informacje podczas logowania.

9 Administracja

Przegląd

Menu Administracja służy do funkcji administracyjnych Zabbix. To menu jest dostępne wyłącznie dla użytkowników typu SuperAdmin.

1 Informacje ogólne

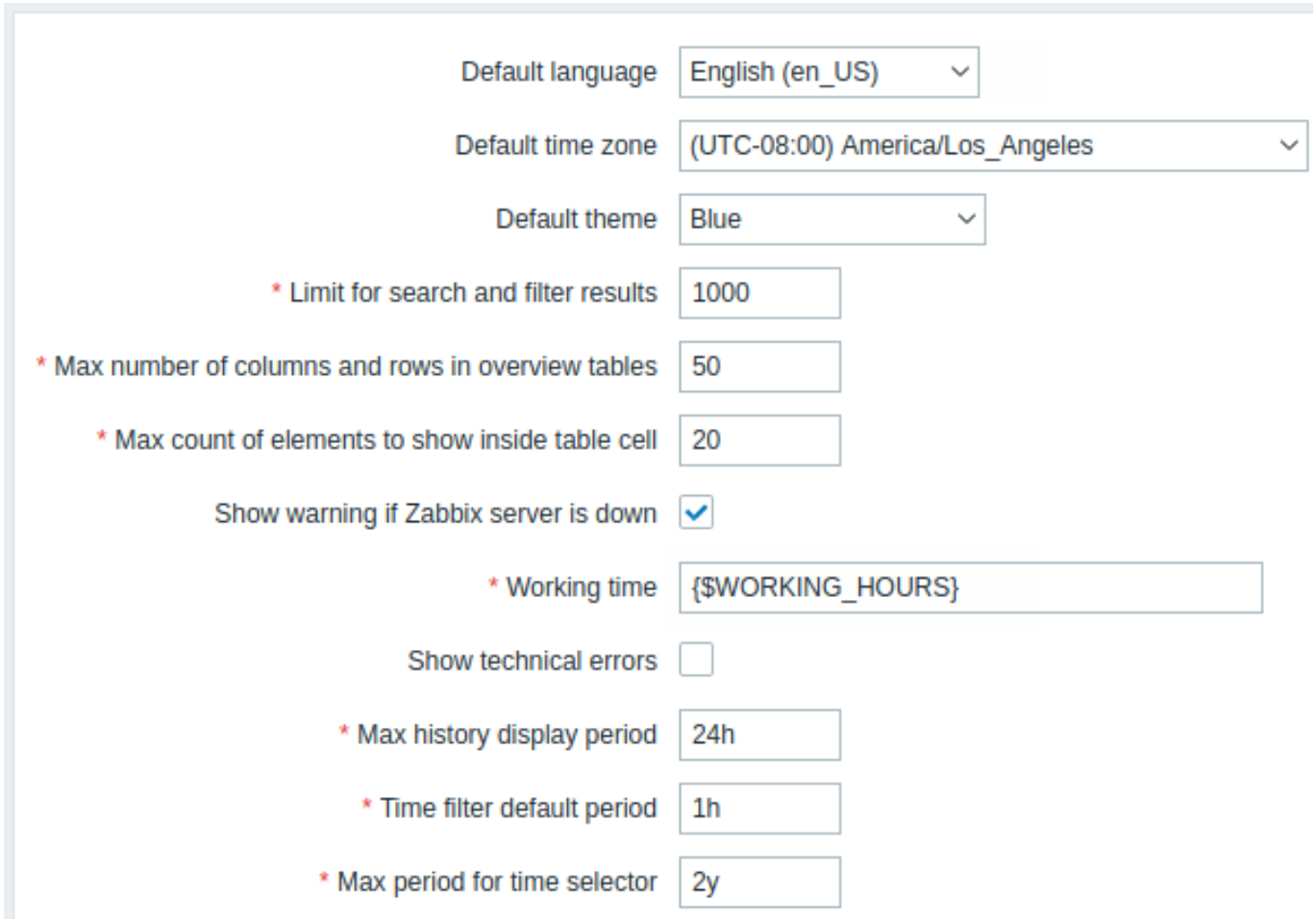
Przegląd

Sekcja *Administration* > *General* zawiera szereg podsekcji służących do ustawiania domyślnych wartości związanych z frontendem oraz dostosowywania Zabbixa.

Lista dostępnych podsekcji pojawia się po kliknięciu *General* w sekcji menu *Administration*. Możliwe jest również przełączanie się między podsekcjami za pomocą listy rozwijanej tytułu w lewym górnym rogu.

GUI

Ta sekcja umożliwia dostosowanie kilku domyślnych ustawień związanych z frontendem.



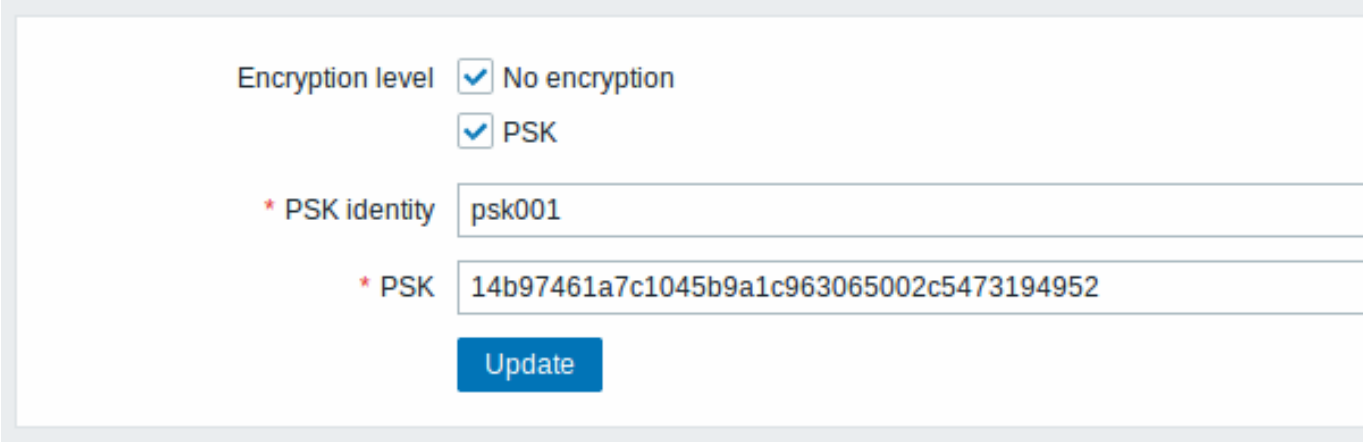
Parametry konfiguracji:

Parameter	Description
Default language	Domyślny język dla użytkowników, którzy nie określili języka w swoich profilach, oraz dla użytkowników gościnnych. Więcej informacji można znaleźć w sekcji Installation of additional frontend languages.
Default time zone	Domyślna strefa czasowa dla użytkowników, którzy nie określili strefy czasowej w swoich profilach, oraz dla użytkowników gościnnych.
Default theme	Domyślny motyw dla użytkowników, którzy nie określili motywu w swoich profilach, oraz dla użytkowników gościnnych.
Limit for search and filter results	Maksymalna liczba elementów (wierszy), które będą wyświetlane na liście w interfejsie webowym, na przykład w *Zbieranie danych* > *Hosts*. *Uwaga*: jeśli ustawiono na przykład wartość „50”, tylko pierwsze 50 elementów będzie wyświetlane na wszystkich listach frontend, których to dotyczy. Jeśli jakaś lista zawiera więcej niż pięćdziesiąt elementów, zostanie to oznaczone znakiem „+” w komunikacie *”Displaying 1 to 50 of **50+** found”*. Ponadto, jeśli używane jest filtrowanie i nadal istnieje więcej niż 50 dopasowań, wyświetlonych zostanie tylko pierwszych 50. Należy pamiętać, że zwiększenie wartości tego parametru może prowadzić do pogorszenia wydajności i zwiększonego zużycia pamięci po stronie serwera WWW.
Max number of columns and rows in overview tables	Maksymalna liczba kolumn i wierszy wyświetlanych w widżecie pulpitu *Trigger overview*. Ten sam limit dotyczy zarówno kolumn, jak i wierszy. Jeśli istnieje więcej wierszy i/lub kolumn niż wyświetlanych, system pokaże ostrzeżenie na dole tabeli: ”Not all results are displayed. Please provide more specific search criteria.”
Max count of elements to show inside table cell	W przypadku wpisów wyświetlanych w pojedynczej komórce tabeli zostanie pokazana nie więcej niż liczba skonfigurowana tutaj.
Show warning if Zabbix server is down	Ten parametr włącza wyświetlanie komunikatu ostrzegawczego w oknie przeglądarki, jeśli nie można połączyć się z serwerem Zabbix (możliwe, że nie działa). Komunikat pozostaje widoczny nawet po przewinięciu strony w dół. Po najechaniu kursorem jest tymczasowo ukrywany, aby odsłonić treść znajdującą się pod nim.

Parameter	Description
Working time	Ten parametr obowiązujący w całym systemie definiuje godziny pracy. Na wykresach czas pracy jest wyświetlany jako białe tło, a czas poza pracą jako szary. Opis formatu czasu znajduje się na stronie Time period specification. Obsługiwane są makra użytkownika.
Show technical errors	Jeśli zaznaczone, wszyscy zarejestrowani użytkownicy będą mogli widzieć błędy techniczne (PHP/SQL). Jeśli odznaczone, informacje te są dostępne tylko dla Zabbix Super Admins oraz użytkowników należących do grup użytkowników z włączonym trybem debugowania.
Max history display period	Maksymalny okres, dla którego będą wyświetlane dane historyczne w *Monitoring* > *Latest data*, informacje o web scenario hosta w *Monitoring* > *Hosts* oraz w widżecie pulpitu *Top items*. Dozwolony zakres: 24 godziny (domyślnie) - 1 tydzień. Obsługiwane są sufiksy czasu, np. 1w (jeden tydzień), 36h (36 godzin).
Time filter default period	Domyślny okres czasu używany na wykresach i pulpitach. Dozwolony zakres: 1 minuta - 10 lat (domyślnie: 1 godzina). Obsługiwane są sufiksy czasu, np. 10m (dziesięć minut), 5w (pięć tygodni). Uwaga: gdy użytkownik zmieni okres czasu podczas przeglądania wykresu, okres ten zostanie zapisany jako preferencja użytkownika, zastępując globalną wartość domyślną lub wcześniejszy wybór użytkownika.
Max period for time selector	Maksymalny dostępny okres czasu dla wykresów i pulpitów. Użytkownicy nie będą mogli wizualizować starszych danych. Dozwolony zakres: 1 rok - 10 lat (domyślnie: 2 lata). Obsługiwane są sufiksy czasu, np. 1y (jeden rok), 365w (365 tygodni).

Autorejestracja

W tej sekcji można skonfigurować poziom szyfrowania dla autorejestracji aktywnego agent.



Parametry oznaczone gwiazdką są obowiązkowe.

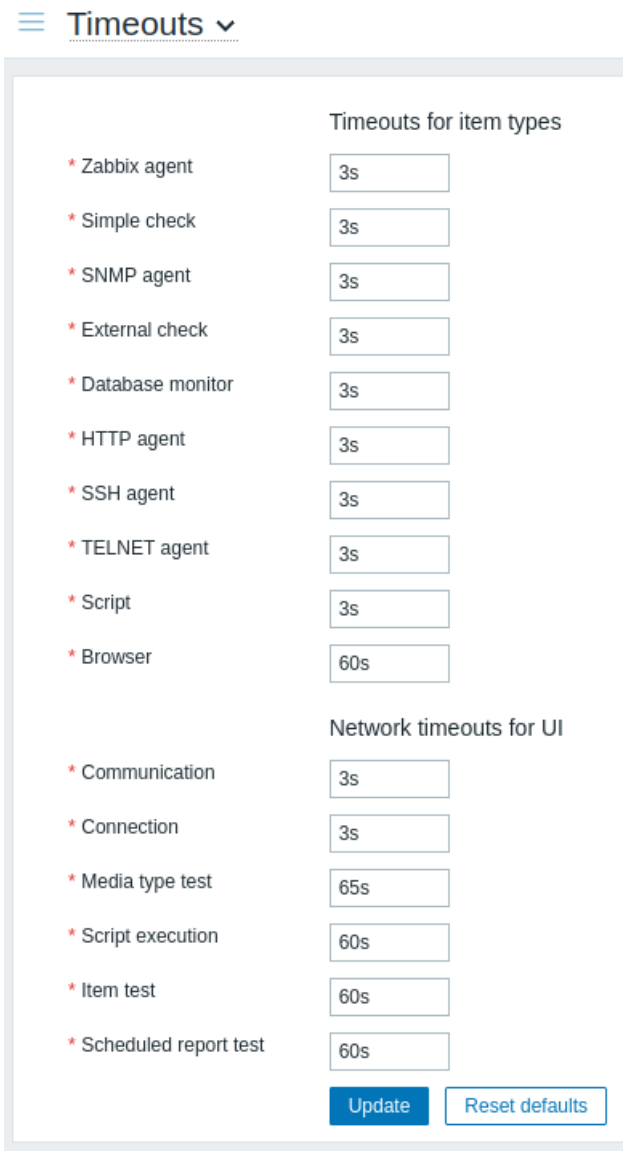
Parametry konfiguracji:

Parameter	Description
Poziom szyfrowania	Wybierz jedną lub obie opcje poziomu szyfrowania: **Brak szyfrowania** - dozwolone są nieszyfrowane połączenia **PSK** - dozwolone są szyfrowane połączenia TLS z użyciem klucza wstępnie współdzielonego
Tożsamość PSK	Wprowadź ciąg tożsamości klucza wstępnie współdzielonego. To pole jest dostępne tylko wtedy, gdy jako *Poziom szyfrowania* wybrano opcję 'PSK'. Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć.
PSK	Wprowadź klucz wstępnie współdzielony (parzysta liczba znaków szesnastkowych). Maksymalna długość: 512 cyfr szesnastkowych (PSK o rozmiarze 256 bajtów), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, 64 cyfry szesnastkowe (PSK o rozmiarze 32 bajtów), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952 To pole jest dostępne tylko wtedy, gdy jako *Poziom szyfrowania* wybrano opcję 'PSK'.

Zobacz także: Secure autoregistration

Limity czasu

W tej sekcji można ustawić globalne limity czasu dla typów pozycji oraz limity czasu sieci. Wszystkie pola w tym formularzu są wymagane.

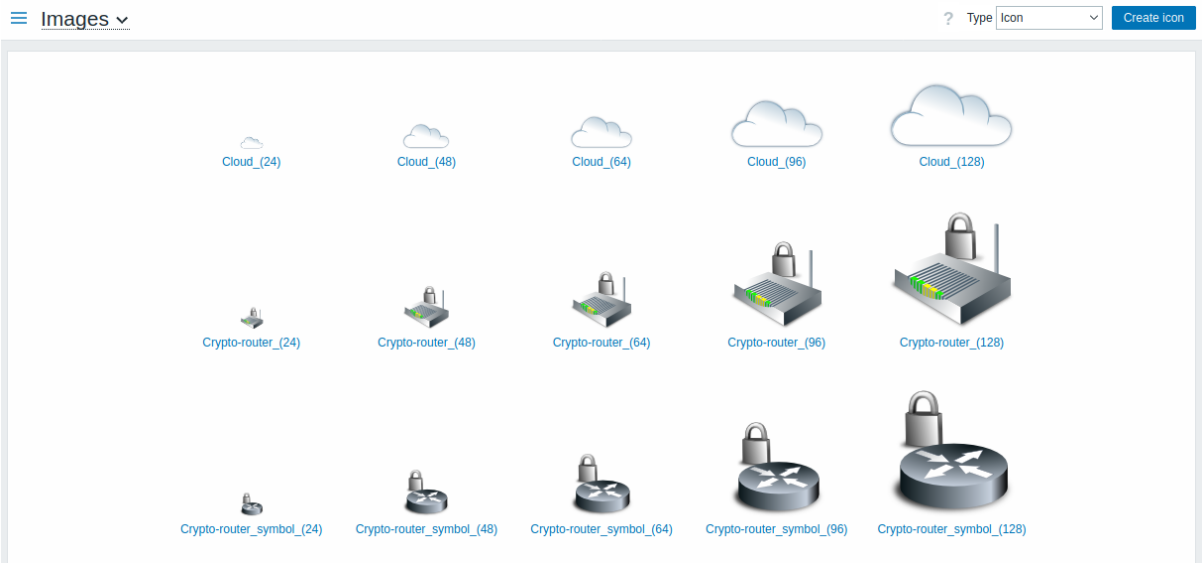


Parameter	Description
Limity czasu dla typów pozycji	Ile sekund czekać na odpowiedź od monitorowanej pozycji (na podstawie jej typu). Dozwolony zakres: 1 - 600 s (domyślnie: 3 s; domyślnie dla typu pozycji Browser: 60 s). Obsługiwane są sufiksy czasu, np. 30s, 1m, oraz makra użytkownika. Obsługiwane typy pozycji: - Zabbix agent (zarówno kontrole pasywne, jak i aktywne) - Simple check (z wyjątkiem pozycji `icmpping*`, `vmware.*`) - SNMP agent (tylko dla pozycji SNMP `walk[OID]` i `get[OID]`) - External check - Database monitor - HTTP agent - SSH agent - TELNET agent - Script - Browser Należy pamiętać, że jeśli używany jest proxy i ma on skonfigurowane limity czasu, ustawienia limitów czasu proxy zastąpią ustawienia globalne. Jeśli dla konkretnych pozycji ustawiono limity czasu, zastąpią one ustawienia proxy i globalne.
Limity czasu sieci dla UI	

Parameter		Description
	Communication	Ile sekund czekać przed zamknięciem bezczynnego gniazda (jeśli połączenie z serwerem Zabbix zostało wcześniej nawiązane, ale frontend nie może w tym czasie zakończyć operacji odczytu/wysyłania danych, połączenie zostanie przerwane). Dozwolony zakres: 1 - 300 s (domyślnie: 3 s).
	Connection	Ile sekund czekać przed przerwaniem próby połączenia z serwerem Zabbix. Dozwolony zakres: 1 - 30 s (domyślnie: 3 s).
	Media type test	Ile sekund czekać na odpowiedź podczas testowania typu mediów. Dozwolony zakres: 1 - 300 s (domyślnie: 65 s).
	Script execution	Ile sekund czekać na odpowiedź podczas wykonywania skryptu. Dozwolony zakres: 1 - 300 s (domyślnie: 60 s). Ten limit czasu dotyczy całego łańcucha wykonywania skryptu, który może mieć różną długość. Na przykład, jeśli skrypt jest wykonywany na agent, jest to podróż tam i z powrotem przez serwer (ewentualnie także proxy) do agent i z powrotem. W przypadku skryptów wykonywanych na agent obowiązuje limit czasu serwer lub proxy. W przypadku skryptów wykonywanych wyłącznie na aktywnym agent, prawdopodobnie trzeba zwiększyć domyślny limit czasu serwer/proxy. Limit czasu serwer/proxy musi być większy niż częstotliwość odświeżania aktywnej kontroli, w przeciwnym razie limit czasu serwer/proxy może zostać przekroczony, zanim aktywny agent w ogóle otrzyma skrypt. Zobacz także: Script timeout.
	Item test	Ile sekund czekać na zwrócone dane podczas testowania pozycji. Dozwolony zakres: 1 - 600 s (domyślnie: 60 s).
	Scheduled report test	Ile sekund czekać na zwrócone dane podczas testowania zaplanowanego raportu. Dozwolony zakres: 1 - 300 s (domyślnie: 60 s).

Obrazy

Sekcja Obrazy wyświetla wszystkie obrazy dostępne w Zabbix. Obrazy są przechowywane w bazie danych.

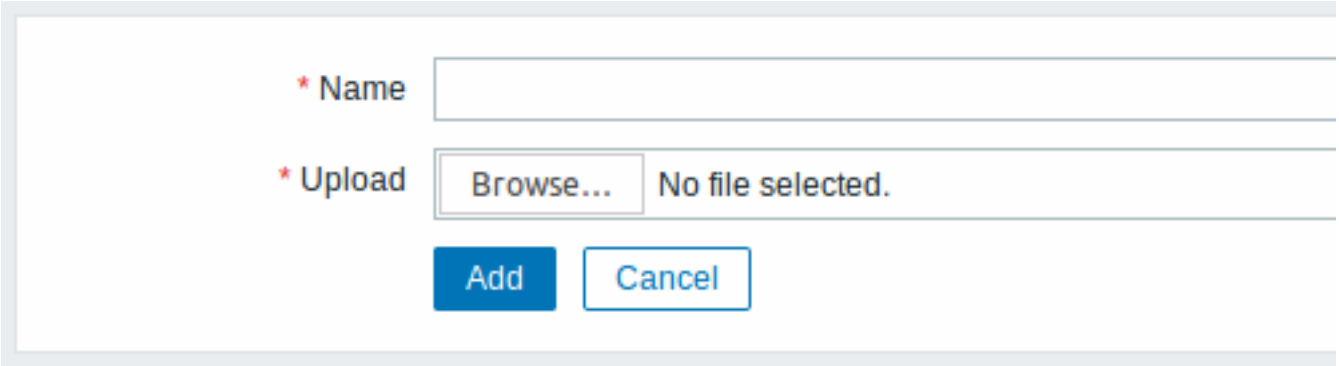


Lista rozwijana *Typ* umożliwia przełączanie między ikonami a obrazami tła:

- Ikony są używane do wyświetlania elementów mapy sieci.
- Tła są używane jako obrazy tła map sieci.

Dodawanie obrazu

Możesz dodać własny obraz, klikając przycisk *Utwórz ikonę* lub *Utwórz tło* w prawym górnym rogu.



Atrybuty obrazu:

Parameter	Description
Name	Unikalna nazwa obrazu.
Upload	Wybierz plik (PNG, JPEG, GIF lub WebP) z lokalnego systemu, który ma zostać przesłany do Zabbix. *Uwaga* może być możliwe przesłanie innych formatów, które zostaną przekonwertowane do PNG podczas wysyłania. Do przetwarzania obrazów używana jest biblioteka GD, dlatego obsługiwane formaty zależą od używanej wersji biblioteki (wymagana jest wersja 2.0.28 lub nowsza).

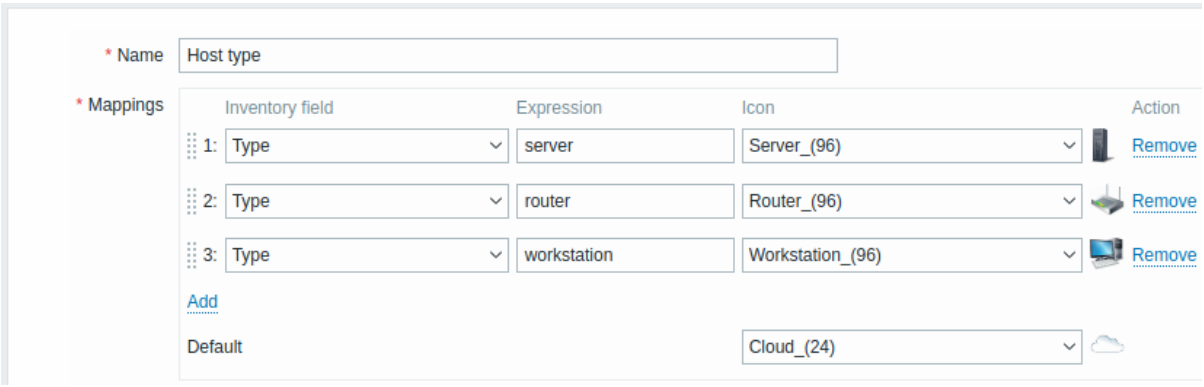
Note:
Maksymalny rozmiar pliku do przesłania jest ograniczony przez wartość ZBX_MAX_IMAGE_SIZE, która wynosi 1024x1024 bajtów lub 1 MB.
Przesyłanie obrazu może się nie powieść, jeśli rozmiar obrazu jest bliski 1 MB, a parametr konfiguracyjny MySQL `max_allowed_packet` ma wartość domyślną 1 MB. W takim przypadku zwiększ parametr max_allowed_packet.

Mapowanie ikon

Ta sekcja umożliwia tworzenie mapowania określonych hostów z określonymi ikonami. Do utworzenia mapowania wykorzystywane są informacje z pól inwentarza hosta.

Mapowania można następnie wykorzystać w konfiguracji mapy sieciowej, aby automatycznie przypisywać odpowiednie ikony do pasujących hostów.

Aby utworzyć nową mapę ikon, kliknij *Create icon map* w prawym górnym rogu.



Parametry konfiguracji:

Parameter	Description
Name	Unikalna nazwa mapy ikon.
Mappings	Lista mapowań. Kolejność mapowań określa, które z nich będzie miało priorytet. Możesz przenosić mapowania w górę i w dół listy metodą przeciągnij i upuść.
Inventory field	Pole inwentarza hosta, które będzie sprawdzane w celu znalezienia dopasowania.
Expression	Wyrażenie regularne opisujące dopasowanie.
Icon	Ikona używana, jeśli zostanie znalezione dopasowanie do wyrażenia.
Default	Domyślna ikona do użycia.

Wyrażenia regularne

Ta sekcja umożliwia tworzenie niestandardowych wyrażeń regularnych, które można używać w kilku miejscach we frontendzie. Szczegóły można znaleźć w sekcji Wyrażenia regularne.

Opcje wyświetlania wyzwalaczy

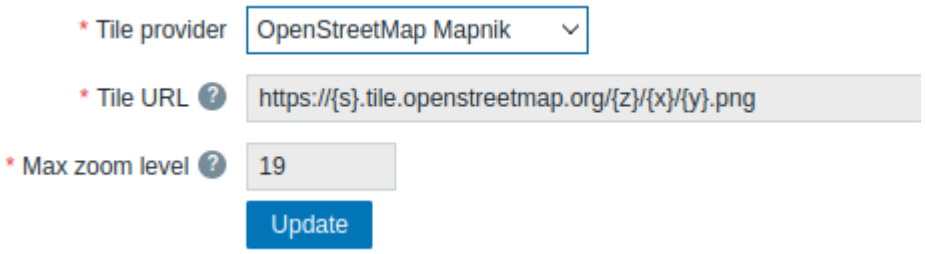
Ta sekcja umożliwia dostosowanie sposobu wyświetlania statusu wyzwalacza w frontend oraz nazw i kolorów poziomu wyzwalacza.



Parameter	Description
Use custom event status colors *Unacknowledged PROBLEM events*, *Acknowledged PROBLEM events*, *Unacknowledged RESOLVED events*, *Acknowledged RESOLVED events*	Zaznaczenie tego parametru włącza dostosowywanie kolorów dla potwierdzonych/niepotwierdzonych problemów. Wprowadź nowy kod koloru lub kliknij kolor, aby wybrać nowy z dostępnej palety. Jeśli zaznaczono pole *blinking*, wyzwalacze będą migać przez pewien czas po zmianie statusu, aby były bardziej widoczne.
Display OK triggers for	Okres wyświetlania wyzwalaczy OK. Dozwolony zakres: 0 - 24 godziny. Obsługiwane są sufiksy czasu, np. 5m, 2h, 1d.
On status change triggers blink for	Czas migania wyzwalacza. Dozwolony zakres: 0 - 24 godziny. Obsługiwane są sufiksy czasu, np. 5m, 2h, 1d.
Not classified, *Information*, *Warning*, *Average*, *High*, *Disaster*	Niestandardowe nazwy poziomów ważności i/lub kolory wyświetlane zamiast domyślnych systemowych. Wprowadź nowy kod koloru lub kliknij kolor, aby wybrać nowy z dostępnej palety. Pamiętaj, że niestandardowe nazwy poziomów ważności wprowadzone tutaj będą używane we wszystkich lokalizacjach. Jeśli musisz przetłumaczyć je na inne języki dla określonych użytkowników, zobacz stronę Dostosowywanie poziomów ważności wyzwalaczy.

Mapy geograficzne

Ta sekcja umożliwia wybór dostawcy usługi kafelków map geograficznych oraz konfigurację ustawień dostawcy usługi dla widżetu dashboardu Geomap. Aby zapewnić wizualizację z użyciem map geograficznych, Zabbix korzysta z otwartoźródłowej biblioteki JavaScript do interaktywnych map Leaflet. Należy pamiętać, że Zabbix nie ma wpływu na jakość obrazów dostarczanych przez zewnętrznych dostawców kafelków, w tym przez predefiniowanych dostawców kafelków.



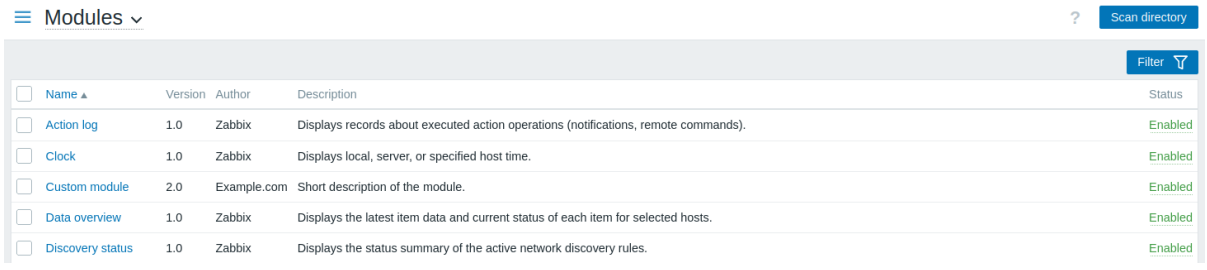
Parameter	Description
Tile provider	Wybierz jednego z dostępnych dostawców usługi kafelków albo wybierz *Other*, aby dodać innego dostawcę kafelków lub kafelki hostowane samodzielnie (zobacz Using a custom tile service provider).
Tile URL	Szablon adresu URL (do 2048 znaków) służący do ładowania i wyświetlania warstwy kafelków na mapach geograficznych. To pole jest edytowalne tylko wtedy, gdy *Tile provider* ma ustawioną wartość *Other*. Obsługiwane są następujące symbole zastępcze: *{s}* reprezentuje jedną z dostępnych subdomen; *{z}* reprezentuje parametr poziomu powiększenia w adresie URL; *{x}* i *{y}* reprezentują współrzędne kafelka; *{r}* można użyć, aby dodać do adresu URL "@2x" i wczytać kafelki retina. Przykład: `https://{s}.example.com/{z}/{x}/{y}{r}.png`
Attribution text	Tekst atrybucji dostawcy kafelków, który ma być wyświetlany w małym polu tekstowym na mapie. To pole jest widoczne tylko wtedy, gdy *Tile provider* ma ustawioną wartość *Other*.
Max zoom level	Maksymalny poziom powiększenia mapy. To pole jest edytowalne tylko wtedy, gdy *Tile provider* ma ustawioną wartość *Other*.

Korzystanie z niestandardowego dostawcy usługi kafelków

Widżet Geomap może wczytywać rastrowe obrazy kafelków z niestandardowej, hostowanej samodzielnie lub zewnętrznej usługi dostawcy kafelków. Aby użyć niestandardowej zewnętrznej usługi dostawcy kafelków albo samodzielnie hostowanego folderu lub serwera kafelków, wybierz *Other* w polu *Tile provider* i określ niestandardowy adres URL w polu *Tile URL*, używając odpowiednich symboli zastępczych.

Moduły

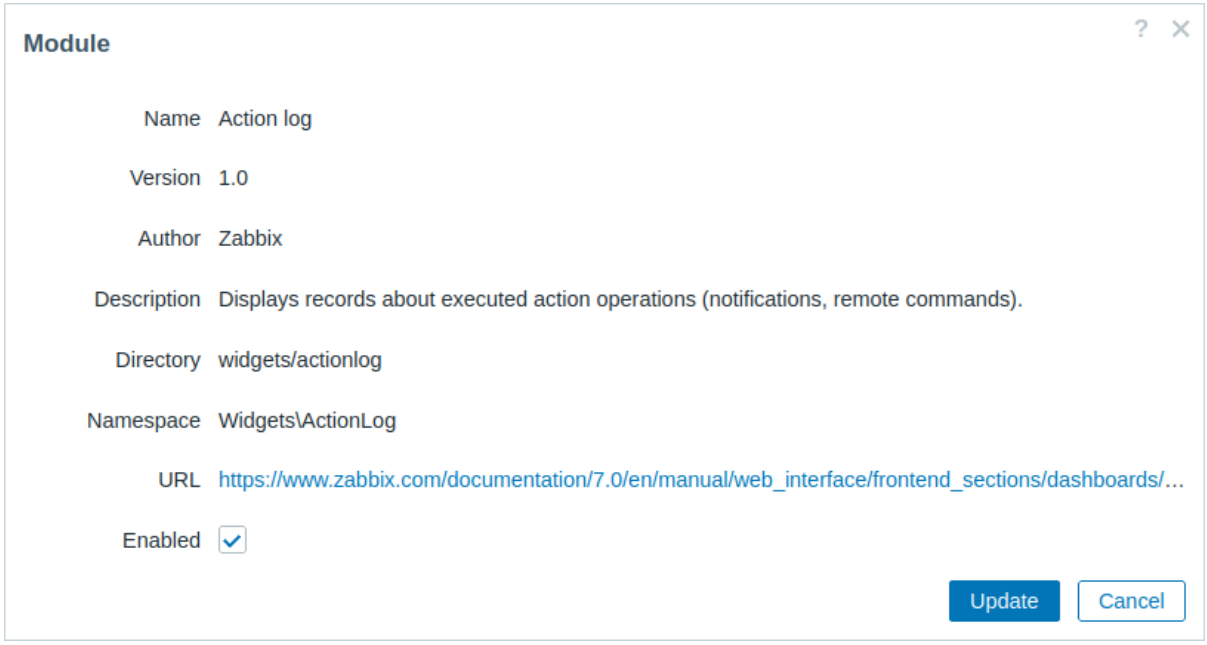
Ta sekcja umożliwia zarządzanie niestandardowymi, a także wbudowanymi modułami frontend.



Kliknij *Skanuj katalog*, aby zarejestrować/wyrejestrować dowolne moduły niestandardowe. Zarejestrowane moduły pojawią się na liście; niezarejestrowane moduły zostaną z niej usunięte.

Kliknij status modułu na liście, aby włączyć/wyłączyć moduł. Możesz także zbiorczo włączać/wyłączać moduły, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* poniżej listy.

Kliknij nazwę modułu na liście, aby wyświetlić jego szczegóły w wyskakującym oknie.

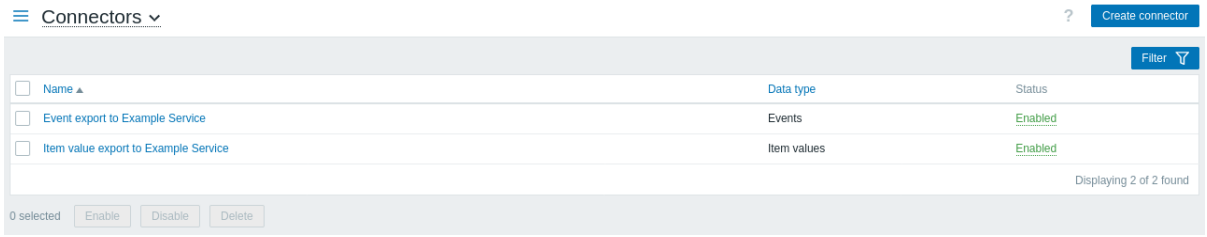


Status modułu można również zaktualizować w wyskakującym oknie szczegółów modułu; w tym celu zaznacz/odznacz pole wyboru *Włączony*, a następnie kliknij *Aktualizuj*.

Możesz filtrować moduły według nazwy lub statusu (włączony/wyłączony).

Łączniki

Ta sekcja umożliwia konfigurację łączników dla strumieniowania danych Zabbix do systemów zewnętrznych streaming to external systems przez HTTP.



Kliknij *Create connector*, aby skonfigurować nowy connector.

Możesz filtrować łączniki według nazwy lub statusu (włączone/wyłączone). Kliknij status łącznika na liście, aby włączyć/wyłączyć łącznik. Możesz także masowo włączać/wyłączać łączniki, zaznaczając je na liście, a następnie klikając przyciski *Enable/Disable* poniżej listy.

Inne

Ta sekcja umożliwia konfigurację różnych pozostałych parametrów frontend.



Parametr	Opis
URL frontend	Adres URL (do 2048 znaków) do interfejsu webowego Zabbix. Ten parametr jest używany przez usługę webową Zabbix do komunikacji z frontend i powinien zostać określony, aby włączyć raporty cykliczne.
Grupa dla wykrytych hostów	Hosty wykryte przez wykrywanie sieci i autorejestrację agent zostaną automatycznie umieszczone w wybranej tutaj grupie hostów.
Domyślny tryb inwentaryzacji hosta	Domyślny tryb inwentaryzacji hosta. Będzie stosowany zawsze, gdy nowy host lub prototyp hosta zostanie utworzony przez serwer lub frontend, chyba że zostanie nadpisany podczas wykrywania/autorejestracji hosta przez operację *Ustaw tryb inwentaryzacji hosta*.

Parametr	Opis
Grupa użytkowników dla komunikatu o niedostępności bazy danych	Grupa użytkowników do wysyłania komunikatu alarmowego o niedostępności bazy danych. Serwer Zabbix zależy od dostępności bazy danych backendu. Nie może działać bez bazy danych. Jeśli baza danych jest niedostępna, wybrani użytkownicy mogą zostać powiadomieni przez Zabbix. Powiadomienia będą wysyłane do ustawionej tutaj grupy użytkowników przy użyciu włączonych wpisów mediów użytkownika. Powiadomienia są przesyłane przy użyciu następujących typów mediów (jeśli są włączone): e-mail; SMS; niestandardowe skrypty alertów. Nawet jeśli wpis mediów typu webhook jest skonfigurowany i włączony, nie będzie wyzwalał powiadomień. Serwer Zabbix nie zatrzyma się; będzie czekał, aż baza danych ponownie stanie się dostępna, aby kontynuować przetwarzanie. Powiadomienie składa się z następującej treści: `[MySQL|PostgreSQL] database <DB Name> [on <DB Host>:<DB Port>] is not available: <error message depending on the type of DBMS (database)>` <DB Host> nie jest dodawany do komunikatu, jeśli jest zdefiniowany jako pusta wartość, a <DB Port> nie jest dodawany, jeśli ma wartość domyślną („0”). Menedżer alertów (specjalny proces serwera Zabbix) próbuje ustanowić nowe połączenie z bazą danych co 10 sekund. Jeśli baza danych nadal jest niedostępna, menedżer alertów ponawia wysyłanie alertów, ale nie częściej niż co 15 minut.
Rejestruj niedopasowane pułapki SNMP	Rejestruj pułapkę SNMP, jeśli nie znaleziono odpowiadających interfejsów SNMP.

Autoryzacja

Parametr	Opis
Próby logowania	Liczba nieudanych prób logowania, po której możliwość zalogowania się zostanie zablokowana.
Interwał blokady logowania	Okres, przez który logowanie będzie zabronione po przekroczeniu limitu *Próby logowania*. Dozwolony zakres: 30 - 3600 sekund. Obsługiwane są sufiksy czasu, np. 90s, 5m, 1h.

Przechowywanie sekretów

Parametr	Opis
Dostawca sejfu	Wybierz oprogramowanie do zarządzania sekretami do przechowywania wartości makr użytkownika - *HashiCorp Vault* (domyślnie) lub *CyberArk Vault*.
Rozwiązuj makra sejfu sekretów przez	Rozwiązuj makra sejfu sekretów przez: **serwer Zabbix** - sekrety są pobierane przez serwer Zabbix i przekazywane do proxy, gdy jest to potrzebne (domyślnie); **serwer Zabbix i proxy** - sekrety są pobierane zarówno przez serwer Zabbix, jak i proxy, co pozwala im rozwiązywać makra niezależnie.

Zobacz także: Przechowywanie sekretów.

Bezpieczeństwo

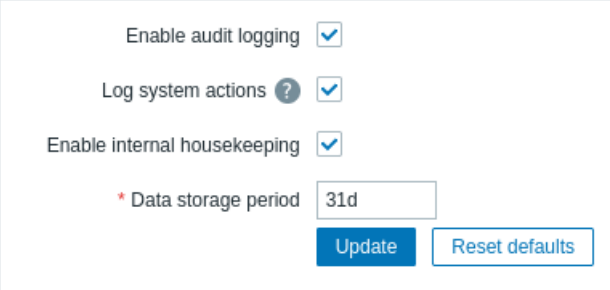
Parametr	Opis
Validate URI schemes	Odznacz to pole wyboru, aby wyłączyć walidację schematów URI (domyślnie włączona). Jeśli jest zaznaczone, możesz określić rozdzieloną przecinkami listę dozwolonych schematów URI (domyślnie: http,https,ftp,file,mailto,tel,ssh). Dotyczy wszystkich pól we frontend, w których używane są URI (na przykład adresów URL elementów mapy).

Parametr	Opis
Use X-Frame-Options HTTP header	Odznacz to pole wyboru, aby wyłączyć nagłówek HTTP X-Frame-options (niezalecane). Jeśli jest zaznaczone, możesz określić wartość nagłówka HTTP X-Frame-options. Obsługiwane wartości: **SAMEORIGIN** (domyślnie) lub **'self'** (musi być ujęte w pojedyncze cudzysłowy) - strona może być wyświetlana w ramce tylko w tym samym źródle co sama strona; **DENY** lub **'none'** (musi być ujęte w pojedyncze cudzysłowy) - strona nie może być wyświetlana w ramce, niezależnie od witryny próbującej to zrobić; **ciąg nazw hostów oddzielonych spacjami**; dodanie **'self'** (musi być ujęte w pojedyncze cudzysłowy) do listy pozwala na wyświetlanie strony w ramce w tym samym źródle co sama strona. Zwróć uwagę, że użycie **'self'** lub **'none'** bez pojedynczych cudzysłowów spowoduje potraktowanie ich jako nazw hostów.
Use iframe sandboxing	Odznacz to pole wyboru, aby wyłączyć umieszczanie zawartości pobranego adresu URL w sandboxie (niezalecane). Jeśli jest zaznaczone, możesz określić wyjątki sandboxingu iframe; nieokreślone ograniczenia nadal będą stosowane. Jeśli to pole jest puste, zastosowanie mają wszystkie ograniczenia atrybutu sandbox. Aby uzyskać więcej informacji, zobacz opis atrybutu `sandbox`.

2 Dziennik audytu

Przegląd

Ta sekcja umożliwia konfigurację ustawień dziennika audytu.



Dostępne są następujące parametry:

Parametr	Opis
Włącz rejestrowanie audytu	Włącz (domyślnie) lub wyłącz rejestrowanie audytu.
Rejestruj działania systemowe	Włącz (domyślnie) lub wyłącz rejestrowanie audytu działań związanych z wykrywaniem niskiego poziomu, wykrywaniem sieci oraz autorejestracją wykonywanych przez serwer (użytkownik systemowy).
Włącz wewnętrzny housekeeping	Włącz (domyślnie) lub wyłącz wewnętrzny housekeeping dla rekordów dziennika audytu.
Okres przechowywania danych	Liczba dni, przez które rekordy dziennika audytu powinny być przechowywane, zanim zostaną usunięte przez housekeeper. Wymagane, jeśli housekeeping jest włączony. Domyślnie: 31 dni.

Parametry kompresji dziennika audytu są dostępne w sekcji *Administracja* > *Housekeeping*, w bloku *Kompresja historii, trendów i dziennika audytu*, który staje się widoczny przy użyciu TimescaleDB.

3 Porządkowanie danych

Przegląd

Sekcja *Administracja* > *Housekeeping* umożliwia skonfigurowanie wewnętrznego mechanizmu housekeeping w Zabbix, który automatycznie usuwa z bazy danych nieaktualne dane lub dane usunięte przez użytkownika. Zapobiega to nieograniczonemu wzrostowi bazy danych, który prowadziłby do niepotrzebnego zużycia miejsca na dysku i pogorszenia wydajności bazy danych.

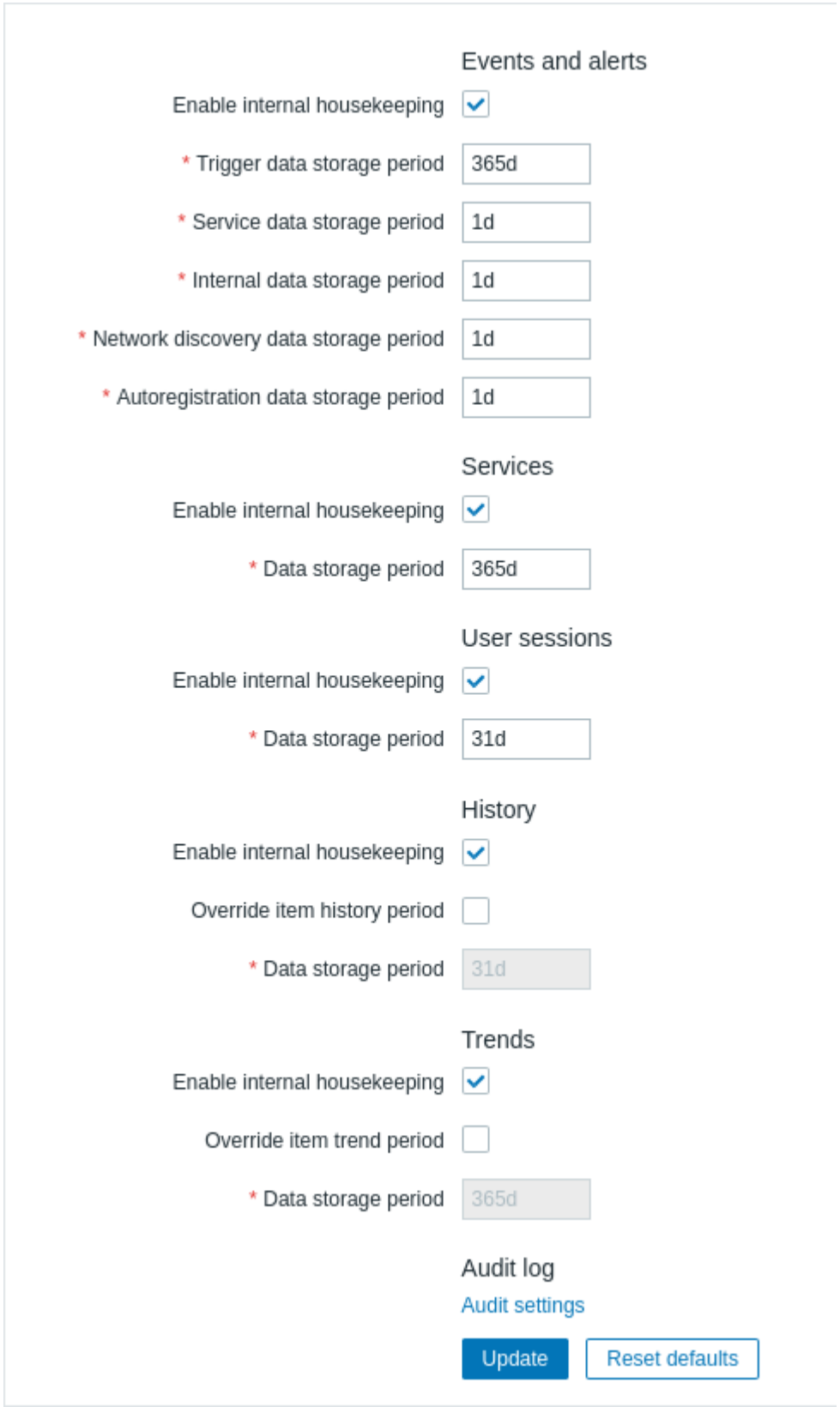
Housekeeping można włączyć i skonfigurować dla następujących typów danych:

- Zdarzenia i alerty z wyzwalaczy, usług, danych wewnętrznych, wykrywania sieci i autorejestracji
- Usługi
- Sesje użytkowników
- Historia i trendy dla pozycji

Note:
Housekeeping dziennika audytu jest konfigurowany oddzielnie.

Konfiguracja

Dostępne są następujące parametry:



Parameter	Description
Enable internal housekeeping	Włącz lub wyłącz wewnętrzny housekeeper (domyślnie włączony). Jeśli jest włączony, proces housekeeper serwera Zabbix automatycznie usuwa z bazy danych dane, które przekraczają *Data storage period*.

Parameter	Description
Data storage period	Określ, jak długo dane mają być przechowywane przed usunięciem przez housekeepera. Wymagane, jeśli włączono wewnętrzny housekeeper. Zakres: 1 dzień (1 godzina dla historii) - 25 lat; lub ”0”. Obsługiwane są Time suffixes (np. 1d, 1w). Dla *Events and alerts* okres przechowywania danych jest ustawiany oddzielnie dla wyzwalaczy, usług, danych wewnętrznych, wykrywania sieci i automatycznej rejestracji. Housekeeper usuwa tylko te zdarzenia, które nie są powiązane z problemami. Na przykład zdarzenie problem/odzyskanie starsze niż *Data storage period* nie zostanie usunięte, jeśli jest powiązane z nierozwiązanym problemem. Gdy housekeeper usuwa przestarzałe encje, najpierw usuwa problemy, a następnie zdarzenia. Zwróć uwagę, że gdy pozycja/wyzwalacz zostanie usunięta, jej problemy również są usuwane; jednak powiązane zdarzenia problem/odzyskanie pozostają w bazie danych do czasu, aż housekeeper je usunie.
Override item history period	Dla *History* i *Trends* okresy przechowywania danych określają również, jak długo dane pozostają widoczne w *Monitoring* > *Latest data*, nawet jeśli wewnętrzny housekeeper jest wyłączony. Jeśli ta opcja jest włączona, okres przechowywania historii/trendów określony w item configuration zostanie zastąpiony ustawieniem *Data storage period* (z wyjątkiem pozycji z włączoną opcją *Do not store*). Z tej opcji można korzystać nawet wtedy, gdy wewnętrzny housekeeper jest wyłączony, a używany jest zewnętrzny housekeeper.
Override item trend period	

Jeśli używasz **TimescaleDB**, dostępna staje się sekcja *History, trends and audit log compression*.

Note:
Dla TimescaleDB włącz *Override item history period*, *Override item trend period* oraz *Enable internal housekeeping* dla historii i trendów, aby w pełni korzystać z automatycznego partycjonowania. Jeśli te opcje są wyłączone, dane przechowywane w tabelach historii i trendów nadal będą partycjonowane, ale housekeeper nie będzie usuwał przestarzałych partycji, a także będą wyświetlane ostrzeżenia konfiguracyjne. Gdy usuwanie przestarzałych partycji jest włączone, serwer Zabbix i frontend nie będą już śledzić usuniętych pozycji, a historia dla tych pozycji zostanie wyczyszczona po usunięciu przestarzałej partycji.

Przycisk *Reset defaults* umożliwia cofnięcie wszystkich wprowadzonych zmian.

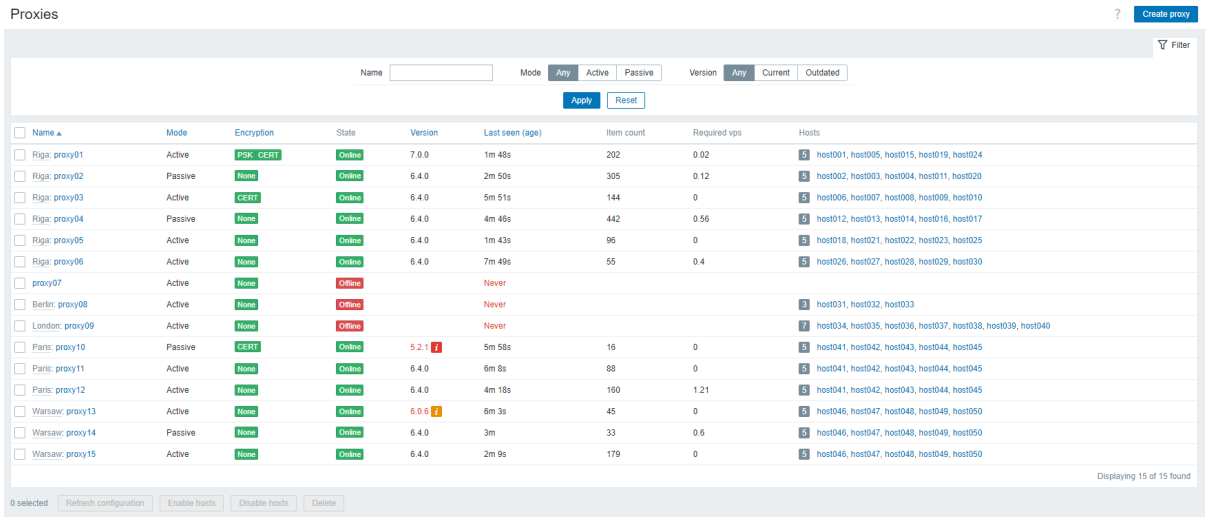
4 Proxy

Przegląd

W sekcji *Administracja* → *Proxy* można skonfigurować proxy do monitorowania rozproszonego w frontendzie Zabbix.

Proxy

Wyświetlana jest lista istniejących proxy wraz z ich szczegółami.



Wyświetlane dane:

Column	Description
Name	Nazwa proxy. Kliknięcie nazwy proxy otwiera formularz konfiguracji proxy. Jeśli proxy należy do grupy proxy, nazwa grupy jest wyświetlana przed nazwą proxy jako szary odnośnik. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy proxy.
Mode	Tryb proxy — *Aktywny* lub *Pasywny*.
Encryption	Stan szyfrowania połączeń z proxy: **Brak** — brak szyfrowania; **PSK** — użycie klucza współdzielonego; **Cert** — użycie certyfikatu.
State	Stan proxy: **Nieznany** — proxy zostało utworzone, gdy serwer Zabbix był wyłączony, lub serwer nie zaktualizował jeszcze stanu; **Online** — proxy komunikowało się z serwerem Zabbix (pasywny proxy odpowiedział na żądanie serwera; aktywny proxy wysłał żądanie) w okresie failover; **Offline** — proxy nie komunikowało się z serwerem Zabbix w okresie failover.
Version	Wersja proxy (trzycyfrowy numer wersji). Jeśli proxy jest przestarzałe lub nieobsługiwane, numer wersji jest wyróżniony (na czerwono), a wyświetlana jest ikona statusu informacyjnego (żółta lub czerwona). Najedź kursorem na ikonę, aby zobaczyć szczegóły.
Last seen (age)	Czas, kiedy proxy było ostatnio widziane przez serwer.
Item count	Liczba włączonych pozycji na włączonych hostach przypisanych do proxy.
Required vps	Wymagana wydajność proxy (liczba wartości, które muszą być zbierane na sekundę).
Hosts	Liczba włączonych hostów przypisanych do proxy oraz lista hostów monitorowanych przez proxy. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta.

Aby skonfigurować nowe proxy, kliknij przycisk *Create proxy* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

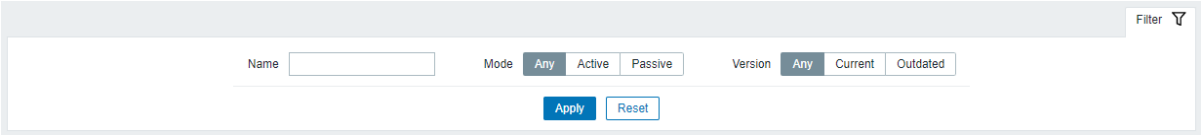
- *Odśwież konfigurację* - odśwież konfigurację proxy;
- *Włącz hosty* - zmień status hostów monitorowanych przez proxy na *Monitorowany*;
- *Wyłącz hosty* - zmień status hostów monitorowanych przez proxy na *Niemonitorowany*;
- *Usuń* - usuń proxy.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich proxy, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te proxy, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą proxy. Po jego kliknięciu pojawi się filtr, w którym można filtrować proxy według nazwy, trybu i wersji. Zwróć uwagę, że opcja filtra *Nieaktualne* wyświetla zarówno nieaktualne (częściowo obsługiwane), jak i nieobsługiwane proxy.



5 Grupy proxy

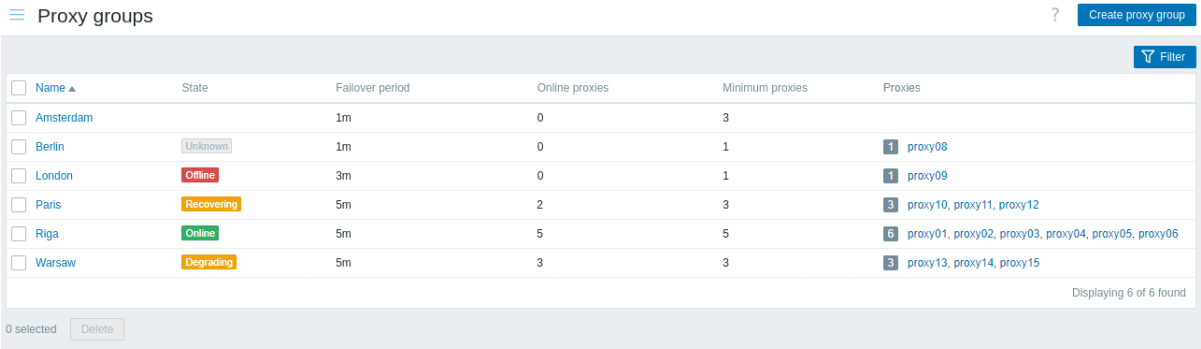
Przegląd

W *Administracja* → *Grupy proxy* można skonfigurować grupy proxy.

Grupy proxy są używane w równoważeniu obciążenia proxy z automatyczną dystrybucją hostów między proxy oraz wysoką dostępnością między proxy.

Grupy proxy

Wyświetlana jest lista istniejących grup proxy wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy proxy. Kliknięcie nazwy grupy proxy otwiera formularz konfiguracji grupy proxy.
Stan	Stan grupy proxy (wyświetlany, jeśli grupa zawiera co najmniej jeden proxy): **Nieznany** — grupa została utworzona, gdy serwer Zabbix był wyłączony, lub serwer nie zaktualizował jeszcze stanu; **Online** — minimalna liczba proxy komunikowała się z serwerem Zabbix w okresie przełączenia awaryjnego; **Pogarszający się** — grupa wkrótce przejdzie w stan offline, ponieważ liczba proxy online spadła poniżej progu *Minimalna liczba proxy*; **Offline** — mniej niż minimalna liczba proxy komunikowała się z serwerem Zabbix w okresie przełączenia awaryjnego; **Odzyskiwanie** — grupa wkrótce wróci do stanu online, ponieważ liczba proxy online osiągnęła próg *Minimalna liczba proxy*. Stan grupy proxy można monitorować za pomocą pozycji zabbix[proxy group,<name>,state] oraz zabbix[proxy group,discovery].
Okres przełączenia awaryjnego	Okres, w którym proxy w grupie proxy musi komunikować się z serwerem Zabbix, aby był uznawany za online.
Proxy online	Liczba proxy online (wyświetlana na czerwono, jeśli jest niższa od minimum grupy).
Minimalna liczba proxy	Minimalna liczba proxy online wymagana do utrzymania grupy proxy w stanie online.
Proxy	Liczba proxy w grupie oraz lista proxy w grupie z odnośnikami do formularza konfiguracji proxy. Maksymalna liczba wyświetlanych proxy jest ograniczona przez wartość *Max count of elements to show inside table cell*.

Aby skonfigurować nową grupę proxy, kliknij przycisk *Create proxy groups* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

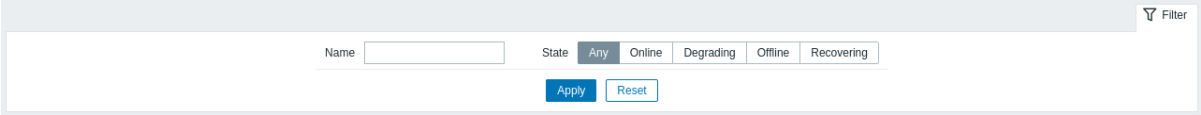
- *Usuń* - usuń grupy proxy.

Aby skorzystać z tych opcji, zaznacz pola wyboru przy odpowiednich grupach proxy, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te grupy proxy, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

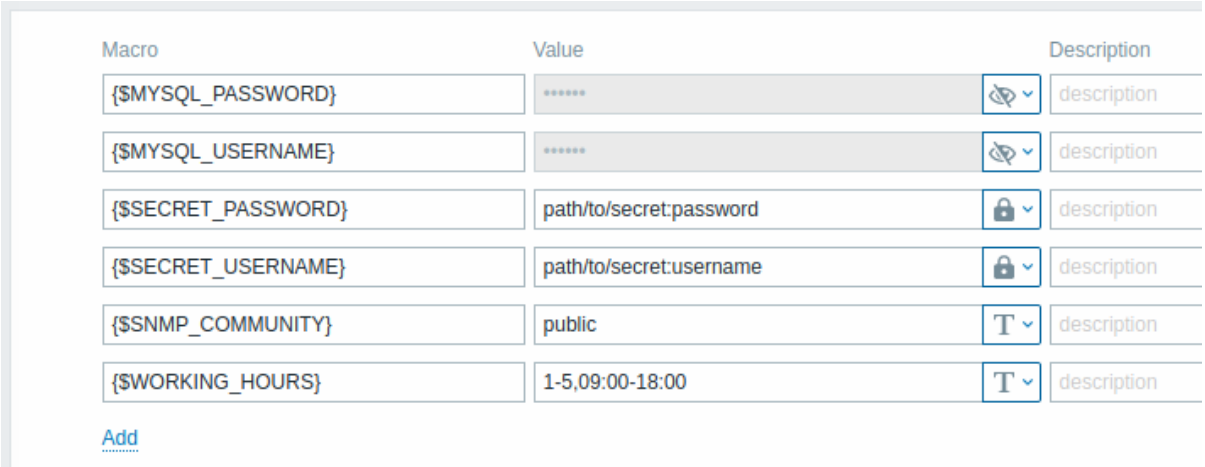
Link *Filtr* jest dostępny nad listą grup proxy. Po jego kliknięciu pojawi się filtr, za pomocą którego można filtrować grupy proxy według nazwy i statusu.



6 Makra

Przegląd

Ta sekcja umożliwia definiowanie globalnych makr użytkownika jako par nazwa-wartość. Należy pamiętać, że wartości makr mogą być przechowywane jako zwykły tekst, tajny tekst lub sekret Vault. Obsługiwane jest również dodawanie opisu.



7 Kolejka

Przegląd

W sekcji *Administracja* → *Kolejka* wyświetlane są pozycje oczekujące na aktualizację.

W idealnej sytuacji po otwarciu tej sekcji wszystko powinno być „zielone”, co oznacza brak pozycji w kolejce. Jeśli wszystkie pozycje są aktualizowane bez opóźnień, żadna z nich nie oczekuje. Jednak z powodu niewystarczającej wydajności serwera niektóre pozycje mogą być opóźnione, a informacje o tym są wyświetlane w tej sekcji. Więcej informacji można znaleźć w sekcji Kolejka.

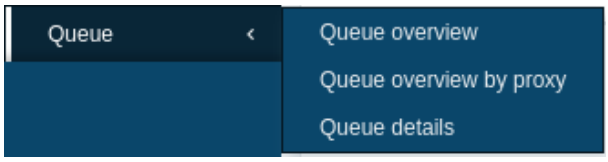
Note:
Zabbix proxy nie uwzględnia okresów konserwacji; szczegóły można znaleźć w sekcji Obliczanie kolejek podczas konserwacji.

Note:
Kolejka jest dostępna tylko wtedy, gdy działa serwer Zabbix. Pozycje nie są uwzględniane w kolejce, jeśli interfejs pozycji staje się niedostępny z powodu problemów z połączeniem lub gdy agent nie działa prawidłowo.

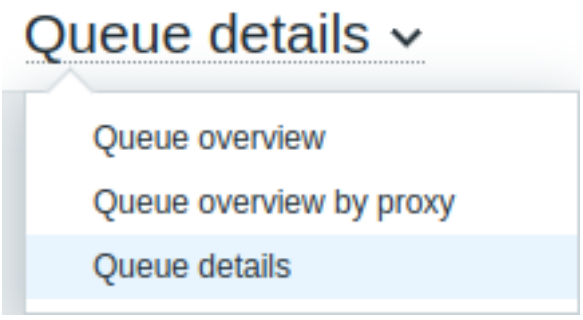
Sekcja *Administracja* → *Kolejka* zawiera następujące strony:

- Przegląd kolejki — wyświetla kolejkę według typu pozycji;
- Przegląd kolejki według proxy — wyświetla kolejkę według proxy;
- Szczegóły kolejki — wyświetla listę opóźnionych pozycji.

Lista dostępnych stron pojawia się po kliknięciu *Kolejka* w sekcji menu *Administracja*. Możliwe jest również przełączanie się między stronami za pomocą listy rozwijanej tytułu w lewym górnym rogu.



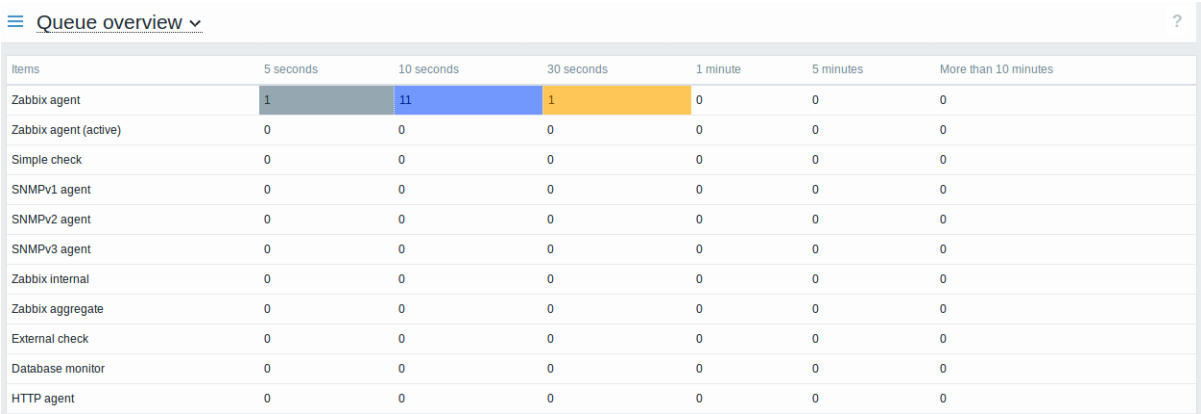
Menu trzeciego poziomu.



Lista rozwijana tytułu.

Przegląd według typu pozycji

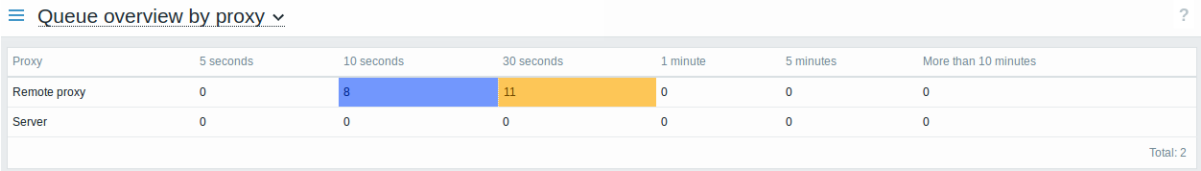
Na tym ekranie łatwo ustalić, czy problem jest związany z jednym czy z kilkoma typami pozycji.



Każdy wiersz zawiera typ pozycji. Każda kolumna pokazuje liczbę oczekujących pozycji — oczekujących odpowiednio przez 5–10 sekund / 10–30 sekund / 30–60 sekund / 1–5 minut / 5–10 minut lub ponad 10 minut.

Przegląd według proxy

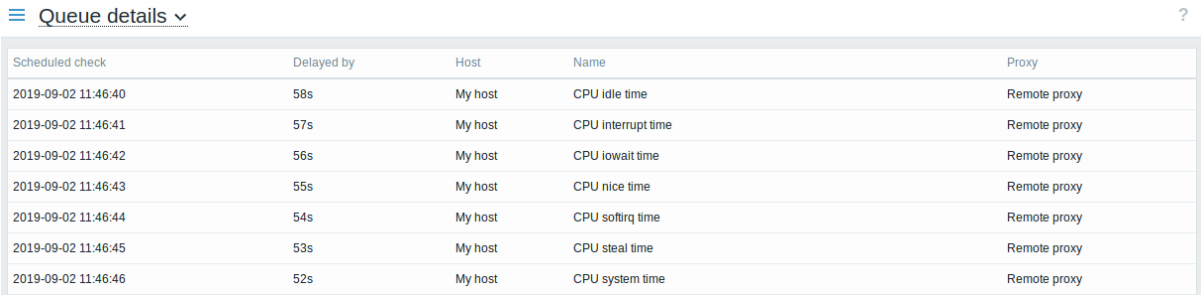
Na tym ekranie łatwo ustalić, czy problem jest związany z jednym z proxy czy z serwerem.



Każdy wiersz zawiera proxy, a na końcu listy znajduje się serwer. Każda kolumna pokazuje liczbę oczekujących pozycji — oczekujących odpowiednio przez 5–10 sekund / 10–30 sekund / 30–60 sekund / 1–5 minut / 5–10 minut lub ponad 10 minut.

Lista oczekujących pozycji

Na tym ekranie wyświetlana jest każda oczekująca pozycja.



Wyświetlane dane:

Kolumna	Opis
Zaplanowane sprawdzenie	Wyświetlany jest czas, w którym sprawdzenie powinno zostać wykonane.
Opóźnienie	Wyświetlana jest długość opóźnienia.
Host	Wyświetlany jest host pozycji.
Nazwa	Wyświetlana jest nazwa oczekującej pozycji.
Proxy	Wyświetlana jest nazwa proxy, jeśli host jest monitorowany przez proxy.

Możliwe komunikaty o błędach

Możesz napotkać sytuację, w której nie są wyświetlane żadne dane i pojawia się następujący komunikat o błędzie:



Komunikat o błędzie w tym przypadku jest następujący:

```
Cannot display item queue. Permission denied
```

Dzieje się tak, gdy parametry konfiguracji PHP w pliku *zabbix.conf.php* — `$ZBX_SERVER` lub oba: `$ZBX_SERVER` i `$ZBX_SERVER_PORT` — wskazują na istniejący serwer Zabbix, który używa innej bazy danych.

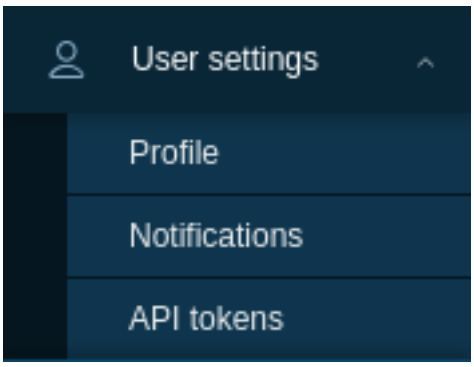
3 Ustawienia użytkownika

Przegląd

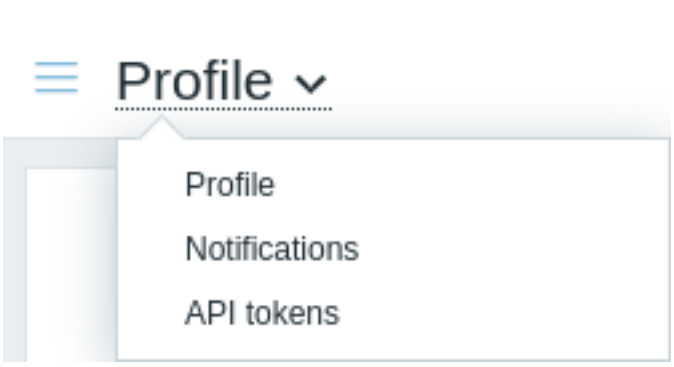
W zależności od uprawnień roli użytkownika sekcja *Ustawienia użytkownika* może zawierać następujące strony:

- *Profil* lub *Profil użytkownika* — do dostosowywania wybranych funkcji frontendu Zabbix;
- *Powiadomienia* — do dostosowywania powiadomień bieżącego użytkownika;
- *Tokeny API* — do zarządzania tokenami API przypisanymi do bieżącego użytkownika.

Lista dostępnych stron pojawia się po kliknięciu ikony użytkownika w dolnej części menu Zabbix (niedostępne dla użytkownika *guest*). Możliwe jest również przełączanie się między stronami za pomocą rozwijanej listy tytułu w lewym górnym rogu.



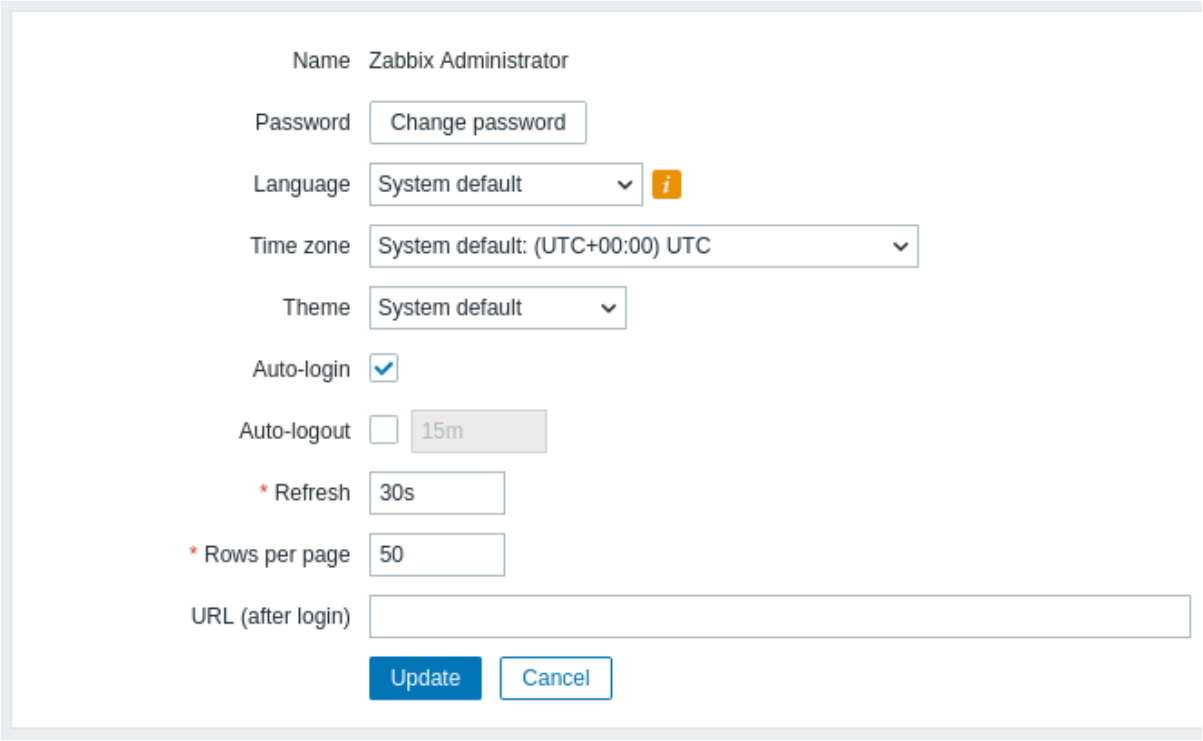
Menu trzeciego poziomu.



Rozwijana lista tytułu.

Profil użytkownika

Sekcja *Profil użytkownika* udostępnia opcje ustawienia niestandardowego języka interfejsu, motywu kolorystycznego, liczby wierszy wyświetlanych na listach itp. Zmiany wprowadzone tutaj będą miały zastosowanie tylko do bieżącego użytkownika.



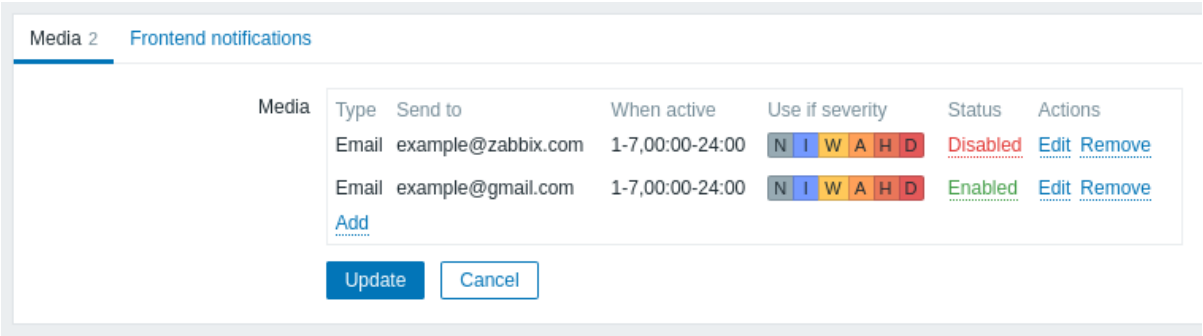
Parameter	Description
Name	Wyświetlane są imię i nazwisko użytkownika. W przypadku braku imienia i nazwiska wyświetlana jest nazwa użytkownika.
Password	Kliknij przycisk *Change password*, aby otworzyć trzy pola: *Old password*, *New password*, *New password (once again)*. Po pomyślnej zmianie hasła użytkownik zostanie wylogowany ze wszystkich aktywnych sesji. Pamiętaj, że hasło można zmienić tylko dla użytkowników korzystających z wewnętrznego uwierzytelniania Zabbix internal authentication.
Language	Wybierz preferowany język interfejsu lub wybierz **System default**, aby użyć domyślnych ustawień systemowych. Wybranie *English (en_US)* włączy również amerykański format daty i czasu w frontend. Więcej informacji można znaleźć w sekcji Installation of additional frontend languages.
Time zone	Wybierz strefę czasową, aby zastąpić globalną time zone na poziomie użytkownika, lub wybierz **System default**, aby użyć globalnych ustawień strefy czasowej.
Theme	Wybierz motyw kolorystyczny dla swojego profilu: **System default** - użyj domyślnych ustawień systemowych; **Blue** - standardowy niebieski motyw; **Dark** - alternatywny ciemny motyw; **High-contrast light** - jasny motyw o wysokim kontraście; **High-contrast dark** - ciemny motyw o wysokim kontraście.
Auto-login	Zaznacz to pole wyboru, aby Zabbix zapamiętał Cię i logował Cię automatycznie przez 30 dni. Podczas logowania z włączoną opcją *Remember for 30 days*: - licznik automatycznego wylogowania jest resetowany (sesja trwa do końca 30-dniowego okresu). - automatyczne logowanie jest włączane podczas kolejnych wizyt w ciągu 30 dni. Podczas logowania bez opcji *Remember for 30 days*: - parametr automatycznego logowania jest czyszczony. - automatyczne wylogowanie działa zgodnie z skonfigurowanym limitem czasu. Wykorzystywane są do tego pliki cookie przeglądarki.

Parameter	Description
Auto-logout	Po zaznaczeniu tego pola wyboru nastąpi automatyczne wylogowanie po ustawionej liczbie sekund (minimum 90 sekund, maksimum 1 dzień). Pamiętaj, że jeśli logowanie odbywa się z włączoną opcją *Remember for 30 days*, licznik automatycznego wylogowania zostanie zastąpiony, a sesja pozostanie aktywna przez cały ten okres. Obsługiwane są sufiksy czasu, na przykład: 90s, 5m, 2h, 1d. Pamiętaj, że ta opcja nie będzie działać w następujących przypadkach: * Gdy strony menu *Monitoring* odświeżają informacje w tle. W przypadku pozostawienia otwartych stron, które odświeżają dane w określonych odstępach czasu (dashboardy, wykresy, najnowsze dane itp.), czas życia sesji zostaje odpowiednio wydłużony, co wyłącza funkcję automatycznego wylogowania. * Jeśli logowanie odbywa się z zaznaczoną opcją *Remember me for 30 days*. * Gdy uwierzytelnianie jest wykonywane za pośrednictwem dostawcy tożsamości SSO (IdP), automatyczne wylogowanie kończy tylko sesję Zabbix. Sesja SSO z IdP pozostaje aktywna, więc użytkownik może zalogować się ponownie bez podawania hasła. *Auto-logout* może również przyjmować wartość ”0”, co oznacza, że funkcja automatycznego wylogowania zostanie wyłączona po aktualizacji ustawień profilu.
Refresh	Ustaw, jak często informacje na stronach menu *Monitoring* będą odświeżane (minimum 0 sekund, maksimum 1 godzina). Obsługiwane są sufiksy czasu, na przykład: 30s, 90s, 1m, 1h.
Rows per page	Ustaw, ile wierszy będzie wyświetlanych na stronę na listach. Mniejsza liczba wierszy (i mniejsza liczba rekordów do wyświetlenia) skutkuje krótszym czasem ładowania.
URL (after login)	Ustaw konkretny adres URL, który ma być wyświetlany po zalogowaniu. Zamiast domyślnego *Dashboards* może to być na przykład adres URL *Monitoring* > *Triggers*.

Powiadomienia

Sekcja *Powiadomienia* zawiera opcje dostosowywania bieżących powiadomień użytkownika.

Karta *Media* umożliwia określenie szczegółów mediów dla użytkownika, takich jak typy mediów i adresy, których należy używać, oraz momenty ich użycia do dostarczania powiadomień.



Jeśli typ medium został wyłączony:

- po nazwie wyświetlana jest żółta ikona informacji;
- w kolumnie *Status* wyświetlany jest tekst ”Disabled”.

Note:
Uprawnienia użytkownika do zmiany szczegółów mediów dla siebie mogą być przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit own media*). Uprawnienia użytkownika superadmina do zmiany szczegółów mediów dla innych mogą być również przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit user media*)

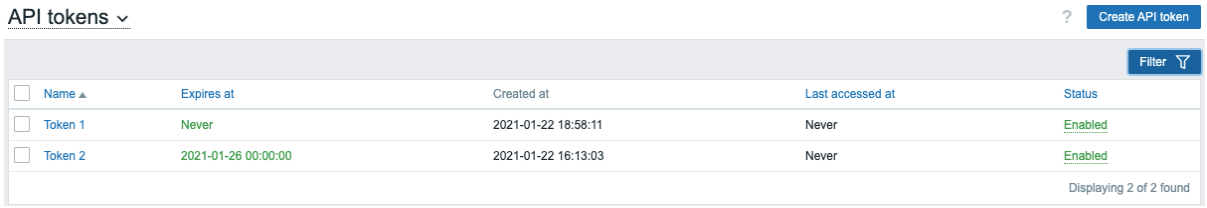
Należy pamiętać, że w przypadku użytkowników aprowizowanych:

- media użytkownika aprowizowanego nie mogą zostać usunięte;
- media użytkownika aprowizowanego mogą zostać wyłączone/włączone;
- takie pola mediów użytkownika aprowizowanego jak *When active*, *Use if severity* i *Enabled* mogą być ręcznie edytowane;
- dla użytkowników aprowizowanych można ręcznie dodać dodatkowe media użytkownika (na przykład dodatkowe adresy e-mail);
- ręcznie dodane media użytkownika mogą zostać usunięte.

Karta *Frontend notifications* umożliwia ustawienie globalnych powiadomień.

Tokeny API

Sekcja *Tokeny API* umożliwia przeglądanie tokenów przypisanych do użytkownika, edycję szczegółów tokenów oraz tworzenie nowych tokenów. Ta sekcja jest dostępna dla użytkownika tylko wtedy, gdy działanie *Manage API tokens* jest dozwolone w ustawieniach roli użytkownika.



Tokeny API można filtrować według nazwy, daty wygaśnięcia lub statusu (*Włączony/Wyłączony*). Kliknij status tokenu na liście, aby szybko włączyć/wyłączyć token. Możesz także włączyć/wyłączyć wiele tokenów jednocześnie, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* pod listą.

Attention:
Użytkownicy nie mogą wyświetlić wartości *Auth token* tokenów przypisanych do nich w Zabbix. Wartość *Auth token* jest wyświetlana tylko raz — bezpośrednio po utworzeniu tokenu. Jeśli token został utracony, należy go wygenerować ponownie.

1 Globalne powiadomienia

Przegląd

Powiadomienia globalne umożliwiają wyświetlanie problemów w czasie rzeczywistym bezpośrednio na bieżącym ekranie w frontend Zabbix.

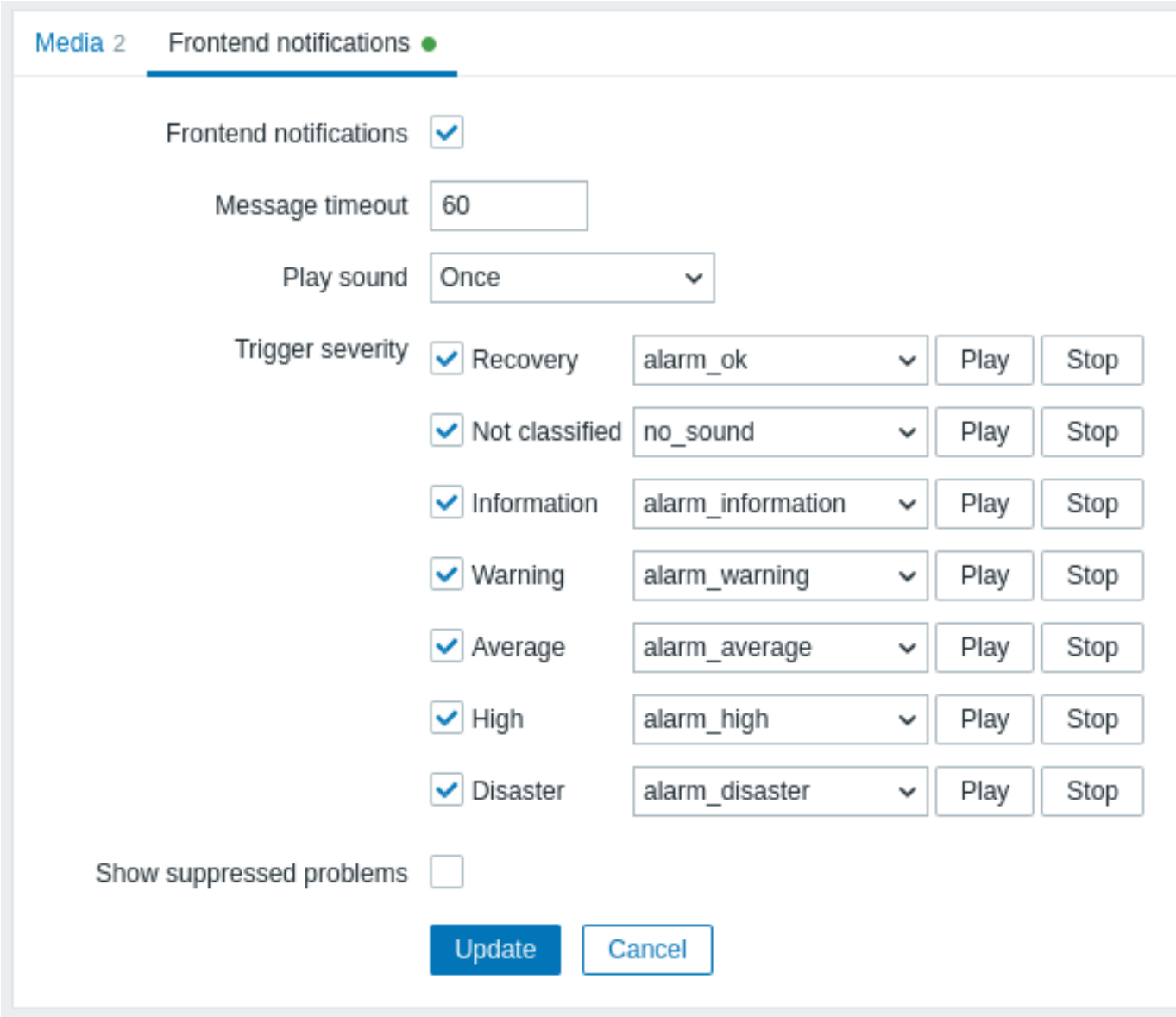
Bez powiadomień globalnych, podczas pracy poza sekcjami *Problemy* lub *Pulpit*, nie otrzymywałbyś żadnych informacji o bieżących problemach. Powiadomienia globalne zapewniają wyświetlanie tych informacji niezależnie od bieżącej lokalizacji w frontend Zabbix.

Powiadomienia globalne obejmują zarówno wyświetlanie komunikatu, jak i odtwarzanie dźwięku.

Attention:
Automatyczne odtwarzanie dźwięków może być wyłączone (domyślnie) w najnowszych wersjach przeglądarek. W takich przypadkach należy włączyć to ustawienie ręcznie.

Konfiguracja

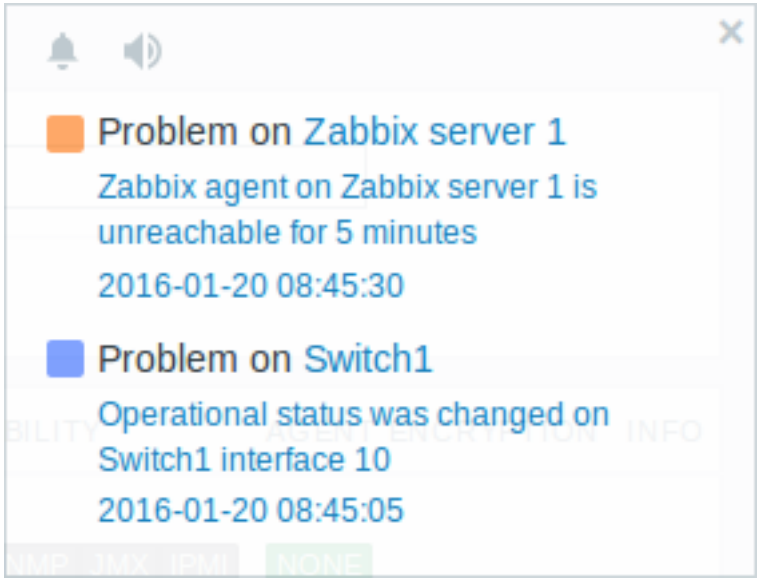
Globalne powiadomienia można włączyć dla każdego użytkownika na karcie **Powiadomienia frontend** w sekcji Powiadomienia.



Parametr	Opis
Powiadomienia frontend	Zaznacz pole wyboru, aby włączyć globalne powiadomienia.
Limit czasu wiadomości	Ustaw czas, przez jaki wiadomość będzie wyświetlana. Domyślnie wiadomości pozostają na ekranie przez 60 sekund. Obsługiwane są sufiksy czasu, na przykład: 30s, 5m, 2h, 1d.
Odtwarzaj dźwięk	Ustaw czas, przez jaki dźwięk będzie odtwarzany. **Raz** - dźwięk jest odtwarzany raz w całości; **10 sekund** - dźwięk jest powtarzany przez 10 sekund; **Limit czasu wiadomości** - dźwięk jest powtarzany, dopóki wiadomość jest widoczna.
Poziom ważności wyzwalacza	Ustaw poziomy ważności wyzwalaczy, dla których będą aktywowane globalne powiadomienia i dźwięki. Możesz także wybrać dźwięki odpowiednie dla różnych poziomów ważności. Jeśli nie zostanie zaznaczony żaden poziom ważności, żadne wiadomości nie będą wyświetlane. Dodatkowo wiadomości o odzyskaniu będą wyświetlane tylko dla zaznaczonych poziomów ważności. Na przykład, jeśli zaznaczone są *Odzyskanie* i *Katastrofa*, globalne powiadomienia będą wyświetlane dla problemów i odzyskań wyzwalaczy o poziomie ważności *Katastrofa*.
Pokaż stłumione problemy	Zaznacz pole wyboru, aby wyświetlać powiadomienia o problemach, które w przeciwnym razie byłyby stłumione (niepokazywane) z powodu konserwacji hosta.

Wyświetlane komunikaty globalne

Gdy komunikaty napływają, są wyświetlane w pływającej sekcji po prawej stronie. Możesz dowolnie zmieniać położenie tej sekcji, przeciągając jej nagłówek.



Dla tej sekcji dostępnych jest kilka elementów sterujących:

- Przycisk **Snooze** wycisza aktualnie aktywny dźwięk alarmu;
- Przycisk **Mute/Unmute** przełącza między odtwarzaniem a całkowitym nieodtwarzaniem dźwięków alarmu.

2 Dźwięk w przeglądarkach

Przegląd

Dźwięk jest używany w powiadomieniach globalnych.

Aby dźwięki były odtwarzane w frontend Zabbix, w profilu użytkownika na karcie *Frontend notifications* musi być włączona opcja *Frontend notifications*, a wszystkie poziomy ważności wyzwalaczy muszą być zaznaczone. Dodatkowo dźwięki powinny być włączone w wyskakującym oknie powiadomień globalnych.

Jeśli z jakiegokolwiek powodu dźwięk nie może być odtwarzany na urządzeniu, przycisk w wyskakującym oknie powiadomień globalnych pozostanie na stałe w stanie „wyciszenia", a po najechaniu kursorem na przycisk zostanie wyświetlony komunikat „Cannot support notification audio for this device".

Dźwięki, w tym domyślne klipy audio, są obsługiwane wyłącznie w formacie MP3.

Dźwięki frontend Zabbix zostały pomyślnie przetestowane w najnowszych przeglądarkach Firefox i Opera w systemie Linux oraz w przeglądarkach Chrome, Firefox, Microsoft Edge i Opera w systemie Windows.

Attention:
Automatyczne odtwarzanie dźwięków może być wyłączone (domyślnie) w najnowszych wersjach przeglądarek. W takich przypadkach należy włączyć to ustawienie ręcznie.

4 Wyszukiwanie globalne

Możliwe jest przeszukiwanie frontend Zabbix pod kątem hostów, grup hostów, szablonów i grup szablonów.

Pole wyszukiwania znajduje się pod logo Zabbix w menu. Wyszukiwanie można rozpocząć, naciskając *Enter* lub klikając ikonę wyszukiwania .



Jeśli istnieje host, który zawiera wprowadzony ciąg znaków w dowolnej części nazwy, pojawi się lista rozwijana zawierająca wszystkie takie hosty (z pasującą częścią wyróżnioną na pomarańczowo). Lista rozwijana wyświetli również host, jeśli widoczna nazwa tego hosta odpowiada nazwie technicznej wprowadzonej jako ciąg wyszukiwania; pasujący host zostanie wyświetlony, ale bez żadnego wyróżnienia.

Atrybuty podlegające wyszukiwaniu

Hosty można wyszukiwać według następujących właściwości:

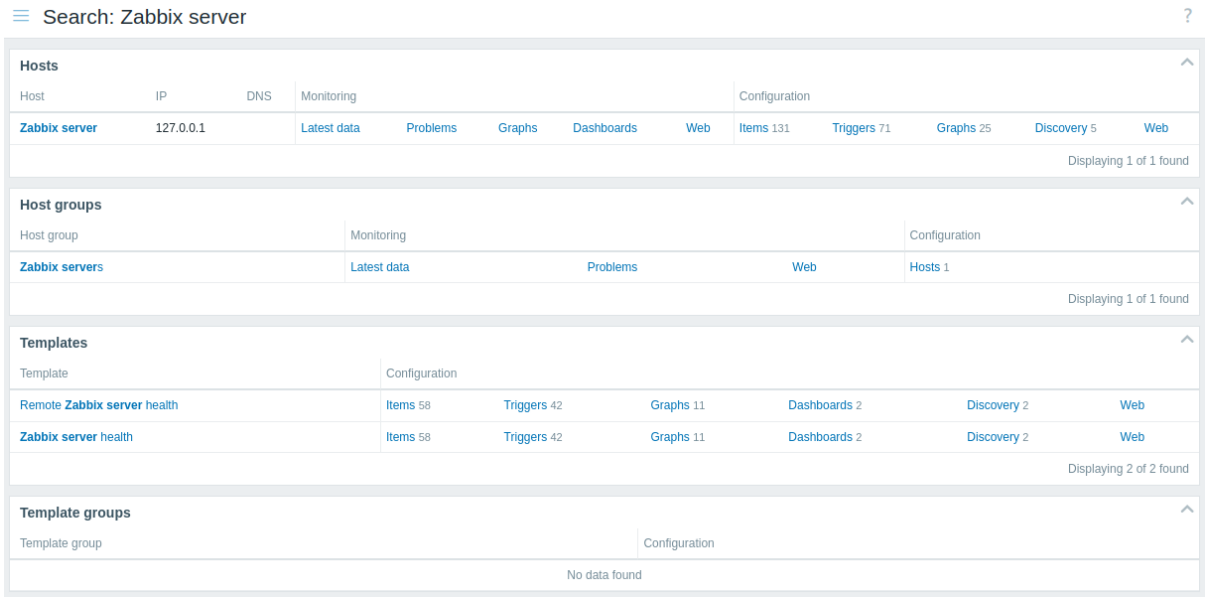
- Nazwa hosta
- Widoczna nazwa
- Adres IP
- Nazwa DNS

Szablony można wyszukiwać według nazwy lub widocznej nazwy. Jeśli wyszukujesz według nazwy, która różni się od widocznej nazwy (szablonu/hosta), w wynikach wyszukiwania jest ona wyświetlana poniżej widocznej nazwy w nawiasach.

Grupy hostów i szablonów można wyszukiwać według nazwy. Określenie grupy nadrzędnej niejawnie wybiera wszystkie grupy zagnieżdżone.

Wyniki wyszukiwania

Wyniki wyszukiwania składają się z czterech oddzielnych bloków dla hostów, grup hostów, szablonów i grup szablonów.



Możliwe jest zwijanie/rozwijanie każdego pojedynczego bloku. Liczba wpisów jest wyświetlana na dole każdego bloku, na przykład: *Wyświetlanie 13 z 13 znalezionych*. Jeśli nie ma żadnych wpisów, liczba wpisów nie jest wyświetlana. Łączna liczba wpisów wyświetlanych w jednym bloku jest ograniczona do 100.

Każdy wpis zawiera odnośniki do danych monitoringu i konfiguracji. Zobacz pełną listę odnośników.

Dla wszystkich danych konfiguracyjnych (takich jak pozycje, wyzwalacze, wykresy) liczba znalezionych encji jest wyświetlana jako liczba obok nazwy encji, w kolorze szarym. **Uwaga:** jeśli encji jest zero, żadna liczba nie jest wyświetlana.

Włączone hosty są wyświetlane na niebiesko, wyłączone hosty na czerwono.

Dostępne linki

Dla każdego wpisu dostępne są następujące linki:

- Hosty
  - Monitorowanie
    - Najnowsze dane
    - Problemy
    - Wykresy
    - Pulpity hosta
    - Scenariusze WWW
  - Konfiguracja
    - Pozycje
    - Wyzwalacze
    - Wykresy
    - Reguły wykrywania
    - Scenariusze WWW
- Grupy hostów
  - Monitorowanie
    - Najnowsze dane
    - Problemy
    - Scenariusze WWW
  - Konfiguracja
    - Hosty
- Szablony
  - Konfiguracja
    - Pozycje
    - Wyzwalacze
    - Wykresy
    - Pulpity szablonu
    - Reguły wykrywania
    - Scenariusze WWW
- Grupy szablonów
  - Konfiguracja
    - Szablony

5 Tryb konserwacji frontendu

Przegląd

Można tymczasowo wyłączyć frontend Zabbix, aby ograniczyć dostęp. Służy do ochrony bazy danych Zabbix przed zmianami wprowadzonymi przez użytkownika, zachowując jej integralność.

Gdy frontend Zabbix jest w trybie konserwacji, możesz bezpiecznie zatrzymać bazę danych i wykonać zadania konserwacyjne.

Użytkownicy ze zdefiniowanych adresów IP będą mogli normalnie wchodzić w interakcję z frontendem w trybie konserwacji.

Konfiguracja

Aby włączyć tryb konserwacji, otwórz plik `maintenance.inc.php` (znajdujący się w `/conf` katalogu dokumentów HTML Zabbix na serwerze WWW) i usuń komentarz z następujących wierszy:

// Tryb konserwacji. define('ZBX_DENY_GUI_ACCESS', 1);

// Tablica adresów IP, które mogą łączyć się z frontendem (opcjonalnie). $ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');

// Komunikat wyświetlany na ekranie ostrzegawczym (opcjonalnie). $ZBX_GUI_ACCESS_MESSAGE = 'Aktualizujemy bazę danych MySQL do godziny 15:00. Bądź na bieżąco...';

Note:
W większości przypadków plik `maintenance.inc.php` znajduje się w katalogu `/conf` katalogu dokumentów HTML Zabbixa na serwerze WWW. Jednak niektóre systemy operacyjne i serwery WWW mogą używać innej lokalizacji.
Na przykład lokalizacja dla:

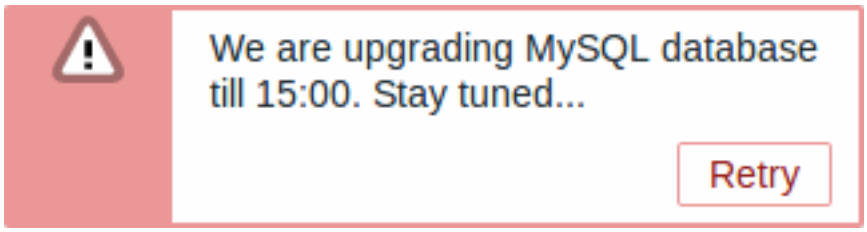
- SUSE i RedHat to `/etc/zabbix/web/maintenance.inc.php`.
- systemów opartych na Debianie to `/usr/share/zabbix/conf/`.

Zobacz także Kopiowanie plików PHP.

Parameter	Details
ZBX_DENY_GUI_ACCESS	Jeśli zdefiniowano z dowolną wartością, tryb konserwacji zostanie włączony. Aby wyłączyć tryb konserwacji, należy zakomentować tę linię lub ją usunąć.
ZBX_GUI_ACCESS_IP_RANGE	Tablica adresów IP, które mają □□□□□□?

Wyświetl

Użytkownicy zobaczą następujący ekran podczas próby dostępu do frontendu Zabbix w trybie konserwacji. Ekran jest odświeżany co 30 sekund, aby powrócić do normalnego stanu bez interwencji użytkownika po zakończeniu konserwacji.



Adresy IP zdefiniowane w *ZBX_GUI_ACCESS_IP_RANGE* będą mogły uzyskać dostęp do frontendu jak zwykle.

6 Parametry strony

Przegląd

Większość stron interfejsu webowego Zabbix obsługuje różne parametry HTTP GET, które określają, co zostanie wyświetlone. Można je przekazywać, podając po adresie URL pary parametr=wartość, oddzielone od adresu URL znakiem zapytania (?) i od siebie nawzajem znakiem ampersand (&).

Monitoring > Problemy

Obsługiwane są następujące parametry:

Parameter	Description	Example
show	Opcja filtra *Pokaż*. Możliwe wartości: 1 - ostatnie problemy; 2 - wszystkie; 3 - w stanie problemu.	show=1
name	Opcja filtra *Problem*: dowolny ciąg znaków.	name=Zabbix agent
severities	Opcja filtra *Severity*: tablica wybranych poziomów ważności w formacie `severities[*]=*` (zastąp * poziomem ważności). Możliwe wartości: 0 - nie sklasyfikowano; 1 - informacja; 2 - ostrzeżenie; 3 - przeciętny; 4 - wysoki; 5 - katastrofa.	severities[3]=3
inventory	Opcja filtra *Host inventory*: tablica pól inwentarza `[field]`, `[value]`	inventory[0][field]=type&invento

Parameter	Description	Example
evaltype	Opcja filtra *Tags*: metoda oceny tagów. Możliwe wartości: 0 - i/lub; 2 - lub.	evaltype=0
tags	Opcja filtra *Tags*: tablica zdefiniowanych tagów `[tag]`, `[operator]`, `[value]` Możliwe wartości operator: 0 - zawiera; 1 - równa się; 2 - nie zawiera; 3 - nie równa się; 4 - istnieje; 5 - nie istnieje.	tags[0][tag]=target&tags[0][ope
show_tags	Opcja filtra *Pokaż tagi*. Możliwe wartości: 0 - brak; 1 - jeden; 2 - dwa; 3 - trzy.	show_tags=3
tag_name_format	Opcja filtra *Nazwa tagu*. Możliwe wartości: 0 - pełna nazwa; 1 - skrócona nazwa; 2 - brak.	tag_name_format=1
tag_priority	Opcja filtra *Priorytet wyświetlania tagów*: ciąg znaków rozdzielony przecinkami określający priorytet wyświetlania tagów	tag_priority=customer, target
show_suppressed	Opcja filtra *Pokaż wyciszone problemy*. Możliwe wartości: 1 - pokaż; 2 - nie pokazuj.	show_suppressed=1
acknowledgement_status	Opcja filtra *Status potwierdzenia*. Możliwe wartości: 0 - wszystkie; 1 - niepotwierdzone; 2 - potwierdzone.	acknowledgement_status=0
acknowledged_by_me	Opcja filtra *Przeze mnie*; obsługiwana tylko z `acknowledgement_status=2`. Możliwe wartości: 0 - wyłączone; 1 - włączone.	acknowledged_by_me=1
compact_view	Opcja filtra *Widok kompaktowy*. Możliwe wartości: 0 - wyłączone; 1 - włączone.	compact_view=1
highlight_row	Opcja filtra *Podświetl cały wiersz*. Możliwe wartości: 0 - wyłączone; 1 - włączone. W przypadku wersji Zabbix wcześniejszych niż 7.4.3 ta opcja jest obsługiwana tylko wtedy, gdy `compact_view=1`.	highlight_row=1
filter_name	Opcja właściwości filtra *Nazwa*: dowolny ciąg znaków	filter_name=Databases

Parameter	Description	Example
filter_show_counter	Opcja właściwości filtra *Pokaż liczbę rekordów*. Możliwe wartości: 0 - wyłączone; 1 - włączone.	filter_show_counter=1
filter_custom_time	Opcja właściwości filtra *Ustaw niestandardowy okres czasu*. Możliwe wartości: 0 - wyłączone; 1 - włączone.	filter_custom_time=1
sort	Kolumna, według której należy sortować. Możliwe wartości: clock - sortowanie według kolumny *Time*; host - sortowanie według kolumny *Host*; severity - sortowanie według kolumny *Severity*; name - sortowanie według kolumny *Problem*.	sort=clock
sortorder	Kolejność sortowania wyników. Możliwe wartości: DESC - malejąco; ASC - rosnąco.	sortorder=DESC
age_state	Opcja filtra *Wiek mniejszy niż*; obsługiwana tylko z `show`=3. Możliwe wartości: 0 - wyłącz parametr `age`; 1 - włącz parametr `age`.	age_state=1
age	Opcja filtra *Wiek mniejszy niż*: liczba całkowita, liczba dni; obsługiwana tylko z `age_state`=1 i `show`=3.	age=7
groupids	Opcja filtra *Grupy hostów*: tablica identyfikatorów grup hostów.	groupids[]=4
hostids	Opcja filtra *Hosty*: tablica identyfikatorów hostów	hostids[]=10084
triggerids	Opcja filtra *Wyzwalacze*: tablica identyfikatorów wyzwalaczy	triggerids[]=22382
show_timeline	Opcja filtra *Pokaż oś czasu*; nieobsługiwana z `compact_view`=1. Możliwe wartości: 0 - nie pokazuj; 1 - pokaż.	show_timeline=1
details	Opcja filtra *Pokaż szczegóły*. Możliwe wartości: 0 - nie pokazuj; 1 - pokaż.	details=1
from	Początek zakresu dat, może być względny (np. now-1m); obsługiwane tylko z `filter_custom_time`=1	from=now-2h
to	Koniec zakresu dat, może być względny (np. now-1m); obsługiwane tylko z `filter_custom_time`=1	to=now

Zobacz też: opcje filtra na stronie Problemy.

Tryb kiosku

Tryb kiosku na obsługiwanych stronach frontend można włączyć za pomocą parametrów URL. Na przykład na pulpitach:

- `/zabbix.php?action=dashboard.view&kiosk=1` - włącz tryb kiosku
- `/zabbix.php?action=dashboard.view&kiosk=0` - włącz tryb normalny

Pokaz slajdów

Możliwe jest włączenie pokazu slajdów na pulpicie:

- `/zabbix.php?action=dashboard.view&slideshow=1` - włącz pokaz slajdów

7 Definicje

Przegląd

Chociaż wiele rzeczy we frontend można skonfigurować przy użyciu samego frontend, niektóre dostosowania są obecnie możliwe tylko przez edycję pliku definicji.

Ten plik to `defines.inc.php`, znajdujący się w katalogu /include katalogu dokumentów HTML Zabbix.

Parametry

Parametry w tym pliku, które mogą być interesujące dla użytkowników:

- ZBX_MIN_PERIOD

Minimalny okres wykresu, w sekundach. Domyślnie jedna minuta.

- GRAPH_YAXIS_SIDE_DEFAULT

Domyślne położenie osi Y na prostych wykresach oraz domyślna wartość na liście rozwijanej podczas dodawania pozycji do wykresów niestandardowych. Możliwe wartości: 0 - lewa, 1 - prawa.

Domyślnie: 0

- ZBX_SESSION_NAME

Ciąg znaków używany jako nazwa ciasteczka sesji frontend Zabbix.

Domyślnie: zbx_sessionid

- ZBX_DATA_CACHE_TTL

Limit czasu TTL w sekundach używany do unieważniania pamięci podręcznej danych odpowiedzi Vault. Ustaw 0, aby wyłączyć buforowanie odpowiedzi Vault.

Domyślnie: 60

- SUBFILTER_VALUES_PER_GROUP

Liczba wartości podfiltra na grupę (na przykład w podfiltrze najnowsze dane).

Domyślnie: 1000

- ZBX_MAX_WIDGET_LINES

Maksymalna liczba wierszy widżetu do wyświetlenia.

Domyślnie: 1000

8 Tworzenie własnego motywu

Przegląd

Domyślnie Zabbix udostępnia kilka predefiniowanych motywów. Możesz postępować zgodnie z przedstawioną tutaj procedurą krok po kroku, aby utworzyć własny motyw. Zachęcamy do podzielenia się efektem swojej pracy ze społecznością Zabbix, jeśli uda Ci się stworzyć coś ciekawego.

Krok 1

Aby zdefiniować własny motyw, należy utworzyć plik CSS i zapisać go w folderze `assets/styles/` (na przykład *custom-theme.css*). Można skopiować pliki z innego motywu i utworzyć swój motyw na ich podstawie albo zacząć od zera.

Krok 2

Dodaj swój motyw do listy motywów zwracanych przez metodę APP::getThemes(). Możesz to zrobić, nadpisując metodę ZBase::getThemes() w klasie APP. Można to zrobić, dodając następujący kod przed zamykającym nawiasem klamrowym w pliku *include/classes/core/APP.php*:

```
public static function getThemes() {
    return array_merge(parent::getThemes(), [
        'custom-theme' => _('Custom theme')
    ]);
```

```
}
```

Attention:
Zwróć uwagę, że nazwa podana w pierwszej parze cudzysłowów musi odpowiadać nazwie pliku motywu bez rozszerzenia.

Aby dodać wiele motywów, po prostu wypisz je pod pierwszym motywem, na przykład:

```
public static function getThemes() {
    return array_merge(parent::getThemes(), [
        'custom-theme' => _('Custom theme'),
        'anothertheme' => _('Another theme'),
        'onemoretheme' => _('One more theme')
    ]);
}
```

Zwróć uwagę, że każdy motyw oprócz ostatniego musi mieć na końcu przecinek.

Note:
Aby zmienić kolory wykresu, wpis musi zostać dodany do tabeli bazy danych *graph_theme*.

Krok 3

Aktywuj nowy motyw.

W frontend Zabbix możesz ustawić ten motyw jako domyślny lub zmienić swój motyw w profilu użytkownika.

Ciesz się nowym wyglądem i stylem!

9 Tryb debugowania

Przegląd

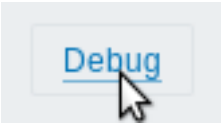
Tryb debugowania może być używany do diagnozowania problemów z wydajnością stron frontend.

Konfiguracja

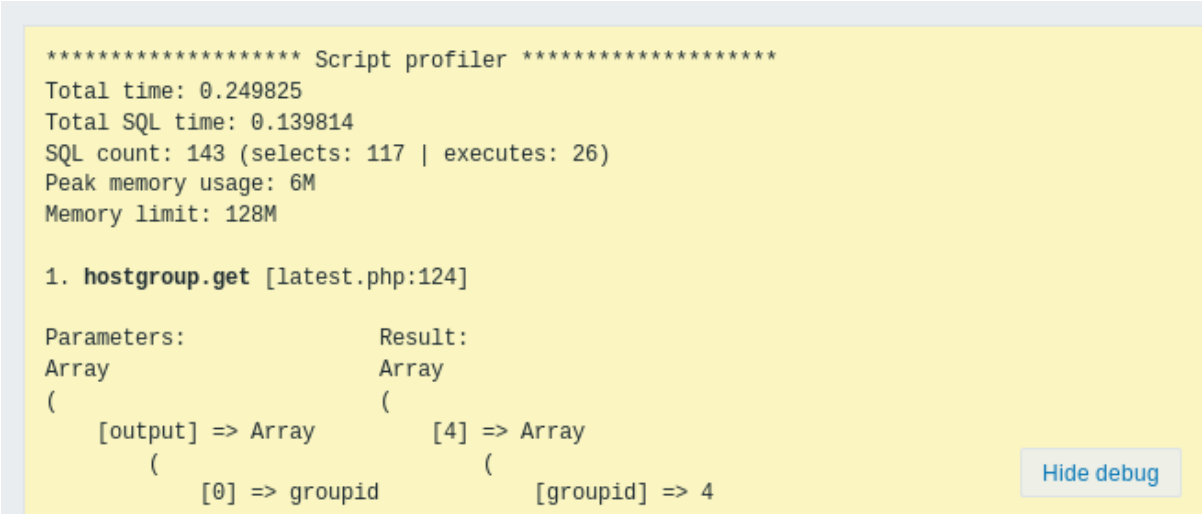
Tryb debugowania można włączyć dla poszczególnych użytkowników należących do grupy użytkowników:

- podczas konfigurowania grupy użytkowników;
- podczas przeglądania skonfigurowanych grup użytkowników.

Gdy opcja *Tryb debugowania* jest włączona dla grupy użytkowników, jej użytkownicy zobaczą przycisk *Debug* w prawym dolnym rogu okna przeglądarki:



Kliknięcie przycisku *Debug* otwiera nowe okno poniżej zawartości strony, które zawiera statystyki SQL dla strony, a także listę wywołań API i poszczególnych instrukcji SQL:



W przypadku problemów z wydajnością strony to okno może zostać użyte do wyszukania głównej przyczyny problemu.

Warning:
Włączony *Tryb debugowania* ma negatywny wpływ na wydajność frontend.

10 Ciasteczka używane przez Zabbix

Przegląd

Ta strona zawiera listę plików cookie używanych przez Zabbix.

Nazwa	Opis	Wartości	Wygasa/Max-Age	HttpOnly[1]	Secure[2]
ZBX_SESSION_NAME	Dane sesji frontend Zabbix, przechowywane jako JSON zakodowany przy użyciu base64		Sesja (wygasa po zakończeniu sesji przeglądania)	+	+ (tylko jeśli HTTPS jest włączony na serwerze WWW)
tab	Numer aktywnej karty; ten plik cookie jest używany tylko na stronach z wieloma kartami (np. na stronie konfiguracji *Host*, *Trigger* lub *Action*) i jest tworzony, gdy użytkownik przechodzi z karty głównej do innej karty (takiej jak karta *Tags* lub *Dependencies*). 0 jest używane dla karty głównej.	Przykład: 1	Sesja (wygasa po zakończeniu sesji przeglądania)	-	-
browserwarning_ignore	Określa, czy ostrzeżenie o używaniu nieaktualnej przeglądarki powinno zostać zignorowane.	yes	Sesja (wygasa po zakończeniu sesji przeglądania)	-	-
system-message-ok	Komunikat do wyświetlenia natychmiast po ponownym załadowaniu strony.	Komunikat w postaci zwykłego tekstu	Sesja (wygasa po zakończeniu sesji przeglądania) lub natychmiast po ponownym załadowaniu strony	+	-

Nazwa	Opis	Wartości	Wygasa/Max-Age	HttpOnly[1]	Secure[2]
system-message-error	Komunikat o błędzie do wyświetlenia natychmiast po ponownym załadowaniu strony.	Komunikat w postaci zwykłego tekstu	Sesja (wygasa po zakończeniu sesji przeglądania) lub natychmiast po ponownym załadowaniu strony	+	-

Note:
Wymuszanie flagi 'HttpOnly' dla plików cookie Zabbix za pomocą dyrektywy serwera WWW nie jest obsługiwane.

Przypisy

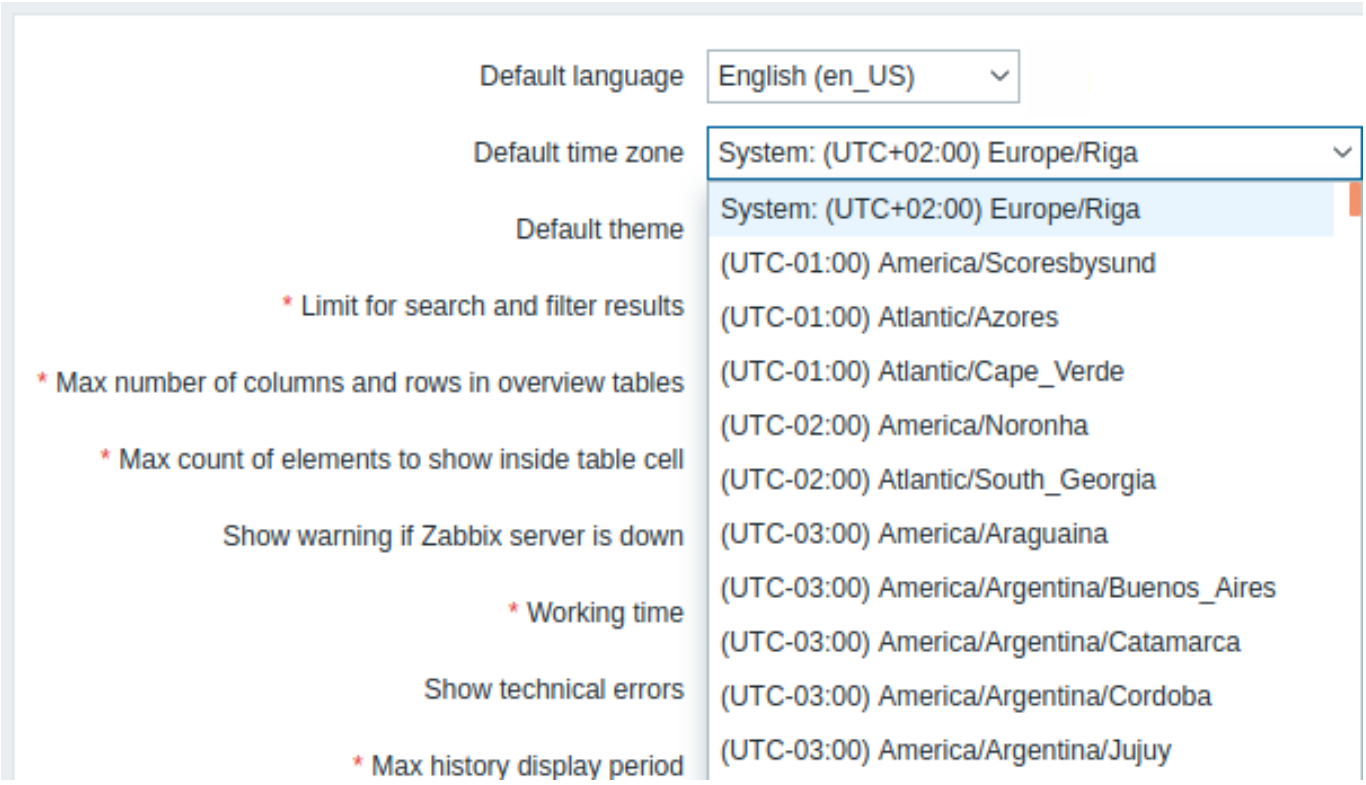
[1] Gdy `HttpOnly` ma wartość 'true', plik cookie będzie dostępny tylko przez protokół HTTP. Oznacza to, że plik cookie nie będzie dostępny dla języków skryptowych, takich jak JavaScript. To ustawienie może skutecznie pomóc ograniczyć kradzież tożsamości poprzez ataki XSS (chociaż nie jest obsługiwane przez wszystkie przeglądarki).

[2] `Secure` wskazuje, że plik cookie powinien być przesyłany wyłącznie przez bezpieczne połączenie HTTPS od klienta. Gdy ma wartość 'true', plik cookie zostanie ustawiony tylko wtedy, gdy istnieje bezpieczne połączenie.

11 Strefy czasowe

Przegląd

Strefę czasową frontend można ustawić globalnie we frontend i dostosować dla poszczególnych użytkowników.



Jeśli wybrano *System*, dla frontend zostanie użyta strefa czasowa serwera WWW (w tym wartość „date.timezone" z php.ini, jeśli jest ustawiona), natomiast serwer Zabbix będzie używać strefy czasowej maszyny, na której działa.

Note:
Serwer Zabbix będzie używać określonej globalnej/strefy czasowej użytkownika tylko podczas rozwijania makr w powiadomieniach (np. {EVENT.TIME} może zostać rozwinięte do innej strefy czasowej dla każdego użytkownika) oraz do limitu czasu, w którym wysyłane są powiadomienia (zobacz ustawienie „When active" w konfiguracji mediów użytkownika).

Wybór strefy czasowej nie wpływa na format daty/godziny we frontend. Zamiast tego można dostosować język interfejsu (podczas instalacji lub w ustawieniach użytkownika) — wybranie *English (en_US)* spowoduje również włączenie amerykańskiego formatu czasu/daty we frontend.

Konfiguracja

Globalna strefa czasowa:

- może być ustawiona ręcznie podczas instalacji frontendu
- może być zmieniona w *Administracja* → *Ogólne* → *GUI*

Strefa czasowa na poziomie użytkownika:

- może być ustawiona podczas konfigurowania/aktualizacji użytkownika
- może być ustawiona przez każdego użytkownika w jego profilu użytkownika

Zobacz także: Wyrównywanie stref czasowych podczas używania interwałów harmonogramu.

12 Resetowanie hasła

Przegląd Ta sekcja opisuje kroki resetowania haseł użytkowników w Zabbix.

Kroki Zwróć się do twojego administratora Zabbix jeżeli zapomniałeś hasła i nie możesz się zalogować.

Użytkownik Super administrator może zmienić hasła wszystkim użytkownikom w Konfiguracja użytkownika.

Jeśli super administrator zapomniał swojego hasła i nie może się zalogować, należy wykonać następujące zapytanie SQL, aby zastosować domyślne hasło dla użytkownika super administratora (zastąp „Admin” na odpowiednią nazwę użytkownika super administratora):

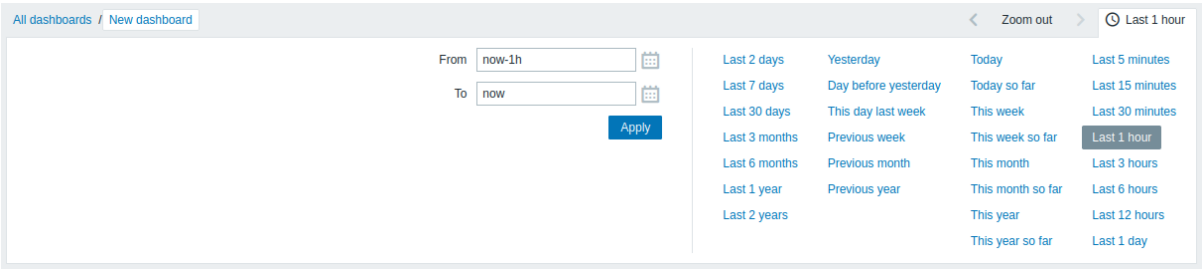
UPDATE users SET passwd = ’$2a$10$ZXIvHAEP2ZM.dLXTm6uPHOMVlARXX7cqjbhM6Fn0cANzkCQBWpMrS’ WHERE username = ’Admin’;

Po uruchomieniu tego zapytania hasło użytkownika zostanie ustawione na *zabbix*. Pamiętaj o zmianie domyślnego hasła przy pierwszym logowaniu.

13 Selektor okresu czasu

Przegląd

Selektor *Time period* umożliwia wybór często używanych okresów jednym kliknięciem myszy. Można go rozwinąć lub zwinąć, klikając kartę *Time period* w prawym górnym rogu.



Opcje takie jak *Today*, *This week* itp. wyświetlają cały okres, łącznie z godzinami/dniami w przyszłości. Opcje takie jak *Today so far*, *This week so far* itp. wyświetlają tylko upłynięte godziny.

Po wybraniu okresu można przesuwać go w czasie do przodu i do tyłu, klikając przyciski strzałek . Przycisk *Zoom out* umożliwia pomniejszenie okresu o 50% w każdym kierunku.

Note:
W przypadku wykresów (z wyjątkiem tych z okresem czasu *Custom*) możliwy jest także wybór wyświetlanego okresu czasu przez zaznaczenie obszaru na wykresie lewym przyciskiem myszy. Po zwolnieniu lewego przycisku myszy wykres zostanie powiększony do zaznaczonego obszaru. Pomniejszanie jest również możliwe przez dwukrotne kliknięcie na wykresie.

Pola *From/To* wyświetlają wybrany okres w składni czasu bezwzględnego (w formacie `Y-m-d H:i:s`) lub w składni czasu względnego. Względny okres czasu może zawierać jedną lub kilka operacji matematycznych (- lub +), na przykład `now-1d` lub `now-1d-2h+5m`.

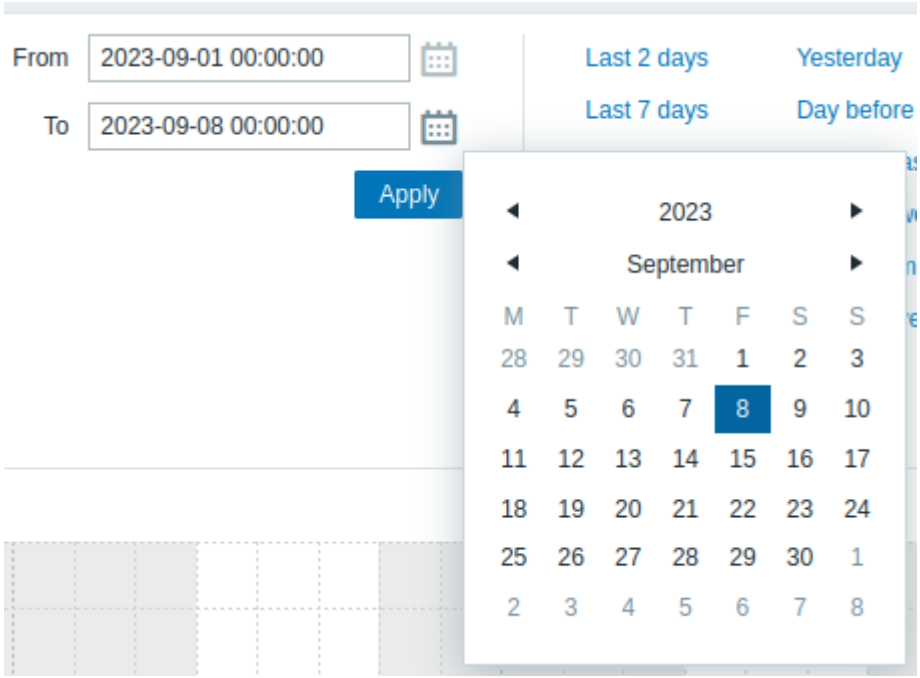
Obsługiwane są następujące skróty czasu względnego:

- `now`
- `s` (sekundy)
- `m` (minuty)
- `h` (godziny)
- `d` (dni)
- `w` (tygodnie)
- `M` (miesiące)
- `y` (lata)

W selektorze *Time period* obsługiwana jest precyzja (na przykład `/M` w `now-1d/M`). Szczegóły precyzji:

Precyzja	Od	Do
`m`	Y-m-d H:m:00	Y-m-d H:m:59
`h`	Y-m-d H:00:00	Y-m-d H:59:59
`d`	Y-m-d 00:00:00	Y-m-d 23:59:59
`w`	Poniedziałek tygodnia 00:00:00	Niedziela tygodnia 23:59:59
`M`	Pierwszy dzień miesiąca 00:00:00	Ostatni dzień miesiąca 23:59:59
`y`	1 stycznia roku 00:00:00	31 grudnia roku 23:59:59

Możliwe jest również wybranie okresu czasu za pomocą *Date picker*. Aby go otworzyć, kliknij ikonę kalendarza obok pól *From/To*.



Note:
W obrębie date pickera można nawigować między rokiem/miesiącem/datą za pomocą `Tab`, `Shift+Tab` oraz klawiszy strzałek na klawiaturze. Naciśnięcie `Enter` potwierdza wybór.

Przykłady

From	To	Wybrany okres
`now/d`	`now/d`	00:00 - 23:59 dzisiaj
`now/d`	`now/d+1d`	00:00 dzisiaj - 23:59 jutro
`now/w`	`now/w`	Poniedziałek 00:00:00 - niedziela 23:59:59 w tym tygodniu
`now-1y/w`	`now-1y/w`	Tydzień od poniedziałku 00:00:00 - niedzieli 23:59:59 rok temu

Attention:
Użycie "now/M+1M" dla parametru *To* może dodać 31 dni, co może spowodować przesunięcie daty o 1-3 dni w zależności od liczby dni w miesiącu. Na przykład, jeśli zostanie użyte w styczniu, wynik może być 02 marca zamiast oczekiwanego 28 lutego. Aby uniknąć tego problemu, użyj "now/M-3d+1M/M", który dokładnie uwzględnia długość miesiąca. Podobnie, jeśli konfigurujesz *From* tak, aby cofać się w czasie, użyj "now/M+3d-1M/M".

19 Najlepsze praktyki

Przegląd Ta sekcja przedstawia najlepsze praktyki dotyczące konfiguracji Zabbix.

Chociaż praktyki te nie są obowiązkowe do działania Zabbix, ich wdrożenie jest zdecydowanie zalecane w celu zapewnienia optymalnego i bezpiecznego korzystania z systemu.

1 Najlepsze praktyki bezpieczeństwa

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące konfigurowania Zabbix w bezpieczny sposób.

Praktyki opisane w tej sekcji nie są wymagane do działania Zabbix, ale są zalecane w celu zapewnienia lepszego bezpieczeństwa systemu.

Kodowanie UTF-8

UTF-8 jest jedynym kodowaniem obsługiwanym przez Zabbix. Wiadomo, że działa bez żadnych luk bezpieczeństwa. Użytkownicy powinni mieć świadomość, że w przypadku korzystania z niektórych innych kodowań istnieją znane problemy związane z bezpieczeństwem.

Ścieżki instalatora Windows

Podczas korzystania z instalatorów Windows zaleca się używanie domyślnych ścieżek dostarczanych przez instalator. Używanie niestandardowych ścieżek bez odpowiednich uprawnień może zagrozić bezpieczeństwu instalacji.

Makra w skryptach globalnych zdefiniowanych przez użytkownika

Aby zwiększyć bezpieczeństwo, zaleca się używanie funkcji makr zamiast zwykłych makr w zdefiniowanych przez użytkownika globalnych skryptach, ponieważ makra nie są automatycznie poprzedzane znakami ucieczki.

Biuletyny bezpieczeństwa Zabbix i baza danych CVE

Zobacz *Biuletyny bezpieczeństwa Zabbix i baza danych CVE*.

Szablony wiadomości e-mail HTML

Podczas tworzenia lub edytowania szablonów wiadomości używanych w wiadomościach e-mail HTML zawsze otaczaj każde makro funkcją makra `htmlencode()`. Na przykład:

```
<b>Problem started</b> at {{EVENT.TIME}.htmlencode()} on {{EVENT.DATE}.htmlencode()}<br><b>Problem name:</
{{EVENT.NAME}.htmlencode()}<br><b>Host:</b> {{HOST.NAME}.htmlencode()}
<br><b>Severity:</b>
{{EVENT.SEVERITY}.htmlencode()}<br><b>Operational
data:</b> {{EVENT.OPDATA}.htmlencode()}<br><b>Original problem ID:</b> {{EVENT.ID}.htmlencode()}<br>{{TRIG
```

Użycie `htmlencode()` zapewnia, że wszelkie znaki HTML w wartościach makr są poprzedzane znakami ucieczki, i zapobiega wstrzyknięciu kodu HTML do powiadomień (na przykład gdy atakujący wstawi do powiadomienia złośliwy link lub link phishingowy).

Uwaga: domyślne wiadomości e-mail HTML dostarczane przez Zabbix mają już zastosowane `htmlencode()` do makr. To zalecenie ma zastosowanie podczas edytowania istniejących szablonów lub tworzenia nowych — przed użyciem szablonu do wysyłania wiadomości e-mail HTML należy sprawdzić, czy makra są zakodowane.

1 Kontrola dostępu

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji kontroli dostępu.

Zasada najmniejszych uprawnień

Konta użytkowników powinny przez cały czas działać z możliwie najmniejszym zakresem uprawnień. Oznacza to, że konta użytkowników w frontend Zabbix, użytkownicy bazy danych lub użytkownik procesów serwer/proxy/agent Zabbix powinni mieć wyłącznie uprawnienia niezbędne do wykonywania zamierzonych funkcji.

Attention:
Przyznanie dodatkowych uprawnień użytkownikowi „zabbix” umożliwi mu dostęp do plików konfiguracyjnych oraz wykonywanie operacji, które mogą zagrozić bezpieczeństwu infrastruktury.

Podczas konfigurowania uprawnień kont użytkowników należy uwzględnić typy użytkowników frontend w Zabbix. Należy pamiętać, że chociaż typ użytkownika *Admin* ma mniej uprawnień niż typ użytkownika *Super Admin*, nadal może zarządzać konfiguracją i wykonywać skrypty niestandardowe.

Note:
Niektóre informacje są dostępne nawet dla użytkowników bez uprawnień uprzywilejowanych. Na przykład, podczas gdy *Alerts* → *Scripts* jest dostępne tylko dla użytkowników *Super Admin*, skrypty można również pobrać za pośrednictwem Zabbix API. W takim przypadku ograniczenie uprawnień do skryptów oraz wykluczenie informacji wrażliwych ze skryptów (na przykład danych dostępowych) może pomóc uniknąć ujawnienia informacji wrażliwych dostępnych w skryptach globalnych.

Bezpieczny użytkownik dla agenta Zabbix

Domyślnie procesy serwera Zabbix, proxy i agenta (lub agent 2) współdzielą jednego użytkownika `zabbix`. Aby zapobiec temu, by agent Zabbix/agent 2 (uruchomiony na tym samym komputerze co serwer/proxy) miał dostęp do poufnych danych w konfiguracji serwera/proxy (na przykład poświadczeń bazy danych), agent powinien być uruchamiany pod innym użytkownikiem:

Dla agenta Zabbix:

1. Utwórz bezpieczną grupę i użytkownika (np. `zabbix-agent`).
2. Ustaw tego użytkownika w parametrze User w pliku konfiguracyjnym agenta.
3. Uruchom ponownie agenta, aby odrzucić uprawnienia i przełączyć się na nowego użytkownika.

Dla agent 2 Zabbix konfiguracja musi zostać zastosowana na poziomie usługi, ponieważ plik konfiguracyjny agent 2 nie obsługuje parametru `User`. Przykład: ZBX-26442.

Cofnięcie dostępu do zapisu do konfiguracji SSL (Windows)

Jeśli skompilowano agent Zabbix w systemie Windows z biblioteką OpenSSL znajdującą się w niezabezpieczonym katalogu (np. `C:\zabbix`, `c:\openssl-64bit`, `C:\OpenSSL-Win64-111-static` lub `C:\dev\openssl`), należy upewnić się, że użytkownikom niebędącym administratorami odebrano uprawnienia do zapisu do tego katalogu. W przeciwnym razie agent będzie wczytywać ustawienia SSL ze ścieżki, która może zostać zmodyfikowana przez nieuprzywilejowanych użytkowników, co stwarza potencjalną lukę bezpieczeństwa.

Wzmacnianie bezpieczeństwa komponentów Zabbix

Niektóre funkcje można wyłączyć, aby zwiększyć bezpieczeństwo komponentów Zabbix:

- wykonywanie skryptów globalnych na serwerze Zabbix można wyłączyć, ustawiając EnableGlobalScripts=0 w konfiguracji serwera;
- wykonywanie skryptów globalnych na proxy Zabbix jest domyślnie wyłączone (można je włączyć, ustawiając EnableRemoteCommands=1 w konfiguracji proxy);
- wykonywanie skryptów globalnych na agentach Zabbix jest domyślnie wyłączone (można je włączyć, dodając parametr AllowKey=system.run[<command>,*] dla każdego dozwolonego polecenia w konfiguracji agenta);
- uwierzytelnianie użytkowników HTTP można wyłączyć, ustawiając `$ALLOW_HTTP_AUTH=false` w pliku konfiguracyjnym frontendu (zabbix.conf.php). Należy pamiętać, że ponowna instalacja frontendu (uruchomienie setup.php) spowoduje usunięcie tego parametru.

Dostęp do ścieżek UNC w systemie Windows przez agent Zabbix

Agenty Zabbix w systemie Windows obsługują ścieżki UNC (udziały SMB, takie jak `\\server\share\file.txt`) w pozycjach takich jak `vfs.file.*`, `vfs.dir.*`, `modbus.get` oraz `perf_counter*`. W niektórych kontekstach może to stanowić zagrożenie bezpieczeństwa.

Gdy system Windows otrzymuje żądanie dostępu do ścieżki UNC, próbuje uwierzytelnić się na tym serwerze. Oznacza to, że złośliwe żądanie do agent Zabbix może ujawnić skrót NTLM serwerowi żądającego. Użytkownicy mogą ograniczyć to ryzyko za pomocą parametrów konfiguracyjnych AllowKey i DenyKey, jeśli zajdzie taka potrzeba.

1 Zabezpieczanie MySQL/MariaDB

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące zabezpieczania bazy danych MySQL/MariaDB.

Note:
W przypadku podstawowej konfiguracji zobacz domyślne instrukcje tworzenia bazy danych MySQL/MariaDB, które obejmują utworzenie użytkownika „zabbix" z pełnymi uprawnieniami do bazy danych Zabbix. Ten użytkownik jest właścicielem bazy danych i ma również niezbędne uprawnienia do modyfikowania struktury bazy danych podczas aktualizacji Zabbix.

Aby zwiększyć bezpieczeństwo, zaleca się tworzenie dodatkowych ról i użytkowników bazy danych z minimalnymi uprawnieniami. Role i użytkownicy powinni być skonfigurowani zgodnie z zasadą najmniejszych uprawnień, to znaczy powinni mieć tylko te uprawnienia, które są niezbędne do wykonywania zamierzonych funkcji.

Tworzenie ról użytkowników

Utwórz następujące role z odpowiadającymi im uprawnieniami:

- **zbx_srv** - rola do uruchamiania serwera Zabbix i proxy:

```
CREATE ROLE 'zbx_srv';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_srv';
FLUSH PRIVILEGES;
```

- **zbx_web** - rola do uruchamiania frontend Zabbix i API:

```
CREATE ROLE 'zbx_web';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_web';
FLUSH PRIVILEGES;
```

- **zbx_bckp** - rola do tworzenia kopii zapasowych tabel:

```
CREATE ROLE 'zbx_bckp';
GRANT LOCK TABLES, TRIGGER, SELECT ON zabbix.* TO 'zbx_bckp';
GRANT process ON *.* TO 'zbx_bckp';
FLUSH PRIVILEGES;
```

Note:
Odtwarzanie tabel i aktualizacja powinny być wykonywane przez właściciela bazy danych.

- **zbx_part** - rola z ograniczonym zestawem uprawnień do partycjonowania bazy danych; pamiętaj, że tę rolę można utworzyć dopiero po utworzeniu bazy danych, ponieważ nadaje ona uprawnienia do określonych tabel bazy danych:

```
CREATE ROLE 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_uint TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_str TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_text TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.history_log TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.trends TO 'zbx_part';
GRANT SELECT, ALTER, DROP ON zabbix.trends_uint TO 'zbx_part';
-- Dla MariaDB: pomiń następną linię (GRANT session_variables_admin ON *.* TO 'zbx_part';)
GRANT session_variables_admin ON *.* TO 'zbx_part';
GRANT SELECT ON zabbix.dbversion TO 'zbx_part';
GRANT SELECT, DELETE ON zabbix.housekeeper TO 'zbx_part';
FLUSH PRIVILEGES;
```

Po utworzeniu ról można je przypisać użytkownikom.

Przypisywanie ról użytkownikom

Aby przypisać utworzone role użytkownikom, utwórz użytkowników i przypisz im odpowiednie role. W razie potrzeby zastąp <user>, <host>, <role> oraz <password>.

```
CREATE USER '<user>'@'<host>' IDENTIFIED BY '<password>';
GRANT '<role>' TO '<user>'@'<host>';
SET DEFAULT ROLE '<role>' TO '<user>'@'<host>';
-- For MariaDB: SET DEFAULT ROLE '<role>' FOR '<user>'@'<host>'
FLUSH PRIVILEGES;
```

Na przykład, aby utworzyć i przypisać rolę do uruchamiania serwer Zabbix i proxy:

```
CREATE USER 'usr_srv'@'localhost' IDENTIFIED BY 'password';
GRANT 'zbx_srv' TO 'usr_srv'@'localhost';
```

```
SET DEFAULT ROLE ALL TO 'usr_srv'@'localhost';
FLUSH PRIVILEGES;
```

2 Zabezpieczanie PostgreSQL/TimescaleDB

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące zabezpieczania bazy danych PostgreSQL.

> **Note:**
> W przypadku podstawowej konfiguracji zobacz domyślne instrukcje tworzenia bazy danych PostgreSQL, które obejmują utworzenie użytkownika „zabbix" z pełnymi uprawnieniami do bazy danych Zabbix. Ten użytkownik jest właścicielem bazy danych i ma również niezbędne uprawnienia do modyfikowania struktury bazy danych podczas aktualizacji Zabbix.

Aby zwiększyć bezpieczeństwo, zaleca się skonfigurowanie bezpiecznego schematu korzystania ze schematów, a także utworzenie dodatkowych ról i użytkowników bazy danych z minimalnymi uprawnieniami. Te role i konta użytkowników powinny być skonfigurowane zgodnie z zasadą najmniejszych uprawnień, to znaczy powinny mieć wyłącznie uprawnienia niezbędne do wykonywania zamierzonych funkcji.

Konfiguracja bazy danych

Utwórz użytkownika, który będzie właścicielem bazy danych, i utwórz bazę danych Zabbix; właściciel bazy danych to użytkownik określony podczas tworzenia bazy danych:

```
createuser -U postgres -h localhost --pwprompt usr_owner
createdb -U postgres -h localhost -O usr_owner -E Unicode -T template0 zabbix
```

> **Note:**
> Czysta instalacja lub aktualizacja bazy danych musi zostać wykonana przez właściciela bazy danych. Wynika to z faktu, że prawo do usunięcia obiektu bazy danych lub zmiany jego definicji jest uprawnieniem przysługującym właścicielowi bazy danych i nie może zostać nadane ani odebrane.

> **Attention:**
> Poniższe polecenia na tej stronie muszą zostać wykonane, gdy połączenie z PostgreSQL jest nawiązane konkretnie z bazą danych `zabbix`.

Utwórz schemat `zabbix` i ustaw właściciela bazy danych (`usr_owner`) jako właściciela tego schematu:

```
CREATE SCHEMA zabbix AUTHORIZATION usr_owner;
```

Skonfiguruj bezpieczny wzorzec użycia schematu:

```
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON DATABASE zabbix FROM PUBLIC;
-- Uwaga: search_path powinien wskazywać na schemat "zabbix":
ALTER ROLE ALL IN DATABASE zabbix SET search_path = "zabbix";
```

Po skonfigurowaniu bazy danych przejdź do tworzenia ról użytkowników.

Tworzenie ról użytkowników

Utwórz następujące role z odpowiadającymi im uprawnieniami:

- **zbx_srv** - rola do uruchamiania serwera Zabbix i proxy:

```
CREATE ROLE zbx_srv;
GRANT CONNECT ON DATABASE zabbix TO zbx_srv;
GRANT USAGE ON SCHEMA zabbix TO zbx_srv;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT DELETE, INSERT, SELECT, UPDATE ON TABLE
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

- **zbx_web** - rola do uruchamiania frontendu Zabbix i API:

```
CREATE ROLE zbx_web;
GRANT CONNECT ON DATABASE zabbix TO zbx_web;
GRANT USAGE ON SCHEMA zabbix TO zbx_web;
```

```
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT DELETE, INSERT, SELECT, UPDATE ON TABLE
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

- **zbx_bckp** - rola do tworzenia kopii zapasowych tabel:

```
CREATE ROLE zbx_bckp;
GRANT CONNECT ON DATABASE zabbix TO zbx_bckp;
GRANT USAGE ON SCHEMA zabbix TO zbx_bckp;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT ON TABLES TO zbx_bckp;
ALTER DEFAULT PRIVILEGES FOR ROLE usr_owner IN SCHEMA zabbix GRANT SELECT, UPDATE, USAGE ON sequences TO z
```

Note:
Przywracanie tabel jest możliwe tylko przez właściciela bazy danych.

Po utworzeniu ról można je przypisać użytkownikom.

Przypisywanie ról użytkowników

Aby przypisać utworzone role użytkowników, utwórz użytkowników i przypisz im odpowiednie role. W razie potrzeby zastąp <user>, <role> i <password>.

```
CREATE USER <user> WITH ENCRYPTED password '<password>';
GRANT <role> TO <user>;
```

Na przykład, aby utworzyć i przypisać rolę do uruchamiania serwera Zabbix i proxy:

```
CREATE USER usr_srv WITH ENCRYPTED password 'password';
GRANT zbx_srv TO usr_srv;
```

Partycjonowanie bazy danych z TimescaleDB

Partycjonowanie bazy danych jest ułatwione dzięki TimescaleDB. Aby korzystać z TimescaleDB, serwer Zabbix wymaga uprawnień właściciela bazy danych.

Jeśli schemat PostgreSQL zabbix został już utworzony w bazie danych zabbix, możesz włączyć TimescaleDB za pomocą następującego polecenia:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb WITH SCHEMA zabbix CASCADE;" | sudo -u postgres psql zabb
```

2 Kryptografia

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji kryptografii.

Konfigurowanie SSL dla frontend Zabbix

W systemach opartych na RHEL zainstaluj pakiet mod_ssl:

```
dnf install mod_ssl
```

Utwórz katalog na klucze SSL:

```
mkdir -p /etc/httpd/ssl/private
chmod 700 /etc/httpd/ssl/private
```

Utwórz certyfikat SSL:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/httpd/ssl/private/apache-selfsigned.key -
```

Wypełnij odpowiednio wyświetlane monity. Najważniejsza jest linia, w której podawana jest wartość Common Name. Musisz wpisać nazwę domeny, która ma być powiązana z twoim serwer. Możesz zamiast tego podać publiczny adres IP, jeśli nie masz nazwy domeny.

```
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 []:
```

Edytuj plik konfiguracji SSL 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
```

Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
systemctl restart httpd.service
```

3 Serwer WWW

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji serwera WWW.

Wymuszanie przekierowania głównego URL do Zabbix SSL

W systemach opartych na RHEL dodaj virtual host do konfiguracji Apache (`/etc/httpd/conf/httpd.conf`) i ustaw stałe przekierowanie dla katalogu głównego dokumentów do adresu URL Zabbix SSL. Zwróć uwagę, że `example.com` należy zastąpić rzeczywistą nazwą serwera.

```
#### Dodaj linie:

<VirtualHost *:*>
    ServerName example.com
    Redirect permanent / https://example.com
</VirtualHost>
```

Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
systemctl restart httpd.service
```

Włączanie HTTP Strict Transport Security (HSTS) na serwerze WWW

Aby chronić frontend Zabbix przed atakami obniżenia wersji protokołu, zalecamy włączenie polityki HSTS na serwerze WWW.

Aby włączyć politykę HSTS dla frontend Zabbix w konfiguracji Apache, wykonaj następujące kroki:

1. Zlokalizuj plik konfiguracyjny swojego hosta wirtualnego:
   - `/etc/httpd/conf/httpd.conf` w systemach opartych na RHEL
   - `/etc/apache2/sites-available/000-default.conf` w Debian/Ubuntu
2. Dodaj następującą dyrektywę do pliku konfiguracyjnego swojego hosta wirtualnego:

```
<VirtualHost *:*>
    Header set Strict-Transport-Security "max-age=31536000"
</VirtualHost>
```

3. Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
#### On RHEL-based systems:
systemctl restart httpd.service

#### On Debian/Ubuntu
systemctl restart apache2.service
```

Wymuszanie bezpiecznych i SameSite cookies sesji w Zabbix

Podczas konfigurowania Zabbix należy wymusić atrybuty secure i SameSite dla cookies sesji, aby zwiększyć bezpieczeństwo i zapobiegać atakom typu cross-site request forgery (CSRF). Jednak wymuszenie `SameSite=Strict` może powodować problemy w niektórych scenariuszach, takich jak:

- Widgety URL na pulpicie wyświetlające komunikat „użytkownik nie jest zalogowany” podczas osadzania iframe z tej samej domeny.
- Użytkownicy uzyskujący dostęp do pulpitu przez HTTP zamiast HTTPS mogą napotkać problemy z logowaniem.
- Brak możliwości udostępniania adresów URL do określonych sekcji menu Zabbix lub hostów.

Aby ograniczyć te problemy, użytkownicy powinni mieć możliwość dostosowania polityki SameSite.

1. Bezpieczne cookies

Ustawienie flagi `secure` zapewnia, że cookies są przesyłane wyłącznie przez HTTPS, co zapobiega ich ujawnieniu przez niezaszyfrowane połączenia.

Aby włączyć bezpieczne cookies w Zabbix, dodaj lub zmodyfikuj następujące ustawienie w konfiguracji serwera WWW:

Dla Apache:

```
Header always edit Set-Cookie ^(.*)$ $1;Secure
```

Dla Nginx:

```
proxy_cookie_path / "/; Secure";
```

Upewnij się, że dostęp do frontend Zabbix odbywa się przez HTTPS; w przeciwnym razie cookies z flagą `Secure` nie będą wysyłane.

2. Konfigurowanie atrybutu SameSite

Ustawienia serwera WWW mogą również wymuszać atrybut SameSite:

Dla Apache:

```
<IfModule mod_headers.c>
    Header onsuccess edit Set-Cookie (.*) "$1; SameSite=Strict"
</IfModule>
```

Dla Nginx (wersja 1.19.3+):

```
proxy_cookie_flags ~ samesite=Strict; # Zastąp ~ przez 'zbx_session', aby zwiększyć precyzję
```

Włączanie Content Security Policy (CSP) na serwerze WWW

Aby chronić frontend Zabbix przed atakami typu Cross Site Scripting (XSS), wstrzykiwaniem danych i podobnymi zagrożeniami, zalecamy włączenie Content Security Policy na serwerze WWW. W tym celu skonfiguruj serwer WWW tak, aby zwracał nagłówek HTTP.

Attention:
Poniższa konfiguracja nagłówka CSP dotyczy wyłącznie domyślnej instalacji frontendu Zabbix oraz przypadków, gdy cała zawartość pochodzi z domeny witryny (z wyłączeniem subdomen). Inna konfiguracja nagłówka CSP może być wymagana, jeśli na przykład konfigurujesz widżet *URL* tak, aby wyświetlał zawartość z subdomen witryny lub domen zewnętrznych, przełączasz się z *OpenStreetMap* na inny silnik map albo dodajesz zewnętrzny CSS lub widżety. Jeśli używasz metody uwierzytelniania wieloskładnikowego Duo Universal Prompt, pamiętaj, aby dodać „duo.com" do dyrektywy CSP w pliku konfiguracji virtual hosta.

Aby włączyć CSP dla frontendu Zabbix w konfiguracji Apache, wykonaj następujące kroki:

1. Zlokalizuj plik konfiguracji virtual hosta:

- `/etc/httpd/conf/httpd.conf` w systemach opartych na RHEL
- `/etc/apache2/sites-available/000-default.conf` w Debian/Ubuntu

2. Dodaj następującą dyrektywę do pliku konfiguracji virtual hosta:

```
<VirtualHost *:*>
    Header set Content-Security-Policy: "default-src 'self' *.openstreetmap.org; script-src 'self' 'unsafe
</VirtualHost>
```

3. Uruchom ponownie usługę Apache, aby zastosować zmiany:

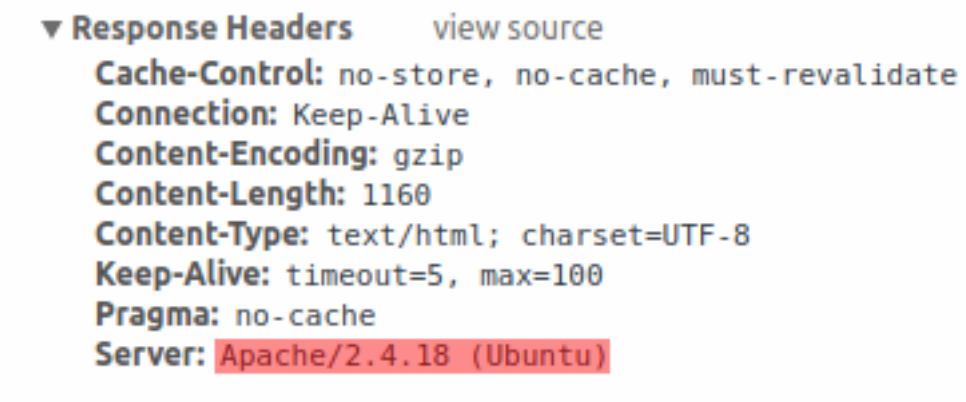
```
#### W systemach opartych na RHEL:
systemctl restart httpd.service

#### W Debian/Ubuntu
systemctl restart apache2.service
```

Wyłączanie ujawniania informacji o serwerze WWW

Aby zwiększyć bezpieczeństwo, zaleca się wyłączenie wszystkich sygnatur serwera WWW.

Domyślnie serwer WWW ujawnia sygnaturę oprogramowania:



Sygnaturę można wyłączyć, dodając następujące parametry do pliku konfiguracyjnego Apache:

```
ServerSignature Off
ServerTokens Prod
```

Sygnaturę PHP (nagłówek HTTP X-Powered-By) można wyłączyć, zmieniając plik konfiguracyjny `php.ini` (domyślnie sygnatura jest wyłączona):

```
expose_php = Off
```

Aby zmiany w plikach konfiguracyjnych zostały zastosowane, wymagane jest ponowne uruchomienie serwera WWW.

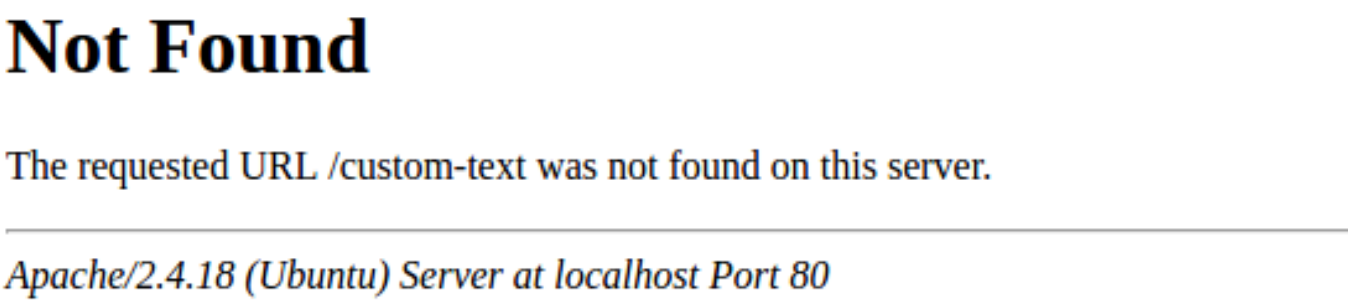
Dla dodatkowego bezpieczeństwa można użyć narzędzia *mod_security* z Apache (pakiet *libapache2-mod-security2*). To narzędzie umożliwia usunięcie sygnatury serwera zamiast usuwania jedynie wersji z sygnatury serwera. Sygnaturę serwera można zmienić na dowolną wartość, ustawiając ”SecServerSignature” na żądaną wartość po zainstalowaniu *mod_security*.

Aby uzyskać pomoc dotyczącą usuwania/zmiany sygnatur oprogramowania, zapoznaj się z dokumentacją swojego serwera WWW.

Wyłączanie domyślnych stron błędów serwera WWW

Aby uniknąć ujawnienia informacji, zaleca się wyłączenie domyślnych stron błędów.

Domyślnie serwer WWW używa wbudowanych stron błędów:



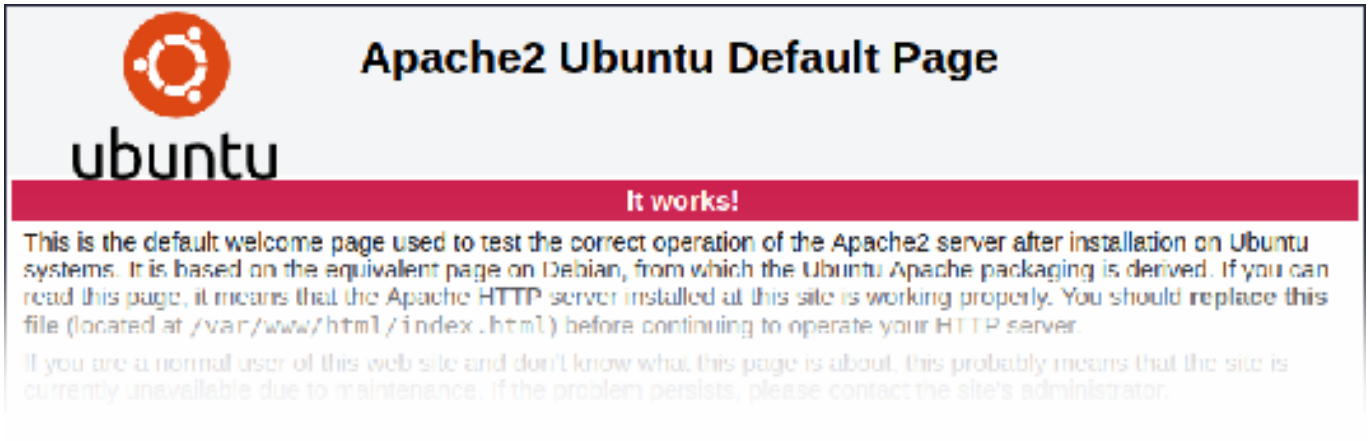
Te domyślne strony błędów należy zastąpić/usunąć. Na przykład dyrektywy „ErrorDocument” można użyć do zdefiniowania niestandardowej strony/tekstu błędu dla serwera WWW Apache.

Aby uzyskać informacje o tym, jak zastąpić/usunąć domyślne strony błędów, zapoznaj się z dokumentacją swojego serwera WWW.

Usuwanie strony testowej serwera WWW

Aby uniknąć ujawnienia informacji, zaleca się usunięcie strony testowej serwera WWW.

Domyślnie katalog główny serwera WWW Apache zawiera testową stronę `index.html`:



Aby uzyskać informacje o tym, jak usunąć domyślne strony testowe, zapoznaj się z dokumentacją swojego serwera WWW.

Ustaw nagłówek odpowiedzi HTTP X-Frame-Options

Domyślnie Zabbix jest skonfigurowany z parametrem *Use X-Frame-Options HTTP header* ustawionym na `SAMEORIGIN`. Oznacza to, że zawartość może być ładowana tylko w ramce, która ma to samo pochodzenie co sama strona.

Elementy frontend Zabbix, które pobierają zawartość z zewnętrznych adresów URL (a mianowicie widżet URL na pulpicie), wyświetlają pobraną zawartość w piaskownicy z włączonymi wszystkimi ograniczeniami sandbox.

Ustawienia te zwiększają bezpieczeństwo frontend Zabbix i zapewniają ochronę przed atakami XSS oraz clickjacking. Użytkownicy *Super admin* mogą w razie potrzeby zmienić parametry *Use iframe sandboxing* i *Use X-Frame-Options HTTP header*. Przed zmianą ustawień domyślnych należy dokładnie rozważyć ryzyko i korzyści. Nie zaleca się całkowitego wyłączania iframe sandboxing ani nagłówka odpowiedzi HTTP X-Frame-Options.

Ukrywanie pliku z listą popularnych haseł

Aby zwiększyć złożoność ataków brute force na hasła, zaleca się ograniczenie dostępu do pliku `ui/data/top_passwords.txt`. Ten plik zawiera listę najczęściej używanych oraz specyficznych kontekstowo haseł i uniemożliwia użytkownikom ustawianie takich haseł (jeśli parametr *Avoid easy-to-guess passwords* jest włączony w polityce haseł).

Aby ograniczyć dostęp do pliku `top_passwords.txt`, zmodyfikuj konfigurację swojego serwera WWW.

W Apache dostęp do pliku można ograniczyć za pomocą pliku `.htaccess`:

```
<Files "top_passwords.txt">
    Order Allow,Deny
    Deny from all
</Files>
```

W NGINX dostęp do pliku można ograniczyć za pomocą dyrektywy `location`:

```
location = /data/top_passwords.txt {
    deny all;
    return 404;
}
```

2 Najlepsze praktyki konfiguracji

Przegląd

Ta sekcja przedstawia najlepsze praktyki dotyczące konfiguracji Zabbix w celu osiągnięcia optymalnej wydajności i łatwości użytkowania. Zalecenia opierają się na wskazówkach deweloperów Zabbix oraz praktycznym doświadczeniu trenerów Zabbix i inżynierów wsparcia.

Każda instalacja Zabbix jest unikalna i niektóre z tych wytycznych mogą nie być odpowiednie dla Twojej konkretnej konfiguracji. Zaleca się jednak, aby w miarę możliwości stosować się do tych wytycznych, by uniknąć typowych potencjalnych problemów.

Note:
Jeśli uważasz, że tę stronę można ulepszyć, chętnie poznamy Twoją opinię! Zaznacz odpowiedni fragment tekstu i naciśnij **ctrl+Enter**, aby zgłosić błąd lub przekazać swoją opinię.

Hosty i pozycje Definiowanie hosta

Host w Zabbix nie jest fizyczną maszyną ani urządzeniem, lecz logiczną jednostką. Do celów monitorowania możesz tworzyć oddzielne hosty dla bazy danych lub na przykład maszyny wirtualnej. Alternatywnie możesz utworzyć ogólny host *Laptop Johna* i monitorować wszystkie metryki w ramach tego hosta.

Najlepszą praktyką jest utworzenie oddzielnego hosta dla każdej niezależnej instancji, takiej jak maszyna wirtualna, baza danych, kontener lub przełącznik sieciowy. Korzystając z tego podejścia:

1. Unikniesz bałaganu w danych monitorowania dzięki oddzielnym pozycjom, wyzwalaczom i powiadomieniom o alertach dla każdego hosta.
2. Precyzyjnie dostosujesz poziomy dostępu użytkowników. Możesz skonfigurować role użytkowników, aby przyznać dostęp do przeglądania i/lub konfiguracji tylko określonych hostów. Zobacz także zasadę najmniejszych uprawnień.

Hosty ze zduplikowanymi pozycjami

Jeśli masz kilka podobnych hostów, takich jak *Przełącznik sieciowy 1* i *Przełącznik sieciowy 2*, Zabbix udostępnia wiele sposobów na szybkie odtworzenie hosta. Możesz po prostu sklonować hosta wraz ze wszystkimi jego metrykami, naciskając przycisk Clone, ale w takim przypadku, aby później zaktualizować pozycję, trzeba będzie zrobić to ręcznie na każdym hoście.

Najlepszą praktyką jest utworzenie szablonu ze wszystkimi wymaganymi metrykami, na przykład *Szablon przełącznika sieciowego*. Następnie pogrupuj podobne hosty w grupę hostów; w powyższym przykładzie może to być *Przełączniki sieciowe*. Teraz w sekcji *Data Collection -> Hosts* możesz filtrować wszystkie hosty według grupy hostów i użyć przycisku *Mass update*, aby podłączyć szablon do wszystkich swoich przełączników sieciowych.

Pozycje zależne

Aby zminimalizować liczbę żądań do monitorowanego obiektu, Zabbix umożliwia tworzenie pozycji nadrzędnych i zależnych. W takim przypadku pozycja nadrzędna zbiera duży zestaw informacji w ramach jednego żądania. Następnie można skonfigurować pozycje zależne tak, aby wyodrębniały określone fragmenty danych z tego zbioru za pomocą preprocessing i zapisywały je jako osobne metryki.

Na przykład pozycja nadrzędna może pobierać odpowiedź JSON lub XML zawierającą wiele metryk albo wykonywać zapytanie do bazy danych, które zwraca wiele kolumn danych (np. liczbę otwartych połączeń, przerwanych połączeń, maksymalną dozwoloną liczbę jednoczesnych połączeń oraz łączną skumulowaną liczbę połączeń od momentu uruchomienia), a pozycje zależne będą analizować i zapisywać każdą wymaganą wartość osobno.

Najlepszą praktyką dla tej konfiguracji jest odrzucanie historii pozycji nadrzędnej zaraz po zebraniu danych i przechowywanie wyłącznie danych pozycji zależnych.

Serwery i proxy

Jeśli wszystkie hosty znajdują się w tej samej sieci lokalnej co serwer Zabbix i nie ma obaw dotyczących skalowalności ani wydajności, możesz nie potrzebować proxy. W większych lub bardziej złożonych środowiskach bezpośrednie monitorowanie hostów przez serwer Zabbix może nie być wystarczające. Dodanie proxy i przypisanie części hostów do tego proxy pozwala na bardziej równomierne rozłożenie obciążenia.

Najlepszą praktyką jest dodanie proxy Zabbix, gdy:

1. Monitorujesz wiele hostów przy użyciu różnych metod zbierania metryk za zaporą sieciową. Proxy będzie zbierać dane z hostów i przekazywać je do serwera Zabbix, zmniejszając potrzebę otwierania wielu portów zapory sieciowej.
2. Monitorujesz zdalne lokalizacje, oddziały i/lub sieci. W przypadku przerwy w połączeniu sieciowym między serwerem Zabbix a zdalnymi lokalizacjami, proxy Zabbix wdrożone w tych lokalizacjach będą kontynuować zbieranie danych i prześlą zebrane dane z powrotem do serwera Zabbix po przywróceniu połączenia sieciowego.
3. Masz wdrożenie na dużą skalę i chcesz zmniejszyć obciążenie serwera Zabbix oraz poprawić wydajność. Definicja wdrożenia na dużą skalę jest bardzo szeroka i zależy nie tylko od liczby hostów, ale także od liczby wartości zbieranych na sekundę.

Sekretne makra

Możesz używać sekretnych makr użytkownika jako sekretnego tekstu lub sekretnych makr vault.

Aby zwiększyć bezpieczeństwo podczas korzystania z sekretnych makr vault, zaleca się skonfigurowanie pobierania wartości makr niezależnie przez serwer Zabbix i proxy Zabbix. Domyślnie wartości sekretnych makr są pobierane przez serwer Zabbix i przekazywane do proxy Zabbix.

20 API

Przegląd API Zabbix umożliwia programowe pobieranie i modyfikowanie konfiguracji Zabbix oraz zapewnia dostęp do danych historycznych. Jest ono szeroko wykorzystywane do:

- tworzenia nowych aplikacji współpracujących z Zabbix;
- integracji Zabbix z oprogramowaniem innych producentów;
- automatyzacji rutynowych zadań.

API Zabbix jest interfejsem API opartym na HTTP i jest dostarczane jako część web frontend. Wykorzystuje protokół JSON-RPC 2.0, co oznacza dwie rzeczy:

- API składa się z zestawu oddzielnych metod;
- żądania i odpowiedzi między klientami a API są kodowane przy użyciu formatu JSON.

Więcej informacji o protokole i formacie JSON można znaleźć w specyfikacji JSON-RPC 2.0 oraz na stronie głównej formatu JSON.

Więcej informacji o integrowaniu funkcjonalności Zabbix z aplikacjami Python można znaleźć w bibliotece Python dla Zabbix.

> **Note:**
> Dostęp użytkownika w Zabbix, obejmujący zarówno konfigurację, jak i dane historyczne, zależy od typu użytkownika, przypisanej roli użytkownika oraz grup użytkowników.

Struktura API składa się z wielu metod, które są umownie pogrupowane w oddzielne API. Każda z metod wykonuje jedno konkretne zadanie. Na przykład metoda `host.create` należy do API *host* i służy do tworzenia nowych hostów. Historycznie API są czasami określane jako „klasy”.

> **Note:**
> Większość API zawiera co najmniej cztery metody: `get`, `create`, `update` i `delete`, służące odpowiednio do pobierania, tworzenia, aktualizowania i usuwania danych, jednak niektóre API mogą udostępniać zupełnie inny zestaw metod.

Wykonywanie żądań Po skonfigurowaniu frontend możesz używać zdalnych żądań HTTP do wywoływania API. Aby to zrobić, musisz wysyłać żądania HTTP POST do pliku `api_jsonrpc.php` znajdującego się w katalogu frontend. Na przykład, jeśli Twój frontend Zabbixa jest zainstalowany pod adresem `https://example.com/zabbix`, żądanie HTTP wywołujące metodę `apiinfo.version` może wyglądać następująco:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"apiinfo.version","params":{},"id":1}'
```

Żądanie musi mieć nagłówek `Content-Type` ustawiony na jedną z następujących wartości: `application/json-rpc`, `application/json` lub `application/jsonrequest`.

Obiekt żądania musi zawierać następujące właściwości:

- `jsonrpc` - wersja protokołu JSON-RPC używana przez API (Zabbix API implementuje wersję JSON-RPC 2.0);
- `method` - wywoływana metoda API;
- `params` - parametry przekazywane do metody API;
- `id` - dowolny identyfikator żądania (jeśli zostanie pominięty, API traktuje żądanie jako powiadomienie).

Jeśli żądanie jest poprawne, odpowiedź zwrócona przez API powinna wyglądać następująco:

```
{
    "jsonrpc": "2.0",
    "result": "7.4.0",
    "id": 1
}
```

Obiekt odpowiedzi z kolei zawiera następujące właściwości:

- `jsonrpc` - wersja protokołu JSON-RPC;
- `result` - dane zwrócone przez metodę;
- `id` - identyfikator odpowiadającego mu żądania.

Przykładowy przepływ pracy Poniższa sekcja przeprowadzi Cię przez kilka przykładów użycia bardziej szczegółowo.

Uwierzytelnianie Aby uzyskać dostęp do jakichkolwiek danych w Zabbix, musisz:

- użyć istniejącego tokenu API (utworzonego w Zabbix frontend lub przy użyciu Token API);
- użyć tokenu uwierzytelniającego uzyskanego za pomocą metody user.login.

Na przykład, jeśli chcesz uzyskać nowy token uwierzytelniający, logując się jako standardowy użytkownik *Admin*, żądanie JSON będzie wyglądać następująco:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"user.login","params":{"username":"Admin","password":"zabbix"},"id":1}
```

Jeśli podasz poprawne dane logowania, odpowiedź zwrócona przez API powinna zawierać token uwierzytelniający użytkownika:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Metody autoryzacji Za pomocą nagłówka "Authorization"

Wszystkie żądania API wymagają uwierzytelnienia lub tokenu API. Dane uwierzytelniające można przekazać, używając nagłówka Authorization w żądaniu:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33'
```

Attention:
Jeśli występują problemy z uwierzytelnianiem, zobacz Przekazywanie nagłówka Authorization.

API Zabbix akceptuje nagłówki bez rozróżniania wielkości liter (np. `authorization`, `Authorization` i `AUTHORIZATION` są traktowane tak samo).

Nagłówek Authorization jest obsługiwany w żądaniach między źródłami (CORS).

Za pomocą pliku cookie Zabbix

Plik cookie *"zbx_session"* jest używany do autoryzacji żądania API z interfejsu Zabbix UI wykonywanego przy użyciu JavaScript (z modułu lub niestandardowego widżetu).

Pobieranie hostów Masz teraz prawidłowy token uwierzytelnienia użytkownika (przedstawiony jako zmienna w poniższych przykładach), którego można użyć do uzyskania dostępu do danych w Zabbix. Na przykład możesz użyć metody host.get, aby pobrać identyfikatory, nazwy hostów i interfejsy wszystkich skonfigurowanych hostów:

Żądanie:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data @data.json
```

Note:
`data.json` to plik zawierający zapytanie JSON. Zamiast pliku możesz przekazać zapytanie w argumencie `--data`.

data.json

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "hostid",
            "host"
        ],
```

```
        "selectInterfaces": [
            "interfaceid",
            "ip"
        ]
    },
    "id": 2
}
```

Obiekt odpowiedzi będzie zawierał żądane dane o hostach:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "interfaces": [
                {
                    "interfaceid": "1",
                    "ip": "127.0.0.1"
                }
            ]
        }
    ],
    "id": 2
}
```

Note:
Ze względów wydajnościowych zawsze zaleca się podawanie na liście właściwości obiektu, które chcesz pobrać. Dzięki temu unikniesz pobierania wszystkiego.

Tworzenie nowej pozycji Teraz utwórz nową pozycję na hoście „Zabbix server", używając danych uzyskanych z poprzedniego żądania `host.get`. Można to zrobić za pomocą metody item.create:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"item.create","params":{"name":"Free disk space on /home/joe/","key_":
```

Pomyślna odpowiedź będzie zawierać ID nowo utworzonej pozycji, którego można użyć do odwoływania się do tej pozycji w kolejnych żądaniach:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 3
}
```

Note:
Metoda `item.create`, podobnie jak inne *metody create*, może również przyjmować tablice obiektów i tworzyć wiele pozycji za pomocą jednego wywołania API.

Tworzenie wielu wyzwalaczy Jeśli więc *metody create* akceptują tablice, możesz dodać wiele wyzwalaczy, na przykład takie:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
```

```
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"trigger.create","params":[{"description":"Obciążenie procesora jest z
```

Pomyślna odpowiedź będzie zawierać identyfikatory nowo utworzonych wyzwalaczy:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 4
}
```

Aktualizowanie pozycji Włącz pozycję, ustawiając jej status na „0”:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"item.update","params":{"itemid":"10092","status":0},"id":5}'
```

Pomyślna odpowiedź będzie zawierać ID zaktualizowanej pozycji:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 5
}
```

Note:
Metoda `item.update`, podobnie jak inne *metody aktualizacji*, może również przyjmować tablice obiektów i aktualizować wiele pozycji za pomocą jednego wywołania API.

Aktualizacja wielu wyzwalaczy Włącz wiele wyzwalaczy, ustawiając ich status na ”0”:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"trigger.update","params":[{"triggerid":"13938","status":0},{"triggeri
```

Pomyślna odpowiedź będzie zawierać identyfikatory zaktualizowanych wyzwalaczy:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938",
            "13939"
        ]
    },
    "id": 6
}
```

Note:
Jest to preferowana metoda aktualizacji. Niektóre metody API, takie jak `host.massupdate`, umożliwiają napisanie prostszego kodu. Nie zaleca się jednak używania tych metod, ponieważ zostaną usunięte w przyszłych wydaniach.

Obsługa błędów Do tej pory wszystko, czego próbowano, działało poprawnie. Ale co by się stało, gdyby wykonać nieprawidłowe wywołanie API? Spróbuj utworzyć kolejny host, wywołując host.create, ale pomijając wymagany parametr `groups`:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer ${AUTHORIZATION_TOKEN}' \
  --header 'Content-Type: application/json-rpc' \
  --data '{"jsonrpc":"2.0","method":"host.create","params":{"host":"Linux server","interfaces":[{"type":1,
```

Odpowiedź będzie wtedy zawierać komunikat o błędzie:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32602,
        "message": "Invalid params.",
        "data": "No groups for host \"Linux server\"."
    },
    "id": 7
}
```

Jeśli wystąpił błąd, zamiast właściwości `result` obiekt odpowiedzi będzie zawierał właściwość `error` z następującymi danymi:

- `code` - kod błędu;
- `message` - krótkie podsumowanie błędu;
- `data` - bardziej szczegółowy komunikat o błędzie.

Błędy mogą wystąpić w różnych przypadkach, na przykład przy użyciu nieprawidłowych wartości wejściowych, przekroczeniu limitu czasu sesji lub próbie uzyskania dostępu do nieistniejących obiektów. Aplikacja powinna potrafić w poprawny sposób obsługiwać tego rodzaju błędy.

Wersje API Aby uprościć wersjonowanie API, od Zabbix 2.0.4 wersja API odpowiada wersji samego Zabbix. Możesz użyć metody apiinfo.version, aby sprawdzić wersję API, z którą pracujesz. Może to być przydatne przy dostosowywaniu aplikacji do korzystania z funkcji specyficznych dla danej wersji.

Zabbix gwarantuje wsteczną kompatybilność funkcji w obrębie wersji głównej. W przypadku wprowadzania zmian niezgodnych wstecznie między głównymi wydaniami Zabbix zwykle pozostawia stare funkcje jako przestarzałe w następnym wydaniu, a usuwa je dopiero w kolejnym. Sporadycznie Zabbix może usuwać funkcje między głównymi wydaniami bez zapewnienia jakiejkolwiek wstecznej kompatybilności. Ważne jest, aby nigdy nie polegać na żadnych przestarzałych funkcjach i jak najszybciej przechodzić na nowsze alternatywy.

> **Note:**
> Możesz śledzić wszystkie zmiany wprowadzone w API w dzienniku zmian API.

Dalsza lektura Masz już wystarczającą wiedzę, aby rozpocząć pracę z API Zabbix, jednak nie poprzestawaj na tym. W ramach dalszej lektury zalecamy zapoznanie się z listą dostępnych API.

Odniesienie do metody

Ta sekcja zawiera przegląd funkcji udostępnianych przez Zabbix API i pomoże Ci odnaleźć się w dostępnych klasach i metodach.

Monitorowanie API Zabbix umożliwia dostęp do historii i innych danych zebranych podczas monitorowania.

Pulpity nawigacyjne

Zarządzaj pulpitami i na ich podstawie twórz zaplanowane raporty.

API pulpitu | API szablonu pulpitu | API raportu

Klaster o wysokiej dostępności

Pobierz listę węzłów serwera i ich status.

API klastra wysokiej dostępności

Historia

Pobierz wartości historyczne zebrane przez procesy monitorowania Zabbix do prezentacji lub dalszego przetwarzania.

API historii

Trendy

Pobierz wartości trendów obliczone przez serwer Zabbix do prezentacji lub dalszego przetwarzania.

Interfejs API trendów

Wydarzenia

Pobierz zdarzenia generowane przez wyzwalacze, wykrywanie sieci i inne systemy Zabbix do bardziej elastycznego zarządzania sytuacjami lub integrację z narzędziami firm trzecich.

Interfejs API zdarzeń

Problemy

Pobierz problemy zgodnie z podanymi parametrami.

Problem API

Mapy

Konfiguruj mapy, aby tworzyć szczegółowe, dynamiczne reprezentacje Twojej infrastruktury IT.

API mapy

Zadania

Interakcja z menedżerem zadań serwera Zabbix, tworzenie zadań i pobieranie odpowiedzi.

API zadań

Usługi API Zabbix umożliwia dostęp do danych zebranych podczas monitorowania usług.

Umowa dotycząca poziomu usług

Zdefiniuj Wytyczne Poziomu Usług (SLO), pobierz szczegółowe Wskaźniki Poziomu Usług (SLI) wydajności usługi.

SLA API

Usługi

Zarządzaj usługami do monitorowania poziomu usług i pobieraj szczegółowe informacje o SLA dowolnej usługi.

API usługi

Zbieranie danych API Zabbix umożliwia zarządzanie konfiguracją systemu monitoringu.

Hosty i grupy hostów

Zarządzaj grupami hostów, hostami i wszystkim, co jest z nimi związane, w tym interfejsami hosta, makrami hosta i okresami konserwacji.

API hosta | APi Grupy gospodarzy | API interfejsu hosta | API makr użytkownika | API map wartości | API konserwacji

Elementy

Zdefiniuj elementy do monitorowania.

API elementu

Wyzwalacze

Skonfiguruj wyzwalacze, aby powiadamiały Cię o problemach w systemie. Zarządzaj zależnościami wyzwalacza.

API Wyzwalaczy

Wykresy

Edytuj wykresy lub oddzielne elementy wykresu, aby lepiej prezentować zebrane dane.

API Wykresów | API elementu wykresu

Szablony i grupy szablonów

Zarządzaj szablonami i łącz je z hostami lub innymi szablonami.

API szablonów | API grup szablonów | API map wartości

Wykrywanie niskiego poziomu

Skonfiguruj reguły wykrywania niskiego poziomu, jak również prototypy elementów, wyzwalaczy i wykresów do monitorowania jednostek dynamicznych.

API reguł LLD | API prototypu elementu | API prototypu wyzwalacza | API prototypu wykresu | API prototypu hosta

Korelacja zdarzeń

Twórz niestandardowe reguły korelacji zdarzeń.

API korelacji

Wykrywanie sieci

Zarządzaj regułami wykrywania na poziomie sieci, aby automatycznie wyszukiwać i monitorować nowe hosty. Uzyskaj pełny dostęp do informacji o wykrytych usługach i hostah.

API reguły wykrywania | API sprawdzenia wykrywania | API wykrycia hosta | API wykrycia usługi

Eksport i import

Eksportuj i importuj dane konfiguracyjne Zabbix do tworzenia kopii zapasowych konfiguracji, migracji lub aktualizacji konfiguracji na dużą skalę.

API konfiguracjiI

Monitorowanie sieci WWW

Skonfiguruj scenariusze internetowe, aby monitorować swoje aplikacje i usługi internetowe.

API monitoringu WWW

Alerty API Zabbix umożliwia zarządzanie akcjami i alertami systemu monitoringu.

Akcje i alerty

Zdefiniuj akcje i operacje, aby powiadamiać użytkowników o określonych zdarzeniach lub automatycznie wykonywać zdalne polecenia. Uzyskaj dostęp do informacji o generowanych alertach i ich odbiorcach.

API akcji | API alarmów

Typy mediów

Skonfiguruj typy mediów i wiele sposobów, na jakie użytkownicy będą otrzymywać alerty.

API typów mediów

Skrypty

Konfiguruj i wykonuj skrypty, które pomogą Ci w codziennych zadaniach.

API skryptu

Użytkownicy API Zabbix umożliwia zarządzanie użytkownikami systemu monitoringu.

Użytkownicy i grupy użytkowników

Dodaj użytkowników, którzy będą mieli dostęp do Zabbix, przypisz ich do grup użytkowników i nadaj uprawnienia. Utwórz role do szczegółowego zarządzania prawami użytkowników.

API użytkowników | API grup użytkowników | API katalogu użytkowników | API ról użytkowników

Tokeny API

Zarządzaj tokenami autoryzacyjnymi.

API tokena

Uwierzytelnianie

Zmień opcje konfiguracji uwierzytelniania.

API uwierzytelniania

Administracja Za pomocą API Zabbix można zmieniać ustawienia administracyjne systemu monitoringu.

Ogólne

Zmień wybrane globalne opcje konfiguracji.

API autorejestracji | API map ikon | API obrazów | API ustawień | API wyrażeń regularnych | API modułów | API konektorów

Dziennik audytu

Śledź zmiany konfiguracji dokonane przez każdego użytkownika.

API dziennika audytu

Porządkowanie

Skonfiguruj porządkowanie.

Housekeeping API

Proxy i grupy proxy

Zarządzaj proxy używanymi w rozproszonej konfiguracji monitorowania.

Proxy API | API grup proxy

Makra

Zarządzaj makrami.

API makr użytkownika

Informacje o interfejsie API Pobierz wersję API Zabbix, aby Twoja aplikacja mogła używać funkcji specyficznych dla wersji.

API informacje o API

Akcja

Ta klasa jest przeznaczona do pracy z akcjami.

Odwołania do obiektów:

- Akcja
- Operacja akcji
  - Wiadomość operacji akcji
  - Warunek operacji akcji
- Operacja odzyskiwania akcji
- Operacja aktualizacji akcji
- Filtr akcji
  - Warunek filtra akcji

Dostępne metody:

- action.create - tworzenie nowych akcji
- action.delete - usuwanie akcji
- action.get - pobieranie akcji
- action.update - aktualizowanie akcji

Obiekt Akcji

Następujące obiekty są bezpośrednio powiązane z API `action`.

Akcja

Obiekt akcji ma następujące właściwości.

Właściwość	Typ	Opis
actionid	ID	ID akcji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
esc_period	string	Domyślny czas trwania kroku operacji. Musi wynosić co najmniej 60 sekund. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Zachowanie właściwości: - *obsługiwane* jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
eventsource	integer	Typ zdarzeń, które będzie obsługiwać akcja. Zobacz właściwość `source` zdarzenia, aby uzyskać listę obsługiwanych typów zdarzeń. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
name	string	Nazwa akcji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
status	integer	Określa, czy akcja jest włączona, czy wyłączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.
pause_symptoms	integer	Określa, czy wstrzymać eskalację, jeśli zdarzenie jest zdarzeniem symptomatycznym. Możliwe wartości: 0 - Nie wstrzymuj eskalacji dla problemów symptomatycznych; 1 - *(domyślnie)* Wstrzymaj eskalację dla problemów symptomatycznych. Zachowanie właściwości: - *obsługiwane* jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”
pause_suppressed	integer	Określa, czy wstrzymać eskalację w okresach konserwacji. Możliwe wartości: 0 - Nie wstrzymuj eskalacji; 1 - *(domyślnie)* Wstrzymaj eskalację. Zachowanie właściwości: - *obsługiwane* jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”
notify_if_canceled	integer	Określa, czy wysyłać powiadomienie po anulowaniu eskalacji. Możliwe wartości: 0 - Nie powiadamiaj po anulowaniu eskalacji; 1 - *(domyślnie)* Powiadamiaj po anulowaniu eskalacji. Zachowanie właściwości: - *obsługiwane* jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”

Operacja akcji

Obiekt operacji akcji definiuje operację, która zostanie wykonana, gdy akcja zostanie uruchomiona. Ma następujące właściwości.

Właściwość	Typ	Opis
operationtype	integer	Typ operacji. Możliwe wartości: 0 - wyślij wiadomość; 1 - skrypt globalny; 2 - dodaj host; 3 - usuń host; 4 - dodaj do grupy hostów; 5 - usuń z grupy hostów; 6 - podłącz szablon; 7 - odłącz szablon; 8 - włącz host; 9 - wyłącz host; 10 - ustaw tryb inwentaryzacji hosta; 13 - dodaj tagi hosta; 14 - usuń tagi hosta. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”: 0 - ”wyślij wiadomość”; 1 - ”skrypt globalny”. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie wewnętrzne”: 0 - ”wyślij wiadomość”. Zachowanie właściwości: - *wymagane*
esc_period	string	Czas trwania kroku eskalacji w sekundach. Musi być większy niż 60 sekund. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Jeśli ustawiono 0 lub 0s, zostanie użyty domyślny okres eskalacji akcji. Domyślnie: 0s. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
esc_step_from	integer	Krok, od którego ma się rozpocząć eskalacja. Domyślnie: 1. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
esc_step_to	integer	Krok, na którym ma się zakończyć eskalacja. Domyślnie: 1. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
evaltype	integer	Metoda oceny warunku operacji. Możliwe wartości: 0 - *(domyślnie)* And/Or; 1 - And; 2 - Or.

Właściwość	Typ	Opis
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „skrypt globalny”
opcommand_grp	array	Grupy hostów, na których mają być uruchamiane skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_hst` nie jest ustawione
opcommand_hst	array	Host, na którym mają być uruchamiane skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_grp` nie jest ustawione
opconditions	array	Warunki operacji używane dla akcji wyzwalaczy. Obiekt warunku operacji jest opisany szczegółowo poniżej.
opgroup	array	Grupy hostów, do których mają zostać dodane hosty. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „dodaj do grupy hostów” lub „usuń z grupy hostów”
opmessage	object	Obiekt zawierający dane o wiadomości wysyłanej przez operację. Obiekt wiadomości operacji jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość”
opmessage_grp	array	Grupy użytkowników, do których mają być wysyłane wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których mają być wysyłane wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_grp` nie jest ustawione
optemplate	array	Szablony do podłączenia do hostów. Szablony muszą mieć zdefiniowaną właściwość `templateid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „podłącz szablon” lub „odłącz szablon”

Właściwość	Typ	Opis
opinventory	object	Tryb inwentaryzacji ustawiany dla hosta. Inwentaryzacja musi mieć zdefiniowaną właściwość `inventory_mode`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „ustaw tryb inwentaryzacji hosta”
optag	array	Tagi hosta do dodania lub usunięcia. Tagi muszą mieć zdefiniowaną właściwość `tag`. Właściwość `value` jest opcjonalna. Zachowanie właściwości: - *obsługiwane*, jeśli `operationtype` jest ustawione na „dodaj tagi hosta” lub „usuń tagi hosta”.

Wiadomość operacji akcji

Obiekt wiadomości operacji zawiera dane o wiadomości, która zostanie wysłana przez operację. Ma następujące właściwości.

Właściwość	Type	Opis
default_msg	integer	Czy użyć domyślnego tekstu i tematu wiadomości akcji. Możliwe wartości: 0 - użyj danych z operacji; 1 - *(domyślnie)* użyj danych z typu nośnika.
mediatypeid	ID	ID typu nośnika, który zostanie użyty do wysłania wiadomości. Zachowanie właściwości: - *obsługiwane*, jeśli `operationtype` obiektu Action operation object, Action recovery operation object lub Action update operation object jest ustawione na „send message”, albo jeśli `operationtype` obiektu Action update operation object jest ustawione na „notify all involved”
message	string	Tekst wiadomości operacji. Zachowanie właściwości: - *obsługiwane*, jeśli `default_msg` jest ustawione na „use the data from the operation”
subject	string	Temat wiadomości operacji. Zachowanie właściwości: - *obsługiwane*, jeśli `default_msg` jest ustawione na „use the data from the operation”

Warunek operacji akcji

Obiekt warunku operacji akcji definiuje warunek, który musi zostać spełniony, aby wykonać bieżącą operację. Ma następujące właściwości.

Property	Type	Description
conditiontype	integer	Typ warunku. Możliwe wartości: 14 - zdarzenie potwierdzone. Zachowanie właściwości: - *wymagane*

Property	Type	Description
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*
operator	integer	Operator warunku. Możliwe wartości: 0 - *(domyślnie)* =

Dla każdego typu warunku operacji obsługiwane są następujące operatory i wartości.

Condition	Condition name	Supported operators	Expected value
14	Zdarzenie potwierdzone	=	Czy zdarzenie jest potwierdzone. Możliwe wartości: 0 - niepotwierdzone; 1 - potwierdzone.

Operacja odzyskiwania akcji

Obiekt operacji odzyskiwania akcji definiuje operację, która zostanie wykonana po rozwiązaniu problemu. Operacje odzyskiwania są możliwe **tylko** dla akcji wyzwalacza, wewnętrznych i usług. Ma następujące właściwości.

Właściwość	Typ	Opis
operationtype	integer	Typ operacji. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”: 0 - wyślij wiadomość; 1 - skrypt globalny; 11 - powiadom wszystkich zaangażowanych. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie wewnętrzne”: 0 - wyślij wiadomość; 11 - powiadom wszystkich zaangażowanych. Zachowanie właściwości: - *wymagane*
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „skrypt globalny”
opcommand_grp	array	Grupy hostów, na których mają być uruchamiane skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, a `operationtype` jest ustawione na „skrypt globalny”, i `opcommand_hst` nie jest ustawione

Właściwość	Typ	Opis
opcommand_hst	array	Host, na którym mają być uruchamiane skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, a `operationtype` jest ustawione na „skrypt globalny”, i `opcommand_grp` nie jest ustawione
opmessage	object	Obiekt zawierający dane o wiadomości wysyłanej przez operację odzyskiwania. Obiekt wiadomości operacji jest opisany szczegółowo powyżej. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość”
opmessage_grp	array	Grupy użytkowników, do których mają być wysyłane wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość” i `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których mają być wysyłane wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość” i `opmessage_grp` nie jest ustawione

Operacja aktualizacji akcji

Obiekt operacji aktualizacji akcji definiuje operację, która zostanie wykonana po zaktualizowaniu problemu (dodaniu komentarza, potwierdzeniu, zmianie ważności lub ręcznym zamknięciu). Operacje aktualizacji są możliwe **tylko** dla akcji wyzwalaczy i usług. Ma następujące właściwości.

Właściwość	Type	Opis
operationtype	integer	Typ operacji. Możliwe wartości: 0 - wyślij wiadomość; 1 - skrypt globalny; 12 - powiadom wszystkich zaangażowanych. Zachowanie właściwości: - *required*
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „skrypt globalny”

Właściwość	Type	Opis
opcommand_grp	array	Grupy hostów, na których mają być uruchamiane skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *required* jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_hst` nie jest ustawione
opcommand_hst	array	Host, na którym mają być uruchamiane skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *required* jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_grp` nie jest ustawione
opmessage	object	Obiekt zawierający dane o wiadomości wysyłanej przez operację aktualizacji. Obiekt wiadomości operacji jest opisany szczegółowo powyżej.
opmessage_grp	array	Grupy użytkowników, do których mają być wysyłane wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których mają być wysyłane wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_grp` nie jest ustawione

Filtr akcji

Obiekt filtra akcji definiuje zestaw warunków, które muszą zostać spełnione, aby wykonać skonfigurowane operacje akcji. Ma następujące właściwości.

Właściwość	Typ	Opis
conditions	array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*
evaltype	integer	Metoda obliczania warunku filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do obliczania warunków filtra. Wyrażenie zawiera identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do obliczania warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „custom expression”

Warunek filtra akcji

Obiekt warunku filtra akcji definiuje określony warunek, który musi zostać sprawdzony przed uruchomieniem operacji akcji.

Właściwość	Typ	Opis
conditiontype	integer	Typ warunku. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”: 0 - grupa hostów; 1 - host; 2 - wyzwalacz; 3 - nazwa zdarzenia; 4 - ważność wyzwalacza; 6 - okres czasu; 13 - szablon hosta; 16 - problem jest stłumiony; 25 - tag zdarzenia; 26 - wartość tagu zdarzenia. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez regułę wykrywania”: 7 - IP hosta; 8 - typ wykrytej usługi; 9 - port wykrytej usługi; 10 - status wykrywania; 11 - czas trwania działania lub niedostępności; 12 - odebrana wartość; 18 - reguła wykrywania; 19 - sprawdzenie wykrywania; 20 - proxy; 21 - obiekt wykrywania. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez aktywną autorejestrację agent”: 20 - proxy; 22 - nazwa hosta; 24 - metadane hosta. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie wewnętrzne”: 0 - grupa hostów; 1 - host; 13 - szablon hosta; 23 - typ zdarzenia; 25 - tag zdarzenia; 26 - wartość tagu zdarzenia. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przy aktualizacji statusu usługi”: 25 - tag zdarzenia; 26 - wartość tagu zdarzenia; 27 - usługa; 28 - nazwa usługi. Zachowanie właściwości: - *wymagane*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
value2	string	Dodatkowa wartość do porównania. Zachowanie właściwości: - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji wyzwalaczy, a typ warunku (patrz niżej) to „26” - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie wewnętrzne”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji wewnętrznych, a typ warunku (patrz niżej) to „26” - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przy aktualizacji statusu usługi”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji usług, a typ warunku (patrz niżej) to „26”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać tylko wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu ich pobrania.
operator	integer	Operator warunku. Możliwe wartości: 0 - *(domyślnie)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 4 - w; 5 - jest większe lub równe; 6 - jest mniejsze lub równe; 7 - nie w; 8 - pasuje; 9 - nie pasuje; 10 - Tak; 11 - Nie.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod action.get i action.create.

Dla każdego typu warunku obsługiwane są następujące operatory i wartości.

Warunek	Nazwa warunku	Obsługiwane operatory	Oczekiwana wartość
0	Grupa hostów	równa się, nie równa się	ID grupy hostów.
1	Host	równa się, nie równa się	ID hosta.
2	Wyzwalacz	równa się, nie równa się	ID wyzwalacza.
3	Nazwa zdarzenia	zawiera, nie zawiera	Nazwa zdarzenia.
4	Ważność wyzwalacza	równa się, nie równa się, jest większe lub równe, jest mniejsze lub równe	Ważność wyzwalacza. Listę obsługiwanych ważności wyzwalacza można znaleźć we właściwości trigger `severity` property.

Warunek	Nazwa warunku	Obsługiwane operatory	Oczekiwana wartość
5	Wartość wyzwalacza	równa się	Wartość wyzwalacza. Listę obsługiwanych wartości wyzwalacza można znaleźć we właściwości trigger `value` property.
6	Okres czasu	w, nie w	Czas wystąpienia zdarzenia jako okres czasu.
7	IP hosta	równa się, nie równa się	Jeden lub kilka zakresów IP do sprawdzenia, oddzielonych przecinkami. Więcej informacji o obsługiwanych formatach zakresów IP można znaleźć w sekcji konfiguracja wykrywania sieci.
8	Typ wykrytej usługi	równa się, nie równa się	Typ wykrytej usługi. Typ usługi odpowiada typowi sprawdzenia wykrywania użytego do wykrycia usługi. Listę obsługiwanych typów można znaleźć we właściwości discovery check `type` property.
9	Port wykrytej usługi	równa się, nie równa się	Jeden lub kilka zakresów portów, oddzielonych przecinkami.
10	Status wykrywania	równa się	Status wykrytego obiektu. Możliwe wartości: 0 - host lub usługa działa; 1 - host lub usługa nie działa; 2 - host lub usługa wykryta; 3 - host lub usługa utracona.
11	Czas trwania działania lub niedostępności	jest większe lub równe, jest mniejsze lub równe	Czas w sekundach określający, jak długo wykryty obiekt znajduje się w bieżącym stanie.
12	Odebrane wartości	równa się, nie równa się, jest większe lub równe, jest mniejsze lub równe, zawiera, nie zawiera	Wartość zwrócona podczas wykonywania sprawdzenia wykrywania Zabbix agent, SNMPv1, SNMPv2 lub SNMPv3.
13	Szablon hosta	równa się, nie równa się	ID podłączonego szablonu.
16	Problem jest stłumiony	Tak, Nie	Wartość nie jest wymagana: użycie operatora „Tak” oznacza, że problem musi być stłumiony, „Nie” — niestłumiony.
18	Reguła wykrywania	równa się, nie równa się	ID reguły wykrywania.
19	Sprawdzenie wykrywania	równa się, nie równa się	ID sprawdzenia wykrywania.
20	Proxy	równa się, nie równa się	ID proxy.
21	Obiekt wykrywania	równa się	Typ obiektu, który wywołał zdarzenie wykrywania. Możliwe wartości: 1 - wykryty host; 2 - wykryta usługa.
22	Nazwa hosta	zawiera, nie zawiera, pasuje, nie pasuje	Nazwa hosta. W warunkach autorejestracji dla operatorów *pasuje* i *nie pasuje* obsługiwane jest użycie wyrażenia regularnego.
23	Typ zdarzenia	równa się	Określone zdarzenie wewnętrzne. Możliwe wartości: 0 - pozycja w stanie „nieobsługiwane”; 1 - pozycja w stanie „normalne”; 2 - reguła LLD w stanie „nieobsługiwane”; 3 - reguła LLD w stanie „normalne”; 4 - wyzwalacz w stanie „nieznane”; 5 - wyzwalacz w stanie „normalne”.

Warunek	Nazwa warunku	Obsługiwane operatory	Oczekiwana wartość
24	Metadane hosta	zawiera, nie zawiera, pasuje, nie pasuje	Metadane automatycznie zarejestrowanego hosta. Użycie wyrażenia regularnego jest obsługiwane dla operatorów *pasuje* i *nie pasuje*.
25	Tag	równa się, nie równa się, zawiera, nie zawiera	Tag zdarzenia.
26	Wartość tagu	równa się, nie równa się, zawiera, nie zawiera	Wartość tagu zdarzenia.
27	Usługa	równa się, nie równa się	ID usługi.
28	Nazwa usługi	równa się, nie równa się	Nazwa usługi.

action.create

Opis

`object action.create(object/array actions)`

Ta metoda umożliwia tworzenie nowych akcji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Akcje do utworzenia.

Oprócz standardowych właściwości akcji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra akcji dla akcji.
operations	array	Operacje akcji do utworzenia dla akcji.
recovery_operations	array	Operacje odzyskiwania akcji do utworzenia dla akcji.
update_operations	array	Operacje aktualizacji akcji do utworzenia dla akcji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID utworzonych akcji we właściwości `actionids`. Kolejność zwracanych identyfikatorów ID jest zgodna z kolejnością przekazanych akcji.

Przykłady

Utwórz akcję wyzwalacza

Utwórz akcję wyzwalacza, która rozpocznie się, gdy wyzwalacz (zawierający słowo „memory” w nazwie) z hosta „10084” przejdzie do stanu PROBLEM. Akcja będzie miała 4 skonfigurowane operacje. Pierwsza, wykonywana natychmiast operacja wyśle wiadomość do wszystkich użytkowników z grupy użytkowników „7” za pośrednictwem typu mediów „1”. Jeśli zdarzenie nie zostanie rozwiązane w ciągu 30 minut, druga operacja uruchomi skrypt „5” (skrypt o zakresie „Action operation”) na wszystkich hostach z grupy „2”. Jeśli zdarzenie zostanie rozwiązane, operacja odzyskiwania powiadomi wszystkich użytkowników, którzy otrzymali jakiekolwiek wiadomości dotyczące problemu. Jeśli zdarzenie zostanie zaktualizowane, operacja potwierdzenia/aktualizacji powiadomi (z niestandardowym tematem i wiadomością) wszystkich użytkowników, którzy otrzymali jakiekolwiek wiadomości dotyczące problemu.

Żądanie:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "action.create",
    "params": {
        "name": "Trigger action",
        "eventsource": 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": [
                    {
                        "usrgrpid": "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"
                }
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "17"
        ]
    },
    "id": 1
}
```

Utwórz akcję wykrywania

Utwórz akcję wykrywania, która połączy szablon „10001” z wykrytymi hostami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Discovery action",
        "eventsource": 1,
        "filter": {
            "evaltype": 0,
            "conditions": [
                {
                    "conditiontype": 21,
                    "operator": 0,
                    "value": "1"
                },
                {
                    "conditiontype": 10,
                    "operator": 0,
                    "value": "2"
                }
            ]
        },
        "operations": [
            {
                "operationtype": 6,
                "optemplate": [
                    {
                        "templateid": "10001"
                    }
                ]
            }
        ]
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Używanie niestandardowego filtra wyrażeń

Utwórz akcję wyzwalacza, która używa niestandardowego wyrażenia - „A and (B or C)” - do oceniania warunków akcji. Gdy wyzwalacz o ważności wyższej lub równej „Warning” z hosta „10084” lub hosta „10106” przejdzie do stanu PROBLEM, akcja wyśle wiadomość do wszystkich użytkowników z grupy użytkowników „7” za pośrednictwem typu nośnika „1”. Identyfikatory formuły „A”, „B” i „C” zostały wybrane arbitralnie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Trigger action",
        "eventsource": 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": [
                    {
                        "usrgrpid": "7"
                    }
```

```
                ],
                "opmessage": {
                    "default_msg": 1,
                    "mediatypeid": "1"
                }
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Utwórz regułę autorejestracji agent

Utwórz akcję autorejestracji, która dodaje host do grupy hostów „2”, gdy nazwa hosta zawiera „SRV” lub metadane zawierają „AlmaLinux”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Register Linux servers",
        "eventsource": "2",
        "filter": {
            "evaltype": "2",
            "conditions": [
                {
                    "conditiontype": "22",
                    "operator": "2",
                    "value": "SRV"
                },
                {
                    "conditiontype": "24",
                    "operator": "2",
                    "value": "AlmaLinux"
                }
            ]
        },
        "operations": [
            {
                "operationtype": "4",
                "opgroup": [
                    {
                        "groupid": "2"
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            19
        ]
    },
    "id": 1
}
```

Utwórz regułę autorejestracji agent z tagami hosta

Utwórz akcję autorejestracji, która dodaje host do grupy hostów „2” oraz dodaje dwa tagi hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Register Linux servers with tags",
        "eventsource": "2",
        "operations": [
            {
                "operationtype": "4",
                "opgroup": [
                    {
                        "groupid": "2"
                    }
                ]
            },
            {
                "operationtype": "13",
                "optag": [
                    {
                        "tag": "location",
                        "value": "office"
                    },
                    {
                        "tag": "city",
                        "value": "Riga"
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            20
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr akcji
- Operacja akcji
- Skrypt

Źródło

CAction::create() w *ui/include/classes/api/services/CAction.php*.

action.delete

Opis

`object action.delete(array actionIds)`

Ta metoda pozwala na usuwanie akcji.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkownika po więcej informacji.

Parametry

`(array)` identyfikatory ID akcji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID usuniętych akcji we właściwości `actionids`.

Przykłady

Usuń wiele akcji

Usuń dwie akcje.

Zapytanie:

```
{
"jsonrpc": "2.0",
"method": "action.delete",
"params": [
"17",
"18"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"actionids": [
"17",
"18"
]
},
"id": 1
}
```

Źródło

CAction::delete() w *ui/include/classes/api/services/CAction.php*.

action.get

Description

`integer/array action.get(object parameters)`

Metoda pozwala na pobranie akcji zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkownika po więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
actionids	ID/array	Zwraca tylko akcje o podanych ID.
groupids	ID/array	Zwraca tylko akcje, które używają podanych grup hostów w warunkach akcji.
hostids	ID/array	Zwraca tylko akcje, które używają podanych hostów w warunkach akcji.
triggerids	ID/array	Zwraca tylko akcje, które używają podanych wyzwalaczy w warunkach akcji.
mediatypeids	ID/array	Zwraca tylko akcje, które używają podanych typów mediów do wysyłania wiadomości.
usrgrpids	ID/array	Zwraca tylko akcje, które są skonfigurowane do wysyłania wiadomości do podanych grup użytkowników.
userids	ID/array	Zwraca tylko akcje, które są skonfigurowane do wysyłania wiadomości do podanych użytkowników.
scriptids	ID/array	Zwraca tylko akcje, które są skonfigurowane do uruchamiania podanych skryptów.
selectFilter	query	Zwraca właściwość `filter` z filtrem warunków akcji.
selectOperations	query	Zwraca właściwość `operations` z operacjami akcji.
selectRecoveryOperations	query	Zwraca właściwość `recovery_operations` z operacjami odzyskiwania akcji.
selectUpdateOperations	query	Zwraca właściwość `update_operations` z operacjami aktualizacji akcji.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `actionid`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobierz akcje wyzwalaczy

Pobierz wszystkie skonfigurowane akcje wyzwalaczy wraz z warunkami akcji i operacjami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
```

```
        "output": "extend",
        "selectOperations": "extend",
        "selectRecoveryOperations": "extend",
        "selectUpdateOperations": "extend",
        "selectFilter": "extend",
        "filter": {
            "eventsource": 0
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "3",
            "name": "Report problems to Zabbix administrators",
            "eventsource": "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": "Wyzwalacz: {TRIGGER.NAME}\r\nStatus wyzwalacza: {TRIGGER.STATUS}\r\nWa
                        "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": "Zaktualizowano: {TRIGGER.NAME}",
                        "message": "{USER.FULLNAME} zaktualizował problem o {EVENT.UPDATE.DATE} {EVENT.UPD
                        "mediatypeid": "1"
                    },
                    "opmessage_grp": [
                        {
                            "usrgrpid": "7"
                        }
                    ],
                    "opmessage_usr": []
                },
                {
                    "operationid": "33",
                    "operationtype": "1",
                    "evaltype": "0",
                    "opcommand": {
                        "scriptid": "3"
                    },
                    "opcommand_hst": [
                        {
                            "hostid": "10084"
                        }
                    ],
                    "opcommand_grp": []
                }
            ]
        }
    ],
    "id": 1
}
```


Pobierz akcje wykrywania

Pobierz wszystkie skonfigurowane akcje wykrywania wraz z warunkami i operacjami akcji. Filtr używa typu oceny „and”, więc właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "action.get",
  "params": {
    "output": "extend",
    "selectOperations": "extend",
    "selectFilter": "extend",
    "filter": {
      "eventsource": 1
    }
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "actionid": "2",
      "name": "Auto discovery. Linux servers.",
      "eventsource": "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
}
```


Zobacz także

- Filtr akcji
- Operacja akcji

Źródło

CAction::get() w *ui/include/classes/api/services/CAction.php*.

action.update

Description

`object action.update(object/array actions)`

Ta metoda pozwala na aktualizowanie istniejących akcji.

> **Note:**
> Ta metoda jest dostępna wyłącznie użytkownikom typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli. Zobacz Role użytkownika po więcej informacji.

Parametry

`(object/array)` Właściwości akcji do zaktualizowania.

Właściwość `actionid` musi być zdefiniowana dla każdej akcji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości akcji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra akcji, który zastąpi bieżący filtr.
operations	array	Operacje akcji, które zastąpią istniejące operacje.

Parametr	Typ	Opis
recovery_operations	array	Operacje odzyskiwania akcji, które zastąpią istniejące operacje odzyskiwania. Zachowanie parametru: - *obsługiwane*, jeśli `eventsource` obiektu akcji jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
update_operations	array	Operacje aktualizacji akcji, które zastąpią istniejące operacje aktualizacji. Zachowanie parametru: - *obsługiwane*, jeśli `eventsource` obiektu akcji jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID zaktualizowanych akcji we właściwości `actionids`.

Przykłady

Wyłącz akcję

Wyłącz akcję, to znaczy ustaw jej status na „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.update",
    "params": {
        "actionid": "2",
        "status": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr akcji
- Operacja akcji

Źródło

CAction::update() in *ui/include/classes/api/services/CAction.php*.

Alarm

Ta klasa jest przeznaczona do pracy z alarmami.

Odwołania do obiektu:

- Alert

Dostępne metody:

- alert.get - pobieranie alarmów

Alert obiektu

Następujące obiekty są bezpośrednio powiązane z API `alert`.

Alert

> **Note:**
> Alerty są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt alertu zawiera informacje o tym, czy określone operacje akcji zostały wykonane pomyślnie. Ma następujące właściwości.

Właściwość	Typ	Opis
alertid	ID	ID alertu.
actionid	ID	ID akcji, która wygenerowała alert.
alerttype	integer	Typ alertu. Możliwe wartości: 0 - wiadomość; 1 - zdalne polecenie.
clock	timestamp	Czas wygenerowania alertu.
error	string	Tekst błędu, jeśli wystąpiły problemy z wysłaniem wiadomości lub uruchomieniem polecenia.
esc_step	integer	Krok eskalacji akcji, podczas którego został wygenerowany alert.
eventid	ID	ID zdarzenia, które wyzwoliło akcję.
mediatypeid	ID	ID typu nośnika, który został użyty do wysłania wiadomości.
message	text	Treść wiadomości. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
retries	integer	Liczba prób wysłania wiadomości przez Zabbix.
sendto	string	Adres, nazwa użytkownika lub inny identyfikator odbiorcy. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
status	integer	Status wskazujący, czy operacja akcji została wykonana pomyślnie. Możliwe wartości, jeśli `alerttype` jest ustawione na ”message”: 0 - wiadomość nie została wysłana; 1 - wiadomość została wysłana; 2 - niepowodzenie po określonej liczbie prób; 3 - nowy alert nie został jeszcze przetworzony przez menedżera alertów. Możliwe wartości, jeśli `alerttype` jest ustawione na ”remote command”: 0 - polecenie nie zostało uruchomione; 1 - polecenie zostało uruchomione; 2 - podjęto próbę uruchomienia polecenia na Zabbix agent, ale był niedostępny.
subject	string	Temat wiadomości. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
userid	ID	ID użytkownika, do którego wysłano wiadomość.
p_eventid	ID	ID zdarzenia problemu, które wygenerowało alert.
acknowledgeid	ID	ID potwierdzenia, które wygenerowało alert.

alert.get

Opis

`integer/array alert.get(object parameters)`

Metoda pozwala na pobieranie alertów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
alertids	ID/array	Zwraca tylko alerty o podanych identyfikatorach.
actionids	ID/array	Zwraca tylko alerty wygenerowane przez podane akcje.
eventids	ID/array	Zwraca tylko alerty wygenerowane przez podane zdarzenia.
groupids	ID/array	Zwraca tylko alerty wygenerowane przez obiekty z podanych grup hostów.
hostids	ID/array	Zwraca tylko alerty wygenerowane przez obiekty z podanych hostów.
mediatypeids	ID/array	Zwraca tylko alerty wiadomości, w których użyto podanych typów mediów.
objectids	ID/array	Zwraca tylko alerty wygenerowane przez podane obiekty
userids	ID/array	Zwraca tylko alerty wiadomości, które zostały wysłane do podanych użytkowników.
eventobject	integer	Zwraca tylko alerty wygenerowane przez zdarzenia powiązane z obiektami podanego typu. Listę obsługiwanych typów obiektów można znaleźć w `object` zdarzenia. Domyślnie: 0 - wyzwalacz.
eventsource	integer	Zwraca tylko alerty wygenerowane przez zdarzenia podanego typu. Listę obsługiwanych typów zdarzeń można znaleźć w `source` zdarzenia. Domyślnie: 0 - zdarzenia wyzwalacza.
time_from	timestamp	Zwraca tylko alerty wygenerowane po podanym czasie.
time_till	timestamp	Zwraca tylko alerty wygenerowane przed podanym czasem.
selectHosts	query	Zwraca właściwość `hosts` z danymi hostów, które wyzwoliły operację akcji.
selectMediatypes	query	Zwraca właściwość `mediatypes` z tablicą typów mediów użytych dla alertu wiadomości. Ograniczenia zależne od typu użytkownika można znaleźć w `mediatype.get`.
selectUsers	query	Zwraca właściwość `users` z tablicą użytkowników, do których była adresowana wiadomość. Ograniczenia zależne od typu użytkownika można znaleźć w `user.get`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `alertid`, `clock`, `eventid`, `mediatypeid`, `sendto`, `status`.
countOutput	boolean	Parametry te są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	

Parametr	Typ	Opis
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów albo;
- liczbę zwracanych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobierz alerty według ID akcji

Pobierz wszystkie alerty wygenerowane przez akcję ”3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "alert.get",
    "params": {
        "output": "extend",
        "actionids": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Host
- Typ nośnika
- Użytkownik

Źródło

CAlert::get() in *ui/include/classes/api/services/CAlert.php*.

API info

Klasa jest przeznaczona do pobierania metainformacji o interfejsie API.

Dostępne metody:

- apiinfo.version - pobieranie wersji API Zabbix

apiinfo.version

Opis

`string apiinfo.version(array)`

Ta metoda pozwala na pobranie wersji API Zabbix.

> **Attention:**
> Ta metoda jest dostępna tylko dla nieuwierzytelnionych użytkowników.

Parametry

`(array)` Metoda akceptuje pustą tablicę.

Zwracane wartości

`(string)` Zwraca wersję API Zabbix.

> **Note:**
> Zaczynając od wersji Zabbix 2.0.4, wersja API jest taka sama jak wersja Zabbix.

Przykłady

Pobieranie wersji API

Pobierz wersję API Zabbixa.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "apiinfo.version",
    "params": [],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "7.4.0",
    "id": 1
}
```

Źródło

CAPIInfo::version() in *ui/include/classes/api/services/CAPIInfo.php*.

Autorejestracja

Ta klasa jest przeznaczona do pracy z autorejestracją.

Odniesienia do obiektów:

- Autoregistration

Dostępne metody:

- autoregistration.get - pobierz autorejestrację
- autoregistration.update - aktualizuj autorejestrację

Obiekt Autorejestracja

Następujące obiekty są bezpośrednio powiązane z API `autoregistration`.

Autorejestracja

Obiekt autorejestracji ma następujące właściwości.

Właściwość	Type	Opis
tls_accept	integer	Typ dozwolonych połączeń przychodzących dla autorejestracji. Możliwe wartości: 1 - zezwalaj na połączenia nieszyfrowane; 2 - zezwalaj na TLS z PSK; 3 - zezwalaj zarówno na połączenia nieszyfrowane, jak i TLS z PSK.
tls_psk_identity	string	Tożsamość PSK; musi być sparowana tylko z jednym PSK (w obrębie autoregistration, hosts i proxies). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK należy użyć. Zachowanie właściwości: *- tylko do zapisu*
tls_psk	string	Klucz współdzielony (PSK); musi mieć co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: *- tylko do zapisu*

autoregistration.get

Opis

`object autoregistration.get(object parameters)`

Ta metoda pozwala na pobranie obiektu rejestracji automatycznej według podanych parametrów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkownika po więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

`(object)` Zwraca obiekt rejestracji automatycznej.

Przykład

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "autoregistration.get",
  "params": {
    "output": "extend"
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tls_accept": "3"
  },
  "id": 1
}
```

Źródło

CAutoregistration::get() w *ui/include/classes/api/services/CAutoregistration.php*.

autoregistration.update

Opis

`object autoregistration.update(object autoregistration)`

Ta metoda pozwala na aktualizację obiektu autorejestracji

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkownika po więcej informacji.

Parametry

`(object)` Właściwości rejestracji automatycznej do aktualizacji.

Zwracane wartości

`(boolean )` Zwraca wartość logiczną true jako wynik po udanej aktualizacji.

Przykłady

Zapytanie:

```
{
    "jsonrpc": "2.0",
    "method": "autoregistration.update",
    "params": {
        "tls_accept": "3",
        "tls_psk_identity": "PSK 001",
        "tls_psk": "11111595725ac58dd977beef14b97461a7c1045b9a1c923453302c5473193478"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Źródło

CAutoregistration::update() w *ui/include/classes/api/services/CAutoregistration.php*.

Connector

Ta klasa jest przeznaczona do pracy z connectorami.

Odwołania do obiektów:

- Connector
- Filtr tagów

Dostępne metody:

- connector.create - tworzenie nowych connectorów
- connector.delete - usuwanie connectorów
- connector.get - pobieranie connectorów
- connector.update - aktualizacja connectorów

Obiekt łącznika

Następujące obiekty są bezpośrednio powiązane z API `connector`.

Connector

Obiekt connector ma następujące właściwości.

Property	Type	Description
connectorid	ID	ID connectora. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa connectora. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
url	string	Adres URL punktu końcowego, czyli adres URL odbiorcy. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
protocol	integer	Protokół komunikacji. Możliwe wartości: 0 - *(domyślnie)* Zabbix Streaming Protocol v1.0.
data_type	integer	Typ danych. Możliwe wartości: 0 - *(domyślnie)* Wartości pozycji; 1 - Zdarzenia.

Property	Type	Description
item_value_type	integer	Suma typów wartości pozycji, które mają zostać wysłane. Możliwe wartości: 1 - Liczbowe (float); 2 - Znakowe; 4 - Log; 8 - Liczbowe (bez znaku); 16 - Tekst; 32 - Binarny. Domyślnie: 31 - Wszystkie typy pozycji (z wyjątkiem binarnych). Zachowanie właściwości: - *obsługiwane* jeśli `data_type` jest ustawione na ”Item values”.
max_records	integer	Maksymalna liczba zdarzeń lub pozycji, które można wysłać w jednej wiadomości. Możliwe wartości: 0-2147483647 (maksymalna wartość 32-bitowej liczby całkowitej ze znakiem). Domyślnie: 0 - Bez ograniczeń.
max_senders	integer	Liczba procesów nadawczych uruchamianych dla tego connectora. Możliwe wartości: 1-100. Domyślnie: 1.
max_attempts	integer	Liczba prób. Możliwe wartości: 1-5. Domyślnie: 1.
attempt_interval	string	Interwał między próbami ponowienia. Akceptuje sekundy. Możliwe wartości: 0s-10s. Domyślnie: 5s. Zachowanie właściwości: - *obsługiwane* jeśli `max_attempts` jest większe niż 1.
timeout	string	Limit czasu. Obsługiwane są sufiksy czasu (np. 30s, 1m). Obsługiwane są makra użytkownika. Możliwe wartości: 1s-60s. Domyślnie: 5s.
http_proxy	string	Łańcuch połączenia proxy HTTP(S) podany jako *[protocol]://[username[:password]@]proxy.example.c* Obsługiwane są makra użytkownika.

Property	Type	Description
authtype	integer	Metoda uwierzytelniania HTTP. Możliwe wartości: 0 - *(domyślnie)* Brak; 1 - Basic; 2 - NTLM; 3 - Kerberos; 4 - Digest; 5 - Bearer.
username	string	Nazwa użytkownika. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *obsługiwane* jeśli `authtype` jest ustawione na ”Basic”, ”NTLM”, ”Kerberos” lub ”Digest”
password	string	Hasło. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *obsługiwane* jeśli `authtype` jest ustawione na ”Basic”, ”NTLM”, ”Kerberos” lub ”Digest”
token	string	Token Bearer. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagane* jeśli `authtype` jest ustawione na ”Bearer”
verify_peer	integer	Czy sprawdzać, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - Nie sprawdzaj; 1 - *(domyślnie)* Sprawdzaj.
verify_host	integer	Czy sprawdzać, czy nazwa hosta dla połączenia odpowiada nazwie w certyfikacie hosta. Możliwe wartości: 0 - Nie sprawdzaj; 1 - *(domyślnie)* Sprawdzaj.
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Obsługiwane są makra użytkownika.
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Obsługiwane są makra użytkownika.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane są makra użytkownika.
description	text	Opis connectora.
status	integer	Czy connector jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
tags_evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* I/Lub; 2 - Lub.

Filtr tagów

Filtr tagów umożliwia eksportowanie tylko pasujących wartości pozycji lub zdarzeń. Jeśli nie jest ustawiony, wyeksportowane zostanie wszystko. Obiekt filtra tagów ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek operatora. Możliwe wartości: 0 - *(domyślnie)* Równa się; 1 - Nie równa się; 2 - Zawiera; 3 - Nie zawiera; 12 - Istnieje; 13 - Nie istnieje.
value	string	Wartość tagu.

connector.create

Opis

`object connector.create(object/array connectors)`

Ta metoda umożliwia tworzenie nowych obiektów konektorów.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Obiekty connectorów do utworzenia.

Oprócz standardowych właściwości connectora metoda akceptuje następujące parametry.

Parametr	Type	Opis
tags	array	Filtr tagów connectora.

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID utworzonych konektorów we właściwości `connectorids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych konektorów.

Przykłady

Tworzenie konektora

Utwórz konektor, aby eksportować zdarzenia wyzwalaczy z filtrem tagów. Uwierzytelnianie HTTP zostanie wykonane przy użyciu tokenu Bearer.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.create",
    "params": [
        {
            "name": "Export of events",
            "data_type": 1,
            "url": "{$DATA_EXPORT_URL}",
            "authtype": 5,
```

```
            "token": "{$DATA_EXPORT_BEARER_TOKEN}",
            "tags": [
                {
                    "tag": "service",
                    "operator": 0,
                    "value": "mysqld"
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorid": [
            "3"
        ]
    },
    "id": 1
}
```

Źródło

CConnector::create() w *ui/include/classes/api/services/CConnector.php*.

connector.delete

Opis

`object connector.delete(array connectorids)`

Ta metoda pozwala na usuwanie wpisów konektora.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(tablica)` Identyfikatory konektorów do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych konektorów pod właściwością `connectorids`.

Przykłady

Usuwanie wielu konektorów

Usuń dwa wpisy konektora.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "connector.delete",
"params": [
3,
5
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"connectorids": [
"3",
"5"
]
},
"id": 1
}
```

Źródło

CConnector::delete() w *ui/include/classes/api/services/CConnector.php*.

connector.get

Opis

`integer/array connector.get(object parameters)`

Metoda pozwala na pobranie obiektów konektora zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
connectorids	ID/array	Zwraca tylko connectory o podanych identyfikatorach.
selectTags	query	Zwraca właściwość `tags` z filtrem tagów connectora. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `connectorid`, `name`, `data_type`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich konektorów

Pobierz wszystkie dane o wszystkich konektorach i ich właściwościach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.get",
    "params": {
        "output": "extend",
        "selectTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "connectorid": "1",
            "name": "Export of item values",
            "protocol": "0",
            "data_type": "0",
            "url": "{$DATA_EXPORT_VALUES_URL}",
            "item_value_type": "31",
            "authtype": "4",
            "username": "{$DATA_EXPORT_VALUES_USERNAME}",
            "password": "{$DATA_EXPORT_VALUES_PASSWORD}",
            "token": "",
            "max_records": "0",
            "max_senders": "4",
            "max_attempts": "2",
            "attempt_interval": "10s",
            "timeout": "10s",
            "http_proxy": "{$DATA_EXPORT_VALUES_PROXY}",
            "verify_peer": "1",
            "verify_host": "1",
            "ssl_cert_file": "{$DATA_EXPORT_VALUES_SSL_CERT_FILE}",
            "ssl_key_file": "{$DATA_EXPORT_VALUES_SSL_KEY_FILE}",
            "ssl_key_password": "",
            "description": "",
            "status": "1",
            "tags_evaltype": "0",
            "tags": [
                {
                    "tag": "component",
                    "operator": "0",
                    "value": "memory"
                }
            ]
        },
        {
            "connectorid": "2",
            "name": "Export of events",
            "protocol": "0",
            "data_type": "1",
            "url": "{$DATA_EXPORT_EVENTS_URL}",
            "item_value_type": "31",
            "authtype": "5",
            "username": "",
            "password": "",
            "token": "{$DATA_EXPORT_EVENTS_BEARER_TOKEN}",
            "max_records": "0",
            "max_senders": "2",
```

```
            "max_attempts": "1",
            "attempt_interval": "5s",
            "timeout": "5s",
            "http_proxy": "",
            "verify_peer": "1",
            "verify_host": "1",
            "ssl_cert_file": "",
            "ssl_key_file": "",
            "ssl_key_password": "",
            "description": "",
            "status": "1",
            "tags_evaltype": "0",
            "tags": [
                {
                    "tag": "scope",
                    "operator": "0",
                    "value": "performance"
                }
            ]
        }
    ],
    "id": 1
}
```

Źródło

CConnector:get() w *ui/include/classes/api/services/CConnector.php*.

connector.update

Opis

`object connector.update(object/array connectors)`

Ta metoda pozwala zaktualizować istniejące konektory.

noteclassic Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników. :::

Parametry

`(object/array)` Właściwości connectora do zaktualizowania.

Właściwość `connectorid` musi być zdefiniowana dla każdego connectora, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości connectora metoda akceptuje następujące parametry.

Parametr	Typ	Opis
tags	array	Filtr tagów connectora, który zastąpi bieżący filtr tagów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych konektorów we właściwości `connectorids`.

Przykłady

Zmiana typu uwierzytelniania HTTP

Zmień typ uwierzytelniania HTTP na Bearer dla connectora o ID „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": {
        "connectorid": 3,
```

```
        "authtype": 5,
        "token": "{$DATA_EXPORT_BEARER_TOKEN}"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3"
        ]
    },
    "id": 1
}
```

Aktualizowanie filtra tagów

Zmień filtr tagów dla konektora o ID „5”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": [
        {
            "connectorid": 5,
            "tags_evaltype": 2,
            "tags": [
                {
                    "tag": "service",
                    "operator": 0,
                    "value": "mysqld"
                },
                {
                    "tag": "error",
                    "operator": 12,
                    "value": ""
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CConnector::update() w *ui/include/classes/api/services/CConnector.php*.

Dziennik audytu

Ta klasa została zaprojektowana do pracy z dziennikiem audytu.

Odwołania do obiektów:

- Dziennik audytu

Dostępne metody:

- auditlog.get - pobieranie rekordów dziennika audytu

Obiekt dziennika audytu

Następujące obiekty są bezpośrednio powiązane z API `auditlog`.

Dziennik audytu

Obiekt dziennika audytu zawiera informacje o działaniach użytkownika. Ma następujące właściwości.

Właściwość	Typ	Opis
auditid	ID	ID wpisu dziennika audytu. Generowane przy użyciu algorytmu CUID.
userid	ID	Userid autora wpisu dziennika audytu.
username	string	Nazwa użytkownika autora wpisu dziennika audytu.
clock	timestamp	Znacznik czasu utworzenia wpisu dziennika audytu.
ip	string	Adres IP autora wpisu dziennika audytu.
action	integer	Działanie wpisu dziennika audytu. Możliwe wartości: 0 - Dodanie; 1 - Aktualizacja; 2 - Usunięcie; 4 - Wylogowanie; 7 - Wykonanie; 8 - Logowanie; 9 - Nieudane logowanie; 10 - Czyszczenie historii; 11 - Odświeżenie konfiguracji; 12 - Push.

Właściwość	Typ	Opis
resourcetype	integer	Typ zasobu wpisu dziennika audytu. Możliwe wartości: 0 - Użytkownik; 3 - Typ nośnika; 4 - Host; 5 - Akcja; 6 - Wykres; 11 - Grupa użytkowników; 13 - Wyzwalacz; 14 - Grupa hostów; 15 - Pozycja; 16 - Obraz; 17 - Mapa wartości; 18 - Usługa; 19 - Mapa; 22 - Scenariusz WWW; 23 - Reguła wykrywania; 25 - Skrypt; 26 - Proxy; 27 - Okres utrzymania; 28 - Wyrażenie regularne; 29 - Makro; 30 - Szablon; 31 - Prototyp wyzwalacza; 32 - Mapowanie ikon; 33 - Pulpit; 34 - Korelacja zdarzeń; 35 - Prototyp wykresu; 36 - Prototyp pozycji; 37 - Prototyp hosta; 38 - Autorejestracja; 39 - Moduł; 40 - Ustawienia; 41 - Housekeeping; 42 - Uwierzytelnianie; 43 - Pulpit szablonu; 44 - Rola użytkownika; 45 - Token API; 46 - Zaplanowany raport; 47 - Węzeł wysokiej dostępności; 48 - SLA; 49 - Katalog użytkowników; 50 - Grupa szablonów; 51 - Łącznik; 52 - Reguła LLD; 53 - Historia; 54 - Uwierzytelnianie wieloskładnikowe; 55 - Grupa proxy; 56 - Prototyp reguły LLD.
resourceid	ID	Identyfikator zasobu wpisu dziennika audytu.
resource_cuid	ID	Unikalny identyfikator zasobu wpisu dziennika audytu wygenerowany przy użyciu algorytmu CUID.
resourcename	string	Czytelna dla człowieka nazwa zasobu wpisu dziennika audytu.
recordsetid	ID	ID zestawu rekordów wpisu dziennika audytu. Rekordy dziennika audytu utworzone podczas tej samej operacji będą miały to samo ID zestawu rekordów. Generowane przy użyciu algorytmu CUID.

Właściwość	Typ	Opis
details	text	Szczegóły wpisu dziennika audytu. Szczegóły są przechowywane jako obiekt JSON, gdzie każda nazwa właściwości jest ścieżką do właściwości lub zagnieżdżonego obiektu, w którym wystąpiła zmiana, a każda wartość zawiera dane (w formacie tablicy) o zmianie tej właściwości lub zagnieżdżonego obiektu. Możliwe formaty wartości: ["add"] - Zagnieżdżony obiekt został dodany; ["add", "<value>"] - Właściwość dodanego obiektu jest równa <value>; ["update"] - Zagnieżdżony obiekt został zaktualizowany; ["update", "<new value>", "<old value>"] - Właściwość zaktualizowanego obiektu została zmieniona z <old value> na <new value>; ["delete"] - Zagnieżdżony obiekt został usunięty.

auditlog.get

Opis

`integer/array auditlog.get(object parameters)`

Ta metoda umożliwia pobranie wpisów dziennika audytu zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna wyłącznie dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
auditids	ID/array	Zwraca tylko dziennik audytu o podanych ID.
userids	ID/array	Zwraca tylko dziennik audytu utworzony przez podanych użytkowników.
time_from	timestamp	Zwraca tylko wpisy dziennika audytu, które zostały utworzone po podanym czasie lub o podanym czasie.
time_till	timestamp	Zwraca tylko wpisy dziennika audytu, które zostały utworzone przed podanym czasem lub o podanym czasie.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `auditid`, `userid`, `clock`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;

- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie dziennika audytu

Pobierz dwa najnowsze wpisy dziennika audytu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "auditlog.get",
    "params": {
        "output": "extend",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 2
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "auditid": "cksstgfam0001yhdcc41y20q2",
            "userid": "1",
            "username": "Admin",
            "clock": "1629975715",
            "ip": "127.0.0.1",
            "action": "1",
            "resourcetype": "0",
            "resourceid": "0",
            "resource_cuid": "0",
            "resourcename": "Jim",
            "recordsetid": "cksstgfal0000yhdcso67ondl",
            "details": "{\"user.name\":[\"update\",\"Jim\",\"\"],\"user.medias[37]\":[\"add\"],\"user.medi
        },
        {
            "auditid": "ckssofl0p0001yhdcqxclsg8r",
            "userid": "1",
            "username": "Admin",
            "clock": "1629967278",
            "ip": "127.0.0.1",
            "action": "0",
            "resourcetype": "0",
            "resourceid": "20",
            "resource_cuid": "0",
            "resourcename": "John",
            "recordsetid": "ckssofl0p0000yhdcpxyo1jgo",
            "details": "{\"user.username\":[\"add\",\"John\"], \"user.userid:\":[\"add\",\"20\"],\"user.us
        }
    ],
    "id": 1
}
```

Zobacz także

- Audit log object

Źródło

CAuditLog::get() in *ui/include/classes/api/services/CAuditLog.php*.

Grupa hostów

Ta klasa jest przeznaczona do pracy z grupami hostów.

Odwołania do obiektów:

- Grupa hostów

Dostępne metody:

- hostgroup.create - tworzy nowe grupy hostów
- hostgroup.delete - usuwa grupy hostów
- hostgroup.get - pobiera grupy hostów
- hostgroup.massadd - dodaje powiązane obiekty do grup hostów
- hostgroup.massremove - usuwa powiązane obiekty z grup hostów
- hostgroup.massupdate - zastępuje lub usuwa powiązane obiekty z grup hostów
- hostgroup.propagate - propaguje uprawnienia i filtry tagów do podgrup grup hostów
- hostgroup.update - aktualizuje grupy hostów

Obiekt grupy hostów

Następujące obiekty są bezpośrednio powiązane z API `hostgroup`.

Grupa hostów

Obiekt grupy hostów ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy hostów. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie grupy hostów. Możliwe wartości: 0 - zwykła grupa hostów; 4 - wykryta grupa hostów. Zachowanie właściwości: - *tylko do odczytu*
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych grup hostów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.

hostgroup.create

Opis

`object hostgroup.create(object/array hostGroups)`

Ta metoda pozwala na tworzenie nowych grup hostów.

Note:
Ta metoda jest dostępna tylko dla użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Grupy hostów do utworzenia.

Metoda akceptuje grupy hostów z standardowymi właściwościami grup hostów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup hostów we właściwości `groupids`. Kolejność zwróconych ID odpowiada kolejności przekazanych grup hostów.

Przykłady

Tworzenie grupy hostów

Utwórz grupę hostów o nazwie „Serwery Linux”.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostgroup.create",
"params": {
"name": "Serwery Linux"
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"groupids": [
"107819"
]
},
"id": 1
}
```

Źródło

CHostGroup::create() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.delete

Opis

`object hostgroup.delete(array hostGroupIds)`

Ta metoda umożliwia usuwanie grup hostów.

Grupa hostów nie może zostać usunięta, jeśli:

- zawiera hosty, które należą wyłącznie do tej grupy;
- jest oznaczona jako wewnętrzna;
- jest używana przez prototyp hosta;
- jest używana w skrypcie globalnym;
- jest używana w warunku korelacji.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID grup hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup hostów w właściwości `groupids`.

Przykłady

Usuwanie wielu grup hostów

Usuń dwie grupy hostów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostgroup.delete",
"params": [
"107824",
"107825"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"groupids": [
"107824",
"107825"
]
},
"id": 1
}
```

Źródło

CHostGroup::delete() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.get

Opis

`integer/array hostgroup.get(object parameters)`

Ta metoda umożliwia pobieranie grup hostów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwraca tylko grupy hostów zawierające hosty z podanymi wykresami.
groupids	ID/array	Zwraca tylko grupy hostów o podanych identyfikatorach grup hostów.
hostids	ID/array	Zwraca tylko grupy hostów zawierające podane hosty.
maintenanceids	ID/array	Zwraca tylko grupy hostów, na które wpływają podane okresy konserwacji.
triggerids	ID/array	Zwraca tylko grupy hostów zawierające hosty z podanymi wyzwalaczami.
with_graphs	boolean	Zwraca tylko grupy hostów zawierające hosty z wykresami.
with_graph_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami wykresów.
with_hosts	boolean	Zwraca tylko grupy hostów zawierające hosty.
with_httptests	boolean	Zwraca tylko grupy hostów zawierające hosty z testami WWW. Nadpisuje parametr `with_monitored_httptests`.

Parametr	Typ	Opis
with_items	boolean	Zwraca tylko grupy hostów zawierające hosty z pozycjami. Nadpisuje parametry `with_monitored_items` i `with_simple_graph_items`.
with_item_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami pozycji, które są włączone do tworzenia i mają numeryczny typ informacji.
with_monitored_httptests	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi testami WWW.
with_monitored_hosts	boolean	Zwraca tylko grupy hostów zawierające monitorowane hosty.
with_monitored_items	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi pozycjami. Nadpisuje parametr `with_simple_graph_items`.
with_monitored_triggers	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi wyzwalaczami. Wszystkie pozycje używane w wyzwalaczu również muszą być włączone.
with_simple_graph_items	boolean	Zwraca tylko grupy hostów zawierające hosty z numerycznymi pozycjami.
with_triggers	boolean	Zwraca tylko grupy hostów zawierające hosty z wyzwalaczami. Nadpisuje parametr `with_monitored_triggers`.
selectDiscoveryRules	query	Zwraca właściwość `discoveryRules` z regułami LLD, które wykryły grupę hostów.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z obiektami wykrywania grup hostów. Każdy obiekt wykrywania grupy hostów jest prototypem grupy hostów powiązanym z wykrytą grupą hostów i ma następujące właściwości: `parent_group_prototypeid` - `(ID)` identyfikator prototypu grupy hostów, z którego wykryto grupę hostów; `name` - `(string)` nazwa prototypu grupy hostów; `ts_delete` - `(timestamp)` czas, po którym grupa hostów, która nie jest już wykrywana, zostanie usunięta; `status` - `(int)` status wykrywania grupy hostów: 0 - *(domyślnie)* grupa hostów jest wykryta, 1 - grupa hostów nie jest już wykrywana.
selectHostPrototypes	query	Zwraca właściwość `hostPrototypes` z prototypami hostów, które wykryły tę grupę hostów.
selectHosts	query	Zwraca właściwość `hosts` z hostami należącymi do grupy hostów. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki będą sortowane według `host`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	

Parametr	Typ	Opis
startSearch	boolean	
selectGroupDiscoveries	query	Zwraca właściwość `groupDiscoveries` z obiektami wykrywania grup hostów. Każdy obiekt wykrywania grupy hostów jest prototypem grupy hostów powiązanym z wykrytą grupą hostów. To zapytanie jest **przestarzałe**; zamiast niego użyj `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch grupach hostów o nazwach „Zabbix servers” i „Linux servers”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Zabbix servers",
                "Linux servers"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "2",
            "name": "Linux servers",
            "flags": "0",
            "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
        },
        {
            "groupid": "4",
            "name": "Zabbix servers",
            "flags": "0",
            "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
    ],
    "id": 1
}
```

Zobacz także

- Host

Źródło

CHostGroup::get() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massadd

Opis

`object hostgroup.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych grup hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup hostów do zaktualizowania oraz obiekty do dodania do wszystkich grup hostów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
groups	object/array	Grupy hostów, które mają zostać zaktualizowane. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametrów: - *wymagane*
hosts	object/array	Hosty do dodania do wszystkich grup hostów. Hosty muszą mieć zdefiniowaną tylko właściwość `hostid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Dodawanie hostów do grup hostów

Dodaj dwa hosty do grup hostów o identyfikatorach 5 i 6.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "5"
            },
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30001"
            }
        ]
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host

Źródło

CHostGroup::massAdd() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massremove

Opis

`object hostgroup.massremove(object parameters)`

Ta metoda pozwala na usuwanie powiązanych obiektów z wielu grup hostów.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry zawierające identyfikatory grup hostów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Type	Opis
groupids	ID/array	Identyfikatory grup hostów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
hostids	ID/array	Identyfikatory hostów, które mają zostać usunięte ze wszystkich grup hostów.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Usuwanie hostów z grup hostów

Usuń dwa hosty z podanych grup hostów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
```

```
        ],
        "hostids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CHostGroup::massRemove() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massupdate

Opis

`object hostgroup.massupdate(object parameters)`

Ta metoda umożliwia zastąpienie hostów i szablonów określonymi elementami w wielu grupach hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup hostów do aktualizacji oraz obiekty, które powinny zostać zaktualizowane.

Parameter	Type	Description
groups	object/array	Grupy hostów do zaktualizowania. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *required*
hosts	object/array	Hosty do zastąpienia bieżących hostów w podanych grupach hostów. Wszystkie pozostałe hosty, z wyjątkiem wymienionych, zostaną wykluczone z grup hostów. Wykryte hosty nie zostaną objęte zmianą. Hosty muszą mieć zdefiniowaną tylko właściwość `hostid`. Zachowanie parametru: - *required*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów w właściwości `groupids`.

Przykłady

Zastępowanie hostów w grupie hostów

Zastąp wszystkie hosty w grupie hostów hostem wskazanym poniżej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massupdate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

Zobacz także

- hostgroup.update
- hostgroup.massadd
- Host

Źródło

CHostGroup::massUpdate() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.propagate

Opis

`object hostgroup.propagate(object parameters)`

Ta metoda umożliwia zastosowanie uprawnień i filtrów tagów do wszystkich podgrup grupy hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

|Parametr|Type|Opis|

|--|--|------|

|groups|object/array|Grupy hostów do propagowania.

Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`.

Zach
parametrów:
- *wymagane*|

|permissions|boolean|Ustaw na ”true”, aby propagować uprawnienia.

Zachowanie parametrów:
- *wymagane*, jeśli `tag_filters` nie jest ustawione|

|tag_filters|boolean|Ustaw na ”true”, aby propagować filtry tagów.

Zachowanie parametrów:
- *wymagane*, jeśli `permissions` nie jest ustawione|

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory propagowanych grup hostów

we właściwości `groupids`.

Przykłady

Propagowanie uprawnień grupy hostów i filtrów tagów do jej podgrup.

Propagowanie uprawnień grupy hostów i filtrów tagów do jej podgrup.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "permissions": true,
        "tag_filters": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

Zobacz także

- hostgroup.update
- hostgroup.massadd
- Host

Źródło

CHostGroup::propagate() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.update

Opis

`object hostgroup.update(object/array hostGroups)`

Ta metoda umożliwia aktualizację istniejących grup hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą być cofnięte w ustawieniach ról użytkownika. Zobacz Role użytkownika aby uzyskać więcej informacji.

Parametry

`(object/array)` Właściwości grupy hostów, które mają zostać zaktualizowane.

Właściwość `groupid` musi być zdefiniowana dla każdej grupy hostów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Zmiana nazwy grupy hostów

Zmień nazwę grupy hostów na „Linux hosts”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": {
        "groupid": "7",
        "name": "Linux hosts"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Źródło

CHostGroup::update() w *ui/include/classes/api/services/CHostGroup.php*.

Grupa proxy

Ta klasa jest przeznaczona do pracy z grupami proxy.

Odwołania do obiektów:

- Grupa proxy

Dostępne metody:

- proxygroup.create - utwórz nowe grupy proxy
- proxygroup.delete - usuń grupy proxy
- proxygroup.get - pobierz grupy proxy
- proxygroup.update - zaktualizuj grupy proxy

Obiekt grupy proxy

Następujące obiekty są bezpośrednio powiązane z API `proxygroup`.

Grupa proxy

Obiekt grupy proxy ma następujące właściwości.

Właściwość	Typ	Opis
proxy_groupid	ID	ID grupy proxy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy proxy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	text	Opis grupy proxy.
failover_delay	string	Okres, w którym proxy w grupie proxy musi komunikować się z serwer Zabbix, aby było uznawane za online. Obsługiwane są sufiksy czasu, np. 30s, 1m. Obsługiwane są makra użytkownika. Możliwe wartości: 10s-15m. Domyślnie: 1m.
min_online	string	Minimalna liczba proxy online wymagana do utrzymania grupy proxy w stanie online. Obsługiwane są makra użytkownika. Zakres możliwych wartości: 1-1000. Domyślnie: 1.
state	integer	Stan grupy proxy. Możliwe wartości: 0 - Nieznany; 1 - Offline; 2 - Odzyskiwanie; 3 - Online; 4 - Pogarszanie. Zachowanie właściwości: - *tylko do odczytu*

proxygroup.create

Opis

`object proxygroup.create(object/array proxyGroups)`

Ta metoda umożliwia tworzenie nowych grup proxy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w Rolach użytkowników.

Parametry

`(object/array)` Grupy proxy do utworzenia.

Metoda akceptuje grupy proxy ze standardowymi właściwościami grup proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup proxy we właściwości `proxy_groupids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup proxy.

Przykłady

Utwórz grupę proxy

Utwórz grupę proxy z niestandardowymi ustawieniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.create",
    "params": {
        "name": "Proxy group",
        "failover_delay": "5m",
        "min_online": "10"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::create() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.delete

Opis

`object proxygroup.delete(array proxyGroupIds)`

Ta metoda umożliwia usuwanie grup proxy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(array)` ID grup proxy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup proxy we właściwości `proxy_groupids`.

Przykłady

Usuń wiele grup proxy

Usuń dwie grupy proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.delete",
    "params": [
        "5",
        "10"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5",
            "10"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::delete() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.get

Opis

`integer/array proxygroup.get(object parameters)`

Ta metoda umożliwia pobieranie grup proxy zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
proxy_groupids	ID/array	Zwraca tylko grupy proxy o podanych identyfikatorach.
proxyids	ID/array	Zwraca tylko grupy proxy zawierające podane proxy.
selectProxies	query	Zwraca właściwość `proxies` z proxy należącymi do grupy proxy. Obsługuje `count`.

Parametr	Typ	Opis
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `proxy_groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich grup proxy

Pobierz wszystkie skonfigurowane grupy proxy wraz z proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.get",
    "params": {
        "output": "extend",
        "selectProxies": ["proxyid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxy_groupid": "1",
            "name": "Proxy group 1",
            "failover_delay": "1m",
            "min_online": "3",
            "description": "",
            "state": "1",
            "proxies": [
                {
                    "proxyid": "1",
                    "name": "proxy 1"
                },
                {
                    "proxyid": "2",
                    "name": "proxy 2"
                }
            ]
        },
        {
```

```
            "proxy_groupid": "2",
            "name": "Proxy group 2",
            "failover_delay": "10m",
            "min_online": "3",
            "description": "",
            "state": "3",
            "proxies": [
                {
                    "proxyid": "3",
                    "name": "proxy 3"
                },
                {
                    "proxyid": "4",
                    "name": "proxy 4"
                },
                {
                    "proxyid": "5",
                    "name": "proxy 5"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Proxy

Źródło

CProxyGroup::get() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.update

Opis

`object proxygroup.update(object/array proxyGroups)`

Ta metoda umożliwia aktualizację istniejących grup proxy.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Właściwości grupy proxy do zaktualizowania.

Właściwość `proxy_groupid` musi być zdefiniowana dla każdej grupy proxy, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje grupy proxy ze standardowymi właściwościami grupy proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup proxy we właściwości `proxy_groupids`.

Przykłady

Zmień minimalną liczbę proxy online

Zmień minimalną liczbę proxy online wymaganą, aby grupa była online.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.update",
```

```
    "params": {
        "proxy_groupid": "5",
        "min_online": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::update() w *ui/include/classes/api/services/CProxyGroup.php*.

Grupa szablonów

Ta klasa została zaprojektowana do pracy z grupami szablonów.

Odwołania do obiektów:

- Grupa szablonów

Dostępne metody:

- templategroup.create - utwórz nowe grupy szablonów
- templategroup.delete - usuń grupy szablonów
- templategroup.get - pobierz grupy szablonów
- templategroup.massadd - dodaj powiązane obiekty do grup szablonów
- templategroup.massremove - usuń powiązane obiekty z grup szablonów
- templategroup.massupdate - zastąp lub usuń powiązane obiekty z grup szablonów
- templategroup.propagate - propaguj uprawnienia do podgrup grup szablonów
- templategroup.update - aktualizuj grupy szablonów

Obiekt grupy szablonów

Następujące obiekty są bezpośrednio powiązane z API `templategroup`.

Grupa szablonów

Obiekt grupy szablonów ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy szablonów. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy szablonów. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych grup szablonów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.

templategroup.create

Opis

`object templategroup.create(object/array templateGroups)`

Ta metoda umożliwia tworzenie nowych grup szablonów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy szablonów do utworzenia. Metoda akceptuje grupy szablonów ze standardowymi właściwościami grup szablonów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup szablonów we właściwości `groupids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup szablonów.

Przykłady

Tworzenie grupy szablonów

Utwórz grupę szablonów o nazwie "Templates/Databases".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.create",
    "params": {
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107820"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::create() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.delete

Opis

`object templategroup.delete(array templateGroupIds)`

Ta metoda umożliwia usuwanie grup szablonów.

Grupa szablonów nie może zostać usunięta, jeśli zawiera szablony należące wyłącznie do tej grupy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID grup szablonów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup szablonów we właściwości `groupids`.

Przykłady

Usuwanie wielu grup szablonów

Usuń dwie grupy szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.delete",
    "params": [
        "107814",
        "107815"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107814",
            "107815"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::delete() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.get

Opis

`integer/array templategroup.get(object parameters)`

Ta metoda umożliwia pobieranie grup szablonów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwracaj tylko grupy szablonów zawierające szablony z podanymi wykresami.
groupids	ID/array	Zwracaj tylko grupy szablonów o podanych identyfikatorach grup szablonów.
templateids	ID/array	Zwracaj tylko grupy szablonów zawierające podane szablony.
triggerids	ID/array	Zwracaj tylko grupy szablonów zawierające szablony z podanymi wyzwalaczami.

Parametr	Typ	Opis
with_graphs	boolean	Zwracaj tylko grupy szablonów zawierające szablony z wykresami.
with_graph_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami wykresów.
with_httptests	boolean	Zwracaj tylko grupy szablonów zawierające szablony z testami WWW.
with_items	boolean	Zwracaj tylko grupy szablonów zawierające szablony z pozycjami. Nadpisuje parametry `with_simple_graph_items`.
with_item_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami pozycji, które są włączone do tworzenia i mają numeryczny typ informacji.
with_simple_graph_items	boolean	Zwracaj tylko grupy szablonów zawierające szablony z numerycznymi pozycjami.
with_templates	boolean	Zwracaj tylko grupy szablonów zawierające szablony.
with_triggers	boolean	Zwracaj tylko grupy szablonów zawierające szablony z wyzwalaczami.
selectTemplates	query	Zwracaj właściwość `templates` zawierającą szablony należące do grupy szablonów. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectTemplates` - wyniki będą sortowane według `template`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch grupach szablonów o nazwach „Templates/Databases” oraz „Templates/Modules”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
```

```
                "Templates/Databases",
                "Templates/Modules"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "13",
            "name": "Templates/Databases",
            "uuid": "748ad4d098d447d492bb935c907f652f"
        },
        {
            "groupid": "8",
            "name": "Templates/Modules",
            "uuid": "57b7ae836ca64446ba2c296389c009b7"
        }
    ],
    "id": 1
}
```

Zobacz także

- Szablon

Źródło

CTemplateGroup::get() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massadd

Opis

`object templategroup.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych grup szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup szablonów do zaktualizowania oraz obiekty do dodania do wszystkich grup szablonów.

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy szablonów, które mają zostać zaktualizowane. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametrów: - *wymagane*

Parametr	Typ	Opis
templates	object/array	Szablony do dodania do wszystkich grup szablonów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Dodawanie szablonów do grup szablonów

Dodaj dwa szablony do grup szablonów o identyfikatorach 12 i 13.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "12"
            },
            {
                "groupid": "13"
            }
        ],
        "templates": [
            {
                "templateid": "10486"
            },
            {
                "templateid": "10487"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "12",
            "13"
        ]
    },
    "id": 1
}
```

Zobacz także

- Szablon

Źródło

CTemplateGroup::massAdd() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massremove

Opis

`object templategroup.massremove(object parameters)`

Ta metoda umożliwia usunięcie powiązanych obiektów z wielu grup szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup szablonów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Typ	Opis
groupids	ID/array	Identyfikatory grup szablonów, które mają zostać zaktualizowane. Zachowanie parametrów: - *wymagane*
templateids	ID/array	Identyfikatory szablonów, które mają zostać usunięte ze wszystkich grup szablonów. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Usuwanie szablonów z grup szablonów

Usuń dwa szablony z podanych grup szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "templateids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
```

```
    "id": 1
}
```

Źródło

CTemplateGroup::massRemove() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massupdate

Opis

`object templategroup.massupdate(object parameters)`

Ta metoda umożliwia zastąpienie szablonów określonymi szablonami w wielu grupach szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup szablonów do aktualizacji oraz obiekty, które powinny zostać zaktualizowane.

Parameter	Type	Description
groups	object/array	Grupy szablonów do zaktualizowania. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametru: - *required*
templates	object/array	Szablony do zastąpienia bieżącego szablonu w podanych grupach szablonów. Wszystkie pozostałe szablony, z wyjątkiem wymienionych, zostaną wykluczone z grup szablonów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`. Zachowanie parametru: - *required*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów w właściwości `groupids`.

Przykłady

Zastępowanie szablonów w grupie szablonów

Zastąp wszystkie szablony w grupie szablonów wskazanymi szablonami.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massupdate",
    "params": {
        "groups": [
            {
                "groupid": "8"
            }
```

```
        ],
        "templates": [
            {
                "templateid": "40050"
            }
        ]
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "8",
        ]
    },
    "id": 1
}
```

Zobacz też

- templategroup.update
- templategroup.massadd
- Szablon

Źródło

CTemplateGroup::massUpdate() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.propagate

Opis

`object templategroup.propagate(object parameters)`

Ta metoda umożliwia zastosowanie uprawnień do wszystkich podgrup grup szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

|Parametr|Type|Opis|

|--|--|------|

|groups|object/array|Grupy szablonów do propagowania.

Grupy szablonów muszą mieć zdefiniowaną tylko właściwość `groupid`.

Zachowanie parametrów:
- *wymagane*|

|permissions|boolean|Ustaw `true`, jeśli należy propagować uprawnienia.

Zachowanie parametrów:
- *wymagane*|

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory propagowanych grup szablonów

we właściwości `groupids`.

Przykłady

Propagowanie uprawnień grupy szablonów do jej podgrup.

Propaguj uprawnienia grupy szablonów do jej podgrup.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "15"
            }
        ],
        "permissions": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "15",
        ]
    },
    "id": 1
}
```

Zobacz też

- templategroup.update
- templategroup.massadd
- Szablon

Źródło

CTemplateGroup::propagate() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.update

Opis

`object templategroup.update(object/array templateGroups)`

Ta metoda umożliwia aktualizację istniejących grup szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object/array)` Właściwości grupy szablonów, które mają zostać zaktualizowane.

Właściwość `groupid` musi być zdefiniowana dla każdej grupy szablonów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Zmiana nazwy grupy szablonów

Zmień nazwę grupy szablonów na ”Templates/Databases”

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.update",
```

```
    "params": {
        "groupid": "7",
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::update() w *ui/include/classes/api/services/CTemplateGroup.php*.

Grupa użytkowników

Ta klasa została zaprojektowana do pracy z grupami użytkowników.

Odwołania do obiektów:

- Grupa użytkowników
- Uprawnienie
- Uprawnienie oparte na tagach

Dostępne metody:

- usergroup.create - utwórz nowe grupy użytkowników
- usergroup.delete - usuń grupy użytkowników
- usergroup.get - pobierz grupy użytkowników
- usergroup.update - zaktualizuj grupy użytkowników

Obiekt grupy użytkowników

Następujące obiekty są bezpośrednio powiązane z API `usergroup`.

Grupa użytkowników

Obiekt grupy użytkowników ma następujące właściwości.

Właściwość	Typ	Opis
usrgrpid	ID	ID grupy użytkowników. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy użytkowników. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
debug_mode	integer	Określa, czy tryb debugowania jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* wyłączony; 1 - włączony.

Właściwość	Typ	Opis
gui_access	integer	Metoda uwierzytelniania we frontend użytkowników w grupie. Możliwe wartości: 0 - *(domyślnie)* użyj domyślnej metody uwierzytelniania systemu; 1 - użyj uwierzytelniania wewnętrznego; 2 - użyj uwierzytelniania LDAP; 3 - wyłącz dostęp do frontend.
mfa_status	integer	Określa, czy MFA jest włączone, czy wyłączone dla użytkowników w grupie. Możliwe wartości: 0 - wyłączone (dla wszystkich skonfigurowanych metod MFA); 1 - włączone (dla wszystkich skonfigurowanych metod MFA).
mfaid	ID	Metoda MFA używana dla użytkowników w grupie. Zachowanie właściwości: - *obsługiwane* dla użytkowników typu *Super admin* - *obsługiwane*, jeśli `mfa_status` obiektu Authentication object jest ustawione na „Enabled”
users_status	integer	Określa, czy grupa użytkowników jest włączona, czy wyłączona. Dla użytkowników deprovisioned grupa użytkowników nie może być włączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.
userdirectoryid	ID	ID katalogu użytkowników używanego do uwierzytelniania. Zachowanie właściwości: - *obsługiwane* dla użytkowników typu *Super admin* - *obsługiwane*, jeśli `gui_access` jest ustawione na „use the system default authentication method” lub „use LDAP authentication”

Uprawnienie

Obiekt uprawnienia ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID grupy hostów lub grupy szablonów, do której ma zostać dodane uprawnienie. Zachowanie właściwości: - *wymagane*
permission	integer	Poziom dostępu do grupy hostów lub grupy szablonów. Możliwe wartości: 0 - dostęp zabroniony; 2 - dostęp tylko do odczytu; 3 - dostęp do odczytu i zapisu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Uprawnienie oparte na tagach

Obiekt uprawnienia opartego na tagach ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów, do której należy dodać uprawnienie. Zachowanie właściwości: - *wymagane*
tag	string	Nazwa tagu.
value	string	Wartość tagu.

usergroup.create

Opis

`object usergroup.create(object/array userGroups)`

Ta metoda umożliwia tworzenie nowych grup użytkowników.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy użytkowników do utworzenia.

Oprócz standardowych właściwości grupy użytkowników, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
hostgroup_rights	object/array	Uprawnienia grup hostów do przypisania do grupy użytkowników.
templategroup_rights	object/array	Uprawnienia grup szablonów do przypisania do grupy użytkowników.
tag_filters	array	Uprawnienia oparte na tagach do przypisania do grupy użytkowników.
users	object/array	Użytkownicy do dodania do grupy użytkowników. Użytkownik musi mieć zdefiniowaną wyłącznie właściwość `userid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup użytkowników we właściwości `usrgrpids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup użytkowników.

Przykłady

Tworzenie grupy użytkowników

Utwórz grupę użytkowników *Operation managers* z zabronionym dostępem do grupy hostów „2” i dodaj do niej użytkownika.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.create",
    "params": {
        "name": "Operation managers",
        "hostgroup_rights": {
            "id": "2",
            "permission": 0
        },
        "users": [
            {
                "userid": "12"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uprawnienie

Źródło

CUserGroup::create() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.delete

Opis

`object usergroup.delete(array userGroupIds)`

Ta metoda umożliwia usunięcie grup użytkownika.

Attention:
Deprovisioned users group (grupa użytkowników określona dla `disabled_usrgrpid` w Authentication) nie może zostać usunięta.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID grup użytkowników do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup użytkowników we właściwości `usrgrpids`.

Przykłady

Usuwanie wielu grup użytkowników

Usuń dwie grupy użytkowników.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usergroup.delete",
"params": [
"20",
"21"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"usrgrpids": [
"20",
```

```
"21"
]
},
"id": 1
}
```

Źródło

CUserGroup::delete() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.get

Opis

`integer/array usergroup.get(object parameters)`

Ta metoda umożliwia pobieranie grup użytkowników zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
mfaids	ID/array	Zwraca tylko grupy użytkowników z podanymi metodami MFA. Zachowanie parametrów: - *obsługiwane* dla użytkowników typu *Super admin*
mfa_status	integer	Zwraca tylko grupy użytkowników z podanym statusem MFA. Listę obsługiwanych statusów można znaleźć na stronie grupy użytkowników.
status	integer	Zwraca tylko grupy użytkowników z podanym statusem. Listę obsługiwanych statusów można znaleźć na stronie grupy użytkowników.
userids	ID/array	Zwraca tylko grupy użytkowników zawierające podanych użytkowników.
usrgrpids	ID/array	Zwraca tylko grupy użytkowników o podanych identyfikatorach.
selectTagFilters	query	Zwraca uprawnienia grup użytkowników oparte na tagach we właściwości `tag_filters`.
selectUsers	query	Zwraca użytkowników z grupy użytkowników we właściwości `users`. Zobacz `user.get`, aby sprawdzić ograniczenia zależne od typu użytkownika.
selectHostGroupRights	query	Zwraca uprawnienia grup użytkowników do grup hostów we właściwości `hostgroup_rights`. Listę poziomów dostępu do grup hostów można znaleźć na stronie grupy użytkowników.
selectTemplateGroupRights	query	Zwraca uprawnienia grup użytkowników do grup szablonów we właściwości `templategroup_rights`. Listę poziomów dostępu do grup szablonów można znaleźć na stronie grupy użytkowników.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania.

Parametr	Typ	Opis
output	query	Właściwości obiektu grupy użytkowników, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: `usrgrpid`, `name`, `gui_access`, `users_status`, `debug_mode`, `mfa_status`. Domyślnie: `extend`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `usrgrpid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie włączonych grup użytkowników

Pobierz wszystkie włączone grupy użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.get",
    "params": {
        "output": "extend",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "usrgrpid": "7",
            "name": "Zabbix administrators",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "1",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "8",
            "name": "Guests",
```

```
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "11",
            "name": "Enabled debug mode",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "1",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "12",
            "name": "No access to the frontend",
            "gui_access": "2",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "14",
            "name": "Read only",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        },
        {
            "usrgrpid": "18",
            "name": "Deny",
            "gui_access": "0",
            "users_status": "0",
            "debug_mode": "0",
            "userdirectoryid": "0",
            "mfa_status": "0",
            "mfaid": "0"
        }
    ],
    "id": 1
}
```


Zobacz także

- Użytkownik

Źródło

CUserGroup::get() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.update

Opis

`object usergroup.update(object/array userGroups)`

Ta metoda umożliwia aktualizację istniejących grup użytkowników.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości grup użytkowników do zaktualizowania.

Właściwość `usrgrpid` musi być zdefiniowana dla każdej grupy użytkowników, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości grup użytkowników, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
hostgroup_rights	object/array	Uprawnienia do grup hostów, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
templategroup_rights	object/array	Uprawnienia do grup szablonów, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
tag_filters	array	Uprawnienia oparte na tagach, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
users	object/array	Użytkownicy, którzy zastąpią bieżących użytkowników przypisanych do grupy użytkowników. Użytkownik musi mieć zdefiniowaną tylko właściwość `userid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup użytkowników we właściwości `usrgrpids`.

Przykłady

Włączanie grupy użytkowników i aktualizacja uprawnień

Włącz grupę użytkowników i zapewnij jej dostęp do odczytu i zapisu do grup hostów „2” i „4”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.update",
    "params": {
        "usrgrpid": "17",
        "users_status": "0",
        "hostgroup_rights": [
            {
                "id": "2",
                "permission": 3
            },
            {
                "id": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "17"
        ]
```

```
    },
    "id": 1
}
```

Zobacz także

- Uprawnienie

Źródło

CUserGroup::update() w *ui/include/classes/api/services/CUserGroup.php*.

Historia

Ta klasa jest przeznaczona do pracy z danymi historii.

Odwołania do obiektów:

- Historia typu float
- Historia typu integer
- Historia typu string
- Historia tekstowa
- Historia logów

Dostępne metody:

- history.clear - wyczyść dane historii
- history.get - pobierz dane historii
- history.push - wyślij dane historii do serwer Zabbix

Obiekt Historia

Następujące obiekty są bezpośrednio powiązane z API `history`.

Note:
Obiekty historii różnią się w zależności od typu informacji elementu. Są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Historia wartości zmiennoprzecinkowych

Obiekt historii wartości zmiennoprzecinkowych ma następujące właściwości.

Właściwość	Type	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	float	Otrzymana wartość.

Historia liczb całkowitych

Obiekt historii liczb całkowitych ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Czas, kiedy ta wartość została odebrana.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie odebrania wartości.
value	integer	Odebrana wartość.

Historia ciągów znaków

Obiekt historii ciągów znaków ma następujące właściwości.

Właściwość	Type	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	string	Otrzymana wartość.

Historia tekstu

Obiekt historii tekstu ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID wpisu historii.
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	text	Otrzymana wartość.

Historia logów

Obiekt historii logów ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID wpisu historii.
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
logeventid	integer	ID wpisu dziennika zdarzeń systemu Windows.
ns	integer	Nanosekundy w momencie otrzymania wartości.
severity	integer	Poziom wpisu dziennika zdarzeń systemu Windows.
source	string	Źródło wpisu dziennika zdarzeń systemu Windows.
timestamp	timestamp	Czas wpisu dziennika zdarzeń systemu Windows.
value	text	Otrzymana wartość.

Historia binarna

Obiekt historii binarnej ma następujące właściwości.

Property	Type	Description
clock	timestamp	Czas, kiedy ta wartość została odebrana.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy, kiedy wartość została odebrana.
value	text	Odebrana wartość.

history.clear

Opis

`object history.clear(array itemids)`

Ta metoda pozwala wyczyścić historię pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` Identyfikatory pozycji do wyczyszczenia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory wyczyszczonych pozycji we właściwości `itemids`.

Przykłady

Wyczyść historię

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "history.clear",
"params": [
"10325",
"13205"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"wynik": {
"itemids": [
"10325",
"13205"
]
},
"id": 1
}
```

Źródło

CHistory::clear() w *ui/include/classes/api/services/CHistory.php*.

history.get

Opis

`integer/array history.get(object parameters)`

Ta metoda umożliwia pobranie danych historycznych zgodnie z podanymi parametrami.

Attention:
Ta metoda może zwrócić dane historyczne usuniętej encji, jeśli dane te nie zostały jeszcze usunięte przez housekeeper.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
history	integer	Typy obiektów historii do zwrócenia. Możliwe wartości: 0 - liczba zmiennoprzecinkowa; 1 - znak; 2 - log; 3 - *(domyślnie)* liczba całkowita bez znaku; 4 - tekst; 5 - binarny.
hostids	ID/array	Zwraca tylko historię z podanych hostów.
itemids	ID/array	Zwraca tylko historię z podanych pozycji.

Parameter	Type	Description
time_from	timestamp	Zwraca tylko wartości, które zostały odebrane po podanym czasie lub dokładnie w tym czasie.
time_till	timestamp	Zwraca tylko wartości, które zostały odebrane przed podanym czasem lub dokładnie w tym czasie.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `clock`, `ns`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie zostaną podane dodatkowe opcje, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Nie obsługuje wyszukiwania według `value`, jeśli `history` ma ustawioną wartość 6 (JSON).
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych historii pozycji

Zwraca 10 najnowszych wartości otrzymanych z pozycji typu numeric(float).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "history.get",
    "params": {
        "output": "extend",
        "history": 0,
        "itemids": "23296",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 10
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23296",
            "clock": "1351090996",
```

```
            "value": "0.085",
            "ns": "563157632"
        },
        {
            "itemid": "23296",
            "clock": "1351090936",
            "value": "0.16",
            "ns": "549216402"
        },
        {
            "itemid": "23296",
            "clock": "1351090876",
            "value": "0.18",
            "ns": "537418114"
        },
        {
            "itemid": "23296",
            "clock": "1351090816",
            "value": "0.21",
            "ns": "522659528"
        },
        {
            "itemid": "23296",
            "clock": "1351090756",
            "value": "0.215",
            "ns": "507809457"
        },
        {
            "itemid": "23296",
            "clock": "1351090696",
            "value": "0.255",
            "ns": "495509699"
        },
        {
            "itemid": "23296",
            "clock": "1351090636",
            "value": "0.36",
            "ns": "477708209"
        },
        {
            "itemid": "23296",
            "clock": "1351090576",
            "value": "0.375",
            "ns": "463251343"
        },
        {
            "itemid": "23296",
            "clock": "1351090516",
            "value": "0.315",
            "ns": "447947017"
        },
        {
            "itemid": "23296",
            "clock": "1351090456",
            "value": "0.275",
            "ns": "435307141"
        }
    ],
    "id": 1
}
```


Źródło

CHistory::get() w *ui/include/classes/api/services/CHistory.php*.

history.push

Opis

`object history.push(object/array itemHistoryData)`

Ta metoda umożliwia wysyłanie danych historii pozycji do serwer Zabbix.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Dane historii pozycji do wysłania.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
itemid	ID	ID powiązanej pozycji. Zachowanie parametru: - *wymagany*, jeśli `host` i `key` nie są ustawione
host	string	Nazwa techniczna hosta. Zachowanie parametru: - *wymagany*, jeśli `itemid` nie jest ustawione
key	string	Klucz pozycji. Zachowanie parametru: - *wymagany*, jeśli `itemid` nie jest ustawione
value	mixed	Wartość pozycji. Zachowanie parametru: - *wymagany*
clock	timestamp	Czas otrzymania wartości.
ns	integer	Nanosekundy w momencie otrzymania wartości.

Zwracane wartości

`(object)` Zwraca wynik operacji wysyłania danych.

Przykłady

Wysyłanie danych historii pozycji

Wyślij dane historii pozycji do serwera Zabbix dla pozycji „10600”, „10601” i „999999”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "history.push",
    "params": [
        {
            "itemid": 10600,
            "value": 0.5,
            "clock": 1690891294,
            "ns": 45440940
        },
        {
            "itemid": 10600,
            "value": 0.6,
            "clock": 1690891295,
```

```
            "ns": 312431
        },
        {
            "itemid": 10601,
            "value": "[Tue Aug 01 15:01:35 2023] [error] [client 1.2.3.4] File does not exist: /var/www/ht
        },
        {
            "itemid": 999999,
            "value": 123
        }
    ],
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "Item is disabled."
            },
            {
                "error": "No permissions to referred object or it does not exist."
            }
        ]
    },
    "id": 1
}
```


Zobacz także

- Pozycje trapper
- pozycje HTTP agent
- Host
- Pozycja

Źródło

CHistory::push() w *ui/include/classes/api/services/CHistory.php*.

host

Ta klasa jest przeznaczona do pracy z hostami.

Odwołania do obiektów:

- Host
- Host inventory
- Host tag

Dostępne metody:

- host.create - tworzenie nowych hostów
- host.delete - usuwanie hostów
- host.get - pobieranie hostów
- host.massadd - dodawanie powiązanych obiektów do hostów
- host.massremove - usuwanie powiązanych obiektów z hostów

- host.massupdate - zastępowanie lub usuwanie powiązanych obiektów z hostów
- host.update - aktualizowanie hostów

Obiekt hosta

Następujące obiekty są bezpośrednio powiązane z interfejsem API hosta.

Host

Obiekt host ma następujące właściwości.

Właściwość	Typ	Opis
hostid	ID	ID hosta. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
host	string	Techniczna nazwa hosta. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	text	Opis hosta.
flags	integer	Pochodzenie hosta. Możliwe wartości: 0 - zwykły host; 4 - host przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*
inventory_mode	integer	Tryb wypełniania inwentarza hosta. Możliwe wartości: -1 - *(domyślnie)* wyłączony; 0 - ręczny; 1 - automatyczny.
ipmi_authtype	integer	Algorytm uwierzytelniania IPMI. Możliwe wartości: -1 - *(domyślnie)* domyślny; 0 - brak; 1 - MD2; 2 - MD5 4 - straight; 5 - OEM; 6 - RMCP+.
ipmi_password	string	Hasło IPMI.
ipmi_privilege	integer	Poziom uprawnień IPMI. Możliwe wartości: 1 - callback; 2 - *(domyślnie)* użytkownik; 3 - operator; 4 - administrator; 5 - OEM.
ipmi_username	string	Nazwa użytkownika IPMI.
maintenance_from	timestamp	Czas rozpoczęcia obowiązującego okresu utrzymaniowego. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
maintenance_status	integer	Status obowiązującego okresu utrzymaniowego. Możliwe wartości: 0 - *(domyślnie)* brak utrzymania; 1 - utrzymanie obowiązuje. Zachowanie właściwości: - *tylko do odczytu*
maintenance_type	integer	Typ obowiązującego okresu utrzymaniowego. Możliwe wartości: 0 - *(domyślnie)* utrzymanie z gromadzeniem danych; 1 - utrzymanie bez gromadzenia danych. Zachowanie właściwości: - *tylko do odczytu*
maintenanceid	ID	ID utrzymania, które obecnie obowiązuje dla hosta. Zachowanie właściwości: - *tylko do odczytu*
name	string	Widoczna nazwa hosta. Domyślnie: wartość właściwości `host`.
monitored_by	integer	Źródło używane do monitorowania hosta. Możliwe wartości: 0 - *(domyślnie)* serwer Zabbix; 1 - proxy; 2 - grupa proxy.
proxyid	ID	ID proxy używanego do monitorowania hosta. Zachowanie właściwości: - *wymagane* jeśli `monitored_by` jest ustawione na ”Proxy”
proxy_groupid	ID	ID grupy proxy używanej do monitorowania hosta. Zachowanie właściwości: - *wymagane* jeśli `monitored_by` jest ustawione na ”Proxy group”
status	integer	Status i funkcja hosta. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
tls_connect	integer	Połączenia do hosta. Możliwe wartości: 1 - *(domyślnie)* bez szyfrowania; 2 - PSK; 4 - certyfikat.
tls_accept	integer	Połączenia od hosta. Możliwe wartości bitmapy: 1 - *(domyślnie)* bez szyfrowania; 2 - PSK; 4 - certyfikat. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 6 dla PSK i certyfikatu).
tls_issuer	string	Wystawca certyfikatu.
tls_subject	string	Podmiot certyfikatu.

Właściwość	Typ	Opis
tls_psk_identity	string	Tożsamość PSK; musi być sparowana tylko z jednym PSK (w autoregistration, hosts i proxies). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona wysyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć. Zachowanie właściwości: - *tylko do zapisu* - *wymagane* jeśli `tls_connect` jest ustawione na ”PSK” lub `tls_accept` zawiera bit ”PSK”
tls_psk	string	Klucz współdzielony (PSK); musi mieć co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: - *tylko do zapisu* - *wymagane* jeśli `tls_connect` jest ustawione na ”PSK” lub `tls_accept` zawiera bit ”PSK”
active_available	integer	Status dostępności aktywnego interfejsu hosta. Możliwe wartości: 0 - status interfejsu jest nieznany; 1 - interfejs jest dostępny; 2 - interfejs jest niedostępny. Zachowanie właściwości: - *tylko do odczytu*
assigned_proxyid	ID	ID proxy przypisanego przez serwer Zabbix, jeśli host jest monitorowany przez grupę proxy. Zachowanie właściwości: - *tylko do odczytu*

Inwentarz hosta

Obiekt inwentarza hosta ma następujące właściwości.

Note:
Każda właściwość ma własny unikalny numer ID, który jest używany do powiązania pól inwentarza hosta z pozycjami.

ID	Właściwość	Typ	Opis	Maksymalna długość
4	alias	string	Alias.	128 znaków
11	asset_tag	string	Znacznik zasobu.	64 znaki
28	chassis	string	Obudowa.	64 znaki
23	contact	string	Osoba kontaktowa.	65535 znaków dla baz danych SQL
32	contract_number	string	Numer umowy.	64 znaki
47	date_hw_decomm	string	Data wycofania sprzętu z eksploatacji.	64 znaki
46	date_hw_expiry	string	Data wygaśnięcia obsługi technicznej sprzętu.	64 znaki
45	date_hw_install	string	Data instalacji sprzętu.	64 znaki
44	date_hw_purchase	string	Data zakupu sprzętu.	64 znaki
34	deployment_status	string	Status wdrożenia.	64 znaki
14	hardware	string	Sprzęt.	255 znaków
15	hardware_full	string	Szczegółowy opis sprzętu.	65535 znaków dla baz danych SQL
39	host_netmask	string	Maska podsieci hosta.	39 znaków
38	host_networks	string	Sieci hosta.	65535 znaków dla baz danych SQL
40	host_router	string	Router hosta.	39 znaków
30	hw_arch	string	Architektura sprzętu.	32 znaki
33	installer_name	string	Nazwa instalatora.	64 znaki
24	location	string	Lokalizacja.	65535 znaków dla baz danych SQL

ID	Właściwość	Typ	Opis	Maksymalna długość
25	location_lat	string	Szerokość geograficzna lokalizacji.	16 znaków
26	location_lon	string	Długość geograficzna lokalizacji.	16 znaków
12	macaddress_a	string	Adres MAC A.	64 znaki
13	macaddress_b	string	Adres MAC B.	64 znaki
29	model	string	Model.	64 znaki
3	name	string	Nazwa.	128 znaków
27	notes	string	Notatki.	65535 znaków dla baz danych SQL
41	oob_ip	string	Adres IP OOB.	39 znaków
42	oob_netmask	string	Maska podsieci hosta OOB.	39 znaków
43	oob_router	string	Router OOB.	39 znaków
5	os	string	Nazwa systemu operacyjnego.	128 znaków
6	os_full	string	Szczegółowa nazwa systemu operacyjnego.	255 znaków
7	os_short	string	Krótka nazwa systemu operacyjnego.	128 znaków
61	poc_1_cell	string	Główny numer telefonu komórkowego POC.	64 znaki
58	poc_1_email	string	Główny adres e-mail.	128 znaków
57	poc_1_name	string	Nazwa głównego POC.	128 znaków
63	poc_1_notes	string	Notatki głównego POC.	65535 znaków dla baz danych SQL
59	poc_1_phone_a	string	Główny telefon POC A.	64 znaki
60	poc_1_phone_b	string	Główny telefon POC B.	64 znaki
62	poc_1_screen	string	Nazwa ekranowa głównego POC.	64 znaki
68	poc_2_cell	string	Drugorzędny numer telefonu komórkowego POC.	64 znaki
65	poc_2_email	string	Drugorzędny adres e-mail POC.	128 znaków
64	poc_2_name	string	Nazwa drugorzędnego POC.	128 znaków
70	poc_2_notes	string	Notatki drugorzędnego POC.	65535 znaków dla baz danych SQL
66	poc_2_phone_a	string	Drugorzędny telefon POC A.	64 znaki
67	poc_2_phone_b	string	Drugorzędny telefon POC B.	64 znaki
69	poc_2_screen	string	Nazwa ekranowa drugorzędnego POC.	64 znaki
8	serialno_a	string	Numer seryjny A.	64 znaki
9	serialno_b	string	Numer seryjny B.	64 znaki
48	site_address_a	string	Adres lokalizacji A.	128 znaków
49	site_address_b	string	Adres lokalizacji B.	128 znaków
50	site_address_c	string	Adres lokalizacji C.	128 znaków
51	site_city	string	Miasto lokalizacji.	128 znaków
53	site_country	string	Kraj lokalizacji.	64 znaki
56	site_notes	string	Notatki dotyczące lokalizacji.	65535 znaków dla baz danych SQL
55	site_rack	string	Lokalizacja szafy w miejscu instalacji.	128 znaków
52	site_state	string	Województwo/stan lokalizacji.	64 znaki
54	site_zip	string	Kod ZIP/pocztowy lokalizacji.	64 znaki
16	software	string	Oprogramowanie.	255 znaków
18	software_app_a	string	Aplikacja A.	64 znaki
19	software_app_b	string	Aplikacja B.	64 znaki
20	software_app_c	string	Aplikacja C.	64 znaki
21	software_app_d	string	Aplikacja D.	64 znaki
22	software_app_e	string	Aplikacja E.	64 znaki
17	software_full	string	Szczegóły oprogramowania.	65535 znaków dla baz danych SQL
10	tag	string	Tag.	64 znaki
1	type	string	Typ.	64 znaki
2	type_full	string	Szczegóły typu.	64 znaki
35	url_a	string	URL A.	2048 znaków
36	url_b	string	URL B.	2048 znaków
37	url_c	string	URL C.	2048 znaków
31	vendor	string	Dostawca.	64 znaki

Tag hosta

Obiekt tagu hosta ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu hosta. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu hosta.
automatic	integer	Typ tagu hosta. Możliwe wartości: 0 - *(domyślnie)* ręczny (tag utworzony przez użytkownika); 1 - automatyczny (tag utworzony przez wykrywanie niskiego poziomu) Zachowanie właściwości: - *tylko do odczytu*

host.create

Opis

`object host.create(object/array hosts)`

Ta metoda pozwala tworzyć nowe hosty.

Note:
Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Hosty do utworzenia.

Oprócz standardowych właściwości hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy hostów, do których należy dodać host. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *wymagany*
interfaces	object/array	Interfejsy, które mają zostać utworzone dla hosta.
tags	object/array	Tagi hosta.
templates	object/array	Szablony, które mają zostać podłączone do hosta. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla hosta.
inventory	object	Właściwości inwentarza hosta.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych hostów we właściwości `hostids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych hostów.

Przykłady

Tworzenie hosta

Utwórz host o nazwie „Linux server” z interfejsem IP i tagami, dodaj go do grupy, podłącz do niego szablon i ustaw adresy MAC w inwentarzu hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Linux server",
        "interfaces": [
            {
                "type": 1,
                "main": 1,
                "useip": 1,
                "ip": "192.168.3.1",
                "dns": "",
                "port": "10050"
            }
        ],
        "groups": [
            {
                "groupid": "50"
            }
        ],
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server"
            }
        ],
        "templates": [
            {
                "templateid": "20045"
            }
        ],
        "macros": [
            {
                "macro": "{$USER_ID}",
                "value": "123321"
            },
            {
                "macro": "{$USER_LOCATION}",
                "value": "0:0:0",
                "description": "latitude, longitude and altitude coordinates"
            }
        ],
        "inventory_mode": 0,
        "inventory": {
            "macaddress_a": "01234",
            "macaddress_b": "56768"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "107819"
        ]
    },
    "id": 1
}
```

Tworzenie hosta z interfejsem SNMP

Utwórz host o nazwie ”SNMP host” z interfejsem SNMPv3 ze szczegółami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "SNMP host",
        "interfaces": [
            {
                "type": 2,
                "main": 1,
                "useip": 1,
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
                "details": {
                    "version": 3,
                    "bulk": 0,
                    "securityname": "mysecurityname",
                    "contextname": "",
                    "securitylevel": 1
                }
            }
        ],
        "groups": [
            {
                "groupid": "4"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10658"
        ]
    },
    "id": 1
}
```

Tworzenie hosta z szyfrowaniem PSK

Utwórz host o nazwie „PSK host” ze skonfigurowanym szyfrowaniem PSK. Pamiętaj, że host musi być wstępnie skonfigurowany do używania PSK.

Żądanie:

```
{
    "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"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Tworzenie hosta monitorowanego przez proxy

Utwórz host, który będzie monitorowany przez proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Host monitored by proxy",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 1,
        "proxyid": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10591"
        ]
    },
    "id": 1
}
```

Tworzenie hosta monitorowanego przez grupę proxy

Utwórz host, który będzie monitorowany przez grupę proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Host monitored by proxy group",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 2,
        "proxy_groupid": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10592"
        ]
    },
    "id": 1
}
```

Zobacz także

- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Inwentaryzacja hosta
- Tag hosta
- Proxy
- Grupa proxy

Źródło

CHost::create() w *ui/include/classes/api/services/CHost.php*.

host.delete

Opis

`object host.delete(array hosts)`

Ta metoda umożliwia usuwanie hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych hostów w właściwości `hostids`.

Przykłady

Usuwanie wielu hostów

Usuń dwa hosty.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "host.delete",
"params": [
"13",
"32"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostids": [
"13",
"32"
]
},
"id": 1
}
```

Źródło

CHost::delete() w *ui/include/classes/api/services/CHost.php*.

host.get

Opis

`integer/array host.get(object parameters)`

Metoda pozwala na pobieranie hostów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwracaj tylko hosty należące do podanych grup.
dserviceids	ID/array	Zwracaj tylko hosty powiązane z podanymi wykrytymi usługami.
graphids	ID/array	Zwracaj tylko hosty, które mają podane wykresy.
hostids	ID/array	Zwracaj tylko hosty o podanych identyfikatorach hostów.
httptestids	ID/array	Zwracaj tylko hosty, które mają podane testy WWW.
interfaceids	ID/array	Zwracaj tylko hosty używające podanych interfejsów.
itemids	ID/array	Zwracaj tylko hosty, które mają podane pozycje.
maintenanceids	ID/array	Zwracaj tylko hosty objęte podanymi okresami konserwacji.
monitored_hosts	flag	Zwracaj tylko monitorowane hosty.
proxyids	ID/array	Zwracaj tylko hosty monitorowane przez podane proxy.
proxy_groupids	ID/array	Zwracaj tylko hosty monitorowane przez podane grupy proxy.
templated_hosts	flag	Zwracaj zarówno hosty, jak i szablony.
templateids	ID/array	Zwracaj tylko hosty połączone z podanymi szablonami.
triggerids	ID/array	Zwracaj tylko hosty, które mają podane wyzwalacze.
with_items	flag	Zwracaj tylko hosty, które mają pozycje. Nadpisuje parametry `with_monitored_items` i `with_simple_graph_items`.

Parametr	Typ	Opis
with_item_prototypes	flag	Zwracaj tylko hosty, które mają prototypy pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	flag	Zwracaj tylko hosty, które mają prototypy pozycji włączone do tworzenia i z numerycznym typem informacji.
with_graphs	flag	Zwracaj tylko hosty, które mają wykresy.
with_graph_prototypes	flag	Zwracaj tylko hosty, które mają prototypy wykresów.
with_httptests	flag	Zwracaj tylko hosty, które mają testy WWW. Nadpisuje parametr `with_monitored_httptests`.
with_monitored_httptests	flag	Zwracaj tylko hosty, które mają włączone testy WWW.
with_monitored_items	flag	Zwracaj tylko hosty, które mają włączone pozycje. Nadpisuje parametr `with_simple_graph_items`.
with_monitored_triggers	flag	Zwracaj tylko hosty, które mają włączone wyzwalacze. Wszystkie pozycje używane w wyzwalaczu również muszą być włączone.
with_simple_graph_items	flag	Zwracaj tylko hosty, które mają pozycje z numerycznym typem informacji.
with_triggers	flag	Zwracaj tylko hosty, które mają wyzwalacze. Nadpisuje parametr `with_monitored_triggers`.
withProblemsSuppressed	boolean	Jeśli ustawiono na `true`, zwracaj tylko hosty z wyciszonymi problemami.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
severities	integer/array	Zwracaj hosty, które mają tylko problemy o podanych poziomach ważności. Dotyczy tylko sytuacji, gdy obiektem problemu jest wyzwalacz.
tags	object/array	Zwracaj tylko hosty z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie hosty. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
inheritedTags	boolean	Zwracaj hosty, które mają podane `tags` również we wszystkich połączonych z nimi szablonach. Możliwe wartości: `true` - połączone szablony również muszą mieć podane `tags`; `false` - *(domyślnie)* tagi połączonych szablonów są ignorowane.

Parametr	Typ	Opis
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania hosta. Obiekt wykrywania hosta łączy wykryty host z prototypem hosta, z którego został wykryty. Ma następujące właściwości: `host` - `(string)` identyfikator hosta; `parent_hostid` - `(string)` identyfikator prototypu hosta, z którego host został utworzony; `status` - `(int)` status wykrywania hosta: 0 - *(domyślnie)* host jest wykryty, 1 - host nie jest już wykryty; `ts_delete` - `(timestamp)` czas, kiedy host, który nie jest już wykryty, zostanie usunięty; `ts_disable` - `(timestamp)` czas, kiedy host, który nie jest już wykryty, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy host został wyłączony przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą wykrywania niskiego poziomu, która utworzyła host (z prototypu hosta w monitorowaniu VMware).
selectDiscoveryRules	query	Zwracaj właściwość `discoveryRules` z regułami LLD hosta. Obsługuje `count`.
selectGraphs	query	Zwracaj właściwość `graphs` z wykresami hosta. Obsługuje `count`.
selectHostGroups	query	Zwracaj właściwość `hostgroups` z danymi grup hostów, do których należy host.
selectHttpTests	query	Zwracaj właściwość `httpTests` ze scenariuszami WWW hosta. Obsługuje `count`.
selectInterfaces	query	Zwracaj właściwość `interfaces` z interfejsami hosta. Obsługuje `count`.
selectInventory	query	Zwracaj właściwość `inventory` z danymi inwentarza hosta.
selectItems	query	Zwracaj właściwość `items` z pozycjami hosta. Obsługuje `count`.
selectMacros	query	Zwracaj właściwość `macros` z makrami hosta.
selectParentTemplates	query	Zwracaj właściwość `parentTemplates` z szablonami, z którymi host jest połączony. Oprócz pól obiektu Template zawiera ona `link_type` - `(integer)` sposób, w jaki szablon jest połączony z hostem. Możliwe wartości: 0 - *(domyślnie)* połączony ręcznie; 1 - połączony automatycznie przez LLD. Obsługuje `count`.
selectDashboards	query	Zwracaj właściwość `dashboards`. Obsługuje `count`.
selectTags	query	Zwracaj właściwość `tags` z tagami hosta.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami obecnymi we wszystkich szablonach połączonych z hostem.
selectTriggers	query	Zwracaj właściwość `triggers` z wyzwalaczami hosta. Obsługuje `count`.
selectValueMaps	query	Zwracaj właściwość `valuemaps` z mapowaniami wartości hosta.

Parametr	Typ	Opis
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: właściwości interfejsu hosta.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectParentTemplates` - wyniki będą sortowane według `host`; `selectInterfaces`; `selectItems` - sortowane według `name`; `selectDiscoveryRules` - sortowane według `name`; `selectTriggers` - sortowane według `description`; `selectGraphs` - sortowane według `name`; `selectDashboards` - sortowane według `name`.
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Obsługuje dodatkowe właściwości: właściwości interfejsu hosta.
searchInventory	object	Zwracaj hosty, których dane inwentarza pasują do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectDiscoveries	query	Zwracaj właściwość `discoveries` z regułami wykrywania niskiego poziomu hosta. Obsługuje `count`. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryRules`.

Parametr	Typ	Opis
selectHostDiscovery	query	Zwracaj właściwość `hostDiscovery` z danymi obiektu wykrywania hosta. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch hostach o nazwach „Zabbix server” i „Linux server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "filter": {
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10160",
            "proxyid": "0",
            "host": "Zabbix server",
            "status": "0",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Zabbix server",
            "flags": "0",
            "description": "Serwer monitorowania Zabbix.",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "inventory_mode": "1",
            "active_available": "1",
            "assigned_proxyid": "0"
```

```
        },
        {
            "hostid": "10167",
            "proxyid": "0",
            "host": "Linux server",
            "status": "0",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Linux server",
            "flags": "0",
            "description": "",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "inventory_mode": "1",
            "active_available": "1",
            "assigned_proxyid": "0"
        }
    ],
    "id": 1
}
```

Pobieranie grup hostów

Pobierz grupy hostów, do których należy host „Zabbix server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectHostGroups": "extend",
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10085",
            "hostgroups": [
                {
                    "groupid": "2",
                    "name": "Linux servers",
                    "flags": "0",
                    "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
```

```
                },
                {
                    "groupid": "4",
                    "name": "Zabbix servers",
                    "flags": "0",
                    "uuid": "6f6799aa69e844b4b3918f779f2abf08"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie podłączonych szablonów

Pobierz identyfikatory i nazwy szablonów podłączonych do hosta „10084”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectParentTemplates": [
            "templateid",
            "name"
        ],
        "hostids": "10084"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "parentTemplates": [
                {
                    "name": "Linux",
                    "templateid": "10001"
                },
                {
                    "name": "Zabbix Server",
                    "templateid": "10047"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie hostów według szablonu

Pobierz hosty, które mają podłączony szablon „10001” (*Linux by Zabbix agent*).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "name"],
        "templateids": "10001"
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie według danych inwentaryzacyjnych hosta

Pobierz hosty, które zawierają „Linux” w polu inwentaryzacyjnym hosta „OS”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "host"
        ],
        "selectInventory": [
            "os"
        ],
        "searchInventory": {
            "os": "Linux"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "inventory": {
                "os": "Linux Ubuntu"
            }
        },
        {
```

```
            "hostid": "10107",
            "host": "Linux server",
            "inventory": {
                "os": "Linux Mint"
            }
        }
    ],
    "id": 1
}
```

Wyszukiwanie według tagów hosta

Pobierz hosty, które mają tag ”host-name” równy ”linux-server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10085",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server",
                    "automatic": "0"
                },
                {
                    "tag": "os",
                    "value": "rhel-7",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobierz hosty, które mają te tagi nie tylko na poziomie hosta, ale także w swoich podłączonych nadrzędnych szablonach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
```

```
        "tags": [
            {
                "tag": "os",
                "value": "rhel-7",
                "operator": 1
            }
        ],
        "inheritedTags": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10623",
            "name": "PC room 1"
        },
        {
            "hostid": "10601",
            "name": "Office"
        }
    ],
    "id": 1
}
```

Pobierz hosty z wyłączeniem tych, które mają tag ”component”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "component",
                "value": "",
                "operator": 5
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "tags": [
                {
                    "tag": "class",
                    "value": "os",
                    "automatic": "0"
                },
                {
```

```
                    "tag": "target",
                    "value": "linux",
                    "automatic": "0"
                }
            ]
        },
        {
            "hostid": "10629",
            "tags": [
                {
                    "tag": "class",
                    "value": "application",
                    "automatic": "0"
                },
                {
                    "tag": "target",
                    "value": "browser",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie hosta za pomocą tagów i tagów szablonu

Pobierz host z tagami oraz wszystkimi tagami, które są powiązane z nadrzędnymi szablonami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "hostids": 10502,
        "selectTags": ["tag", "value"],
        "selectInheritedTags": ["tag", "value"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10502",
            "name": "Desktop",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server"
                },
                {
                    "tag": "os",
                    "value": "rhel-7"
                }
            ],
            "inheritedTags": [
                {
                    "tag": "class",
                    "value": "os"
```

```
                },
                {
                    "tag": "target",
                    "value": "linux"
                },
                {
                    "tag": "os",
                    "value": "rhel-7"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie hostów według ważności problemu

Pobierz hosty, które mają problemy o ważności „Disaster”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": 5
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10160",
            "name": "Zabbix server"
        }
    ],
    "id": 1
}
```

Pobierz hosty, które mają problemy o ważności „Average” i „High”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": [3, 4]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "20170",
            "name": "Database"
        },
```

```
        {
            "hostid": "20183",
            "name": "workstation"
        }
    ],
    "id": 1
}
```

Pobieranie pozycji hosta

Pobierz pozycje dla hosta "Zabbix server". Żądanie jest ograniczone do właściwości pozycji `itemid`, `name` i `status`. Ze względu na duży rozmiar odpowiedzi w przykładzie pokazano tylko podzbiór pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectItems": ["itemid","name","status"],
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostid": "10084",
      "items": [
        {
          "itemid": "42227",
          "name": "Host name of Zabbix agent running",
          "status": "0"
        },
        {
          "itemid": "42237",
          "name": "Zabbix agent ping",
          "status": "0"
        },
        {
          "itemid": "42250",
          "name": "Version of Zabbix agent running",
          "status": "0"
        },
        {
          "itemid": "42251",
          "name": "Maximum number of open file descriptors",
          "status": "0"
        },
        {
          "itemid": "42252",
          "name": "Maximum number of processes",
          "status": "0"
        },
        {
          "itemid": "69869",
```

```
            "name": "Interface enp0s3: Inbound packets discarded",
            "status": "0"
          },
          {
            "itemid": "69870",
            "name": "Interface enp0s3: Inbound packets with errors",
            "status": "0"
          },
          {
            "itemid": "69871",
            "name": "Interface enp0s3: Bits received",
            "status": "0"
          },
          {
            "itemid": "69872",
            "name": "Interface enp0s3: Outbound packets discarded",
            "status": "0"
          },
          {
            "itemid": "69873",
            "name": "Interface enp0s3: Outbound packets with errors",
            "status": "0"
          },
          {
            "itemid": "69874",
            "name": "Interface enp0s3: Bits sent",
            "status": "0"
          },
          {
            "itemid": "42253",
            "name": "Number of processes",
            "status": "0"
          },
          {
            "itemid": "42254",
            "name": "Number of running processes",
            "status": "0"
          },
          {
            "itemid": "42255",
            "name": "System boot time",
            "status": "0"
          }
        ]
      }
    ],
    "id": 1
}
```


Zobacz także

- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Proxy
- Grupa proxy

Źródło

CHost::get() w *ui/include/classes/api/services/CHost.php*.

host.massadd

Opis

`object host.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory hostów do zaktualizowania oraz obiekty do dodania do wszystkich hostów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
hosts	object/array	Hosty do zaktualizowania. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`. Zachowanie parametrów: - *wymagane*
groups	object/array	Grupy hostów do dodania do podanych hostów. Grupy hostów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
interfaces	object/array	Interfejsy hostów do utworzenia dla podanych hostów.
macros	object/array	Makra użytkownika do utworzenia dla podanych hostów.
templates	object/array	Szablony do podłączenia do podanych hostów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości `hostids`.

Przykłady

Dodawanie makr

Dodaj dwa nowe makra do dwóch hostów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "10160"
            },
            {
                "hostid": "10167"
            }
        ],
        "macros": [
            {
                "macro": "{$TEST1}",
                "value": "MACROTEST1"
            },
            {
                "macro": "{$TEST2}",
                "value": "MACROTEST2",
```

```
                "description": "Opis testowy"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10160",
            "10167"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta

Źródło

CHost::massAdd() w *ui/include/classes/api/services/CHost.php*.

host.massremove

Opis

`object host.massremove(object parameters)`

Ta metoda umożliwia usuwanie powiązanych obiektów z wielu hostów.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory hostów do zaktualizowania oraz obiekty, które należy usunąć.

Parametr	Typ	Opis
hostids	ID/array	Identyfikatory hostów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
groupids	ID/array	Identyfikatory grup hostów, z których należy usunąć podane hosty.
interfaces	object/array	Interfejsy hosta do usunięcia z podanych hostów. Obiekt interfejsu hosta musi mieć zdefiniowane tylko właściwości `ip`, `dns` i `port`.
macros	string/array	Makra użytkownika do usunięcia z podanych hostów.
templateids	ID/array	Identyfikatory szablonów, które należy odłączyć od podanych hostów.
templateids_clear	ID/array	Identyfikatory szablonów, które należy odłączyć i wyczyścić z podanych hostów.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości hostids.

Przykłady

Odłączanie szablonów

Odłącz szablon od dwóch hostów i usuń wszystkie encje pochodzące z szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.massremove",
    "params": {
        "hostids": ["69665", "69666"],
        "templateids_clear": "325"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- Makro użytkownika
- Interfejs hosta

Źródło

CHost::massRemove() w *ui/include/classes/api/services/CHost.php*.

host.massupdate

Opis

`object host.massupdate(object parameters)`

Ta metoda umożliwia jednoczesne zastąpienie lub usunięcie powiązanych obiektów oraz aktualizację właściwości wielu hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do jej wywołania mogą zostać odebrane w ustawieniach ról użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

(object) Parametry zawierające identyfikatory hostów do aktualizacji oraz właściwości, które powinny zostać zaktualizowane.

Oprócz standardowych właściwości hosta, metoda przyjmuje następujące parametry.

Parameter	Type	Description
hosts	object/array	Hosty do aktualizacji. Hosty muszą mieć zdefiniowaną wyłącznie właściwość hostid. Zachowanie parametru: - *required*

Parameter	Type	Description
groups	object/array	Grupy hostów do zastąpienia bieżących grup hostów, do których należą hosty. Grupy hostów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
interfaces	object/array	Interfejsy hosta do zastąpienia bieżących interfejsów hosta na podanych hostach.
inventory	object	Właściwości inwentarza hosta. Trybu inwentarza hosta nie można zaktualizować za pomocą parametru `inventory`; zamiast tego użyj `inventory_mode`.
macros	object/array	Makra użytkownika do zastąpienia bieżących makr użytkownika na podanych hostach.
templates	object/array	Szablony do zastąpienia aktualnie powiązanych szablonów na podanych hostach. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.
templates_clear	object/array	Szablony do odłączenia i usunięcia z podanych hostów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID zaktualizowanych hostów w właściwości `hostids`.

Przykłady

Włączanie wielu hostów

Włącz monitorowanie dwóch hostów, tj. ustaw ich status na 0.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.massupdate",
    "params": {
        "hosts": [
            {
                "hostid": "69665"
            },
            {
                "hostid": "69666"
            }
        ],
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- host.massadd
- host.massremove
- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta

Źródło

CHost::massUpdate() w *ui/include/classes/api/services/CHost.php*.

host.update

Opis

`object host.update(object/array hosts)`

Ta metoda umożliwia aktualizację istniejących hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object/array)` Właściwości hosta do zaktualizowania.

Właściwość `hostid` musi być zdefiniowana dla każdego hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Należy jednak pamiętać, że aktualizacja technicznej nazwy hosta spowoduje również aktualizację widocznej nazwy hosta (jeśli nie została podana osobno) na wartość technicznej nazwy hosta.

Oprócz standardowych właściwości hosta, metoda akceptuje następujące parametry.

Parameter	Type	Description
groups	object/array	Grupy hostów zastępujące bieżące grupy hostów, do których należy host. Wszystkie grupy hostów, które nie zostały wymienione w żądaniu, zostaną odłączone. Grupy hostów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
interfaces	object/array	Interfejsy hosta zastępujące bieżące interfejsy hosta. Wszystkie interfejsy, które nie zostały wymienione w żądaniu, zostaną usunięte.
tags	object/array	Tagi hosta zastępujące bieżące tagi hosta. Wszystkie tagi, które nie zostały wymienione w żądaniu, zostaną usunięte.
inventory	object	Właściwości inwentarza hosta.
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika. Wszystkie makra, które nie zostały wymienione w żądaniu, zostaną usunięte.
templates	object/array	Szablony zastępujące aktualnie podłączone szablony. Wszystkie szablony, które nie zostały wymienione w żądaniu, zostaną jedynie odłączone. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Parameter	Type	Description
templates_clear	object/array	Szablony do odłączenia i wyczyszczenia z hosta. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości `hostids`.

Przykłady

Włączanie monitorowania hosta

Włącz monitorowanie hosta, czyli ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Odłączanie szablonów

Odłącz i wyczyść dwa szablony z hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "templates_clear": [
            {
                "templateid": "10124"
            },
            {
                "templateid": "10125"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr hosta

Zastąp wszystkie makra hosta dwoma nowymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "macros": [
            {
                "macro": "{$PASS}",
                "value": "password"
            },
            {
                "macro": "{$DISC}",
                "value": "sda",
                "description": "Updated description"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Aktualizowanie inwentarza hosta

Zmień tryb inwentarza i dodaj lokalizację

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "inventory_mode": 0,
        "inventory": {
            "location": "Latvia, Riga"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów hosta

Zastąp wszystkie tagi hosta nowym tagiem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "tags": {
            "tag": "os",
            "value": "rhel-7"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr wykrytego hosta

Przekształć makro utworzone przez regułę wykrywania z trybu „automatic” na „manual” i zmień jego wartość na „new-value”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "macros": {
            "hostmacroid": "5541",
            "value": "new-value",
            "automatic": "0"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
```

```
    "id": 1
}
```

Aktualizacja szyfrowania hosta

Zaktualizuj host „10590”, aby używał wyłącznie szyfrowania PSK dla połączeń z hosta do serwera Zabbix, oraz zmień tożsamość PSK i klucz PSK. Zwróć uwagę, że host musi być wstępnie skonfigurowany do używania PSK.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.massadd
- host.massupdate
- host.massremove
- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Inwentarz hosta
- Tag hosta
- Proxy
- Grupa proxy

Źródło

CHost::update() w *ui/include/classes/api/services/CHost.php*.

Interfejs hosta

Ta klasa jest przeznaczona do pracy z interfejsami hosta.

Odwołania do obiektów:

- Interfejs hosta
  - Szczegóły

Dostępne metody:

- hostinterface.create - tworzy nowe interfejsy hosta
- hostinterface.delete - usuwa interfejsy hosta
- hostinterface.get - pobiera interfejsy hosta
- hostinterface.massadd - dodaje interfejsy hosta do hostów

- hostinterface.massremove - usuwa interfejsy hosta z hostów
- hostinterface.replacehostinterfaces - zastępuje interfejsy hosta na hoście
- hostinterface.update - aktualizuje interfejsy hosta

Obiekt interfejsu hosta

Następujące obiekty są bezpośrednio powiązane z API `hostinterface`.

Interfejs hosta

Obiekt interfejsu hosta ma następujące właściwości.

Attention:
Należy pamiętać, że zarówno właściwości `ip`, jak i `dns` są *wymagane* dla operacji tworzenia. Jeśli nie chcesz używać DNS, ustaw go jako pusty ciąg znaków.

Właściwość	Type	Opis
interfaceid	ID	ID interfejsu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
available	integer	Dostępność interfejsu hosta. Możliwe wartości: 0 - *(domyślnie)* nieznana; 1 - dostępny; 2 - niedostępny. Zachowanie właściwości: - *tylko do odczytu*
hostid	ID	ID hosta, do którego należy interfejs. Zachowanie właściwości: - *stała* - *wymagane* dla operacji tworzenia
type	integer	Typ interfejsu. Możliwe wartości: 1 - Agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
ip	string	Adres IP używany przez interfejs. Może być pusty, jeśli połączenie jest nawiązywane przez DNS. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
dns	string	Nazwa DNS używana przez interfejs. Może być pusta, jeśli połączenie jest nawiązywane przez IP. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Type	Opis
port	string	Numer portu używany przez interfejs. Może zawierać makra użytkownika. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
useip	integer	Czy połączenie powinno być nawiązywane przez IP. Możliwe wartości: 0 - połącz przy użyciu nazwy DNS hosta; 1 - połącz przy użyciu adresu IP hosta. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
main	integer	Czy interfejs jest używany jako domyślny na hoście. Tylko jeden interfejs danego typu może być ustawiony jako domyślny na hoście. Możliwe wartości: 0 - nie jest domyślny; 1 - domyślny. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
details	object	Dodatkowy obiekt details dla interfejsu. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SNMP”
disable_until	timestamp	Czas następnego odpytywania niedostępnego interfejsu hosta. Zachowanie właściwości: - *tylko do odczytu*
error	string	Tekst błędu, jeśli interfejs hosta jest niedostępny. Zachowanie właściwości: - *tylko do odczytu*
errors_from	timestamp	Czas, kiedy interfejs hosta stał się niedostępny. Zachowanie właściwości: - *tylko do odczytu*

Szczegóły

Obiekt szczegółów ma następujące właściwości.

Właściwość	Typ	Opis
version	integer	Wersja interfejsu SNMP. Możliwe wartości: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Zachowanie właściwości: - *wymagane*
bulk	integer	Czy używać zbiorczych żądań SNMP. Możliwe wartości: 0 - nie używaj zbiorczych żądań; 1 - (domyślnie) - używaj zbiorczych żądań.

Właściwość	Typ	Opis
community	string	Społeczność SNMP. Używana tylko przez interfejsy SNMPv1 i SNMPv2. Zachowanie właściwości: - *wymagane* jeśli `version` jest ustawione na "SNMPv1" lub "SNMPv2c"
max_repetitions	integer	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Używane tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Domyślnie: 10.
securityname	string	Nazwa bezpieczeństwa SNMPv3. Używana tylko przez interfejsy SNMPv3.
securitylevel	integer	Poziom bezpieczeństwa SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv.
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Używane tylko przez interfejsy SNMPv3.
privpassphrase	string	Hasło prywatności SNMPv3. Używane tylko przez interfejsy SNMPv3.
authprotocol	integer	Protokół uwierzytelniania SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512.
privprotocol	integer	Protokół prywatności SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C.
contextname	string	Nazwa kontekstu SNMPv3. Używana tylko przez interfejsy SNMPv3.

hostinterface.create

Opis

`object hostinterface.create(object/array hostInterfaces)`

Ta metoda umożliwia tworzenie nowych interfejsów hosta.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Interfejsy hosta do utworzenia.

Metoda akceptuje interfejsy hosta ze standardowymi właściwościami interfejsu hosta.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych interfejsów hosta we właściwości `interfaceids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych interfejsów hosta.

Przykłady

Utwórz nowy interfejs

Utwórz dodatkowy interfejs IP agenta na hoście „30052”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.create",
    "params": {
        "hostid": "30052",
        "main": "0",
        "type": "1",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "port": "10050"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30062"
        ]
    },
    "id": 1
}
```

Utwórz interfejs ze szczegółami SNMP

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.create",
    "params": {
        "hostid": "10456",
        "main": "0",
        "type": "2",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "port": "1601",
        "details": {
            "version": "2",
            "bulk": "1",
            "community": "{$SNMP_COMMUNITY}"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
```

```
            "30063"
        ]
    },
    "id": 1
}
```

Zobacz także

- hostinterface.massadd
- host.massadd

Źródło

CHostInterface::create() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.delete

Opis

`object hostinterface.delete(array hostInterfaceIds)`

Ta metoda umożliwia usuwanie interfejsów hosta.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID interfejsów hosta do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych interfejsów hosta we właściwości `interfaceids`.

Przykłady

Usuń interfejs hosta

Usuń interfejs hosta o identyfikatorze 30062.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostinterface.delete",
"params": [
"30062"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"interfaceids": [
"30062"
]
},
"id": 1
}
```

Zobacz również

- hostinterface.massremove
- host.massremove

Źródło

CHostInterface::delete() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.get

Opis

`integer/array hostinterface.get(object parameters)`

Ta metoda umożliwia pobieranie interfejsów hosta zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
hostids	ID/array	Zwraca tylko interfejsy hosta używane przez podane hosty.
interfaceids	ID/array	Zwraca tylko interfejsy hosta o podanych identyfikatorach.
itemids	ID/array	Zwraca tylko interfejsy hosta używane przez podane pozycje.
triggerids	ID/array	Zwraca tylko interfejsy hosta używane przez pozycje w podanych wyzwalaczach.
selectItems	query	Zwraca właściwość `items` z pozycjami, które używają interfejsu. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów używających interfejsu.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectItems`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `interfaceid`, `dns`, `ip`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie interfejsów hosta

Pobierz wszystkie dane o interfejsach używanych przez host „30057”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.get",
    "params": {
        "output": "extend",
        "hostids": "30057"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "interfaceid": "50039",
            "hostid": "30057",
            "main": "1",
            "type": "1",
            "useip": "0",
            "ip": "",
            "dns": "localhost",
            "port": "10050",
            "available": "0",
            "error": "",
            "errors_from": "0",
            "disable_until": "0",
            "details": []
        },
        {
            "interfaceid": "55082",
            "hostid": "30057",
            "main": "1",
            "type": "2",
            "useip": "1",
            "ip": "127.0.0.1",
            "dns": "",
            "port": "161",
            "available": "0",
            "error": "",
            "errors_from": "0",
            "disable_until": "0",
            "details": {
                "version": "2",
                "bulk": "0",
                "community": "{$SNMP_COMMUNITY}",
                "max_repetitions": "10"
            }
        }
    ],
    "id": 1
}
```

Zobacz także

- Host
- Item

Źródło

CHostInterface::get() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.massadd

Opis

`object hostinterface.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie interfejsów hosta do wielu hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry zawierające interfejsy hosta, które mają zostać utworzone na podanych hostach.

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
interfaces	object/array	Interfejsy hosta, które mają zostać utworzone na podanych hostach. Zachowanie parametru: - *wymagany*
hosts	object/array	Hosty, które mają zostać zaktualizowane. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych interfejsów hosta we właściwości `interfaceids`.

Przykłady

Tworzenie interfejsów

Utwórz interfejs na dwóch hostach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30052"
            }
        ],
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "main": 0,
            "port": "10050",
            "type": 1,
            "useip": 1
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30069",
            "30070"
        ]
    },
    "id": 1
}
```

Zobacz także

- hostinterface.create
- host.massadd
- Host

Źródło

CHostInterface::massAdd() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.massremove

Opis

`object hostinterface.massremove(object parameters)`

Ta metoda pozwala na usuwanie interfejsów hosta z podanych hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry zawierające identyfikatory hostów do zaktualizowania oraz interfejsy do usunięcia.

Parametr	Typ	Opis
interfaces	object/array	Interfejsy hosta do usunięcia z podanych hostów. Obiekt interfejsu hosta musi mieć zdefiniowane tylko właściwości `ip`, `dns` i `port`. Zachowanie parametru: - *wymagany*
hostids	ID/array	Identyfikatory hostów do zaktualizowania. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych interfejsów hosta we właściwości `interfaceids`.

Przykłady

Usuwanie interfejsów

Usuń interfejs SNMP „127.0.0.1” z dwóch hostów.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "hostinterface.massremove",
"parametry": {
```

```
"idy hosta": [
"30050",
"30052"
],
"interfejsy": {
"dns": "",
"ip": "127.0.0.1",
"port": "161"
}
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"wynik": {
"idy interfejsu": [
"30069",
"30070"
]
},
"id": 1
}
```

Zobacz także

- hostinterface.delete
- host.massremove

Źródło

CHostInterface::massRemove() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.replacehostinterfaces

Opis

`object hostinterface.replacehostinterfaces(object parameters)`

Ta metoda umożliwia zastąpienie wszystkich interfejsów hosta na wskazanym hoście.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające ID hosta do zaktualizowania oraz nowe interfejsy hosta.

Parameter	Type	Description
interfaces	object/array	Interfejsy hosta do zastąpienia bieżących interfejsów hosta. Zachowanie parametru: - *wymagany*
hostid	ID	ID hosta do zaktualizowania. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych interfejsów hosta w właściwości `interfaceids`.

Przykłady

Zastępowanie interfejsów hosta

Zastąp wszystkie interfejsy hosta jednym interfejsem agent.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.replacehostinterfaces",
    "params": {
        "hostid": "30052",
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "main": 1,
            "port": "10050",
            "type": 1,
            "useip": 1
        }
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30081"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- host.massupdate

Źródło

CHostInterface::replaceHostInterfaces() w *ui/include/classes/api/services/CHostInterface.php*.

hostinterface.update

Opis

`object hostinterface.update(object/array hostInterfaces)`

Ta metoda umożliwia aktualizację istniejących interfejsów hosta.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Właściwości interfejsu hosta, które mają zostać zaktualizowane.

Właściwość `interfaceid` musi być zdefiniowana dla każdego interfejsu hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych interfejsów hosta we właściwości `interfaceids`.

Przykłady

Zmiana portu interfejsu hosta

Zmień port interfejsu hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.update",
    "params": {
        "interfaceid": "30048",
        "port": "30050"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30048"
        ]
    },
    "id": 1
}
```

Źródło

CHostInterface::update() w *ui/include/classes/api/services/CHostInterface.php*.

Katalog użytkowników

Ta klasa jest przeznaczona do pracy z katalogami użytkowników.

Odwołania do obiektów:

- Katalog użytkowników
  - Mapowania typów mediów
  - Mapowania grup provisioningu

Dostępne metody:

- userdirectory.create - utwórz nowy katalog użytkowników
- userdirectory.delete - usuń katalog użytkowników
- userdirectory.get - pobierz katalog użytkowników
- userdirectory.update - zaktualizuj katalog użytkowników
- userdirectory.test - przetestuj połączenie z katalogiem użytkowników

Obiekt katalogu użytkownika

Następujące obiekty są bezpośrednio powiązane z API `userdirectory`.

Katalog użytkowników

Obiekt katalogu użytkowników ma następujące właściwości.

Property	Type	Description
userdirectoryid	ID	ID katalogu użytkowników. Jeśli katalog użytkowników zostanie usunięty, wartość właściwości obiektu User `userdirectoryid` zostanie ustawiona na ”0” dla wszystkich użytkowników powiązanych z usuniętym katalogiem użytkowników. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
idp_type	integer	Typ protokołu uwierzytelniania używanego przez dostawcę tożsamości dla katalogu użytkowników. Uwaga: może istnieć tylko jeden katalog użytkowników typu SAML. Możliwe wartości: 1 - Katalog użytkowników typu LDAP; 2 - Katalog użytkowników typu SAML. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
group_name	string	Atrybut katalogu użytkowników LDAP/SAML zawierający nazwę grupy używaną do mapowania grup między katalogiem użytkowników LDAP/SAML a Zabbix. Przykład: *cn* Zachowanie właściwości: - *wymagane* jeśli `provision_status` ma wartość ”Enabled” i `saml_jit_status` w obiekcie Authentication ma wartość ”Enabled for configured SAML IdPs”
user_username	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` ma wartość ”SCIM provisioning is enabled”) zawierający nazwę użytkownika, która jest używana jako wartość właściwości obiektu User `name` podczas aprowizacji użytkownika. Przykłady: *cn*, *commonName*, *displayName*, *name*
user_lastname	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` ma wartość ”SCIM provisioning is enabled”) zawierający nazwisko użytkownika, które jest używane jako wartość właściwości obiektu User `surname` podczas aprowizacji użytkownika. Przykłady: *sn*, *surname*, *lastName*
provision_status	integer	Status aprowizacji katalogu użytkowników. Możliwe wartości: 0 - *(domyślnie)* Wyłączone (aprowizacja użytkowników utworzonych przez ten katalog użytkowników jest wyłączona); 1 - Włączone (aprowizacja użytkowników utworzonych przez ten katalog użytkowników jest włączona; dodatkowo status aprowizacji LDAP lub SAML (`ldap_jit_status` lub `saml_jit_status` w obiekcie Authentication) musi być włączony).
provision_groups	array	Tablica obiektów mapowań grup aprowizacji służących do mapowania wzorca grupy użytkowników LDAP/SAML na grupę użytkowników i rolę użytkownika Zabbix. Zachowanie właściwości: - *wymagane* jeśli `provision_status` ma wartość ”Enabled”
provision_media	array	Tablica obiektów mapowań typów mediów służących do mapowania atrybutów mediów użytkownika LDAP/SAML (np. adresu e-mail) na media użytkownika Zabbix do wysyłania powiadomień.

Property	Type	Description
Właściwości specyficzne dla LDAP:		
name	string	Unikalna nazwa katalogu użytkowników. Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość ”User directory of type LDAP”
host	string	Nazwa hosta, adres IP lub URI serwera LDAP. URI musi zawierać schemat (`ldap://` lub `ldaps://`), host i port (opcjonalnie). Przykłady: *host.example.com* *127.0.0.1* *ldap://ldap.example.com:389* Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość ”User directory of type LDAP”
port	integer	Port serwera LDAP. Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość ”User directory of type LDAP”
base_dn	string	Bazowa ścieżka katalogu użytkowników LDAP do kont użytkowników. Przykłady: *ou=Users,dc=example,dc=org* *ou=Users,ou=system* (dla OpenLDAP) *DC=company,DC=com* (dla Microsoft Active Directory) *uid=%{user},dc=example,dc=com* (dla bezpośredniego wiązania użytkownika; symbol zastępczy ”*%{user}*” jest obowiązkowy) Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość ”User directory of type LDAP”
search_attribute	string	Atrybut katalogu użytkowników LDAP, według którego identyfikowane jest konto użytkownika na podstawie informacji podanych w żądaniu logowania. Przykłady: *uid* (dla OpenLDAP) *sAMAccountName* (dla Microsoft Active Directory) Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość ”User directory of type LDAP”
bind_dn	string	Konto serwera LDAP do wiązania i wyszukiwania w serwerze LDAP. W przypadku bezpośredniego wiązania użytkownika i anonimowego wiązania `bind_dn` musi być puste. Przykłady: *uid=ldap_search,ou=system* (dla OpenLDAP) *CN=ldap_search,OU=user_group,DC=company,DC=com* (dla Microsoft Active Directory) *CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość ”User directory of type LDAP”

Property	Type	Description
bind_password	string	Hasło LDAP konta do wiązania i wyszukiwania w serwerze LDAP. W przypadku bezpośredniego wiązania użytkownika i anonimowego wiązania `bind_password` musi być puste. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
description	string	Opis katalogu użytkowników. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
group_basedn	string	Bazowa ścieżka katalogu użytkowników LDAP do grup; używana do konfiguracji sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowane podczas aprowizacji użytkownika, jeśli ustawiono `group_membership`. Przykład: *ou=Groups,dc=example,dc=com* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
group_filter	string	Ciąg filtra służący do pobierania grup katalogu użytkowników LDAP, do których należy użytkownik; używany do konfiguracji sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowane podczas aprowizacji użytkownika, jeśli ustawiono `group_membership`. Obsługiwane symbole zastępcze `group_filter`: *%{attr}* - atrybut wyszukiwania (zastępowany wartością właściwości `search_attribute`); *%{groupattr}* - atrybut grupy (zastępowany wartością właściwości `group_member`); *%{host}* - nazwa hosta, adres IP lub URI serwera LDAP (zastępowany wartością właściwości `host`); *%{user}* - nazwa użytkownika Zabbix. Domyślnie: *(%{groupattr}=%{user})* Przykłady: - *(member=uid=%{ref},ou=Users,dc=example,dc=com)* dopasuje "User1", jeśli obiekt grupy LDAP zawiera atrybut "*member*" z wartością "*uid=User1,ou=Users,dc=example,dc=com*", i zwróci grupę, do której należy "User1"; - *(%{groupattr}=cn=%{ref},ou=Users,ou=Zabbix,DC=example,DC=com)* dopasuje "User1", jeśli obiekt grupy LDAP zawiera atrybut określony w właściwości `group_member` z wartością "*cn=User1,ou=Users,ou=Zabbix,DC=example,DC=com*", i zwróci grupę, do której należy "User1". Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"

Property	Type	Description
group_member	string	Atrybut katalogu użytkowników LDAP zawierający informacje o członkach grupy; używany do konfiguracji sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowane podczas aprowizacji użytkownika, jeśli ustawiono `group_membership`. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
group_membership	string	Atrybut katalogu użytkowników LDAP zawierający informacje o grupach, do których należy użytkownik. Przykład: *memberOf* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
search_filter	string	Niestandardowy ciąg filtra używany do lokalizowania i uwierzytelniania użytkownika w katalogu użytkowników LDAP na podstawie informacji podanych w żądaniu logowania. Obsługiwane symbole zastępcze `search_filter`: *%{attr}* - nazwa atrybutu wyszukiwania (np. *uid*, *sAMAccountName*); *%{user}* - nazwa użytkownika Zabbix. Domyślnie: *(%{attr}=%{user})* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
start_tls	integer	Opcja konfiguracji serwera LDAP, która umożliwia zabezpieczenie komunikacji z serwerem LDAP przy użyciu Transport Layer Security (TLS). Uwaga: `start_tls` musi mieć wartość "Disabled" dla hostów używających protokołu `ldaps://`. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
user_ref_attr	string	Atrybut katalogu użytkowników LDAP używany do odwołania do obiektu użytkownika. Wartość `user_ref_attr` jest używana do pobierania wartości z określonego atrybutu w katalogu użytkowników i zastępowania nimi symbolu zastępczego *%{ref}* w ciągu `group_filter`. Przykłady: *cn*, *uid*, *member*, *uniqueMember* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type LDAP"
Właściwości specyficzne dla SAML:		
idp_entityid	string	URI identyfikujący dostawcę tożsamości, używany do komunikacji z dostawcą tożsamości w komunikatach SAML. Przykład: *https://idp.example.com/idp* Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość "User directory of type SAML"

Property	Type	Description
sp_entityid	string	URL lub dowolny ciąg znaków identyfikujący dostawcę usług dostawcy tożsamości. Przykłady: *https://idp.example.com/sp* *zabbix*
username_attribute	string	Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość "User directory of type SAML" Atrybut katalogu użytkowników SAML (również atrybut SCIM, jeśli `scim_status` ma wartość "SCIM provisioning is enabled") zawierający nazwę użytkownika, która jest porównywana z wartością właściwości obiektu User `username` podczas uwierzytelniania. Przykłady: *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*
sso_url	string	Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość "User directory of type SAML" URL usługi pojedynczego logowania SAML dostawcy tożsamości, do której Zabbix będzie wysyłać żądania uwierzytelniania SAML. Przykład: *http://idp.example.com/idp/sso/saml*
slo_url	string	Zachowanie właściwości: - *wymagane* jeśli `idp_type` ma wartość "User directory of type SAML" URL usługi pojedynczego wylogowania SAML dostawcy tożsamości, do której Zabbix będzie wysyłać żądania wylogowania SAML. Przykład: *https://idp.example.com/idp/slo/saml*
encrypt_nameid	integer	Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML" Czy identyfikator Name ID w SAML powinien być szyfrowany. Możliwe wartości: 0 - *(domyślnie)* Nie szyfruj Name ID; 1 - Szyfruj Name ID.
encrypt_assertions	integer	Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML" Czy asercje SAML powinny być szyfrowane. Możliwe wartości: 0 - *(domyślnie)* Nie szyfruj asercji; 1 - Szyfruj asercje.
nameid_format	string	Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML" Format Name ID dostawcy usług dostawcy tożsamości SAML. Przykłady: *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* Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"

Property	Type	Description
scim_status	integer	Czy aprowizacja SCIM dla SAML jest włączona czy wyłączona. Możliwe wartości: 0 - *(domyślnie)* Aprowizacja SCIM jest wyłączona; 1 - Aprowizacja SCIM jest włączona. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"
sign_assertions	integer	Czy asercje SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - *(domyślnie)* Nie podpisuj asercji; 1 - Podpisuj asercje. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"
sign_authn_requests	integer	Czy żądania AuthN SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - *(domyślnie)* Nie podpisuj żądań AuthN; 1 - Podpisuj żądania AuthN. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"
sign_messages	integer	Czy komunikaty SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - *(domyślnie)* Nie podpisuj komunikatów; 1 - Podpisuj komunikaty. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"
sign_logout_requests	integer	Czy żądania wylogowania SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - *(domyślnie)* Nie podpisuj żądań wylogowania; 1 - Podpisuj żądania wylogowania. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"
sign_logout_responses	integer	Czy odpowiedzi wylogowania SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - *(domyślnie)* Nie podpisuj odpowiedzi wylogowania; 1 - Podpisuj odpowiedzi wylogowania. Zachowanie właściwości: - *obsługiwane* jeśli `idp_type` ma wartość "User directory of type SAML"

Mapowania typów mediów

Obiekt mapowań typów mediów ma następujące właściwości.

Właściwość	Typ	Opis
userdirectory_mediaid	ID	ID mapowania typu mediów. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
name	string	Widoczna nazwa na liście mapowań typów mediów. Zachowanie właściwości: - *wymagane*
mediatypeid	ID	ID typu mediów, który ma zostać utworzony; używane jako wartość właściwości `mediatypeid` obiektu Media. Zachowanie właściwości: - *wymagane*
attribute	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` jest ustawione na „Udostępnianie SCIM jest włączone”), który zawiera media użytkownika (np. *user@example.com*), używane jako wartość właściwości `sendto` obiektu Media. Jeśli występuje w danych otrzymanych od dostawcy tożsamości LDAP/SAML, a wartość nie jest pusta, spowoduje to utworzenie mediów dla udostępnionego użytkownika. Zachowanie właściwości: - *wymagane*
active	integer	Wartość właściwości `active` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Możliwe wartości: 0 - *(domyślnie)* włączone; 1 - wyłączone.
severity	integer	Wartość właściwości `severity` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Domyślnie: 63.
period	string	Wartość właściwości `period` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Domyślnie: 1-7,00:00-24:00.

Mapowania grup provisioningu

Mapowania grup provisioningu mają następujące właściwości.

Właściwość	Typ	Opis
name	string	Pełna nazwa grupy (np. *Zabbix administrators*) w katalogu użytkowników LDAP/SAML (również SCIM, jeśli `scim_status` jest ustawione na „SCIM provisioning is enabled”). Obsługuje znak wieloznaczny „*”. Unikalna wśród wszystkich mapowań grup provisioningu. Zachowanie właściwości: - *wymagane*
roleid	ID	ID roli użytkownika, która ma zostać przypisana użytkownikowi. Jeśli dopasowanych zostanie wiele mapowań grup provisioningu, użytkownikowi zostanie przypisana rola o najwyższym typie użytkownika (*User*, *Admin* lub *Super admin*). Jeśli istnieje wiele ról z tym samym typem użytkownika, użytkownikowi zostanie przypisana pierwsza rola (posortowana alfabetycznie). Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
user_groups	array	Tablica obiektów ID grup użytkowników Zabbix. Każdy obiekt ma następujące właściwości: `usrgrpid` - (`ID`) ID grupy użytkowników Zabbix, która ma zostać przypisana użytkownikowi. Jeśli dopasowanych zostanie wiele mapowań grup provisioningu, użytkownikowi zostaną przypisane grupy użytkowników Zabbix ze wszystkich dopasowanych mapowań. Zachowanie właściwości: *- wymagane*

userdirectory.create

Opis

`object userdirectory.create(object/array userDirectory)`

Ta metoda umożliwia tworzenie nowych katalogów użytkowników.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*.

Parametry

(`object/array`) Katalogi użytkowników do utworzenia.

Metoda akceptuje katalogi użytkowników ze standardowymi właściwościami katalogu użytkownika.

Zwracane wartości

(`object`) Zwraca obiekt zawierający identyfikatory utworzonych katalogów użytkowników we właściwości `userdirectoryids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych katalogów użytkowników.

Przykłady

Tworzenie katalogu użytkowników

Utwórz katalog użytkowników, aby uwierzytelniać użytkowników za pomocą StartTLS przez LDAP. Pamiętaj, że aby uwierzytelniać użytkowników przez LDAP, musi być włączone uwierzytelnianie LDAP.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.create",
    "params": {
        "idp_type": "1",
        "name": "LDAP API server #1",
        "host": "ldap://local.ldap",
        "port": "389",
        "base_dn": "ou=Users,dc=example,dc=org",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "bind_password": "ldapsecretpassword",
        "search_attribute": "uid",
        "start_tls": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
```

```
    "id": 1
}
```

Tworzenie katalogu użytkowników (z włączonym udostępnianiem JIT)

Utwórz katalog użytkowników, aby uwierzytelniać użytkowników przez LDAP (z włączonym udostępnianiem JIT). Pamiętaj, że aby uwierzytelniać użytkowników przez LDAP, należy włączyć uwierzytelnianie LDAP.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.create",
    "params": {
            "idp_type": "1",
            "name": "AD server",
            "provision_status": "1",
            "description": "",
            "host": "host.example.com",
            "port": "389",
            "base_dn": "DC=zbx,DC=local",
            "search_attribute": "sAMAccountName",
            "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
            "start_tls": "0",
            "search_filter": "",
            "group_basedn": "OU=Zabbix,DC=zbx,DC=local",
            "group_name": "CN",
            "group_member": "member",
            "group_filter": "(%{groupattr}=CN=%{ref},OU=Users,OU=Zabbix,DC=zbx,DC=local)",
            "group_membership": "",
            "user_username": "givenName",
            "user_lastname": "sn",
            "user_ref_attr": "CN",
            "provision_media": [
                {
                    "name": "example.com",
                    "mediatypeid": "1",
                    "attribute": "user@example.com"
                }
            ],
            "provision_groups": [
                {
                    "name": "*",
                    "roleid": "4",
                    "user_groups": [
                        {
                            "usrgrpid": "8"
                        }
                    ]
                },
                {
                    "name": "Zabbix administrators",
                    "roleid": "2",
                    "user_groups": [
                        {
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        }
                    ]
                }
            ]
        },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CUserDirectory::create() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.delete

Opis

`object userdirectory.delete(array userDirectoryIds)`

Ta metoda umożliwia usuwanie katalogów użytkowników. Katalog użytkowników nie może zostać usunięty, jeśli jest bezpośrednio używany przez co najmniej jedną grupę użytkowników.
 Domyślny katalog użytkowników LDAP nie może zostać usunięty, gdy `authentication.ldap_configured` jest ustawione na 1 lub gdy pozostało więcej katalogów użytkowników.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*.

Parametry

`(tablica)` Identyfikatory katalogów użytkownika do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych katalogów użytkownika w ramach właściwości `userdirectoryids`.

Przykłady

Usuwanie wielu katalogów użytkowników

Usuń dwa katalogi użytkowników.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "userdirectory.delete",
"params": [
"2",
"12"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"userdirectoryids": [
"2",
"12"
]
},
```

```
"id": 1
}
```

Źródło

CUserDirectory::delete() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.get

Opis

`integer/array userdirectory.get(object parameters)`

Ta metoda umożliwia pobieranie katalogów użytkowników zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
userdirectoryids	ID/array	Zwracaj tylko katalogi użytkowników o podanych ID.
selectUsrgrps	query	Zwróć właściwość `usrgrps` z grupami użytkowników powiązanymi z katalogiem użytkowników. Obsługuje `count`.
selectProvisionMedia	query	Zwróć właściwość `provision_media` z mapowaniami typów mediów powiązanymi z katalogiem użytkowników.
selectProvisionGroups	query	Zwróć właściwość `provision_groups` z mapowaniami grup provisioningowych powiązanymi z katalogiem użytkowników.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `name`.
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami pojedyncza wartość lub tablica wartości. Obsługiwane właściwości: `userdirectoryid`, `idp_type`, `provision_status`.
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługiwane właściwości: `name`, `description`. Katalog użytkowników typu SAML będzie miał pustą wartość zarówno dla właściwości `name`, jak i `description`. Obie właściwości można zmienić za pomocą operacji `userdirectory.update`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	

Parameter	Type	Description
startSearch	boolean	

Wartości zwracane

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie katalogów użytkowników

Pobierz wszystkie katalogi użytkowników z dodatkowymi właściwościami wyświetlającymi mapowania typów mediów oraz mapowania grup provisioningu powiązane z każdym katalogiem użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.get",
    "params": {
        "output": "extend",
        "selectProvisionMedia": "extend",
        "selectProvisionGroups": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
       {
            "userdirectoryid": "1",
            "idp_type": "2",
            "name": "",
            "provision_status": "1",
            "description": "",
            "group_name": "groups",
            "user_username": "",
            "user_lastname": "",
            "idp_entityid": "http://example.com/simplesaml/saml2/idp/metadata.php",
            "sso_url": "http://example.com/simplesaml/saml2/idp/SSOService.php",
            "slo_url": "",
            "username_attribute": "uid",
            "sp_entityid": "zabbix",
            "nameid_format": "",
            "sign_messages": "0",
            "sign_assertions": "0",
            "sign_authn_requests": "0",
            "sign_logout_requests": "0",
            "sign_logout_responses": "0",
            "encrypt_nameid": "0",
            "encrypt_assertions": "0",
            "scim_status": "1",
            "provision_media": [
                {
                    "userdirectory_mediaid": "1",
                    "name": "example.com",
                    "mediatypeid": "1",
                    "attribute": "user@example.com",
                    "active": "0",
                    "severity": "63",
```

```
                "period": "1-7,00:00-24:00"
            }
        ],
        "provision_groups": [
            {
                "name": "*",
                "roleid": "1",
                "user_groups": [
                    {
                        "usrgrpid": "13"
                    }
                ]
            }
        ]
    },
    {
        "userdirectoryid": "2",
        "idp_type": "1",
        "name": "AD server",
        "provision_status": "1",
        "description": "",
        "host": "host.example.com",
        "port": "389",
        "base_dn": "DC=zbx,DC=local",
        "search_attribute": "sAMAccountName",
        "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
        "start_tls": "0",
        "search_filter": "",
        "group_basedn": "OU=Zabbix,DC=zbx,DC=local",
        "group_name": "CN",
        "group_member": "member",
        "group_filter": "(%{groupattr}=CN=%{ref},OU=Users,OU=Zabbix,DC=zbx,DC=local)",
        "group_membership": "",
        "user_username": "givenName",
        "user_lastname": "sn",
        "user_ref_attr": "CN",
        "provision_media": [
            {
                "userdirectory_mediaid": "2",
                "name": "example.com",
                "mediatypeid": "1",
                "attribute": "user@example.com",
                "active": "0",
                "severity": "63",
                "period": "1-7,00:00-24:00"
            }
        ],
        "provision_groups": [
            {
                "name": "*",
                "roleid": "4",
                "user_groups": [
                    {
                        "usrgrpid": "8"
                    }
                ]
            },
            {
                "name": "Zabbix administrators",
                "roleid": "2",
                "user_groups": [
                    {
```

```
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        }
                    ]
                }
            ]
        },
        {
            "userdirectoryid": "3",
            "idp_type": "1",
            "name": "LDAP API server #1",
            "provision_status": "0",
            "description": "",
            "host": "ldap://local.ldap",
            "port": "389",
            "base_dn": "ou=Users,dc=example,dc=org",
            "search_attribute": "uid",
            "bind_dn": "cn=ldap_search,dc=example,dc=org",
            "start_tls": "1",
            "search_filter": "",
            "group_basedn": "",
            "group_name": "",
            "group_member": "",
            "group_filter": "",
            "group_membership": "",
            "user_username": "",
            "user_lastname": "",
            "user_ref_attr": "",
            "provision_media": [],
            "provision_groups": []
        }
    ],
    "id": 1
}
```


Zobacz także

- Grupa użytkowników

Źródło

CUserDirectory::get() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.test

Opis

`object userdirectory.test(array userDirectory)`

Ta metoda umożliwia przetestowanie ustawień połączenia katalogu użytkowników.

Note:
Ta metoda umożliwia również sprawdzenie, jakie skonfigurowane dane odpowiadają ustawieniom katalogu użytkowników na potrzeby provisioningu użytkowników (np. jaka rola użytkownika, grupy użytkowników i media użytkownika zostaną przypisane użytkownikowi). Dla tego typu testu żądanie API powinno zostać wykonane dla katalogu użytkowników, który ma parametr `provision_status` ustawiony na enabled.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*.

Parametry

(object) Właściwości katalogu użytkowników.

Ponieważ API userdirectory.get nie zwraca pola bind_password, należy podać userdirectoryid i/lub bind_password.
Oprócz standardowych właściwości katalogu użytkowników, metoda akceptuje następujące parametry.

Parametr	Type	Opis
test_username	string	Nazwa użytkownika do przetestowania w katalogu użytkowników.
test_password	string	Hasło powiązane z nazwą użytkownika do przetestowania w katalogu użytkowników.

Zwracane wartości

(bool) Zwraca true w przypadku powodzenia.

Przykłady

Testuj katalog użytkowników dla istniejącego użytkownika

Przetestuj katalog użytkowników „3” dla „user1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "3",
        "host": "127.0.0.1",
        "port": "389",
        "base_dn": "ou=Users,dc=example,dc=org",
        "search_attribute": "uid",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "bind_password": "password",
        "test_username": "user1",
        "test_password": "password"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Testuj katalog użytkowników dla nieistniejącego użytkownika

Przetestuj katalog użytkowników „3” dla nieistniejącego użytkownika „user2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "3",
        "host": "127.0.0.1",
        "port": "389",
        "base_dn": "ou=Users,dc=example,dc=org",
        "search_attribute": "uid",
        "bind_dn": "cn=ldap_search,dc=example,dc=org",
        "test_username": "user2",
        "test_password": "password"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32500,
        "message": "Application error.",
        "data": "Incorrect user name or password or account is temporarily blocked."
    },
    "id": 1
}
```

Testuj katalog użytkowników na potrzeby provisioningu użytkowników

Przetestuj katalog użytkowników „3”, aby sprawdzić, które skonfigurowane dane odpowiadają ustawieniom katalogu użytkowników dla provisioningu użytkownika „user3” (np. jaka rola użytkownika, grupy użytkowników i media użytkownika zostaną przypisane temu użytkownikowi).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.test",
    "params": {
        "userdirectoryid": "2",
        "host": "host.example.com",
        "port": "389",
        "base_dn": "DC=zbx,DC=local",
        "search_attribute": "sAMAccountName",
        "bind_dn": "CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local",
        "test_username": "user3",
        "test_password": "password"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "username": "user3",
        "name": "John",
        "surname": "Doe",
        "medias": [],
        "usrgrps": [
            {
                "usrgrpid": "8"
            },
            {
                "usrgrpid": "7"
            }
        ],
        "roleid": "2",
        "userdirectoryid": "2"
    },
    "id": 1
}
```

Źródło

CUserDirectory::test() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.update

Opis

`object userdirectory.update(object/array userDirectory)`

Ta metoda umożliwia aktualizację istniejących katalogów użytkowników.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*.

Parametry

`(object/array)` Właściwości katalogu użytkowników, które mają zostać zaktualizowane.

Właściwość `userdirectoryid` musi być zdefiniowana dla każdego katalogu użytkowników, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych katalogów użytkowników we właściwości `userdirectoryids`.

Przykłady

Aktualizacja hasła bind dla katalogu użytkowników

Ustaw nowe hasło bind dla katalogu użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "3",
        "bind_password": "newldappassword"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Zaktualizuj mapowania dla katalogu użytkowników

Zaktualizuj mapowania grup udostępniania oraz mapowania typów mediów dla katalogu użytkowników „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "2",
        "provision_media": [
                {
                    "userdirectory_mediaid": "2"
                }
            ],
            "provision_groups": [
                {
                    "name": "Zabbix administrators",
                    "roleid": "2",
                    "user_groups": [
                        {
```

```
                            "usrgrpid": "7"
                        },
                        {
                            "usrgrpid": "8"
                        },
                        {
                            "usrgrpid": "11"
                        }
                    ]
                }
            ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CUserDirectory::update() w *ui/include/classes/api/services/CUserDirectory.php*.

Konfiguracja

Ta klasa jest przeznaczona do eksportowania i importowania danych konfiguracji Zabbix.

Dostępne metody:

- configuration.export - eksport konfiguracji
- configuration.import - import konfiguracji
- configuration.importcompare - porównanie pliku importu z bieżącymi elementami systemu

configuration.export

Opis

`string configuration.export(object parameters)`

Ta metoda umożliwia eksport danych konfiguracyjnych jako serializowanego ciągu znaków.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry definiujące obiekty do wyeksportowania oraz format do użycia.

Parameter	Type	Description
format	string	Format, w jakim dane muszą zostać wyeksportowane. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON; `raw` - nieprzetworzona tablica PHP. Zachowanie parametru: - *wymagany*
prettyprint	boolean	Sprawia, że wynik jest bardziej czytelny dla człowieka poprzez dodanie wcięć. Możliwe wartości: `true` - dodaj wcięcia; `false` - *(domyślnie)* nie dodawaj wcięć.
options	object	Obiekty do wyeksportowania. Obiekt `options` ma następujące parametry: `host_groups` - `(array)` identyfikatory grup hostów do eksportu; `hosts` - `(array)` identyfikatory hostów do eksportu; `images` - `(array)` identyfikatory obrazów do eksportu; `maps` - `(array)` identyfikatory map do eksportu; `mediaTypes` - `(array)` identyfikatory typów mediów do eksportu; `template_groups` - `(array)` identyfikatory grup szablonów do eksportu; `templates` - `(array)` identyfikatory szablonów do eksportu. Użytkownicy typu *Admin* i *User* mogą eksportować tylko te obiekty, dla których mają uprawnienia *read-only* lub *read-write* permission, a także obrazy, ale nie typy mediów. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(string)` Zwraca serializowany ciąg znaków zawierający żądane dane konfiguracyjne.

Przykłady

Eksportowanie szablonu

Wyeksportuj konfigurację szablonu ”10571” jako ciąg XML.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.export",
    "params": {
        "options": {
            "templates": [
                "10571"
            ]
        },
        "format": "xml"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>7.4</version><template_
    "id": 1
}
```


Źródło

CConfiguration::export() w *ui/include/classes/api/services/CConfiguration.php*.

configuration.import

Opis

`boolean configuration.import(object parameters)`

Ta metoda umożliwia import danych konfiguracyjnych z serializowanego ciągu znaków.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające dane do importu oraz reguły określające, jak dane powinny zostać obsłużone.

Parameter	Type	Description
format	string	Format serializowanego ciągu. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON. Zachowanie parametru: - *required*
source	string	Serializowany ciąg zawierający dane konfiguracyjne. Zachowanie parametru: - *required*
rules	object	Reguły określające sposób importu nowych i istniejących obiektów. Użytkownicy typu *Admin* mogą importować tylko te obiekty, dla których mają uprawnienie *read-write* permission, a także mapy. Na przykład host i jego elementy (pozycje, wyzwalacze, wykresy itd.) mogą zostać zaimportowane tylko wtedy, gdy grupa użytkowników, do której należy użytkownik, ma uprawnienia do grupy hostów, do której będzie należał importowany host. Użytkownicy typu *Admin* nie mogą importować obrazów ani typów mediów. Parametr `rules` został szczegółowo opisany w tabeli poniżej. Zachowanie parametru: - *required*

Attention:
Jeśli nie zostaną podane żadne reguły, konfiguracja nie zostanie zaktualizowana.

Obiekt `rules` obsługuje następujące parametry.

Parameter	Type	Description
discoveryRules	object	Reguły importu reguł LLD. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe reguły LLD; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące reguły LLD zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, reguły LLD nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
graphs	object	Reguły importu wykresów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wykresy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wykresy zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wykresy nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
host_groups	object	Reguły importu grup hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy hostów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy hostów zostaną zaktualizowane; domyślnie: `false`.
template_groups	object	Reguły importu grup szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy szablonów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy szablonów zostaną zaktualizowane; domyślnie: `false`.
hosts	object	Reguły importu hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe hosty; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące hosty zostaną zaktualizowane; domyślnie: `false`.
httptests	object	Reguły importu scenariuszy web. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe scenariusze web; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące scenariusze web zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, scenariusze web nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
images	object	Reguły importu obrazów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe obrazy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące obrazy zostaną zaktualizowane; domyślnie: `false`.

Parameter	Type	Description
items	object	Reguły importu pozycji. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe pozycje; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące pozycje zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, pozycje nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
maps	object	Reguły importu map. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe mapy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy zostaną zaktualizowane; domyślnie: `false`.
mediaTypes	object	Reguły importu typów mediów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe typy mediów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące typy mediów zostaną zaktualizowane; domyślnie: `false`.
templateLinkage	object	Reguły importu powiązań szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, szablony, które nie są powiązane z importowanym hostem lub szablonem, ale są obecne w importowanych danych, zostaną powiązane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, szablony powiązane z importowanym hostem lub szablonem, ale nieobecne w importowanych danych, zostaną odłączone bez usuwania elementów (pozycji, wyzwalaczy itd.) odziedziczonych po odłączonych szablonach; domyślnie: `false`.
templates	object	Reguły importu szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe szablony; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące szablony zostaną zaktualizowane; domyślnie: `false`.
templateDashboards	object	Reguły importu dashboardów szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe dashboardy szablonów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące dashboardy szablonów zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, dashboardy szablonów nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Parameter	Type	Description
triggers	object	Reguły importu wyzwalaczy. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wyzwalacze; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wyzwalacze zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wyzwalacze nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
valueMaps	object	Reguły importu map wartości hosta lub szablonu. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe mapy wartości; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy wartości zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, mapy wartości nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Zwracane wartości

`(boolean)` Zwraca `true` jeżeli importowanie się powiodło.

Przykłady

Importowanie szablonu

Zaimportuj konfigurację szablonu zawartą w ciągu XML. Jeśli w ciągu XML brakuje jakichkolwiek pozycji lub wyzwalaczy, zostaną one usunięte z bazy danych, a wszystko inne pozostanie bez zmian.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.import",
    "params": {
        "format": "xml",
        "rules": {
            "templates": {
                "createMissing": true,
                "updateExisting": true
            },
            "items": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "triggers": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "valueMaps": {
                "createMissing": true,
                "updateExisting": false
            }
        },
        "source": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>7.4</version><templ
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Źródło

CConfiguration::import() w *ui/include/classes/api/services/CConfiguration.php*.

configuration.importcompare

Opis

`array configuration.importcompare(object parameters)`

Ta metoda umożliwia porównanie pliku importu z bieżącymi elementami systemu i pokazuje, co zostanie zmienione, jeśli ten plik importu zostanie zaimportowany.

noteclassic Metoda ta jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników. :::

Parametry

`(object)` Parametry zawierające możliwe dane do zaimportowania oraz reguły określające sposób ich przetwarzania.

Parameter	Type	Description
format	string	Format serializowanego ciągu. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON. Zachowanie parametru: - *required*
source	string	Serializowany ciąg zawierający dane konfiguracyjne. Zachowanie parametru: - *required*
rules	object	Reguły określające sposób porównywania nowych i istniejących obiektów. Użytkownicy typu *Admin* i *User* mogą porównywać nowe obiekty tylko z istniejącymi, dla których mają uprawnienia *read-only* lub *read-write* permission. Parametr `rules` jest szczegółowo opisany w tabeli poniżej. Zachowanie parametru: - *required*

Attention:
Jeśli nie zostaną podane żadne reguły, nie będzie nic do zaktualizowania, a wynik będzie pusty.

Note:
Porównanie będzie wykonywane tylko dla grup hostów i szablonów. Wyzwalacze i wykresy będą porównywane tylko dla importowanych szablonów, wszystkie pozostałe zostaną uznane za ”new”.

Obiekt `rules` obsługuje następujące parametry.

Parameter	Type	Description
discoveryRules	object	Reguły określające sposób importu reguł LLD. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe reguły LLD; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące reguły LLD zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, reguły LLD nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
graphs	object	Reguły określające sposób importu wykresów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wykresy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wykresy zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wykresy nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
host_groups	object	Reguły określające sposób importu grup hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy hostów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy hostów zostaną zaktualizowane; domyślnie: `false`.
template_groups	object	Reguły określające sposób importu grup szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy szablonów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy szablonów zostaną zaktualizowane; domyślnie: `false`.
hosts	object	Reguły określające sposób importu hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe hosty; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące hosty zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie wpływa na wynik. Jest dozwolony wyłącznie ze względów spójności z `configuration.import`.
httptests	object	Reguły określające sposób importu scenariuszy webowych. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe scenariusze webowe; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące scenariusze webowe zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, scenariusze webowe nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Parameter	Type	Description
images	object	Reguły określające sposób importu obrazów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe obrazy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące obrazy zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie wpływa na wynik. Jest dozwolony wyłącznie ze względów spójności z `configuration.import`.
items	object	Reguły określające sposób importu pozycji. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe pozycje; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące pozycje zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, pozycje nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
maps	object	Reguły określające sposób importu map. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe mapy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie wpływa na wynik. Jest dozwolony wyłącznie ze względów spójności z `configuration.import`.
mediaTypes	object	Reguły określające sposób importu typów mediów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe typy mediów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące typy mediów zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie wpływa na wynik. Jest dozwolony wyłącznie ze względów spójności z `configuration.import`.
templateLinkage	object	Reguły określające sposób importu powiązań szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, szablony, które nie są powiązane z importowanym hostem lub szablonem, ale są obecne w importowanych danych, zostaną powiązane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, szablony, które są powiązane z importowanym hostem lub szablonem, ale nie są obecne w importowanych danych, zostaną odłączone bez usuwania encji (pozycji, wyzwalaczy itp.) odziedziczonych z odłączonych szablonów; domyślnie: `false`.
templates	object	Reguły określające sposób importu szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe szablony; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące szablony zostaną zaktualizowane; domyślnie: `false`.

Parameter	Type	Description
templateDashboards	object	Reguły określające sposób importu pulpitów szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe pulpity szablonów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące pulpity szablonów zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, pulpity szablonów nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
triggers	object	Reguły określające sposób importu wyzwalaczy. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wyzwalacze; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wyzwalacze zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wyzwalacze nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
valueMaps	object	Reguły określające sposób importu map wartości hosta lub szablonu. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe mapy wartości; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy wartości zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, mapy wartości nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Zwracane wartości

`(array)` Zwraca tablicę ze zmianamy w konfiguracji, które zostaną wprowadzone.

Przykłady

Porównanie importu szablonu

Porównaj szablon zawarty w ciągu XML z bieżącymi elementami systemu i pokaż, co zostanie zmienione, jeśli ten szablon zostanie zaimportowany.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.importcompare",
    "params": {
        "format": "xml",
        "rules": {
            "discoveryRules": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "graphs": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "host_groups": {
                "createMissing": true,
                "updateExisting": true
            },
```

```
            "template_groups": {
                "createMissing": true,
                "updateExisting": true
            },
            "httptests": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "items": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "templateLinkage": {
                "createMissing": true,
                "deleteMissing": true
            },
            "templates": {
                "createMissing": true,
                "updateExisting": true
            },
            "templateDashboards": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "triggers": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            },
            "valueMaps": {
                "createMissing": true,
                "updateExisting": true,
                "deleteMissing": true
            }
        },
        "source": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>7.4</version><templ
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templates": {
            "updated": [
                {
                    "before": {
                        "uuid": "5aef0444a82a4d8cb7a95dc4c0c85330",
                        "template": "New template",
                        "name": "New template",
                        "groups": [
                            {
                                "name": "Templates"
                            }
                        ]
                    },
                    "after": {
                        "uuid": "5aef0444a82a4d8cb7a95dc4c0c85330",
```

```
            "template": "New template",
            "name": "New template",
            "groups": [
                {
                    "name": "Templates"
                }
            ]
        },
        "items": {
            "added": [
                {
                    "after": {
                        "uuid": "648006da5971424ead0c47ddbbf1ea2e",
                        "name": "CPU utilization",
                        "key": "system.cpu.util",
                        "value_type": "FLOAT",
                        "units": "%"
                    },
                    "triggers": {
                        "added": [
                            {
                                "after": {
                                    "uuid": "736225012c534ec480c2a66a91322ce0",
                                    "expression": "avg(/New template/system.cpu.util,3m)>70",
                                    "name": "CPU utilization too high on 'New host' for 3 minu
                                    "priority": "WARNING"
                                }
                            }
                        ]
                    }
                }
            ],
            "removed": [
                {
                    "before": {
                        "uuid": "6805d4c39a624a8bab2cc8ab63df1ab3",
                        "name": "CPU load",
                        "key": "system.cpu.load",
                        "value_type": "FLOAT"
                    },
                    "triggers": {
                        "removed": [
                            {
                                "before": {
                                    "uuid": "ab4c2526c2bc42e48a633082255ebcb3",
                                    "expression": "avg(/New template/system.cpu.load,3m)>2",
                                    "name": "CPU load too high on 'New host' for 3 minutes",
                                    "priority": "WARNING"
                                }
                            }
                        ]
                    }
                }
            ],
            "updated": [
                {
                    "before": {
                        "uuid": "7f1e6f1e48aa4a128e5b6a958a5d11c3",
                        "name": "Zabbix agent ping",
                        "key": "agent.ping"
                    },
                    "after": {
```

```
                                    "uuid": "7f1e6f1e48aa4a128e5b6a958a5d11c3",
                                    "name": "Zabbix agent ping",
                                    "key": "agent.ping",
                                    "delay": "3m"
                                }
                            }
                        ]
                    }
                }
            ]
        }
    },
    "id": 1
}
```


Źródło

CConfiguration::importcompare() w *ui/include/classes/api/services/CConfiguration.php*.

Konserwacja

Ta klasa jest przeznaczona do pracy z okresami konserwacji.

Odwołania do obiektów:

- Konserwacja
- Okres czasu
- Tag problemu

Dostępne metody:

- maintenance.create - utwórz nowe okresy konserwacji
- maintenance.delete - usuń okresy konserwacji
- maintenance.get - pobierz okresy konserwacji
- maintenance.update - zaktualizuj okresy konserwacji

Obiekt konserwacji

Następujące obiekty są bezpośrednio powiązane z API `maintenance`.

Konserwacja

Obiekt konserwacji ma następujące właściwości.

Właściwość	Type	Opis
maintenanceid	ID	ID konserwacji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa konserwacji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
active_since	timestamp	Czas, od którego konserwacja staje się aktywna (włącznie). Podana wartość zostanie zaokrąglona w dół do pełnych minut. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Type	Opis
active_till	timestamp	Czas, do którego konserwacja pozostaje aktywna (z wyłączeniem tej chwili). Podana wartość zostanie zaokrąglona w dół do pełnych minut. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	string	Opis konserwacji.
maintenance_type	integer	Typ konserwacji. Możliwe wartości: 0 - *(domyślnie)* ze zbieraniem danych; 1 - bez zbierania danych.
tags_evaltype	integer	Metoda oceny tagów problemów. Możliwe wartości: 0 - *(domyślnie)* I/Lub; 2 - Lub.

Okres czasu

Obiekt okresu czasu służy do definiowania okresów, w których konserwacja ma wejść w życie. Ma następujące właściwości.

Property	Type	Description
period	integer	Czas trwania okresu konserwacji w sekundach. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Możliwy zakres wartości: 300 - 86399940. Domyślnie: 3600.
timeperiod_type	integer	Typ okresu czasu. Możliwe wartości: 0 - *(domyślnie)* tylko jeden raz; 2 - codziennie; 3 - co tydzień; 4 - co miesiąc.
start_date	timestamp	Data, kiedy okres konserwacji ma wejść w życie. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Domyślnie: bieżąca data. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”one time only”
start_time	integer	Pora dnia, o której konserwacja rozpoczyna się, wyrażona w sekundach. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Domyślnie: 0. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”daily”, ”weekly” lub ”monthly”

Property	Type	Description
every	integer	Dla okresów dziennych i tygodniowych `every` określa interwały dni lub tygodni, z jakimi konserwacja ma wejść w życie. Wartość domyślna, jeśli `timeperiod_type` jest ustawione na ”daily” lub ”weekly”: 1. Dla okresów miesięcznych, gdy ustawiono `day`, właściwość `every` określa dzień miesiąca, w którym konserwacja ma wejść w życie. Wartość domyślna, jeśli `timeperiod_type` jest ustawione na ”monthly” i ustawiono `day`: 1. Dla okresów miesięcznych, gdy ustawiono `dayofweek`, właściwość `every` określa tydzień miesiąca, w którym konserwacja ma wejść w życie. Możliwe wartości, jeśli `timeperiod_type` jest ustawione na ”monthly” i ustawiono `dayofweek`: 1 - *(domyślnie)* pierwszy tydzień; 2 - drugi tydzień; 3 - trzeci tydzień; 4 - czwarty tydzień; 5 - ostatni tydzień. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”daily”, ”weekly” lub ”monthly”
dayofweek	integer	Dni tygodnia, w których konserwacja ma wejść w życie. Możliwe wartości bitmapy: 1 - poniedziałek; 2 - wtorek; 4 - środa; 8 - czwartek; 16 - piątek; 32 - sobota; 64 - niedziela. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 21 dla poniedziałku, środy i piątku). Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”weekly” lub jeśli `timeperiod_type` jest ustawione na ”monthly” i `day` nie jest ustawione
day	integer	Dzień miesiąca, w którym konserwacja ma wejść w życie. Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”monthly” i `dayofweek` nie jest ustawione

Property	Type	Description
month	integer	Miesiące, w których konserwacja ma wejść w życie. Możliwe wartości bitmapy: 1 - styczeń; 2 - luty; 4 - marzec; 8 - kwiecień; 16 - maj; 32 - czerwiec; 64 - lipiec; 128 - sierpień; 256 - wrzesień; 512 - październik; 1024 - listopad; 2048 - grudzień. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 585 dla stycznia, kwietnia, lipca i października). Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”monthly”

Znacznik problemu

Obiekt znacznika problemu służy do definiowania, które problemy mają być tłumione, gdy konserwacja stanie się aktywna. Znaczniki można określić tylko wtedy, gdy `maintenance_type` obiektu Maintenance object jest ustawione na „with data collection”. Ma on następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa znacznika problemu. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek operatora. Możliwe wartości: 0 - Równa się; 2 - *(domyślnie)* Zawiera.
value	string	Wartość znacznika problemu.

maintenance.create

Opis

`object maintenance.create(object/array maintenances)`

Ta metoda pozwala na tworzenie nowych konserwacji.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Okna konserwacyjne do utworzenia.

Oprócz standardowych właściwości okna konserwacyjnego, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy hostów, które zostaną objęte konserwacją. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *wymagany*, jeśli `hosts` nie jest ustawione
hosts	object/array	Hosty, które zostaną objęte konserwacją. Hosty muszą mieć zdefiniowaną tylko właściwość `hostid`. Zachowanie parametru: - *wymagany*, jeśli `groups` nie jest ustawione
timeperiods	object/array	Okresy czasu konserwacji. Zachowanie parametru: - *wymagany*
tags	object/array	Tagi problemów. Określają, które problemy mają być tłumione. Jeśli nie podano tagów, wszystkie problemy hostów z aktywnej konserwacji będą tłumione. Zachowanie parametru: - *obsługiwany*, jeśli `maintenance_type` obiektu Maintenance object jest ustawione na "with data collection"

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych konserwacji w ramach właściwości `maintenanceids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych konserwacji.

Przykłady

Tworzenie okresu konserwacji

Utwórz okres konserwacji ze zbieraniem danych dla grupy hostów o ID „2" oraz z tagami problemów **service:mysqld** i **error**. Musi on być aktywny od 17.03.2026 do 17.03.2027, wchodzić w życie w każdą niedzielę o 18:00 i trwać jedną godzinę.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.create",
    "params": {
        "name": "Sunday maintenance",
        "active_since": 1773720240,
        "active_till": 1805256240,
        "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": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Okres czasu

Źródło

CMaintenance::create() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.delete

Opis

`object maintenance.delete(array maintenanceIds)`

Ta metoda pozwala na usuwanie okresów konserwacji.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` Identyfikatory okresów konserwacji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych okresów konserwacji we właściwości `maintenanceids`.

Przykłady

Usuwanie wielu okresów konserwacji

Usuń dwa okresy konserwacji.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "maintenance.delete",
"params": [
"3",
"1"
],
"id": 1
}
```

Odpowiedź:


```
{
"jsonrpc": "2.0",
"result": {
"maintenanceids": [
"3",
"1"
]
},
"id": 1
}
```


Źródło

CMaintenance::delete() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.get

Opis

`integer/array maintenance.get(object parameters)`

Ta metoda umożliwia pobieranie okresów konserwacji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwraca tylko okresy konserwacji przypisane do podanych grup hostów.
hostids	ID/array	Zwraca tylko okresy konserwacji przypisane do podanych hostów.
maintenanceids	ID/array	Zwraca tylko okresy konserwacji o podanych ID.
selectHostGroups	query	Zwraca właściwość `hostgroups` zawierającą grupy hostów przypisane do okresu konserwacji.
selectHosts	query	Zwraca właściwość `hosts` zawierającą hosty przypisane do okresu konserwacji.
selectTags	query	Zwraca właściwość `tags` zawierającą tagi problemów okresu konserwacji.
selectTimeperiods	query	Zwraca właściwość `timeperiods` zawierającą okresy czasu okresu konserwacji.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `maintenanceid`, `name`, `maintenance_type`, `active_since`, `active_till`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie okresów konserwacji

Pobierz wszystkie skonfigurowane okresy konserwacji oraz dane o przypisanych grupach hostów, zdefiniowanych okresach czasu i tagach problemów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.get",
    "params": {
        "output": "extend",
        "selectHostGroups": "extend",
        "selectTimeperiods": "extend",
        "selectTags": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "maintenanceid": "3",
            "name": "Sunday maintenance",
            "maintenance_type": "0",
            "description": "",
            "active_since": "1358844540",
            "active_till": "1390466940",
            "tags_evaltype": "0",
            "hostgroups": [
                {
                    "groupid": "4",
                    "name": "Zabbix servers",
                    "flags": "0",
                    "uuid": "6f6799aa69e844b4b3918f779f2abf08"
                }
            ],
            "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
}
```

Zobacz także

- Host
- Grupa hostów
- Okres czasu

Źródło

CMaintenance::get() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.update

Opis

`object maintenance.update(object/array maintenances)`

Ta metoda umożliwia aktualizację istniejących konserwacji.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Właściwości maintenance do zaktualizowania.

Właściwość `maintenanceid` musi być zdefiniowana dla każdego maintenance, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości maintenance, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy hostów zastępujące bieżące grupy. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *wymagane*, jeśli `hosts` nie jest ustawione
hosts	object/array	Hosty zastępujące bieżące hosty. Hosty muszą mieć zdefiniowaną tylko właściwość `hostid`. Zachowanie parametru: - *wymagane*, jeśli `groups` nie jest ustawione
timeperiods	object/array	Okresy czasu maintenance zastępujące bieżące okresy.
tags	object/array	Tagi problemów zastępujące bieżące tagi. Zachowanie parametru: - *obsługiwane*, jeśli `maintenance_type` obiektu Maintenance object jest ustawione na "with data collection"

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych konserwacji w ramach właściwości `maintenanceids`.

Przykłady

Przypisywanie różnych hostów

Zastąp hosty aktualnie przypisane do konserwacji dwoma innymi.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "maintenance.update",
"params": {
"maintenanceid": "3",
"hosts": [
{"hostid": "10085"},
{"hostid": "10084"}
]
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"maintenanceids": [
"3"
]
},
"id": 1
}
```

Zobacz także

- Okres czasu

Źródło

CMaintenance::update() w *ui/include/classes/api/services/CMaintenance.php*.

Kontrola wykrywania

Ta klasa została zaprojektowana do pracy z kontrolami wykrywania.

Odwołania do obiektów:

- Kontrola wykrywania

Dostępne metody:

- dcheck.get - pobieranie kontroli wykrywania

Obiekt sprawdzania odkrycia

Następujące obiekty są bezpośrednio powiązane z API `dcheck`.

Kontrola wykrywania

Obiekt kontroli wykrywania definiuje konkretną kontrolę wykonywaną przez regułę wykrywania sieci. Ma następujące właściwości.

Właściwość	Typ	Opis
dcheckid	ID	ID kontroli wykrywania.
druleid	ID	ID reguły wykrywania, do której należy ta kontrola.

Właściwość	Typ	Opis
key_	string	Klucz pozycji (jeśli `type` jest ustawione na ”Zabbix agent”) lub OID SNMP (jeśli `type` jest ustawione na ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”). Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”
ports	string	Jeden lub kilka zakresów portów do sprawdzenia, oddzielonych przecinkami. Domyślnie: 0. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SSH” (0), ”LDAP” (1), ”SMTP” (2), ”FTP” (3), ”HTTP” (4), ”POP” (5), ”NNTP” (6), ”IMAP” (7), ”TCP” (8), ”Zabbix agent” (9), ”SNMPv1 agent” (10), ”SNMPv2 agent” (11), ”SNMPv3 agent” (13), ”HTTPS” (14) lub ”Telnet” (15)
snmp_community	string	Społeczność SNMP. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”SNMPv1 agent” lub ”SNMPv2 agent”
snmpv3_authpassphrase	string	Hasło uwierzytelniania. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
snmpv3_authprotocol	integer	Protokół uwierzytelniania. Możliwe wartości: 0 - *(domyślnie)* MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
snmpv3_contextname	string	Nazwa kontekstu SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
snmpv3_privpassphrase	string	Hasło prywatności. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authPriv”

Właściwość	Typ	Opis
snmpv3_privprotocol	integer	Protokół prywatności. Możliwe wartości: 0 - *(domyślnie)* DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authPriv”
snmpv3_securitylevel	string	Poziom bezpieczeństwa. Możliwe wartości: 0 - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
snmpv3_securityname	string	Nazwa bezpieczeństwa. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
type	integer	Typ kontroli. Możliwe wartości: 0 - SSH; 1 - LDAP; 2 - SMTP; 3 - FTP; 4 - HTTP; 5 - POP; 6 - NNTP; 7 - IMAP; 8 - TCP; 9 - Zabbix agent; 10 - SNMPv1 agent; 11 - SNMPv2 agent; 12 - ICMP ping; 13 - SNMPv3 agent; 14 - HTTPS; 15 - Telnet. Zachowanie właściwości: - *wymagane*
uniq	integer	Czy używać tej kontroli jako kryterium unikalności urządzenia. Dla reguły wykrywania można skonfigurować tylko jedną unikalną kontrolę. Możliwe wartości: 0 - *(domyślnie)* nie używaj tej kontroli jako kryterium unikalności; 1 - użyj tej kontroli jako kryterium unikalności. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”

Właściwość	Typ	Opis
host_source	integer	Źródło nazwy hosta. Możliwe wartości: 1 - *(domyślnie)* DNS; 2 - IP; 3 - wartość wykrywania tej kontroli.
name_source	integer	Źródło widocznej nazwy. Możliwe wartości: 0 - *(domyślnie)* nie określono; 1 - DNS; 2 - IP; 3 - wartość wykrywania tej kontroli.
allow_redirect	integer	Zezwalaj na sytuację, w której cel sprawdzany za pomocą ICMP ping odpowiada z innego adresu IP. Możliwe wartości: 0 - *(domyślnie)* traktuj przekierowane odpowiedzi tak, jakby docelowy host był niedostępny (niepowodzenie); 1 - traktuj przekierowane odpowiedzi tak, jakby docelowy host był dostępny (sukces). Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”ICMP ping”

dcheck.get

Opis

`integer/array dcheck.get(object parameters)`

Metoda służy do pobierania danych wykrywania na podstawie podanych parametrów.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz [Role użytkownika] (/manual/web_interface/frontend_sections/users/user_roles) po więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dcheckids	ID/array	Zwraca tylko kontrole wykrywania o podanych identyfikatorach.
druleids	ID/array	Zwraca tylko kontrole wykrywania należące do podanych reguł wykrywania.
dserviceids	ID/array	Zwraca tylko kontrole wykrywania, które wykryły podane wykryte usługi.
selectDRules	query	Zwraca reguły wykrywania powiązane z kontrolami wykrywania.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dcheckid`, `druleid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parametr	Typ	Opis
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę otrzymanych obiektów, jeżeli parametr `countOutput` został użyty.

Przykłady

Pobieranie testów wykrywania dla reguły wykrywania

Pobierz wszystkie testy wykrywania, które są używane przez regułę ”6”.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "dcheck.get",
  "params": {
    "output": "extend",
    "dcheckids": "6"
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "dcheckid": "6",
      "druleid": "4",
      "type": "3",
      "key_": "",
      "snmp_community": "",
      "ports": "21",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "uniq": "0",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "snmpv3_contextname": "",
      "host_source": "1",
      "name_source": "0",
      "allow_redirect": "0"
    }
  ],
  "id": 1
}
```

Źródło

CDCheck::get() w *ui/include/classes/api/services/CDCheck.php*.

Korelacja

Ta klasa jest przeznaczona do pracy z korelacjami.

Odwołania do obiektów:

- Korelacja
- Operacja korelacji
- Filtr korelacji
  - Warunek filtra korelacji

Dostępne metody:

- correlation.create - tworzenie nowych korelacji
- correlation.delete - usuwanie korelacji
- correlation.get - pobieranie korelacji
- correlation.update - aktualizowanie korelacji

Obiekt Correlation =

Następujące obiekty są bezpośrednio powiązane z API `correlation`.

Korelacja

Obiekt korelacji ma następujące właściwości.

Właściwość	Typ	Opis
correlationid	ID	ID korelacji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa korelacji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	string	Opis korelacji.
status	integer	Określa, czy korelacja jest włączona czy wyłączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.

Operacja korelacji

Obiekt operacji korelacji definiuje operację, która zostanie wykonana po uruchomieniu korelacji. Ma następujące właściwości.

Właściwość	Type	Opis
type	integer	Typ operacji. Możliwe wartości: 0 - zamknij stare zdarzenia; 1 - zamknij nowe zdarzenie. Zachowanie właściwości: - *wymagane*

Filtr korelacji

Obiekt filtra korelacji definiuje zestaw warunków, które muszą zostać spełnione, aby wykonać skonfigurowane operacje korelacji. Ma następujące właściwości.

Właściwość	Typ	Opis
conditions	array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*
evaltype	integer	Metoda oceny warunków filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra. Wyrażenie zawiera identyfikatory odnoszące się do określonych warunków filtra za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odnoszące się do określonych warunków filtra za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „custom expression”

Warunek filtra korelacji

Obiekt warunku filtra korelacji definiuje określony warunek, który musi zostać sprawdzony przed uruchomieniem operacji korelacji.

Property	Type	Description
type	integer	Typ warunku. Możliwe wartości: 0 - tag starego zdarzenia; 1 - tag nowego zdarzenia; 2 - grupa hostów nowego zdarzenia; 3 - para tagów zdarzenia; 4 - wartość tagu starego zdarzenia; 5 - wartość tagu nowego zdarzenia. Zachowanie właściwości: - *wymagane*
tag	string	Tag zdarzenia (starego lub nowego). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „tag starego zdarzenia”, „tag nowego zdarzenia”, „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”

Property	Type	Description
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „grupa hostów nowego zdarzenia”
oldtag	string	Tag starego zdarzenia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „para tagów zdarzenia”
newtag	string	Tag starego zdarzenia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „para tagów zdarzenia”
value	string	Wartość tagu zdarzenia (starego lub nowego). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu ich pobrania.
operator	integer	Operator warunku. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „grupa hostów nowego zdarzenia”, „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod correlation.get i correlation.create.

Dla każdego typu warunku obsługiwane są następujące operatory i wartości.

Condition	Condition name	Supported operators	Expected value
2	Grupa hostów	=, <>	ID grupy hostów.
4	Wartość tagu starego zdarzenia	=, <>, like, not like	string
5	Wartość tagu nowego zdarzenia	=, <>, like, not like	string

correlation.create

Opis

`object correlation.create(object/array correlations)`

Ta metoda umożliwia tworzenie nowych korelacji.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Korelacje do utworzenia.

Oprócz standardowych właściwości korelacji, metoda akceptuje następujące parametry.

Parametr	Type	Opis
operations	array	Operacje korelacji do utworzenia dla korelacji. Zachowanie parametru: - *wymagany*
filter	object	Obiekt filtra korelacji dla korelacji. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID utworzonych korelacji we właściwości `correlationids`. Kolejność zwracanych ID są zgodne z kolejności przekazanych korelacji.

Przykłady

Utwórz nową korelację tagów zdarzeń

Utwórz korelację, używając metody oceny `AND/OR`, z jednym warunkiem i jedną operacją. Domyślnie korelacja będzie włączona.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.create",
    "params": {
        "name": "new event tag correlation",
        "filter": {
            "evaltype": 0,
            "conditions": [
                {
                    "type": 1,
                    "tag": "ok"
                }
            ]
        },
        "operations": [
            {
                "type": 0
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Używanie niestandardowego filtra wyrażenia

Utwórz korelację, która będzie używać niestandardowego warunku filtra. Identyfikatory formuły „A” lub „B” zostały wybrane arbitralnie. Typ warunku będzie „Grupa hostów” z operatorem „<>”.

Żądanie:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "correlation.create",
    "params": {
        "name": "new host group correlation",
        "description": "a custom description",
        "status": 0,
        "filter": {
            "evaltype": 3,
            "formula": "A or B",
            "conditions": [
                {
                    "type": 2,
                    "operator": 1,
                    "formulaid": "A"
                },
                {
                    "type": 2,
                    "operator": 1,
                    "formulaid": "B"
                }
            ]
        },
        "operations": [
            {
                "type": 1
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::create() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.delete

Opis

`object correlation.delete(array correlationids)`

Ta metoda umożliwia usuwanie korelacji.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` IDs korelacji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID usuniętych korelacji we właściwości `correlationids`.

Przykład

Usuń wiele korelacji

Usuń dwie korelacje.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "correlation.delete",
"params": [
"1",
"2"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"correlationids": [
"1",
"2"
]
},
"id": 1
}
```

Źródło

CCorrelation::delete() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.get

Opis

`integer/array correlation.get(object parameters)`

Ta metoda umożliwia pobieranie korelacji zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
correlationids	ID/array	Zwraca tylko korelacje o podanych identyfikatorach.
selectFilter	query	Zwraca właściwość `filter` z warunkami korelacji.
selectOperations	query	Zwraca właściwość `operations` z operacjami korelacji.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `correlationid`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	

Parametr	Typ	Opis
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobieranie korelacji

Pobierz wszystkie skonfigurowane korelacje wraz z warunkami i operacjami korelacji. Filtr używa typu oceny „and/or”, dlatego właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectFilter": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "correlationid": "1",
            "name": "Correlation 1",
            "description": "",
            "status": "0",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [
                    {
                        "type": "3",
                        "oldtag": "error",
                        "newtag": "ok",
                        "formulaid": "A"
                    }
                ],
                "eval_formula": "A"
            },
            "operations": [
                {
                    "type": "0"
                }
            ]
        }
```

```
    ],
    "id": 1
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::get() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.update

Opis

`object correlation.update(object/array correlations)`

Ta metoda umożliwia aktualizację istniejących korelacji.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości korelacji do zaktualizowania.

Właściwość `correlationid` musi być zdefiniowana dla każdej korelacji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości korelacji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra korelacji zastępujący bieżący filtr.
operations	array	Operacje korelacji zastępujące istniejące operacje.

Zwracane wartości

`(object)` Zwraca obiekt zawierajacy identyfikatory zaktualizowanych korelacji we właściwości ‘correlationids.

Przykłady

Wyłącz korelację

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "status": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
```

```
    "id": 1
}
```

Zastąp warunki, ale zachowaj metodę oceny

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "filter": {
            "conditions": [
                {
                    "type": 3,
                    "oldtag": "error",
                    "newtag": "ok"
                }
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::update() w *ui/include/classes/api/services/CCorrelation.php*.

Makro użytkownika

Ta klasa jest przeznaczona do pracy z makrami użytkownika na poziomie hosta oraz globalnymi makrami użytkownika.

Odwołania do obiektów:

- Makro globalne
- Makro hosta
    - Konfiguracja makra

Dostępne metody:

- usermacro.create - utwórz nowe makra hosta
- usermacro.createglobal - utwórz nowe makra globalne
- usermacro.delete - usuń makra hosta
- usermacro.deleteglobal - usuń makra globalne
- usermacro.get - pobierz makra hosta i makra globalne
- usermacro.update - zaktualizuj makra hosta
- usermacro.updateglobal - zaktualizuj makra globalne

Obiekt makra użytkownika

Następujące obiekty są bezpośrednio powiązane z API `usermacro`.

Makro globalne

Obiekt makra globalnego ma następujące właściwości.

Właściwość	Typ	Opis
globalmacroid	ID	ID makra globalnego. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Opis makra.
macro	string	Ciąg makra. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ makra. Możliwe wartości: 0 - *(domyślnie)* Makro tekstowe; 1 - Makro sekretne; 2 - Sekret Vault.
value	string	Wartość makra. Zachowanie właściwości: - *tylko do zapisu*, jeśli `type` jest ustawione na „Makro sekretne” - *wymagane* dla operacji tworzenia

Makro hosta

Obiekt makra hosta definiuje makro dostępne na hoście, prototypie hosta lub szablonie. Ma następujące właściwości.

Właściwość	Typ	Opis
hostmacroid	ID	ID makra. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
automatic	integer	Określa, czy makro jest kontrolowane przez regułę wykrywania. Możliwe wartości: 0 - *(domyślnie)* Makro jest zarządzane przez użytkownika; 1 - Makro jest zarządzane przez regułę wykrywania. Użytkownik nie może tworzyć automatycznego makra. Aby zaktualizować automatyczne makro, musi ono zostać przekształcone na ręczne.
config	object/array	Konfiguracja makra odpowiedzialna za sposób wyświetlania makra w Kreatorze hosta.
description	string	Opis makra.
hostid	ID	ID hosta, prototypu hosta lub szablonu, do którego należy makro. Zachowanie właściwości: - *stała* - *wymagane* dla operacji tworzenia
macro	string	Ciąg makra. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
type	integer	Typ makra. Możliwe wartości: 0 - *(domyślnie)* Makro tekstowe; 1 - Makro tajne; 2 - Sekret z magazynu.
value	string	Wartość makra. Zachowanie właściwości: - *tylko do zapisu*, jeśli `type` jest ustawione na „Makro tajne” - *wymagane* dla operacji tworzenia

Konfiguracja makr

Obiekt konfiguracji makr definiuje sposób wyświetlania makra w Kreatorze hosta.

Właściwość	Typ	Opis
type	integer	Typ pola wejściowego makra. Możliwe wartości: 0 - Makro nie jest używane w Kreatorze hosta; 1 - Pole tekstowe; 2 - Lista; 3 - Pole wyboru. Zachowanie właściwości: - *wymagane*
label	string	Etykieta pola wejściowego makra. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Pole tekstowe”, „Lista” lub „Pole wyboru”
description	text	Tekst pomocy wyświetlany obok pola wejściowego makra. Obsługuje formatowanie Markdown.
priority	integer	Pozycja pola wejściowego makra na liście makr. Jeśli `priority` nie jest ustawione, makro jest dodawane na końcu listy niepogrupowanych makr.
required	integer	Oznacza makro jako obowiązkowe. Możliwe wartości: 0 - Nieobowiązkowe; 1 - Obowiązkowe. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Pole tekstowe” lub „Lista”
regex	string	Wyrażenie regularne służące do walidacji danych wejściowych użytkownika w polu tekstowym. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Pole tekstowe”
section_name	string	Etykieta zwijanej sekcji, w której makro jest grupowane. Jeśli `section_name` nie jest ustawione, makro nie jest grupowane. Niepogrupowane makra są wyświetlane jako pierwsze; pogrupowane makra są wyświetlane poniżej, uporządkowane według ich `priority` w obrębie każdej zwijanej sekcji.

Właściwość	Typ	Opis
options	text	Ciąg JSON definiujący elementy listy lub wartości pola wyboru. Dla list: tablica obiektów z właściwościami `value` i `text`. Przykład: `[{"value": "http", "text": "HTTP"}, {"value": "https", "text": "HTTPS"}]` Dla pól wyboru: obiekt z właściwościami `checked` i `unchecked`. Przykład: `{"checked": true, "unchecked": false}` Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Lista” lub „Pole wyboru”

usermacro.create

Opis

`object usermacro.create(object/array hostMacros)`

Ta metoda umożliwia tworzenie nowych makr hosta.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Makra hosta do utworzenia.

Metoda akceptuje makra hosta ze standardowymi właściwościami makr hosta.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych makr hosta we właściwości `hostmacroids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych makr hosta.

Przykłady

Tworzenie makra hosta

Utwórz makro hosta ”{$SNMP_COMMUNITY}” o wartości ”public” na hoście ”10198”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.create",
    "params": {
        "hostid": "10198",
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "11"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::create() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.createglobal

Opis

`object usermacro.createglobal(object/array globalMacros)`

Ta metoda umożliwia tworzenie nowych makr globalnych.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Zobacz User roles , aby uzyskać więcej informacji.

Parametry

`(object/array)` Makra globalne do utworzenia.

Metoda akceptuje makra globalne ze standardowymi właściwościami makr globalnych.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych makr globalnych we właściwości `globalmacroids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych makr globalnych.

Przykłady

Tworzenie makra globalnego

Utwórz makro globalne "{$SNMP_COMMUNITY}" o wartości "public".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.createglobal",
    "params":  {
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::createGlobal() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.delete

Opis

`object usermacro.delete(array hostMacroIds)`

Ta metoda pozwala na usuwanie makr hosta.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Role użytkownika.

Parametry

`(array)` ID makr hosta do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych makr hosta we właściwości `hostmacroids`.

Przykłady

Usuwanie wielu makr hosta

Usuń dwa makra hosta.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usermacro.delete",
"params": [
"32",
"11"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostmacroids": [
"32",
"11"
]
},
"id": 1
}
```

Źródło

CUserMacro::delete() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.deleteglobal

Opis

`object usermacro.deleteglobal(array globalMacroIds)`

Ta metoda pozwala na usuwanie makr globalnych.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID globalnych makr do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych globalnych makr we właściwości `globalmacroids`.

Przykłady

Usuwanie wielu makr globalnych

Usuń dwie makra globalne.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usermacro.deleteglobal",
"params": [
"32",
"11"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"globalmacroids": [
"32",
"11"
]
},
"id": 1
}
```

Źródło

CUserMacro::deleteGlobal() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.get

Opis

`integer/array usermacro.get(object parameters)`

Ta metoda umożliwia pobieranie makr hosta i makr globalnych zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
globalmacro	flag	Zwraca makra globalne zamiast makr hosta.
globalmacroids	ID/array	Zwraca tylko makra globalne o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko makra hosta należące do hostów lub szablonów z podanych grup hostów lub grup szablonów.
hostids	ID/array	Zwraca tylko makra należące do podanych hostów lub szablonów.
hostmacroids	ID/array	Zwraca tylko makra hosta o podanych identyfikatorach.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko makra użytkownika prototypu hosta dziedziczone z szablonu.
selectHostGroups	query	Zwraca grupy hostów, do których należy makro hosta, we właściwości `hostgroups`. Używane tylko podczas pobierania makr hosta.
selectHosts	query	Zwraca hosty, do których należy makro hosta, we właściwości `hosts`. Używane tylko podczas pobierania makr hosta.

Parametr	Typ	Opis
selectTemplateGroups	query	Zwraca grupy szablonów, do których należy makro szablonu, we właściwości `templategroups`. Używane tylko podczas pobierania makr szablonu.
selectTemplates	query	Zwraca szablony, do których należy makro hosta, we właściwości `templates`. Używane tylko podczas pobierania makr hosta.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `macro`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie makr hosta dla hosta

Pobierz wszystkie makra hosta zdefiniowane dla hosta „10198”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10198"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "9",
            "hostid": "10198",
            "macro": "{$INTERFACE}",
            "value": "eth0",
            "description": "",
            "type": "0",
            "automatic": "0"
        },
        {
            "hostmacroid": "11",
```

```
            "hostid": "10198",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0",
            "automatic": "0"
        }
    ],
    "id": 1
}
```

Pobieranie makr hosta dla szablonu

Pobierz makra hosta zdefiniowane dla szablonu „10265”, które zawierają „STATUS” w nazwie makra.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10265",
        "search": {
            "macro": "STATUS"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "6709",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.HOST}",
            "value": "",
            "description": "Nazwa hosta lub adres IP hosta strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "1",
                "section_name": "",
                "label": "Host statusu Apache",
                "description": "Nazwa hosta lub adres IP hosta strony statusu Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "814",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PATH}",
            "value": "server-status?auto",
            "description": "Ścieżka URL strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "3",
```

```
                "section_name": "",
                "label": "Ścieżka strony statusu Apache",
                "description": "Ścieżka URL strony statusu Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "815",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PORT}",
            "value": "80",
            "description": "Port strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "2",
                "section_name": "",
                "label": "Port strony statusu Apache",
                "description": "W zakresie od 1 do 65535 włącznie.",
                "required": "1",
                "regex": "^-?([0-9]+|(([0-9]+)\\.([0-9]+)))$",
                "options": []
            }
        },
        {
            "hostmacroid": "816",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.SCHEME}",
            "value": "http",
            "description": "Schemat żądania, którym może być HTTP lub HTTPS.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "2",
                "priority": "4",
                "section_name": "",
                "label": "Schemat żądania",
                "description": "Schemat żądania, którym może być HTTP lub HTTPS.",
                "required": "0",
                "regex": "",
                "options": [
                    {
                        "value": "http",
                        "text": "HTTP"
                    },
                    {
                        "value": "https",
                        "text": "HTTPS"
                    }
                ]
            }
        }
    ],
    "id": 1
}
```


Pobieranie makr globalnych

Pobierz wszystkie makra globalne.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "globalmacro": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "globalmacroid": "6",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0"
        }
    ],
    "id": 1
}
```

Źródło

CUserMacro::get() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.update

Opis

`object usermacro.update(object/array hostMacros)`

Ta metoda umożliwia aktualizację istniejących makr hosta.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości makra hosta, które mają zostać zaktualizowane.

Właściwość `hostmacroid` musi być zdefiniowana dla każdego makra hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych makr hosta we właściwości `hostmacroids`.

Przykłady

Zmiana wartości makra hosta

Zmień wartość makra hosta na „public”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "public"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Zmień wartość makra utworzonego przez regułę wykrywania

Zmień utworzone przez regułę wykrywania makro „automatic” na „manual” i zmień jego wartość na „new-value”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "new-value",
        "automatic": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::update() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.updateglobal

Opis

`object usermacro.updateglobal(object/array globalMacros)`

Ta metoda umożliwia aktualizację istniejących globalnych makr.

Note:
Ta metoda jest dostępna wyłącznie dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości makra globalnego, które mają zostać zaktualizowane.

Właściwość `globalmacroid` musi być zdefiniowana dla każdego makra globalnego, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanego globalnego makra pod właściwością globalmacroids.

Przykłady

Zmiana wartości makra globalnego

Zmień wartość makra globalnego na „public”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.updateglobal",
    "params": {
        "globalmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::updateGlobal() in *ui/include/classes/api/services/CUserMacro.php*.

Mapa

Ta klasa jest przeznaczona do pracy z mapami.

Odwołania do obiektów:

- Mapa
- Element mapy
  - Element mapy Host
  - Element mapy Grupa hostów
  - Element mapy Mapa
  - Element mapy Wyzwalacz
  - Element mapy Tag
  - Element mapy URL
- Łącze mapy
  - Wyzwalacz łącza mapy
  - Wskaźniki łącza mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Dostępne metody:

- map.create - utwórz nowe mapy
- map.delete - usuń mapy
- map.get - pobierz mapy
- map.update - zaktualizuj mapy

Obiekt mapy

Następujące obiekty są bezpośrednio powiązane z API `map`.

Mapa

Obiekt mapy ma następujące właściwości.

Właściwość	Typ	Opis
sysmapid	ID	ID mapy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
height	integer	Wysokość mapy w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość mapy w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
backgroundid	ID	ID obrazu używanego jako tło mapy.
background_scale	integer	Czy włączyć proporcjonalne skalowanie obrazu tła. Możliwe wartości: 0 - wyłącz skalowanie; 1 - *(domyślnie)* włącz skalowanie obrazu.
expand_macros	integer	Czy rozwijać makra w etykietach podczas konfigurowania mapy. Możliwe wartości: 0 - *(domyślnie)* nie rozwijaj makr; 1 - rozwijaj makra.
expandproblem	integer	Czy wyzwalacz problemu będzie wyświetlany dla elementów z pojedynczym problemem. Możliwe wartości: 0 - zawsze wyświetlaj liczbę problemów; 1 - *(domyślnie)* wyświetlaj wyzwalacz problemu, jeśli istnieje tylko jeden problem.
grid_align	integer	Czy włączyć wyrównywanie do siatki. Możliwe wartości: 0 - wyłącz wyrównywanie do siatki; 1 - *(domyślnie)* włącz wyrównywanie do siatki.
grid_show	integer	Czy wyświetlać siatkę na mapie. Możliwe wartości: 0 - nie wyświetlaj siatki; 1 - *(domyślnie)* wyświetlaj siatkę.
grid_size	integer	Rozmiar siatki mapy w pikselach. Obsługiwane wartości: 20, 40, 50, 75 i 100. Domyślnie: 50.
highlight	integer	Czy podświetlanie ikon jest włączone. Możliwe wartości: 0 - podświetlanie wyłączone; 1 - *(domyślnie)* podświetlanie włączone.
iconmapid	ID	ID mapy ikon używanej na mapie.

Właściwość	Typ	Opis
label_format	integer	Czy włączyć zaawansowane etykiety. Możliwe wartości: 0 - *(domyślnie)* wyłącz zaawansowane etykiety; 1 - włącz zaawansowane etykiety.
label_location	integer	Położenie etykiety elementu mapy. Możliwe wartości: 0 - *(domyślnie)* dół; 1 - lewo; 2 - prawo; 3 - góra.
label_string_host	string	Niestandardowa etykieta dla elementów typu host. Zachowanie właściwości: - *wymagane*, jeśli `label_type_host` jest ustawione na ”custom”
label_string_hostgroup	string	Niestandardowa etykieta dla elementów grupy hostów. Zachowanie właściwości: - *wymagane*, jeśli `label_type_hostgroup` jest ustawione na ”custom”
label_string_image	string	Niestandardowa etykieta dla elementów obrazu. Zachowanie właściwości: - *wymagane*, jeśli `label_type_image` jest ustawione na ”custom”
label_string_map	string	Niestandardowa etykieta dla elementów mapy. Zachowanie właściwości: - *wymagane*, jeśli `label_type_map` jest ustawione na ”custom”
label_string_trigger	string	Niestandardowa etykieta dla elementów wyzwalacza. Zachowanie właściwości: - *wymagane*, jeśli `label_type_trigger` jest ustawione na ”custom”
label_type	integer	Typ etykiety elementu mapy. Możliwe wartości: 0 - etykieta; 1 - adres IP; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic.
label_type_host	integer	Typ etykiety dla elementów typu host. Możliwe wartości: 0 - etykieta; 1 - adres IP; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
label_type_hostgroup	integer	Typ etykiety dla elementów grupy hostów. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.

Właściwość	Typ	Opis
label_type_image	integer	Typ etykiety dla elementów grupy hostów. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 4 - nic; 5 - niestandardowa.
label_type_map	integer	Typ etykiety dla elementów mapy. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
label_type_trigger	integer	Typ etykiety dla elementów wyzwalacza. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
markelements	integer	Czy podświetlać elementy mapy, których status ostatnio się zmienił. Możliwe wartości: 0 - *(domyślnie)* nie podświetlaj elementów; 1 - podświetlaj elementy.
severity_min	integer	Minimalny poziom ważności wyzwalaczy, które będą wyświetlane na mapie. Zobacz właściwość `severity` wyzwalacza, aby uzyskać listę obsługiwanych poziomów ważności wyzwalaczy.
show_element_label	integer	Jak domyślnie wyświetlać etykiety elementów. Możliwe wartości: 0 - zawsze wyświetlaj; 1 - *(domyślnie)* ukrywaj automatycznie.
show_link_label	integer	Jak domyślnie wyświetlać etykiety łączy. Możliwe wartości: 0 - zawsze wyświetlaj; 1 - *(domyślnie)* ukrywaj automatycznie.
show_unack	integer	Jak mają być wyświetlane problemy. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj liczbę wszystkich problemów; 1 - wyświetlaj tylko liczbę niepotwierdzonych problemów; 2 - wyświetlaj osobno liczbę potwierdzonych i niepotwierdzonych problemów.
userid	ID	ID użytkownika będącego właścicielem mapy.
private	integer	Typ udostępniania mapy. Możliwe wartości: 0 - mapa publiczna; 1 - *(domyślnie)* mapa prywatna.
show_suppressed	integer	Czy wyciszone problemy są wyświetlane. Możliwe wartości: 0 - *(domyślnie)* ukrywaj wyciszone problemy; 1 - wyświetlaj wyciszone problemy.

Element mapy

Obiekt elementu mapy definiuje obiekt wyświetlany na mapie. Ma on następujące właściwości.

Właściwość	Typ	Opis
selementid	ID	ID elementu mapy. Zachowanie właściwości: - *tylko do odczytu*
elements	array	Obiekt danych elementu. Zachowanie właściwości: - *wymagane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
elementtype	integer	Typ elementu mapy. Możliwe wartości: 0 - host; 1 - mapa; 2 - wyzwalacz; 3 - grupa hostów; 4 - obraz. Zachowanie właściwości: - *wymagane*
iconid_off	ID	ID obrazu używanego do wyświetlania elementu w stanie domyślnym. Zachowanie właściwości: - *wymagane*
areatype	integer	Sposób wyświetlania oddzielnych hostów grupy hostów. Możliwe wartości: 0 - *(domyślnie)* element grupy hostów zajmie całą mapę; 1 - element grupy hostów będzie miał stały rozmiar.
elementsubtype	integer	Sposób wyświetlania elementu grupy hostów na mapie. Możliwe wartości: 0 - *(domyślnie)* wyświetl grupę hostów jako pojedynczy element; 1 - wyświetl każdy host w grupie oddzielnie.
evaltype	integer	Warunek filtrowania tagów elementu mapy, metoda obliczania. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
height	integer	Wysokość elementu grupy hostów o stałym rozmiarze w pikselach. Domyślnie: 200.
iconid_disabled	ID	ID obrazu używanego do wyświetlania wyłączonych elementów mapy. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
iconid_maintenance	ID	ID obrazu używanego do wyświetlania elementów mapy w konserwacji. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
iconid_on	ID	ID obrazu używanego do wyświetlania elementów mapy z problemami. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
label	string	Etykieta elementu.

Właściwość	Typ	Opis
label_location	integer	Położenie etykiety elementu mapy. Możliwe wartości: -1 - *(domyślnie)* położenie domyślne; 0 - dół; 1 - lewo; 2 - prawo; 3 - góra.
permission	integer	Typ poziomu uprawnień. Możliwe wartości: -1 - brak; 2 - tylko do odczytu; 3 - odczyt-zapis.
show_label	integer	Sposób wyświetlania etykiety elementu. Możliwe wartości: -1 - *(domyślnie)* domyślne ustawienie mapy; 0 - zawsze wyświetlaj; 1 - ukrywaj automatycznie.
sysmapid	ID	ID mapy, do której należy element. Zachowanie właściwości: - *tylko do odczytu*
urls	array	Adresy URL elementu mapy. Obiekt URL elementu mapy jest opisany szczegółowo poniżej.
use_iconmap	integer	Czy mapowanie ikon musi być używane dla elementów typu host. Możliwe wartości: 0 - nie używaj mapowania ikon; 1 - *(domyślnie)* używaj mapowania ikon.
viewtype	integer	Algorytm rozmieszczania elementu grupy hostów. Możliwe wartości: 0 - *(domyślnie)* siatka.
width	integer	Szerokość elementu grupy hostów o stałym rozmiarze w pikselach. Domyślnie: 200.
x	integer	Współrzędna X elementu w pikselach. Domyślnie: 0.
y	integer	Współrzędna Y elementu w pikselach. Domyślnie: 0.
zindex	integer	Wartość używana do porządkowania elementów mapy (z-index). Domyślnie: 0.

Element mapy Host

Obiekt elementu mapy Host definiuje jeden element hosta.

Właściwość	Typ	Opis
hostid	ID	ID hosta.

Element mapy Grupa hostów

Obiekt elementu mapy Grupa hostów definiuje jeden element grupy hostów.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów.

Obiekt elementu mapy Map

Obiekt elementu mapy Map definiuje jeden element mapy.

Właściwość	Type	Opis
sysmapid	ID	ID mapy.

Wyzwalacz elementu mapy

Obiekt wyzwalacza elementu mapy definiuje jeden lub więcej elementów wyzwalacza.

Właściwość	Type	Opis
triggerid	ID	ID wyzwalacza.

Tag elementu mapy

Obiekt Tag elementu mapy ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu elementu mapy. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek operatora tagu elementu mapy. Możliwe wartości: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
value	string	Wartość tagu elementu mapy.

URL elementu mapy

Obiekt URL elementu mapy definiuje klikalny link, który będzie dostępny dla określonego elementu mapy. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapelementurlid	ID	ID URL elementu mapy. Zachowanie właściwości: - *tylko do odczytu*
name	string	Podpis linku. Zachowanie właściwości: - *wymagane*
url	string	URL linku. Zachowanie właściwości: - *wymagane*
selementid	ID	ID elementu mapy, do którego należy URL.

Łącze mapy

Obiekt łącza mapy definiuje połączenie między dwoma elementami mapy. Ma następujące właściwości.

Właściwość	Typ	Opis
linkid	ID	ID łącza mapy. Zachowanie właściwości: - *tylko do odczytu*
sysmapid	ID	ID mapy, do której należy łącze.
selementid1	ID	ID pierwszego elementu mapy połączonego z jednej strony. Zachowanie właściwości: - *wymagane*
selementid2	ID	ID pierwszego elementu mapy połączonego z drugiej strony. Zachowanie właściwości: - *wymagane*
drawtype	integer	Styl rysowania linii łącza. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia kreskowana.
color	string	Kolor linii jako szesnastkowy kod koloru. Domyślnie: 000000.
label	string	Etykieta łącza.
show_label	integer	Sposób wyświetlania etykiety łącza. Możliwe wartości: -1 - *(domyślnie)* domyślne ustawienie mapy; 0 - zawsze wyświetlaj; 1 - ukrywaj automatycznie.
indicator_type	integer	Wybór typu wskaźnika łącza. Możliwe wartości: 0 - *(domyślnie)* łącze statyczne; 1 - wyzwalacz; 2 - wartość pozycji.
linktriggers	array	Wyzwalacze łącza mapy używane jako wskaźniki stanu łącza. Obiekt wyzwalacza łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”trigger” (1).
itemid	ID	ID pozycji. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”item value” (2).
highlights	array	Wyróżnienia łącza mapy używane jako wskaźniki stanu łącza. Obiekt wskaźników łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”item value” (2).

Właściwość	Typ	Opis
thresholds	array	Progi łącza mapy używane jako wskaźniki stanu łącza. Obiekt wskaźników łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na "item value" (2).
permission	integer	Typ poziomu uprawnień. Możliwe wartości: -1 - brak; 2 - tylko do odczytu; 3 - odczyt-zapis.

Wyzwalacz łącza mapy

Obiekt wyzwalacza łącza mapy definiuje wskaźnik stanu łącza mapy na podstawie stanu wyzwalacza. Ma następujące właściwości:

Właściwość	Typ	Opis
triggerid	ID	ID wyzwalacza używanego jako wskaźnik łącza. Zachowanie właściwości: - *wymagane*
color	string	Kolor wskaźnika jako szesnastkowy kod koloru. Domyślnie: `DD0000`.
drawtype	integer	Styl rysowania wskaźnika. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia przerywana.

Wskaźniki łączy mapy

Obiekt wskaźników łączy mapy definiuje wskaźnik stanu łącza mapy na podstawie wartości pozycji. Ma następujące właściwości:

Właściwość	Typ	Opis
pattern	string	Wyrażenie regularne do porównania. Dostępne tylko dla wyróżnień.
sortorder	integer	Używane do definiowania kolejności sortowania wyróżnień. Dostępne tylko dla wyróżnień.
threshold	string	Wartość liczbowa do porównania. Dostępne tylko dla progów.
drawtype	integer	Styl rysowania wskaźnika. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia kreskowana.
color	string	Kolor wskaźnika jako szesnastkowy kod koloru. Domyślnie: `DD0000`. Zachowanie właściwości: - *wymagane*

URL mapy

Obiekt URL mapy definiuje klikalny link, który będzie dostępny dla wszystkich elementów określonego typu na mapie. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapurlid	ID	ID URL mapy. Zachowanie właściwości: - *tylko do odczytu*
name	string	Podpis linku. Zachowanie właściwości: - *wymagane*
url	string	URL linku. Zachowanie właściwości: - *wymagane*
elementtype	integer	Typ elementu mapy, dla którego URL będzie dostępny. Listę obsługiwanych typów można znaleźć we właściwości `type` elementu mapy. Domyślnie: 0.
sysmapid	ID	ID mapy, do której należy URL.

Użytkownik mapy

Lista uprawnień do mapy na podstawie użytkowników. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapuserid	ID	ID użytkownika mapy. Zachowanie właściwości: - *tylko do odczytu*
userid	ID	ID użytkownika. Zachowanie właściwości: - *wymagane*
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

Grupa użytkowników mapy

Lista uprawnień do mapy opartych na grupach użytkowników. Ma następujące właściwości:

Właściwość	Type	Opis
sysmapusrgrpid	ID	ID grupy użytkowników mapy. Property behavior: - *tylko do odczytu*
usrgrpid	ID	ID grupy użytkowników. Property behavior: - *wymagane*

Właściwość	Type	Opis
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Property behavior: - *wymagane*

Kształty mapy

Obiekt kształtu mapy definiuje kształt geometryczny (z tekstem lub bez) wyświetlany na mapie. Ma następujące właściwości:

Właściwość	Type	Opis
sysmap_shapeid	ID	ID elementu kształtu mapy. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Typ elementu kształtu mapy. Możliwe wartości: 0 - prostokąt; 1 - elipsa. Właściwość jest wymagana podczas tworzenia nowych kształtów. Zachowanie właściwości: - *wymagane*
x	integer	Współrzędne X kształtu w pikselach. Domyślnie: 0.
y	integer	Współrzędne Y kształtu w pikselach. Domyślnie: 0.
width	integer	Szerokość kształtu w pikselach. Domyślnie: 200.
height	integer	Wysokość kształtu w pikselach. Domyślnie: 200.
text	string	Tekst kształtu.
font	integer	Czcionka tekstu wewnątrz kształtu. Możliwe wartości: 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 Domyślnie: 9.

Właściwość	Type	Opis
font_size	integer	Rozmiar czcionki w pikselach. Domyślnie: 11.
font_color	string	Kolor czcionki. Domyślnie: 000000.
text_halign	integer	Wyrównanie poziome tekstu. Możliwe wartości: 0 - środek; 1 - do lewej; 2 - do prawej. Domyślnie: 0.
text_valign	integer	Wyrównanie pionowe tekstu. Możliwe wartości: 0 - środek; 1 - góra; 2 - dół. Domyślnie: 0.
border_type	integer	Typ obramowania. Możliwe wartości: 0 - brak; 1 - ————; 2 - ┅; 3 - – – –. Domyślnie: 0.
border_width	integer	Szerokość obramowania w pikselach. Domyślnie: 0.
border_color	string	Kolor obramowania. Domyślnie: 000000.
background_color	string	Kolor tła (kolor wypełnienia). Domyślnie: (puste).
zindex	integer	Wartość używana do porządkowania wszystkich kształtów i linii (z-index). Domyślnie: 0.

Linie mapy

Obiekt linii mapy definiuje linię wyświetlaną na mapie. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmap_shapeid	ID	ID elementu kształtu mapy. Zachowanie właściwości: *- tylko do odczytu*
x1	integer	Współrzędne X punktu 1 linii w pikselach. Domyślnie: 0.
y1	integer	Współrzędne Y punktu 1 linii w pikselach. Domyślnie: 0.

Właściwość	Typ	Opis
x2	integer	Współrzędne X punktu 2 linii w pikselach. Domyślnie: 200.
y2	integer	Współrzędne Y punktu 2 linii w pikselach. Domyślnie: 200.
line_type	integer	Typ linii. Możliwe wartości: 0 - brak; 1 - ——————; 2 - ┅; 3 - – – –. Domyślnie: 0.
line_width	integer	Szerokość linii w pikselach. Domyślnie: 0.
line_color	string	Kolor linii. Domyślnie: 000000.
zindex	integer	Wartość używana do porządkowania wszystkich kształtów i linii (z-index). Domyślnie: 0.

map.create

Opis

`object map.create(object/array maps)`

Ta metoda umożliwia tworzenie nowych map.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Mapy do utworzenia.

Oprócz standardowych właściwości mapy, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
links	array	Łącza mapy do utworzenia na mapie.
selements	array	Elementy mapy do utworzenia na mapie.
urls	array	Adresy URL mapy do utworzenia na mapie.
users	array	Udostępnienia użytkownikowi mapy do utworzenia na mapie.
userGroups	array	Udostępnienia grupie użytkowników mapy do utworzenia na mapie.
shapes	array	Kształty mapy do utworzenia na mapie.
lines	array	Linie mapy do utworzenia na mapie.

> **Note:**
> Aby utworzyć łącza mapy, należy ustawić `selementid` elementu mapy na dowolną wartość, a następnie użyć tej wartości do odwołania się do tego elementu we właściwościach łączy `selementid1` lub `selementid2`. Po utworzeniu elementu wartość ta zostanie zastąpiona poprawnym identyfikatorem wygenerowanym przez Zabbix. Zobacz przykład.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych map we właściwości sysmapids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych map.

Przykłady

Utwórz pustą mapę

Utwórz mapę bez elementów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map",
        "width": 600,
        "height": 600
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Utwórz mapę hostów

Utwórz mapę z dwoma elementami hostów i łączem między nimi. Zwróć uwagę na użycie tymczasowych wartości „selementid1” i „selementid2” w obiekcie łącza mapy do odwoływania się do elementów mapy.

Żądanie:

```
{
    "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"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Utwórz mapę wyzwalaczy

Utwórz mapę z elementem wyzwalacza, który zawiera dwa wyzwalacze.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Mapa wyzwalaczy",
        "width": 600,
        "height": 600,
        "selements": [
            {
                "elements": [
                    {"triggerid": "12345"},
                    {"triggerid": "67890"}
                ],
                "elementtype": 2,
                "iconid_off": "2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Udostępnianie mapy

Utwórz mapę z dwoma typami udostępniania (użytkownik i grupa użytkowników).

Żądanie:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "map.create",
    "params": {
        "name": "Map sharing",
        "width": 600,
        "height": 600,
        "users": [
            {
                "userid": "4",
                "permission": "3"
            }
        ],
        "userGroups": [
            {
                "usrgrpid": "7",
                "permission": "2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Kształty mapy

Utwórz mapę z tytułem będącym nazwą mapy.

Żądanie:

```
{
    "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}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
```

```
            "10"
        ]
    },
    "id": 1
}
```

Linie mapy

Utwórz linię mapy.

Żądanie:

```
{
    "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"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "11"
        ]
    },
    "id": 1
}
```

Zobacz także

- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształt mapy
- Linia mapy

Źródło

CMap::create() w *ui/include/classes/api/services/CMap.php*.

map.delete

Opis

`object map.delete(array mapIds)`

Ta metoda umożliwia usuwanie map.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID map do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych map we właściwości `sysmapids`.

Przykłady

Usuń wiele map

Usuń dwie mapy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.delete",
    "params": [
        "12",
        "34"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "12",
            "34"
        ]
    },
    "id": 1
}
```

Źródło

CMap::delete() w *ui/include/classes/api/services/CMap.php*.

map.get

Opis

`integer/array map.get(object parameters)`

Ta metoda umożliwia pobieranie map zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
sysmapids	ID/array	Zwraca tylko mapy o podanych identyfikatorach.
userids	ID/array	Zwraca tylko mapy należące do użytkowników o podanych identyfikatorach.

Parametr	Typ	Opis
expandUrls	flag	Dodaje globalne adresy URL map do odpowiednich elementów mapy i rozwija makra we wszystkich adresach URL elementów mapy.
selectIconMap	query	Zwraca właściwość `iconmap` z mapą ikon używaną na mapie.
selectLinks	query	Zwraca właściwość `links` z połączeniami mapy między elementami.
selectSelements	query	Zwraca właściwość `selements` z elementami mapy.
selectUrls	query	Zwraca właściwość `urls` z adresami URL mapy.
selectUsers	query	Zwraca właściwość `users` z użytkownikami, którym mapa jest udostępniona.
selectUserGroups	query	Zwraca właściwość `userGroups` z grupami użytkowników, którym mapa jest udostępniona.
selectShapes	query	Zwraca właściwość `shapes` z kształtami mapy.
selectLines	query	Zwraca właściwość `lines` z liniami mapy.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`, `width`, `height`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie mapy

Pobierz wszystkie dane o mapie „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.get",
    "params": {
        "output": "extend",
        "selectSelements": "extend",
        "selectLinks": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "selectShapes": "extend",
        "selectLines": "extend",
        "sysmapids": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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",
                "viewtype": "0",
                "use_iconmap": "1",
                "show_label": "-1",
                "zindex":"0",
                "elements": [],
                "urls": [],
                "tags": [
                    {
                        "tag": "service",
                        "value": "mysqld",
                        "operator": "0"
                    }
                ]
            },
            {
                "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",
                "show_label": "0",
                "zindex":"0",
                "elements": [],
                "urls": [],
                "tags": []
            },
            {
                "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",
                "show_label": "1",
                "zindex":"0",
                "elements": [
                    {
                        "hostid": "10084"
                    }
                ],
                "urls": [],
                "tags": []
            }
        ],
        "links": [
            {
                "linkid": "23",
                "sysmapid": "3",
                "selementid1": "10",
                "selementid2": "11",
                "drawtype": "0",
                "color": "00CC00",
                "label": "",
                "show_label": "1",
                "indicator_type": "0",
                "itemid": "0",
                "linktriggers": [],
                "thresholds": [],
                "highlights": []
            }
        ],
        "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",
        "background_scale": "1",
        "label_type": "2",
        "label_location": "3",
        "show_element_label": "0",
        "show_link_label": "1",
        "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
}
```

Zobacz także

- Mapa ikon
- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Źródło

CMap::get() w *ui/include/classes/api/services/CMap.php*.

map.update

Opis

`object map.update(object/array maps)`

Ta metoda umożliwia aktualizację istniejących map.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Właściwości mapy do zaktualizowania.

Właściwość `mapid` musi być zdefiniowana dla każdej mapy, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości mapy metoda akceptuje następujące parametry.

Parametr	Type	Opis
links	array	Łącza mapy zastępujące istniejące łącza.
selements	array	Elementy mapy zastępujące istniejące elementy.
urls	array	Adresy URL mapy zastępujące istniejące adresy URL.
users	array	Udostępnienia użytkownikowi mapy zastępujące istniejące elementy.
userGroups	array	Udostępnienia grupie użytkowników mapy zastępujące istniejące elementy.
shapes	array	Kształty mapy zastępujące istniejące kształty.
lines	array	Linie mapy zastępujące istniejące linie.

> **Note:**
> Aby utworzyć łącza mapy między nowymi elementami mapy, należy ustawić `selementid` elementu na dowolną wartość, a następnie użyć tej wartości do odwołania się do tego elementu we właściwościach łączy `selementid1` lub `selementid2`. Po utworzeniu elementu wartość ta zostanie zastąpiona poprawnym ID wygenerowanym przez Zabbix. Zobacz przykład dla map.create.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych map we właściwości `sysmapids`.

Przykłady

Zmień rozmiar mapy

Zmień rozmiar mapy na 1200x1200 pikseli.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "8",
        "width": 1200,
        "height": 1200
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Zmień właściciela mapy

Dostępne tylko dla administratorów i super administratorów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "9",
        "userid": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Zobacz także

- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Źródło

CMap::update() w *ui/include/classes/api/services/CMap.php*.

Mapa ikon

Ta klasa jest przeznaczona do pracy z mapami ikon.

Odwołania do obiektów:

- Mapa ikon
- Mapowanie ikon

Dostępne metody:

- iconmap.create - utwórz nowe mapy ikon
- iconmap.delete - usuń mapy ikon
- iconmap.get - pobierz mapy ikon
- iconmap.update - zaktualizuj mapy ikon

Obiekt mapy ikon

Następujące obiekty są bezpośrednio powiązane z API `iconmap`.

Mapa ikon

Obiekt mapy ikon ma następujące właściwości.

Właściwość	Typ	Opis
iconmapid	ID	ID mapy ikon. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
default_iconid	ID	ID domyślnej ikony. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy ikon. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Mapowanie ikon

Obiekt mapowania ikon definiuje konkretną ikonę, która ma być używana dla hostów o określonej wartości pola inwentarza. Ma następujące właściwości.

Właściwość	Typ	Opis
iconid	ID	ID ikony używanej przez mapowanie ikon. Zachowanie właściwości: - *wymagane*
expression	string	Wyrażenie służące do dopasowania do pola inwentarza. Zachowanie właściwości: - *wymagane*
inventory_link	integer	ID pola inwentarza hosta. Listę obsługiwanych pól inwentarza można znaleźć w obiekcie inwentarza hosta. Zachowanie właściwości: - *wymagane*
sortorder	integer	Pozycja mapowania ikon na mapie ikon. Zachowanie właściwości: - *tylko do odczytu*

iconmap.create

Opis

`object iconmap.create(object/array iconMaps)`

Ta metoda umożliwia tworzenie nowych map ikon.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Mapy ikon do utworzenia.

Oprócz standardowych właściwości map ikon, metoda akceptuje następujące parametry.

Parametr	Type	Opis
mappings	array	Mapowania ikon, które mają zostać utworzone dla mapy ikon. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych map ikon we właściwości `iconmapids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych map ikon.

Przykłady

Utwórz mapę ikon

Utwórz mapę ikon, aby wyświetlać hosty różnych typów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.create",
    "params": {
        "name": "Type icons",
        "default_iconid": "2",
        "mappings": [
            {
                "inventory_link": 1,
                "expression": "server",
                "iconid": "3"
            },
            {
                "inventory_link": 1,
                "expression": "switch",
                "iconid": "4"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2"
        ]
```

```
    },
    "id": 1
}
```

Zobacz także

- Mapowanie ikon

Źródło

CIconMap::create() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.delete

Opis

`object iconmap.delete(array iconMapIds)`

Ta metoda umożliwia usuwanie map ikon.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID map ikon do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych map ikon we właściwości `iconmapids`.

Przykłady

Usuń wiele map ikon

Usuń dwie mapy ikon.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.delete",
    "params": [
        "2",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CIconMap::delete() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.get

Opis

`integer/array iconmap.get(object parameters)`

Ta metoda umożliwia pobieranie map ikon zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
iconmapids	ID/array	Zwraca tylko mapy ikon o podanych identyfikatorach.
sysmapids	ID/array	Zwraca tylko mapy ikon używane w podanych mapach.
selectMappings	query	Zwraca właściwość `mappings` z używanymi mapowaniami ikon.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `iconmapid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie mapy ikon

Pobierz wszystkie dane o mapie ikon „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.get",
    "params": {
        "iconmapids": "3",
        "output": "extend",
        "selectMappings": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mappings": [
                {
                    "iconmappingid": "3",
                    "iconmapid": "3",
                    "iconid": "6",
                    "inventory_link": "1",
                    "expression": "server",
                    "sortorder": "0"
                },
                {
                    "iconmappingid": "4",
                    "iconmapid": "3",
                    "iconid": "10",
                    "inventory_link": "1",
                    "expression": "switch",
                    "sortorder": "1"
                }
            ],
            "iconmapid": "3",
            "name": "Host type icons",
            "default_iconid": "2"
        }
    ],
    "id": 1
}
```


Zobacz także

- Mapowanie ikon

Źródło

CIconMap::get() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.update

Opis

`object iconmap.update(object/array iconMaps)`

Ta metoda umożliwia aktualizację istniejących map ikon.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

`(object/array)` Właściwości mapy ikon do zaktualizowania.

Właściwość `iconmapid` musi być zdefiniowana dla każdej mapy ikon, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości mapy ikon metoda akceptuje następujące parametry.

Parametr	Type	Opis
mappings	array	Mapowania ikon zastępujące istniejące mapowania ikon.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych map ikon we właściwości `iconmapids`.

Przykłady

Zmień nazwę mapy ikon

Zmień nazwę mapy ikon na „OS icons”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.update",
    "params": {
        "iconmapid": "1",
        "name": "OS icons"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Mapowanie ikon

Źródło

CIconMap::update() w *ui/include/classes/api/services/CIconMap.php*.

Mapa wartości

Ta klasa jest przeznaczona do pracy z mapami wartości.

Odwołania do obiektów:

- Mapa wartości
  - Mapowania wartości

Dostępne metody:

- valuemap.create - tworzenie nowych map wartości
- valuemap.delete - usuwanie map wartości
- valuemap.get - pobieranie map wartości
- valuemap.update - aktualizowanie map wartości

Obiekt mapy wartości

Następujące obiekty są bezpośrednio powiązane z API `valuemap`.

Mapa wartości

Obiekt mapy wartości ma następujące właściwości.

Właściwość	Typ	Opis
valuemapid	ID	ID mapy wartości. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
hostid	ID	ID hosta lub szablonu, do którego należy mapa wartości. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy wartości. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
mappings	array	Mapowania wartości dla bieżącej mapy wartości. Obiekt mapowania jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych map wartości z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli mapa wartości należy do szablonu

Mapowania wartości

Obiekt mapowań wartości definiuje mapowania wartości mapy wartości. Ma następujące właściwości.

Właściwość	Type	Opis
type	integer	Typ dopasowania mapowania. Możliwe wartości: 0 - *(domyślnie)* mapowanie zostanie zastosowane, jeśli wartość jest równa; 1 - mapowanie zostanie zastosowane, jeśli wartość jest większa lub równa[1]; 2 - mapowanie zostanie zastosowane, jeśli wartość jest mniejsza lub równa[1]; 3 - mapowanie zostanie zastosowane, jeśli wartość mieści się w zakresie (zakresy są domknięte; można zdefiniować wiele zakresów oddzielonych przecinkiem)[1]; 4 - mapowanie zostanie zastosowane, jeśli wartość pasuje do wyrażenia regularnego[2]; 5 - jeśli nie zostaną znalezione żadne dopasowania, mapowanie nie zostanie zastosowane i zostanie użyta wartość domyślna. Jeśli `type` jest ustawione na ”0”, ”1”, ”2”, ”3”, ”4”, wówczas `value` nie może być puste. Jeśli `type` jest ustawione na ”5”, wówczas `value` musi być puste.
value	string	Wartość oryginalna. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”1”, ”2”, ”3”, ”4” - *obsługiwane* jeśli `type` jest ustawione na ”5”
newvalue	string	Wartość, na którą mapowana jest wartość oryginalna. Zachowanie właściwości: - *wymagane*

[1] obsługiwane tylko dla pozycji mających typ wartości ”numeric unsigned”, ”numeric float”.
 [2] obsługiwane tylko dla pozycji mających typ wartości ”character”.

valuemap.create

Opis

`object valuemap.create(object/array valuemaps)`

Ta metoda umożliwia tworzenie nowych map wartości.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Mapowania wartości do utworzenia.

Metoda akceptuje mapowania wartości ze standardowymi właściwościami mapowania wartości.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory właściwości `valuemapids` utworzonych map wartości. Kolejność zwróconych identyfikatorów jest zgodna z kolejnością przekazywanych map wartości.

Przykłady

Tworzenie mapowania wartości

Utwórz jedno mapowanie wartości z dwoma odwzorowaniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.create",
    "params": {
        "hostid": "50009",
        "name": "Stan usługi",
        "mappings": [
            {
                "type": "1",
                "value": "1",
                "newvalue": "Działa"
            },
            {
                "type": "5",
                "newvalue": "Niedostępna"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CValueMap::create() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.delete

Opis

`object valuemap.delete(array valuemapids)`

Ta metoda pozwala na usuwanie map wartości.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` Identyfikatory map wartości do usunięcia.

Zwracane wartości

`(array)` Zwraca obiekt zawierający identyfikatory usuniętych map wartości we właściwości `valuemapids`.

Przykłady

Usuwanie wielu map wartości

Usuń dwie mapy wartości.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "valuemap.delete",
"params": [
"1",
"2"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"valuemapids": [
"1",
"2"
]
},
"id": 1
}
```

Źródło

CValueMap::delete() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.get

Opis

`integer/array valuemap.get(object parameters)`

Ta metoda umożliwia pobieranie map wartości zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

(object) Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
valuemapids	ID/array	Zwracaj tylko mapowania wartości o podanych identyfikatorach.
selectMappings	query	Zwracaj mapowania wartości dla bieżącej mapy wartości we właściwości `mappings`. Obsługuje `count`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `valuemapid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwróć wartości

(integer/array) Zwraca albo:

- tablicę obiektów;
- ilość pobranych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobieranie map wartości

Pobierz wszystkie skonfigurowane mapy wartości.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobierz jedną mapę wartości wraz z jej mapowaniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend",
        "selectMappings": "extend",
        "valuemapids": ["4"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CValueMap::get() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.update

Opis

`object valuemap.update(object/array valuemaps)`

Ta metoda umożliwia aktualizację istniejących map wartości.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości mapy wartości, które mają zostać zaktualizowane.

Właściwość `valuemapid` musi być zdefiniowana dla każdej mapy wartości, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID zaktualizowanych wartości map we własności `valuemapids`.

Przykłady

Zmiana nazwy mapy wartości

Zmień nazwę mapy wartości na „Device status”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "name": "Device status"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Zmienianie mapowań dla jednej mapy wartości.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
```

```
        "valuemapid": "2",
        "mappings": [
            {
                "type": "0",
                "value": "0",
                "newvalue": "Online"
            },
            {
                "type": "0",
                "value": "1",
                "newvalue": "Offline"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CValueMap::update() w *ui/include/classes/api/services/CValueMap.php*.

MFA

Ta klasa jest przeznaczona do pracy z metodami MFA (uwierzytelnianie wieloskładnikowe).

Odwołania do obiektów:

- MFA

Dostępne metody:

- mfa.create - tworzenie nowych metod MFA
- mfa.delete - usuwanie metod MFA
- mfa.get - pobieranie metod MFA
- mfa.update - aktualizowanie metod MFA

Obiekt MFA (Multi-Factor Authentication)

Następujące obiekty są bezpośrednio powiązane z API `mfa`.

MFA

Obiekt MFA (uwierzytelnianie wieloskładnikowe) ma następujące właściwości.

Właściwość	Typ	Opis
mfaid	ID	ID metody MFA. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
type	integer	Typ metody MFA. Możliwe wartości: 1 - TOTP (jednorazowe hasła oparte na czasie); 2 - Duo Universal Prompt.
name	string	Unikalna nazwa metody MFA. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
hash_function	integer	Typ funkcji skrótu używanej do generowania kodów TOTP. Możliwe wartości: 1 - SHA-1; 2 - SHA-256; 3 - SHA-512. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „TOTP”
code_length	integer	Długość kodu weryfikacyjnego. Możliwe wartości: 6 - 6 cyfr; 8 - 8 cyfr. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „TOTP”
api_hostname	string	Nazwa hosta API dostarczona przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”
clientid	string	ID klienta dostarczone przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”
client_secret	string	Tajny klucz klienta dostarczony przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”

mfa.create

Opis

`object mfa.create(object/array MFA methods)`

Ta metoda umożliwia tworzenie nowych metod MFA.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Metody MFA do utworzenia.

Ta metoda akceptuje metody MFA ze standardowymi właściwościami metod MFA.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych metod MFA we właściwości `mfaids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pozycji.

Przykłady

Tworzenie metod MFA

Utwórz metodę MFA „Zabbix TOTP” wykorzystującą jednorazowe hasła oparte na czasie (TOTP), z funkcją skrótu do generowania kodów TOTP ustawioną na SHA-1 oraz długością kodu weryfikacyjnego ustawioną na 6 cyfr.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.create",
    "params": {
        "type": 1,
        "name": "Zabbix TOTP",
        "hash_function": 1,
        "code_length": 6
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::create() w *ui/include/classes/api/services/CMfa.php*.

mfa.delete

Opis

`object mfa.delete(array mfaids)`

Ta metoda umożliwia usuwanie metod MFA.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(array)` ID metod MFA do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych metod MFA we właściwości `mfaids`.

Przykłady

Usuwanie metod MFA

Usuń metodę MFA.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.delete",
    "params": [
        "2"
    ],
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::delete() w *ui/include/classes/api/services/CMfa.php*.

mfa.get

Opis

`integer/array mfa.get(object parameters)`

Ta metoda umożliwia pobieranie metod MFA zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
mfaids	ID/array	Zwracaj tylko metody MFA o podanych ID.
selectUsrgrps	query	Zwraca właściwość `usrgrps` z grupami użytkowników powiązanymi z metodami MFA. Obsługuje `count`.
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Obsługiwane właściwości: `mfaid` - ID metody MFA; `type` - typ metody MFA.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługiwane właściwości: `name`.
countOutput excludeSearch	boolean boolean	Te parametry są opisane w komentarzu referencyjnym.

Parameter	Type	Description
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Wyszukiwanie metod MFA po nazwie

Pobierz wszystkie metody MFA, które mają w nazwie „Zabbix”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.get",
    "params": {
        "output": "extend",
        "search": {
            "name": "Zabbix"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mfaid": "1",
            "type": "1",
            "name": "Zabbix TOTP 1",
            "hash_function": "1",
            "code_length": "6",
            "api_hostname": "",
            "clientid": ""
        },
        {
            "mfaid": "2",
            "type": "1",
            "name": "Zabbix TOTP 2",
            "hash_function": "3",
            "code_length": "8",
            "api_hostname": "",
            "clientid": ""
        }
    ],
    "id": 1
}
```

Źródło

CMfa::get() w *ui/include/classes/api/services/CMfa.php*.

mfa.update

Opis

`object mfa.update(object/array MFA methods)`

Ta metoda umożliwia aktualizację istniejących metod MFA.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Właściwości metody MFA do zaktualizowania.

Właściwość `mfaid` musi być zdefiniowana dla każdej pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje metody MFA ze standardowymi właściwościami metody MFA.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych metod MFA we właściwości `mfaids`.

Przykłady

Aktualizowanie właściwości metody

Zaktualizuj funkcję skrótu używaną do generowania kodów TOTP oraz długość kodu weryfikacyjnego dla metody MFA „Zabbix TOTP” wykorzystującej jednorazowe hasła oparte na czasie (TOTP).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.update",
    "params": {
        "mfaid": "1",
        "hash_function": 3,
        "code_length": 8
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::update() w *ui/include/classes/api/services/CMfa.php*.

Moduły

Ta klasa została zaprojektowana do współpracy z modułami frontend.

Odwołania do obiektów:

- Module

Dostępne metody:

- module.create - instalowanie nowych modułów
- module.delete - odinstalowywanie modułów
- module.get - pobieranie modułów
- module.update - aktualizowanie modułów

Obiekt modułu

Następujące obiekty są bezpośrednio powiązane z API `module`.

Moduł

Obiekt modułu ma następujące właściwości.

Właściwość	Typ	Opis
moduleid	ID	ID modułu zapisane w bazie danych. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
id	string	Unikalny identyfikator modułu zdefiniowany przez dewelopera w pliku manifest.json modułu. Możliwe wartości dla wbudowanych modułów: zobacz opis właściwości „type” w widżecie pulpitu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
relative_path	string	Ścieżka do katalogu modułu względem katalogu frontend Zabbix. Możliwe wartości: `widgets/*` - dla wbudowanych modułów widżetów; `modules/*` - dla modułów innych firm. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
status	integer	Określa, czy moduł jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* Wyłączony; 1 - Włączony.
config	object	Konfiguracja modułu.

module.create

Opis

`object module.create(object/array modules)`

Ta metoda umożliwia instalację nowych modułów frontend.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w Rolach użytkowników.

Attention:
Pliki modułów muszą zostać ręcznie rozpakowane do odpowiednich podkatalogów, zgodnie z właściwością `relative_path` modułów.

Parametry

`(object/array)` Moduły do utworzenia.

Metoda akceptuje moduły ze standardowymi właściwościami modułu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zainstalowanych modułów we właściwości `moduleids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych modułów.

Przykłady

Instalowanie modułu

Zainstaluj moduł ze statusem „Włączony”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.create",
    "params": {
        "id": "example_module",
        "relative_path": "modules/example_module",
        "status": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Zobacz również

- Moduł
- Moduły frontendowe

Źródło

CModule::create() w *ui/include/classes/api/services/CModule.php*.

module.delete

Opis

`object module.delete(array moduleids)`

Ta metoda umożliwia odinstalowanie modułów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w User roles.

> **Attention:**
> Pliki modułów muszą zostać usunięte ręcznie.

Parametry

`(array)` ID modułów do odinstalowania.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory odinstalowanych modułów we właściwości `moduleids`.

Przykłady

Odinstalowywanie wielu modułów

Odinstaluj moduły „27” i „28”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.delete",
    "params": [
        "27",
        "28"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "27",
            "28"
        ]
    },
    "id": 1
}
```

Źródło

CModule::delete() w *ui/include/classes/api/services/CModule.php*.

module.get

Opis

`integer/array module.get(object parameters)`

Ta metoda umożliwia pobieranie modułów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna wyłącznie dla użytkownika typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
moduleids	ID/array	Zwraca tylko moduły o podanych identyfikatorach.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `moduleid`, `relative_path`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie modułu według ID

Pobierz wszystkie dane o modułach „1”, „2” i „25”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.get",
    "params": {
        "output": "extend",
        "moduleids": [
            "1",
            "2",
            "25"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "moduleid": "1",
            "id": "actionlog",
            "relative_path": "widgets/actionlog",
            "status": "1",
            "config": []
        },
        {
            "moduleid": "2",
            "id": "clock",
            "relative_path": "widgets/clock",
            "status": "1",
            "config": []
        },
        {
            "moduleid": "25",
            "id": "example",
            "relative_path": "modules/example_module",
            "status": "1",
            "config": []
        }
    ],
    "id": 1
}
```

Zobacz także

- Moduł
- Widget pulpitu
- Moduły frontend

Źródło

CModule::get() w *ui/include/classes/api/services/CModule.php*.

module.update

Opis

`object module.update(object/array modules)`

Ta metoda umożliwia aktualizację istniejących modułów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości modułu do zaktualizowania.

Właściwość `moduleid` musi być zdefiniowana dla każdego modułu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko określone właściwości.

Metoda akceptuje moduły ze standardowymi właściwościami modułu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych modułów we właściwości `moduleids`.

Przykłady

Wyłączanie modułu

Wyłącz moduł „25”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.update",
    "params": {
        "moduleid": "25",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Zobacz także

- Moduł
- Moduły frontend

Źródło

CModule::update() w *ui/include/classes/api/services/CModule.php*.

Obraz

Ta klasa została zaprojektowana do pracy z obrazami.

Odwołania do obiektów:

- Image

Dostępne metody:

- image.create - tworzenie nowych obrazów
- image.delete - usuwanie obrazów
- image.get - pobieranie obrazów
- image.update - aktualizacja obrazów

Obiekt obrazu

Następujące obiekty są bezpośrednio powiązane z API `image`.

Obraz

Obiekt obrazu ma następujące właściwości.

Właściwość	Typ	Opis
imageid	ID	ID obrazu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa obrazu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
imagetype	integer	Typ obrazu. Możliwe wartości: 1 - *(domyślnie)* ikona; 2 - obraz tła. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
image	string	Obraz zakodowany w Base64. Maksymalny rozmiar zakodowanego obrazu wynosi 1 MB. Maksymalny rozmiar można dostosować, zmieniając wartość stałej `ZBX_MAX_IMAGE_SIZE`. Obsługiwane formaty obrazów: PNG, JPEG, GIF i WebP. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

image.create

Opis

`object image.create(object/array images)`

Ta metoda umożliwia tworzenie nowych obrazów.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Obrazy do utworzenia.

Metoda akceptuje obrazy ze standardowymi właściwościami obrazu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych obrazów we właściwości `imageids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych obrazów.

Przykłady

Utwórz obraz ikony

Utwórz ikonę chmury.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 1,
        "name": "Cloud_(24)",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAAApgB
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188"
        ]
    },
    "id": 1
}
```


Utwórz obraz tła

Utwórz mały obraz tła, który będzie używany jako wzorzec.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 2,
        "name": "background_image",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAAYCAIAAABvFaqvAAABhWlDQ1BJQ0MgcHJvZmlsZQAAKJF9kb1Lw0AYxp+m
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "189"
        ]
    },
    "id": 1
}
```


Źródło

CImage::create() w *ui/include/classes/api/services/CImage.php*.

image.delete

Opis

`object image.delete(array imageIds)`

Ta metoda umożliwia usuwanie obrazów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` IDs obrazów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych obrazów w właściwości `imageids`.

Przykłady

Usuń wiele obrazów

Usuń dwa obrazy.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "image.delete",
"params": [
"188",
"192"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"imageids": [
"188",
"192"
]
},
"id": 1
}
```

Źródło

CImage::delete() w *ui/include/classes/api/services/CImage.php*.

image.get

Opis

`integer/array image.get(object parameters)`

Ta metoda umożliwia pobieranie obrazów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
imageids	ID/array	Zwraca tylko obrazy o podanych identyfikatorach.

Parametr	Typ	Opis
sysmapids	ID/array	Zwraca obrazy używane na podanych mapach.
select_image	flag	Zwraca właściwość `image` zawierającą obraz zakodowany w Base64.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `imageid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie obrazu

Pobierz wszystkie dane obrazu o ID „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.get",
    "params": {
        "output": "extend",
        "select_image": true,
        "imageids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "imageid": "2",
            "imagetype": "1",
            "name": "Cloud_(24)",
            "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAA
        }
    ],
    "id": 1
}
```

Źródło

CImage::get() w *ui/include/classes/api/services/CImage.php*.

image.update

Opis

`object image.update(object/array images)`

Ta metoda umożliwia aktualizację istniejących obrazów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości obrazu do zaktualizowania.

Właściwość `imageid` musi być zdefiniowana dla każdego obrazu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje obrazy ze standardowymi właściwościami obrazu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych obrazów we właściwości `imageids`.

Przykłady

Zmień nazwę obrazu

Zmień nazwę obrazu na „Cloud icon”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.update",
    "params": {
        "imageid": "2",
        "name": "Cloud icon"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CImage::update() w *ui/include/classes/api/services/CImage.php*.

Porządkowanie

Ta klasa jest zaprojektowana do pracy z porządkowaniem.

Referencje obiektu:

- Housekeeping

Dostępne metody:

- housekeeping.get - retrieve housekeeping
- housekeeping.update - update housekeeping

Obiekt sprzątania (housekeeping)

Następujące obiekty są bezpośrednio powiązane z API `housekeeping`.

Housekeeping

Obiekt ustawień ma następujące właściwości.

Właściwość	Typ	Opis
hk_events_mode	integer	Włącza wewnętrzny housekeeping dla zdarzeń i alertów. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_events_trigger	string	Okres przechowywania danych wyzwalaczy. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 365d.
hk_events_service	string	Okres przechowywania danych usług. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 1d.
hk_events_internal	string	Okres przechowywania danych wewnętrznych. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 1d.
hk_events_discovery	string	Okres przechowywania danych wykrywania sieci. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 1d.
hk_events_autoreg	string	Okres przechowywania danych autorejestracji. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 1d.
hk_services_mode	integer	Włącza wewnętrzny housekeeping dla usług. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_services	string	Okres przechowywania danych usług. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 365d.
hk_audit_mode	integer	Włącza wewnętrzny housekeeping dla audytu. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_audit	string	Okres przechowywania danych audytu. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 31d.
hk_sessions_mode	integer	Włącza wewnętrzny housekeeping dla sesji. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_sessions	string	Okres przechowywania danych sesji. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 31d.

Właściwość	Typ	Opis
hk_history_mode	integer	Włącza wewnętrzny housekeeping dla historii. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_history_global	integer	Zastępuje okres historii pozycji. Możliwe wartości: 0 - Nie zastępuj; 1 - *(domyślnie)* Zastąp.
hk_history	string	Okres przechowywania danych historii. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 31d.
hk_trends_mode	integer	Włącza wewnętrzny housekeeping dla trendów. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
hk_trends_global	integer	Zastępuje okres trendów pozycji. Możliwe wartości: 0 - Nie zastępuj; 1 - *(domyślnie)* Zastąp.
hk_trends	string	Okres przechowywania danych trendów. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 365d.
db_extension	string	Flaga konfiguracji rozszerzenia DB. Jeśli ta flaga jest ustawiona na ”timescaledb”, serwer zmienia swoje zachowanie w zakresie housekeeping i usuwania pozycji. Zachowanie właściwości: - *tylko do odczytu*
compression_availability	integer	Czy kompresja danych jest obsługiwana przez bazę danych (lub jej rozszerzenie). Możliwe wartości: 0 - Niedostępna; 1 - Dostępna. Zachowanie właściwości: - *tylko do odczytu*
compression_status	integer	Włącza kompresję TimescaleDB dla historii i trendów. Możliwe wartości: 0 - *(domyślnie)* Wyłączona; 1 - Włączona.
compress_older	string	Kompresuje rekordy historii i trendów starsze niż określony okres. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Domyślnie: 7d.

housekeeping.get

Opis

`object housekeeping.get(object parameters)`

Ta metoda umożliwia pobranie obiektu housekeeping zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników każdego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Wartości zwracane

`(object)` Zwraca obiekt housekeeping.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "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": "31d",
        "hk_sessions_mode": "1",
        "hk_sessions": "365d",
        "hk_history_mode": "1",
        "hk_history_global": "0",
        "hk_history": "31d",
        "hk_trends_mode": "1",
        "hk_trends_global": "0",
        "hk_trends": "365d",
        "db_extension": "",
        "compression_status": "0",
        "compress_older": "7d"
    },
    "id": 1
}
```

Źródło

CHousekeeping ::get() w *ui/include/classes/api/services/CHousekeeping.php*.

housekeeping.update

Opis

`object housekeeping.update(object housekeeping)`

Ta metoda umożliwia aktualizację istniejących ustawień housekeeping.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Właściwości housekeeping do zaktualizowania.

Zwracane wartości

`(array)` Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.update",
    "params": {
        "hk_events_mode": "1",
        "hk_events_trigger": "200d",
        "hk_events_internal": "2d",
        "hk_events_discovery": "2d"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "hk_events_mode",
        "hk_events_trigger",
        "hk_events_internal",
        "hk_events_discovery"
    ],
    "id": 1
}
```

Źródło

CHousekeeping::update() w *ui/include/classes/api/services/CHousekeeping.php*.

Pozycja

Ta klasa jest przeznaczona do pracy z pozycjami.

Odwołania do obiektów:

- Pozycja
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Tag pozycji
- Przetwarzanie wstępne pozycji

Dostępne metody:

- item.create - utwórz nowe pozycje
- item.delete - usuń pozycje

- item.get - pobierz pozycje
- item.update - zaktualizuj pozycje

Obiekt pozycji

Następujące obiekty są bezpośrednio powiązane z API `item`.

Item

Note:
Pozycji webowych nie można bezpośrednio tworzyć, aktualizować ani usuwać za pomocą API Zabbixa.

Obiekt item ma następujące właściwości.

Property	Type	Description
itemid	ID	ID pozycji. Property behavior: - *read-only* - *required* for update operations
delay	string	Interwał aktualizacji pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej niestandardowych interwałów, wszystkie oddzielone średnikami. Niestandardowe interwały mogą być mieszanką interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisać jako dwa makra oddzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Property behavior: - *required* if `type` is set to ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”Calculated” (15), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), or if `type` is set to ”Zabbix agent (active)” (7) and `key_` does not contain ”mqtt.get”
hostid	ID	ID hosta lub szablonu, do którego należy pozycja. Property behavior: - *constant* - *required* for create operations
interfaceid	ID	ID interfejsu hosta dla pozycji. Property behavior: - *required* if item belongs to host and `type` is set to ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap”, or ”SNMP agent” - *supported* if item belongs to host and `type` is set to ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent”, or ”HTTP agent” - *read-only* for discovered objects
key_	string	Klucz pozycji. Property behavior: - *required* for create operations - *read-only* for inherited objects or discovered objects

Property	Type	Description
name	string	Nazwa pozycji. Obsługuje makra użytkownika. Property behavior: - *required* for create operations - *read-only* for inherited objects or discovered objects
name_resolved	string	Nazwa pozycji z rozwiniętymi makrami użytkownika. Property behavior: - *read-only*
type	integer	Typ pozycji. Possible values: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 9 - Web item; 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 15 - Calculated; 16 - JMX agent; 17 - SNMP trap; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser. Property behavior: - *required* for create operations - *read-only* for inherited objects or discovered objects
url	string	Łańcuch URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Property behavior: - *required* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
value_type	integer	Typ informacji przechowywanej przez pozycję. Possible values: 0 - numeric float; 1 - character; 2 - log; 3 - numeric unsigned; 4 - text; 5 - binary. Property behavior: - *required* for create operations - *read-only* for inherited objects or discovered objects

Property	Type	Description
allow_traps	integer	Pozwala na wypełnianie wartości podobnie jak w przypadku pozycji trapper. 0 - *(default)* Nie zezwalaj na przyjmowanie danych przychodzących; 1 - Zezwalaj na przyjmowanie danych przychodzących. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for discovered objects
authtype	integer	Metoda uwierzytelniania. Possible values if `type` is set to ”SSH agent”: 0 - *(default)* password; 1 - public key. Possible values if `type` is set to ”HTTP agent”: 0 - *(default)* none; 1 - basic; 2 - NTLM; 3 - Kerberos; 4 - Digest. Property behavior: - *supported* if `type` is set to ”SSH agent” or ”HTTP agent” - *read-only* for inherited objects (if `type` is set to ”HTTP agent”) or discovered objects
description	string	Opis pozycji. Property behavior: - *read-only* for discovered objects
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości pozycji. Property behavior: - *read-only*
flags	integer	Pochodzenie pozycji. Possible values: 0 - a plain item; 4 - an item converted from prototype. Property behavior: - *read-only*
follow_redirects	integer	Podążaj za przekierowaniami odpowiedzi podczas pobierania danych. Possible values: 0 - Do not follow redirects; 1 - *(default)* Follow redirects. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects

Property	Type	Description
history	string	Jednostka czasu określająca, jak długo mają być przechowywane dane historyczne. Obsługuje także makro użytkownika. Domyślnie: 31d. Property behavior: - *read-only* for discovered objects
http_proxy	string	Łańcuch połączenia proxy HTTP(S). Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
inventory_link	integer	ID pola inwentarza hosta, które jest wypełniane przez pozycję. Listę obsługiwanych pól inwentarza hosta i ich identyfikatorów znajdziesz na stronie inwentarza hosta. Domyślnie: 0. Property behavior: - *supported* if `value_type` is set to ”numeric float”, ”character”, ”numeric unsigned”, or ”text” - *read-only* for discovered objects
ipmi_sensor	string	Czujnik IPMI. Property behavior: - *required* if `type` is set to ”IPMI agent” and `key_` is not set to ”ipmi.get” - *supported* if `type` is set to ”IPMI agent” - *read-only* for inherited objects or discovered objects
jmx_endpoint	string	Niestandardowy ciąg połączenia agenta JMX. Domyślna wartość: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Property behavior: - *supported* if `type` is set to ”JMX agent” - *read-only* for discovered objects
lastclock	timestamp	Czas ostatniej aktualizacji wartości pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Możesz wydłużyć ten okres, zmieniając wartość parametru *Maksymalny okres wyświetlania historii* w sekcji menu *Administracja → Ogólne*. Property behavior: - *read-only*
lastns	integer	Nanosekundy ostatniej aktualizacji wartości pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Możesz wydłużyć ten okres, zmieniając wartość parametru *Maksymalny okres wyświetlania historii* w sekcji menu *Administracja → Ogólne*. Property behavior: - *read-only*

Property	Type	Description
lastvalue	string	Ostatnia wartość pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Możesz wydłużyć ten okres, zmieniając wartość parametru *Maksymalny okres wyświetlania historii* w sekcji menu *Administracja → Ogólne*. Property behavior: - *read-only*
logtimefmt	string	Format czasu w wpisach dziennika. Property behavior: - *supported* if `value_type` is set to ”log” - *read-only* for inherited objects or discovered objects
master_itemid	ID	ID pozycji nadrzędnej. Property behavior: - *required* if `type` is set to ”Dependent item” - *read-only* for inherited objects or discovered objects
output_format	integer	Czy odpowiedź ma zostać przekonwertowana do JSON. 0 - *(default)* Store raw; 1 - Convert to JSON. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
params	string	Dodatkowe parametry zależne od typu pozycji: - wykonany skrypt dla pozycji SSH agent i TELNET agent; - zapytanie SQL dla pozycji database monitor; - formuła dla pozycji calculated; - skrypt dla pozycji script i browser. Property behavior: - *required* if `type` is set to ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script”, or ”Browser” - *read-only* for inherited objects (if `type` is set to ”Script” or ”Browser”) or discovered objects
parameters	object/array	Dodatkowe parametry, jeśli `type` jest ustawiony na ”Script” lub ”Browser”. Tablica obiektów z właściwościami `name` i `value`, przy czym `name` musi być unikalne. Property behavior: - *supported* if `type` is set to ”Script” or ”Browser” - *read-only* for inherited objects or discovered objects
password	string	Hasło do uwierzytelniania. Property behavior: - *required* if `type` is set to ”JMX agent” and `username` is set - *supported* if `type` is set to ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor”, or ”HTTP agent” - *read-only* for inherited objects (if `type` is set to ”HTTP agent”) or discovered objects
post_type	integer	Typ treści żądania POST przechowywanej we właściwości `posts`. Possible values: 0 - *(default)* Raw data; 2 - JSON data; 3 - XML data. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects

Property	Type	Description
posts	string	Dane treści żądania HTTP(S). Property behavior: - *required* if `type` is set to ”HTTP agent” and `post_type` is set to ”JSON data” or ”XML data” - *supported* if `type` is set to ”HTTP agent” and `post_type` is set to ”Raw data” - *read-only* for inherited objects or discovered objects
prevvalue	string	Poprzednia wartość pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Możesz wydłużyć ten okres, zmieniając wartość parametru *Maksymalny okres wyświetlania historii* w sekcji menu *Administracja → Ogólne*. Property behavior: - *read-only*
privatekey	string	Nazwa pliku klucza prywatnego. Property behavior: - *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” - *read-only* for discovered objects
publickey	string	Nazwa pliku klucza publicznego. Property behavior: - *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” - *read-only* for discovered objects
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
request_method	integer	Typ metody żądania. Possible values: 0 - *(default)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
retrieve_mode	integer	Która część odpowiedzi ma zostać zapisana. Possible values if `request_method` is set to ”GET”, ”POST”, or ”PUT”: 0 - *(default)* Body; 1 - Headers; 2 - Both body and headers will be stored. Possible values if `request_method` is set to ”HEAD”: 1 - Headers. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects

Property	Type	Description
snmp_oid	string	OID SNMP. Property behavior: - *required* if `type` is set to ”SNMP agent” - *read-only* for inherited objects or discovered objects
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
ssl_key_password	string	Hasło do pliku klucza SSL. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
state	integer	Stan pozycji. Possible values: 0 - *(default)* normal; 1 - not supported. Property behavior: - *read-only*
status	integer	Status pozycji. Possible values: 0 - *(default)* enabled item; 1 - disabled item.
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługuje także makra użytkownika jako część listy rozdzielonej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
templateid	ID	ID nadrzędnego elementu szablonu. *Wskazówka*: Użyj właściwości `hostid`, aby określić szablon, do którego należy pozycja. Property behavior: - *read-only*

Property	Type	Description
timeout	string	Limit czasu żądania pobierania danych pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m). Akceptuje także makra użytkownika. Zakres możliwych wartości: 1-600s. Domyślnie: "" - użyj ustawień proxy/globalnych. Property behavior: - *supported* if `type` is set to "Zabbix agent" (0), "Simple check" (3) and `key_` does not start with "`vmware.`" and "`icmpping`", "Zabbix agent (active)" (7), "External check" (10), "Database monitor" (11), "SSH agent" (13), "TELNET agent" (14), "HTTP agent" (19), "SNMP agent" (20) and `snmp_oid` starts with "`walk[`" or "`get[`", "Script" (21), "Browser" (22) - *read-only* for inherited and discovered objects
trapper_hosts	string	Dozwolone hosty. Property behavior: - *readonly* for discovered objects - *supported* if `type` is set to "Zabbix trapper", or if `type` is set to "HTTP agent" and `allow_traps` is set to "Allow to accept incoming data"
trends	string	Jednostka czasu określająca, jak długo mają być przechowywane dane trendów. Obsługuje także makro użytkownika. Domyślnie: 365d. Property behavior: - *supported* if `value_type` is set to "numeric float" or "numeric unsigned" - *read-only* for discovered objects
units	string	Jednostki wartości. Property behavior: - *supported* if `value_type` is set to "numeric float" or "numeric unsigned" - *read-only* for inherited objects or discovered objects
username	string	Nazwa użytkownika do uwierzytelniania. Property behavior: - *required* if `type` is set to "SSH agent", "TELNET agent", or if `type` is set to "JMX agent" and `password` is set - *supported* if `type` is set to "Simple check", "Database monitor", or "HTTP agent" - *read-only* for inherited objects (if `type` is set to "HTTP agent") or discovered objects
uuid	string	Uniwersalny unikalny identyfikator, używany do powiązania importowanej pozycji z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany. Property behavior: - *supported* if the item belongs to a template
valuemapid	ID	ID powiązanej mapy wartości. Property behavior: - *supported* if `value_type` is set to "numeric float", "character", or "numeric unsigned" - *read-only* for inherited objects or discovered objects

Property	Type	Description
verify_host	integer	Czy sprawdzać, czy nazwa hosta dla połączenia zgadza się z nazwą w certyfikacie hosta. Possible values: 0 - *(default)* Do not validate; 1 - Validate. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects
verify_peer	integer	Czy sprawdzać, czy certyfikat hosta jest autentyczny. Possible values: 0 - *(default)* Do not validate; 1 - Validate. Property behavior: - *supported* if `type` is set to ”HTTP agent” - *read-only* for inherited objects or discovered objects

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Type	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Tag pozycji

Obiekt tagu pozycji ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu pozycji. Zachowanie właściwości: - *required*
value	string	Wartość tagu pozycji.

Przetwarzanie wstępne pozycji

Obiekt przetwarzania wstępnego pozycji ma następujące właściwości.

Property	Type	Description
type	integer	Typ opcji przetwarzania wstępnego. Możliwe wartości: 1 - Niestandardowy mnożnik; 2 - Przycinanie z prawej; 3 - Przycinanie z lewej; 4 - Przycinanie; 5 - Wyrażenie regularne; 6 - Wartość logiczna na dziesiętną; 7 - Ósemkowa na dziesiętną; 8 - Szesnastkowa na dziesiętną; 9 - Prosta zmiana; 10 - Zmiana na sekundę; 11 - XML XPath; 12 - JSONPath; 13 - W zakresie; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 18 - Sprawdzenie błędu przy użyciu wyrażenia regularnego; 19 - Odrzuć niezmienione; 20 - Odrzuć niezmienione z heartbeat; 21 - JavaScript; 22 - Wzorzec Prometheus; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąp; 26 - Sprawdź jako nieobsługiwane; 27 - XML do JSON; 28 - Wartość z przebiegu SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Zachowanie właściwości: - *wymagane*

Property	Type	Description
params	string	Dodatkowe parametry używane przez opcję przetwarzania wstępnego. Wiele parametrów jest oddzielanych znakiem nowej linii (\n). Jeśli `type` jest ustawione na ”Check unsupported”, parametry mają składnię `<scope>[\n<pattern>]`, gdzie *pattern* jest wyrażeniem regularnym, a *scope* jest jednym z: -1 - dopasuj dowolny błąd; 0 - sprawdź, czy komunikat błędu pasuje do *pattern*; 1 - sprawdź, czy komunikat błędu nie pasuje do *pattern*. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”Custom multiplier” (1), ”Right trim” (2), ”Left trim” (3), ”Trim” (4), ”Regular expression” (5), ”XML XPath” (11), ”JSONPath” (12), ”In range” (13), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Check for error using regular expression” (18), ”Discard unchanged with heartbeat” (20), ”JavaScript” (21), ”Prometheus pattern” (22), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Replace” (25), Check unsupported (26), ”SNMP walk value” (28), ”SNMP walk to JSON” (29), lub ”SNMP get value” (30)
error_handler	integer	Typ akcji używany w przypadku niepowodzenia kroku przetwarzania wstępnego. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzuć wartość; 2 - Ustaw wartość niestandardową; 3 - Ustaw niestandardowy komunikat błędu. Możliwe wartości, jeśli `type` jest ustawione na ”Check unsupported”: 1 - Odrzuć wartość; 2 - Ustaw wartość niestandardową; 3 - Ustaw niestandardowy komunikat błędu. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”Custom multiplier” (1), ”Regular expression” (5), ”Boolean to decimal” (6), ”Octal to decimal” (7), ”Hexadecimal to decimal” (8), ”Simple change” (9), ”Change per second” (10), ”XML XPath” (11), ”JSONPath” (12), ”In range” (13), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Check for error using regular expression” (18), ”Prometheus pattern” (22), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Check unsupported” (26), ”XML to JSON” (27), ”SNMP walk value” (28), ”SNMP walk to JSON” (29), lub ”SNMP get value” (30)
error_handler_params	string	Parametry obsługi błędów. Zachowanie właściwości: - *wymagane* jeśli `error_handler` jest ustawione na ”Set custom value” lub ”Set custom error message”

Następujące parametry i obsługi błędów są obsługiwane dla każdego typu przetwarzania wstępnego.

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
1	Custom multiplier	number^1, 6^			0, 1, 2, 3

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
2	Right trim	list of characters[2]			
3	Left trim	list of characters[2]			
4	Trim	list of characters[2]			
5	Regular expression	pattern[3]	output[2]		0, 1, 2, 3
6	Boolean to decimal				0, 1, 2, 3
7	Octal to decimal				0, 1, 2, 3
8	Hexadecimal to decimal				0, 1, 2, 3
9	Simple change				0, 1, 2, 3
10	Change per second				0, 1, 2, 3
11	XML XPath	path[4]			0, 1, 2, 3
12	JSONPath	path[4]			0, 1, 2, 3
13	In range	min^1, 6^	max^1, 6^		0, 1, 2, 3
14	Matches regular expression	pattern[3]			0, 1, 2, 3
15	Does not match regular expression	pattern[3]			0, 1, 2, 3
16	Check for error in JSON	path[4]			0, 1, 2, 3
17	Check for error in XML	path[4]			0, 1, 2, 3

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
18	Check for error using regular expression	pattern[3]	output[2]		0, 1, 2, 3
19	Discard unchanged				
20	Discard unchanged with heartbeat	seconds^5, 6^			
21	JavaScript	script[2]			
22	Prometheus pattern	pattern^6, 7^	`value`, `label`, `function`	output^8, 9^	0, 1, 2, 3
23	Prometheus to JSON	pattern^6, 7^			0, 1, 2, 3
24	CSV to JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Replace	search string[2]	replacement[2]		
26	Check unsupported	scope[1]	pattern^3, 6^		1, 2, 3
27	XML to JSON				0, 1, 2, 3
28	SNMP walk value	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS		0, 1, 2, 3
29	SNMP walk to JSON[10]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS	0, 1, 2, 3
30	SNMP get value	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS			0, 1, 2, 3

[1] liczba zmiennoprzecinkowa (liczby całkowite są niejawnie konwertowane do wartości zmiennoprzecinkowych)
 [2] ciąg znaków
 [3] wyrażenie regularne
 [4] JSONPath lub XML XPath
 [5] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [6] makro użytkownika
 [7] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy element wzorca Prometheus (nazwa metryki, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [8] wynik Prometheus zgodny ze składnią: `<label name>` (może być makrem użytkownika), jeśli jako drugi parametr wybrano `label`.
 [9] jedna z funkcji agregujących: `sum`, `min`, `max`, `avg`, `count`, jeśli jako drugi parametr wybrano `function`.
 [10] obsługuje wiele rekordów "Field name,OID prefix,Format" rozdzielonych znakiem nowej linii.

item.create

Opis

`object item.create(object/array items)`

Ta metoda umożliwia tworzenie nowych pozycji.

> **Note:**
> Pozycje web nie mogą być tworzone za pomocą Zabbix API.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Pozycje do utworzenia.

Oprócz standardowych właściwości pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
preprocessing	array	Opcje wstępnego przetwarzania pozycji.
tags	array	Tagi pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pozycji we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pozycji.

Przykłady

Tworzenie pozycji

Utwórz numeryczną pozycję typu Zabbix agent z 2 tagami pozycji do monitorowania wolnego miejsca na dysku na hoście o ID „30074".

Żądanie:

```
{
    "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": "component",
                "value": "storage"
            },
            {
                "tag": "equipment",
```

```
                "value": "workstation"
            }
        ],
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24758"
        ]
    },
    "id": 1
}
```

Tworzenie pozycji inwentaryzacji hosta

Utwórz pozycję agenta Zabbix, aby wypełnić pole inwentaryzacji hosta „OS”.

Żądanie:

```
{
    "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
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 1
}
```

Tworzenie pozycji z przetwarzaniem wstępnym

Utwórz pozycję przy użyciu niestandardowego mnożnika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Device uptime",
        "key_": "sysUpTime",
        "hostid": "10084",
        "type": 20,
```

```
        "snmp_oid": "SNMPv2-MIB::sysUpTime.0",
        "value_type": 1,
        "delay": "60s",
        "interfaceid": "83",
        "preprocessing": [
            {
                "type": 1,
                "params": "0.01",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44210"
        ]
    },
    "id": 1
}
```

Tworzenie zależnej pozycji

Utwórz zależną pozycję dla nadrzędnej pozycji o ID 24759. Dozwolone są tylko zależności w obrębie tego samego hosta, dlatego nadrzędna i zależna pozycja powinny mieć ten sam hostid.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
      "hostid": "30074",
      "name": "Dependent test item",
      "key_": "dependent.item",
      "type": 18,
      "master_itemid": "24759",
      "value_type": 2
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Utwórz pozycję HTTP agent

Utwórz pozycję z metodą żądania POST z preprocessingiem odpowiedzi JSON.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "url":"http://127.0.0.1/http.php",
        "query_fields": [
            {
                "name": "mode",
                "value": "json"
            },
            {
                "name": "min",
                "value": "10"
            },
            {
                "name": "max",
                "value": "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": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Utwórz pozycję skryptową

Utwórz proste zbieranie danych za pomocą pozycji skryptowej.

Żądanie:

```
{
    "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 HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
        "parameters": [
            {
                "name": "host",
                "value": "{HOST.CONN}"
            }
        ],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItem::create() w *ui/include/classes/api/services/CItem.php*.

item.delete

Opis

`object item.delete(array itemIds)`

Ta metoda umożliwia usuwanie pozycji.

Note:
Pozycje web nie mogą być usuwane za pomocą Zabbix API.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID pozycji do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pozycji we właściwości `itemids`.

Przykłady

Usuwanie wielu pozycji

Usuń dwie pozycje.
 Pozycje zależne i prototypy pozycji są usuwane automatycznie, jeśli główna pozycja zostanie usunięta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.delete",
    "params": [
        "22982",
        "22986"
    ],
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22982",
            "22986"
        ]
    },
    "id": 1
}
```

Źródło

CItem::delete() w *ui/include/classes/api/services/CItem.php*.

item.get

Opis

`integer/array item.get(object parameters)`

Metoda umożliwia pobieranie pozycji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
itemids	ID/array	Zwracaj tylko pozycje o podanych identyfikatorach.
groupids	ID/array	Zwracaj tylko pozycje należące do hostów z podanych grup.
templateids	ID/array	Zwracaj tylko pozycje należące do podanych szablonów.
hostids	ID/array	Zwracaj tylko pozycje należące do podanych hostów.
proxyids	ID/array	Zwracaj tylko pozycje monitorowane przez podane proxy.
interfaceids	ID/array	Zwracaj tylko pozycje korzystające z podanych interfejsów hosta.
graphids	ID/array	Zwracaj tylko pozycje używane w podanych wykresach.
triggerids	ID/array	Zwracaj tylko pozycje używane w podanych wyzwalaczach.
webitems	flag	Uwzględnij web items w wyniku.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko pozycje odziedziczone z szablonu.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko pozycje należące do szablonów.
monitored	boolean	Jeśli ustawione na `true`, zwracaj tylko włączone pozycje należące do monitorowanych hostów.
group	string	Zwracaj tylko pozycje należące do grupy o podanej nazwie.
host	string	Zwracaj tylko pozycje należące do hosta o podanej nazwie.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.

Parameter	Type	Description
tags	array	Zwracaj tylko pozycje z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie pozycje. Możliwe wartości operatora: 0 - *(domyślnie)* Contains; 1 - Equals; 2 - Does not contain; 3 - Does not equal; 4 - Exists; 5 - Does not exist.
with_triggers	boolean	Jeśli ustawione na `true`, zwracaj tylko pozycje używane w wyzwalaczach.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy pozycja.
selectInterfaces	query	Zwraca właściwość `interfaces` z tablicą interfejsów hosta używanych przez pozycję.
selectTriggers	query	Zwraca właściwość `triggers` z wyzwalaczami, w których używana jest pozycja. Obsługuje `count`.
selectGraphs	query	Zwraca właściwość `graphs` z wykresami, które zawierają pozycję. Obsługuje `count`.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania pozycji. Obiekt wykrywania pozycji łączy wykrytą pozycję z prototypem pozycji, z którego została wykryta. Ma następujące właściwości: `parent_itemid` - `(string)` identyfikator prototypu pozycji, z którego utworzono pozycję; `key_` - `(string)` klucz prototypu pozycji; `status` - `(int)` status wykrywania pozycji: 0 - *(domyślnie)* pozycja została wykryta, 1 - pozycja nie jest już wykrywana; `ts_delete` - `(timestamp)` czas, po którym pozycja, która nie jest już wykrywana, zostanie usunięta; `ts_disable` - `(timestamp)` czas, po którym pozycja, która nie jest już wykrywana, zostanie wyłączona; `disable_source` - `(int)` wskaźnik tego, czy pozycja została wyłączona przez regułę LLD, czy ręcznie: 0 - *(domyślnie)* wyłączona automatycznie, 1 - wyłączona przez regułę LLD.
selectDiscoveryRule	query	Zwraca właściwość `discoveryRule` z regułą LLD, która utworzyła pozycję.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami wstępnego przetwarzania pozycji.
selectTags	query	Zwraca tagi pozycji w właściwości `tags`.
selectValueMap	query	Zwraca właściwość `valuemap` z mapą wartości pozycji.
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Nie obsługuje właściwości typu danych `text`. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy pozycja.

Parameter	Type	Description
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectGraphs` - wyniki będą sortowane według `name`; `selectTriggers` - wyniki będą sortowane według `description`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `history`, `trends`, `type`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectItemDiscovery	query	Zwraca właściwość `itemDiscovery` z obiektem wykrywania pozycji. Obiekt wykrywania pozycji łączy pozycję z prototypem pozycji, z którego została utworzona. To zapytanie jest **przestarzałe**; zamiast niego użyj `selectDiscoveryData`.

Wartości zwracane

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Znajdowanie pozycji według klucza

Pobierz wszystkie pozycje używane w wyzwalaczach dla określonego ID hosta, które zawierają słowo ”system.cpu” w kluczu pozycji, i posortuj wyniki według nazwy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10084",
        "with_triggers": true,
        "search": {
            "key_": "system.cpu"
        },
        "sortfield": "name"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "42269",
```

```
        "type": "18",
        "snmp_oid": "",
        "hostid": "10084",
        "name": "CPU utilization",
        "key_": "system.cpu.util",
        "delay": "0",
        "history": "7d",
        "trends": "365d",
        "status": "0",
        "value_type": "0",
        "trapper_hosts": "",
        "units": "%",
        "logtimefmt": "",
        "templateid": "42267",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "0",
        "description": "CPU utilization in %.",
        "inventory_link": "0",
        "evaltype": "0",
        "jmx_endpoint": "",
        "master_itemid": "42264",
        "timeout": "",
        "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",
        "uuid": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "CPU utilization"
    },
    {
        "itemid": "42259",
        "type": "0",
        "snmp_oid": "",
        "hostid": "10084",
```

```
        "name": "Load average (15m avg)",
        "key_": "system.cpu.load[all,avg15]",
        "delay": "1m",
        "history": "7d",
        "trends": "365d",
        "status": "0",
        "value_type": "0",
        "trapper_hosts": "",
        "units": "",
        "logtimefmt": "",
        "templateid": "42219",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "evaltype": "0",
        "jmx_endpoint": "",
        "master_itemid": "0",
        "timeout": "",
        "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",
        "uuid": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (15m avg)"
    },
    {
        "itemid": "42249",
        "type": "0",
        "snmp_oid": "",
        "hostid": "10084",
        "name": "Load average (1m avg)",
        "key_": "system.cpu.load[all,avg1]",
        "delay": "1m",
```

```
        "history": "7d",
        "trends": "365d",
        "status": "0",
        "value_type": "0",
        "trapper_hosts": "",
        "units": "",
        "logtimefmt": "",
        "templateid": "42209",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "evaltype": "0",
        "jmx_endpoint": "",
        "master_itemid": "0",
        "timeout": "",
        "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",
        "uuid": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (1m avg)"
    },
    {
        "itemid": "42257",
        "type": "0",
        "snmp_oid": "",
        "hostid": "10084",
        "name": "Load average (5m avg)",
        "key_": "system.cpu.load[all,avg5]",
        "delay": "1m",
        "history": "7d",
        "trends": "365d",
        "status": "0",
```

```
        "value_type": "0",
        "trapper_hosts": "",
        "units": "",
        "logtimefmt": "",
        "templateid": "42217",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "evaltype": "0",
        "jmx_endpoint": "",
        "master_itemid": "0",
        "timeout": "",
        "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",
        "uuid": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (5m avg)"
    },
    {
        "itemid": "42260",
        "type": "0",
        "snmp_oid": "",
        "hostid": "10084",
        "name": "Number of CPUs",
        "key_": "system.cpu.num",
        "delay": "1m",
        "history": "7d",
        "trends": "365d",
        "status": "0",
        "value_type": "3",
        "trapper_hosts": "",
        "units": "",
```

```
            "logtimefmt": "",
            "templateid": "42220",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "1",
            "description": "",
            "inventory_link": "0",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "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",
            "uuid": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0",
            "name_resolved": "Number of CPUs"
        }
    ],
    "id": 1
}
```


Znajdowanie pozycji zależnych według klucza

Pobierz wszystkie pozycje zależne z hosta o ID ”10116”, które mają słowo ”apache” w kluczu.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10116",
        "search": {
            "key_": "apache"
        },
```

```
        "filter": {
            "type": 18
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "25550",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "name": "Days",
            "key_": "apache.status.uptime.days",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "25545",
            "timeout": "",
            "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",
            "uuid": "",
            "state": "0",
```

```
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0",
            "name_resolved": "Days"
        },
        {
            "itemid": "25555",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "name": "Hours",
            "key_": "apache.status.uptime.hours",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "25545",
            "timeout": "",
            "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",
            "uuid": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
```

```
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0",
            "name_resolved": "Hours"
        }
    ],
    "id": 1
}
```

Znajdź pozycja HTTP agent

Znajdź pozycja HTTP agent z typem treści POST XML dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "hostids": "10255",
        "filter": {
            "type": 19,
            "post_type": 3
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "28252",
            "type": "19",
            "snmp_oid": "",
            "hostid": "10255",
            "name": "template item",
            "key_": "ti",
            "delay": "30s",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
```

```
            "url": "localhost",
            "query_fields": [
                {
                    "name": "mode",
                    "value": "xml"
                }
            ],
            "posts": "<body>\r\n<![CDATA[{$MACRO}<foo></bar>]]>\r\n</body>",
            "status_codes": "200",
            "follow_redirects": "0",
            "post_type": "3",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "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",
            "uuid": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "",
            "prevvalue": "",
            "name_resolved": "template item"
        }
    ],
    "id": 1
}
```

Pobieranie pozycji z regułami przetwarzania wstępnego

Pobierz wszystkie pozycje i ich reguły przetwarzania wstępnego dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": ["itemid", "name", "key_"],
        "selectPreprocessing": "extend",
        "hostids": "10254"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemid": "23865",
        "name": "HTTP agent example JSON",
        "key_": "json",
        "preprocessing": [
            {
                "type": "12",
                "params": "$.random",
```

```
                "error_handler": "1",
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Zobacz także

- Reguła wykrywania
- Wykres
- Host
- Interfejs hosta
- Wyzwalacz

Źródło

CItem::get() w *ui/include/classes/api/services/CItem.php*.

item.update

Opis

`object item.update(object/array items)`

Ta metoda umożliwia aktualizację istniejących pozycji.

Note:
Pozycje web nie mogą być aktualizowane za pomocą Zabbix API.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości pozycji do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdej pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
preprocessing	array	Opcje wstępnego przetwarzania pozycji, które zastąpią bieżące opcje wstępnego przetwarzania. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych
tags	array	Tagi pozycji. Zachowanie parametru: - *tylko do odczytu* dla obiektów wykrytych

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych pozycji we właściwości `itemids`.

Przykłady

Włączanie pozycji

Włącz pozycję, czyli ustaw jej status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "10092",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizacja zależnej pozycji

Zaktualizuj nazwę zależnej pozycji oraz ID pozycji nadrzędnej. Dozwolone są tylko zależności w obrębie tego samego hosta, dlatego pozycja nadrzędna i zależna pozycja powinny mieć ten sam `hostid`.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "name": "Dependent item updated name",
        "master_itemid": "25562",
        "itemid": "189019"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189019"
        ]
    },
    "id": 1
}
```

Aktualizacja pozycji HTTP agent

Włącz przechwytywanie wartości pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "allow_traps": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Aktualizacja pozycji z przetwarzaniem wstępnym

Zaktualizuj pozycję za pomocą reguły przetwarzania wstępnego pozycji „W zakresie”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "preprocessing": [
            {
                "type": 13,
                "params": "\n100",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Aktualizowanie pozycji skryptowej

Zaktualizuj pozycję skryptową innym skryptem i usuń niepotrzebne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "params": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItem::update() w *ui/include/classes/api/services/CItem.php*.

Pozycja wykresu

Ta klasa jest przeznaczona do pracy z pozycjami wykresu.

Odwołania do obiektów:

- Pozycja wykresu

Dostępne metody:

- graphitem.get - pobieranie pozycji wykresu

Obiekt elementu wykresu

Następujące obiekty są bezpośrednio powiązane z API `graphitem`.

Element wykresu

Note:
Elementy wykresu można modyfikować wyłącznie przez API `graph`.

Obiekt elementu wykresu ma następujące właściwości.

Właściwość	Typ	Opis
gitemid	ID	ID elementu wykresu. Zachowanie właściwości: - *tylko do odczytu*
color	string	Kolor rysowania elementu wykresu jako szesnastkowy kod koloru. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
itemid	ID	ID pozycji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
calc_fnc	integer	Wartość pozycji, która będzie wyświetlana. Możliwe wartości: 1 - wartość minimalna; 2 - *(domyślnie)* wartość średnia; 4 - wartość maksymalna; 7 - wszystkie wartości; 9 - ostatnia wartość, używana tylko przez wykresy kołowe i rozbite.
drawtype	integer	Styl rysowania elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* linia; 1 - wypełniony obszar; 2 - pogrubiona linia; 3 - kropka; 4 - linia przerywana; 5 - linia gradientowa.

Właściwość	Typ	Opis
graphid	ID	ID wykresu, do którego należy element wykresu.
sortorder	integer	Pozycja pozycji na wykresie. Domyślnie: zaczyna się od „0” i zwiększa o jeden dla każdego wpisu.
type	integer	Typ elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* prosty; 2 - suma wykresu, używana tylko przez wykresy kołowe i rozbite.
yaxisside	integer	Strona wykresu, po której będzie rysowana skala Y elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* lewa strona; 1 - prawa strona.

graphitem.get

Opis

`integer/array graphitem.get(object parameters)`

Ta metoda umożliwia pobieranie elementów wykresu zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników każdego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwraca tylko pozycje wykresu należące do podanych wykresów.
itemids	ID/array	Zwraca tylko pozycje wykresu o podanych identyfikatorach pozycji.
type	integer	Zwraca tylko pozycje wykresu o podanym typie. Listę obsługiwanych typów pozycji wykresu można znaleźć na stronie obiektu pozycji wykresu.
selectGraphs	query	Zwraca właściwość `graphs` zawierającą tablicę wykresów, do których należy pozycja.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `gitemid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie pozycji wykresu z wykresu

Pobierz wszystkie pozycje wykresu używane na wykresie wraz z dodatkowymi informacjami o pozycji i hoście.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphitem.get",
    "params": {
        "output": "extend",
        "graphids": "387"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "gitemid": "1242",
            "graphid": "387",
            "itemid": "22665",
            "drawtype": "1",
            "sortorder": "1",
            "color": "FF5555",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        },
        {
            "gitemid": "1243",
            "graphid": "387",
            "itemid": "22668",
            "drawtype": "1",
            "sortorder": "2",
            "color": "55FF55",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        },
        {
            "gitemid": "1244",
            "graphid": "387",
            "itemid": "22671",
            "drawtype": "1",
            "sortorder": "3",
            "color": "009999",
            "yaxisside": "0",
            "calc_fnc": "2",
            "type": "0"
        }
    ],
    "id": 1
}
```

Zobacz także

- Wykres

Źródło

CGraphItem::get() w *ui/include/classes/api/services/CGraphItem.php*.

Problem

Ta klasa jest przeznaczona do pracy z problemami.

Odwołania do obiektów:

- Problem
  - URL typu nośnika
- Tag problemu

Dostępne metody:

- problem.get - pobieranie problemów

Obiekt problemu

Następujące obiekty są bezpośrednio powiązane z API `problem`.

Problem

Note:
Problemy są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt problemu ma następujące właściwości.

Właściwość	Typ	Opis
eventid	ID	ID zdarzenia problemu.
source	integer	Typ zdarzenia problemu. Możliwe wartości: 0 - zdarzenie utworzone przez wyzwalacz; 3 - zdarzenie wewnętrzne; 4 - zdarzenie utworzone podczas aktualizacji stanu usługi.
object	integer	Typ obiektu powiązanego ze zdarzeniem problemu. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”: 0 - wyzwalacz. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - wyzwalacz; 4 - pozycja; 5 - reguła LLD. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone podczas aktualizacji stanu usługi”: 6 - usługa.
objectid	ID	ID powiązanego obiektu.
clock	timestamp	Czas utworzenia zdarzenia problemu.
ns	integer	Nanosekundy w momencie utworzenia zdarzenia problemu.
r_eventid	ID	ID zdarzenia przywrócenia.
r_clock	timestamp	Czas utworzenia zdarzenia przywrócenia.
r_ns	integer	Nanosekundy w momencie utworzenia zdarzenia przywrócenia.
cause_eventid	ID	ID zdarzenia przyczyny.
correlationid	ID	ID reguły korelacji, jeśli to zdarzenie zostało przywrócone przez globalną regułę korelacji.
userid	ID	ID użytkownika, który zamknął problem (jeśli problem został zamknięty ręcznie).
name	string	Rozwiązana nazwa problemu.
acknowledged	integer	Stan potwierdzenia problemu. Możliwe wartości: 0 - niepotwierdzony; 1 - potwierdzony.

Właściwość	Typ	Opis
severity	integer	Bieżący poziom ważności problemu. Możliwe wartości: 0 - niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
suppressed	integer	Czy problem jest stłumiony. Możliwe wartości: 0 - problem jest w stanie normalnym; 1 - problem jest stłumiony.
opdata	string	Dane operacyjne z rozwiniętymi makrami.
urls	array	Aktywne adresy URL typu nośnika.

URL typu nośnika

Obiekt URL typu nośnika ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa URL zdefiniowanego typu nośnika.
url	string	Wartość URL zdefiniowanego typu nośnika.

Wyniki będą zawierać wpisy tylko dla aktywnych typów nośników z włączonym wpisem menu zdarzeń. Makra użyte we właściwościach zostaną rozwinięte, ale jeśli jedna z właściwości zawiera nierozwinięte makro, obie właściwości zostaną wykluczone z wyników. Obsługiwane makra opisano w sekcji *Obsługiwane makra*.

Znacznik problemu

Obiekt znacznika problemu ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa znacznika problemu.
value	string	Wartość znacznika problemu.

problem.get

Opis

`integer/array problem.get(object parameters)`

Ta metoda umożliwia pobieranie problemów zgodnie z podanymi parametrami.

Ta metoda służy do pobierania nierozwiązanych problemów. Możliwe jest również, jeśli to określono, dodatkowe pobieranie niedawno rozwiązanych problemów. Okres, który określa, jak stare oznacza „niedawno”, jest zdefiniowany w *Administracja → Ogólne*. Problemy, które zostały rozwiązane przed tym okresem, nie są przechowywane w tabeli problemów. Aby pobrać problemy, które zostały rozwiązane wcześniej w przeszłości, użyj metody `event.get`.

Attention:
Ta metoda może zwracać problemy usuniętej jednostki, jeśli problemy te nie zostały jeszcze usunięte przez housekeeper.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
eventids	ID/array	Zwraca tylko problemy o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko problemy utworzone przez obiekty należące do podanych grup hostów.
hostids	ID/array	Zwraca tylko problemy utworzone przez obiekty należące do podanych hostów.
objectids	ID/array	Zwraca tylko problemy utworzone przez podane obiekty.
source	integer	Zwraca tylko problemy o podanym typie. Listę obsługiwanych typów zdarzeń można znaleźć na stronie obiektu zdarzenia problemu. Domyślnie: 0 - problem utworzony przez wyzwalacz.
object	integer	Zwraca tylko problemy utworzone przez obiekty podanego typu. Listę obsługiwanych typów obiektów można znaleźć na stronie obiektu zdarzenia problemu. Domyślnie: 0 - wyzwalacz.
acknowledged	boolean	Jeśli ustawiono na `true`, zwraca tylko potwierdzone problemy.
action	integer	Zwraca tylko problemy, dla których wykonano podane akcje aktualizacji zdarzenia. W przypadku wielu akcji należy użyć sumy dowolnych akceptowalnych wartości mapy bitowej jako maski bitowej (na przykład 34 dla potwierdzenia i wyciszenia zdarzenia).
action_userids	ID/array	Zwraca tylko problemy z podanymi identyfikatorami użytkowników, którzy wykonali akcje aktualizacji zdarzenia problemu.
suppressed	boolean	Jeśli ustawiono na `true`, zwraca tylko wyciszone problemy.
symptom	boolean	Jeśli ustawiono na `true`, zwraca tylko zdarzenia problemów będące symptomami.
severities	integer/array	Zwraca tylko problemy o podanych poziomach ważności zdarzeń. Ma zastosowanie tylko wtedy, gdy object to trigger.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwraca tylko problemy z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie problemy. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
recent	boolean	Jeśli ustawiono na `true`, zwraca ostatnie problemy, w tym te rozwiązane w ciągu `ok_period`, oraz wszystkie nierozwiązane problemy. Jeśli ustawiono na `false`, zwraca tylko nierozwiązane problemy.
eventid_from	string	Zwraca tylko problemy o identyfikatorach większych lub równych podanemu identyfikatorowi.
eventid_till	string	Zwraca tylko problemy o identyfikatorach mniejszych lub równych podanemu identyfikatorowi.
time_from	timestamp	Zwraca tylko problemy utworzone po podanym czasie lub dokładnie o podanym czasie.
time_till	timestamp	Zwraca tylko problemy utworzone przed podanym czasem lub dokładnie o podanym czasie.

Parametr	Typ	Opis
selectAcknowledges	query	Zwraca właściwość `acknowledges` z aktualizacjami problemu. Aktualizacje problemu są sortowane w odwrotnej kolejności chronologicznej. Obiekt aktualizacji problemu ma następujące właściwości: `acknowledgeid` - `(ID)` identyfikator aktualizacji; `userid` - `(ID)` identyfikator użytkownika, który zaktualizował zdarzenie; `eventid` - `(ID)` identyfikator zaktualizowanego zdarzenia; `clock` - `(timestamp)` czas aktualizacji zdarzenia; `message` - `(string)` treść wiadomości; `action` - `(integer)` typ akcji aktualizacji (zobacz `event.acknowledge`); `old_severity` - `(integer)` ważność zdarzenia przed tą akcją aktualizacji; `new_severity` - `(integer)` ważność zdarzenia po tej akcji aktualizacji; `suppress_until` - `(timestamp)` czas, do którego zdarzenie będzie wyciszone; `taskid` - `(ID)` identyfikator zadania, jeśli bieżące zdarzenie przechodzi zmianę rangi. Obsługuje `count`.
selectTags	query	Zwraca właściwość `tags` z tagami problemu. Format wyjściowy: `[{"tag": "<tag>", "value": "<value>"}, ...]`.
selectSuppressionData	query	Zwraca właściwość `suppression_data` z listą aktywnych okresów utrzymaniowych i ręcznych wyciszeń: `maintenanceid` - `(ID)` identyfikator okresu utrzymaniowego; `userid` - `(ID)` identyfikator użytkownika, który wyciszył problem; `suppress_until` - `(integer)` czas, do którego problem jest wyciszony.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `eventid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie zdarzeń problemów wyzwalacza

Pobierz ostatnie zdarzenia dla wyzwalacza „15112”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "selectAcknowledges": "extend",
        "selectTags": "extend",
        "selectSuppressionData": "extend",
        "objectids": "15112",
        "recent": true,
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "Agent Zabbix na localhost jest niedostępny przez 5 minut",
            "acknowledged": "1",
            "severity": "3",
            "cause_eventid": "0",
            "opdata": "",
            "acknowledges": [
                {
                    "acknowledgeid": "14443",
                    "userid": "1",
                    "eventid": "1245463",
                    "clock": "1472457281",
                    "message": "problem rozwiązany",
                    "action": "6",
                    "old_severity": "0",
                    "new_severity": "0",
                    "suppress_until": "1472511600",
                    "taskid": "0"
                }
            ],
            "suppression_data": [
                {
                    "maintenanceid": "15",
                    "suppress_until": "1472511600",
                    "userid": "0"
                }
            ],
            "suppressed": "1",
            "tags": [
```

```
                {
                    "tag": "test-tag",
                    "value": "test-value"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie problemów potwierdzonych przez określonego użytkownika

Pobieranie problemów potwierdzonych przez użytkownika o ID=10

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "1248566",
            "source": "0",
            "object": "0",
            "objectid": "15142",
            "clock": "1472457242",
            "ns": "209442442",
            "r_eventid": "1245468",
            "r_clock": "1472457285",
            "r_ns": "125644870",
            "correlationid": "0",
            "userid": "10",
            "name": "Agent Zabbix na localhost jest niedostępny przez 5 minut",
            "acknowledged": "1",
            "severity": "3",
            "cause_eventid": "0",
            "opdata": "",
            "acknowledges": [
                {
                    "userid": "10",
                    "action": "2"
                }
            ],
            "suppressed": "0"
        }
    ],
    "id": 1
}
```

Zobacz także

- Alert
- Pozycja
- Host
- Reguła LLD
- Wyzwalacz

Źródło

CEvent::get() w *ui/include/classes/api/services/CProblem.php*.

Prototyp hosta

Ta klasa jest przeznaczona do pracy z prototypami hostów.

Odwołania do obiektów:

- Prototyp hosta
- Powiązanie grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs
  - Szczegóły niestandardowego interfejsu

Dostępne metody:

- hostprototype.create - tworzenie nowych prototypów hostów
- hostprototype.delete - usuwanie prototypów hostów
- hostprototype.get - pobieranie prototypów hostów
- hostprototype.update - aktualizacja prototypów hostów

Obiekt prototypu hosta

Następujące obiekty są bezpośrednio powiązane z API `hostprototype`.

Prototyp hosta

Obiekt prototypu hosta ma następujące właściwości.

Właściwość	Type	Opis
hostid	ID	ID prototypu hosta. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
host	string	Nazwa techniczna prototypu hosta. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Widoczna nazwa prototypu hosta. Domyślnie: wartość właściwości `host`. Zachowanie właściwości: - *tylko do odczytu* dla obiektów dziedziczonych
status	integer	Status prototypu hosta. Możliwe wartości: 0 - *(domyślnie)* monitorowany host; 1 - niemonitorowany host.

Właściwość	Type	Opis
flags	integer	Pochodzenie prototypu hosta. Możliwe wartości: 2 - prototyp hosta; 6 - wykryty prototyp hosta Zachowanie właściwości: - *tylko do odczytu*
inventory_mode	integer	Tryb uzupełniania inwentarza hosta. Możliwe wartości: -1 - *(domyślnie)* wyłączony; 0 - ręczny; 1 - automatyczny.
templateid	ID	ID nadrzędnego szablonu prototypu hosta. Zachowanie właściwości: - *tylko do odczytu*
discover	integer	Status wykrywania prototypu hosta. Możliwe wartości: 0 - *(domyślnie)* nowe hosty będą wykrywane; 1 - nowe hosty nie będą wykrywane, a istniejące hosty zostaną oznaczone jako utracone.
custom_interfaces	integer	Źródło niestandardowych interfejsów dla hostów utworzonych przez prototyp hosta. Możliwe wartości: 0 - *(domyślnie)* dziedziczenie interfejsów z nadrzędnego hosta; 1 - użycie niestandardowych interfejsów prototypów hostów. Zachowanie właściwości: - *tylko do odczytu* dla obiektów dziedziczonych
uuid	string	Uniwersalny unikatowy identyfikator używany do łączenia importowanych prototypów hostów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane* jeśli prototyp hosta należy do szablonu

Łącze grupy

Obiekt łącza grupy łączy prototyp hosta z grupą hostów. Ma następujące właściwości.

Właściwość	Type	Opis
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *wymagane*

Prototyp grupy

Obiekt prototypu grupy definiuje grupę, która zostanie utworzona dla wykrytego hosta. Ma następujące właściwości.

Właściwość	Type	Opis
group_prototypeid	ID	ID prototypu grupy. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Type	Opis
name	string	Nazwa prototypu grupy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Tag szablonu hosta

Obiekt tagu szablonu hosta ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu szablonu hosta. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu szablonu hosta.

Interfejs niestandardowy

Interfejsy niestandardowe są obsługiwane, jeśli `custom_interfaces` obiektu Host prototype object jest ustawione na „użyj niestandardowych interfejsów prototypów hostów”. Obiekt interfejsu niestandardowego ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ interfejsu. Możliwe wartości: 1 - Agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Zachowanie właściwości: - *wymagane*
useip	integer	Czy połączenie powinno być nawiązywane przez IP. Możliwe wartości: 0 - połącz przy użyciu nazwy DNS hosta; 1 - połącz przy użyciu adresu IP hosta. Zachowanie właściwości: - *wymagane*
ip	string	Adres IP używany przez interfejs. Może zawierać makra. Zachowanie właściwości: - *wymagane*, jeśli `useip` jest ustawione na „połącz przy użyciu adresu IP hosta”
dns	string	Nazwa DNS używana przez interfejs. Może zawierać makra. Zachowanie właściwości: - *wymagane*, jeśli `useip` jest ustawione na „połącz przy użyciu nazwy DNS hosta”
port	string	Numer portu używany przez interfejs. Może zawierać makra użytkownika i LLD. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
main	integer	Czy interfejs jest używany jako domyślny na hoście. Tylko jeden interfejs danego typu może być ustawiony jako domyślny na hoście. Możliwe wartości: 0 - nie jest domyślny; 1 - domyślny. Zachowanie właściwości: - *wymagane*
details	object	Dodatkowy obiekt dla szczegółów interfejsu niestandardowego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SNMP”

Szczegóły niestandardowego interfejsu

Obiekt szczegółów ma następujące właściwości.

Właściwość	Typ	Opis
version	integer	Wersja interfejsu SNMP. Możliwe wartości: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Zachowanie właściwości: - *wymagane*
bulk	integer	Czy używać zbiorczych żądań SNMP. Możliwe wartości: 0 - nie używaj zbiorczych żądań; 1 - *(domyślnie)* - używaj zbiorczych żądań.
community	string	Społeczność SNMP. Zachowanie właściwości: - *wymagane* jeśli `version` jest ustawione na ”SNMPv1” lub ”SNMPv2c”
max_repetitions	integer	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Używane tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Domyślnie: 10.
securityname	string	Nazwa bezpieczeństwa SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”
securitylevel	integer	Poziom bezpieczeństwa SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”

Właściwość	Typ	Opis
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
privpassphrase	string	Hasło prywatności SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authPriv”
authprotocol	integer	Protokół uwierzytelniania SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
privprotocol	integer	Protokół prywatności SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authPriv”
contextname	string	Nazwa kontekstu SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”

hostprototype.create

Opis

`object hostprototype.create(object/array hostPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy hostów do utworzenia.

Oprócz standardowych właściwości prototypu hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groupLinks	array	Powiązania grup, które mają zostać utworzone dla prototypu hosta. Zachowanie parametru: - *wymagany*
ruleid	ID	ID reguły LLD, do której należy prototyp hosta. Zachowanie parametru: - *wymagany*
groupPrototypes	array	Prototypy grup, które mają zostać utworzone dla prototypu hosta.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla prototypu hosta.
tags	object/array	Tagi prototypu hosta.
interfaces	object/array	Niestandardowe interfejsy prototypu hosta.
templates	object/array	Szablony, które mają zostać podlinkowane do prototypu hosta. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów hostów we właściwości `hostids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów hostów.

Przykłady

Tworzenie prototypu hosta

Utwórz prototyp hosta "{#VM.NAME}" w regule LLD "23542" z prototypem grupy "{#HV.NAME}", parą tagów "datacenter": "{#DATACENTER.NAME}" oraz niestandardowym interfejsem SNMPv2 127.0.0.1:161 ze społecznością {$SNMP_COMMUNITY}. Połącz go z grupą hostów "2".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.create",
    "params": {
        "host": "{#VM.NAME}",
        "ruleid": "23542",
        "custom_interfaces": "1",
        "groupLinks": [
            {
                "groupid": "2"
            }
        ],
        "groupPrototypes": [
            {
                "name": "{#HV.NAME}"
            }
        ],
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            }
        ],
        "interfaces": [
            {
                "main": "1",
                "type": "2",
                "useip": "1",
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
                "details": {
```

```
                    "version": "2",
                    "bulk": "1",
                    "community": "{$SNMP_COMMUNITY}"
                }
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103"
        ]
    },
    "id": 1
}
```

Zobacz także

- Powiązanie grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs
- Makro użytkownika

Źródło

CHostPrototype::create() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.delete

Opis

`object hostprototype.delete(array hostPrototypeIds)`

Ta metoda umożliwia usuwanie prototypów hosta.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Role użytkownika.

Parametry

`(array)` ID prototypów hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów hostów we właściwości `hostids`.

Przykłady

Usuwanie wielu prototypów hosta

Usuń dwa prototypy hosta.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostprototype.delete",
"params": [
"10103",
"10105"
],
```

```
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostids": [
"10103",
"10105"
]
},
"id": 1
}
```

Źródło

CHostPrototype::delete() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.get

Opis

`integer/array hostprototype.get(object parameters)`

Metoda umożliwia pobieranie prototypów hostów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
hostids	ID/array	Zwraca tylko prototypy hostów o podanych ID.
discoveryids	ID/array	Zwraca tylko prototypy hostów należące do podanych reguł LLD.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko pozycje odziedziczone z szablonu.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania prototypu hosta. Obiekt wykrywania prototypu hosta łączy wykryty prototyp hosta z prototypem hosta, z którego został wykryty. Ma następujące właściwości: `host` - `(string)` ID hosta; `parent_hostid` - `(string)` ID prototypu hosta, z którego został utworzony prototyp hosta; `status` - `(int)` status wykrywania prototypu hosta: 0 - *(domyślnie)* prototyp hosta został wykryty, 1 - prototyp hosta nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, w którym prototyp hosta, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, w którym prototyp hosta, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik tego, czy prototyp hosta został wyłączony przez regułę LLD, czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwraca właściwość `discoveryRule` z regułą LLD, do której należy prototyp hosta.

Parameter	Type	Description
selectDiscoveryRulePrototype	query	Zwraca właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do której należy prototyp hosta.
selectInterfaces	query	Zwraca właściwość `interfaces` z niestandardowymi interfejsami prototypu hosta.
selectGroupLinks	query	Zwraca właściwość `groupLinks` z powiązaniami grup prototypu hosta.
selectGroupPrototypes	query	Zwraca właściwość `groupPrototypes` z prototypami grup prototypu hosta.
selectMacros	query	Zwraca właściwość `macros` z makrami prototypu hosta.
selectParentHost	query	Zwraca właściwość `parentHost` z hostem, do którego należy prototyp hosta.
selectTags	query	Zwraca właściwość `tags` z tagami prototypu hosta.
selectTemplates	query	Zwraca właściwość `templates` z szablonami powiązanymi z prototypem hosta. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`, `discovered`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Wartości zwracane

`(integer/array)` Zwraca albo:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie prototypów hostów z reguły LLD

Pobierz wszystkie prototypy hostów, ich powiązania z grupami, prototypy grup oraz tagi z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.get",
    "params": {
        "output": "extend",
        "selectInterfaces": "extend",
        "selectGroupLinks": "extend",
        "selectGroupPrototypes": "extend",
        "selectTags": "extend",
        "discoveryids": "23554"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "hostid": "10092",
            "host": "{#HV.UUID}",
            "name": "{#HV.UUID}",
            "status": "0",
            "templateid": "0",
            "flags": "2",
            "discover": "0",
            "custom_interfaces": "1",
            "inventory_mode": "-1",
            "groupLinks": [
                {
                    "group_prototypeid": "4",
                    "hostid": "10092",
                    "groupid": "7",
                    "templateid": "0"
                }
            ],
            "groupPrototypes": [
                {
                    "group_prototypeid": "7",
                    "hostid": "10092",
                    "name": "{#CLUSTER.NAME}",
                    "templateid": "0"
                }
            ],
            "tags": [
                {
                    "tag": "datacenter",
                    "value": "{#DATACENTER.NAME}"
                },
                {
                    "tag": "instance-type",
                    "value": "{#INSTANCE_TYPE}"
                }
            ],
            "interfaces": [
                {
                    "main": "1",
                    "type": "2",
                    "useip": "1",
                    "ip": "127.0.0.1",
                    "dns": "",
                    "port": "161",
                    "details": {
                        "version": "2",
                        "bulk": "1",
                        "community": "{$SNMP_COMMUNITY}",
                        "max_repetitions": "10"
                    }
                }
            ]
        }
    ],
    "id": 1
}
```


Zobacz także

- Link grupy
- Prototyp grupy
- Makro użytkownika

Source

CHostPrototype::get() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.update

Opis

`object hostprototype.update(object/array hostPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu hosta do zaktualizowania.

Właściwość `hostid` musi być zdefiniowana dla każdego prototypu hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groupLinks	array	Dowiązania grup zastępujące bieżące dowiązania grup w prototypie hosta. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
groupPrototypes	array	Prototypy grup zastępujące istniejące prototypy grup w prototypie hosta. Wszystkie prototypy grup, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika. Wszystkie makra, które nie zostały wymienione w żądaniu, zostaną usunięte.
tags	object/array	Tagi prototypu hosta zastępujące bieżące tagi. Wszystkie tagi, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
interfaces	object/array	Niestandardowe interfejsy prototypu hosta zastępujące bieżące interfejsy. Obiekt niestandardowego interfejsu powinien zawierać wszystkie swoje parametry. Wszystkie interfejsy, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *obsługiwane*, jeśli `custom_interfaces` obiektu Host prototype object jest ustawione na "use host prototypes custom interfaces" - *tylko do odczytu* dla obiektów dziedziczonych
templates	object/array	Szablony zastępujące aktualnie podlinkowane szablony. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów hostów we właściwości `hostids`.

Przykłady

Wyłączanie prototypu hosta

Wyłącz prototyp hosta, czyli ustaw jego status na „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "status": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów prototypu hosta

Zastąp tagi prototypu hosta nowymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            },
            {
                "tag": "instance-type",
                "value": "{#INSTANCE_TYPE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizowanie niestandardowych interfejsów prototypu hosta

Zastąp dziedziczone interfejsy niestandardowymi interfejsami prototypu hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "custom_interfaces": "1",
        "interfaces": [
            {
                "main": "1",
                "type": "2",
                "useip": "1",
                "ip": "127.0.0.1",
                "dns": "",
                "port": "161",
                "details": {
                    "version": "2",
                    "bulk": "1",
                    "community": "{$SNMP_COMMUNITY}"
                }
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Zobacz także

- Łącze grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs
- Makro użytkownika

Źródło

CHostPrototype::update() w *ui/include/classes/api/services/CHostPrototype.php*.

Prototyp pozycji

Ta klasa jest przeznaczona do pracy z prototypami pozycji.

Odwołania do obiektów:

- Prototyp pozycji
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Tag prototypu pozycji
- Przetwarzanie wstępne prototypu pozycji

Dostępne metody:

- itemprototype.create - utwórz nowe prototypy pozycji

- itemprototype.delete - usuń prototypy pozycji
- itemprototype.get - pobierz prototypy pozycji
- itemprototype.update - zaktualizuj prototypy pozycji

Obiekt prototypu pozycji

Następujące obiekty są bezpośrednio powiązane z API `itemprototype`.

Prototyp pozycji

Obiekt prototypu pozycji ma następujące właściwości.

Property	Type	Description
itemid	ID	ID prototypu pozycji. Property behavior: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
delay	string	Interwał aktualizacji prototypu pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej niestandardowych interwałów, wszystkie oddzielone średnikami. Niestandardowe interwały mogą być mieszanką interwałów elastycznych i harmonogramowanych. Akceptuje makra użytkownika i makra LLD. Jeśli są używane, wartość musi być pojedynczym makrem. Nie są obsługiwane wielokrotne makra ani makra połączone z tekstem. Interwały elastyczne można zapisać jako dwa makra oddzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Property behavior: - *wymagane* jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”Calculated” (15), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), lub jeśli `type` jest ustawione na ”Zabbix agent (active)” (7) i `key_` nie zawiera ”mqtt.get”
hostid	ID	ID hosta, do którego należy prototyp pozycji. Property behavior: - *stałe* - *wymagane* dla operacji tworzenia
interfaceid	ID	ID interfejsu hosta prototypu pozycji. Property behavior: - *wymagane* jeśli prototyp pozycji należy do hosta i `type` jest ustawione na ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap” lub ”SNMP agent” - *obsługiwane* jeśli prototyp pozycji należy do hosta i `type` jest ustawione na ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent”
key_	string	Klucz prototypu pozycji. Property behavior: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
name	string	Nazwa prototypu pozycji. Obsługuje makra użytkownika. Property behavior: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
type	integer	Typ prototypu pozycji. Możliwe wartości: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 15 - Calculated; 16 - JMX agent; 17 - SNMP trap; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser. Property behavior: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
url	string	Łańcuch URL. Obsługuje makra LLD, makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
value_type	integer	Typ informacji prototypu pozycji. Możliwe wartości: 0 - numeric float; 1 - character; 2 - log; 3 - numeric unsigned; 4 - text; 5 - binary. Property behavior: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Zezwala na wypełnianie wartości podobnie jak w pozycji trapper. 0 - *(domyślnie)* Nie zezwalaj na przyjmowanie danych przychodzących; 1 - Zezwalaj na przyjmowanie danych przychodzących. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent”

Property	Type	Description
authtype	integer	Metoda uwierzytelniania. Możliwe wartości jeśli `type` jest ustawione na ”SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości jeśli `type` jest ustawione na ”HTTP agent”: 0 - *(domyślnie)* brak; 1 - basic; 2 - NTLM; 3 - Kerberos; 4 - Digest. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”SSH agent” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
description	string	Opis prototypu pozycji.
flags	integer	Origin prototypu pozycji. Możliwe wartości: 2 - prototyp pozycji; 6 - wykryty prototyp pozycji Property behavior: - *tylko do odczytu*
follow_redirects	integer	Podążaj za przekierowaniami odpowiedzi podczas pobierania danych. Możliwe wartości: 0 - Nie podążaj za przekierowaniami; 1 - *(domyślnie)* Podążaj za przekierowaniami. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
history	string	Jednostka czasu określająca, jak długo mają być przechowywane dane historyczne. Akceptuje także makro użytkownika i makro LLD. Domyślnie: 31d.
http_proxy	string	Łańcuch połączenia proxy HTTP(S). Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ipmi_sensor	string	Czujnik IPMI. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”IPMI agent” i `key_` nie jest ustawione na ”ipmi.get” - *obsługiwane* jeśli `type` jest ustawione na ”IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
jmx_endpoint	string	Niestandardowy łańcuch połączenia agenta JMX. Domyślnie: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”JMX agent”
logtimefmt	string	Format czasu w wpisach dziennika. Property behavior: - *obsługiwane* jeśli `value_type` jest ustawione na ”log” - *tylko do odczytu* dla obiektów dziedziczonych
master_itemid	ID	ID pozycji nadrzędnej. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź ma zostać przekonwertowana do JSON. Możliwe wartości: 0 - *(domyślnie)* Zapisz surowe dane; 1 - Konwertuj do JSON. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
params	string	Dodatkowe parametry zależne od typu prototypu pozycji: - wykonywany skrypt dla prototypów pozycji SSH agent i TELNET agent; - zapytanie SQL dla prototypów pozycji database monitor; - formuła dla prototypów pozycji calculated; - skrypt dla prototypów pozycji script i browser. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”Script” lub ”Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` jest ustawione na ”Script” lub ”Browser”. Tablica obiektów z właściwościami `name` i `value`, gdzie `name` musi być unikalne. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”JMX agent” i `username` jest ustawione - *obsługiwane* jeśli `type` jest ustawione na ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)

Property	Type	Description
post_type	integer	Typ treści danych POST przechowywanych we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* Surowe dane; 2 - Dane JSON; 3 - Dane XML. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Property behavior: - *wymagane* jeśli `type` jest ustawione na ”HTTP agent” i `post_type` jest ustawione na ”Dane JSON” lub ”Dane XML” - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” i `post_type` jest ustawione na ”Surowe dane” - *tylko do odczytu* dla obiektów dziedziczonych
privatekey	string	Nazwa pliku klucza prywatnego. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”SSH agent” i `authtype` jest ustawione na ”klucz publiczny”
publickey	string	Nazwa pliku klucza publicznego. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”SSH agent” i `authtype` jest ustawione na ”klucz publiczny”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Która część odpowiedzi ma zostać zapisana. Możliwe wartości jeśli `request_method` jest ustawione na ”GET”, ”POST” lub ”PUT”: 0 - *(domyślnie)* Treść; 1 - Nagłówki; 2 - Zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości jeśli `request_method` jest ustawione na ”HEAD”: 1 - Nagłówki. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
snmp_oid	string	SNMP OID. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
status	integer	Status prototypu pozycji. Możliwe wartości: 0 - *(domyślnie)* włączony prototyp pozycji; 1 - wyłączony prototyp pozycji; 3 - nieobsługiwany prototyp pozycji.
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługuje także makra użytkownika lub makra LLD jako część listy rozdzielonej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	ID nadrzędnego prototypu pozycji szablonu. Property behavior: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m). Akceptuje także makra użytkownika i makra LLD. Zakres możliwych wartości: 1-600s. Domyślnie: ”” - użyj ustawień proxy/globalnych. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3) i `key_` nie zaczyna się od ”`vmware.`” ani ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) i `snmp_oid` zaczyna się od ”`walk[`” lub ”`get[`”, ”Script” (21), ”Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych
trapper_hosts	string	Dozwolone hosty. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”Zabbix trapper”, lub jeśli `type` jest ustawione na ”HTTP agent” i `allow_traps` jest ustawione na ”Zezwalaj na przyjmowanie danych przychodzących”

Property	Type	Description
trends	string	Jednostka czasu określająca, jak długo mają być przechowywane dane trendów. Akceptuje także makro użytkownika i makro LLD. Domyślnie: 365d. Property behavior: - *obsługiwane* jeśli `value_type` jest ustawione na ”numeric float” lub ”numeric unsigned”
units	string	Jednostki wartości. Property behavior: - *obsługiwane* jeśli `value_type` jest ustawione na ”numeric float” lub ”numeric unsigned” - *tylko do odczytu* dla obiektów dziedziczonych
username	string	Nazwa użytkownika do uwierzytelniania. Property behavior: - *wymagane* jeśli `type` jest ustawione na ”SSH agent” lub ”TELNET agent”, albo jeśli `type` jest ustawione na ”JMX agent” i `password` jest ustawione - *obsługiwane* jeśli `type` jest ustawione na ”Simple check”, ”Database monitor” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
uuid	string	Uniwersalny unikalny identyfikator, używany do powiązania importowanych prototypów pozycji z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany. Property behavior: - *obsługiwane* jeśli prototyp pozycji należy do szablonu
valuemapid	ID	ID powiązanej mapy wartości. Property behavior: - *obsługiwane* jeśli `value_type` jest ustawione na ”numeric float”, ”character” lub ”numeric unsigned” - *tylko do odczytu* dla obiektów dziedziczonych
verify_host	integer	Czy należy sprawdzać, czy nazwa hosta dla połączenia zgadza się z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
verify_peer	integer	Czy należy sprawdzać, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Property behavior: - *obsługiwane* jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
discover	integer	Status wykrywania prototypu pozycji. Możliwe wartości: 0 - *(domyślnie)* nowe pozycje będą wykrywane; 1 - nowe pozycje nie będą wykrywane, a istniejące pozycje zostaną oznaczone jako utracone.

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Tag szablonu pozycji

Obiekt tagu szablonu pozycji ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu szablonu pozycji. Property behavior: - *required*
value	string	Wartość tagu szablonu pozycji.

Preprocessing prototypu pozycji

Obiekt preprocessing prototypu pozycji ma następujące właściwości.

Property	Type	Description
type	integer	Typ opcji preprocessing. Możliwe wartości: 1 - Niestandardowy mnożnik; 2 - Przycinanie z prawej; 3 - Przycinanie z lewej; 4 - Przycinanie; 5 - Wyrażenie regularne; 6 - Wartość logiczna na dziesiętną; 7 - Ósemkowa na dziesiętną; 8 - Szesnastkowa na dziesiętną; 9 - Prosta zmiana; 10 - Zmiana na sekundę; 11 - XML XPath; 12 - JSONPath; 13 - W zakresie; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 18 - Sprawdzenie błędu za pomocą wyrażenia regularnego; 19 - Odrzuć niezmienione; 20 - Odrzuć niezmienione z heartbeat; 21 - JavaScript; 22 - Wzorzec Prometheus; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąp; 26 - Sprawdź jako nieobsługiwane; 27 - XML do JSON; 28 - Wartość z przebiegu SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Property behavior: - *required*
params	string	Dodatkowe parametry używane przez opcję preprocessing. Wiele parametrów jest oddzielonych znakiem nowej linii (\n). Jeśli `type` ma wartość ”Check unsupported”, parametry mają składnię `<scope>[\n<pattern>]`, gdzie *pattern* jest wyrażeniem regularnym, a *scope* jest jedną z wartości: -1 - dopasuj dowolny błąd; 0 - sprawdź, czy komunikat błędu pasuje do *pattern*; 1 - sprawdź, czy komunikat błędu nie pasuje do *pattern*. Property behavior: - *required* jeśli `type` ma wartość ”Custom multiplier” (1), ”Right trim” (2), ”Left trim” (3), ”Trim” (4), ”Regular expression” (5), ”XML XPath” (11), ”JSONPath” (12), ”In range” (13), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Check for error using regular expression” (18), ”Discard unchanged with heartbeat” (20), ”JavaScript” (21), ”Prometheus pattern” (22), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Replace” (25), ”Check unsupported” (26), ”SNMP walk value” (28), ”SNMP walk to JSON” (29) lub ”SNMP get value” (30)

Property	Type	Description
error_handler	integer	Typ akcji używany w przypadku niepowodzenia kroku preprocessing. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzuć wartość; 2 - Ustaw wartość niestandardową; 3 - Ustaw niestandardowy komunikat błędu. Możliwe wartości, jeśli `type` ma wartość ”Check unsupported”: 1 - Odrzuć wartość; 2 - Ustaw wartość niestandardową; 3 - Ustaw niestandardowy komunikat błędu. Property behavior: - *required* jeśli `type` ma wartość ”Custom multiplier” (1), ”Regular expression” (5), ”Boolean to decimal” (6), ”Octal to decimal” (7), ”Hexadecimal to decimal” (8), ”Simple change” (9), ”Change per second” (10), ”XML XPath” (11), ”JSONPath” (12), ”In range” (13), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Check for error using regular expression” (18), ”Prometheus pattern” (22), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Check unsupported” (26), ”XML to JSON” (27), ”SNMP walk value” (28), ”SNMP walk to JSON” (29) lub ”SNMP get value” (30)
error_handler_params	string	Parametry obsługi błędów. Property behavior: - *required* jeśli `error_handler` ma wartość ”Set custom value” lub ”Set custom error message”

Następujące parametry i obsługi błędów są obsługiwane dla każdego typu preprocessing.

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
1	Custom multiplier	number^1, 6^			0, 1, 2, 3
2	Right trim	list of characters[2]			
3	Left trim	list of characters[2]			
4	Trim	list of characters[2]			
5	Regular expression	pattern[3]	output[2]		0, 1, 2, 3
6	Boolean to decimal				0, 1, 2, 3
7	Octal to decimal				0, 1, 2, 3
8	Hexadecimal to decimal				0, 1, 2, 3

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
9	Simple change				0, 1, 2, 3
10	Change per second				0, 1, 2, 3
11	XML XPath	path[4]			0, 1, 2, 3
12	JSONPath	path[4]			0, 1, 2, 3
13	In range	min^1, 6^	max^1, 6^		0, 1, 2, 3
14	Matches regular expression	pattern[3]			0, 1, 2, 3
15	Does not match regular expression	pattern[3]			0, 1, 2, 3
16	Check for error in JSON	path[4]			0, 1, 2, 3
17	Check for error in XML	path[4]			0, 1, 2, 3
18	Check for error using regular expression	pattern[3]	output[2]		0, 1, 2, 3
19	Discard unchanged				
20	Discard unchanged with heartbeat	seconds^5, 6^			
21	JavaScript	script[2]			
22	Prometheus pattern	pattern^6, 7^	`value`, `label`, `function`	output^8, 9^	0, 1, 2, 3
23	Prometheus to JSON	pattern^6, 7^			0, 1, 2, 3

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
24	CSV to JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Replace	search string[2]	replacement[2]		
26	Check un-sup-ported	scope[1]	pattern^3, 6^		1, 2, 3
27	XML to JSON				0, 1, 2, 3
28	SNMP walk value	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS		0, 1, 2, 3
29	SNMP walk to JSON[10]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS	0, 1, 2, 3
30	SNMP get value	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Integer z BITS			0, 1, 2, 3

[1] liczba całkowita lub zmiennoprzecinkowa
 [2] ciąg znaków
 [3] wyrażenie regularne
 [4] JSONPath lub XML XPath
 [5] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [6] makro użytkownika, makro LLD
 [7] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy element wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika lub makrem LLD.
 [8] wynik Prometheus zgodny ze składnią: `<label name>` (może być makrem użytkownika lub makrem LLD), jeśli `label` jest wybrane jako drugi parametr.
 [9] jedna z funkcji agregujących: `sum`, `min`, `max`, `avg`, `count`, jeśli `function` jest wybrane jako drugi parametr.
 [10] obsługuje wiele rekordów ”Field name,OID prefix,Format records” rozdzielonych znakiem nowej linii.

itemprototype.create

Opis

`object itemprototype.create(object/array itemPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototyp pozycji do utworzenia.

Oprócz standardowych właściwości prototypu pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
ruleid	ID	ID reguły LLD, do której należy pozycja. Zachowanie parametru: - *wymagany*
preprocessing	array	Opcje przetwarzania wstępnego prototypu pozycji.
tags	array	Tagi prototypu pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów pozycji we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów pozycji.

Przykłady

Tworzenie prototypu pozycji

Utwórz prototyp pozycji, aby monitorować wolne miejsce na dysku w wykrytym systemie plików. Wykryte pozycje powinny być numerycznymi pozycjami agenta Zabbix, aktualizowanymi co 30 sekund.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Free disk space on {#FSNAME}",
        "key_": "vfs.fs.size[{#FSNAME},free]",
        "hostid": "10197",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27666"
        ]
    },
    "id": 1
}
```

Tworzenie prototypu pozycji z przetwarzaniem wstępnym

Utwórz pozycję, używając zmiany na sekundę oraz niestandardowego mnożnika jako drugiego kroku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Incoming network traffic on {#IFNAME}",
        "key_": "net.if.in[{#IFNAME}]",
        "hostid": "10001",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "delay": "60s",
```

```
        "units": "bps",
        "interfaceid": "1155",
        "preprocessing": [
            {
                "type": 10,
                "params": "",
                "error_handler": 0,
                "error_handler_params": ""
            },
            {
                "type": 1,
                "params": "8",
                "error_handler": 2,
                "error_handler_params": "10"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Tworzenie prototypu zależnej pozycji

Utwórz prototyp zależnej pozycji dla prototypu pozycji nadrzędnej o ID 44211. Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu/reguły wykrywania), dlatego pozycja nadrzędna i zależna pozycja powinny mieć takie samo hostid i ruleid.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
      "hostid": "10001",
      "ruleid": "27665",
      "name": "Dependent test item prototype",
      "key_": "dependent.prototype",
      "type": 18,
      "master_itemid": "44211",
      "value_type": 3
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44212"
        ]
    },
    "id": 1
}
```

Utwórz prototyp pozycji HTTP agent

Utwórz prototyp pozycji z adresem URL używającym makra użytkownika, pól zapytania i niestandardowych nagłówków.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "type": "19",
        "hostid": "10254",
        "ruleid": "28256",
        "interfaceid": "2",
        "name": "api item prototype example",
        "key_": "api_http_item",
        "value_type": 3,
        "url": "{$URL_PROTOTYPE}",
        "query_fields": [
            {
                "name": "min",
                "value": "10"
            },
            {
                "name": "max",
                "value" "100"
            }
        ],
        "headers": [
            {
                "name": "X-Source",
                "value": "api"
            }
        ],
        "delay": "35"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Utwórz prototyp pozycji skryptowej

Utwórz proste zbieranie danych przy użyciu prototypu pozycji skryptowej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Script example",
        "key_": "custom.script.itemprototype",
        "hostid": "12345",
        "type": 21,
        "value_type": 4,
        "params": "var request = new HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
```

```
        "parameters": [
            {
                "name": "host",
                "value": "{HOST.CONN}"
            }
        ],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::create() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.delete

Opis

`object itemprototype.delete(array itemPrototypeIds)`

Ta metoda pozwala na usuwanie prototypów pozycji.

Note:
Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Role użytkownika.

Parametry

`(array)` ID prototypów pozycji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów pozycji we właściwości `prototypeids`.

Przykłady

Usuwanie wielu prototypów pozycji

Usuń dwa prototypy pozycji.
 Zależne prototypy pozycji są usuwane automatycznie, jeśli pozycja nadrzędna lub prototyp pozycji zostanie usunięty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.delete",
    "params": [
        "27352",
        "27356"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "prototypeids": [
            "27352",
            "27356"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::delete() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.get

Opis

`integer/array itemprototype.get(object parameters)`

Ta metoda umożliwia pobieranie prototypów pozycji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji User roles.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
discoveryids	ID/array	Zwraca tylko prototypy pozycji należące do podanych reguł LLD.
graphids	ID/array	Zwraca tylko prototypy pozycji używane w podanych prototypach wykresów.
hostids	ID/array	Zwraca tylko prototypy pozycji należące do podanych hostów.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko prototypy pozycji odziedziczone z szablonu.
itemids	ID/array	Zwraca tylko prototypy pozycji o podanych ID.
monitored	boolean	Jeśli ustawione na `true`, zwraca tylko włączone prototypy pozycji należące do monitorowanych hostów.
templated	boolean	Jeśli ustawione na `true`, zwraca tylko prototypy pozycji należące do szablonów.
templateids	ID/array	Zwraca tylko prototypy pozycji należące do podanych szablonów.
triggerids	ID/array	Zwraca tylko prototypy pozycji używane w podanych prototypach wyzwalaczy.

Parameter	Type	Description
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania prototypu pozycji. Obiekt wykrywania prototypu pozycji łączy wykryty prototyp pozycji z prototypem pozycji, z którego został wykryty. Ma następujące właściwości: `parent_itemid` - `(string)` ID prototypu pozycji, z którego został utworzony prototyp pozycji; `key_` - `(string)` klucz prototypu pozycji; `status` - `(int)` status wykrywania prototypu pozycji: 0 - *(domyślnie)* prototyp pozycji został wykryty, 1 - prototyp pozycji nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, w którym prototyp pozycji, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, w którym prototyp pozycji, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik tego, czy prototyp pozycji został wyłączony przez regułę LLD, czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwraca właściwość `discoveryRule` z regułą niskopoziomowego wykrywania, do której należy prototyp pozycji.
selectDiscoveryRulePrototype	query	Zwraca właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp pozycji.
selectGraphs	query	Zwraca właściwość `graphs` z prototypami wykresów, w których używany jest prototyp pozycji. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy prototyp pozycji.
selectTags	query	Zwraca tagi prototypu pozycji w właściwości `tags`.
selectTriggers	query	Zwraca właściwość `triggers` z prototypami wyzwalaczy, w których używany jest prototyp pozycji. Obsługuje `count`.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami przetwarzania wstępnego prototypu pozycji.
selectValueMap	query	Zwraca właściwość `valuemap` z mapą wartości prototypu pozycji.
filter	object	Zwraca tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Nie obsługuje właściwości typu `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy prototyp pozycji.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectGraphs` - wyniki będą sortowane według `name`; `selectTriggers` - wyniki będą sortowane według `description`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`, `history`, `trends`, `discovered`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	

Parameter	Type	Description
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów pozycji z reguły LLD

Pobierz wszystkie prototypy pozycji dla określonego identyfikatora reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "",
            "flags": "2",
            "interfaceid": "0",
            "description": "",
            "evaltype": "0",
```

```
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "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": "",
            "flags": "2",
            "interfaceid": "0",
            "description": "Obciążenie procesora jest obliczane jako obciążenie CPU systemu podzielone prz
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "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": []
        }
    ],
    "id": 1
}
```

Znajdowanie zależnej pozycji

Znajdź jedną zależną pozycję dla określonego ID pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "filter": {
            "type": 18,
            "master_itemid": "25545"
        },
        "limit": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "25547",
            "type": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "name": "Seconds",
            "key_": "apache.status.uptime.seconds",
            "delay": "0",
            "history": "90d",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
```

```
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "evaltype": "0",
            "master_itemid": "25545",
            "jmx_endpoint": "",
            "timeout": "",
            "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": []
        }
    ],
    "id": 1
}
```

Znajdź prototyp pozycji HTTP agent

Znajdź prototyp pozycji HTTP agent z metodą żądania HEAD dla określonego hosta ID.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "hostids": "10254",
        "filter": {
            "type": 19,
            "request_method": 3
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "28257",
            "type": "19",
            "snmp_oid": "",
            "hostid": "10254",
            "name": "discovered",
            "key_": "item[{#INAME}]",
```

```
            "delay": "{#IUPDATE}",
            "history": "90d",
            "trends": "30d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "28255",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "2",
            "interfaceid": "2",
            "description": "",
            "evaltype": "0",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "{#IURL}",
            "query_fields": [],
            "posts": "",
            "status_codes": "",
            "follow_redirects": "0",
            "post_type": "0",
            "http_proxy": "",
            "headers": [],
            "retrieve_mode": "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",
            "discover": "0",
            "uuid": "",
            "parameters": []
        }
    ],
    "id": 1
}
```


Zobacz także

- Host
- Prototyp wykresu
- Prototyp wyzwalacza

Źródło

CItemPrototype::get() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.update

Opis

`object itemprototype.update(object/array itemPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu pozycji do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdego prototypu pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu pozycji, metoda akceptuje następujące parametry.

Parametr	Type	Opis
preprocessing	array	Opcje wstępnego przetwarzania prototypu pozycji, które zastąpią bieżące opcje wstępnego przetwarzania. Zachowanie parametrów: - *tylko do odczytu* dla obiektów dziedziczonych
tags	array	Tagi prototypu pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów pozycji we właściwości `itemids`.

Przykłady

Zmiana interfejsu prototypu pozycji

Zmień interfejs hosta, który będzie używany przez wykryte pozycje.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "27428",
        "interfaceid": "132"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27428"
        ]
    },
    "id": 1
}
```

Aktualizacja prototypu zależnej pozycji

Zaktualizuj prototyp zależnej pozycji, podając nowe ID prototypu pozycji nadrzędnej. Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu/reguły wykrywania), dlatego pozycja nadrzędna i zależna pozycja powinny mieć takie samo `hostid` i `ruleid`.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
```

```
    "params": {
        "master_itemid": "25570",
        "itemid": "189030"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189030"
        ]
    },
    "id": 1
}
```

Aktualizacja prototypu pozycji HTTP agent

Zmień pola zapytania i usuń wszystkie niestandardowe nagłówki.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid":"28305",
        "query_fields": [
            {
                "name": "random",
                "value": "qwertyuiopasdfghjklzxcvbnm"
            }
        ],
        "headers": []
    }
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Aktualizacja opcji przetwarzania wstępnego pozycji

Zaktualizuj prototyp pozycji za pomocą reguły przetwarzania wstępnego pozycji „Mnożnik niestandardowy”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 1,
                "params": "4",
```

```
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Aktualizowanie prototypu pozycji skryptowej

Zaktualizuj prototyp pozycji skryptowej, używając innego skryptu, i usuń zbędne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::update() w *ui/include/classes/api/services/CItemPrototype.php*.

Prototyp reguły LLD

Ta klasa jest przeznaczona do pracy z prototypami reguł wykrywania niskiego poziomu.

Odwołania do obiektów:

- Prototyp reguły LLD
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Filtr prototypu reguły LLD
  - Warunek filtra prototypu reguły LLD
- Ścieżka makra prototypu LLD
- Przetwarzanie wstępne prototypu reguły LLD

- Nadpisania prototypu reguły LLD
  - Filtr nadpisania prototypu reguły LLD
    - Warunek filtra nadpisania prototypu reguły LLD
  - Operacja nadpisania prototypu reguły LLD
    - Status operacji nadpisania prototypu reguły LLD
    - Wykrywanie operacji nadpisania prototypu reguły LLD
    - Okres operacji nadpisania prototypu reguły LLD
    - Historia operacji nadpisania prototypu reguły LLD
    - Trendy operacji nadpisania prototypu reguły LLD
    - Istotność operacji nadpisania prototypu reguły LLD
    - Tag operacji nadpisania prototypu reguły LLD
    - Szablon operacji nadpisania prototypu reguły LLD
    - Inwentarz operacji nadpisania prototypu reguły LLD

Dostępne metody:

- discoveryruleprototype.create - utwórz nowe prototypy reguł LLD
- discoveryruleprototype.delete - usuń prototypy reguł LLD
- discoveryruleprototype.get - pobierz prototypy reguł LLD
- discoveryruleprototype.update - zaktualizuj prototypy reguł LLD

Obiekt prototypu reguły LLD

Następujące obiekty są bezpośrednio powiązane z API `discoveryruleprototype`.

Prototyp reguły LLD

Obiekt prototypu reguły wykrywania niskiego poziomu ma następujące właściwości.

Property	Type	Description
itemid	ID	ID prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
ruleid	ID	ID nadrzędnej reguły LLD/prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji tworzenia
delay	string	Interwał aktualizacji prototypu reguły LLD. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej niestandardowych interwałów, oddzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowanych. Akceptuje makra użytkownika. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra zmieszane z tekstem nie są obsługiwane. Interwały elastyczne mogą być zapisane jako dwa makra oddzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), lub jeśli `type` jest ustawione na ”Zabbix agent (active)” (7), a `key_` nie zawiera ”mqtt.get”

Property	Type	Description
hostid	ID	ID hosta, do którego należy prototyp reguły LLD. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie prototypu reguły LLD. Możliwe wartości: 3 - prototyp reguły wykrywania niskiego poziomu; 7 - wykryty prototyp reguły wykrywania niskiego poziomu. Zachowanie właściwości: - *tylko do odczytu*
interfaceid	ID	ID interfejsu hosta prototypu reguły LLD. Zachowanie właściwości: - *wymagane*, jeśli prototyp reguły LLD należy do hosta, a `type` jest ustawione na ”Zabbix agent”, ”IPMI agent”, ”JMX agent” lub ”SNMP agent” - *obsługiwane*, jeśli prototyp reguły LLD należy do hosta, a `type` jest ustawione na ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent”
key_	string	Klucz prototypu reguły LLD. Wymagane jest co najmniej jedno makro LLD. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Nazwa prototypu reguły LLD. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
type	integer	Typ prototypu reguły LLD. Możliwe wartości: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 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; 22 - Browser; 23 - Nested. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
url	string	Ciąg URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Zezwala na wypełnianie wartości podobnie jak w pozycji trapper. Możliwe wartości: 0 - *(domyślnie)* Nie zezwalaj na przyjmowanie danych przychodzących; 1 - Zezwalaj na przyjmowanie danych przychodzących. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent”
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` jest ustawione na ”SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` jest ustawione na ”HTTP agent”: 0 - *(domyślnie)* brak; 1 - basic; 2 - NTLM. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”SSH agent” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
description	string	Opis prototypu reguły LLD.
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Podążaj za przekierowaniami odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - Nie podążaj za przekierowaniami; 1 - *(domyślnie)* Podążaj za przekierowaniami. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
http_proxy	string	Ciąg połączenia proxy HTTP(S). Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”IPMI agent”, a `key_` nie jest ustawione na ”ipmi.get” - *obsługiwane*, jeśli `type` jest ustawione na ”IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych
jmx_endpoint	string	Niestandardowy ciąg połączenia JMX agent. Domyślnie: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”JMX agent”
lifetime	string	Okres czasu, po którym pozycje, które nie są już wykrywane, zostaną usunięte. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Domyślnie: `7d`.
lifetime_type	integer	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: 0 - *(domyślnie)* Usuń po osiągnięciu progu lifetime; 1 - Nie usuwaj; 2 - Usuń natychmiast.
enabled_lifetime	string	Okres czasu, po którym pozycje, które nie są już wykrywane, zostaną wyłączone. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Domyślnie: `0`.
enabled_lifetime_type	integer	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: 0 - Wyłącz po osiągnięciu progu lifetime; 1 - Nie wyłączaj; 2 - *(domyślnie)* Wyłącz natychmiast.
master_itemid	ID	ID nadrzędnej pozycji. Reguła wykrywania nie może być nadrzędną pozycją dla innej reguły wykrywania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do JSON. Możliwe wartości: 0 - *(domyślnie)* Przechowuj surowe dane; 1 - Konwertuj do JSON. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
params	string	Dodatkowe parametry zależne od typu prototypu reguły LLD: - wykonywany skrypt dla reguł LLD SSH i Telnet; - zapytanie SQL dla reguł LLD monitorowania bazy danych; - formuła dla obliczanych reguł LLD; - skrypt dla reguł LLD typu Script i Browser. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”Script” lub ”Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` jest ustawione na ”Script” lub ”Browser”. Tablica obiektów z właściwościami `name` i `value`, gdzie `name` musi być unikalne. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”JMX agent”, a `username` jest ustawione - *obsługiwane*, jeśli `type` jest ustawione na ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
post_type	integer	Typ treści danych POST przechowywanej we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* Surowe dane; 2 - Dane JSON; 3 - Dane XML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”HTTP agent”, a `post_type` jest ustawione na ”JSON data” lub ”XML data” - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent”, a `post_type` jest ustawione na ”Raw data” - *tylko do odczytu* dla obiektów dziedziczonych
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent”, a `authtype` jest ustawione na ”public key”
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent”, a `authtype` jest ustawione na ”public key”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Która część odpowiedzi powinna zostać zapisana. Możliwe wartości, jeśli `request_method` jest ustawione na ”GET”, ”POST” lub ”PUT”: 0 - *(domyślnie)* Treść; 1 - Nagłówki; 2 - Zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` jest ustawione na ”HEAD”: 1 - Nagłówki. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
state	integer	Stan prototypu reguły LLD. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - nieobsługiwany. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Status prototypu reguły LLD. Możliwe wartości: 0 - *(domyślnie)* włączony prototyp reguły LLD; 1 - wyłączony prototyp reguły LLD.

Property	Type	Description
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługuje również makra użytkownika jako część listy oddzielonej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	ID nadrzędnego prototypu reguły LLD szablonu. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m). Akceptuje również makra użytkownika. Możliwy zakres wartości: 1-600s. Domyślnie: ”” - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3) i `key_` nie zaczyna się od ”`vmware.`” ani ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20), a `snmp_oid` zaczyna się od ”`walk[`” lub ”`get[`”, ”Script” (21), ”Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Zabbix trapper”, lub jeśli `type` jest ustawione na ”HTTP agent”, a `allow_traps` jest ustawione na ”Allow to accept incoming data”
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent”, ”TELNET agent”, lub jeśli `type` jest ustawione na ”JMX agent”, a `password` jest ustawione - *obsługiwane*, jeśli `type` jest ustawione na ”Simple check”, ”Database monitor” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych prototypów reguł LLD z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp reguły LLD należy do szablonu
verify_host	integer	Czy weryfikować, że nazwa hosta dla połączenia odpowiada nazwie w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* Nie weryfikuj; 1 - Weryfikuj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
verify_peer	integer	Czy weryfikować, że certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* Nie weryfikuj; 1 - Weryfikuj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Filtr prototypu reguły LLD

Obiekt filtra prototypu reguły LLD definiuje zestaw warunków, które mogą być używane do filtrowania wykrytych obiektów. Ma następujące właściwości:

Właściwość	Type	Opis
conditions	object/array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*

Właściwość	Type	Opis
evaltype	integer	Metoda oceny warunków filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra. Wyrażenie zawiera identyfikatory odwołujące się do określonych warunków filtra przez ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do określonych warunków filtra przez ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „wyrażenie niestandardowe”

Warunek filtra prototypu reguły LLD

Obiekt warunku filtra prototypu reguły LLD definiuje oddzielne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Zachowanie właściwości: - *wymagane*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*, jeśli `operator` jest ustawiony na „matches regular expression” lub „does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` obiektu filtra prototypu reguły LLD jest ustawiony na „custom expression”
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* pasuje do wyrażenia regularnego; 9 - nie pasuje do wyrażenia regularnego; 12 - istnieje; 13 - nie istnieje.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod discoveryruleprototype.get oraz discoveryruleprototype.create.

Ścieżka makra LLD

Ścieżka makra LLD ma następujące właściwości:

Właściwość	Type	Opis
lld_macro	string	Makro LLD. Zachowanie właściwości: - *wymagane*
path	string	Selektor wartości, która zostanie przypisana do odpowiadającego jej makra. Zachowanie właściwości: - *wymagane*

Wstępne przetwarzanie prototypu reguły LLD

Obiekt wstępnego przetwarzania prototypu reguły LLD ma następujące właściwości.

Właściwość	Type	Opis
type	integer	Typ opcji wstępnego przetwarzania. Możliwe wartości: 5 - Wyrażenie regularne; 11 - XML XPath; 12 - JSONPath; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdź błąd w JSON; 17 - Sprawdź błąd w XML; 20 - Odrzuć niezmienione z heartbeat; 21 - JavaScript; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zamień; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Property behavior: - *wymagane*
params	string	Dodatkowe parametry używane przez opcję wstępnego przetwarzania. Wiele parametrów jest rozdzielanych znakiem nowej linii (\n). Property behavior: - *wymagane*, jeśli `type` jest ustawione na ”Wyrażenie regularne” (5), ”XML XPath” (11), ”JSONPath” (12), ”Pasuje do wyrażenia regularnego” (14), ”Nie pasuje do wyrażenia regularnego” (15), ”Sprawdź błąd w JSON” (16), ”Sprawdź błąd w XML” (17), ”Odrzuć niezmienione z heartbeat” (20), ”JavaScript” (21), ”Prometheus do JSON” (23), ”CSV do JSON” (24), ”Zamień” (25), ”Wartość SNMP walk” (28), ”SNMP walk do JSON” (29) lub ”Wartość SNMP get” (30)

Właściwość	Type	Opis
error_handler	integer	Typ akcji używany w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzuć wartość; 2 - Ustaw własną wartość; 3 - Ustaw własny komunikat błędu. Property behavior: - *wymagane*, jeśli `type` jest ustawione na ”Wyrażenie regularne” (5), ”XML XPath” (11), ”JSONPath” (12), ”Pasuje do wyrażenia regularnego” (14), ”Nie pasuje do wyrażenia regularnego” (15), ”Sprawdź błąd w JSON” (16), ”Sprawdź błąd w XML” (17), ”Prometheus do JSON” (23), ”CSV do JSON” (24), ”XML do JSON” (27), ”Wartość SNMP walk” (28), ”SNMP walk do JSON” (29) lub ”Wartość SNMP get” (30)
error_handler_params	string	Parametry obsługi błędów. Property behavior: - *wymagane*, jeśli `error_handler` jest ustawione na ”Ustaw własną wartość” lub ”Ustaw własny komunikat błędu”

Dla każdego typu wstępnego przetwarzania obsługiwane są następujące parametry i procedury obsługi błędów.

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
5	Wyrażenie regularne	wzorzec[1]	wynik[2]		0, 1, 2, 3
11	XML XPath	ścieżka[3]			0, 1, 2, 3
12	JSONPath	ścieżka[3]			0, 1, 2, 3
14	Pasuje do wyrażenia regularnego	wzorzec[1]			0, 1, 2, 3
15	Nie pasuje do wyrażenia regularnego	wzorzec[1]			0, 1, 2, 3
16	Sprawdź błąd w JSON	ścieżka[3]			0, 1, 2, 3
17	Sprawdź błąd w XML	ścieżka[3]			0, 1, 2, 3
20	Odrzuć niezmienione z heartbeat	sekundy^4, 5^			
21	JavaScript	skrypt[2]			

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
23	Prometheus do JSON	wzorzec^5, 6^			0, 1, 2, 3
24	CSV do JSON	znak[2]	znak[2]	0,1	0, 1, 2, 3
25	Zamień	szukany ciąg[2]	zamiennik[2]		
27	XML do JSON				0, 1, 2, 3
28	Wartość SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3
29	SNMP walk do JSON[7]	Nazwa pola[2]	Prefiks OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Wartość SNMP get	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] wyrażenie regularne
 [2] ciąg znaków
 [3] JSONPath lub XML XPath
 [4] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [5] makro użytkownika
 [6] Wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy składnik wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [7] Obsługuje wiele rekordów "Nazwa pola,Prefiks OID,Format", rozdzielonych znakiem nowej linii.

Nadpisania prototypów reguł LLD

Obiekt nadpisań prototypów reguł LLD definiuje zestaw reguł (filtrów, warunków i operacji), które są używane do nadpisywania właściwości różnych obiektów prototypów. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Unikalna nazwa nadpisania. Zachowanie właściwości: - *wymagane*
step	integer	Unikalny numer porządkowy nadpisania. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
stop	integer	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli wystąpi dopasowanie. Możliwe wartości: 0 - *(domyślnie)* nie zatrzymuj przetwarzania nadpisań; 1 - zatrzymaj przetwarzanie nadpisań, jeśli filtr pasuje.
filter	object	Filtr nadpisania.
operations	object/array	Operacje nadpisania.

Filtr nadpisania prototypu reguły LLD

Obiekt filtra nadpisania prototypu reguły LLD definiuje zestaw warunków, które po dopasowaniu do wykrytego obiektu powodują zastosowanie nadpisania. Ma on następujące właściwości:

Property	Type	Description
conditions	object/array	Zestaw warunków filtra nadpisania używanych do dopasowywania wykrytych obiektów. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia w formule. Property behavior: - *required*
evaltype	integer	Metoda oceny warunku filtra nadpisania. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Property behavior: - *required*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra nadpisania. Wyrażenie zawiera identyfikatory odwołujące się do konkretnych warunków filtra nadpisania za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Property behavior: - *read-only*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów nadpisania z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do konkretnych warunków filtra nadpisania za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra nadpisania: żaden warunek nie może pozostać nieużyty ani pominięty. Property behavior: - *required* jeśli `evaltype` jest ustawione na ”custom expression”

Warunek filtra nadpisania prototypu reguły LLD

Obiekt warunku filtra nadpisania prototypu reguły LLD definiuje osobne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Property behavior: - *wymagane*
value	string	Wartość do porównania. Property behavior: - *wymagane*, jeśli `operator` jest ustawiony na „matches regular expression” lub „does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków. Property behavior: - *wymagane*, jeśli `evaltype` obiektu LLD rule prototype override filter object jest ustawione na „custom expression”
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* matches regular expression; 9 - does not match regular expression; 12 - exists; 13 - does not exist.

Operacja nadpisania prototypu reguły LLD

Operacja nadpisania prototypu reguły LLD jest połączeniem warunków i działań wykonywanych na obiekcie prototypu. Ma następujące właściwości:

Właściwość	Type	Opis
operationobject	integer	Typ wykrytego obiektu, dla którego ma zostać wykonane działanie. Możliwe wartości: 0 - Prototyp pozycji; 1 - Prototyp wyzwalacza; 2 - Prototyp wykresu; 3 - Prototyp hosta. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek nadpisania operator. Możliwe wartości: 0 - *(domyślnie)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 8 - pasuje; 9 - nie pasuje.
value	string	Wzorzec do dopasowania do nazwy prototypu pozycji, wyzwalacza, wykresu lub hosta, zależnie od wybranego obiektu.
opstatus	object	Obiekt statusu operacji nadpisania dla obiektów prototypu pozycji, wyzwalacza i hosta.
opdiscover	object	Obiekt statusu wykrywania operacji nadpisania (wszystkie typy obiektów).
opperiod	object	Obiekt okresu operacji nadpisania (interwału aktualizacji) dla obiektu prototypu pozycji.
ophistory	object	Obiekt historii operacji nadpisania dla obiektu prototypu pozycji.

Właściwość	Type	Opis
optrends	object	Obiekt trendów operacji nadpisania dla obiektu prototypu pozycji.
opseverity	object	Obiekt ważności operacji nadpisania dla obiektu prototypu wyzwalacza.
optag	object/array	Obiekt tagu operacji nadpisania dla obiektów prototypu wyzwalacza i hosta.
optemplate	object/array	Obiekt szablonu operacji nadpisania dla obiektu prototypu hosta.
opinventory	object	Obiekt inwentaryzacji operacji nadpisania dla obiektu prototypu hosta.

Status operacji nadpisania prototypu reguły LLD

Status operacji nadpisania prototypu reguły LLD ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
status	integer	Nadpisuje status dla wybranego obiektu. Możliwe wartości: 0 - Utwórz jako włączone; 1 - Utwórz jako wyłączone. Zachowanie właściwości: - *wymagane*

Operacja discover nadpisania prototypu reguły LLD

Status discover operacji nadpisania prototypu reguły LLD jest ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
discover	integer	Nadpisuje status discover dla wybranego obiektu. Możliwe wartości: 0 - Tak, kontynuuj wykrywanie obiektów; 1 - Nie, nowe obiekty nie będą wykrywane, a istniejące zostaną oznaczone jako utracone. Zachowanie właściwości: - *wymagane*

Okres operacji nadpisania prototypu reguły LLD

Okres operacji nadpisania prototypu reguły LLD jest wartością interwału aktualizacji ustawianą dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
delay	string	Nadpisuje interwał aktualizacji prototypu pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej interwałów niestandardowych, wszystkie rozdzielone średnikami. Interwały niestandardowe mogą stanowić mieszankę interwałów elastycznych i harmonogramowanych. Akceptuje makra użytkownika lub makra LLD. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra zmieszane z tekstem nie są obsługiwane. Interwały elastyczne mogą być zapisane jako dwa makra rozdzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *required*

Historia operacji nadpisania prototypu reguły LLD

Wartość historii operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
history	string	Nadpisuje historię prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane historii powinny być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *wymagane*

Trendy operacji nadpisania prototypu reguły LLD

Wartość trendów operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
trends	string	Nadpisuje trendy prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane trendów powinny być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *wymagane*

Ważność operacji nadpisania prototypu reguły LLD

Wartość ważności operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytego wyzwalacza. Ma następujące właściwości:

Właściwość	Type	Opis
severity	integer	Nadpisuje ważność prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa. Zachowanie właściwości: - *wymagane*

Znacznik operacji nadpisania prototypu reguły LLD

Obiekt znacznika operacji nadpisania prototypu reguły LLD zawiera nazwę i wartość znacznika, które są ustawiane dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Type	Opis
tag	string	Nowa nazwa znacznika. Zachowanie właściwości: - *wymagane*
value	string	Nowa wartość znacznika.

Szablon operacji nadpisania prototypu reguły LLD

Obiekt szablonu operacji nadpisania prototypu reguły LLD, który jest połączony z wykrytym hostem. Ma następujące właściwości:

Właściwość	Type	Opis
templateid	ID	Nadpisuje szablon szablonów połączonych z prototypem hosta. Zachowanie właściwości: - *wymagane*

Wartość trybu inwentaryzacji operacji nadpisania prototypu reguły LLD

Wartość trybu inwentaryzacji operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytego hosta. Ma następujące właściwości:

Właściwość	Type	Opis
inventory_mode	integer	Nadpisuje tryb inwentaryzacji prototypu hosta. Możliwe wartości: -1 - wyłączone; 0 - *(domyślnie)* ręczne; 1 - automatyczne. Zachowanie właściwości: - *wymagane*

discoveryruleprototype.create

Opis

`object discoveryruleprototype.create(object/array lldRules)`

Ta metoda umożliwia tworzenie nowych prototypów reguł LLD.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy reguł LLD do utworzenia.

Oprócz standardowych właściwości prototypu reguły LLD, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr prototypu reguły LLD dla reguły LLD.
preprocessing	object/array	Opcje przetwarzania wstępnego prototypu reguły LLD.
lld_macro_paths	object/array	Opcje lld_macro_path prototypu reguły LLD.
overrides	object/array	Opcje nadpisań prototypu reguły LLD.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów reguł LLD we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów reguł LLD.

Przykłady

Tworzenie prototypu reguły LLD

Utwórz prototyp reguły LLD (Typ: Zagnieżdżony), aby wykrywać tablespace'y w instancji bazy danych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.create",
    "params": {
        "name": "Discover tablespaces for {#DB}",
        "key_": "db.tablespace.discovery[{#DB}]",
        "hostid": "10084",
        "ruleid": "47251",
        "type": 23
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47252"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr prototypu reguły LLD
- Ścieżki makr LLD
- Wstępne przetwarzanie prototypu reguły LLD

Źródło

CDiscoveryRulePrototype::create() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.delete

Opis

`object discoveryruleprototype.delete(array lldRuleIds)`

Ta metoda umożliwia usuwanie prototypów reguł LLD.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(array)` ID prototypów reguł LLD do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów reguł LLD we właściwości `ruleids`.

Przykłady

Usuwanie wielu prototypów reguł LLD

Usuń dwa prototypy reguł LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.delete",
    "params": [
        "47252",
        "47253"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "ruleids": [
            "47252",
            "47253"
        ]
    },
    "id": 1
}
```

Źródło

CDiscoveryRulePrototype::delete() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.get

Opis

`integer/array discoveryruleprototype.get(object parameters)`

Metoda umożliwia pobranie prototypów reguł LLD zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
itemids	ID/array	Zwraca tylko prototypy reguł LLD o podanych ID.
groupids	ID/array	Zwraca tylko prototypy reguł LLD należące do hostów z podanych grup.
hostids	ID/array	Zwraca tylko prototypy reguł LLD należące do podanych hostów.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko prototypy reguł LLD odziedziczone z szablonu.
interfaceids	ID/array	Zwraca tylko prototypy reguł LLD używające podanych interfejsów hosta.
monitored	boolean	Jeśli ustawione na `true`, zwraca tylko włączone prototypy reguł LLD należące do monitorowanych hostów.
templated	boolean	Jeśli ustawione na `true`, zwraca tylko prototypy reguł LLD należące do szablonów.
templateids	ID/array	Zwraca tylko prototypy reguł LLD należące do podanych szablonów.
selectDiscoveryRule	query	Zwraca właściwość `discoveryRule` z nadrzędną regułą LLD dla prototypu reguły LLD.
selectDiscoveryRulePrototype	query	Zwraca właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD dla prototypu reguły LLD.
selectDiscoveryRulePrototypes	query	Zwraca właściwość `discoveryRulePrototypes` z podrzędnymi prototypami reguł LLD należącymi do prototypu reguły LLD. Obsługuje `count`.
selectFilter	query	Zwraca właściwość `filter` z danymi filtra używanego przez prototyp reguły LLD.
selectGraphs	query	Zwraca właściwość `graphs` z prototypami wykresów należącymi do prototypu reguły LLD. Obsługuje `count`.
selectHostPrototypes	query	Zwraca właściwość `hostPrototypes` z prototypami hostów należącymi do prototypu reguły LLD. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy prototyp reguły LLD.
selectItems	query	Zwraca właściwość `items` z prototypami pozycji należącymi do prototypu reguły LLD. Obsługuje `count`.
selectTriggers	query	Zwraca właściwość `triggers` z prototypami wyzwalaczy należącymi do prototypu reguły LLD. Obsługuje `count`.
selectLLDMacroPaths	query	Zwraca właściwość `lld_macro_paths` z listą makr LLD i ścieżek do wartości przypisanych do każdego odpowiadającego makra.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami przetwarzania wstępnego prototypu reguły LLD.
selectOverrides	query	Zwraca właściwość `lld_rule_overrides` z listą filtrów, warunków i operacji nadpisywania wykonywanych na obiektach prototypów.
filter	object	Zwraca tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Nie obsługuje właściwości typu `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp reguły LLD.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectItems`, `selectGraphs`, `selectTriggers`.

Parameter	Type	Description
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Wartości zwracane

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie prototypów reguł wykrywania z hosta

Pobierz wszystkie prototypy reguł wykrywania dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.get",
    "params": {
        "hostids": "10084"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "47253",
            "type": "23",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Wykryj tablespace dla {#DB}",
            "key_": "db.tablespace.discovery[{#DB}]",
            "delay": "0",
            "history": "31d",
            "trends": "365d",
            "status": "0",
            "value_type": "4",
            "trapper_hosts": "",
            "units": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
```

```
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "3",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "lifetime": "7d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "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",
            "discover": "0",
            "uuid": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "parameters": []
        }
    ],
    "id": 1
}
```


Zobacz także

- Prototyp wykresu
- Host
- Prototyp pozycji
- Filtr prototypu reguły LLD
- Prototyp wyzwalacza

Source

CDiscoveryRulePrototype::get() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.update

Opis

`object discoveryruleprototype.update(object/array lldRules)`

Ta metoda umożliwia aktualizację istniejących prototypów reguł LLD.

Należy pamiętać, że aktualizacja już wykrytych prototypów jest ograniczona.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object/array) Właściwości prototypu reguły LLD do zaktualizowania.

Właściwość itemid musi być zdefiniowana dla każdego prototypu reguły LLD, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu reguły LLD, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr prototypu reguły LLD zastępujący istniejący filtr.
preprocessing	object/array	Opcje preprocessingu prototypu reguły LLD zastępujące istniejące opcje preprocessingu. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
lld_macro_paths	object/array	Opcje lld_macro_path prototypu reguły LLD zastępujące istniejące opcje lld_macro_path. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
overrides	object/array	Opcje nadpisań prototypu reguły LLD zastępujące istniejące opcje nadpisań. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów reguł LLD we właściwości itemids.

Przykłady

Aktualizacja opcji preprocessing prototypu reguły LLD

Zaktualizuj prototyp reguły LLD za pomocą reguły preprocessing JSONPath. Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.update",
    "params": {
        "itemid": "47253",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.tablespaces",
                "error_handler": 1
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47253"
        ]
    },
    "id": 1
}
```

Źródło

CDiscoveryRulePrototype::update() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

Prototyp wykresu

Ta klasa jest przeznaczona do pracy z prototypami wykresów.

Odwołania do obiektów:

- Prototyp wykresu

Dostępne metody:

- graphprototype.create - utwórz nowe prototypy wykresów
- graphprototype.delete - usuń prototypy wykresów
- graphprototype.get - pobierz prototypy wykresów
- graphprototype.update - zaktualizuj prototypy wykresów

Obiekt prototypu grafu

Następujące obiekty są bezpośrednio powiązane z API `graphprototype`.

Prototyp wykresu

Obiekt prototypu wykresu ma następujące właściwości.

Właściwość	Typ	Opis
graphid	ID	ID prototypu wykresu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
height	integer	Wysokość prototypu wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa prototypu wykresu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość prototypu wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie prototypu wykresu. Możliwe wartości: 2 - prototyp wykresu; 6 - wykryty prototyp wykresu Zachowanie właściwości: - *tylko do odczytu*
graphtype	integer	Typ układu prototypu wykresu. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - skumulowany; 2 - kołowy; 3 - rozdzielony.
percent_left	float	Lewy percentyl. Domyślnie: 0.
percent_right	float	Prawy percentyl. Domyślnie: 0.

Właściwość	Typ	Opis
show_3d	integer	Czy wykryte wykresy kołowe i rozdzielone mają być wyświetlane w 3D. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj w 2D; 1 - wyświetlaj w 3D.
show_legend	integer	Czy legenda ma być wyświetlana na wykrytym wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
show_work_period	integer	Czy czas pracy ma być wyświetlany na wykrytym wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
templateid	ID	ID prototypu wykresu nadrzędnego szablonu. Zachowanie właściwości: - *tylko do odczytu*
yaxismax	float	Stała maksymalna wartość osi Y.
yaxismin	float	Stała minimalna wartość osi Y.
ymax_itemid	ID	ID pozycji używanej jako maksymalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymax_type` było ustawione na ”calculated”.
ymax_type	integer	Metoda obliczania wartości maksymalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
ymin_itemid	ID	ID pozycji używanej jako minimalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymin_type` było ustawione na ”calculated”.
ymin_type	integer	Metoda obliczania wartości minimalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
discover	integer	Status wykrywania prototypu wykresu. Możliwe wartości: 0 - *(domyślnie)* nowe wykresy będą wykrywane; 1 - nowe wykresy nie będą wykrywane, a istniejące wykresy zostaną oznaczone jako utracone.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych prototypów wykresów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp wykresu należy do szablonu

graphprototype.create

Opis

`object graphprototype.create(object/array graphPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów wykresów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy wykresów do utworzenia.

Oprócz standardowych właściwości prototypu wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Elementy wykresu, które mają zostać utworzone dla prototypów wykresów. Elementy wykresu mogą odwoływać się zarówno do pozycji, jak i prototypów pozycji, ale musi być obecny co najmniej jeden prototyp pozycji. Zachowanie parametrów: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów wykresów we właściwości `graphids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów wykresów.

Przykłady

Tworzenie prototypu wykresu

Utwórz prototyp wykresu z dwiema pozycjami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.create",
    "params": {
        "name": "Disk space usage {#FSNAME}",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pozycja wykresu

Źródło

CGraphPrototype::create() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.delete

Opis

`object graphprototype.delete(array graphPrototypeIds)`

Ta metoda pozwala na usuwanie prototypów grafów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametery

`(array)` ID prototypów wykresów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów wykresów we właściwości `graphids`.

Przykłady

Usuwanie wielu prototypów wykresów

Usuń dwa prototypy wykresów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "graphprototype.delete",
"params": [
"652",
"653"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"graphids": [
"652",
"653"
]
},
"id": 1
}
```

Źródło

CGraphPrototype::delete() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.get

Opis

`integer/array graphprototype.get(object parameters)`

Ta metoda umożliwia pobieranie prototypów wykresów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
discoveryids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych reguł wykrywania.
graphids	ID/array	Zwracaj tylko prototypy wykresów o podanych ID.
groupids	ID/array	Zwracaj tylko prototypy wykresów należące do hostów lub szablonów w podanych grupach hostów lub grupach szablonów.
hostids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych hostów.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wykresów odziedziczone z szablonu.
itemids	ID/array	Zwracaj tylko prototypy wykresów zawierające podane prototypy pozycji.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wykresów należące do szablonów.
templateids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych szablonów.
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu wykresu. Obiekt wykrywania prototypu wykresu łączy wykryty prototyp wykresu z prototypem wykresu, z którego został wykryty. Ma następujące właściwości: `parent_graphid` - `(ID)` ID prototypu wykresu, z którego wykres został utworzony; `status` - `(int)` status wykrywania wykresu: 0 - *(domyślnie)* prototyp wykresu jest wykryty, 1 - prototyp wykresu nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, kiedy prototyp wykresu, który nie jest już wykrywany, zostanie usunięty.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą LLD, do której należy prototyp wykresu.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp wykresu.
selectGraphItems	query	Zwracaj właściwość `gitems` z elementami wykresu używanymi w prototypie wykresu.
selectHostGroups	query	Zwracaj właściwość `hostgroups` z grupami hostów, do których należy prototyp wykresu.
selectHosts	query	Zwracaj właściwość `hosts` z hostami, do których należy prototyp wykresu.
selectItems	query	Zwracaj właściwość `items` z pozycjami i prototypami pozycji używanymi w prototypie wykresu.
selectTemplateGroups	query	Zwracaj właściwość `templategroups` z grupami szablonów, do których należy prototyp wykresu.
selectTemplates	query	Zwracaj właściwość `templates` z szablonami, do których należy prototyp wykresu.

Parametr	Type	Opis
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp wykresu; `hostid` - ID hosta, do którego należy prototyp wykresu.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `graphid`, `name`, `graphtype`, `discovered`.
countOutput	boolean	Te parametry są opisane w reference commentary.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów wykresów z reguły LLD

Pobierz wszystkie prototypy wykresów z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "1017",
            "name": "Wykorzystanie miejsca na dysku {#FSNAME}",
            "width": "600",
            "height": "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",
            "flags": "2"
            "discover": "0"
        }
    ],
    "id": 1
}
```

Zobacz także

- Reguła wykrywania
- Pozycja wykresu
- Pozycja
- Host
- Grupa hostów
- Szablon
- Grupa szablonów

Źródło

CGraphPrototype::get() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.update

Opis

`object graphprototype.update(object/array graphPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów wykresów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu wykresu do zaktualizowania.

Właściwość `graphid` musi być zdefiniowana dla każdego prototypu wykresu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Pozycje wykresu służące do zastąpienia istniejących pozycji wykresu. Jeśli pozycja wykresu ma zdefiniowaną właściwość `gitemid`, zostanie zaktualizowana; w przeciwnym razie zostanie utworzona nowa pozycja wykresu.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów wykresów we właściwości `graphids`.

Przykłady

Zmiana rozmiaru prototypu wykresu

Zmień rozmiar prototypu wykresu na 1100 na 400 pikseli.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.update",
    "params": {
        "graphid": "439",
        "width": 1100,
        "height": 400
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Źródło

CGraphPrototype::update() w *ui/include/classes/api/services/CGraphPrototype.php*.

Prototyp wyzwalacza

Ta klasa jest przeznaczona do pracy z prototypami wyzwalaczy.

Odwołania do obiektów:

- Prototyp wyzwalacza
- Tag prototypu wyzwalacza

Dostępne metody:

- triggerprototype.create - tworzenie nowych prototypów wyzwalaczy
- triggerprototype.delete - usuwanie prototypów wyzwalaczy
- triggerprototype.get - pobieranie prototypów wyzwalaczy
- triggerprototype.update - aktualizowanie prototypów wyzwalaczy

Obiekt prototypu wyzwalacza

Następujące obiekty są bezpośrednio powiązane z API `triggerprototype`.

Prototyp wyzwalacza

Obiekt prototypu wyzwalacza ma następujące właściwości.

Property	Type	Description
triggerid	ID	ID prototypu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Nazwa prototypu wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Property	Type	Description
expression	string	Wyrażenie wyzwalacza. Musi zawierać co najmniej jeden prototyp pozycji. Prototypy pozycji z `value_type` ustawionym na 5 (binary) nie mogą być używane w wyrażeniach wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
event_name	string	Nazwa zdarzenia generowanego przez wyzwalacz.
opdata	string	Dane operacyjne.
comments	string	Dodatkowe komentarze do prototypu wyzwalacza.
priority	integer	Ważność prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* nie sklasyfikowano; 1 - informacja; 2 - ostrzeżenie; 3 - przeciętny; 4 - wysoki; 5 - katastrofa.
status	integer	Określa, czy prototyp wyzwalacza jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
flags	integer	Pochodzenie prototypu wyzwalacza. Możliwe wartości: 2 - prototyp wyzwalacza; 6 - wykryty prototyp wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*
templateid	ID	ID nadrzędnego prototypu wyzwalacza szablonu. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Określa, czy prototyp wyzwalacza może generować wiele zdarzeń problemu. Możliwe wartości: 0 - *(domyślnie)* nie generuj wielu zdarzeń; 1 - generuj wiele zdarzeń.
url	string	Adres URL powiązany z prototypem wyzwalacza.
url_name	string	Etykieta adresu URL powiązanego z prototypem wyzwalacza.
recovery_mode	integer	Tryb generowania zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* Expression; 1 - Recovery expression; 2 - None.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza. Musi zawierać co najmniej jeden prototyp pozycji. Prototypy pozycji z `value_type` ustawionym na 5 (binary) nie mogą być używane w wyrażeniach odzyskiwania wyzwalacza.
correlation_mode	integer	Zamykanie zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* wszystkie problemy; 1 - wszystkie problemy, jeśli wartości tagów są zgodne.

Property	Type	Description
correlation_tag	string	Tag do dopasowania.
manual_close	integer	Zezwalaj na ręczne zamknięcie. Możliwe wartości: 0 - *(domyślnie)* nie; 1 - tak.
discover	integer	Status wykrywania prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* nowe wyzwalacze będą wykrywane; 1 - nowe wyzwalacze nie będą wykrywane, a istniejące wyzwalacze zostaną oznaczone jako utracone.
uuid	string	Uniwersalny unikalny identyfikator, używany do powiązania importowanych prototypów wyzwalacza z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane* jeśli prototyp wyzwalacza należy do szablonu

Znacznik prototypu wyzwalacza

Obiekt znacznika prototypu wyzwalacza ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa znacznika prototypu wyzwalacza. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika prototypu wyzwalacza.

triggerprototype.create

Opis

`object triggerprototype.create(object/array triggerPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów wyzwalaczy.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy wyzwalaczy do utworzenia.

Oprócz standardowych właściwości prototypu wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze i prototypy wyzwalaczy, od których zależy prototyp wyzwalacza. Wyzwalacze muszą mieć zdefiniowaną tylko właściwość `triggerid`.
tags	array	Tagi prototypu wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów wyzwalaczy we właściwości `triggerids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów wyzwalaczy.

Przykłady

Tworzenie prototypu wyzwalacza

Utwórz prototyp wyzwalacza, aby wykrywać, kiedy system plików ma mniej niż 20% wolnego miejsca na dysku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.create",
    "params": {
        "description": "Wolne miejsce na dysku jest mniejsze niż 20% na woluminie {#FSNAME}",
        "expression": "last(/Zabbix server/vfs.fs.size[{#FSNAME},pfree])<20",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17372"
        ]
    },
    "id": 1
}
```

Źródło

CTriggerPrototype::create() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.delete

Opis

`object triggerprototype.delete(array triggerPrototypeIds)`

Ta metoda pozwala na usuwanie prototypów wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Role użytkownika.

Parametry

`(array)` ID prototypów wyzwalaczy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów wyzwalaczy we właściwości `triggerids`.

Przykłady

Usuwanie wielu prototypów wyzwalaczy

Usuń dwa prototypy wyzwalaczy.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "triggerprototype.delete",
"params": [
"12002",
"12003"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"triggerids": [
"12002",
"12003"
]
},
"id": 1
}
```

Źródło

CTriggerPrototype::delete() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.get

Opis

`integer/array triggerprototype.get(object parameters)`

Metoda umożliwia pobieranie prototypów wyzwalaczy zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
active	flag	Zwracaj tylko włączone prototypy wyzwalaczy, które należą do monitorowanych hostów.
discoveryids	ID/array	Zwracaj tylko prototypy wyzwalaczy, które należą do podanych reguł LLD.
functions	string/array	Zwracaj tylko wyzwalacze, które używają podanych funkcji. Listę obsługiwanych funkcji znajdziesz na stronie Supported functions.
group	string	Zwracaj tylko prototypy wyzwalaczy, które należą do hostów lub szablonów z grup hostów lub grup szablonów o podanej nazwie.
groupids	ID/array	Zwracaj tylko prototypy wyzwalaczy, które należą do hostów lub szablonów z podanych grup hostów lub grup szablonów.
host	string	Zwracaj tylko prototypy wyzwalaczy, które należą do hostów o podanej nazwie.
hostids	ID/array	Zwracaj tylko prototypy wyzwalaczy, które należą do podanych hostów.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wyzwalaczy odziedziczone z szablonu.
maintenance	boolean	Jeśli ustawione na `true`, zwracaj tylko włączone prototypy wyzwalaczy, które należą do hostów w konserwacji.

Parameter	Type	Description
min_severity	integer	Zwracaj tylko prototypy wyzwalaczy o ważności większej lub równej podanej ważności.
monitored	flag	Zwracaj tylko włączone prototypy wyzwalaczy, które należą do monitorowanych hostów i zawierają wyłącznie włączone pozycje.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wyzwalaczy, które należą do szablonów.
templateids	ID/array	Zwracaj tylko prototypy wyzwalaczy, które należą do podanych szablonów.
triggerids	ID/array	Zwracaj tylko prototypy wyzwalaczy o podanych ID.
expandExpression	flag	Rozszerz funkcje i makra w wyrażeniu wyzwalacza.
selectDependencies	query	Zwracaj prototypy wyzwalaczy oraz wyzwalacze, od których zależy prototyp wyzwalacza, w właściwości `dependencies`.
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu wyzwalacza. Obiekt wykrywania prototypu wyzwalacza łączy wykryty prototyp wyzwalacza z prototypem wyzwalacza, z którego został wykryty. Ma następujące właściwości: `parent_triggerid` - `(ID)` ID prototypu wyzwalacza, na podstawie którego utworzono prototyp wyzwalacza; `status` - `(int)` status wykrywania prototypu wyzwalacza: 0 - *(domyślnie)* prototyp wyzwalacza został wykryty, 1 - prototyp wyzwalacza nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, po którym prototyp wyzwalacza, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, po którym prototyp wyzwalacza, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik informujący, czy prototyp wyzwalacza został wyłączony przez regułę LLD, czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj regułę LLD, do której należy prototyp wyzwalacza, w właściwości `discoveryRule`.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp wyzwalacza.
selectFunctions	query	Zwracaj funkcje używane w prototypie wyzwalacza w właściwości `functions`. Obiekty funkcji reprezentują funkcje używane w wyrażeniu wyzwalacza i mają następujące właściwości: `functionid` - `(ID)` ID funkcji; `itemid` - `(ID)` ID pozycji używanej w funkcji; `function` - `(string)` nazwa funkcji; `parameter` - `(string)` parametr przekazany do funkcji. Parametr zapytania jest zastępowany symbolem $ w zwracanym ciągu.
selectHostGroups	query	Zwracaj grupy hostów, do których należy prototyp wyzwalacza, w właściwości `hostgroups`.
selectHosts	query	Zwracaj hosty, do których należy prototyp wyzwalacza, w właściwości `hosts`.
selectItems	query	Zwracaj pozycje i prototypy pozycji używane przez prototyp wyzwalacza w właściwości `items`.
selectTags	query	Zwracaj tagi prototypu wyzwalacza w właściwości `tags`.
selectTemplateGroups	query	Zwracaj grupy szablonów, do których należy prototyp wyzwalacza, w właściwości `templategroups`.

Parameter	Type	Description
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp wyzwalacza; `hostid` - ID hosta, do którego należy prototyp wyzwalacza.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki będą sortowane według `host`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `triggerid`, `description`, `status`, `priority`, `discovered`.
countOutput	boolean	Te parametry są opisane w reference commentary.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie prototypów wyzwalaczy z reguły LLD

Pobierz wszystkie prototypy wyzwalaczy oraz ich funkcje z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": "extend",
        "selectFunctions": "extend",
        "discoveryids": "22450"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13272",
```

```
            "expression": "{12598}<20",
            "description": "Free inodes is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "6ce467d05e8745409a177799bed34bb3",
            "url_name": "",
            "functions": [
                {
                    "functionid": "12598",
                    "itemid": "22454",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        },
        {
            "triggerid": "13266",
            "expression": "{13500}<20",
            "description": "Free disk space is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "74a1fc62bfe24b7eabe4e244c70dc384",
            "url_name": "",
            "functions": [
                {
                    "functionid": "13500",
                    "itemid": "22686",
                    "parameter": "$",
```

```
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie określonego prototypu wyzwalacza z tagami

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17373",
            "description": "Wolne miejsce na dysku jest mniejsze niż 20% na woluminie {#FSNAME}",
            "tags": [
                {
                    "tag": "volume",
                    "value": "{#FSNAME}"
                },
                {
                    "tag": "type",
                    "value": "{#FSTYPE}"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Reguła wykrywania
- Pozycja
- Host
- Grupa hostów
- Grupa szablonów

Źródło

CTriggerPrototype::get() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.update

Opis

`object triggerprototype.update(object/array triggerPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w rolach użytkowników.

Parametry

`(object/array)` Właściwości prototypu wyzwalacza do zaktualizowania.

Właściwość `triggerid` musi być zdefiniowana dla każdego prototypu wyzwalacza, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze i prototypy wyzwalaczy, od których zależy prototyp wyzwalacza. Dla wyzwalaczy musi być zdefiniowana tylko właściwość `triggerid`.
tags	array	Tagi prototypu wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów wyzwalaczy we właściwości `triggerids`.

Przykłady

Włączanie prototypu wyzwalacza

Włącz prototyp wyzwalacza, to znaczy ustaw jego status na ”0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie tagów prototypu wyzwalacza

Zastąp tagi dla jednego prototypu wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "17373",
```

```
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Źródło

CTriggerPrototype::update() w *ui/include/classes/api/services/CTriggerPrototype.php*.

Proxy

Ta klasa jest przeznaczona do pracy z proxy.

Odwołania do obiektów:

- Proxy

Dostępne metody:

- proxy.create - tworzenie nowych proxy
- proxy.delete - usuwanie proxy
- proxy.get - pobieranie proxy
- proxy.update - aktualizacja proxy

Obiekt proxy

Następujące obiekty są bezpośrednio powiązane z API `proxy`.

Proxy

Obiekt proxy ma następujące właściwości.

Property	Type	Description
proxyid	ID	ID proxy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa proxy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Property	Type	Description
proxy_groupid	ID	ID grupy proxy. 0 - proxy nie jest przypisany do żadnej grupy proxy.
local_address	string	Adres dla aktywnych agentów. Adres IP lub nazwa DNS, z którą ma zostać nawiązane połączenie. Zachowanie właściwości: - *wymagane*, jeśli `proxy_groupid` nie jest równe 0
local_port	string	Numer portu lokalnego proxy, z którym ma zostać nawiązane połączenie. Obsługiwane są makra użytkownika. Domyślnie: 10051. Zachowanie właściwości: - *obsługiwane*, jeśli `proxy_groupid` nie jest równe 0
operating_mode	integer	Typ proxy. Możliwe wartości: 0 - aktywny proxy; 1 - pasywny proxy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	text	Opis proxy.
lastaccess	timestamp	Czas, kiedy proxy ostatnio połączył się z serwerem. Zachowanie właściwości: - *tylko do odczytu*
address	string	Adres IP lub nazwa DNS, z którą ma zostać nawiązane połączenie. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagane*, jeśli tryb działania Zabbix proxy jest pasywny
port	string	Numer portu, z którym ma zostać nawiązane połączenie. Obsługiwane są makra użytkownika. Domyślnie: 10051. Zachowanie właściwości: - *obsługiwane*, jeśli tryb działania Zabbix proxy jest pasywny
allowed_addresses	string	Rozdzielone przecinkami adresy IP lub nazwy DNS aktywnego Zabbix proxy.
tls_connect	integer	Połączenia do proxy. Możliwe wartości: 1 - *(domyślnie)* Bez szyfrowania; 2 - PSK; 4 - certyfikat.
tls_accept	integer	Połączenia z proxy. Możliwe wartości mapy bitowej: 1 - *(domyślnie)* Bez szyfrowania; 2 - PSK; 4 - certyfikat. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości mapy bitowej (na przykład 6 dla PSK i certyfikatu).

Property	Type	Description
tls_issuer	string	Wystawca certyfikatu.
tls_subject	string	Temat certyfikatu.
tls_psk_identity	string	Tożsamość PSK; musi być sparowana tylko z jednym PSK (w obrębie autoregistration, hosts i proxies). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` jest ustawione na ”PSK” lub `tls_accept` zawiera bit ”PSK”
tls_psk	string	Klucz współdzielony (PSK); musi mieć co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` jest ustawione na ”PSK” lub `tls_accept` zawiera bit ”PSK”
custom_timeouts	integer	Czy zastąpić globalne limity czasu pozycji na poziomie proxy. Możliwe wartości: 0 - *(domyślnie)* użyj ustawień globalnych; 1 - zastąp limity czasu.
timeout_zabbix_agent	string	Na przetwarzanie poświęcaj nie więcej niż `timeout_*` sekund. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m). Akceptuje również makra użytkownika. Możliwy zakres wartości: 1-600s. Domyślnie: ””. Zachowanie właściwości: - *wymagane*, jeśli `custom_timeouts` jest ustawione na 1.
timeout_simple_check		
timeout_snmp_agent		
timeout_external_check		
timeout_db_monitor		
timeout_http_agent		
timeout_ssh_agent		
timeout_telnet_agent		
timeout_script		
timeout_browser		
version	integer	Wersja proxy. Trzyczęściowy numer wersji Zabbix, gdzie dla każdej części używane są dwie cyfry dziesiętne, np. 60401 dla wersji 6.4.1, 70002 dla wersji 7.0.2 itd. 0 - Nieznana wersja proxy. Zachowanie właściwości: - *tylko do odczytu*

Property	Type	Description
compatibility	integer	Wersja proxy w porównaniu z wersją Zabbix serwer. Możliwe wartości: 0 - Niezdefiniowana; 1 - Bieżąca wersja (proxy i serwer mają tę samą wersję główną); 2 - Nieaktualna wersja (wersja proxy jest starsza niż wersja serwer, ale jest częściowo obsługiwana); 3 - Nieobsługiwana wersja (wersja proxy jest starsza niż wersja poprzedniego wydania LTS serwer lub wersja główna serwer jest starsza niż wersja główna proxy). Zachowanie właściwości: - *tylko do odczytu*
state	integer	Stan proxy. Możliwe wartości: 0 - Nieznany; 1 - Offline; 2 - Online. Zachowanie właściwości: - *tylko do odczytu*

proxy.create

Opis

`object proxy.create(object/array proxies)`

Ta metoda umożliwia tworzenie nowych proxy.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w Rolach użytkowników.

Parametry

`(object/array)` Proxy do utworzenia.

Oprócz standardowych właściwości proxy, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
hosts	array	Hosty, które mają być monitorowane przez proxy. Jeśli host jest już monitorowany przez inny proxy, zostanie przypisany ponownie do bieżącego proxy. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych proxy we właściwości `proxyids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych proxy.

Przykłady

Utwórz aktywny proxy

Utwórz aktywny proxy „Active proxy” i przypisz do niego host do monitorowania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
            {
                "hostid": "10279"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10280"
        ]
    },
    "id": 1
}
```

Utwórz pasywny proxy

Utwórz pasywny proxy „Passive proxy” i przypisz do niego dwa hosty do monitorowania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Passive proxy",
        "operating_mode": "1",
        "address": "127.0.0.1",
        "port": "10051",
        "hosts": [
            {
                "hostid": "10192"
            },
            {
                "hostid": "10139"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10284"
        ]
    },
    "id": 1
}
```

Tworzenie proxy i dodawanie go do grupy proxy

Utwórz aktywny proxy „Active proxy” i dodaj go do grupy proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "proxy_groupid": "1",
        "operating_mode": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Tworzenie proxy z szyfrowaniem opartym na certyfikatach

Utwórz aktywny proxy z szyfrowaniem opartym na certyfikatach. Proxy jest skonfigurowany do monitorowania hosta „10785”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
            {
                "hostid": "10785"
            }
        ],
        "tls_accept": "4",
        "tls_issuer": "CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com",
        "tls_subject": "CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::create() w *ui/include/classes/api/services/CProxy.php*.

proxy.delete

Opis

`object proxy.delete(array proxies)`

Ta metoda umożliwia usuwanie proxy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID proxy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych proxy we właściwości `proxyids`.

Przykłady

Usuń wiele proxy

Usuń dwa proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.delete",
    "params": [
        "10286",
        "10285"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10286",
            "10285"
        ]
    },
    "id": 1
}
```

Źródło

CProxy::delete() w *ui/include/classes/api/services/CProxy.php*.

proxy.get

Opis

`integer/array proxy.get(object parameters)`

Ta metoda umożliwia pobieranie proxy zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
proxyids	ID/array	Zwracaj tylko proxy o podanych identyfikatorach.
proxy_groupids	ID/array	Zwracaj tylko proxy należące do podanych grup proxy.
selectAssignedHosts	query	Zwracaj właściwość `assignedHosts` z hostami przypisanymi do proxy. Obsługuje `count`.
selectHosts	query	Zwracaj właściwość `hosts` z hostami monitorowanymi przez proxy. Obsługuje `count`.
selectProxyGroup	query	Zwracaj właściwość `proxyGroup` z obiektem grupy proxy.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `proxyid`, `name`, `operating_mode`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich proxy

Pobierz wszystkie skonfigurowane proxy i ich interfejsy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxyid": "11",
            "name": "Active proxy",
            "proxy_groupid": "0",
            "local_address": "",
            "local_port": "10051",
```

```
            "operating_mode": "0",
            "description": "",
            "allowed_addresses": "",
            "address": "127.0.0.1",
            "port": "10051",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "custom_timeouts": "0",
            "timeout_zabbix_agent": "",
            "timeout_simple_check": "",
            "timeout_snmp_agent": "",
            "timeout_external_check": "",
            "timeout_db_monitor": "",
            "timeout_http_agent": "",
            "timeout_ssh_agent": "",
            "timeout_telnet_agent": "",
            "timeout_script": "",
            "last_access": "1693391880",
            "version": "70000",
            "compatibility": "1",
            "state": "1"
        },
        {
            "proxyid": "12",
            "name": "Passive proxy",
            "proxy_groupid": "1",
            "local_address": "127.0.0.1",
            "local_port": "10051",
            "operating_mode": "1",
            "description": "",
            "allowed_addresses": "",
            "address": "127.0.0.1",
            "port": "10051",
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "custom_timeouts": "1",
            "timeout_zabbix_agent": "5s",
            "timeout_simple_check": "5s",
            "timeout_snmp_agent": "5s",
            "timeout_external_check": "5s",
            "timeout_db_monitor": "5s",
            "timeout_http_agent": "5s",
            "timeout_ssh_agent": "5s",
            "timeout_telnet_agent": "5s",
            "timeout_script": "5s",
            "lastaccess": "1693391875",
            "version": "60400",
            "compatibility": "2",
            "state": "2"
        }
    ],
    "id": 1
}
```


Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::get() w *ui/include/classes/api/services/CProxy.php*.

proxy.update

Opis

`object proxy.update(object/array proxies)`

Ta metoda umożliwia aktualizację istniejących proxy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w Rolach użytkowników.

Parametry

`(object/array)` Właściwości proxy do zaktualizowania.

Właściwość `proxyid` musi być zdefiniowana dla każdego proxy, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości proxy metoda akceptuje następujące parametry.

Parametr	Typ	Opis
hosts	array	Hosty do monitorowania przez proxy. Jeśli host jest już monitorowany przez inny proxy, zostanie ponownie przypisany do bieżącego proxy. Dla hostów musi być zdefiniowana tylko właściwość `hostid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych proxy we właściwości `proxyids`.

Przykłady

Zmień hosty monitorowane przez proxy

Zaktualizuj proxy, aby monitorował dwa podane hosty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "hosts": [
            {
                "hostid": "10294"
            },
            {
                "hostid": "10295"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
```

```
    },
    "id": 1
}
```

Zmień status proxy

Zmień proxy na aktywny proxy i zmień jego nazwę na „Active proxy”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "name": "Active proxy",
        "operating_mode": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Dodawanie proxy do grupy proxy

Zaktualizuj proxy o ID „5” i dodaj go do grupy proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "proxyid": "5",
        "proxy_groupid": "1",
        "local_address": "127.0.0.1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::update() w *ui/include/classes/api/services/CProxy.php*.

Pulpit

Ta klasa jest przeznaczona do pracy z pulpitami.

Odwołania do obiektów:

- Pulpit
- Strona pulpitu
  - Widżet pulpitu
    - Pole widżetu pulpitu
- Grupa użytkowników pulpitu
- Użytkownik pulpitu

Dostępne metody:

- dashboard.create - tworzenie nowych pulpitów
- dashboard.delete - usuwanie pulpitów
- dashboard.get - pobieranie pulpitów
- dashboard.update - aktualizowanie pulpitów

Obiekt Pulpitu Nawigacyjnego

Następujące obiekty są bezpośrednio związane z API `dashboard`.

Pulpit

Obiekt pulpitu ma następujące właściwości.

Właściwość	Typ	Opis
dashboardid	ID	ID pulpitu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa pulpitu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
userid	ID	ID użytkownika będącego właścicielem pulpitu.
private	integer	Typ udostępniania pulpitu. Możliwe wartości: 0 - pulpit publiczny; 1 - *(domyślnie)* pulpit prywatny.
display_period	integer	Domyślny okres wyświetlania strony (w sekundach). Możliwe wartości: 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 30.
auto_start	integer	Automatyczne uruchamianie pokazu slajdów. Możliwe wartości: 0 - nie uruchamiaj automatycznie pokazu slajdów; 1 - *(domyślnie)* automatycznie uruchamiaj pokaz slajdów.

Strona pulpitu

Obiekt strony pulpitu ma następujące właściwości.

Właściwość	Typ	Opis
dashboard_pageid	ID	ID strony pulpitu. Zachowanie właściwości: - *tylko do odczytu*
name	string	Nazwa strony pulpitu. Domyślnie: pusty ciąg znaków.
display_period	integer	Okres wyświetlania strony pulpitu (w sekundach). Możliwe wartości: 0, 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 0 (zostanie użyty domyślny okres wyświetlania strony).
widgets	array	Tablica obiektów widgetu pulpitu.

Widget dashboardowy

Obiekt widgetu dashboardu ma następujące właściwości.

Property	Type	Description
widgetid	ID	ID widgetu dashboardu. Property behavior: - *read-only*

Property	Type	Description
type	string	Typ widgetu dashboardu. Possible values: actionlog - Dziennik akcji; clock - Zegar; discovery - Status wykrywania; favgraphs - Ulubione wykresy; favmaps - Ulubione mapy; gauge - Miernik; geomap - Geomap; graph - Wykres (klasyczny); graphprototype - Prototyp wykresu; honeycomb - Honeycomb; hostavail - Dostępność hosta; hostcard - Karta hosta; hostnavigator - Nawigator hostów; itemcard - Karta pozycji; itemhistory - Historia pozycji; itemnavigator - Nawigator pozycji; item - Wartość pozycji; map - Mapa; navtree - Drzewo nawigacji mapy; piechart - Wykres kołowy; problemhosts - Hosty z problemami; problems - Problemy; problemsbysv - Problemy według ważności; slareport - Raport SLA; svggraph - Wykres; systeminfo - Informacje o systemie; tophosts - Najlepsze hosty; topitems - Najlepsze pozycje; toptriggers - Najlepsze wyzwalacze; trigover - Przegląd wyzwalaczy; url - URL; web - Monitorowanie WWW. Property behavior: - *required*
name	string	Niestandardowa nazwa widgetu.
x	integer	Pozycja pozioma od lewej strony dashboardu. Possible values range from 0 to 71.
y	integer	Pozycja pionowa od górnej strony dashboardu. Possible values range from 0 to 63.
width	integer	Szerokość widgetu. Possible values range from 1 to 72.
height	integer	Wysokość widgetu. Possible values range from 1 to 64.
view_mode	integer	Tryb widoku widgetu. Possible values: 0 - *(default)* domyślny widok widgetu; 1 - z ukrytym nagłówkiem;
fields	array	Tablica obiektów dashboard widget field. Property behavior: - zobacz poszczególne widgety w Dashboard widget fields

Pole widżetu pulpitu

Obiekt pola widżetu pulpitu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ pola widżetu. Możliwe wartości: 0 - Liczba całkowita; 1 - Ciąg znaków; 2 - Grupa hostów; 3 - Host; 4 - Pozycja; 5 - Prototyp pozycji; 6 - Wykres; 7 - Prototyp wykresu; 8 - Mapa; 9 - Usługa; 10 - SLA; 11 - Użytkownik; 12 - Akcja; 13 - Typ nośnika. Zachowanie właściwości: - *wymagane*
name	string	Nazwa pola widżetu. Możliwe wartości: patrz Pola widżetu pulpitu. Zachowanie właściwości: - *wymagane*
value	mixed	Wartość pola widżetu zależna od typu. Możliwe wartości: patrz Pola widżetu pulpitu. Zachowanie właściwości: - *wymagane*

Grupa użytkowników pulpitu

Lista uprawnień pulpitu oparta na grupach użytkowników. Ma następujące właściwości.

Właściwość	Typ	Opis
usrgrpid	ID	ID grupy użytkowników. Zachowanie właściwości: - *wymagane*
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

Użytkownik pulpitu

Lista uprawnień pulpitu na podstawie użytkowników. Ma następujące właściwości.

Właściwość	Typ	Opis
userid	ID	ID użytkownika. Zachowanie właściwości: - *wymagane*
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

dashboard.create

Opis

`object dashboard.create(object/array dashboards)`

Ta metoda umożliwia tworzenie nowych dashboardów.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Pulpity do utworzenia.

Oprócz standardowych właściwości pulpitu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony pulpitu, które mają zostać utworzone dla pulpitu. Strony pulpitu będą uporządkowane w tej samej kolejności, w jakiej zostały podane. Zachowanie parametrów: - *wymagane*
users	array	Udziały użytkownika pulpitu, które mają zostać utworzone na pulpicie.
userGroups	array	Udziały grupy użytkowników pulpitu, które mają zostać utworzone na pulpicie.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pulpitów we właściwości `dashboardids`. Kolejność zwróconych ID jest zgodna z kojenością przekazanych pulpitów.

Przykłady

Tworzenie pulpitu

Utwórz pulpit o nazwie „My dashboard” z jednym widżetem Problemy z tagami oraz z użyciem dwóch typów udostępniania (grupa użytkowników i użytkownik) na jednej stronie pulpitu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
```

```
            {
                "widgets": [
                    {
                        "type": "problems",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "service"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "zabbix_server"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": "7",
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu

- Grupa użytkowników pulpitu

Źródło

CDashboard::create() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.delete

Opis

`object dashboard.delete(array dashboardids)`

Ta metoda umożliwia usuwanie pulpitów.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` identyfikatory pulpitów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pulpitów we właściwości `dashboardids`.

Przykłady

Usuwanie wielu pulpitów

Usuń dwa pulpity.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "dashboard.delete",
"params": [
"2",
"3"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"2",
"3"
]
},
"id": 1
}
```

Źródło

CDashboard::delete() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.get

Opis

`integer/array dashboard.get(object parameters)`

Ta metoda umożliwia pobieranie pulpitów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dashboardids	ID/array	Zwraca tylko pulpity z podanymi ID.
selectPages	query	Zwraca właściwość `pages` z prawidłowo uporządkowanymi stronami pulpitu.
selectUsers	query	Zwraca właściwość `users` z użytkownikami, którym udostępniono pulpit.
selectUserGroups	query	Zwraca właściwość `userGroups` z grupami użytkowników, którym udostępniono pulpit.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dashboardid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablica obiektów;
- ilość pobranych obiektów, jeżeli parametr `countOutput` został użyty.

Przykłady

Pobieranie pulpitu nawigacyjnego według ID

Pobierz wszystkie dane o pulpitach nawigacyjnych „1” i „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "dashboardids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dashboardid": "1",
            "name": "Dashboard",
            "userid": "1",
            "private": "0",
            "display_period": "30",
            "auto_start": "1",
            "users": [],
            "userGroups": [],
            "pages": [
                {
                    "dashboard_pageid": "1",
                    "name": "",
                    "display_period": "0",
                    "widgets": [
                        {
                            "widgetid": "9",
                            "type": "systeminfo",
                            "name": "",
                            "x": "12",
                            "y": "8",
                            "width": "12",
                            "height": "5",
                            "view_mode": "0",
                            "fields": []
                        },
                        {
                            "widgetid": "8",
                            "type": "problemsbysv",
                            "name": "",
                            "x": "12",
                            "y": "4",
                            "width": "12",
                            "height": "4",
                            "view_mode": "0",
                            "fields": []
                        },
                        {
                            "widgetid": "7",
                            "type": "problemhosts",
                            "name": "",
                            "x": "12",
                            "y": "0",
                            "width": "12",
                            "height": "4",
                            "view_mode": "0",
                            "fields": []
                        },
                        {
                            "widgetid": "6",
                            "type": "discovery",
                            "name": "",
                            "x": "6",
                            "y": "9",
                            "width": "18",
                            "height": "4",
                            "view_mode": "0",
                            "fields": []
                        },
```

```
                    {
                        "widgetid": "5",
                        "type": "web",
                        "name": "",
                        "x": "0",
                        "y": "9",
                        "width": "18",
                        "height": "4",
                        "view_mode": "0",
                        "fields": []
                    },
                    {
                        "widgetid": "4",
                        "type": "problems",
                        "name": "",
                        "x": "0",
                        "y": "3",
                        "width": "12",
                        "height": "6",
                        "view_mode": "0",
                        "fields": []
                    },
                    {
                        "widgetid": "3",
                        "type": "favmaps",
                        "name": "",
                        "x": "8",
                        "y": "0",
                        "width": "12",
                        "height": "3",
                        "view_mode": "0",
                        "fields": []
                    },
                    {
                        "widgetid": "1",
                        "type": "favgraphs",
                        "name": "",
                        "x": "0",
                        "y": "0",
                        "width": "12",
                        "height": "3",
                        "view_mode": "0",
                        "fields": []
                    }
                ]
            },
            {
                "dashboard_pageid": "2",
                "name": "",
                "display_period": "0",
                "widgets": []
            },
            {
                "dashboard_pageid": "3",
                "name": "Custom page name",
                "display_period": "60",
                "widgets": []
            }
        ]
    },
    {
        "dashboardid": "2",
```

```
            "name": "My dashboard",
            "userid": "1",
            "private": "1",
            "display_period": "60",
            "auto_start": "1",
            "users": [
                {
                    "userid": "4",
                    "permission": "3"
                }
            ],
            "userGroups": [
                {
                    "usrgrpid": "7",
                    "permission": "2"
                }
            ],
            "pages": [
                {
                    "dashboard_pageid": "4",
                    "name": "",
                    "display_period": "0",
                    "widgets": [
                        {
                            "widgetid": "10",
                            "type": "problems",
                            "name": "",
                            "x": "0",
                            "y": "0",
                            "width": "12",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "2",
                                    "name": "groupids",
                                    "value": "4"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```


Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu
- Grupa użytkowników pulpitu

Źródło

CDashboard::get() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.update

Opis

`object dashboard.update(object/array dashboards)`

Ta metoda umożliwia aktualizację istniejących pulpitów.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w User roles.

Parametry

`(object/array)` Właściwości dashboardu do zaktualizowania.

Właściwość `dashboardid` musi być zdefiniowana dla każdego dashboardu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości dashboardu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony dashboardu, które zastąpią istniejące strony dashboardu. Strony dashboardu są aktualizowane za pomocą właściwości `dashboard_pageid`. Nowe strony dashboardu zostaną utworzone dla obiektów bez właściwości `dashboard_pageid`, a istniejące strony dashboardu zostaną usunięte, jeśli nie zostaną ponownie użyte. Strony dashboardu zostaną uporządkowane w tej samej kolejności, w jakiej zostały podane. Zaktualizowane zostaną tylko określone właściwości stron dashboardu.
users	array	Udostępnienia użytkownikom dashboardu, które zastąpią istniejące elementy.
userGroups	array	Udostępnienia grupom użytkowników dashboardu, które zastąpią istniejące elementy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych pulpitów pod właściwością `dashboardids`.

Przykłady

Zmienianie nazwy pulpitu

Zmień nazwę pulpitu na „SQL server status”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "name": "SQL server status"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Aktualizowanie stron pulpitu

Zmień nazwę pierwszej strony pulpitu, zastąp widgety na drugiej stronie pulpitu i dodaj nową stronę jako trzecią. Usuń wszystkie pozostałe strony pulpitu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "pages": [
            {
                "dashboard_pageid": 1,
                "name": "Strona o zmienionej nazwie"
            },
            {
                "dashboard_pageid": 2,
                "widgets": [
                    {
                        "type": "clock",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zmień właściciela pulpitu

Dostępne tylko dla administratorów i super administratorów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "userid": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu
- Grupa użytkowników pulpitu

Źródło

CDashboard::update() w *ui/include/classes/api/services/CDashboard.php*.

Pola widżetu pulpitu

Ta strona zawiera linki nawigacyjne do parametrów widżetów dashboardu oraz możliwych wartości właściwości dla odpowiednich obiektów dashboard widget field.

Aby zobaczyć parametry i wartości właściwości dla każdego widżetu, przejdź do indywidualnych stron widżetów dla:

- Action log
- Clock
- Discovery status
- Favorite graphs
- Favorite maps
- Gauge
- Geomap
- Graph
- Graph (classic)
- Graph prototype
- Honeycomb
- Host availability
- Host card
- Host navigator
- Item card
- Item history
- Item navigator
- Item value
- Map
- Map navigation tree
- Pie chart
- Problem hosts
- Problems
- SLA report
- System information
- Problems by severity
- Top hosts
- Top items
- Top triggers
- Trigger overview
- URL
- Web monitoring

> **Attention:**
> Przestarzałe widżety zostaną usunięte w nadchodzącym głównym wydaniu.

1 Log akcji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Action log* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć niestandardowe widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Action log*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Action log*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Recipients	11	userids.0	ID użytkownika. Uwaga: Aby skonfigurować wielu użytkowników, utwórz obiekt pola widżetu pulpitu dla każdego użytkownika, zwiększając numer w nazwie właściwości.
Actions	12	actionids.0	ID akcji. Uwaga: Aby skonfigurować wiele akcji, utwórz obiekt pola widżetu pulpitu dla każdej akcji, zwiększając numer w nazwie właściwości.
Media types	13	mediatypeids.0	ID typu mediów. Uwaga: Aby skonfigurować wiele typów mediów, utwórz obiekt pola widżetu pulpitu dla każdego typu mediów, zwiększając numer w nazwie właściwości.
Status	0	statuses.0	0 - W trakcie; 1 - Wysłane/Wykonane; 2 - Niepowodzenie. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości, zwiększając numer w nazwie właściwości.
Search string	1	message	Dowolna wartość tekstowa.
Time period	1	time_period._referenceDASHBOARD._timeperiod	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli ustawiono `time_period.to`

Parameter	type	name	value
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli ustawiono `time_period.from`
Sort entries by	0	sort_triggers	3 - Czas (rosnąco); 4 - *(domyślnie)* Czas (malejąco); 5 - Typ (rosnąco); 6 - Typ (malejąco); 7 - Status (rosnąco); 8 - Status (malejąco); 11 - Odbiorca (rosnąco); 12 - Odbiorca (malejąco).
Show lines	0	show_lines	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 25.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Zegara*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz `dashboard.create`.

Konfigurowanie widżetu *Dziennik działań*

Skonfiguruj widżet *Dziennik działań*, który wyświetla 10 wpisów ze szczegółami operacji działania, posortowanych według czasu (rosnąco). Dodatkowo wyświetlaj szczegóły tylko dla tych operacji działania, które próbowały wysłać wiadomość e-mail do użytkownika „1”, ale zakończyły się niepowodzeniem.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "show_lines",
                                "value": 10
                            },
                            {
                                "type": 0,
                                "name": "sort_triggers",
                                "value": 3
                            },
                            {
                                "type": 11,
                                "name": "userids.0",
```

```
                                "value": 1
                            },
                            {
                                "type": 13,
                                "name": "mediatypeids.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "statuses.0",
                                "value": 2
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

2 Zegar

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu umożliwiają skonfigurowanie widżetu *Clock* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widżetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widżety i tworzyć własne widżety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widżetów. Aby zapewnić pomyślne utworzenie lub aktualizację widżetu *Clock*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Clock*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Typ czasu	0	time_type	0 - *(domyślnie)* Czas lokalny; 1 - Czas serwera; 2 - Czas hosta.
Typ zegara	0	clock_type	0 - *(domyślnie)* Analogowy; 1 - Cyfrowy.

Następujące parametry są obsługiwane, jeśli *Typ czasu* jest ustawiony na „Czas hosta”.

Parametr	type	name	value
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametru: - *wymagane*

Następujące parametry są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”.

Parametr	type	name	value
Pokaż	0	show.0	1 - Data; 2 - *(domyślnie)* Czas; 3 - Strefa czasowa. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z inkrementowanym numerem w nazwie właściwości.

Konfiguracja zaawansowana

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”.

Parametr	type	name	value
Kolor tła	1	bg_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Data

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Clock type* jest ustawiony na „Digital”, a *Show* jest ustawione na „Date”.

Parametr	type	name	value
Pogrubienie	0	date_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	date_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Czas

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”, a *Pokaż* jest ustawiony na „Czas”.

Parametr	typ	nazwa	wartość
Pogrubienie	0	time_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	time_color	Kod koloru szesnastkowego (np. `FF0000`). Domyślnie: `""` (puste).
Sekundy	0	time_sec	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Format	0	time_format	0 - *(domyślny)* 24-godzinny; 1 - 12-godzinny.

Strefa czasowa

Następujące zaawansowane parametry konfiguracyjne są obsługiwane, jeśli *Clock type* jest ustawiony na „Digital”, a *Show* jest ustawione na „Time zone”.

Parametr	type	name	value
Pogrubienie	0	tzone_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	tzone_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Strefa czasowa	1	tzone_timezone	Prawidłowy ciąg strefy czasowej (np. `Europe/Riga`, `system`, `UTC` itd.). Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP. Domyślnie: `local`. Zachowanie parametru: - *obsługiwany*, jeśli *Time type* jest ustawiony na „Local time” lub „Server time”
Format	0	tzone_format	0 - *(domyślnie)* Krótki; 1 - Pełny. Zachowanie parametru: - *obsługiwany*, jeśli *Time type* jest ustawiony na „Local time” lub „Server time”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Zegara*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz `dashboard.create`.

Konfigurowanie widżetu *Clock*

Skonfiguruj widżet *Clock*, który wyświetla lokalną datę, godzinę i strefę czasową w dostosowanym zegarze cyfrowym.

Żądanie:

```
{
    "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": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "clock_type",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.1",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "show.2",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "date_color",
                                "value": "E1E1E1"
                            },
                            {
                                "type": 0,
                                "name": "time_bold",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "tzone_color",
                                "value": "E1E1E1"
                            },
                            {
                                "type": 1,
                                "name": "tzone_timezone",
                                "value": "Europe/Riga"
                            },
                            {
                                "type": 0,
                                "name": "tzone_format",
```

```
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

3 Stan wykrywania

Opis

Te parametry i możliwe wartości właściwości dla odpowiednich obiektów pól widżetów dashboard pozwalają skonfigurować widżet *Discovery status* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Status wykrywania*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Status wykrywania*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Discovery status*

Skonfiguruj widżet *Discovery status* z interwałem odświeżania ustawionym na 15 minut.

Żądanie:

```
{
    "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": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 900
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- dashboard.create
- dashboard.update

4 Ulubione wykresy

Opis

Te parametry i możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboard pozwalają skonfigurować widget *Favorite graphs* w metodach dashboard.create i dashboard.update.

Parametry

Następujące parametry są obsługiwane dla widżetu *Favorite graphs*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Ulubione wykresy*. Więcej informacji na temat konfigurowania pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Ulubione wykresy*

Skonfiguruj widżet *Ulubione wykresy* z interwałem odświeżania ustawionym na 10 minut.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mój pulpit",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "favgraphs",
                        "name": "Ulubione wykresy",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            }
                        ]
                    }
                ]
            }
```

```
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

5 Ulubione mapy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu pulpitu umożliwiają skonfigurowanie widżetu *Ulubione mapy* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Ulubione mapy*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Ulubione mapy*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Ulubione mapy*

Skonfiguruj widżet *Ulubione mapy* z interwałem odświeżania ustawionym na 10 minut.

Żądanie:

```
{
    "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": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

6 Wskaźnik

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Gauge* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Gauge*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Gauge*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Item	4	itemid.0	ID pozycja. Zachowanie parametru: - *wymagany*, jeśli *Item (Widget)* nie jest ustawiony
Item (Widget)	1	itemid._reference	Zamiast ID pozycja: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla pozycji. Zachowanie parametru: - *wymagany*, jeśli *Item* nie jest ustawiony
Min	1	min	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. "1d", "2w", "4K", "8G"). Domyślnie: "0".
Max	1	max	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. "1d", "2w", "4K", "8G"). Domyślnie: "100".
Value arc	1	value_arc_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Arc background	1	empty_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Background	1	bg_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Parameter	type	name	value
Show	0	show.0	1 - Opis; 2 - Wartość; 3 - Wskaźnik; 4 - Skala; 5 - Łuk wartości. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości z numerem zwiększanym w nazwie właściwości. Domyślnie: 1, 2, 4, 5. Wartości ”Needle” i ”Scale” nie są obsługiwane, jeśli oba warunki są spełnione: - nie jest ustawiony żaden obiekt pola widżetu dashboardu dla *Show* z wartością ”Value arc”; - parametr *Show arc* zaawansowanej konfiguracji jest ustawiony na ”Disabled”. Parametry zaawansowanej konfiguracji dla opcji *Show* nie są obsługiwane, jeśli nie są ustawione żadne obiekty pól widżetu dashboardu z odpowiednimi wartościami.
Override host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor *Host* dashboardu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.

Konfiguracja zaawansowana

Poniższe zaawansowane parametry konfiguracji są obsługiwane przez widżet *Gauge*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odnosi się do miejsca progu na liście posortowanej rosnąco. Jeśli jednak progi zostaną skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontendzie Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parametr	type	name	value
Kąt	0	angle	Możliwe wartości: 180 *(domyślnie)* lub 270.
Opis			
Opis	1	description	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {ITEM.*}, {INVENTORY.*}, makra użytkownika. Domyślnie: {ITEM.NAME}.
Rozmiar	0	desc_size	Możliwe wartości z zakresu 1-100. Domyślnie: 15.
Pozycja pionowa	0	desc_v_pos	0 - Góra; 1 - *(domyślnie)* Dół.
Pogrubienie	0	desc_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	desc_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Wartość			

Parametr	type	name	value
Miejsca dziesiętne	0	decimal_places	Możliwe wartości z zakresu 1-10. Domyślnie: 2.
Rozmiar	0	value_size	Możliwe wartości z zakresu 1-100. Domyślnie: 25.
Pogrubienie	0	value_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	value_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Jednostki			
Jednostki (pole wyboru)	0	units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Jednostki (wartość)	1	units	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone”
Rozmiar	0	units_size	Możliwe wartości z zakresu 1-100. Domyślnie: 25. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone”
Pogrubienie	0	units_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone”
Pozycja	0	units_pos	0 - Przed wartością; 1 - Nad wartością; 2 - *(domyślnie)* Za wartością; 3 - Pod wartością. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone” Ten parametr jest ignorowany, jeśli jest ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Kolor	1	units_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Łuk wartości			
Rozmiar łuku	0	value_arc_size	Możliwe wartości z zakresu 1-100. Domyślnie: 20.
Wskazówka			
Kolor	1	needle_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste). Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu pulpitu dla *Pokaż* z wartością ”Value arc” lub *Pokaż łuk* jest ustawione na ”Włączone”
Skala			

Parametr	type	name	value
Pokaż jednostki	0	scale_show_units	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone” oraz ustawiono obiekt pola widżetu pulpitu dla *Pokaż* z wartością ”Value arc” lub *Pokaż łuk* jest ustawione na ”Włączone”
Rozmiar	0	scale_size	Możliwe wartości z zakresu 1-100. Domyślnie: 15. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu pulpitu dla *Pokaż* z wartością ”Value arc” lub *Pokaż łuk* jest ustawione na ”Włączone”
Miejsca dziesiętne	0	scale_decimal_places	Możliwe wartości z zakresu 1-10. Domyślnie: 0. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu pulpitu dla *Pokaż* z wartością ”Value arc” lub *Pokaż łuk* jest ustawione na ”Włączone”
Progi			
Kolor	1	thresholds.0.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	thresholds.0.threshold	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”).
Pokaż etykiety	0	th_show_labels	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Progi* są ustawione oraz ustawiono obiekt pola widżetu pulpitu dla *Pokaż* z wartością ”Value arc” lub *Pokaż łuk* jest ustawione na ”Włączone”
Pokaż łuk	0	th_show_arc	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Progi* są ustawione
Rozmiar łuku	0	th_arc_size	Możliwe wartości z zakresu 1-100. Domyślnie: 5. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż łuk* jest ustawione na ”Włączone”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Gauge*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Gauge*

Skonfiguruj widżet *Gauge*, który wyświetla wartość pozycji dla pozycji „44474” (Interface enp0s3: Bits sent). Dodatkowo dostrój wizualnie widżet za pomocą wielu zaawansowanych opcji, w tym progów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
```

```
        "pages": [
            {
                "widgets": [
                    {
                        "type": "gauge",
                        "name": "Gauge",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 44474
                            },
                            {
                                "type": 1,
                                "name": "min",
                                "value": "100000"
                            },
                            {
                                "type": 1,
                                "name": "max",
                                "value": "1000000"
                            },
                            {
                                "type": 0,
                                "name": "show.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.1",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "show.2",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "show.4",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "show.5",
                                "value": 5
                            },
                            {
                                "type": 0,
                                "name": "angle",
                                "value": 270
                            },
                            {
                                "type": 0,
                                "name": "desc_size",
                                "value": 10
                            },
```

```
                {
                    "type": 0,
                    "name": "desc_bold",
                    "value": 1
                },
                {
                    "type": 0,
                    "name": "decimal_places",
                    "value": 0
                },
                {
                    "type": 0,
                    "name": "value_bold",
                    "value": 1
                },
                {
                    "type": 0,
                    "name": "units_size",
                    "value": 15
                },
                {
                    "type": 0,
                    "name": "units_pos",
                    "value": 3
                },
                {
                    "type": 1,
                    "name": "needle_color",
                    "value": "3C3C3C"
                },
                {
                    "type": 1,
                    "name": "thresholds.0.color",
                    "value": "FF465C"
                },
                {
                    "type": 1,
                    "name": "thresholds.0.threshold",
                    "value": "700000"
                },
                {
                    "type": 1,
                    "name": "thresholds.1.color",
                    "value": "FFD54F"
                },
                {
                    "type": 1,
                    "name": "thresholds.1.threshold",
                    "value": "500000"
                },
                {
                    "type": 1,
                    "name": "thresholds.2.color",
                    "value": "0EC9AC"
                },
                {
                    "type": 1,
                    "name": "thresholds.2.threshold",
                    "value": "100000"
                },
                {
                    "type": 0,
```

```
                                "name": "th_show_labels",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "th_show_arc",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "th_arc_size",
                                "value": 15
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

7 Mapa geograficzna

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Geomap* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wiąże się z ryzykiem nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Geomap*, należy zapoznać się z działaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Geomap*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.

Parameter	type	name	value
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Initial view	1	default_view	Wartości *latitude*, *longitude*, *zoom level* rozdzielone przecinkami (*opcjonalnie*, możliwy zakres wartości od 0 do 30). Przykład: `40.6892494,-74.0466891,10`.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Geomap*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Geomap*

Skonfiguruj widżet *Geomap*, który wyświetla hosty z grup hostów „2” i „22” na podstawie następującej konfiguracji tagów: tag o nazwie „component” zawiera wartość „node”, a tag o nazwie „location” jest równy wartości „New York”. Dodatkowo ustaw początkowy widok mapy na współrzędne „40.6892494” (szerokość geograficzna), „-74.0466891” (długość geograficzna) z poziomem powiększenia „10”.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 22
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "default_view",
                                "value": "40.6892494,-74.0466891,10"
                            },
                            {
                                "type": 0,
                                "name": "evaltype",
                                "value": 2
                            },
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "node"
                            },
                            {
                                "type": 1,
                                "name": "tags.1.tag",
                                "value": "location"
                            },
                            {
                                "type": 0,
                                "name": "tags.1.operator",
                                "value": 1
                            },
                            {
                                "type": 1,
```

```
                                "name": "tags.1.value",
                                "value": "New York"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

8 Graph

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Graph* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Dla widżetu *Graph* obsługiwane są następujące parametry.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Zestaw danych

Poniższe parametry są obsługiwane podczas konfigurowania *Zestawu danych*.

Note:
Pierwsza liczba w nazwie właściwości (np. ds.0.hosts.0, ds.0.items.0) oznacza konkretny zestaw danych, natomiast druga liczba, jeśli występuje, oznacza skonfigurowany host lub pozycja.

Parameter	type	name	value
Typ zestawu danych	0	ds.0.dataset_type	0 - Lista pozycji; 1 - *(domyślnie)* Wzorce pozycji.
Pozycje	4	ds.0.itemids.0	ID pozycja. Podczas konfigurowania widżetu na dashboardzie szablonu należy ustawiać tylko pozycje skonfigurowane w szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widżetu dashboardu dla każdej pozycji, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagane* jeśli *Typ zestawu danych* jest ustawiony na "Lista pozycji", a *Pozycje (Widget)* nie są ustawione
Pozycje (Widget)	1	ds.0.itemids.0._reference	Zamiast ID pozycja: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widżetów, utwórz obiekt pola widżetu dashboardu dla każdego widżetu, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagane* jeśli *Typ zestawu danych* jest ustawiony na "Lista pozycji", a *Pozycje* nie są ustawione
Kolor	1	ds.0.color.0	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *wymagane* jeśli *Typ zestawu danych* jest ustawiony na "Lista pozycji"

Parameter	type	name	value
Wzorce hostów	1	ds.0.hosts.0	Nazwa lub wzorzec host (np. ”Zabbix*”). Zachowanie parametru: - *wymagane* jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji” Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Wzorce pozycji	1	ds.0.items.0	Nazwa lub wzorzec pozycja (np. ”*: Number of processed *values per second”). Podczas konfigurowania widżetu na dashboardzie szablonu należy ustawiać tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagane* jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”
Kolor	1	ds.0.color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwane* jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”, a *Paleta kolorów* nie jest ustawiona
Paleta kolorów	0	ds.0.color_palette	Indeks palety kolorów. Możliwe wartości mieszczą się w zakresie 0-11. Domyślnie: 0. Zachowanie parametru: - *obsługiwane* jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”, a *Kolor* nie jest ustawiony
Rysowanie	0	ds.0.type	0 - *(domyślnie)* Linia; 1 - Punkty; 2 - Schodkowy; 3 - Słupkowy.
Skumulowane	0	ds.0.stacked	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Szerokość	0	ds.0.width	Zachowanie parametru: - *obsługiwane* jeśli *Rysowanie* jest ustawione na ”Linia”, ”Schodkowy” lub ”Słupkowy” Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 1.
Rozmiar punktu	0	ds.0.pointsize	Zachowanie parametru: - *obsługiwane* jeśli *Rysowanie* jest ustawione na ”Linia” lub ”Schodkowy” Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 3.
Przezroczystość	0	ds.0.transparency	Zachowanie parametru: - *obsługiwane* jeśli *Rysowanie* jest ustawione na ”Punkty” Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 5.

Parameter	type	name	value
Wypełnienie	0	ds.0.fill	Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 3. Zachowanie parametru: - *obsługiwane* jeśli *Rysowanie* jest ustawione na ”Linia” lub ”Schodkowy”
Brakujące dane	0	ds.0.missingdatafunc	0 - *(domyślnie)* Brak; 1 - Połączone; 2 - Traktuj jako 0; 3 - Ostatnio znane. Zachowanie parametru: - *obsługiwane* jeśli *Rysowanie* jest ustawione na ”Linia” lub ”Schodkowy”
Zastąp host	1	ds.0.override_hostid	`ABCDE` - ustaw kompatybilny widżet (z parametrem ”Reference” ustawionym na ”`ABCDE`”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta dashboardu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Oś Y	0	ds.0.axisy	0 - *(domyślnie)* Lewa; 1 - Prawa.
Przesunięcie czasu	1	ds.0.timeshift	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Można używać sufiksów czasu. Dozwolone są również wartości ujemne. Domyślnie: "" (puste).
Funkcja agregacji	0	ds.0.aggregate_function	0 - *(domyślnie)* nieużywane; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Interwał agregacji	1	ds.0.aggregate_interval	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Można używać sufiksów czasu. Domyślnie: 1h.
Agreguj	0	ds.0.aggregate_grouping	0 - *(domyślnie)* Każda pozycja; 1 - Zestaw danych. Zachowanie parametru: - *obsługiwane* jeśli *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first” lub ”last”
Przybliżenie	0	ds.0.approximation	1 - min; 2 - *(domyślnie)* avg; 4 - max; 7 - all.
Etykieta zestawu danych	1	ds.0.data_set_label	Dowolna wartość tekstowa. Domyślnie: "" (puste).

Opcje wyświetlania

Następujące parametry są obsługiwane podczas konfigurowania *Opcji wyświetlania*.

Parameter	type	name	value
Wybór danych historycznych	0	source	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy.
Proste wyzwalacze	0	simple_triggers	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Czas pracy	0	working_time	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Linia percentyla (lewa)			
Status	0	percentile_left	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Oś Y* (w konfiguracji *Zestaw danych*) jest ustawiona na ”Lewa”
Wartość	0	percentile_left_value	Dopuszczalne wartości mieszczą się w zakresie 1-100. Zachowanie parametru: - *obsługiwane* jeśli *Oś Y* (w konfiguracji *Zestaw danych*) jest ustawiona na ”Lewa”
Linia percentyla (prawa)			
Status	0	percentile_right	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Oś Y* (w konfiguracji *Zestaw danych*) jest ustawiona na ”Prawa”
Wartość	0	percentile_right_value	Dopuszczalne wartości mieszczą się w zakresie 1-100. Zachowanie parametru: - *obsługiwane* jeśli *Oś Y* (w konfiguracji *Zestaw danych*) jest ustawiona na ”Prawa”

Okres czasu

Następujące parametry są obsługiwane podczas konfigurowania *Okresu czasu*.

Parameter	type	name	value
Okres czasu	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor *Okres czasu* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Okres czasu* nie jest ustawiony - *wymagany* jeśli `time_period.to` jest ustawiony

Parameter	type	name	value
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Okres czasu* nie jest ustawiony - *wymagany* jeśli `time_period.from` jest ustawiony

Osie

Następujące parametry są obsługiwane przy konfigurowaniu *Osi*.

Parametr	type	name	value
Lewe Y	0	lefty	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zbiór danych*) jest ustawiona na „Lewa”
Prawe Y	0	righty	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zbiór danych*) jest ustawiona na „Prawa”
Skala	0	lefty_scale	0 - *(domyślnie)* Liniowa; 1 - Logarytmiczna.
		righty_scale	
Min	1	lefty_min	Dowolna wartość liczbowa. Domyślnie: "" (puste).
		righty_min	
Max	1	lefty_max	Dowolna wartość liczbowa. Domyślnie: "" (puste).
		righty_max	
Jednostki (typ)	0	lefty_units	0 - *(domyślnie)* Auto; 1 - Statyczne.
		righty_units	
Jednostki (wartość)	1	lefty_static_units	Dowolna wartość tekstowa. Domyślnie: "" (puste).
		righty_static_units	
Oś X	0	xaxis	0 - Wyłączona; 1 - *(domyślnie)* Włączona.

Legenda

Następujące parametry są obsługiwane podczas konfigurowania *Legend*.

Parameter	type	name	value
Pokaż legendę	0	legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone”

Parameter	type	name	value
Wyświetl min./śr./maks.	0	legend_statistic	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Pokaż funkcję agregacji	0	legend_aggregation	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Wiersze	0	legend_lines_mode	0 - *(domyślnie)* Stałe; 1 - Zmienne. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Liczba wierszy/ Maksymalna liczba wierszy	0	legend_lines	Dostępne wartości mieszczą się w zakresie 1-10. Domyślnie: 1. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Liczba kolumn	0	legend_columns	Dostępne wartości mieszczą się w zakresie 1-4. Domyślnie: 4. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż legendę* jest ustawione na ”Włączone” oraz *Wyświetl min./śr./maks.* jest ustawione na ”Wyłączone”

Problemy

Następujące parametry są obsługiwane przy konfigurowaniu *Problemów*.

Parametr	type	name	value
Pokaż problemy	0	show_problems	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Tylko wybrane pozycje	0	graph_item_problems	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Hosty problemów	1	problemhosts.0	Nazwa hosta. Uwaga: Liczba w nazwie właściwości odnosi się do skonfigurowanego hosta. Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z kolejnym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Poziom ważności	0	severities.0	0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z kolejnym numerem w nazwie właściwości.
Problem	1	problem_name	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Tagi problemu			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*

Nadpisywanie

Następujące parametry są obsługiwane podczas konfigurowania *Nadpisywania*.

Note:
Pierwsza liczba w nazwie właściwości (np. or.0.hosts.0, or.0.items.0) reprezentuje określony zestaw danych, natomiast druga liczba, jeśli występuje, reprezentuje skonfigurowany host lub pozycję.

Parameter	type	name	value
Wzorce hostów	1	or.0.hosts.0	Nazwa lub wzorzec host (np. `Zabbix*`). Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie. Zachowanie parametru: - *wymagany* podczas konfigurowania *Nadpisywania*
Wzorce pozycji	1	or.0.items.0	Nazwa lub wzorzec pozycja (np. `*: Number of processed *values per second`). Podczas konfigurowania widżetu na szablonowym dashboardzie należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagany* podczas konfigurowania *Nadpisywania*
Kolor bazowy	1	or.0.color	Szesnastkowy kod koloru (np. `FF0000`).
Szerokość	0	or.0.width	Dopuszczalne wartości: 1-10.
Rysuj	0	or.0.type	0 - Linia; 1 - Punkty; 2 - Schodkowy; 3 - Słupek.
Przezroczystość	0	or.0.transparency	Dopuszczalne wartości: 1-10.
Wypełnienie	0	or.0.fill	Dopuszczalne wartości: 1-10.
Rozmiar punktu	0	or.0.pointsize	Dopuszczalne wartości: 1-10.
Brakujące dane	0	or.0.missingdatafunc	0 - Brak; 1 - Połączone; 2 - Traktuj jako 0; 3 - Ostatnio znane.
Oś Y	0	or.0.axisy	0 - Lewa; 1 - Prawa.
Przesunięcie czasu	1	or.0.timeshift	Prawidłowy ciąg czasu (np. `3600`, `1h` itd.). Możesz używać sufiksów czasu. Dozwolone są wartości ujemne.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Graph*. Aby uzyskać więcej informacji na temat konfiguracji pulpitu, zobacz `dashboard.create`.

Konfigurowanie widżetu *Graph*

Skonfiguruj widżet *Graph* w następujący sposób:

- 2 zestawy danych dla łącznie 9 pozycji na 1 hoście.
- Pierwszy zestaw danych jest typu „Item list” i składa się z 3 pozycji, które są reprezentowane przez linie o różnych kolorach, ale tej samej szerokości, przezroczystości i wypełnieniu.
- Drugi zestaw danych jest typu „Item patterns”, składa się z 6 pozycji, ma skonfigurowaną agregację i jest reprezentowany przez linię z niestandardowym kolorem, szerokością, przezroczystością i wypełnieniem.
- Drugi zestaw danych ma również niestandardową etykietę zestawu danych.
- Dane na wykresie są wyświetlane dla okresu ostatnich 3 godzin.
- Problemy na wykresie są wyświetlane tylko dla skonfigurowanych pozycji.
- Wykres ma dwie osie Y, z których prawa oś Y wyświetla wartości tylko dla drugiego zestawu danych.
- Legenda wykresu wyświetla skonfigurowane pozycje w 4 wierszach, a także minimalne, maksymalne i średnie wartości zestawów danych.

Żądanie:

```
{
    "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": 36,
                    "height": 5,
                    "view_mode": 0,
                    "fields": [
                        {
                            "type": 0,
                            "name": "ds.0.dataset_type",
                            "value": 0
                        },
                        {
                            "type": 4,
                            "name": "ds.0.itemids.0",
                            "value": 23264
                        },
                        {
                            "type": 1,
                            "name": "ds.0.color.0",
                            "value": "FF0000"
                        },
                        {
                            "type": 4,
                            "name": "ds.0.itemids.1",
                            "value": 23269
                        },
                        {
                            "type": 1,
                            "name": "ds.0.color.1",
                            "value": "BF00FF"
                        },
                        {
                            "type": 4,
                            "name": "ds.0.itemids.2",
                            "value": 23257
                        },
                        {
                            "type": 1,
                            "name": "ds.0.color.2",
                            "value": "0040FF"
                        },
                        {
                            "type": 0,
                            "name": "ds.0.width",
                            "value": 3
                        },
                        {
                            "type": 0,
                            "name": "ds.0.transparency",
                            "value": 3
                        },
                        {
                            "type": 0,
                            "name": "ds.0.fill",
                            "value": 1
                        },
                        {
                            "type": 1,
```

```
                "name": "ds.1.hosts.0",
                "value": "Zabbix server"
            },
            {
                "type": 1,
                "name": "ds.1.items.0",
                "value": "*: Number of processed *values per second"
            },
            {
                "type": 1,
                "name": "ds.1.color",
                "value": "000000"
            },
            {
                "type": 0,
                "name": "ds.1.transparency",
                "value": 0
            },
            {
                "type": 0,
                "name": "ds.1.fill",
                "value": 0
            },
            {
                "type": 0,
                "name": "ds.1.axisy",
                "value": 1
            },
            {
                "type": 0,
                "name": "ds.1.aggregate_function",
                "value": 3
            },
            {
                "type": 1,
                "name": "ds.1.aggregate_interval",
                "value": "1m"
            },
            {
                "type": 0,
                "name": "ds.1.aggregate_grouping",
                "value": 1
            },
            {
                "type": 1,
                "name": "ds.1.data_set_label",
                "value": "Number of processed values per second"
            },
            {
                "type": 0,
                "name": "graph_time",
                "value": 1
            },
            {
                "type": 1,
                "name": "time_period.from",
                "value": "now-3h"
            },
            {
                "type": 1,
                "name": "time_period.to",
                "value": "now"
```

```
                            },
                            {
                                "type": 0,
                                "name": "legend_statistic",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "legend_lines",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "show_problems",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "YZABC"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

9 Graph (classic)

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Graph (classic)* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph (classic)*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Graph (classic)*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Source	0	source_type	0 - *(domyślnie)* Graph; 1 - Simple graph.
Graph	6	graphid.0	ID Graph. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Graph"
Graph (Widget)	1	graphid._reference	Zamiast ID Graph: `ABCDE._graphid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla wykresów. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Simple graph" i *Graph* nie jest ustawione
Item	4	itemid.0	ID pozycja. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Simple graph" i *Item (Widget)* nie jest ustawione
Item (Widget)	1	itemid._reference	Zamiast ID pozycja: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla pozycji. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Simple graph" i *Item* nie jest ustawione
Time period	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itp.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawione - *wymagany* jeśli `time_period.to` jest ustawione

Parameter	type	name	value
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itp.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawione - *wymagany* jeśli `time_period.from` jest ustawione
Show legend	0	show_legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Override host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor *Host* na pulpicie jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Wykres (klasyczny)*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz `dashboard.create`.

Konfigurowanie widżetu *Graph (classic)*

Skonfiguruj widżet *Graph (classic)*, który wyświetla prosty wykres dla pozycji „42269".

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "source_type",
                                "value": 1
                            },
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42269
                            },
```

```
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "RSTUV"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

10 Graph prototype

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Graph prototype* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć niestandardowe widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph prototype*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Graph prototype*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Source	0	source_type	2 - *(domyślnie)* Graph prototype; 3 - Simple graph prototype.
Graph prototype	7	graphid.0	ID Graph prototype. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Graph prototype"
Item prototype	5	itemid.0	ID Item prototype. Zachowanie parametru: - *wymagany* jeśli *Source* jest ustawione na "Simple graph prototype"
Time period	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli `time_period.to` jest ustawiony
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli `time_period.from` jest ustawiony
Show legend	0	show_legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Override host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor *Host* na pulpicie jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Columns	0	columns	Możliwe wartości mieszczą się w zakresie od 1 do 24. Domyślnie: 2.
Rows	0	rows	Możliwe wartości mieszczą się w zakresie od 1 do 16. Domyślnie: 1.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Prototyp wykresu*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Graph prototype*

Skonfiguruj widżet *Graph prototype*, który wyświetla siatkę 3 wykresów (3 kolumny, 1 wiersz) utworzonych z prototypu pozycji (ID: ”42316”) przez wykrywanie niskopoziomowe.

Żądanie:

```
{
    "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": 48,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "source_type",
                                "value": 3
                            },
                            {
                                "type": 5,
                                "name": "itemid.0",
                                "value": 42316
                            },
                            {
                                "type": 0,
                                "name": "columns",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "OPQWX"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
```

```
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

11 Honeycomb

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Honeycomb* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Honeycomb*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widgetu *Honeycomb*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widgetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.

Parameter	type	name	value
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widgetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Host tags			
Evaluation type	0	evaltype_host	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Tag name	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.

Parameter	type	name	value
Tag value	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Item patterns	1	items.0	Nazwa lub wzorzec pozycji. Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widgetu dashboardu dla każdego wzorca pozycji, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagane*
Item tags			
Evaluation type	0	evaltype_item	0 - *(domyślnie)* And/Or; 2 - Or.
Tag name	1	item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Item tags*
Operator	0	item_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Item tags*
Tag value	1	item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Item tags*
Show hosts in maintenance	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Show	0	show.0	1 - Etykieta główna; 2 - Etykieta pomocnicza. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widgetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości. Domyślnie: 1, 2.

Parameter	type	name	value
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. ABCDE lub JBPNL). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widget. Zachowanie parametru: - *wymagane*

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane dla widżetu *Honeycomb*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odnosi się do pozycji progu na liście, posortowanej rosnąco. Jeśli jednak progi są skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontend Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parameter	type	name	value
Primary label			
Type	0	primary_label_type	0 - *(domyślnie)* Tekst; 1 - Wartość.
Text	1	primary_label	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {ITEM.*}, {INVENTORY.*}, makra użytkownika. Domyślnie: {HOST.NAME} Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na "Text"
Decimal places	0	primary_label_decimal_places	Zakres możliwych wartości: 0-6. Domyślnie: 2. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na "Value"
Size (type)	0	primary_label_size_type	0 - *(domyślnie)* Automatyczny; 1 - Niestandardowy.
Size	0	primary_label_size	Zakres możliwych wartości: 1-100. Domyślnie: 20. Zachowanie parametru: - *obsługiwany*, jeśli *Size* (type) jest ustawione na "Custom"
Bold	0	primary_label_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Color	1	primary_label_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: na podstawie `theme` obiektu Settings object i User object: 1F2C33 dla "blue-theme" lub "hc-light"; EEEEEE dla "dark-theme" lub "hc-dark".
Units (checkbox)	0	primary_label_units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na "Value"

Parameter	type	name	value
Units (value)	1	primary_label_units	Dowolna wartość tekstowa. "" (puste) Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value” i *Units* (checkbox) jest ustawione na ”Enabled”
Position	0	primary_label_units_pos	0 - Przed wartością; 1 - *(domyślnie)* Po wartości. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value” i *Units* (checkbox) jest ustawione na ”Enabled” Ten parametr jest ignorowany, jeśli zostanie ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Secondary label			
Type	0	secondary_label_type	0 - Tekst; 1 - *(domyślnie)* Wartość.
Text	1	secondary_label	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {ITEM.*}, {INVENTORY.*}, makra użytkownika. Domyślnie: {{ITEM.LASTVALUE}.fmtnum(2)} Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Text”
Decimal places	0	secondary_label_decimal_places	Zakres możliwych wartości: 0-6. Domyślnie: 2. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value”
Size (type)	0	secondary_label_size_type	0 - *(domyślnie)* Automatyczny; 1 - Niestandardowy.
Size	0	secondary_label_size	Zakres możliwych wartości: 1-100. Domyślnie: 30. Zachowanie parametru: - *obsługiwany*, jeśli *Size* (type) jest ustawione na ”Custom”
Bold	0	secondary_label_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Color	1	secondary_label_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: na podstawie `theme` obiektu Settings object i User object: `1F2C33` dla ”blue-theme” lub ”hc-light”; `EEEEEE` dla ”dark-theme” lub ”hc-dark”.
Units (checkbox)	0	secondary_label_units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value”
Units (value)	1	secondary_label_units	Dowolna wartość tekstowa. "" (puste) Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value” i *Units* (checkbox) jest ustawione na ”Enabled”

Parameter	type	name	value
Position	0	secondary_label_position	0 - Przed wartością; 1 - *(domyślnie)* Po wartości. Zachowanie parametru: - *obsługiwany*, jeśli *Type* jest ustawione na ”Value” i *Units* (checkbox) jest ustawione na ”Enabled” Ten parametr jest ignorowany, jeśli zostanie ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Background color			
Background color	1	bg_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: na podstawie `theme` obiektu Settings object i User object: D9E7ED dla ”blue-theme”; 3D5059 dla ”dark-theme”; AAD7E9 dla ”hc-light”; 335463 dla ”hc-dark”.
Thresholds			
Color interpolation	0	interpolation	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Color	1	thresholds.0.color	Szesnastkowy kod koloru (np. FF0000).
Threshold	1	thresholds.0.threshold	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. ”1d”, ”2w”, ”4K”, ”8G”).

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Honeycomb*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Honeycomb*

Skonfiguruj widżet *Honeycomb*, który wyświetla wykorzystanie procesów serwera Zabbix. Dodatkowo zmień główną etykietę komórek honeycomb i dostrój wizualnie widżet za pomocą progów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "honeycomb",
                        "name": "Honeycomb",
                        "x": "0",
                        "y": "0",
                        "width": "24",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 3,
```

```
                            "name": "hostids.0",
                            "value": 10084
                        },
                        {
                            "type": 1,
                            "name": "items.0",
                            "value": "Zabbix server: Utilization*"
                        },
                        {
                            "type": 1,
                            "name": "primary_label",
                            "value": "{ITEM.NAME}"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.0.color",
                            "value": "0EC9AC"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.0.threshold",
                            "value": "0"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.1.color",
                            "value": "FFD54F"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.1.threshold",
                            "value": "70"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.2.color",
                            "value": "FF465C"
                        },
                        {
                            "type": 1,
                            "name": "thresholds.2.threshold",
                            "value": "90"
                        },
                        {
                            "type": 1,
                            "name": "reference",
                            "value": "KSTMQ"
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
```

```
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

12 Dostępność hosta

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu pulpitu umożliwiają skonfigurowanie widżetu *Dostępność hosta* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Dostępność hosta*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Typ interfejsu	0	interface_type.0	0 - Brak; 1 - agent Zabbix (sprawdzanie pasywne); 2 - SNMP; 3 - IPMI; 4 - JMX; 5 - agent Zabbix (sprawdzanie aktywne). Domyślnie: 1, 2, 3, 4, 5 (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z numerem zwiększanym w nazwie właściwości.
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż tylko sumy	0	only_totals	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Dostępność hosta*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Dostępność hosta*

Skonfiguruj widżet *Dostępność hosta*, który wyświetla informacje o dostępności (w układzie pionowym) dla hostów z grupy hostów „4” ze skonfigurowanymi interfejsami „Zabbix agent” i „SNMP”.

Żądanie:

```
{
    "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": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "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
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

######13 Karta hosta {#manual-api-reference-dashboard-widget_fields-host_card}

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Karta hosta* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale wprowadza również ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Karta hosta*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Host card*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host	3	hostid.0	ID host. Zachowanie parametru: - *wymagany*, jeśli *Host (Widget/Dashboard)* nie jest ustawiony Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Host (Widget/Dashboard)	1	hostid._reference	Zamiast ID host: `DASHBOARD.hostid` - ustaw selektor *Host* jako źródło danych dla host; `ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla host. Zachowanie parametru: - *wymagany*, jeśli *Host* nie jest ustawiony Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Show suppressed problems	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Show	0	sections.0	0 - Grupy hostów; 1 - Opis; 2 - Monitorowanie; 3 - Dostępność; 4 - Monitorowany przez; 5 - Szablony; 6 - Inwentarz; 7 - Tagi. Uwaga: Numer w nazwie właściwości odnosi się do kolejności sekcji na liście sekcji. Aby skonfigurować wiele sekcji, utwórz obiekt pola widżetu dashboardu dla każdej sekcji, zwiększając numer w nazwie właściwości.

Następujące parametry są obsługiwane, jeśli *Show* jest ustawione na "Inventory".

Parameter	type	name	value
Inventory fields	0	inventory.0	ID inwentarza. Uwaga: Aby skonfigurować wiele pól inwentarza, utwórz obiekt pola widżetu dashboardu dla każdego pola inwentarza, zwiększając numer w nazwie właściwości.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Host card*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Host card*

Skonfiguruj widżet *Host card*, który wyświetla następujące sekcje: „Monitoring”, „Availability”, „Monitored by”, „Inventory” i „Tags”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostcard",
                        "name": "Host card",
                        "x": 0,
                        "y": 0,
                        "width": 14,
                        "height": 7,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 3,
                                "name": "hostid.0",
                                "value": 10084
                            },
                            {
                                "type": 0,
                                "name": "show_suppressed",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "sections.0",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "sections.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "sections.2",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "sections.3",
                                "value": 6
                            },
                            {
                                "type": 0,
                                "name": "sections.4",
                                "value": 7
                            },
                            {
                                "type": 0,
                                "name": "inventory.0",
                                "value": 25
                            },
                            {
```

```
                                "type": 0,
                                "name": "inventory.1",
                                "value": 26
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

14 Nawigator hostów

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Host navigator* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Host navigator*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane dla widżetu *Nawigator hostów*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z inkrementowanym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wzorce hostów	1	hosts.0	Nazwa lub wzorzec hosta. Uwaga: Aby skonfigurować wiele wzorców hostów, utwórz obiekt pola widżetu pulpitu dla każdego wzorca hosta z inkrementowanym numerem w nazwie właściwości. W przypadku wielu wzorców hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą hosty pasujące do skonfigurowanych wzorców hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Status hosta	0	status	-1 - *(domyślnie)* Dowolny; 0 - Włączony; 1 - Wyłączony. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Tagi hosta			
Typ oceny	0	host_tags_evaltype	0 - *(domyślnie)* I/Lub; 2 - Lub. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Poziom ważności	0	severities.0	0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z inkrementowanym numerem w nazwie właściwości.
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż problemy	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Nieukryte; 2 - Brak.
Grupuj według			
Atrybut	0	group_by.0.attribute	0 - Grupa hostów; 1 - Wartość tagu; 2 - Poziom ważności. Uwaga: Numer w nazwie właściwości odnosi się do kolejności atrybutu na liście atrybutów grupowania. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Grupuj według*

Parametr	type	name	value
Wartość	1	group_by.0.tag_name	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do atrybutu grupowania ustawionego w parametrze *Atrybut*. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Grupuj według*, gdy *Atrybut* jest ustawiony na "Wartość tagu"
Limit hostów	0	show_lines	Możliwe wartości mieszczą się w zakresie 1-9999. Domyślnie: 100. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Host navigator*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Host navigator*

Skonfiguruj widżet *Host navigator*, który wyświetla hosty pogrupowane według ich grupy hostów, a następnie według wartości tagu „city".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostnavigator",
                        "name": "Host navigator",
                        "x": "0",
                        "y": "0",
                        "width": "12",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 2
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 4
                            },
                            {
```

```
                                "type": 0,
                                "name": "group_by.0.attribute",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "group_by.1.attribute",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "group_by.1.tag_name",
                                "value": "city"
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "SWKLB"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

15 Karta pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Item*

card w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Item card*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Item card*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Item	4	itemid.0	ID pozycja. Zachowanie parametru: - *wymagany*, jeśli *Item (Widget)* nie jest ustawiony
Item (Widget)	1	itemid._reference	Zamiast ID pozycja: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla pozycji.
Show	0	sections.0	Zachowanie parametru: - *wymagany*, jeśli *Item* nie jest ustawiony 0 - Opis; 1 - Tekst błędu; 2 - Metryki; 3 - Najnowsze dane; 4 - Typ informacji; 5 - Wyzwalacze; 6 - Interfejs hosta; 7 - Typ; 8 - Inwentarz hosta; 9 - Tagi. Uwaga: Numer w nazwie właściwości odnosi się do kolejności sekcji na liście sekcji. Aby skonfigurować wiele sekcji, utwórz obiekt pola widżetu dashboardu dla każdej sekcji, zwiększając numer w nazwie właściwości.
Override host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor *Host* na dashboardzie jako źródło danych dla hostów. Domyślnie: "" (pusty) Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.

Sparkline

Następujące parametry są obsługiwane, jeśli *Show* jest ustawione na "Latest data".

Parameter	type	name	value
Width	0	sparkline.width	Możliwe wartości mieszczą się w zakresie 0-10. Domyślnie: 1.
Fill	0	sparkline.fill	Możliwe wartości mieszczą się w zakresie 0-10. Domyślnie: 3.
Color	1	sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `42A5F5`.
Time period	1	sparkline.time_period.reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `""` (puste)
From	1	sparkline.time_period.from	Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*. Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itp.). Domyślnie: now-1h. Zachowanie parametru: - *supported* jeśli *Time period* nie jest ustawiony - *required* jeśli `sparkline.time_period.to` jest ustawiony
To	1	sparkline.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itp.). Domyślnie: now. Zachowanie parametru: - *supported* jeśli *Time period* nie jest ustawiony - *required* jeśli `time_period.from` jest ustawiony
History data	0	sparkline.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Karta pozycji*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Karta pozycji*

Skonfiguruj widżet *Karta pozycji*, który wyświetla następujące sekcje: „Opis”, „Najnowsze dane”, „Wyzwalacze” i „Tagi”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemcard",
                        "name": "Item card",
                        "x": 0,
```

```
                        "y": 0,
                        "width": 14,
                        "height": 7,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42257
                            },
                            {
                                "type": 0,
                                "name": "sections.0",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "sections.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "sections.2",
                                "value": 5
                            },
                            {
                                "type": 0,
                                "name": "sections.3",
                                "value": 9
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

16 Historia pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Historia pozycji* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Historia pozycji*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Item history*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Layout	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Columns (see below)			
Show lines	0	show_lines	Dopuszczalny zakres wartości: 1-100. Domyślnie: 25.
Override host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor *Host* na pulpicie jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Advanced configuration (see below)			
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: *- required*

Kolumny

Kolumny mają wspólne parametry oraz dodatkowe parametry zależne od konfiguracji parametru *Pozycja*.

Note:
Dla wszystkich parametrów związanych z kolumnami liczba w nazwie właściwości (np. columns.0.name) odnosi się do kolumny, dla której parametr jest skonfigurowany.

Następujące parametry są obsługiwane dla wszystkich kolumn.

Parametr	type	name	value
Nazwa	1	columns.0.name	Dowolna wartość tekstowa. Zachowanie parametru: - *wymagane*
Pozycja	4	columns.0.itemid	ID pozycji. Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko pozycje skonfigurowane w szablonie. Zachowanie parametru: - *wymagane*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu numerycznego.

Parametr	type	name	value
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Bez zmian; 2 - Pasek; 3 - Wskaźniki.
Min	1	columns.0.min	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli *Wyświetlanie* jest ustawione na „Pasek” lub „Wskaźniki”
Max	1	columns.0.max	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli *Wyświetlanie* jest ustawione na „Pasek” lub „Wskaźniki”
Progi			
Kolor	1	columns.0.thresholds.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	columns.0.thresholds.threshold	Dowolna wartość liczbowa. Sufiksy (np. „1d”, „2w”, „4K”, „8G”) są obsługiwane.
Dane historyczne	0	columns.0.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu znakowego, tekstowego lub log.

Parametr	type	name	value
Wyróżnienia			
Wyróżnienie	1	columns.0.highlights.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	columns.0.highlights.pattern	Dowolne wyrażenie regularne.
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Bez zmian; 4 - HTML; 5 - Pojedyncza linia.
Pojedyncza linia	0	columns.0.max_length	Możliwe wartości mieszczą się w zakresie 1-500. Domyślnie: 100. Zachowanie parametru: - *obsługiwane*, jeśli *Wyświetlanie* jest ustawione na „Pojedyncza linia”

Parametr	type	name	value
Użyj czcionki o stałej szerokości	0	columns.0.monospace_font	0 - *(domyślnie)* Użyj domyślnej czcionki; 1 - Użyj czcionki o stałej szerokości.
Wyświetl czas lokalny	0	columns.0.local_time	0 - *(domyślnie)* Wyświetl znacznik czasu; 1 - Wyświetl czas lokalny. Zachowanie parametru: - *obsługiwane*, jeśli *pozycja* jest ustawiona jako pozycja typu log, a *Pokaż znacznik czasu* jest ustawione na „Włączone”

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu binarnego.

Parametr	type	nazwa	wartość
Pokaż miniaturę	1	columns.0.show_thumbnail	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane przez widżet *Item history*.

Parameter	type	name	value
New values	0	sortorder	0 - *(domyślnie)* Góra; 1 - Dół.
Show timestamp	0	show_timestamp	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Show column header	0	show_column_header	0 - Wyłączone; 1 - Poziome; 2 - *(domyślnie)* Pionowe.
Time period	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”`ABCDE`”) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie możesz ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli ustawiono `time_period.to`
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli ustawiono `time_period.from`

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Item history*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Historia pozycji*

Skonfiguruj widżet *Historia pozycji*, który wyświetla najnowsze dane dla dwóch numerycznych pozycji „42269" i „42270". Dodatkowo skonfiguruj kolumny pozycji tak, aby były wyświetlane pionowo, z nazwami kolumn wyświetlanymi poziomo; ogranicz widok do 15 wierszy danych i uwzględnij osobną kolumnę znacznika czasu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemhistory",
                        "name": "Item history",
                        "x": "0",
                        "y": "0",
                        "width": "18",
                        "height": "6",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": "0",
                                "name": "layout",
                                "value": "1"
                            },
                            {
                                "type": "1",
                                "name": "columns.0.name",
                                "value": "CPU utilization"
                            },
                            {
                                "type": "4",
                                "name": "columns.0.itemid",
                                "value": "42269"
                            },
                            {
                                "type": "1",
                                "name": "columns.1.name",
                                "value": "Memory utilization"
                            },
                            {
                                "type": "4",
                                "name": "columns.1.itemid",
                                "value": "42270"
                            },
                            {
                                "type": "0",
                                "name": "show_lines",
                                "value": "15"
                            },
                            {
                                "type": "0",
                                "name": "show_timestamp",
                                "value": "1"
                            },
```

```
                            {
                                "type": "0",
                                "name": "show_column_header",
                                "value": "1"
                            },
                            {
                                "type": "1",
                                "name": "reference",
                                "value": "KIVKD"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

17 Nawigator pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Item navigator* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Item navigator*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Item navigator*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostid` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Host tags			
Evaluation type	0	host_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Tag name	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.

Parameter	type	name	value
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Tag value	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Item pat-terns	1	items.0	Nazwa lub wzorzec pozycji. Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu dashboardu dla każdego wzorca pozycji, zwiększając numer w nazwie właściwości.
Item tags			
Evaluation type	0	item_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Tag name	1	item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Item tags*
Operator	0	item_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Item tags*
Tag value	1	item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Item tags*

Parameter	type	name	value
State	0	state	-1 - *(domyślnie)* Wszystkie; 0 - Normalny; 1 - Niewspierane.
Show problems	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Niewyciszone; 2 - Brak.
Group by			
Attribute	0	group_by.0.attribute	0 - Grupa hostów; 1 - Nazwa hosta; 2 - Wartość tagu hosta; 3 - Wartość tagu pozycji. Uwaga: Numer w nazwie właściwości odnosi się do kolejności atrybutów na liście atrybutów grupowania. Zachowanie parametru: - *wymagany* podczas konfigurowania *Group by*
Value	1	group_by.0.tag_name	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do atrybutu grupowania ustawionego w parametrze *Attribute*. Zachowanie parametru: - *wymagany* podczas konfigurowania *Group by* i gdy *Attribute* jest ustawiony na ”Wartość tagu hosta” lub ”Wartość tagu pozycji”
Item limit	0	show_lines	Dopuszczalny zakres wartości to 1-9999. Domyślnie: 100.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Item navigator*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Nawigator pozycji*

Skonfiguruj widżet *Nawigator pozycji*, który wyświetla do 1000 pozycji pogrupowanych według ich hosta, a następnie według wartości tagu pozycji „component”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemnavigator",
                        "name": "Item navigator",
                        "x": "0",
                        "y": "0",
```

```
                        "width": "12",
                        "height": "5",
                        "view_mode": "0",
                        "fields": [
                            {
                                "type": 0,
                                "name": "group_by.0.attribute",
                                "value": 0
                            },
                            {
                                "type": 0,
                                "name": "group_by.1.attribute",
                                "value": 3
                            },
                            {
                                "type": 1,
                                "name": "group_by.1.tag_name",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "show_lines",
                                "value": 1000
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "DFNLK"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

18 Wartość pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Wartość pozycji* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Wartość pozycji*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Item value*.

Parameter		type	name
Refresh interval		0	rf_rate
Item		4	itemid.0
	Item (Widget)	1	itemid._reference
Show		0	show.0
Override host		1	override_hostid._reference

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane przez widget *Item value*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odnosi się do pozycji progu na liście, posortowanej rosnąco. Jeśli jednak progi są skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widgetu w frontend Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parameter	type	name	value
Background color	1	bg_color	Hexadecimal color code (e.g. FF0000). Domyślnie: "" (puste).
Thresholds			
Color	1	thresholds.0.color	Hexadecimal color code (e.g. FF0000).
Threshold	1	thresholds.0.threshold	Dowolna wartość typu string.
Aggregation function	0	aggregate_function	0 - *(domyślnie)* nie używane; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.

Parameter	type	name	value
Time period	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw *selektor okresu czasu* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu wyłącznie w parametrach *From* i *To*. Zachowanie parametru: - *obsługiwane* jeśli *Aggregation function* jest ustawione na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last”
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Zachowanie parametru: - *obsługiwane* jeśli *Time period* nie jest ustawione, a *Aggregation function* jest ustawione na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagane* jeśli ustawiono `time_period.to`
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Zachowanie parametru: - *obsługiwane* jeśli *Time period* nie jest ustawione, a *Aggregation function* jest ustawione na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagane* jeśli ustawiono `time_period.from`
History data	0	history	0 - *(domyślnie)* Auto; 1 - History; 2 - Trends.

Opis

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli opcja *Pokaż* jest ustawiona na „Opis”.

Parametr	type	name	value
Opis	1	description	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {ITEM.*}, {INVENTORY.*}, makra użytkownika. Domyślnie: {ITEM.NAME}.
Pozycja pozioma	0	desc_h_pos	0 - Lewo; 1 - *(domyślnie)* Środek; 2 - Prawo.
Pozycja pionowa	0	desc_v_pos	Dwa lub więcej elementów (Opis, Wartość, Czas) nie może współdzielić tej samej *Pozycji poziomej* i *Pozycji pionowej*. 0 - Góra; 1 - Środek; 2 - *(domyślnie)* Dół.
Rozmiar	0	desc_size	Dwa lub więcej elementów (Opis, Wartość, Czas) nie może współdzielić tej samej *Pozycji poziomej* i *Pozycji pionowej*. Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 15.
Pogrubienie	0	desc_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parametr	type	name	value
Kolor	1	desc_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).

Wartość

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli dla opcji *Pokaż* ustawiono wartość „Wartość”.

Parametr	type	name	value
Miejsca dziesiętne			
Miejsca dziesiętne	0	decimal_places	Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 2.
Rozmiar	0	decimal_size	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 35.
Pozycja			
Pozycja pozioma	0	value_h_pos	0 - Lewo; 1 - *(domyślnie)* Środek; 2 - Prawo. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Pozycja pionowa	0	value_v_pos	0 - Góra; 1 - *(domyślnie)* Środek; 2 - Dół. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Rozmiar	0	value_size	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 45.
Pogrubienie	0	value_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Kolor	1	value_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Jednostki			
Jednostki (pole wyboru)	0	units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Jednostki (wartość)	1	units	Dowolna wartość tekstowa.
Pozycja	0	units_pos	0 - Przed wartością; 1 - Nad wartością; 2 - *(domyślnie)* Za wartością; 3 - Pod wartością.
Rozmiar	0	units_size	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 35.
Pogrubienie	0	units_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Kolor	1	units_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).

Czas

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli opcja *Pokaż* jest ustawiona na „Czas”.

Parametr	type	name	value
Pozycja pozioma	0	time_h_pos	0 - Lewo; 1 - *(domyślnie)* Środek; 2 - Prawo. Dwa lub więcej elementów (Opis, Wartość, Czas) nie mogą współdzielić tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Pozycja pionowa	0	time_v_pos	0 - *(domyślnie)* Góra; 1 - Środek; 2 - Dół. Dwa lub więcej elementów (Opis, Wartość, Czas) nie mogą współdzielić tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Rozmiar	0	time_size	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 15.
Pogrubienie	0	time_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	time_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Wskaźnik zmiany

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli dla opcji *Pokaż* ustawiono wartość „Wskaźnik zmiany”.

Parametr	type	name	value
Kolor wskaźnika zmiany ↑	1	up_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Kolor wskaźnika zmiany ↓	1	down_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Kolor wskaźnika zmiany ↕	1	updown_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Sparkline

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Show* ma ustawioną wartość ”Sparkline”.

Parameter	type	name	value
Width	0	sparkline.width	Dopuszczalne wartości mieszczą się w zakresie od 0 do 10. Domyślnie: 1.
Fill	0	sparkline.fill	Dopuszczalne wartości mieszczą się w zakresie od 0 do 10. Domyślnie: 3.
Color	1	sparkline.color	Szesnastkowy kod koloru (na przykład `FF0000`). Domyślnie: `42A5F5`.
Time period	1	sparkline.time_period.reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie możesz ustawić okres czasu wyłącznie w parametrach *From* i *To*.

Parameter	type	name	value
From	1	sparkline.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub relatywnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *wymagany*, jeśli ustawiono sparkline.time_period.to
To	1	sparkline.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub relatywnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *wymagany*, jeśli ustawiono sparkline.time_period.from
History data	0	sparkline.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Wartość pozycji*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Wartość pozycji*

Skonfiguruj widżet *Wartość pozycji*, który wyświetla wartość pozycji dla pozycji „42266” (dostępność agenta Zabbix). Dodatkowo dostosuj wizualnie widżet za pomocą wielu zaawansowanych opcji, w tym dynamicznego koloru tła, który zmienia się w zależności od stanu dostępności agenta Zabbix.

Żądanie:

```
{
    "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": 12,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 4,
                                "name": "itemid.0",
                                "value": 42266
                            },
                            {
                                "type": 0,
                                "name": "show.0",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "show.1",
                                "value": 2
                            },
                            {
                                "type": 0,
```

```
            "name": "show.2",
            "value": 3
        },
        {
            "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"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.0.color",
                                "value": "E1E1E1"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.0.threshold",
                                "value": "0"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.color",
                                "value": "D1C4E9"
                            },
                            {
                                "type": 1,
                                "name": "thresholds.1.threshold",
                                "value": "1"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

19 Mapa

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Map* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Map*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Mapa*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Mapa	8	sysmapid.0	ID mapy. Zachowanie parametru: - *wymagany*, jeśli *Mapa (widżet)* nie jest ustawiona
Mapa (widżet)	1	sysmapid._reference	`ABCDE._mapid` - ustawia widżet *Drzewo nawigacji map* (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla map. Zachowanie parametru: - *wymagany*, jeśli *Mapa* nie jest ustawiona
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Mapa*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Mapa*

Skonfiguruj widżet *Mapa*, który wyświetla mapę „1".

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "dashboard.create",
"parametry": {
"nazwa": "Mój pulpit",
```

```
"okres_wyświetlania": 30,
"automatyczny_start": 1,
"strony": [
{
"widżety": [
{
"typ": "mapa",
"nazwa": "Mapa",
"x": 0,
"y": 0,
"szerokość": 54,
"wysokość": 5,
"tryb_widoku": 0,
"pola": [
{
"typ": 8,
"nazwa": "sysmapid.0",
"wartość": 1
}
]
}
]
}
],
"grupy_użytkowników": [
{
"usrgrpid": 7,
"permission": 2
}
],
"users": [
{
"userid": 1,
"permission": 3
}
]
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"3"
]
},
"id": 1
}
```

Konfigurowanie połączonego widżetu *Mapa*

Skonfiguruj widżet *Mapa* połączony z widżetem *Drzewo nawigacji mapy*.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "dashboard.create",
"parametry": {
"nazwa": "Mój pulpit",
"okres_wyświetlania": 30,
"automatyczny_start": 1,
```

```
"strony": [
{
"widżety": [
{
"typ": "mapa",
"nazwa": "Mapa",
"x": 0,
"y": 5,
"szerokość": 54,
"wysokość": 5,
"tryb_widoku": 0,
"pola": [
{
"typ": 1,
"nazwa": "sysmapid._reference",
"wartość": "ABCDE._mapid"
}
]
},
{
"typ": „navtree”,
„name”: „Drzewo nawigacji mapy”,
„x”: 0,
„y”: 0,
„width”: 18,
„height”: 5,
„view_mode”: 0,
„fields”: [
{
„type”: 1,
„name”: „navtree.1.name”,
„value”: „Element A”
},
{
„type”: 1,
„name”: „navtree.2.name”,
„value”: „Element B”
},
{
„type”: 1,
„name”: „navtree.3.name”,
„value”: „Element C”
},
{
„type”: 1,
„name”: „navtree.4.name”,
„value”: „Element A1”
},
{
„type”: 1,
„name”: „navtree.5.name”,
„wartość”: „Element A2”
},
{
„typ”: 1,
„nazwa”: „navtree.6.name”,
„wartość”: „Element B1”
},
{
„typ”: 1,
„nazwa”: „navtree.7.name”,
„wartość”: „Element B2”
```

```
},
{
„typ": 0,
„nazwa": „navtree.4.parent",
„wartość": 1
},
{
„typ": 0,
„nazwa": „navtree.5.parent",
„wartość": 1
},
{
„typ": 0,
„nazwa": „navtree.6.parent",
„wartość": 2
},
{
„typ": 0,
„nazwa": „navtree.7.parent",
„wartość": 2
},
{
"typ": 0,
"nazwa": "navtree.1.order",
"wartość": 1
},
{
"typ": 0,
"nazwa": "navtree.2.order",
"wartość": 2
},
{
"typ": 0,
"nazwa": "navtree.3.order",
"wartość": 3
},
{
"typ": 0,
"nazwa": "navtree.4.order",
"wartość": 1
},
{
"typ": 0,
"nazwa": "navtree.5.order",
"wartość": 2
},
{
"typ": 0,
"nazwa": "navtree.6.order",
"wartość": 1
},
{
"typ": 0,
"nazwa": "navtree.7.order",
"wartość": 2
},
{
"type": 8,
"name": "navtree.6.sysmapid",
"value": 1
},
{
```

```
"type": 1,
"name": "reference",
"value": "ABCDE"
}
]
}
]
}
],
"userGroups": [
{
"usrgrpid": 7,
"permission": 2
}
],
"users": [
{
"userid": 1,
"permission": 3
}
]
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"3"
]
},
"id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`
- Drzewo nawigacyjne mapy

20 Drzewo nawigacji mapy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Drzewo nawigacji map* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Drzewo nawigacji map*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Drzewo nawigacji map*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Pokaż niedostępne mapy	1	show_unavailable	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametrów: - *wymagane*

Następujące parametry są obsługiwane przy konfigurowaniu elementów drzewa nawigacji map.

Parametr	type	name	value
Nazwa	1	navtree.1.name	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości określa numer elementu.
Powiązana mapa	8	navtree.1.sysmapid	ID mapy. Uwaga: Liczba w nazwie właściwości odnosi się do elementu, z którym mapa jest powiązana.
Parametry tworzenia hierarchii elementów	0	navtree.1.parent	Numer elementu nadrzędnego. Uwaga: Liczba w nazwie właściwości odnosi się do elementu podrzędnego. Wartość właściwości odnosi się do elementu nadrzędnego.
	0	navtree.1.order	Pozycja elementu w drzewie nawigacji map. Uwaga: Liczba w nazwie właściwości odnosi się do numeru elementu. Wartość właściwości odnosi się do pozycji elementu w drzewie nawigacji map. Pozycja elementu nadrzędnego jest określana w obrębie całego drzewa nawigacji map. Pozycja elementu podrzędnego jest określana w obrębie elementu nadrzędnego.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Drzewo nawigacji map*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Map navigation tree*

Skonfiguruj widżet *Map navigation tree*, który wyświetla następujące drzewo nawigacji map:

- Element A
  - Element A1
  - Element A2
- Element B
  - Element B1 (zawiera podlinkowaną mapę „1”, którą można wyświetlić w podlinkowanym widżecie *Map widget*)
  - Element B2
- Element C

Żądanie:

```
{
    "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": 18,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "navtree.1.name",
                                "value": "Element A"
                            },
                            {
                                "type": 1,
                                "name": "navtree.2.name",
                                "value": "Element B"
                            },
                            {
                                "type": 1,
                                "name": "navtree.3.name",
                                "value": "Element C"
                            },
                            {
                                "type": 1,
                                "name": "navtree.4.name",
                                "value": "Element A1"
                            },
                            {
                                "type": 1,
                                "name": "navtree.5.name",
                                "value": "Element A2"
                            },
                            {
                                "type": 1,
                                "name": "navtree.6.name",
                                "value": "Element B1"
                            },
                            {
                                "type": 1,
                                "name": "navtree.7.name",
                                "value": "Element B2"
                            },
                            {
                                "type": 0,
                                "name": "navtree.4.parent",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "navtree.5.parent",
                                "value": 1
                            },
                            {
                                "type": 0,
```

```
                            "name": "navtree.6.parent",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.7.parent",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.1.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.2.order",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.3.order",
                            "value": 3
                        },
                        {
                            "type": 0,
                            "name": "navtree.4.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.5.order",
                            "value": 2
                        },
                        {
                            "type": 0,
                            "name": "navtree.6.order",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "navtree.7.order",
                            "value": 2
                        },
                        {
                            "type": 8,
                            "name": "navtree.6.sysmapid",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "reference",
                            "value": "HJQXF"
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
```

```
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`
- Mapy

21 Wykres kołowy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Pie chart* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Pie chart*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Wykres kołowy*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Zestaw danych

Następujące parametry są obsługiwane podczas konfigurowania *Zestawu danych*.

Note:
Pierwsza liczba w nazwie właściwości (np. ds.0.hosts.0, ds.0.items.0) oznacza konkretny zestaw danych, natomiast druga liczba, jeśli występuje, oznacza skonfigurowany host lub pozycję.

Parameter	type	name	value
Typ zestawu danych	0	ds.0.dataset_type	0 - Lista pozycji; 1 - *(domyślnie)* Wzorce pozycji.
Pozycje	4	ds.0.itemids.0	ID pozycji. Podczas konfigurowania widżetu na szablonie dashboardu należy ustawić tylko pozycje skonfigurowane w szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widżetu dashboardu dla każdej pozycji, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagane*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji” i *Pozycje (Widget)* nie są ustawione
Pozycje (Widget)	1	ds.0.itemids.0._reference	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widżetów, utwórz obiekt pola widżetu dashboardu dla każdego widżetu, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagane*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji” i *Pozycje* nie są ustawione
Kolor	1	ds.0.color.0	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *wymagane*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”
Typ pozycji	0	ds.0.type.0	0 - *(domyślnie)* Normalny; 1 - Suma. Wartość ”Suma” można ustawić tylko dla jednej pozycji w całym wykresie. Zachowanie parametru: - *obsługiwane*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”
Wzorce hostów	1	ds.0.hosts.0	Nazwa lub wzorzec hosta (np. ”Zabbix*”). Zachowanie parametru: - *wymagane*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”
Wzorce pozycji	1	ds.0.items.0	Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie dashboardu. Nazwa lub wzorzec pozycji (np. ”*: Number of processed *values per second”). Podczas konfigurowania widżetu na szablonie dashboardu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagane*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”

Parameter	type	name	value
Kolor	1	ds.0.color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwane*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji” i *Paleta kolorów* nie jest ustawiona
Paleta kolorów	0	ds.0.color_palette	Indeks palety kolorów. Możliwe wartości mieszczą się w zakresie od 0 do 11. Domyślnie: 0. Zachowanie parametru: - *obsługiwane*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji” i *Kolor* nie jest ustawiony
Funkcja agregacji	0	ds.0.aggregate_function	1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - *(domyślnie)* last.
Agregacja zestawu danych	0	ds.0.dataset_aggregation	0 - *(domyślnie)* brak; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum. Zachowanie parametru: - *obsługiwane*, jeśli *Typ pozycji* jest ustawiony na ”Suma”
Etykieta zestawu danych	1	ds.0.data_set_label	Dowolna wartość tekstowa. Domyślnie: "" (puste).

Opcje wyświetlania

Następujące parametry są obsługiwane przy konfigurowaniu *opcji wyświetlania*.

Parameter	type	name	value
Wybór danych historycznych	0	source	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.
Rysuj	0	draw_type	0 - *(domyślnie)* Wykres kołowy; 1 - Pierścień.
Szerokość	0	width	20 - 20% promienia; 30 - 30% promienia; 40 - 40% promienia; 50 - *(domyślnie)* 50% promienia. Zachowanie parametru: - *obsługiwane*, jeśli *Rysuj* jest ustawione na ”Pierścień”
Szerokość obramowania	0	stroke	Możliwe wartości mieszczą się w zakresie 0-10. Domyślnie: 0. Zachowanie parametru: - *obsługiwane*, jeśli *Rysuj* jest ustawione na ”Pierścień”

Parameter	type	name	value
Pokaż wartość całkowitą	0	total_show	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Rysuj* jest ustawione na ”Pierścień”
Rozmiar	0	value_size_type	0 - *(domyślnie)* Auto; 1 - Niestandardowy. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Rozmiar (wartość dla rozmiaru niestandardowego)	0	value_size	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 20. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Miejsca dziesiętne	0	decimal_places	Możliwe wartości mieszczą się w zakresie 0-6. Domyślnie: 2. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Jednostki (pole wyboru)	0	units_show	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Jednostki (wartość)	1	units	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na ”Włączone”
Pogrubienie	0	value_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Kolor	1	value_color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość całkowitą* jest ustawione na ”Włączone”
Odstęp między sektorami	0	space	Możliwe wartości mieszczą się w zakresie 0-10. Domyślnie: 1.
Scal sektory mniejsze niż N% (pole wyboru)	0	merge	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parameter	type	name	value
Scal sek- tory mniejsze niż N% (wartość)	0	merge_percent	Możliwe wartości mieszczą się w zakresie 1-10. Domyślnie: 1. Zachowanie parametru: - *obsługiwane*, jeśli *Scal sektory mniejsze niż N%* (pole wyboru) jest ustawione na ”Włączone”
Scal sek- tory mniejsze niż N% (kolor)	1	merge_color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwane*, jeśli *Scal sektory mniejsze niż N%* (pole wyboru) jest ustawione na ”Włączone”

Okres czasu

Następujące parametry są obsługiwane podczas konfigurowania *Okresu czasu*.

Parameter	type	name	value
Okres czasu	1	time_period._referenc	`DASHBOARD._timeperiod` - ustaw selektor *Okres czasu* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli `time_period.to` jest ustawiony
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli `time_period.from` jest ustawiony

Legenda

Następujące parametry są obsługiwane przy konfigurowaniu *Legendy*.

Parametr	type	name	value
Pokaż legendę	0	legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Pokaż wartość	0	legend_value	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Pokaż funkcję agregacji	0	legend_aggregation	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na ”Włączone”

Parametr	type	name	value
Wiersze	0	legend_lines_mode	0 - *(domyślnie)* Stałe; 1 - Zmienne. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Liczba wierszy/ Maksymalna liczba wierszy	0	legend_lines	Możliwe wartości mieszczą się w zakresie od 1 do 10. Domyślnie: 1. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na ”Włączone”
Liczba kolumn	0	legend_columns	Możliwe wartości mieszczą się w zakresie od 1 do 4. Domyślnie: 4. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż wartość* jest ustawione na ”Wyłączone”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Pie chart*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Pie chart*

Skonfiguruj widżet *Pie chart* w następujący sposób:

- 2 zestawy danych dla łącznie 9 pozycji na 1 hoście.
- Pierwszy zestaw danych jest typu ”Item list” i składa się z 3 pozycji, z których wszystkie mają typ ”Normal” i są reprezentowane różnymi kolorami.
- Drugi zestaw danych jest typu ”Item patterns”, składa się z 6 pozycji, ma skonfigurowaną agregację dla każdej pozycji i jest reprezentowany przez niestandardowy kolor.
- Drugi zestaw danych ma również niestandardową etykietę zestawu danych.
- Dane na wykresie kołowym są wyświetlane jako wykres pierścieniowy z niestandardową szerokością oraz wartością całkowitą z jednostkami w środku.
- Dane na wykresie kołowym są wyświetlane i agregowane dla niestandardowego okresu czasu obejmującego ostatnie 3 godziny.
- Legenda wykresu kołowego wyświetla skonfigurowane pozycje w 4 wierszach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "piechart",
                        "name": "Pie chart",
                        "x": 0,
                        "y": 0,
                        "width": 24,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "ds.0.dataset_type",
                                "value": 0
```

```
        },
        {
            "type": 4,
            "name": "ds.0.itemids.0",
            "value": 23264
        },
        {
            "type": 1,
            "name": "ds.0.color.0",
            "value": "FF0000"
        },
        {
            "type": 0,
            "name": "ds.0.type.0",
            "value": 0
        },
        {
            "type": 4,
            "name": "ds.0.itemids.1",
            "value": 23269
        },
        {
            "type": 1,
            "name": "ds.0.color.1",
            "value": "BF00FF"
        },
        {
          "type": 0,
          "name": "ds.0.type.1",
          "value": 0
        },
        {
            "type": 4,
            "name": "ds.0.itemids.2",
            "value": 23257
        },
        {
            "type": 1,
            "name": "ds.0.color.2",
            "value": "0040FF"
        },
        {
          "type": 0,
          "name": "ds.0.type.2",
          "value": 0
        },
        {
            "type": 1,
            "name": "ds.1.hosts.0",
            "value": "Zabbix server"
        },
        {
            "type": 1,
            "name": "ds.1.items.0",
            "value": "*: Number of processed *values per second"
        },
        {
            "type": 1,
            "name": "ds.1.color",
            "value": "000000"
        },
        {
```

```
                            "type": 0,
                            "name": "ds.1.aggregate_function",
                            "value": 3
                        },
                        {
                            "type": 1,
                            "name": "ds.1.data_set_label",
                            "value": "Number of processed values per second"
                        },
                        {
                            "type": 0,
                            "name": "draw_type",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "width",
                            "value": 30
                        },
                        {
                            "type": 0,
                            "name": "total_show",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "units_show",
                            "value": 1
                        },
                        {
                            "type": 0,
                            "name": "graph_time",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "time_period.from",
                            "value": "now-3h"
                        },
                        {
                            "type": 1,
                            "name": "time_period.to",
                            "value": "now"
                        },
                        {
                            "type": 0,
                            "name": "legend_lines",
                            "value": 4
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
```

```
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

22 Hosty z problemami

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Hosty z problemami* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Hosty z problemami*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Problem hosts*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.

Parameter	type	name	value
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.
Exclude host groups	2	exclude_groupids.0	ID grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nieskonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Severity	0	severities.0	0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Problem tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*

Parameter	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Show suppressed problems	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Hide groups without problems	0	hide_empty_groups	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablon dashboardu.
Problem display	0	ext_ack	0 - *(domyślnie)* Wszystkie; 1 - Tylko nierozwiązane; 2 - Rozdzielone.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problem hosts*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problem hosts*

Skonfiguruj widżet *Problem hosts*, który wyświetla hosty z grup hostów „2” i „4”, mające problemy z nazwą zawierającą ciąg „CPU” oraz o następujących poziomach ważności: „Warning”, „Average”, „High”, „Disaster”.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 2
                            },
                            {
                                "type": 2,
                                "name": "groupids.1",
                                "value": 4
                            },
                            {
                                "type": 1,
                                "name": "problem",
                                "value": "cpu"
                            },
                            {
                                "type": 0,
                                "name": "severities.0",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "severities.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "severities.2",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "severities.3",
                                "value": 5
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
```

```
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

23 Problemy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Problems* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Problems*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Problems*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Show	0	show	1 - *(domyślnie)* Ostatnie problemy; 2 - Historia; 3 - Problemy.
Host groups	2	groupids.0	ID host group. Uwaga: Aby skonfigurować wiele host group, utwórz obiekt pola widżetu dashboardu dla każdej host group, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na template dashboard.
Host groups (Widget)	1	groupids._reference	Zamiast ID host group: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla host group. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na template dashboard.

Parameter	type	name	value
Exclude host groups	2	exclude_groupids.0	ID host group. Uwaga: Aby wykluczyć wiele host group, utwórz obiekt pola widżetu dashboardu dla każdej host group, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na template dashboard.
Hosts	3	hostids.0	ID host. Uwaga: Aby skonfigurować wiele host, utwórz obiekt pola widżetu dashboardu dla każdego host, zwiększając numer w nazwie właściwości. W przypadku wielu host parametr *Host groups* musi być albo w ogóle nieskonfigurowany, albo skonfigurowany z co najmniej jedną host group, do której należą skonfigurowane host. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na template dashboard.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID Host: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla host; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla host. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na template dashboard.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Severity	0	severities.0	0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Problem tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*

Parameter	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Show tags	0	show_tags	0 - *(domyślnie)* Brak; 1 - 1; 2 - 2; 3 - 3.
Tag name (format)	0	tag_name_format	0 - *(domyślnie)* Pełny; 1 - Skrócony; 2 - Brak. Zachowanie parametru: - *obsługiwany* jeśli *Show tags* ma wartość ”1”, ”2” lub ”3”
Tag display priority	1	tag_priority	Lista tagów oddzielona przecinkami. Zachowanie parametru: - *obsługiwany* jeśli *Show tags* ma wartość ”1”, ”2” lub ”3”
Show operational data	0	show_opdata	0 - *(domyślnie)* Brak; 1 - Osobno; 2 - Z nazwą problemu.
Show suppressed problems	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Acknowledgement status	0	acknowledgement_status	0 - *(domyślnie)* wszystkie; 1 - Niepotwierdzone; 2 - Potwierdzone.
By me	0	acknowledged_by_me	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parameter	type	name	value
Sort entries by	0	sort_triggers	1 - Severity (malejąco); 2 - Host (rosnąco); 3 - Time (rosnąco); 4 - *(domyślnie)* Time (malejąco); 13 - Severity (rosnąco); 14 - Host (malejąco); 15 - Problem (rosnąco); 16 - Problem (malejąco). Dla wszystkich wartości, z wyjątkiem ”Time (malejąco)” i ”Time (rosnąco)”, parametr *Show timeline* musi mieć ustawioną wartość ”Disabled”. Wartości ”Host (rosnąco)” i ”Host (malejąco)” nie są obsługiwane podczas konfigurowania widżetu na template dashboard.
Show timeline	0	show_timeline	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany* jeśli *Sort entries by* ma ustawioną wartość ”Time (malejąco)” lub ”Time (rosnąco)”
Highlight whole row	0	highlight_row	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Ten parametr nie jest obsługiwany w motywach o wysokim kontraście. Ta opcja jest obsługiwana od Zabbix 7.4.3.
Show lines	0	show_lines	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 25.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problemy*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problemy*

Skonfiguruj widżet *Problemy*, który wyświetla problemy dla grupy hostów „4”, spełniające następujące warunki:

- Problemy, które mają znacznik o nazwie „scope”, zawierający wartości „performance” lub „availability”, albo „capacity”.
- Problemy o następujących poziomach ważności: „Warning”, „Average”, „High”, „Disaster”.

Dodatkowo skonfiguruj widżet tak, aby wyświetlał znaczniki i dane operacyjne.

Żądanie:

```
{
    "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": 36,
                    "height": 5,
                    "view_mode": 0,
                    "fields": [
                        {
                            "type": 2,
                            "name": "groupids.0",
                            "value": 4
                        },
                        {
                            "type": 1,
                            "name": "tags.0.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.0.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.0.value",
                            "value": "performance"
                        },
                        {
                            "type": 1,
                            "name": "tags.1.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.1.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.1.value",
                            "value": "availability"
                        },
                        {
                            "type": 1,
                            "name": "tags.2.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.2.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.2.value",
                            "value": "capacity"
                        },
                        {
                            "type": 0,
                            "name": "severities.0",
```

```
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "severities.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "severities.2",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "severities.3",
                                "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
            }
        ]
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`

- `dashboard.update`

24 Problemy według ważności

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Problemy według ważności* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Problemy według ważności*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Problems by severity*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Exclude host groups	2	exclude_groupids.0	ID grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.

Parameter	type	name	value
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Severity	0	severities.0	0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości, zwiększając numer w nazwie właściwości.
Problem tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub.
Tag name	1	tags.0.tag	Dowolny ciąg znaków. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Tag value	1	tags.0.value	Dowolny ciąg znaków. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Show	0	show_type	0 - *(domyślnie)* Grupy hostów; 1 - Sumy. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie i domyślnie jest ustawiony na "Sumy".
Layout	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy. Zachowanie parametru: - *obsługiwany* jeśli *Show* jest ustawione na "Sumy"

Parameter	type	name	value
Show operational data	0	show_opdata	0 - *(domyślnie)* Brak; 1 - Osobno; 2 - Z nazwą problemu.
Show suppressed problems	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Hide groups without problems	0	hide_empty_groups	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwany* jeśli *Show* jest ustawione na ”Grupy hostów” Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym pulpicie.
Problem display	0	ext_ack	0 - *(domyślnie)* Wszystkie; 1 - Tylko niepotwierdzone; 2 - Rozdzielone.
Show timeline	0	show_timeline	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Reference	1	reference	Dowolny ciąg znaków składający się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problemy według ważności*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problemy według ważności*

Skonfiguruj widżet *Problemy według ważności*, który wyświetla łączną liczbę problemów dla wszystkich grup hostów.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
```

```
                                "type": 0,
                                "name": "show_type",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

26 Informacje o systemie

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Informacje o systemie* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Informacje o systemie*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Informacje o systemie*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Pokaż	0	info_type	0 - *(domyślnie)* Statystyki systemu; 1 - Węzły wysokiej dostępności.
Pokaż szczegóły sprawdzania aktualizacji oprogramowania	0	show_software_update_check_details	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwany*, jeśli `AllowSoftwareUpdateCheck` w konfiguracji serwera Zabbix jest włączony i *Pokaż* jest ustawione na „Statystyki systemu”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Informacje o systemie*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Informacje o systemie*

Skonfiguruj widżet *Informacje o systemie*, który wyświetla statystyki systemowe z interwałem odświeżania wynoszącym 10 minut oraz z włączonym sprawdzaniem aktualizacji oprogramowania.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 0,
                                "name": "rf_rate",
                                "value": 600
                            },
                            {
                                "type": 0,
                                "name": "show_software_update_check_details",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
```

```
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

27 Najlepsze hosty

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Top hosts* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Top hosts*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Top hosts*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Parameter	type	name	value
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Host tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.

Parameter	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Show hosts in maintenance	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Columns (see below)			
Order by	0	column	Numeryczna wartość kolumny z skonfigurowanych kolumn.
Order	0	order	2 - *(domyślnie)* Top N; 3 - Bottom N.
Host limit	0	show_lines	Możliwe wartości mieszczą się w zakresie 1-1000. Domyślnie: 10. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.

Kolumny

Kolumny posiadają wspólne parametry oraz dodatkowe parametry w zależności od konfiguracji parametru *Data*.

Note:
Dla wszystkich parametrów związanych z kolumnami, liczba w nazwie właściwości (np. columns.name.0) odnosi się do kolumny, dla której parametr jest skonfigurowany.

Następujące parametry są obsługiwane dla wszystkich kolumn.

Parametr	type	name	value
Nazwa	1	columns.0.name	Dowolna wartość tekstowa. Zachowanie parametru: - *wymagany*
Dane	0	columns.0.data	1 - Wartość pozycji; 2 - Nazwa hosta; 3 - Tekst. Zachowanie parametru: - *wymagany*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *wymagany*

Wartość pozycji

Następujące parametry są obsługiwane, jeśli *Data* jest ustawione na ”Item value”.

Note:
Pierwsza liczba w nazwie właściwości *Thresholds* (np. columnsthresholds.0.color.0) odnosi się do kolumny, dla której skonfigurowano progi, natomiast druga liczba odnosi się do pozycji progu na liście posortowanej rosnąco. Jeśli jednak progi są skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontend Zabbix (np. `"threshold.0.threshold":"5"` → `"threshold.0.threshold":"1"`; `"threshold.1.threshold":"1"` → `"threshold.1.threshold": "5"`).

Parameter	type	name	value
Item	1	columns.0.item	Prawidłowa nazwa pozycji. Podczas konfigurowania widżetu na szablonowym dashboardzie należy ustawić tylko pozycje skonfigurowane w szablonie.
Display item value as	0	columns.0.display_value_as	0 - *(domyślnie)* Liczbowe; 1 - Tekstowe; 2 - Binarnie.
Display	0	columns.0.display	1 - *(domyślnie)* Jak jest; 2 - Pasek; 3 - Wskaźniki; 6 - Sparkline. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric”
Min	1	columns.0.min	Dowolna wartość liczbowa. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Bar” lub ”Indicators”
Max	1	columns.0.max	Dowolna wartość liczbowa. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Bar” lub ”Indicators”
Decimal places	0	columns.0.decimal_places	Zakres możliwych wartości: 0-10. Domyślnie: 2. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric”

Parameter	type	name	value
Sparkline			
Width	0	columns.0.sparkline.width	Zakres możliwych wartości: 0-10. Domyślnie: 1. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline”
Fill	0	columns.0.sparkline.fill	Zakres możliwych wartości: 0-10. Domyślnie: 3. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline”
Color	1	columns.0.sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `42A5F5`. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline”
Time period	1	columns.0.sparkline.time_period	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline”
From	1	columns.0.sparkline.time_period.from	Początkowy punkt czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline” - *wymagane*, jeśli ustawiono `columns.0.sparkline.time_period.to`
To	1	columns.0.sparkline.time_period.to	Końcowy punkt czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline” - *wymagane*, jeśli ustawiono `columns.0.sparkline.time_period.from`
History data	0	columns.0.sparkline.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na ”Numeric” i *Display* jest ustawione na ”Sparkline”
Thresholds			

Parameter	type	name	value
Color	1	columnsthresholds.0.color	Szesnastkowy kod koloru (np. `FF0000`). Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Numeric"
Threshold	1	columnsthresholds.0.threshold	Dowolna wartość tekstowa. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Numeric"
Highlights			
Color	1	columns.0.highlights.color	Szesnastkowy kod koloru (np. `FF0000`). Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Text"
Pattern	1	columns.0.highlights.pattern	Dowolna wartość tekstowa. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Text"
Show thumbnail	0	columns.0.show_thumbnail	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Binary"
Aggregation function	0	columns.0.aggregate_function	0 - *(domyślnie)* nieużywane; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Time period	1	columns.0.time_period.reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*. Parameter behavior: - *obsługiwane*, jeśli *Aggregation function* jest ustawione na "min", "max", "avg", "count", "sum", "first", "last"
From	1	columns.0.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Parameter behavior: - *obsługiwane*, jeśli *Time period* nie jest ustawione i *Aggregation function* jest ustawione na "min", "max", "avg", "count", "sum", "first", "last" - *wymagane*, jeśli ustawiono `columns.0.time_period.to` - *wymagane*, jeśli ustawiono `columns.0.time_period.to`
To	1	columns.0.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w`, itd.). Parameter behavior: - *obsługiwane*, jeśli *Time period* nie jest ustawione i *Aggregation function* jest ustawione na "min", "max", "avg", "count", "sum", "first", "last" - *wymagane*, jeśli ustawiono `columns.0.time_period.from`

Parameter	type	name	value
History data	0	columns.0.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy. Parameter behavior: - *obsługiwane*, jeśli *Display item value as* jest ustawione na "Numeric"
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikalna w obrębie dashboardu, do którego należy widżet. Parameter behavior: - *wymagane*

Tekst

Następujące parametry są obsługiwane, jeśli *Dane* są ustawione na "Tekst".

Parametr	typ	nazwa	wartość
Tekst	1	columns.0.text	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {INVENTORY.*}. Zachowanie parametru: - *wymagane*, jeśli *Dane* są ustawione na „Tekst"

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Top hosts*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Top hosts*

Skonfiguruj widżet *Top hosts*, który wyświetla hosty o najwyższym wykorzystaniu CPU w grupie hostów „4". Dodatkowo skonfiguruj następujące niestandardowe kolumny: „Nazwa hosta", „Wykorzystanie CPU w %", „Średnia 1m", „Średnia 5m", „Średnia 15m", „Procesy".

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
```

```
                    },
                    {
                        "type": 1,
                        "name": "columns.0.name",
                        "value": "Host"
                    },
                    {
                        "type": 0,
                        "name": "columns.0.data",
                        "value": 2
                    },
                    {
                        "type": 1,
                        "name": "columns.0.base_color",
                        "value": "FFFFFF"
                    },
                    {
                        "type": 1,
                        "name": "columns.1.name",
                        "value": "CPU utilization in %"
                    },
                    {
                        "type": 0,
                        "name": "columns.1.data",
                        "value": 1
                    },
                    {
                        "type": 1,
                        "name": "columns.1.base_color",
                        "value": "4CAF50"
                    },
                    {
                        "type": 1,
                        "name": "columns.1.item",
                        "value": "CPU utilization"
                    },
                    {
                        "type": 0,
                        "name": "columns.1.display",
                        "value": 3
                    },
                    {
                        "type": 1,
                        "name": "columns.1.min",
                        "value": "0"
                    },
                    {
                        "type": 1,
                        "name": "columns.1.max",
                        "value": "100"
                    },
                    {
                        "type": 1,
                        "name": "columnsthresholds.1.color.0",
                        "value": "FFFF00"
                    },
                    {
                        "type": 1,
                        "name": "columnsthresholds.1.threshold.0",
                        "value": "50"
                    },
                    {
```

```
                "type": 1,
                "name": "columnsthresholds.1.color.1",
                "value": "FF8000"
            },
            {
                "type": 1,
                "name": "columnsthresholds.1.threshold.1",
                "value": "80"
            },
            {
                "type": 1,
                "name": "columnsthresholds.1.color.2",
                "value": "FF4000"
            },
            {
                "type": 1,
                "name": "columnsthresholds.1.threshold.2",
                "value": "90"
            },
            {
                "type": 1,
                "name": "columns.2.name",
                "value": "1m avg"
            },
            {
                "type": 0,
                "name": "columns.2.data",
                "value": 1
            },
            {
                "type": 1,
                "name": "columns.2.base_color",
                "value": "FFFFFF"
            },
            {
                "type": 1,
                "name": "columns.2.item",
                "value": "Load average (1m avg)"
            },
            {
                "type": 1,
                "name": "columns.3.name",
                "value": "5m avg"
            },
            {
                "type": 0,
                "name": "columns.3.data",
                "value": 1
            },
            {
                "type": 1,
                "name": "columns.3.base_color",
                "value": "FFFFFF"
            },
            {
                "type": 1,
                "name": "columns.3.item",
                "value": "Load average (5m avg)"
            },
            {
                "type": 1,
                "name": "columns.4.name",
```

```
                            "value": "15m avg"
                        },
                        {
                            "type": 0,
                            "name": "columns.4.data",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "columns.4.base_color",
                            "value": "FFFFFF"
                        },
                        {
                            "type": 1,
                            "name": "columns.4.item",
                            "value": "Load average (15m avg)"
                        },
                        {
                            "type": 1,
                            "name": "columns.5.name",
                            "value": "Processes"
                        },
                        {
                            "type": 0,
                            "name": "columns.5.data",
                            "value": 1
                        },
                        {
                            "type": 1,
                            "name": "columns.5.base_color",
                            "value": "FFFFFF"
                        },
                        {
                            "type": 1,
                            "name": "columns.5.item",
                            "value": "Number of processes"
                        },
                        {
                            "type": 0,
                            "name": "columns.5.decimal_places",
                            "value": 0
                        },
                        {
                            "type": 0,
                            "name": "column",
                            "value": 1
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
```

```
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

28 Najważniejsze pozycje

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu dashboardu umożliwiają konfigurację widżetu *Top items* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widżetu nie są weryfikowane podczas tworzenia ani aktualizacji dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widżetów oraz tworzenie niestandardowych widżetów, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widżetów. Aby zapewnić pomyślne utworzenie lub aktualizację widżetu *Top items*, należy zapoznać się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Top items*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.

Parameter	type	name	value
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Host tags			
Evaluation type	0	host_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Tag name	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.

Parameter	type	name	value
Tag value	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Host tags* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Layout	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Show problems	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Niesuppresowane; 2 - Brak.
Items (see below)			

Zaawansowana konfiguracja

Poniższe zaawansowane parametry konfiguracji są obsługiwane przez widżet *Top items*.

Parametr	type	name	value
Porządkowanie hostów			
Sortuj według	0	host_ordering_order_by	0 - *(domyślnie)* Nazwa hosta; 3 - Wartość pozycji.
Wzorce pozycji	1	host_ordering_item.0	Nazwa lub wzorzec pozycji (np. ”*: Number of processed *values per second”). Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu pulpitu dla każdego wzorca pozycji z inkrementowanym numerem w nazwie właściwości. Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagane*, jeśli *Sortuj według* jest ustawione na ”Wartość pozycji”
Kolejność	0	host_ordering_order	2 - *(domyślnie)* Górne N; 3 - Dolne N.
Limit	0	host_ordering_limit	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 10.
Porządkowanie pozycji			
Sortuj według	0	item_ordering_order_by	1 - Host; 2 - Nazwa pozycji; 3 - *(domyślnie)* Wartość pozycji.
Wzorce hostów	1	item_ordering_host.0	Nazwa lub wzorzec hosta. Uwaga: Aby skonfigurować wiele wzorców hostów, utwórz obiekt pola widżetu pulpitu dla każdego wzorca hosta z inkrementowanym numerem w nazwie właściwości. Zachowanie parametru: - *wymagane*, jeśli *Sortuj według* jest ustawione na ”Host”

Parametr	type	name	value
Kolejność	0	item_ordering_order	2 - *(domyślnie)* Górne N; 3 - Dolne N.
Limit	0	item_ordering_limit	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 10.

Kolumny

Note:
Pierwsza liczba w nazwie właściwości (np. columns.0.items.0, columns.0.item_tags_evaltype) oznacza konkretną kolumnę, natomiast druga liczba, jeśli występuje, oznacza skonfigurowaną encję (np. wzorzec pozycji, tag).

Parameter	type	name	value
Wzorce pozycji	1	columns.0.items.0	Nazwa lub wzorzec pozycji (np. ”*: Number of processed *values per second”). Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu dashboardu dla każdego wzorca pozycji, zwiększając drugą liczbę w nazwie właściwości. Podczas konfigurowania widżetu na dashboardzie szablonu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagane*
Tagi pozycji			
Typ oceny	0	columns.0.item_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	columns.0.item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Druga liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagi pozycji*
Operator	0	columns.0.item_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Druga liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagi pozycji*
Wartość tagu	1	columns.0.item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Druga liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagi pozycji*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. FF0000).

Parameter	type	name	value
Wyświetl wartość jako	0	columns.0.display_value_as	1 - *(domyślnie)* Liczbowo; 2 - Tekst.
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Jak jest; 2 - Słupek; 3 - Wskaźniki; 6 - Sparkline.
Min	1	columns.0.min	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo” i *Wyświetlanie* jest ustawione na ”Słupek” lub ”Wskaźniki”
Maks	1	columns.0.max	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo” i *Wyświetlanie* jest ustawione na ”Słupek” lub ”Wskaźniki”
Sparkline			
Szerokość	0	columns.0.sparkline.width	Możliwe wartości z zakresu 0-10. Domyślnie: 1. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline”
Wypełnienie	0	columns.0.sparkline.fill	Możliwe wartości z zakresu 0-10. Domyślnie: 3. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline”
Kolor	1	columns.0.sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `42A5F5`. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline”
Okres czasu	1	columns.0.sparkline.time_period	`DASHBOARD._timeperiod` - ustaw *selektor Okres czasu* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline”
Od	1	columns.0.sparkline.time_period.from	Prawidłowy format czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline” - *wymagane* jeśli ustawiono `columns.0.sparkline.time_period.to`

Parameter	type	name	value
Do	1	columns.0.sparkline.time_period.to	Prawidłowy zakres czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline” - *wymagane* jeśli ustawiono `columns.0.sparkline.time_period.from`
Dane historyczne	0	columns.0.sparkline.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetlanie* jest ustawione na ”Sparkline”
Progi			
Kolor	1	columns.0.thresholds.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo”
Próg	1	columns.0.thresholds.threshold	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo”
Podświetlenia			
Podświetlenie	1	columns.0.highlights.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Tekst”
Próg	1	columns.0.highlights.pattern	Dopasowane wyrażenie regularne. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Tekst”
Miejsca dziesiętne	0	columns.0.decimal_places	Możliwe wartości z zakresu 0-10. Domyślnie: 2.
Konfiguracja zaawansowana			
Funkcja agregacji	0	columns.0.aggregate_function	Jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo”: 0 - *(domyślnie)* nieużywane; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last. Jeśli *Wyświetl wartość jako* jest ustawione na ”Tekst”: 0 - *(domyślnie)* nieużywane; 4 - count; 6 - first; 7 - last.

Parameter	type	name	value
Okres czasu	1	columns.0.time_period.from	DASHBOARD._timeperiod - ustaw *selektor Okres czasu* jako źródło danych; ABCDE._timeperiod - ustaw kompatybilny widżet (z parametrem reference równym ABCDE) jako źródło danych. Domyślnie: DASHBOARD._timeperiod Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*. Zachowanie parametru: - *obsługiwane* jeśli *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last”
Od	1	columns.0.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub relatywnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwane* jeśli *Okres czasu* nie jest ustawiony i *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagane* jeśli ustawiono columns.0.time_period.to
Do	1	columns.0.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub relatywnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwane* jeśli *Okres czasu* nie jest ustawiony i *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagane* jeśli ustawiono columns.0.time_period.from
Dane historyczne	0	columns.0.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwane* jeśli *Wyświetl wartość jako* jest ustawione na ”Liczbowo”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Top items*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Top items*

Skonfiguruj widżet *Top items*, który wyświetla dane dla hosta „10084” i tylko dla pozycji, dla których tag o nazwie „component” zawiera wartość „cpu”. Dodatkowo wyświetl dane z hostami umieszczonymi u góry i użyj kolorowego paska wskaźnika do reprezentacji komórek.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "topitems",
                        "name": "Top items",
```

```
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 3,
                                "name": "hostids.0",
                                "value": 10084
                            },
                            {
                                "type": 1,
                                "name": "columns.0.items.0",
                                "value": "*"
                            },
                            {
                                "type": 1,
                                "name": "columns.0.item_tags.0.tag",
                                "value": "component"
                            },
                            {
                                "type": 0,
                                "name": "columns.0.item_tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "columns.0.item_tags.0.value",
                                "value": "cpu"
                            },
                            {
                                "type": 0,
                                "name": "columns.0.display",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "layout",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```


Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

29 Najważniejsze wyzwalacze

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Top triggers* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Top triggers*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Top triggers*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - *(domyślnie)* Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonowym dashboardzie.
Problem	1	problem	Nazwa zdarzenia problemu (bez uwzględniania wielkości liter, pełna nazwa lub jej część).

Parameter	type	name	value
Severity	0	severities.0	Ważności wyzwalaczy. 0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Problem tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Time period	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor *Time period* jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub relatywnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli `time_period.to` jest ustawiony

Parameter	type	name	value
To	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub relatywnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany* jeśli *Time period* nie jest ustawiony - *wymagany* jeśli time_period.from jest ustawiony
Trigger limit	0	show_lines	Zakres możliwych wartości: 1-100. Domyślnie: 10.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Top triggers*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Top triggers*

Skonfiguruj widżet *Top triggers*, który wyświetla 5 najważniejszych wyzwalaczy dla grupy hostów „4” wraz z liczbą wszystkich problemów dla każdego wyzwalacza. Widżet wyświetla tylko wyzwalacze o poziomach ważności „Warning”, „Average”, „High” lub „Disaster” oraz problemy, które mają tag o nazwie „scope”, zawierający wartości „performance”, „availability” lub „capacity”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "toptriggers",
                        "name": "Top triggers",
                        "x": 0,
                        "y": 0,
                        "width": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "severities.0",
                                "value": 2
                            },
                            {
                                "type": 0,
                                "name": "severities.1",
                                "value": 3
                            },
                            {
                                "type": 0,
                                "name": "severities.2",
                                "value": 4
                            },
                            {
```

```
                            "type": 0,
                            "name": "severities.3",
                            "value": 5
                        },
                        {
                            "type": 1,
                            "name": "tags.0.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.0.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.0.value",
                            "value": "performance"
                        },
                        {
                            "type": 1,
                            "name": "tags.1.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.1.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.1.value",
                            "value": "availability"
                        },
                        {
                            "type": 1,
                            "name": "tags.2.tag",
                            "value": "scope"
                        },
                        {
                            "type": 0,
                            "name": "tags.2.operator",
                            "value": 0
                        },
                        {
                            "type": 1,
                            "name": "tags.2.value",
                            "value": "capacity"
                        },
                        {
                            "type": 0,
                            "name": "show_lines",
                            "value": 5
                        }
                    ]
                }
            ]
        }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
```

```
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

30 Przegląd wyzwalaczy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Przegląd wyzwalaczy* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Przegląd wyzwalaczy*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Trigger Overview*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Show	0	show	1 - *(domyślnie)* Ostatnie problemy; 2 - Dowolne; 3 - Problemy.

Parameter	type	name	value
Host groups	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Host groups (Widget)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts	3	hostids.0	ID host. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Host groups* musi być albo w ogóle nie skonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID host: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na szablonie pulpitu.
Problem tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* Oraz/Lub; 2 - Lub.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*

Parameter	type	name	value
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagów na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Problem tags*
Show suppressed problems	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Layout	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Przegląd wyzwalaczy*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Przegląd wyzwalaczy*

Skonfiguruj widżet *Przegląd wyzwalaczy*, który wyświetla stany wyzwalaczy dla wszystkich grup hostów mających wyzwalacze z tagiem o nazwie „scope” i zawierającym wartość „availability”.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "tags.0.tag",
                                "value": "scope"
                            },
                            {
                                "type": 0,
                                "name": "tags.0.operator",
                                "value": 0
                            },
                            {
                                "type": 1,
                                "name": "tags.0.value",
                                "value": "availability"
                            }
                        ]
                    }
```

```
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

31 URL

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *URL* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *URL*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widgetu *URL*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - *(domyślnie)* Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Parameter	type	name	value
URL	1	url	Prawidłowy ciąg URL. Zachowanie parametru: - *wymagany*
Override host	1	override_hostid._reference	ABCDE._hostid - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów; DASHBOARD._hostid - ustaw selektor *Host* na dashboardzie jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na szablon dashboardu.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *URL*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widgetu *URL*

Skonfiguruj widget *URL*, który wyświetla stronę główną podręcznika Zabbix.

Request:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "url",
                                "value": "https://www.zabbix.com/documentation/7.4/en"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
```

```
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

32 Monitorowanie sieci Web

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Monitorowanie WWW* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji dashboardu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć niestandardowe widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Monitorowanie WWW*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Web monitoring*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host groups	2	groupids.0	ID host group. Uwaga: Aby skonfigurować wiele host group, utwórz obiekt pola widżetu pulpitu dla każdej host group, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfiguracji widżetu na template dashboard.
Host groups (Widget)	1	groupids._reference	Zamiast ID host group: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla host group. Ten parametr nie jest obsługiwany podczas konfiguracji widżetu na template dashboard.

Parameter	type	name	value
Exclude host groups	2	exclude_groupids.0	ID host group. Uwaga: Aby wykluczyć wiele host group, utwórz obiekt pola widżetu pulpitu dla każdej host group, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfiguracji widżetu na template dashboard.
Hosts	3	hostids.0	ID host. Uwaga: Aby skonfigurować wiele host, utwórz obiekt pola widżetu pulpitu dla każdego host, zwiększając numer w nazwie właściwości. W przypadku wielu host parametr *Host groups* musi być albo całkowicie nieskonfigurowany, albo skonfigurowany z co najmniej jedną host group, do której należą skonfigurowane host. Ten parametr nie jest obsługiwany podczas konfiguracji widżetu na template dashboard.
Hosts (Widget/Dashboard)	1	hostids._reference	Zamiast ID host: `DASHBOARD.hostids` - ustaw selektor *Host* jako źródło danych dla host; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla host. Ten parametr nie jest obsługiwany podczas konfiguracji widżetu na template dashboard.
Scenario tags			
Evaluation type	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Tag name	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfiguracji *Scenario tags*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfiguracji *Scenario tags*
Tag value	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfiguracji *Scenario tags*
Show hosts in maintenance	0	maintenance	0 - Wyłączone; 1 - *(domyślnie)* Włączone.

Parameter	type	name	value
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. ABCDE lub JBPNL). Ta wartość musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Web monitoring*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Web monitoring*

Skonfiguruj widżet *Web monitoring*, który wyświetla podsumowanie statusu aktywnych scenariuszy monitorowania WWW dla grupy hostów „4”.

Żądanie:

```
{
    "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": 18,
                        "height": 3,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 2,
                                "name": "groupids.0",
                                "value": 4
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

Raport SLA 25

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu umożliwiają konfigurację widgetu *Raport SLA* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są walidowane podczas tworzenia ani aktualizacji pulpitu. Pozwala to użytkownikom modyfikować wbudowane widgety i tworzyć własne widgety, ale jednocześnie wprowadza ryzyko nieprawidłowego tworzenia lub aktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub aktualizację widgetu *Raport SLA*, zapoznaj się z zachowaniem parametrów opisanym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Raport SLA*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - *(domyślnie)* Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
SLA	10	slaid.0	ID SLA. Zachowanie parametrów: - *wymagane*
Usługa	9	serviceid.0	ID usługi.
Pokaż okresy	0	show_periods	Możliwe wartości mieszczą się w zakresie 1-100. Domyślnie: 20.
Od	1	date_from	Prawidłowy ciąg daty w formacie `YYYY-MM-DD`. Obsługiwane są daty względne z modyfikatorami `d`, `w`, `M`, `y` (np. `now`, `now/d`, `now/w-1w` itd.).
Do	1	date_to	Prawidłowy ciąg daty w formacie `YYYY-MM-DD`. Obsługiwane są daty względne z modyfikatorami `d`, `w`, `M`, `y` (np. `now`, `now/d`, `now/w-1w` itd.).

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Raport SLA*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Raport SLA*

Skonfiguruj widżet *Raport SLA*, który wyświetla raport SLA dla usługi SLA „4”, usługi „2”, za okres ostatnich 30 dni.

Żądanie:

```
{
    "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": 36,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 10,
                                "name": "slaid.0",
                                "value": 4
                            },
                            {
                                "type": 9,
                                "name": "serviceid.0",
                                "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
            }
        ]
    },
    "id": 1
```

```
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

Pulpit szablonu

Ta klasa została zaprojektowana do pracy z pulpitami szablonów.

Odwołania do obiektów:

- Pulpit szablonu
- Strona pulpitu szablonu
    - Widget pulpitu szablonu
        * Pole widgetu pulpitu szablonu

Dostępne metody:

- templatedashboard.create - utwórz nowe pulpity szablonów
- templatedashboard.delete - usuń pulpity szablonów
- templatedashboard.get - pobierz pulpity szablonów
- templatedashboard.update - zaktualizuj pulpity szablonów

Obiekt pulpitu nawigacyjnego szablonu

Następujące obiekty są bezpośrednio powiązane z API `templateashboard`.

Pulpit szablonu

Obiekt pulpitu szablonu ma następujące właściwości.

Właściwość	Typ	Opis
dashboardid	ID	ID pulpitu szablonu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa pulpitu szablonu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
templateid	ID	ID szablonu, do którego należy pulpit. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
display_period	integer	Domyślny okres wyświetlania strony (w sekundach). Możliwe wartości: 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 30.
auto_start	integer	Automatyczne uruchamianie pokazu slajdów. Możliwe wartości: 0 - nie uruchamiaj automatycznie pokazu slajdów; 1 - *(domyślnie)* uruchamiaj automatycznie pokaz slajdów.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych pulpitów szablonów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.

Strona pulpitu szablonu

Obiekt strony pulpitu szablonu ma następujące właściwości.

Właściwość	Typ	Opis
dashboard_pageid	ID	ID strony pulpitu. Zachowanie właściwości: - *tylko do odczytu*
name	string	Nazwa strony pulpitu. Domyślnie: pusty ciąg.
display_period	integer	Okres wyświetlania strony pulpitu (w sekundach). Możliwe wartości: 0, 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 0 (zostanie użyty domyślny okres wyświetlania strony).
widgets	array	Tablica obiektów widgetu pulpitu szablonu.

Widget dashboardu szablonu

Obiekt widgetu dashboardu szablonu ma następujące właściwości.

Property	Type	Description
widgetid	ID	ID widgetu dashboardu. Property behavior: - *read-only*

Property	Type	Description
type	string	Typ widgetu dashboardu. Possible values: actionlog - Dziennik akcji; clock - Zegar; discovery - Status wykrywania; favgraphs - Ulubione wykresy; favmaps - Ulubione mapy; gauge - Wskaźnik; graph - Wykres (klasyczny); graphprototype - Prototyp wykresu; honeycomb - Honeycomb; hostavail - Dostępność hosta; hostcard - Karta hosta; hostnavigator - Nawigator hostów; itemcard - Karta pozycji; itemnavigator - Nawigator pozycji; item - Wartość pozycji; map - Mapa; navtree - Drzewo nawigacji mapy; piechart - Wykres kołowy; plaintext - Zwykły tekst; problemhosts - Hosty z problemami; problems - Problemy; problemsbysv - Problemy według ważności; slareport - Raport SLA; svggraph - Wykres; systeminfo - Informacje o systemie; tophosts - Najważniejsze hosty; topitems - Najważniejsze pozycje; toptriggers - Najważniejsze wyzwalacze; trigover - Przegląd wyzwalaczy; url - URL; web - Monitorowanie WWW. Property behavior: - *required*
name	string	Niestandardowa nazwa widgetu.
x	integer	Pozycja pozioma liczona od lewej strony dashboardu. Possible values range from 0 to 71.
y	integer	Pozycja pionowa liczona od górnej części dashboardu. Possible values range from 0 to 63.
width	integer	Szerokość widgetu. Possible values range from 1 to 72.
height	integer	Wysokość widgetu. Possible values range from 1 to 64.
view_mode	integer	Tryb widoku widgetu. Possible values: 0 - *(default)* domyślny widok widgetu; 1 - z ukrytym nagłówkiem;
fields	array	Tablica obiektów template dashboard widget field.

Pole widgetu pulpitu szablonu

Obiekt pola widgetu pulpitu szablonu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ pola widgetu. Możliwe wartości: 0 - Integer; 1 - String; 4 - Pozycja; 5 - Prototyp pozycji; 6 - Wykres; 7 - Prototyp wykresu; 8 - Mapa; 9 - Usługa; 10 - SLA; 11 - Użytkownik; 12 - Akcja; 13 - Typ nośnika. Zachowanie właściwości: - *wymagane*
name	string	Nazwa pola widgetu. Możliwe wartości: patrz Pola widgetu pulpitu. Należy pamiętać, że niektóre parametry związane z hostem (np. *Grupy hostów*, *Wyklucz grupy hostów* i *Hosty* w widżecie *Problemy*, *Grupy hostów* w widżecie *Dostępność hostów* itd.) nie są dostępne podczas konfigurowania widgetu na pulpicie szablonu. Wynika to z tego, że pulpity szablonów wyświetlają dane tylko z hosta, do którego szablon jest podłączony. Zachowanie właściwości: - *wymagane*
value	mixed	Wartość pola widgetu zależna od typu. Możliwe wartości: patrz Pola widgetu pulpitu. Należy pamiętać, że niektóre parametry związane z hostem (np. *Grupy hostów*, *Wyklucz grupy hostów* i *Hosty* w widżecie *Problemy*, *Grupy hostów* w widżecie *Dostępność hostów* itd.) nie są dostępne podczas konfigurowania widgetu na pulpicie szablonu. Wynika to z tego, że pulpity szablonów wyświetlają dane tylko z hosta, do którego szablon jest podłączony. Zachowanie właściwości: - *wymagane*

templatedashboard.create

Opis

`object templatedashboard.create(object/array templateDashboards)`

Ta metoda umożliwia tworzenie nowych pulpitów szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Pulpity szablonu do utworzenia.

Oprócz standardowych właściwości pulpitu szablonu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
pages	array	Strony pulpitu szablonu, które mają zostać utworzone dla pulpitu. Strony pulpitu będą uporządkowane w tej samej kolejności, w jakiej zostały podane. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pulpitów nawigacyjnych szablonu we właściwości `dashboardids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pulpitów nawigacyjnych szablonu.

Przykłady

Tworzenie dashboardu szablonu

Utwórz dashboard szablonu o nazwie „Graphs” z jednym widżetem Graph na pojedynczej stronie dashboardu.

Żądanie:

```
{
    "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"
                            }
                        ]
                    }
                ]

            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "32"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu szablonu
- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::create() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.delete

Opis

`object templatedashboard.delete(array templateDashboardIds)`

Ta metoda umożliwia usuwanie szablonów pulpitów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID pulpitów szablonu do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pulpitów szablonu we właściwości `dashboardids`.

Przykłady

Usuwanie wielu pulpitów szablonów

Usuń dwa pulpity szablonów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "templateashboard.delete",
"params": [
"45",
"46"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"45",
"46"
]
},
"id": 1
}
```

Źródło

CTemplateDashboard::delete() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.get

Opis

`integer/array templatedashboard.get(object parameters)`

Ta metoda umożliwia pobieranie pulpitów szablonów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dashboardids	ID/array	Zwraca tylko dashboardy szablonów o podanych identyfikatorach.
templateids	ID/array	Zwraca tylko dashboardy szablonów należące do podanych szablonów.
selectPages	query	Zwraca właściwość `pages` ze stronami dashboardu szablonu, w poprawnej kolejności.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dashboardid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie pulpitów szablonu

Pobierz wszystkie pulpity szablonu z widgetami dla określonego szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "templateids": "10001"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1125"
                                }
                            ]
                        },
                        {
                            "widgetid": "221",
                            "type": "graph",
                            "name": "",
                            "x": "12",
                            "y": "0",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1129"
                                }
                            ]
                        },
                        {
                            "widgetid": "222",
                            "type": "graph",
                            "name": "",
                            "x": "0",
                            "y": "5",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1128"
                                }
                            ]
                        },
                        {
                            "widgetid": "223",
```

```
                            "type": "graph",
                            "name": "",
                            "x": "12",
                            "y": "5",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1126"
                                }
                            ]
                        },
                        {
                            "widgetid": "224",
                            "type": "graph",
                            "name": "",
                            "x": "0",
                            "y": "10",
                            "width": "36",
                            "height": "5",
                            "view_mode": "0",
                            "fields": [
                                {
                                    "type": "6",
                                    "name": "graphid",
                                    "value": "1127"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Strona pulpitu szablonu
- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::get() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.update

Opis

`object templatedashboard.update(object/array templateDashboards)`

Ta metoda umożliwia aktualizację istniejących dashboardów szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości pulpitu nawigacyjnego szablonu do zaktualizowania.

Właściwość dashboardid musi być określona dla każdego pulpitu nawigacyjnego, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko określone właściwości.

Oprócz standardowych właściwości pulpitu nawigacyjnego szablonu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony pulpitu nawigacyjnego szablonu, które zastąpią istniejące strony pulpitu nawigacyjnego. Strony pulpitu nawigacyjnego są aktualizowane za pomocą właściwości dashboard_pageid. Nowe strony pulpitu nawigacyjnego zostaną utworzone dla obiektów bez właściwości dashboard_pageid, a istniejące strony pulpitu nawigacyjnego zostaną usunięte, jeśli nie zostaną ponownie użyte. Strony pulpitu nawigacyjnego zostaną uporządkowane w tej samej kolejności, w jakiej zostały określone. Zaktualizowane zostaną tylko określone właściwości stron pulpitu nawigacyjnego. Dla właściwości pages wymagany jest co najmniej jeden obiekt strony pulpitu nawigacyjnego.

Wartości zwracane

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych pulpitów szablonu we właściwości dashboardids.

Przykłady

Zmiana nazwy dashboardu szablonu

Zmień nazwę dashboardu szablonu na „Performance graphs”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "23",
        "name": "Performance graphs"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "23"
        ]
    },
    "id": 1
}
```

Aktualizowanie stron pulpitu nawigacyjnego szablonu

Zmień nazwę pierwszej strony pulpitu nawigacyjnego, zastąp widgety na drugiej stronie pulpitu nawigacyjnego i dodaj nową stronę jako trzecią. Usuń wszystkie pozostałe strony pulpitu nawigacyjnego.

Żądanie:

```
{
    "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": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::update() w *ui/include/classes/api/services/CTemplateDashboard.php*.

Raport

Ta klasa jest przeznaczona do pracy z zaplanowanymi raportami.

Odwołania do obiektów:

- Report
- Users
- User groups

Dostępne metody:

- report.create - tworzenie nowych zaplanowanych raportów
- report.delete - usuwanie zaplanowanych raportów
- report.get - pobieranie zaplanowanych raportów
- report.update - aktualizowanie zaplanowanych raportów

Obiekt raportu

Następujące obiekty są bezpośrednio powiązane z API `report`.

Raport

Obiekt raportu ma następujące właściwości:

Właściwość	Typ	Opis
reportid	ID	ID raportu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
userid	ID	ID użytkownika, który utworzył raport. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Unikalna nazwa raportu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
dashboardid	ID	ID pulpitu, na podstawie którego tworzony jest raport. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
period	integer	Okres, za który raport zostanie przygotowany. Możliwe wartości: 0 - *(domyślnie)* poprzedni dzień; 1 - poprzedni tydzień; 2 - poprzedni miesiąc; 3 - poprzedni rok.
cycle	integer	Harmonogram powtarzania okresu. Możliwe wartości: 0 - *(domyślnie)* codziennie; 1 - co tydzień; 2 - co miesiąc; 3 - co rok.
start_time	integer	Pora dnia, w sekundach, o której raport zostanie przygotowany do wysłania. Domyślnie: 0.
weekdays	integer	Dni tygodnia do wysyłania raportu. Możliwe wartości bitmapy: 1 - poniedziałek; 2 - wtorek; 4 - środa; 8 - czwartek; 16 - piątek; 32 - sobota; 64 - niedziela. To pole jest maską bitową; akceptowana jest dowolna suma możliwych wartości bitmapy (na przykład 21 dla poniedziałku, środy i piątku). Domyślnie: 0. Zachowanie właściwości: - *wymagane*, jeśli `cycle` jest ustawione na ”weekly”.
active_since	string	Data rozpoczęcia. Możliwe wartości: pusty ciąg - *(domyślnie)* nie określono (zapisywane jako 0); określona data w formacie YYYY-MM-DD (zapisywana jako znacznik czasu początku dnia (00:00:00)).

Właściwość	Typ	Opis
active_till	string	Data zakończenia. Możliwe wartości: pusty ciąg - *(domyślnie)* nie określono (zapisywane jako 0); określona data w formacie YYYY-MM-DD (zapisywana jako znacznik czasu końca dnia (23:59:59)).
subject	string	Temat wiadomości raportu.
message	string	Treść wiadomości raportu.
status	integer	Czy raport jest włączony, czy wyłączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
description	text	Opis raportu.
state	integer	Stan raportu. Możliwe wartości: 0 - *(domyślnie)* raport nie został jeszcze przetworzony; 1 - raport został wygenerowany i pomyślnie wysłany do wszystkich odbiorców; 2 - generowanie raportu nie powiodło się; ”info” zawiera informacje o błędzie; 3 - raport został wygenerowany, ale wysyłka do części (lub wszystkich) odbiorców nie powiodła się; ”info” zawiera informacje o błędzie. Zachowanie właściwości: - *tylko do odczytu*
lastsent	timestamp	Znacznik czasu Unix ostatniego pomyślnie wysłanego raportu. Zachowanie właściwości: - *tylko do odczytu*
info	string	Opis błędu lub dodatkowe informacje. Zachowanie właściwości: - *tylko do odczytu*

Użytkownicy

Obiekt users ma następujące właściwości:

Właściwość	Typ	Opis
userid	ID	ID użytkownika, do którego ma zostać wysłany raport. Zachowanie właściwości: - *wymagane*
access_userid	ID	ID użytkownika, w imieniu którego raport zostanie wygenerowany. 0 - *(domyślnie)* Wygeneruj raport przez odbiorcę.
exclude	integer	Czy wykluczyć użytkownika z listy mailingowej. Możliwe wartości: 0 - *(domyślnie)* Uwzględnij; 1 - Wyklucz.

Grupy użytkowników

Obiekt grup użytkowników ma następujące właściwości:

Właściwość	Typ	Opis
usrgrpid	ID	ID grupy użytkowników, do której ma zostać wysłany raport. Zachowanie właściwości: - *wymagane*
access_userid	ID	ID użytkownika, w imieniu którego zostanie wygenerowany raport. 0 - *(domyślnie)* Wygeneruj raport przez odbiorcę.

report.create

Opis

`object report.create(object/array reports)`

Ta metoda umożliwia tworzenie nowych zaplanowanych raportów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Zaplanowane raporty do utworzenia.

Oprócz standardowych właściwości zaplanowanego raportu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
users	object/array	Użytkownicy, do których ma zostać wysłany raport. Zachowanie parametrów: - *wymagane*, jeśli `user_groups` nie jest ustawione
user_groups	object/array	Grupy użytkowników, do których ma zostać wysłany raport. Zachowanie parametrów: - *wymagane*, jeśli `users` nie jest ustawione

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych zaplanowanych raportów we właściwości `reportids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych zaplanowanych raportów.

Przykłady

Tworzenie zaplanowanego raportu

Utwórz raport tygodniowy, który będzie przygotowywany za poprzedni tydzień w każdy poniedziałek-piątek o 12:00 od 2021-04-01 do 2021-08-31.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.create",
    "params": {
        "userid": "1",
        "name": "Raport tygodniowy",
        "dashboardid": "1",
        "period": "1",
        "cycle": "1",
        "start_time": "43200",
        "weekdays": "31",
        "active_since": "2021-04-01",
        "active_till": "2021-08-31",
```

```
        "subject": "Raport tygodniowy",
        "message": "Tekst towarzyszący raportowi",
        "status": "1",
        "description": "Opis raportu",
        "users": [
            {
                "userid": "1",
                "access_userid": "1",
                "exclude": "0"
            },
            {
                "userid": "2",
                "access_userid": "0",
                "exclude": "1"
            }
        ],
        "user_groups": [
            {
                "usrgrpid": "7",
                "access_userid": "0"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::create() w *ui/include/classes/api/services/CReport.php*.

report.delete

Opis

`object report.delete(array reportids)`

Ta metoda umożliwia usuwanie zaplanowanych raportów.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID raportów okresowych do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych zaplanowanych raportów we właściwości `reportids`.

Przykłady

Usuwanie wielu zaplanowanych raportów

Usuń dwa zaplanowane raporty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CReport::delete() w *ui/include/classes/api/services/CReport.php*.

report.get

Opis

`integer/array report.get(object parameters)`

Ta metoda umożliwia pobieranie zaplanowanych raportów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
reportids	ID/array	Zwraca tylko zaplanowane raporty o podanych identyfikatorach raportów.
expired	boolean	Jeśli ustawiono na `true`, zwraca tylko wygasłe zaplanowane raporty.
selectUsers	query	Zwraca właściwość `users` zawierającą użytkowników, do których skonfigurowano wysyłanie raportu.
selectUserGroups	query	Zwraca właściwość `user_groups` zawierającą grupy użytkowników, do których skonfigurowano wysyłanie raportu.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `reportid`, `name`, `status`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	

Parametr	Typ	Opis
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych raportu

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.get",
    "params": [
        "output": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "reportids": ["1", "2"]
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::get() w *ui/include/classes/api/services/CReport.php*.

report.update

Opis

`object report.update(object/array reports)`

Ta metoda umożliwia aktualizację istniejących zaplanowanych raportów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości zaplanowanego raportu do zaktualizowania.

Właściwość `reportid` musi być zdefiniowana dla każdego zaplanowanego raportu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości zaplanowanego raportu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
users	object/array	Użytkownicy, którzy mają zastąpić bieżących użytkowników przypisanych do zaplanowanego raportu. Zachowanie parametru: - *wymagany*, jeśli `user_groups` nie jest ustawione
user_groups	object/array	Grupy użytkowników, które mają zastąpić bieżące grupy użytkowników przypisane do zaplanowanego raportu. Zachowanie parametru: - *wymagany*, jeśli `users` nie jest ustawione

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych zaplanowanych raportów we właściwości `reportids`.

Przykłady

Wyłączanie zaplanowanego raportu

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.update",
    "params": {
        "reportid": "1",
        "status": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::update() w *ui/include/classes/api/services/CReport.php*.

Reguła LLD

Ta klasa jest przeznaczona do pracy z regułami wykrywania niskiego poziomu.

Odwołania do obiektów:

- Reguła LLD

    - Nagłówek HTTP
    - Pole zapytania HTTP
- Filtr reguły LLD
    - Warunek filtra reguły LLD
- Ścieżka makra LLD
- Przetwarzanie wstępne reguły LLD
- Nadpisania reguły LLD
    - Filtr nadpisania reguły LLD
        * Warunek filtra nadpisania reguły LLD
    - Operacja nadpisania reguły LLD
        * Status operacji nadpisania reguły LLD
        * Wykrywanie operacji nadpisania reguły LLD
        * Okres operacji nadpisania reguły LLD
        * Historia operacji nadpisania reguły LLD
        * Trendy operacji nadpisania reguły LLD
        * Poziom ważności operacji nadpisania reguły LLD
        * Tag operacji nadpisania reguły LLD
        * Szablon operacji nadpisania reguły LLD
        * Inwentaryzacja operacji nadpisania reguły LLD

Dostępne metody:

- discoveryrule.create - utwórz nowe reguły LLD
- discoveryrule.delete - usuń reguły LLD
- discoveryrule.get - pobierz reguły LLD
- discoveryrule.update - zaktualizuj reguły LLD

Obiekt reguły LLD (Low Level Discovery)

Następujące obiekty są bezpośrednio powiązane z API `discoveryrule`.

Reguła LLD

Obiekt reguły wykrywania niskiego poziomu ma następujące właściwości.

Właściwość	Typ	Opis
itemid	ID	ID reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
delay	string	Interwał aktualizacji reguły LLD. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej niestandardowych interwałów, wszystkie oddzielone średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowanych. Akceptuje makra użytkownika. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra zmieszane z tekstem nie są obsługiwane. Interwały elastyczne mogą być zapisane jako dwa makra oddzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), lub jeśli `type` jest ustawione na ”Zabbix agent (active)” (7) i `key_` nie zawiera ”mqtt.get”

Właściwość	Typ	Opis
hostid	ID	ID hosta, do którego należy reguła LLD. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie reguły wykrywania. Możliwe wartości: 1 - reguła wykrywania niskiego poziomu; 5 - reguła wykrywania niskiego poziomu przekonwertowana z prototypu. Zachowanie właściwości: - *tylko do odczytu*
interfaceid	ID	ID interfejsu hosta reguły LLD. Zachowanie właściwości: - *wymagane*, jeśli reguła LLD należy do hosta i `type` jest ustawione na ”Zabbix agent”, ”IPMI agent”, ”JMX agent” lub ”SNMP agent” - *obsługiwane*, jeśli reguła LLD należy do hosta i `type` jest ustawione na ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent”
key_	string	Klucz reguły LLD. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Nazwa reguły LLD. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
type	integer	Typ reguły LLD. Możliwe wartości: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Simple check; 5 - Zabbix internal; 7 - Zabbix agent (active); 10 - External check; 11 - Database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 16 - JMX agent; 18 - pozycja zależna; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 23 - Zagnieżdżony. Ten typ jest dozwolony tylko wtedy, gdy nadrzędna reguła jest szablonem (zakłada się, że jest podłączony do prototypu hosta podczas wykrywania) lub wykrytym hostem (który ma regułę LLD w swoim korzeniu). Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
url	string	Ciąg URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Pozwala wypełniać wartość podobnie jak w pozycji trapper. Możliwe wartości: 0 - *(domyślnie)* Nie zezwalaj na przyjmowanie danych przychodzących; 1 - Zezwalaj na przyjmowanie danych przychodzących. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent”
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` jest ustawione na ”SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` jest ustawione na ”HTTP agent”: 0 - *(domyślnie)* brak; 1 - basic; 2 - NTLM; 3 - Kerberos; 4 - Digest. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”SSH agent” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
description	string	Opis reguły LLD.
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości reguły LLD. Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Podążanie za przekierowaniami odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - Nie podążaj za przekierowaniami; 1 - *(domyślnie)* Podążaj za przekierowaniami. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
http_proxy	string	Ciąg połączenia proxy HTTP(S). Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”IPMI agent” i `key_` nie jest ustawione na ”ipmi.get” - *obsługiwane*, jeśli `type` jest ustawione na ”IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych
jmx_endpoint	string	Niestandardowy ciąg połączenia JMX agent. Domyślnie: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”JMX agent”
lifetime	string	Okres czasu, po którym pozycje, które nie są już wykrywane, zostaną usunięte. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Domyślnie: `7d`.
lifetime_type	integer	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: 0 - *(domyślnie)* Usuń po osiągnięciu progu lifetime; 1 - Nie usuwaj; 2 - Usuń natychmiast.
enabled_lifetime	string	Okres czasu, po którym pozycje, które nie są już wykrywane, zostaną wyłączone. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Domyślnie: `0`.
enabled_lifetime_type	integer	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: 0 - Wyłącz po osiągnięciu progu lifetime; 1 - Nie wyłączaj; 2 - *(domyślnie)* Wyłącz natychmiast.
master_itemid	ID	ID pozycji nadrzędnej. Reguła wykrywania nie może być pozycją nadrzędną dla innej reguły wykrywania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do JSON. Możliwe wartości: 0 - *(domyślnie)* Przechowuj surowe dane; 1 - Konwertuj do JSON. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
params	string	Dodatkowe parametry zależne od typu reguły LLD: - wykonywany skrypt dla reguł LLD SSH i Telnet; - zapytanie SQL dla reguł LLD monitora bazy danych; - formuła dla obliczanych reguł LLD; - skrypt dla reguł LLD typu script i browser. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”Script” lub ”Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` jest ustawione na ”Script” lub ”Browser”. Tablica obiektów z właściwościami `name` i `value`, gdzie `name` musi być unikalne. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Script” lub ”Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”JMX agent” i `username` jest ustawione - *obsługiwane*, jeśli `type` jest ustawione na ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
post_type	integer	Typ treści danych POST przechowywanych we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* Surowe dane; 2 - Dane JSON; 3 - Dane XML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”HTTP agent” i `post_type` jest ustawione na ”JSON data” lub ”XML data” - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” i `post_type` jest ustawione na ”Raw data” - *tylko do odczytu* dla obiektów dziedziczonych
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent” i `authtype` jest ustawione na ”public key”
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent” i `authtype` jest ustawione na ”public key”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Która część odpowiedzi powinna zostać zapisana. Możliwe wartości, jeśli `request_method` jest ustawione na ”GET”, ”POST” lub ”PUT”: 0 - *(domyślnie)* Treść; 1 - Nagłówki; 2 - Zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` jest ustawione na ”HEAD”: 1 - Nagłówki. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
state	integer	Stan reguły LLD. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - nieobsługiwane. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Status reguły LLD. Możliwe wartości: 0 - *(domyślnie)* włączona reguła LLD; 1 - wyłączona reguła LLD.

Właściwość	Typ	Opis
status_codes	string	Zakresy wymaganych kodów statusu HTTP, oddzielone przecinkami. Obsługuje również makra użytkownika jako część listy rozdzielanej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	ID nadrzędnej reguły LLD szablonu. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-600s. Domyślnie: ”” - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Zabbix agent” (0), ”Simple check” (3) i `key_` nie zaczyna się od ”`vmware.`” ani ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) i `snmp_oid` zaczyna się od ”`walk[`” lub ”`get[`”, ”Script” (21), ”Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Zabbix trapper”, lub jeśli `type` jest ustawione na ”HTTP agent” i `allow_traps` jest ustawione na ”Allow to accept incoming data”
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH agent”, ”TELNET agent”, lub jeśli `type` jest ustawione na ”JMX agent” i `password` jest ustawione - *obsługiwane*, jeśli `type` jest ustawione na ”Simple check”, ”Database monitor” lub ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` jest ustawione na ”HTTP agent”)
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych reguł LLD z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli reguła LLD należy do szablonu
verify_host	integer	Czy weryfikować, że nazwa hosta dla połączenia odpowiada nazwie w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* Nie weryfikuj; 1 - Weryfikuj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
verify_peer	integer	Czy weryfikować, że certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* Nie weryfikuj; 1 - Weryfikuj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Filtr reguły LLD

Obiekt filtra reguły LLD definiuje zestaw warunków, które można wykorzystać do filtrowania wykrytych obiektów. Ma on następujące właściwości:

Property	Type	Description
conditions	object/array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia we wzorze. Zachowanie właściwości: - *required*

Property	Type	Description
evaltype	integer	Metoda oceny warunków filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *required*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra. Wyrażenie zawiera identyfikatory odwołujące się do konkretnych warunków filtra za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *read-only*
formula	string	Zdefiniowane przez użytkownika wyrażenie używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do konkretnych warunków filtra za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *required* jeśli `evaltype` jest ustawione na ”custom expression”

Warunek filtra reguły LLD

Obiekt warunku filtra reguły LLD definiuje osobne sprawdzenie wykonywane na wartości makra LLD. Ma on następujące właściwości:

Property	Type	Description
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Property behavior: - *required*
value	string	Wartość do porównania. Property behavior: - *required* if `operator` is set to ”matches regular expression” or ”does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale przy późniejszym pobieraniu zostanie wygenerowany ponownie. Property behavior: - *required* if `evaltype` of LLD rule filter object is set to ”custom expression”
operator	integer	Operator warunku. Possible values: 8 - *(default)* matches regular expression; 9 - does not match regular expression; 12 - exists; 13 - does not exist.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod discoveryrule.get i discoveryrule.create.

Ścieżka makra LLD

Ścieżka makra LLD ma następujące właściwości:

Właściwość	Type	Opis
lld_macro	string	Makro LLD. Zachowanie właściwości: - *wymagane*
path	string	Selektor wartości, która zostanie przypisana do odpowiadającego jej makra. Zachowanie właściwości: - *wymagane*

Preprocessing reguły LLD

Obiekt preprocessing reguły LLD ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ opcji preprocessing. Możliwe wartości: 5 - Wyrażenie regularne; 11 - XML XPath; 12 - JSONPath; 14 - Dopasowanie wyrażenia regularnego; 15 - Niedopasowanie wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 20 - Odrzuć niezmienione z heartbeat; 21 - JavaScript; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąp; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Zachowanie właściwości: - *wymagane*
params	string	Dodatkowe parametry używane przez opcję preprocessing. Wiele parametrów jest oddzielonych znakiem nowej linii (\n). Zachowanie właściwości: - *wymagane* jeśli `type` ma wartość ”Regular expression” (5), ”XML XPath” (11), ”JSONPath” (12), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Discard unchanged with heartbeat” (20), ”JavaScript” (21), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Replace” (25), ”SNMP walk value” (28), ”SNMP walk to JSON” (29) lub ”SNMP get value” (30)

Właściwość	Typ	Opis
error_handler	integer	Typ akcji używany w przypadku niepowodzenia kroku preprocessing. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzuć wartość; 2 - Ustaw własną wartość; 3 - Ustaw własny komunikat błędu. Zachowanie właściwości: - *wymagane* jeśli `type` ma wartość ”Regular expression” (5), ”XML XPath” (11), ”JSONPath” (12), ”Matches regular expression” (14), ”Does not match regular expression” (15), ”Check for error in JSON” (16), ”Check for error in XML” (17), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”XML to JSON” (27), ”SNMP walk value” (28), ”SNMP walk to JSON” (29) lub ”SNMP get value” (30)
error_handler_params	string	Parametry obsługi błędów. Zachowanie właściwości: - *wymagane* jeśli `error_handler` ma wartość ”Set custom value” lub ”Set custom error message”

Następujące parametry i obsługi błędów są obsługiwane dla każdego typu preprocessing.

Typ preprocessing	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane obsługi błędów
5	Wyrażenie regularne	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
14	Dopasowanie wyrażenia regularnego	pattern[1]			0, 1, 2, 3
15	Niedopasowanie wyrażenia regularnego	pattern[1]			0, 1, 2, 3
16	Sprawdzenie błędu w JSON	path[3]			0, 1, 2, 3
17	Sprawdzenie błędu w XML	path[3]			0, 1, 2, 3
20	Odrzuć niezmienione z heartbeat	seconds^4, 5^			
21	JavaScript	script[2]			
23	Prometheus do JSON	pattern^5, 6^			0, 1, 2, 3
24	CSV do JSON	character[2]	character[2]	0,1	0, 1, 2, 3

Typ preprocessing	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane obsługi błędów
25	Zastąp	search string[2]	replacement[2]		
27	XML do JSON				0, 1, 2, 3
28	Wartość SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3
29	SNMP walk do JSON[7]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Wartość SNMP get	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] wyrażenie regularne
 [2] ciąg znaków
 [3] JSONPath lub XML XPath
 [4] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [5] makro użytkownika
 [6] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy element wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [7] Obsługuje wiele rekordów "Field name,OID prefix,Format records" oddzielonych znakiem nowej linii.

Nadpisy reguł LLD

Obiekt nadpisywania reguł LLD definiuje zestaw reguł (filtry, warunki i operacje), które są używane do nadpisywania właściwości różnych obiektów prototypów. Ma on następujące właściwości:

Property	Type	Description
name	string	Unikalna nazwa nadpisania. Property behavior: - *required*
step	integer	Unikalny numer kolejności nadpisania. Property behavior: - *required*
stop	integer	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli nastąpi dopasowanie. Possible values: 0 - *(default)* nie zatrzymuj przetwarzania nadpisań; 1 - zatrzymaj przetwarzanie nadpisań, jeśli filtr pasuje.
filter	object	Filtr nadpisania.
operations	object/array	Operacje nadpisania.

Filtr nadpisywania reguły LLD

Obiekt filtra nadpisywania reguły LLD definiuje zestaw warunków, które jeśli dopasują wykryty obiekt, zostanie zastosowane nadpisanie. Ma on następujące właściwości:

Property	Type	Description
conditions	object/array	Zestaw warunków filtra nadpisywania używanych do dopasowywania wykrytych obiektów. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia we wzorze. Zachowanie właściwości: - *required*
evaltype	integer	Metoda oceny warunków filtra nadpisywania. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *required*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra nadpisywania. Wyrażenie zawiera identyfikatory odwołujące się do konkretnych warunków filtra nadpisywania za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *read-only*
formula	string	Zdefiniowane przez użytkownika wyrażenie używane do oceny warunków filtrów nadpisywania z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do konkretnych warunków filtra nadpisywania za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra nadpisywania: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *required* jeśli `evaltype` jest ustawione na ”custom expression”

Warunek filtra nadpisania reguły LLD

Obiekt warunku filtra nadpisania reguły LLD definiuje osobne sprawdzenie wykonywane na wartości makra LLD. Ma on następujące właściwości:

Property	Type	Description
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Property behavior: - *required*
value	string	Wartość do porównania. Property behavior: - *required* jeśli `operator` jest ustawiony na ”matches regular expression” lub ”does not match regular expression”

Property	Type	Description
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym pobraniu. Property behavior: - *required* jeśli `evaltype` obiektu LLD rule override filter object jest ustawiony na ”custom expression”
operator	integer	Operator warunku. Possible values: 8 - *(default)* matches regular expression; 9 - does not match regular expression; 12 - exists; 13 - does not exist.

Operacja nadpisania reguły LLD

Operacja nadpisania reguły LLD jest kombinacją warunków i akcji wykonywanych na obiekcie prototypu. Ma następujące właściwości:

Property	Type	Description
operationobject	integer	Typ wykrytego obiektu, na którym ma zostać wykonana akcja. Possible values: 0 - Prototyp pozycji; 1 - Prototyp wyzwalacza; 2 - Prototyp wykresu; 3 - Prototyp hosta; 4 - Prototyp wykrywania. Property behavior: - *required*
operator	integer	Operator warunku nadpisania. Possible values: 0 - *(default)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 8 - pasuje do; 9 - nie pasuje do.
value	string	Wzorzec dopasowania nazwy pozycji, wyzwalacza, wykresu lub prototypu hosta, zależnie od wybranego obiektu.
opstatus	object	Obiekt statusu operacji nadpisania dla obiektów typu pozycja, wyzwalacz i prototyp hosta.
opdiscover	object	Obiekt statusu wykrywania operacji nadpisania (wszystkie typy obiektów).
opperiod	object	Obiekt okresu operacji nadpisania (interwału aktualizacji) dla obiektu typu prototyp pozycji.
ophistory	object	Obiekt historii operacji nadpisania dla obiektu typu prototyp pozycji.
optrends	object	Obiekt trendów operacji nadpisania dla obiektu typu prototyp pozycji.
opseverity	object	Obiekt ważności operacji nadpisania dla obiektu typu prototyp wyzwalacza.
optag	object/array	Obiekt tagu operacji nadpisania dla obiektów typu wyzwalacz i prototyp hosta.
optemplate	object/array	Obiekt szablonu operacji nadpisania dla obiektu typu prototyp hosta.
opinventory	object	Obiekt inwentarza operacji nadpisania dla obiektu typu prototyp hosta.

Status operacji nadpisania reguły LLD

Status operacji nadpisania reguły LLD ustawiony dla wykrytego obiektu. Ma on następujące właściwości:

Property	Type	Description
status	integer	Nadpisz status dla wybranego obiektu. Możliwe wartości: 0 - Utwórz włączone; 1 - Utwórz wyłączone. Zachowanie właściwości: - *required*

Operacja nadpisania reguły LLD discover

Operacja nadpisania reguły LLD discover status ustawiony na discovered obiekt. Ma następujące właściwości:

Property	Type	Description
discover	integer	Nadpisz status discover dla wybranego obiektu. Możliwe wartości: 0 - Tak, kontynuuj wykrywanie obiektów; 1 - Nie, nowe obiekty nie będą wykrywane, a istniejące zostaną oznaczone jako utracone. Property behavior: - *required*

Okres operacji nadpisania reguły LLD

Okres operacji nadpisania reguły LLD jest wartością interwału aktualizacji ustawianą dla odkrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
delay	string	Nadpisuje interwał aktualizacji prototypu pozycji. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) oraz opcjonalnie jeden lub więcej interwałów niestandardowych, wszystkie oddzielone średnikami. Interwały niestandardowe mogą stanowić mieszankę interwałów elastycznych i harmonogramowanych. Akceptuje makra użytkownika lub makra LLD. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra zmieszane z tekstem nie są obsługiwane. Interwały elastyczne mogą być zapisane jako dwa makra oddzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *required*

Historia operacji nadpisania reguły LLD

Wartość historii operacji nadpisania reguły LLD ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Property	Type	Description
history	string	Nadpisuje historię prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane historyczne mają być przechowywane. Akceptuje również makro użytkownika i makro LLD. Property behavior: - *required*

Trendy operacji nadpisania reguły LLD

Wartość trendów operacji nadpisania reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
trends	string	Nadpisuje trendy prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane trendów mają być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *required*

Nadrzędna operacja nadpisania poziomu ważności reguły LLD

Wartość poziomu ważności operacji nadpisania reguły LLD, która jest ustawiana dla wykrytego wyzwalacza. Ma następujące właściwości:

Property	Type	Description
severity	integer	Nadpisuje poziom ważności prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* nie sklasyfikowano; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa. Property behavior: - *required*

Znacznik operacji nadpisania reguły LLD

Obiekt znacznika operacji nadpisania reguły LLD zawiera nazwę znacznika i wartość, które są ustawiane dla wykrytego obiektu. Ma on następujące właściwości:

Właściwość	Typ	Opis
tag	string	Nowa nazwa znacznika. Zachowanie właściwości: - *wymagane*
value	string	Nowa wartość znacznika.

Szablon operacji nadpisania reguły LLD

Obiekt szablonu operacji nadpisania reguły LLD, który jest powiązany z wykrytym hostem. Ma następujące właściwości:

Property	Type	Description
templateid	ID	Nadpisuje szablon połączonych szablonów prototypu hosta. Property behavior: - *required*

Inwentarz operacji nadpisania reguły LLD

Wartość trybu inwentarza operacji nadpisania reguły LLD, która jest ustawiana dla wykrytego hosta. Ma ona następujące właściwości:

Property	Type	Description
inventory_mode	integer	Nadpisuje tryb inwentarza prototypu hosta. Możliwe wartości: -1 - wyłączone; 0 - *(domyślnie)* ręczny; 1 - automatyczny. Zachowanie właściwości: - *wymagane*

discoveryrule.create

Opis

`object discoveryrule.create(object/array lldRules)`

Ta metoda umożliwia tworzenie nowych reguł LLD.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Reguły LLD do utworzenia.

Oprócz standardowych właściwości reguły LLD, metoda akceptuje następujące parametry.

Parametr	Type	Opis
filter	object	Filtr reguły LLD dla reguły LLD.
preprocessing	object/array	Opcje przetwarzania wstępnego reguły LLD.
lld_macro_paths	object/array	Opcje lld_macro_path reguły LLD.
overrides	object/array	Opcje nadpisań reguły LLD.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych reguł LLD we właściwości `itemids`. Kolejność zwróconych identyfikatorów ID jest zgodna z kolejnością przekazanych reguł LLD.

Przykłady

Tworzenie reguły LLD

Utwórz regułę LLD agenta Zabbix, aby wykrywać zamontowane systemy plików. Wykryte pozycje będą aktualizowane co 30 sekund.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
```

```
        "name": "Mounted filesystem discovery",
        "key_": "vfs.fs.discovery",
        "hostid": "10197",
        "type": 0,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Używanie filtra

Utwórz regułę LLD z zestawem warunków, według których będą filtrowane wyniki. Warunki zostaną zgrupowane przy użyciu operatora logicznego „and”.

Żądanie:

```
{
    "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"
                }
            ]
        }
    },
    "id": 1
```

```
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Tworzenie reguły LLD ze ścieżkami makr

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Reguła LLD ze ścieżkami makr LLD",
        "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"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Użycie niestandardowego filtra wyrażeń

Utwórz regułę LLD z filtrem, który będzie używać niestandardowego wyrażenia do oceny warunków. Reguła LLD ma wykrywać tylko obiekty, których wartość makra "{#MACRO1}" pasuje jednocześnie do obu wyrażeń regularnych "regex1" i "regex2", a wartość "{#MACRO2}" pasuje do "regex3" lub "regex4". Identyfikatory formuł "A", "B", "C" i "D" zostały wybrane arbitralnie.

Żądanie:

```
{
    "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"
                }
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Używanie niestandardowych pól zapytania i nagłówków

Utwórz regułę LLD z niestandardowymi polami zapytania i nagłówkami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "hostid": "10257",
        "interfaceid": "5",
        "type": 19,
        "name": "API HTTP agent",
        "key_": "api_discovery_rule",
        "delay": "5s",
        "url": "http://127.0.0.1?discoverer.php",
        "query_fields": [
```

```
            {
                "name": "mode",
                "value": "json"
            },
            {
                "name": "elements",
                "value": "2"
            }
        ],
        "headers": [
            {
                "name": "X-Type",
                "value": "api"
            },
            {
                "name": "Authorization",
                "value": "Bearer mF_A.B5f-2.1JcM"
            }
        ],
        "allow_traps": 1,
        "trapper_hosts": "127.0.0.1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Tworzenie reguły LLD z preprocessingiem

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Reguła wykrywania z preprocessingiem",
        "key_": "lld.with.preprocessing",
        "hostid": "10001",
        "ruleid": "27665",
        "type": 0,
        "delay": "60s",
        "interfaceid": "1155",
        "preprocessing": [
            {
                "type": 20,
                "params": "20",
                "error_handler": 0,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Tworzenie reguły LLD z nadpisaniami

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Discover database host",
        "key_": "lld.with.overrides",
        "hostid": "10001",
        "type": 0,
        "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"
                            }
                        ]
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "30980"
        ]
    },
    "id": 1
}
```

Utwórz regułę LLD skryptu

Utwórz proste zbieranie danych za pomocą reguły LLD skryptu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
```

```
        "name": "Script example",
        "key_": "custom.script.lldrule",
        "hostid": "12345",
        "type": 21,
        "params": "var request = new HttpRequest();\nreturn request.post(\"https://postman-echo.com/post\"
        "parameters": [{
            "name": "host",
            "value": "{HOST.CONN}"
        }],
        "timeout": "6s",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Utwórz regułę LLD z określonym okresem czasu wyłączania i bez usuwania

Utwórz regułę LLD z niestandardowym okresem czasu wyłączania encji po tym, jak nie jest już wykrywana, z ustawieniem, że nigdy nie zostanie usunięta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "lld disable after 1h",
        "key_": "lld.disable",
        "hostid": "10001",
        "type": 2,
        "lifetime_type": 1,
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "1h"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr reguły LLD
- Ścieżki makr LLD
- Wstępne przetwarzanie reguły LLD

Źródło

CDiscoveryRule::create() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.delete

Opis

`object discoveryrule.delete(array lldRuleIds)`

Ta metoda pozwala na usuwanie reguł LLD.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` Identyfikatory IDs reguł LLD do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych reguł LLD we właściwości `ruleids`.

Przykłady

Usuwanie wielu reguł LLD

Usuń dwie reguły LLD.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "discoveryrule.delete",
"params": [
"27665",
"27668"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"ruleids": [
"27665",
"27668"
]
},
"id": 1
}
```

Źródło

CDiscoveryRule::delete() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.get

Opis

`integer/array discoveryrule.get(object parameters)`

Metoda umożliwia pobieranie reguł LLD zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
itemids	ID/array	Zwraca tylko reguły LLD o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko reguły LLD należące do hostów z podanych grup.
hostids	ID/array	Zwraca tylko reguły LLD należące do podanych hostów.
inherited	boolean	Jeśli ustawiono `true`, zwraca tylko reguły LLD odziedziczone z szablonu.
interfaceids	ID/array	Zwraca tylko reguły LLD używające podanych interfejsów hosta.
monitored	boolean	Jeśli ustawiono `true`, zwraca tylko włączone reguły LLD należące do monitorowanych hostów.
templated	boolean	Jeśli ustawiono `true`, zwraca tylko reguły LLD należące do szablonów.
templateids	ID/array	Zwraca tylko reguły LLD należące do podanych szablonów.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania reguły LLD. Obiekt wykrywania reguły LLD łączy wykrytą regułę LLD z prototypem reguły LLD, z którego została wykryta.
selectDiscoveryRulePrototypes	query	Zwraca właściwość `discoveryRulePrototypes` z prototypami reguł LLD należącymi do reguły LLD. Obsługuje `count`.
selectFilter	query	Zwraca właściwość `filter` z danymi filtra używanego przez regułę LLD.
selectGraphs	query	Zwraca właściwość `graphs` z prototypami wykresów należącymi do reguły LLD. Obsługuje `count`.
selectHostPrototypes	query	Zwraca właściwość `hostPrototypes` z prototypami hostów należącymi do reguły LLD. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy reguła LLD.
selectItems	query	Zwraca właściwość `items` z prototypami pozycji należącymi do reguły LLD. Obsługuje `count`.
selectTriggers	query	Zwraca właściwość `triggers` z prototypami wyzwalaczy należącymi do reguły LLD. Obsługuje `count`.
selectLLDMacroPaths	query	Zwraca właściwość `lld_macro_paths` z listą makr LLD i ścieżek do wartości przypisanych do każdego odpowiadającego makra.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami przetwarzania wstępnego reguły LLD.
selectOverrides	query	Zwraca właściwość `lld_rule_overrides` z listą filtrów nadpisywania, warunków i operacji wykonywanych na obiektach prototypów.

Parametr	Typ	Opis
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy reguła LLD.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectItems`, `selectGraphs`, `selectTriggers`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie reguł wykrywania z hosta

Pobierz wszystkie reguły wykrywania dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": "extend",
        "hostids": "10202"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "",
        "flags": "1",
        "interfaceid": "119",
        "description": "Discovery of network interfaces as defined in global regular expression \"Netw
        "lifetime": "30d",
        "jmx_endpoint": "",
        "master_itemid": "0",
        "timeout": "",
        "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": "",
        "lifetime_type": "0",
        "enabled_lifetime_type": "2",
        "enabled_lifetime": "0",
        "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": "",
            "flags": "1",
            "interfaceid": "119",
            "description": "Discovery of file systems of different types as defined in global regular expr
            "lifetime": "30d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "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": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "parameters": []
        }
    ],
    "id": 1
}
```


Pobieranie warunków filtra

Pobierz nazwę reguły LLD ”24681” oraz jej warunki filtra. Filtr używa typu oceny ”and”, więc właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "selectFilter": "extend",
        "itemids": ["24681"]
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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
}
```

Pobierz regułę LLD według URL

Pobiera regułę LLD dla hosta na podstawie wartości pola URL reguły. Obsługiwane jest tylko dokładne dopasowanie ciągu URL zdefiniowanego dla reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "hostids": "10257",
        "filter": {
            "type": 19,
            "url": "http://127.0.0.1/discoverer.php"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "",
            "flags": "1",
            "interfaceid": "5",
            "description": "",
            "lifetime": "30d",
            "jmx_endpoint": "",
            "master_itemid": "0",
            "timeout": "",
            "url": "http://127.0.0.1/discoverer.php",
            "query_fields": [
                {
                    "name": "mode",
                    "value": "json"
                },
                {
                    "name": "elements",
                    "value": "2"
                }
            ],
            "posts": "",
            "status_codes": "200",
            "follow_redirects": "1",
            "post_type": "0",
            "http_proxy": "",
            "headers": [
                {
                    "name" : "X-Type",
                    "value": "api"
                },
                {
                    "name": "Authorization",
                    "value": "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": "",
            "lifetime_type": "0",
            "enabled_lifetime_type": "2",
            "enabled_lifetime": "0",
            "state": "0",
            "error": "",
            "parameters": []
        }
    ],
```

```
    "id": 1
}
```

Pobieranie reguły LLD z nadpisaniami

Pobierz jedną regułę LLD, która ma różne ustawienia nadpisywania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "itemids": "30980",
        "selectOverrides": ["name", "step", "stop", "filter", "operations"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": 1
}
```


Zobacz także

- Prototyp wykresu
- Host
- Prototyp pozycji
- Filtr reguły LLD
- Prototyp wyzwalacza

Źródło

CDiscoveryRule::get() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.update

Opis

`object discoveryrule.update(object/array lldRules)`

Ta metoda umożliwia aktualizację istniejących reguł LLD.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object/array)` Właściwości reguły LLD do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdej reguły LLD, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości reguły LLD metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr reguły LLD zastępujący istniejący filtr.
preprocessing	object/array	Opcje preprocessingu reguły LLD zastępujące istniejące opcje preprocessingu. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
lld_macro_paths	object/array	Opcje lld_macro_path reguły LLD zastępujące istniejące opcje lld_macro_path. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
overrides	object/array	Opcje nadpisań reguły LLD zastępujące istniejące opcje nadpisań. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych reguł LLD we właściwości `itemids`.

Przykłady

Dodawanie filtra do reguły LLD

Dodaj filtr, aby zawartość makra *{#FSTYPE}* pasowała do wyrażenia regularnego *@File systems for discovery*.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "filter": {
            "evaltype": 1,
            "conditions": [
                {
                    "macro": "{#FSTYPE}",
                    "value": "@File systems for discovery"
                }
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Dodawanie ścieżek makr LLD

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "lld_macro_paths": [
            {
                "lld_macro": "{#MACRO1}",
                "path": "$.json.path"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Wyłącz przechwytywanie

Wyłącz przechwytywanie LLD dla reguły wykrywania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "28336",
        "allow_traps": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
```

```
    "id": 1
}
```

Aktualizowanie opcji przetwarzania wstępnego reguły LLD

Zaktualizuj regułę LLD za pomocą reguły przetwarzania wstępnego „JSONPath”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.path.to.json",
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Aktualizacja skryptu reguły LLD

Zaktualizuj skrypt reguły LLD innym skryptem i usuń zbędne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Aktualizacja czasu życia reguły LLD

Zaktualizuj regułę LLD, aby wyłączyć encję, która nie jest już wykrywana, po 12 godzinach i usunąć ją po 7 dniach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "46864",
        "lifetime_type": 0,
        "lifetime": "7d",
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "12h"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Źródło

CDiscoveryRule::update() w *ui/include/classes/api/services/CDiscoveryRule.php*.

Reguła wykrywania

Ta klasa została zaprojektowana do pracy z regułami wykrywania sieci.

> **Note:**
> To API jest przeznaczone do pracy z regułami wykrywania sieci. W przypadku reguł wykrywania niskiego poziomu zobacz API reguł LLD.

Odwołania do obiektów:

- Reguła wykrywania

Dostępne metody:

- drule.create - utwórz nowe reguły wykrywania sieci
- drule.delete - usuń reguły wykrywania sieci
- drule.get - pobierz reguły wykrywania sieci
- drule.update - zaktualizuj reguły wykrywania sieci

Obiekt reguły odkrywania

Następujące obiekty są bezpośrednio powiązane z API `drule`.

Reguła wykrywania

Obiekt reguły wykrywania definiuje regułę wykrywania sieci. Ma on następujące właściwości.

Właściwość	Typ	Opis
druleid	ID	ID reguły wykrywania. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
iprange	string	Jeden lub kilka zakresów IP do sprawdzenia, oddzielonych przecinkami. Więcej informacji o obsługiwanych formatach zakresów IP można znaleźć w sekcji konfiguracja wykrywania sieci. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa reguły wykrywania. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
delay	string	Interwał wykonywania reguły wykrywania. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) albo makro użytkownika. Domyślnie: 1h.
proxyid	ID	ID proxy używanego do wykrywania.
status	integer	Określa, czy reguła wykrywania jest włączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.
concurrency_max	integer	Maksymalna liczba równoczesnych kontroli na regułę wykrywania. Możliwe wartości: 0 - *(domyślnie)* nieograniczona liczba kontroli; 1 - jedna kontrola; 2-999 - niestandardowa liczba kontroli.
error	string	Tekst błędu, jeśli podczas wykonywania reguły wykrywania wystąpiły jakiekolwiek problemy. Zachowanie właściwości: - *tylko do odczytu*

drule.create

Opis

`object drule.create(object/array discoveryRules)`

Ta metoda umożliwia tworzenie nowych reguł wykrywania.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Reguły wykrywania do utworzenia.

Oprócz standardowych właściwości reguły wykrywania, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dchecks	array	Kontrole wykrywania do utworzenia dla reguły wykrywania. Zachowanie parametru: - *wymagany*

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych reguł wykrywania we właściwości druleids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych reguł wykrywania.

Przykłady

Utwórz regułę wykrywania

Utwórz regułę wykrywania, aby znaleźć maszyny z uruchomionym agent Zabbix w sieci lokalnej. Reguła musi używać pojedynczego sprawdzenia agent Zabbix na porcie 10050.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.create",
    "params": {
        "name": "Zabbix agent discovery",
        "iprange": "192.168.1.1-255",
        "concurrency_max": "10",
        "dchecks": [
            {
                "type": "9",
                "key_": "system.uname",
                "ports": "10050",
                "uniq": "0"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Kontrola wykrywania

Źródło

CDRule::create() w *ui/include/classes/api/services/CDRule.php*.

drule.delete

Opis

`object drule.delete(array discoveryRuleIds)`

Ta metoda pozwala na usuwanie reguł wykrywania.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

(array) Identyfikatory ID reguł wykrywania do usunięcia.

Wartości zwracane

(object) Zwraca obiekt zawierający identyfikatory usuniętych reguł wykrywania w ramach właściwości druleids.

Przykłady

Usuń wiele reguł wykrywania

Usuń dwie reguły wykrywania.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "drule.delete",
"params": [
"4",
"6"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"druleids": [
"4",
"6"
]
},
"id": 1
}
```

Źródło

CDRule::delete() w *ui/include/classes/api/services/CDRule.php*.

drule.get

Opis

`integer/array drule.get(object parameters)`

Ta metoda umożliwia pobranie reguł wykrywania zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dhostids	ID/array	Zwraca tylko reguły wykrywania, które utworzyły podane wykryte hosty.
druleids	ID/array	Zwraca tylko reguły wykrywania o podanych ID.
dserviceids	ID/array	Zwraca tylko reguły wykrywania, które utworzyły podane wykryte usługi.
selectDChecks	query	Zwraca właściwość `dchecks` z kontrolami wykrywania używanymi przez regułę wykrywania. Obsługuje `count`.
selectDHosts	query	Zwraca właściwość `dhosts` z wykrytymi hostami utworzonymi przez regułę wykrywania. Obsługuje `count`.

Parametr	Typ	Opis
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Ma zastosowanie do następujących podzapytań: `selectDChecks` - wyniki będą sortowane według `dcheckid`; `selectDHosts` - wyniki będą sortowane według `dhostsid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `druleid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz wszystkie reguły wykrywania

Pobierz wszystkie skonfigurowane reguły wykrywania oraz kontrole wykrywania, których używają.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.get",
    "params": {
        "output": "extend",
        "selectDChecks": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "druleid": "2",
            "proxyid": "0",
            "name": "Local network",
            "iprange": "192.168.3.1-255",
            "delay": "5s",
            "status": "0",
            "concurrency_max": "0",
            "error": "",
            "dchecks": [
                {
                    "dcheckid": "7",
                    "druleid": "2",
                    "type": "3",
```

```
                "key_": "",
                "snmp_community": "",
                "ports": "21",
                "snmpv3_securityname": "",
                "snmpv3_securitylevel": "0",
                "snmpv3_authpassphrase": "",
                "snmpv3_privpassphrase": "",
                "uniq": "0",
                "snmpv3_authprotocol": "0",
                "snmpv3_privprotocol": "0",
                "snmpv3_contextname": "",
                "host_source": "1",
                "name_source": "0",
                "allow_redirect": "0"
            },
            {
                "dcheckid": "8",
                "druleid": "2",
                "type": "4",
                "key_": "",
                "snmp_community": "",
                "ports": "80",
                "snmpv3_securityname": "",
                "snmpv3_securitylevel": "0",
                "snmpv3_authpassphrase": "",
                "snmpv3_privpassphrase": "",
                "uniq": "0",
                "snmpv3_authprotocol": "0",
                "snmpv3_privprotocol": "0",
                "snmpv3_contextname": "",
                "host_source": "1",
                "name_source": "0",
                "allow_redirect": "0"
            }
        ]
    },
    {
        "druleid": "6",
        "proxyid": "0",
        "name": "Zabbix agent discovery",
        "iprange": "192.168.1.1-255",
        "delay": "1h",
        "status": "0",
        "concurrency_max": "10",
        "error": "",
        "dchecks": [
            {
                "dcheckid": "10",
                "druleid": "6",
                "type": "9",
                "key_": "system.uname",
                "snmp_community": "",
                "ports": "10050",
                "snmpv3_securityname": "",
                "snmpv3_securitylevel": "0",
                "snmpv3_authpassphrase": "",
                "snmpv3_privpassphrase": "",
                "uniq": "0",
                "snmpv3_authprotocol": "0",
                "snmpv3_privprotocol": "0",
                "snmpv3_contextname": "",
                "host_source": "2",
```

```
                    "name_source": "3",
                    "allow_redirect": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Wykryty host
- Sprawdzenie wykrywania

Źródło

CDRule::get() w *ui/include/classes/api/services/CDRule.php*.

drule.update

Opis

`object drule.update(object/array discoveryRules)`

Ta metoda umożliwia aktualizację istniejących reguł wykrywania.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości reguły wykrywania do zaktualizowania.

Właściwość `druleid` musi być zdefiniowana dla każdej reguły wykrywania, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości reguły wykrywania, metoda akceptuje następujące parametry.

Parametr	Type	Opis
dchecks	array	Kontrole wykrywania zastępujące istniejące kontrole.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych reguł wykrywania właściwości `druleids`.

Przykłady

Zmień zakres IP reguły wykrywania

Zmień zakres IP reguły wykrywania na „192.168.2.1-255”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.update",
    "params": {
        "druleid": "6",
        "iprange": "192.168.2.1-255"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Kontrola wykrywania

Źródło

CDRule::update() w *ui/include/classes/api/services/CDRule.php*.

Rola

Ta klasa jest przeznaczona do pracy z rolami użytkowników.

Odwołania do obiektów:

- Role
- Role rules
  - UI element
  - Service
  - Service tag
  - Module
  - Action

Dostępne metody:

- role.create - utwórz nowe role użytkowników
- role.delete - usuń role użytkowników
- role.get - pobierz role użytkowników
- role.update - zaktualizuj role użytkowników

Obiekt roli

Następujące obiekty są bezpośrednio powiązane z API `role`.

Rola

Obiekt roli ma następujące właściwości:

Właściwość	Typ	Opis
roleid	ID	ID roli. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa roli. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ użytkownika. Możliwe wartości: 1 - *(domyślnie)* Użytkownik; 2 - Admin; 3 - Super admin. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
readonly	integer	Czy rola jest tylko do odczytu. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak. Zachowanie właściwości: - *tylko do odczytu*

Reguły roli

Obiekt reguł roli ma następujące właściwości:

Właściwość	Type	Opis
ui	array	Tablica obiektów elementu UI.
ui.default_access	integer	Określa, czy dostęp do nowych elementów UI jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
services.read.mode	integer	Dostęp tylko do odczytu do usług. Możliwe wartości: 0 - Dostęp tylko do odczytu do usług określonych przez właściwość `services.read.list` lub dopasowanych przez właściwość `services.read.tag`; 1 - *(domyślnie)* Dostęp tylko do odczytu do wszystkich usług.
services.read.list	array	Tablica obiektów Service. Określone usługi, w tym usługi podrzędne, otrzymają dla roli użytkownika dostęp tylko do odczytu. Dostęp tylko do odczytu nie zastąpi dostępu do odczytu i zapisu do usług. Zachowanie właściwości: - *obsługiwane*, jeśli `services.read.mode` jest ustawione na ”0”
services.read.tag	object	Tablica obiektów Service tag. Usługi dopasowane przez tag, w tym usługi podrzędne, otrzymają dla roli użytkownika dostęp tylko do odczytu. Dostęp tylko do odczytu nie zastąpi dostępu do odczytu i zapisu do usług. Zachowanie właściwości: - *obsługiwane*, jeśli `services.read.mode` jest ustawione na ”0”
services.write.mode	integer	Dostęp do odczytu i zapisu do usług. Możliwe wartości: 0 - *(domyślnie)* Dostęp do odczytu i zapisu do usług określonych przez właściwość `services.write.list` lub dopasowanych przez właściwość `services.write.tag`; 1 - Dostęp do odczytu i zapisu do wszystkich usług.
services.write.list	array	Tablica obiektów Service. Określone usługi, w tym usługi podrzędne, otrzymają dla roli użytkownika dostęp do odczytu i zapisu. Dostęp do odczytu i zapisu zastąpi dostęp tylko do odczytu do usług. Zachowanie właściwości: - *obsługiwane*, jeśli `services.write.mode` jest ustawione na ”0”

Właściwość	Type	Opis
services.write.tag	object	Tablica obiektów Service tag. Usługi dopasowane przez tag, w tym usługi podrzędne, otrzymają dla roli użytkownika dostęp do odczytu i zapisu. Dostęp do odczytu i zapisu zastąpi dostęp tylko do odczytu do usług. Zachowanie właściwości: - *obsługiwane*, jeśli `services.write.mode` jest ustawione na ”0”
modules	array	Tablica obiektów modułu.
modules.default_access	integer	Określa, czy dostęp do nowych modułów jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
api.access	integer	Określa, czy dostęp do API jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
api.mode	integer	Tryb traktowania metod API wymienionych we właściwości `api`. Możliwe wartości: 0 - *(domyślnie)* Lista zabronionych; 1 - Lista dozwolonych.
api	array	Tablica metod API.
actions	array	Tablica obiektów akcji.
actions.default_access	integer	Określa, czy dostęp do nowych akcji jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.

Element UI

Obiekt elementu UI ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa elementu UI. Możliwe wartości, jeśli type obiektu Role jest ustawiony na ”User”, ”Admin” lub ”Super admin”: `monitoring.dashboard` - *Pulpity*; `monitoring.problems` - *Monitorowanie → Problemy*; `monitoring.hosts` - *Monitorowanie → Hosty*; `monitoring.latest_data` - *Monitorowanie → Najnowsze dane*; `monitoring.maps` - *Monitorowanie → Mapy*; `services.services` - *Usługi → Usługi*; `services.sla_report` - *Usługi → Raport SLA*; `inventory.overview` - *Inwentaryzacja → Przegląd*; `inventory.hosts` - *Inwentaryzacja → Hosty*; `reports.availability_report` - *Raporty → Raport dostępności*; `reports.top_triggers` - *Raporty → 100 najważniejszych wyzwalaczy*. Możliwe wartości, jeśli type obiektu Role jest ustawiony na ”Admin” lub ”Super admin”: `monitoring.discovery` - *Monitorowanie → Wykrywanie*; `services.sla` - *Usługi → SLA*; `reports.scheduled_reports` - *Raporty → Raporty cykliczne*; `reports.notifications` - *Raporty → Powiadomienia*; `configuration.template_groups` - *Zbieranie danych → Grupy szablonów*; `configuration.host_groups` - *Zbieranie danych → Grupy hostów*; `configuration.templates` - *Zbieranie danych → Szablony*; `configuration.hosts` - *Zbieranie danych → Hosty*; `configuration.maintenance` - *Zbieranie danych → Konserwacja*; `configuration.discovery` - *Zbieranie danych → Wykrywanie*; `configuration.trigger_actions` - *Alerty → Akcje → Akcje wyzwalaczy*; `configuration.service_actions` - *Alerty → Akcje → Akcje usług*; `configuration.discovery_actions` - *Alerty → Akcje → Akcje wykrywania*; `configuration.autoregistration_actions` - *Alerty → Akcje → Akcje autorejestracji*; `configuration.internal_actions` - *Alerty → Akcje → Akcje wewnętrzne*. Możliwe wartości, jeśli type obiektu Role jest ustawiony na ”Super admin”: `reports.system_info` - *Raporty → Informacje systemowe*; `reports.audit` - *Raporty → Dziennik audytu*; `reports.action_log` - *Raporty → Dziennik akcji*; `configuration.event_correlation` - *Zbieranie danych → Korelacja zdarzeń*; `administration.media_types` - *Alerty → Typy mediów*; `administration.scripts` - *Alerty → Skrypty*; `administration.user_groups` - *Użytkownicy → Grupy użytkowników*; `administration.user_roles` - *Użytkownicy → Role użytkowników*; `administration.users` - *Użytkownicy → Użytkownicy*; `administration.api_tokens` - *Użytkownicy → Tokeny API*; `administration.authentication` - *Użytkownicy → Uwierzytelnianie*; `administration.general` - *Administracja → Ogólne*; `administration.audit_log` - *Administracja → Dziennik audytu*; `administration.housekeeping` - *Administracja → Housekeeping*; `administration.proxy_groups` - *Administracja → Grupy proxy*; `administration.proxies` - *Administracja → Proxy*; `administration.macros` - *Administracja → Makra*; `administration.queue` - *Administracja → Kolejka*. Zachowanie właściwości:

Właściwość	Typ	Opis
status	integer	Czy dostęp do elementu UI jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.

Usługa

Właściwość	Type	Opis
serviceid	ID	ID usługi. Property behavior: - *wymagane*

Znacznik usługi

Właściwość	Type	Opis
tag	string	Nazwa znacznika. Jeśli zostanie podany pusty ciąg znaków, znacznik usługi nie będzie używany do dopasowywania usług. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika. Jeśli nie zostanie podana żadna wartość lub zostanie podany pusty ciąg znaków, do dopasowywania usług będzie używana tylko nazwa znacznika.

Moduł

Obiekt modułu ma następujące właściwości:

Właściwość	Typ	Opis
moduleid	ID	ID modułu. Zachowanie właściwości: - *wymagane*
status	integer	Czy dostęp do modułu jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.

Akcja

Obiekt akcji ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa akcji. Możliwe wartości, jeśli `type` obiektu Role jest ustawione na ”User”, ”Admin” lub ”Super admin”: `edit_dashboards` - Tworzenie i edycja pulpitów; `edit_maps` - Tworzenie i edycja map; `add_problem_comments` - Dodawanie komentarzy do problemów; `change_severity` - Zmiana ważności problemu; `acknowledge_problems` - Potwierdzanie problemów; `suppress_problems` - Tłumienie problemów; `close_problems` - Zamykanie problemów; `execute_scripts` - Wykonywanie skryptów; `manage_api_tokens` - Zarządzanie tokenami API; `change_problem_ranking` - Zmiana klasyfikacji problemu z przyczyny na symptom i odwrotnie; `edit_own_media` - Zezwala na tworzenie/edycję własnych mediów. Możliwe wartości, jeśli `type` obiektu Role jest ustawione na ”Admin” lub ”Super admin”: `edit_maintenance` - Tworzenie i edycja okresów konserwacji; `manage_scheduled_reports` - Zarządzanie zaplanowanymi raportami, `manage_sla` - Zarządzanie SLA. Możliwe wartości, jeśli `type` obiektu Role jest ustawione na ”User” lub ”Admin”: `invoke_execute_now` - pozwala wykonywać kontrole pozycji użytkownikom, którzy mają tylko uprawnienia do odczytu na hoście. Możliwe wartości, jeśli `type` obiektu Role jest ustawione na ”Super admin”: `edit_user_media` - Zezwala na tworzenie/edycję mediów dla użytkowników. Zachowanie właściwości: *- wymagane*
status	integer	Czy dostęp do wykonania akcji jest włączony. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.

role.create

Opis

`object role.create(object/array roles)`

Ta metoda umożliwia tworzenie nowych ról.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Role do utworzenia.

Oprócz standardowych właściwości roli, metoda akceptuje następujące parametry.

Parametr	Type	Opis
rules	array	Reguły roli, które mają zostać utworzone dla roli.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych ról we właściwości `roleids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych ról.

Przykłady

Tworzenie roli

Utwórz rolę typu „Użytkownik” z zablokowanym dostępem do dwóch elementów interfejsu użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.create",
    "params": {
        "name": "Operator",
        "type": "1",
        "rules": {
            "ui": [
                {
                    "name": "monitoring.hosts",
                    "status": "0"
                },
                {
                    "name": "monitoring.maps",
                    "status": "0"
                }
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Reguły roli
- Element interfejsu użytkownika
- Moduł
- Akcja

Źródło

CRole::create() w *ui/include/classes/api/services/CRole.php*.

role.delete

Opis

`object role.delete(array roleids)`

Ta metoda pozwala usunąć role.

Note:
Ta metoda jest dostępna dla użytkownika typu *Super admin*. Uprawnienia mogą zostać cofnięte w konfiguracji roli użytkownika. Więcej informacji User roles.

Parametry

`(array)` ID ról do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych ról we właściwości `roleids`.

Przykłady

Usunięcie wielu ról użytkownika

Usunięcie dwóch ról użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.delete",
    "params": [
        "4",
        "5"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CRole::delete() w *ui/include/classes/api/services/CRole.php*.

role.get

Opis

`integer/array role.get(object parameters)`

Ta metoda umożliwia pobieranie ról zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
roleids	ID/array	Zwracaj tylko role o podanych identyfikatorach.

Parametr	Typ	Opis
selectRules	query	Zwracaj właściwość `rules` z regułami roli.
selectUsers	query	Zwracaj właściwość `users` z użytkownikami, do których przypisana jest rola. Ograniczenia zależne od typu użytkownika opisano w `user.get`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `roleid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych roli

Pobierz dane roli "Super admin role" oraz jej reguły dostępu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.get",
    "params": {
        "output": "extend",
        "selectRules": "extend",
        "roleids": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "services.services",
            "status": "1"
        },
        {
            "name": "services.sla_report",
            "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": "services.sla",
            "status": "1"
        },
        {
            "name": "reports.scheduled_reports",
            "status": "1"
        },
        {
            "name": "reports.notifications",
            "status": "1"
        },
        {
            "name": "configuration.template_groups",
            "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.discovery",
                    "status": "1"
                },
                {
                    "name": "configuration.trigger_actions",
                    "status": "1"
                },
                {
                    "name": "configuration.service_actions",
                    "status": "1"
                },
                {
                    "name": "configuration.discovery_actions",
                    "status": "1"
                },
                {
                    "name": "configuration.autoregistration_actions",
                    "status": "1"
                },
                {
                    "name": "configuration.internal_actions",
                    "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.media_types",
                    "status": "1"
                },
                {
                    "name": "administration.scripts",
                    "status": "1"
                },
                {
                    "name": "administration.user_groups",
```

```
                "status": "1"
            },
            {
                "name": "administration.user_roles",
                "status": "1"
            },
            {
                "name": "administration.users",
                "status": "1"
            },
            {
                "name": "administration.api_tokens",
                "status": "1"
            },
            {
                "name": "administration.authentication",
                "status": "1"
            },
            {
                "name": "administration.general",
                "status": "1"
            },
            {
                "name": "administration.audit_log",
                "status": "1"
            },
            {
                "name": "administration.housekeeping",
                "status": "1"
            },
            {
                "name": "administration.proxy_groups",
                "status": "1"
            },
            {
                "name": "administration.proxies",
                "status": "1"
            },
            {
                "name": "administration.macros",
                "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": [
            {
```

```
            "moduleid": 1,
            "status": "1"
        },
        {
            "moduleid": 2,
            "status": "1"
        },
        {
            "moduleid": 3,
            "status": "1"
        },
        {
            "moduleid": 4,
            "status": "1"
        },
        {
            "moduleid": 5,
            "status": "1"
        },
        {
            "moduleid": 6,
            "status": "1"
        },
        {
            "moduleid": 7,
            "status": "1"
        },
        {
            "moduleid": 8,
            "status": "1"
        },
        {
            "moduleid": 9,
            "status": "1"
        },
        {
            "moduleid": 10,
            "status": "1"
        },
        {
            "moduleid": 11,
            "status": "1"
        },
        {
            "moduleid": 12,
            "status": "1"
        },
        {
            "moduleid": 13,
            "status": "1"
        },
        {
            "moduleid": 14,
            "status": "1"
        },
        {
            "moduleid": 15,
            "status": "1"
        },
        {
            "moduleid": 16,
            "status": "1"
```

```
        },
        {
            "moduleid": 17,
            "status": "1"
        },
        {
            "moduleid": 18,
            "status": "1"
        },
        {
            "moduleid": 19,
            "status": "1"
        },
        {
            "moduleid": 20,
            "status": "1"
        },
        {
            "moduleid": 21,
            "status": "1"
        },
        {
            "moduleid": 22,
            "status": "1"
        },
        {
            "moduleid": 23,
            "status": "1"
        },
        {
            "moduleid": 24,
            "status": "1"
        },
        {
            "moduleid": 25,
            "status": "1"
        },
        {
            "moduleid": 26,
            "status": "1"
        },
        {
            "moduleid": 27,
            "status": "1"
        },
        {
            "moduleid": 28,
            "status": "1"
        },
        {
            "moduleid": 29,
            "status": "1"
        },
        {
            "moduleid": 30,
            "status": "1"
        },
        {
            "moduleid": 31,
            "status": "1"
        },
        {
```

```
                "moduleid": 32,
                "status": "1"
            }
        ],
        "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": "suppress_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"
            },
            {
                "name": "manage_sla",
                "status": "1"
            },
            {
                "name": "invoke_execute_now",
                "status": "1"
            },
            {
```

```
                        "name": "change_problem_ranking",
                        "status": "1"
                    },
                    {
                        "name": "edit_own_media",
                        "status": "1"
                    },
                    {
                        "name": "edit_user_media",
                        "status": "1"
                    }
                ],
                "actions.default_access": "1"
            }
        }
    ],
    "id": 1
}
```

Zobacz także

- Reguły roli
- Użytkownik

Źródło

CRole::get() w *ui/include/classes/api/services/CRole.php*.

role.update

Opis

`object role.update(object/array roles)`

Ta metoda umożliwia aktualizację istniejących ról.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object/array)` Właściwości roli do zaktualizowania.

Właściwość `roleid` musi być zdefiniowana dla każdej roli, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości roli metoda akceptuje następujące parametry.

Parametr	Type	Opis
rules	array	Reguły dostępu rules do zaktualizowania dla roli.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych ról we właściwości `roleids`.

Przykłady

Wyłączanie możliwości wykonywania skryptów

Zaktualizuj rolę o ID „5”, wyłącz możliwość wykonywania skryptów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
```

```
    "params": [
        {
            "roleid": "5",
            "rules": {
                "actions": [
                    {
                        "name": "execute_scripts",
                        "status": "0"
                    }
                ]
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Ograniczanie dostępu do API

Zaktualizuj rolę o ID „5”, zabraniając wywoływania dowolnych metod „create”, „update” lub „delete”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "api.access": "1",
                "api.mode": "0",
                "api": ["*.create", "*.update", "*.delete"]
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CRole::update() w *ui/include/classes/api/services/CRole.php*.

Scenariusz WWW

Ta klasa jest przeznaczona do pracy ze scenariuszami WWW.

Odwołania do obiektów:

- Scenariusz WWW
- Krok scenariusza
  - Pole HTTP
- Tag scenariusza WWW

Dostępne metody:

- httptest.create - utwórz nowe scenariusze WWW
- httptest.delete - usuń scenariusze WWW
- httptest.get - pobierz scenariusze WWW
- httptest.update - zaktualizuj scenariusze WWW

Obiekt scenariusza internetowego

Następujące obiekty są bezpośrednio powiązane z API `webcheck`.

Scenariusz WWW

Obiekt scenariusza WWW ma następujące właściwości.

Właściwość	Typ	Opis
httptestid	ID	ID scenariusza WWW. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
hostid	ID	ID hosta, do którego należy scenariusz WWW. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
name	string	Nazwa scenariusza WWW. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
agent	string	Ciąg user agent, który będzie używany przez scenariusz WWW. Domyślnie: Zabbix
authentication	integer	Metoda uwierzytelniania, która będzie używana przez scenariusz WWW. Możliwe wartości: 0 - *(domyślnie)* brak; 1 - podstawowe uwierzytelnianie HTTP; 2 - uwierzytelnianie NTLM; 3 - uwierzytelnianie Kerberos; 4 - uwierzytelnianie Digest.
delay	string	Interwał wykonywania scenariusza WWW. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 1m, 2h, 1d) albo makro użytkownika. Domyślnie: 1m.
headers	array	Nagłówki HTTP, które będą wysyłane podczas wykonywania żądania.
http_password	string	Hasło używane do podstawowego uwierzytelniania HTTP, NTLM, Kerberos lub Digest.
http_proxy	string	Proxy, który będzie używany przez scenariusz WWW, podany jako *http://[username[:password]@]proxy.example.com[:port]*.

Właściwość	Typ	Opis
http_user	string	Nazwa użytkownika używana do podstawowego uwierzytelniania HTTP, NTLM, Kerberos lub Digest.
retries	integer	Liczba prób wykonania każdego kroku przez scenariusz WWW przed zgłoszeniem niepowodzenia. Domyślnie: 1.
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło klucza prywatnego SSL.
status	integer	Określa, czy scenariusz WWW jest włączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
templateid	ID	ID nadrzędnego scenariusza WWW szablonu. Zachowanie właściwości: - *tylko do odczytu*
variables	array	Zmienne scenariusza WWW.
verify_host	integer	Określa, czy należy sprawdzać, czy nazwa hosta dla połączenia jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* pomiń weryfikację hosta; 1 - weryfikuj hosta.
verify_peer	integer	Określa, czy należy sprawdzać autentyczność certyfikatu hosta. Możliwe wartości: 0 - *(domyślnie)* pomiń weryfikację peer; 1 - weryfikuj peer.
uuid	string	Globalny unikalny identyfikator używany do łączenia importowanych scenariuszy WWW z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli scenariusz WWW należy do szablonu

Krok scenariusza

Obiekt kroku scenariusza definiuje określone sprawdzenie scenariusza webowego. Ma on następujące właściwości.

Property	Type	Description
name	string	Nazwa kroku scenariusza. Zachowanie właściwości: - *required*
no	integer	Numer sekwencyjny kroku w scenariuszu webowym. Zachowanie właściwości: - *required*
url	string	Adres URL do sprawdzenia. Zachowanie właściwości: - *required*

Property	Type	Description
follow_redirects	integer	Czy podążać za przekierowaniami HTTP. Possible values: 0 - don't follow redirects; 1 - *(default)* follow redirects.
headers	array	Nagłówki HTTP, które zostaną wysłane podczas wykonywania żądania. Nagłówki kroku scenariusza zastąpią nagłówki określone dla scenariusza webowego.
posts	string/array	Zmienne HTTP POST jako ciąg znaków (surowe dane POST) lub jako tablica pól HTTP (dane pól formularza).
required	string	Tekst, który musi być obecny w odpowiedzi.
retrieve_mode	integer	Część odpowiedzi HTTP, którą krok scenariusza ma pobrać. Possible values: 0 - *(default)* only body; 1 - only headers; 2 - headers and body.
status_codes	string	Zakresy wymaganych kodów statusu HTTP, rozdzielone przecinkami.
timeout	string	Limit czasu żądania w sekundach. Akceptuje sekundy, jednostkę czasu z sufiksem lub makro użytkownika. Default: 15s. Maximum: 1h. Minimum: 1s.
variables	array	Zmienne kroku scenariusza.
query_fields	array	Pola zapytania - tablica pól HTTP, które zostaną dodane do adresu URL podczas wykonywania żądania.

Pole HTTP

Obiekt pola HTTP definiuje nazwę i wartość używane do określania zmiennych scenariusza WWW, nagłówków HTTP oraz pól POST lub pól zapytania. Ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa nagłówka/zmiennej/pola POST lub GET. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka/zmiennej/pola POST lub GET. Zachowanie właściwości: - *wymagane*

Tag scenariusza web

Obiekt tagu scenariusza web ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu scenariusza web. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu scenariusza web.

httptest.create

Opis

`object httptest.create(object/array webScenarios)`

Ta metoda umożliwia tworzenie nowych scenariuszy WWW.

Note:
Utworzenie scenariusza WWW automatycznie utworzy zestaw pozycji monitorowania WWW.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Scenariusze web do utworzenia.

Oprócz standardowych właściwości scenariusza web, metoda akceptuje następujące parametry.

Parametr	Type	Opis
steps	array	Kroki scenariusza. Zachowanie parametru: - *wymagany*
tags	array	Tagi scenariusza web.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych scenariuszy WWW we właściwości `httptestids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych scenariuszy WWW.

Przykłady

Tworzenie scenariusza WWW

Utwórz scenariusz WWW do monitorowania strony głównej firmy. Scenariusz będzie miał dwa kroki, aby sprawdzić stronę główną oraz stronę „O nas” i upewnić się, że zwracają kod statusu HTTP 200.

Żądanie:

```
{
    "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
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
```

```
    },
    "id": 1
}
```

Zobacz także

- Krok scenariusza

Źródło

CHttpTest::create() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.delete

Opis

`object httptest.delete(array webScenarioIds)`

Ta metoda umożliwia usuwanie scenariuszy sieci WWW.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Role użytkownika.

Parametry

`(array)` ID scenariuszy sieci WWW do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych scenariuszy sieci WWW we właściwości `httptestids`.

Przykłady

Usuwanie wielu scenariuszy sieci WWW

Usuń dwa scenariusze sieci WWW.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "httptest.delete",
"params": [
"2",
"3"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"httptestids": [
"2",
"3"
]
},
"id": 1
}
```

Źródło

CHttpTest::delete() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.get

Opis

`integer/array httptest.get(object parameters)`

Ta metoda umożliwia pobieranie scenariuszy WWW zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
groupids	ID/array	Zwracaj tylko web scenarios należące do podanych grup hostów.
hostids	ID/array	Zwracaj tylko web scenarios należące do podanych hostów.
httptestids	ID/array	Zwracaj tylko web scenarios o podanych ID.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko web scenarios odziedziczone z szablonu.
monitored	boolean	Jeśli ustawione na `true`, zwracaj tylko włączone web scenarios należące do monitorowanych hostów.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko web scenarios należące do szablonów.
templateids	ID/array	Zwracaj tylko web scenarios należące do podanych szablonów.
expandName	flag	Rozszerz makra w nazwie web scenario.
expandStepName	flag	Rozszerz makra w nazwach kroków scenariusza.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwracaj tylko web scenarios z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie web scenarios. Możliwe wartości operator: 0 - *(domyślnie)* Contains; 1 - Equals; 2 - Does not contain; 3 - Does not equal; 4 - Exists; 5 - Does not exist.
selectHosts	query	Zwracaj hosty, do których należy web scenario, jako tablicę w właściwości `hosts`.
selectSteps	query	Zwracaj kroki web scenario w właściwości `steps`. Obsługuje `count`.
selectTags	query	Zwracaj tagi web scenario w właściwości `tags`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `httptestid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parameter	Type	Description
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie scenariusza webowego

Pobierz wszystkie dane o scenariuszu webowym "9".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.get",
    "params": {
        "output": "extend",
        "selectSteps": "extend",
        "httptestids": "9"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "httptestid": "9",
            "name": "Homepage check",
            "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
}
```


Zobacz także

- Host
- Krok scenariusza

Źródło

CHttpTest::get() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.update

Opis

`object httptest.update(object/array webScenarios)`

Ta metoda umożliwia aktualizację istniejących scenariuszy WWW.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości scenariusza WWW do zaktualizowania.

Właściwość `httptestid` musi być zdefiniowana dla każdego scenariusza WWW, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości scenariusza WWW, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
steps	array	Kroki scenariusza zastępujące istniejące kroki.
tags	array	Tagi scenariusza WWW.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych scenariuszy WWW we właściwości `httptestid`.

Przykłady

Włączanie scenariusza WWW

Włącz scenariusz WWW, to znaczy ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.update",
    "params": {
        "httptestid": "5",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Krok scenariusza

Źródło

CHttpTest::update() w *ui/include/classes/api/services/CHttpTest.php*.

Skrypt

Ta klasa jest przeznaczona do pracy ze skryptami.

Odwołania do obiektów:

- Script
  - Parametry webhooka
- Debug
  - Wpis dziennika

Dostępne metody:

- script.create - utwórz nowe skrypty
- script.delete - usuń skrypty
- script.execute - uruchom skrypty
- script.get - pobierz skrypty
- script.getscriptsbyevents - pobierz skrypty dla zdarzeń
- script.getscriptsbyhosts - pobierz skrypty dla hostów
- script.update - zaktualizuj skrypty

Obiekt skryptu

Następujące obiekty są bezpośrednio powiązane z API `script`.

Skrypt

Obiekt skryptu ma następujące właściwości.

Właściwość	Typ	Opis
scriptid	ID	ID skryptu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa skryptu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ skryptu. Możliwe wartości, jeśli `scope` jest ustawione na ”action operation”: 0 - Script; 1 - IPMI; 2 - SSH; 3 - TELNET; 5 - Webhook. Możliwe wartości, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”: 6 - URL. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
command	string	Polecenie do uruchomienia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Script”, ”IPMI”, ”SSH”, ”TELNET” lub ”Webhook”
scope	integer	Zakres skryptu. Możliwe wartości: 1 - action operation; 2 - manual host action; 4 - manual event action. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
execute_on	integer	Gdzie uruchomić skrypt. Możliwe wartości: 0 - uruchom na Zabbix agent; 1 - uruchom na Zabbix serwer. Jest to *obsługiwane* tylko wtedy, gdy wykonywanie skryptów globalnych jest włączone na Zabbix serwerze; 2 - *(domyślnie)* uruchom na Zabbix serwerze lub proxy. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Script”
menu_path	string	Foldery oddzielone ukośnikiem, które tworzą nawigację w formie menu w frontend po kliknięciu hosta lub zdarzenia. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”

Właściwość	Typ	Opis
authtype	integer	Metoda uwierzytelniania używana dla skryptu typu SSH. Możliwe wartości: 0 - hasło; 1 - klucz publiczny. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”SSH”
username	string	Nazwa użytkownika używana do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH” lub ”TELNET”
password	string	Hasło używane dla skryptów SSH z uwierzytelnianiem hasłem oraz skryptów TELNET. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”SSH” i `authtype` jest ustawione na ”password”, lub `type` jest ustawione na ”TELNET”
publickey	string	Nazwa pliku klucza publicznego używanego dla skryptów SSH z uwierzytelnianiem kluczem publicznym. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH” i `authtype` jest ustawione na ”public key”
privatekey	string	Nazwa pliku klucza prywatnego używanego dla skryptów SSH z uwierzytelnianiem kluczem publicznym. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”SSH” i `authtype` jest ustawione na ”public key”
port	string	Numer portu używany dla skryptów SSH i TELNET. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”SSH” lub ”TELNET”
groupid	ID	ID grupy hostów, na której można uruchomić skrypt. Jeśli ustawione na ”0”, skrypt będzie dostępny dla wszystkich grup hostów. Domyślnie: 0.
usrgrpid	ID	ID grupy użytkowników, która będzie uprawniona do uruchamiania skryptu. Jeśli ustawione na ”0”, skrypt będzie dostępny dla wszystkich grup użytkowników. Domyślnie: 0. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”
host_access	integer	Uprawnienia do hosta wymagane do uruchomienia skryptu. Możliwe wartości: 2 - *(domyślnie)* odczyt; 3 - zapis. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”

Właściwość	Typ	Opis
confirmation	string	Tekst wyskakującego okna potwierdzenia. Okno wyskakujące pojawi się podczas próby uruchomienia skryptu z frontend Zabbix. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”
timeout	string	Limit czasu wykonania skryptu webhook w sekundach. Obsługiwane są sufiksy czasu (np. 30s, 1m). Możliwe wartości: 1-60s. Domyślnie: 30s. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”Webhook”
parameters	array	Tablica parametrów wejściowych webhooka. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”Webhook”
description	string	Opis skryptu.
url	string	URL zdefiniowany przez użytkownika. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na ”URL”
new_window	integer	Otwórz URL w nowym oknie. Możliwe wartości: 0 - Nie; 1 - *(domyślnie)* Tak. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na ”URL”
manualinput	integer	Wskazuje, czy skrypt akceptuje dane wejściowe podane przez użytkownika. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone; Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na ”manual host action” lub ”manual event action”
manualinput_prompt	string	Tekst monitu dla ręcznego wprowadzania danych. Zachowanie właściwości: - *wymagane*, jeśli `manualinput` jest ustawione na ”Enabled”
manualinput_validator	string	Pole tekstowe używane do walidacji danych wejściowych podanych przez użytkownika. Ciąg składa się albo z wyrażenia regularnego, albo z zestawu wartości oddzielonych przecinkami. Zachowanie właściwości: - *wymagane*, jeśli `manualinput` jest ustawione na ”Enabled”

Właściwość	Typ	Opis
manualinput_validator_type	integer	Określa oczekiwany typ danych wejściowych użytkownika. Możliwe wartości: 0 - *(domyślnie)* String. Oznacza, że manualinput_validator ma być traktowane jako wyrażenie regularne; 1 - List. Oznacza, że manualinput_validator ma być traktowane jako lista możliwych wartości wejściowych oddzielonych przecinkami.
manualinput_default_value	string	Zachowanie właściwości: - *obsługiwane*, jeśli `manualinput` jest ustawione na "Enabled" Wartość domyślna do automatycznego uzupełniania danych wejściowych użytkownika. Zachowanie właściwości: - *obsługiwane*, jeśli `manualinput_validator_type` jest ustawione na "String"

Parametry webhooka

Parametry przekazywane do skryptu webhooka podczas jego wywołania mają następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Obsługuje makra.

Debug

Informacje debugowania wykonanego skryptu webhook. Obiekt debug ma następujące właściwości.

Właściwość	Type	Opis
logs	array	Tablica wpisów dziennika.
ms	string	Czas wykonywania skryptu w milisekundach.

Wpis dziennika

Obiekt wpisu dziennika ma następujące właściwości.

Właściwość	Typ	Opis
level	integer	Poziom dziennika.
ms	string	Czas, który upłynął w milisekundach od uruchomienia skryptu do momentu dodania wpisu dziennika.
message	string	Komunikat dziennika.

script.create

Opis

`object script.create(object/array scripts)`

Ta metoda umożliwia tworzenie nowych skryptów.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object/array) Skrypty do utworzenia.

Metoda akceptuje skrypty ze standardowymi właściwościami skryptu.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych skryptów we właściwości scriptids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych skryptów.

Przykłady

Utwórz skrypt webhook

Utwórz skrypt webhook, który wysyła żądanie HTTP do zewnętrznej usługi.

Żądanie:

```
{
    "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
        "scope": 1,
        "type": 5,
        "timeout": "40s",
        "parameters": [
            {
                "name": "token",
                "value": "{$WEBHOOK.TOKEN}"
            },
            {
                "name": "host",
                "value": "{HOST.HOST}"
            },
            {
                "name": "v",
                "value": "2.2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3"
        ]
    },
    "id": 1
}
```

Utwórz skrypt SSH

Utwórz skrypt SSH z uwierzytelnianiem kluczem publicznym, który może być wykonywany na hoście i ma menu kontekstowe.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "SSH script",
        "command": "my script command",
        "type": 2,
        "authtype": 1,
```

```
        "username": "John",
        "publickey": "pub.key",
        "privatekey": "priv.key",
        "password": "secret",
        "port": "12345",
        "scope": 2,
        "menu_path": "All scripts/SSH",
        "usrgrpid": "7",
        "groupid": "4"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "5"
        ]
    },
    "id": 1
}
```

Utwórz niestandardowy skrypt

Utwórz niestandardowy skrypt, który zrestartuje serwer. Skrypt będzie wymagał uprawnień do zapisu do hosta i poprosi użytkownika o ręczne wprowadzenie danych. Po pomyślnym przesłaniu danych wejściowych skrypt wyświetli komunikat potwierdzający we frontendzie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Reboot server",
        "command": "reboot server {MANUALINPUT}",
        "type": 0,
        "scope": 2,
        "confirmation": "Are you sure you would like to reboot the server {MANUALINPUT}?",
        "manualinput": 1,
        "manualinput_prompt": "Which server you want to reboot?",
        "manualinput_validator": "[1-9]",
        "manualinput_validator_type": 0,
        "manualinput_default_value": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "4"
        ]
    },
    "id": 1
}
```

Utwórz skrypt typu URL

Utwórz skrypt typu URL dla zakresu host, który pozostaje w tym samym oknie i ma tekst potwierdzenia.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "URL script",
        "type": 6,
        "scope": 2,
        "url": "http://zabbix/ui/zabbix.php?action=host.edit&hostid={HOST.ID}",
        "confirmation": "Edytować host {HOST.NAME}?",
        "new_window": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "56"
        ]
    },
    "id": 1
}
```

Utwórz skrypt typu URL z ręcznym wprowadzaniem

Utwórz skrypt typu URL dla zakresu zdarzenia, który otwiera się w nowym oknie i ma ręczne wprowadzanie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Skrypt URL z ręcznym wprowadzaniem",
        "type": 6,
        "scope": 4,
        "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
        "new_window": 1,
        "manualinput": 1,
        "manualinput_prompt": "Wybierz stronę do otwarcia:",
        "manualinput_validator": "dashboard.view,script.list,actionlog.list",
        "manualinput_validator_type": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "57"
        ]
    },
    "id": 1
}
```

Źródło

CScript::create() w *ui/include/classes/api/services/CScript.php*.

script.delete

Opis

`object script.delete(array scriptIds)`

Ta metoda umożliwia usuwanie skryptów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID skryptów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych skryptów we właściwości `scriptids`.

Przykłady

Usuń wiele skryptów

Usuń dwa skrypty.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "script.delete",
"params": [
"3",
"4"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"scriptids": [
"3",
"4"
]
},
"id": 1
}
```

Źródło

CScript::delete() w *ui/include/classes/api/services/CScript.php*.

script.execute

Opis

`object script.execute(object parameters)`

Ta metoda umożliwia uruchomienie skryptu na hoście lub zdarzeniu. Z wyjątkiem skryptów typu URL. Nie są one wykonywalne.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

(object) Parametry zawierające ID skryptu do uruchomienia oraz albo ID hosta, albo ID zdarzenia i wartość manualinput.

Parametr	Type	Opis
scriptid	ID	ID skryptu do uruchomienia. Zachowanie parametru: - *wymagany*
hostid	ID	ID hosta, na którym ma zostać uruchomiony skrypt. Zachowanie parametru: - *wymagany*, jeśli `eventid` nie jest ustawione
eventid	ID	ID zdarzenia, dla którego ma zostać uruchomiony skrypt. Zachowanie parametru: - *wymagany*, jeśli `hostid` nie jest ustawione
manualinput	string	Wartość podana przez użytkownika, z którą ma zostać uruchomiony skrypt, zastępująca makro {MANUALINPUT}.

Zwracane wartości

(object) Zwraca wynik wykonania skryptu.

Właściwość	Type	Opis
response	string	Czy skrypt został uruchomiony pomyślnie. Możliwa wartość - `success`.
value	string	Dane wyjściowe skryptu.
debug	object	Zawiera obiekt debug, jeśli wykonywany jest skrypt webhook. Dla innych typów skryptów zawiera pusty obiekt.

Przykłady

Uruchamianie skryptu webhook

Uruchom skrypt webhook, który wysyła żądanie HTTP do zewnętrznej usługi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "4",
        "hostid": "30079"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Uruchamianie niestandardowego skryptu

Uruchom skrypt „ping” na hoście.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "1",
        "hostid": "30079"
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "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
}
```


Uruchom własny skrypt z ręcznym wprowadzaniem danych

Uruchom skrypt „ping” z poleceniem „ping -c {MANUALINPUT} {HOST.CONN}; case $? in [01]) true;; *) false;; esac” na hoście.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "7",
        "hostid": "30079",
        "manualinput": "2"
    },
    "id": 1
}
```


Odpowiedź:


```
{
    "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_seq=1 tt
        "debug": []
    },
    "id": 1
}
```


Źródło

CScript::execute() w *ui/include/classes/api/services/CScript.php*.

script.get

Opis

`integer/array script.get(object parameters)`

Ta metoda umożliwia pobieranie skryptów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane dla podanych grup hostów.
hostids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane dla podanych hostów.
scriptids	ID/array	Zwraca tylko skrypty o podanych ID.
usrgrpids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane przez użytkowników z podanych grup użytkowników.
selectHostGroups	query	Zwraca właściwość `hostgroups` z grupami hostów, dla których skrypt może być uruchamiany.
selectHosts	query	Zwraca właściwość `hosts` z hostami, dla których skrypt może być uruchamiany.
selectActions	query	Zwraca właściwość `actions` z akcjami, z którymi skrypt jest powiązany.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `scriptid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz wszystkie skrypty

Pobierz wszystkie skonfigurowane skrypty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.get",
```

```
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
```

```
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
```

```
            "parameters": [
                {
                    "name": "token",
                    "value": "{$WEBHOOK.TOKEN}"
                },
                {
                    "name": "host",
                    "value": "{HOST.HOST}"
                },
                {
                    "name": "v",
                    "value": "2.2"
                }
            ]
        },
        {
            "scriptid": "5",
            "name": "URL",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Go to {HOST.NAME}?",
            "type": "6",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://zabbix/ui/zabbix.php?action=latest.view&hostids[]={HOST.ID}",
            "new_window": "0",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator": "",
            "manualinput_validator_type": "0",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "6",
            "name": "URL with user input",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Open zabbix page {MANUALINPUT}?",
            "type": "6",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
```

```
            "privatekey": "",
            "menu_path": "",
            "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
            "new_window": "0",
            "manualinput": "1",
            "manualinput_prompt": "Select a page to open:",
            "manualinput_validator": "dashboard.view,script.list,actionlog.list",
            "manualinput_validator_type": "1",
            "parameters": []
        }
    ],
    "id": 1
}
```

Zobacz także

- Host
- Grupa hostów

Źródło

CScript::get() w *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyevents

Opis

`object script.getscriptsbyevents(object parameters)`

Ta metoda umożliwia pobranie wszystkich dostępnych skryptów dla danego zdarzenia lub określonego skryptu, jeśli podano ID skryptu. Gdy podano manualinput, zastępuje makro {MANUALINPUT} określoną wartością.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Metoda akceptuje obiekt lub tablicę obiektów z następującymi parametrami.

Parametr	Type	Opis
eventid	ID	ID zdarzenia, dla którego mają zostać zwrócone skrypty. Musi być unikalne. Zachowanie parametrów: - *wymagany*
scriptid	ID	ID skryptu do zwrócenia.
manualinput	string	Wartość podanej przez użytkownika wartości makra {MANUALINPUT}.

Zwracane wartości

`(object)` Zwraca obiekt z identyfikatorami zdarzeń jako właściwościami oraz tablicami dostępnych skryptów jako wartościami. Jeśli podano identyfikator skryptu, skojarzona wartość jest tablicą zawierającą określony skrypt.

Note:
Metoda automatycznie rozwinie makra w tekście `confirmation`, tekście `manualinput prompt` oraz w `url`.
Jeśli parametr manualinput zostanie podany, makro {MANUALINPUT} zostanie rozwinięte do określonej wartości.

Przykłady

Pobieranie skryptów według identyfikatorów zdarzeń

Pobierz wszystkie skrypty dostępne dla zdarzeń „632” i „614”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
         "eventid":  "632"
      },
      {
         "eventid":  "614"
      }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "632": [
            {
                "scriptid": "3",
                "name": "Wykryj system operacyjny",
                "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",
                "scope": "4",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "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",
                "scope": "4",
                "port": "",
```

```
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Strona Zabbix do otwarcia:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "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",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
```

```
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        }
    ],
    "614": [
        {
            "scriptid": "3",
            "name": "Wykryj system operacyjny",
            "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",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "1",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "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",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
```

```
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "Strona Zabbix do otwarcia:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "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",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
```

```
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            }
        ]
    },
    "id": 1
}
```

Pobieranie określonego skryptu z wartością manualinput.

Pobierz skrypt o ID „4” dla zdarzenia „632” z wartością manualinput „dashboard.view”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
        "eventid":  "632",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
      "632": [
        {
          "scriptid": "4",
          "name": "Open Zabbix page",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
          "groupid": "0",
          "description": "",
          "confirmation": "Are you sure you want to open page dashboard.view?",
          "type": "6",
          "execute_on": "2",
          "timeout": "30s",
          "scope": "4",
          "port": "",
          "authtype": "0",
          "username": "",
          "password": "",
          "publickey": "",
          "privatekey": "",
          "menu_path": "",
          "url": "http://localhost/ui/zabbix.php?action=dashboard.view",
          "new_window": "1",
          "manualinput": "1",
          "manualinput_prompt": "Zabbix page to open:",
          "manualinput_validator_type": "1",
          "manualinput_validator": "dashboard.view,discovery.view",
          "manualinput_default_value": "",
          "parameters": []
        }
      ]
```

```
    },
    "id": 1
}
```

Źródło

CScript::getScriptsByEvents() w *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyhosts

Opis

`object script.getscriptsbyhosts(object parameters)`

Ta metoda umożliwia pobranie wszystkich dostępnych skryptów dla danego hosta lub określonego skryptu, jeśli podano ID skryptu. Gdy podano manualinput, zastępuje makro {MANUALINPUT} określoną wartością.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Metoda akceptuje obiekt lub tablicę obiektów z następującymi parametrami.

Parametr	Typ	Opis
hostid	ID	ID hosta, dla którego mają zostać zwrócone skrypty. Musi być unikalne. Zachowanie parametrów: - *wymagane*
scriptid	ID	ID skryptu do zwrócenia.
manualinput	string	Wartość makra {MANUALINPUT} podana przez użytkownika.

Zwracane wartości

`(object)` Zwraca obiekt z identyfikatorami hostów jako właściwościami oraz tablicami dostępnych skryptów jako wartościami. Jeśli podano identyfikator skryptu, skojarzona wartość jest tablicą zawierającą określony skrypt.

Note:
Metoda automatycznie rozwinie makra w tekście `confirmation`, tekście `manualinput prompt` oraz w `url`.
Jeśli parametr manualinput zostanie podany, makro {MANUALINPUT} zostanie rozwinięte do określonej wartości.

Przykłady

Pobieranie skryptów według identyfikatorów hostów

Pobierz wszystkie skrypty dostępne na hostach „30079” i „30073”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
       {
          "hostid":  "30079"
       },
       {
          "hostid":  "30073"
       }
    ],
    "id": 1
}
```

Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "30079": [
            {
                "scriptid": "3",
                "name": "Wykryj system operacyjny",
                "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",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "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",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
```

```
        },
        {
            "scriptid": "4",
            "name": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "Strona Zabbix do otwarcia:",
            "manualinput_validator_type": "0",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        }
    ],
    "30073": [
        {
```

```
            "scriptid": "3",
            "name": "Wykryj system operacyjny",
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "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",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
```

```
                "usrgrpid": "0",
                "groupid": "0",
                "description": "",
                "confirmation": "Czy na pewno chcesz otworzyć stronę *UNKNOWN*?",
                "type": "6",
                "execute_on": "2",
                "timeout": "30s",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
                "new_window": "1",
                "manualinput": "1",
                "manualinput_prompt": "Strona Zabbix do otwarcia:",
                "manualinput_validator_type": "1",
                "manualinput_validator": "dashboard.view,discovery.view",
                "manualinput_default_value": "",
                "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",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            }
        ]
    },
    "id": 1
}
```


Pobierz określony skrypt z wartością manualinput.

Pobierz skrypt o ID „4” na hoście „30079” z wartością manualinput „dashboard.view”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
      {
        "hostid":  "30079",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
      "30079": [
        {
          "scriptid": "4",
          "name": "Open Zabbix page",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
          "groupid": "0",
          "description": "",
          "confirmation": "Are you sure you want to open page dashboard.view?",
          "type": "6",
          "execute_on": "2",
          "timeout": "30s",
          "scope": "2",
          "port": "",
          "authtype": "0",
          "username": "",
          "password": "",
          "publickey": "",
          "privatekey": "",
          "menu_path": "",
          "url": "http://localhost/ui/zabbix.php?action=dashboard.view",
          "new_window": "1",
          "manualinput": "1",
          "manualinput_prompt": "Zabbix page to open:",
          "manualinput_validator_type": "1",
          "manualinput_validator": "dashboard.view,discovery.view",
          "manualinput_default_value": "",
          "parameters": []
        }
      ]
    },
    "id": 1
}
```

Źródło

CScript::getScriptsByHosts() w *ui/include/classes/api/services/CScript.php*.

script.update

Opis

`object script.update(object/array scripts)`

Ta metoda umożliwia aktualizację istniejących skryptów.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object/array)` Właściwości skryptu, które mają zostać zaktualizowane.

Właściwość `scriptid` musi być zdefiniowana dla każdego skryptu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian. Wyjątkiem jest zmiana właściwości `type` z 5 (Webhook) na inną: właściwość `parameters` zostanie wyczyszczona.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych skryptów we właściwości `scriptids`.

Przykłady

Zmień polecenie skryptu

Zmień polecenie skryptu na "/bin/ping -c 10 {HOST.CONN} 2>&1".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
        "scriptid": "1",
        "command": "/bin/ping -c 10 {HOST.CONN} 2>&1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Zmień polecenie skryptu i dodaj ręczne dane wejściowe

Zmień polecenie skryptu na "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
      "scriptid": "1",
      "command": "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1",
      "manualinput": "1",
      "manualinput_prompt": "Określ liczbę pakietów ICMP do wysłania za pomocą polecenia ping",
      "manualinput_validator": "^(?:[1-9]|10)$",
      "manualinput_validator_type": "0",
      "manualinput_default_value": "10"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CScript::update() w *ui/include/classes/api/services/CScript.php*.

SLA

Ta klasa została zaprojektowana do pracy z obiektami SLA (Service Level Agreement) używanymi do oceny wydajności infrastruktury IT i usług biznesowych.

Odwołania do obiektów:

- SLA
- Harmonogram SLA
- Wykluczony przestój SLA
- Tag usługi SLA

Dostępne metody:

- sla.create - tworzenie nowych SLA
- sla.delete - usuwanie SLA
- sla.get - pobieranie SLA
- sla.getsli - pobieranie danych Service Level Indicator (SLI) dla SLA
- sla.update - aktualizowanie SLA

Obiekt SLA (Service Level Agreement)

Następujące obiekty są bezpośrednio powiązane z API `sla` (Service Level Agreement).

SLA

Obiekt SLA ma następujące właściwości.

Właściwość	Typ	Opis
slaid	ID	ID SLA. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa SLA. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
period	integer	Okres raportowania SLA. Możliwe wartości: 0 - dziennie; 1 - tygodniowo; 2 - miesięcznie; 3 - kwartalnie; 4 - rocznie. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
slo	float	Minimalny akceptowalny cel poziomu usług (Service Level Objective, SLO) wyrażony w procentach. Jeśli wskaźnik poziomu usług (Service Level Indicator, SLI) spadnie poniżej tej wartości, SLA jest uznawane za będące w stanie problemu/niespełnienia. Możliwe wartości: 0-100 (do 4 cyfr po przecinku). Zachowanie właściwości: - *wymagane* dla operacji tworzenia
effective_date	integer	Data wejścia SLA w życie. Możliwe wartości: znacznik czasu daty w UTC.
timezone	string	Strefa czasowa raportowania, na przykład: `Europe/London`, `UTC`. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
status	integer	Status SLA. Możliwe wartości: 0 - *(domyślnie)* SLA wyłączone; 1 - SLA włączone.
description	string	Opis SLA.

Harmonogram SLA

Obiekt harmonogramu SLA definiuje okresy, w których połączone usługi mają działać prawidłowo. Ma następujące właściwości.

Właściwość	Type	Opis
period_from	integer	Czas rozpoczęcia cyklicznego tygodniowego okresu czasu (włącznie). Możliwe wartości: liczba sekund (licząc od niedzieli). Zachowanie właściwości: - *wymagane*
period_to	integer	Czas zakończenia cyklicznego tygodniowego okresu czasu (wyłącznie). Możliwe wartości: liczba sekund (licząc od niedzieli). Zachowanie właściwości: - *wymagane*

Wyłączony czas niedostępności SLA

Obiekt wyłączonego czasu niedostępności definiuje okresy, w których powiązana usługa lub usługi mają być planowo wyłączone z działania, bez wpływu na SLI, np. podczas planowanej konserwacji. Ma następujące właściwości.

Właściwość	Type	Opis
name	string	Nazwa wykluczonego okresu niedostępności. Zachowanie właściwości: - *wymagane*

Właściwość	Type	Opis
period_from	integer	Czas rozpoczęcia wykluczonego okresu niedostępności (włącznie). Możliwe wartości: znacznik czasu. Zachowanie właściwości: - *wymagane*
period_to	integer	Czas zakończenia wykluczonego okresu niedostępności (wyłącznie). Możliwe wartości: znacznik czasu. Zachowanie właściwości: - *wymagane*

Znacznik usługi SLA

Obiekt znacznika usługi SLA łączy usługi, które mają zostać uwzględnione w obliczeniach SLA. Ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa tagu usługi SLA. Zachowanie właściwości: - *wymagane*
operator	integer	Operator tagu usługi SLA. Możliwe wartości: 0 - *(domyślnie)* Równa się; 2 - Zawiera.
value	string	Wartość tagu usługi SLA.

sla.create

Opis

`object sla.create(object/array SLAs)`

Ta metoda umożliwia tworzenie nowych obiektów SLA.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Obiekty SLA do utworzenia.

Oprócz standardowych właściwości SLA, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
service_tags	array	Tagi usług SLA, które mają zostać utworzone dla SLA. Zachowanie parametrów: - *wymagane*
schedule	array	Harmonogram SLA, który ma zostać utworzony dla SLA. Podanie pustego parametru będzie interpretowane jako harmonogram 24x7. Domyślnie: harmonogram 24x7.
excluded_downtimes	array	Wykluczone przestoje SLA, które mają zostać utworzone dla SLA.

Zwracane wartości

(`object`) Zwraca obiekt zawierający identyfikatory utworzonych SLA we właściwości `slaids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych SLA.

Przykłady

Tworzenie SLA

Poleć utworzenie wpisu SLA dla: * śledzenia dostępności usług związanych z silnikiem SQL; * niestandardowego harmonogramu obejmującego wszystkie dni robocze, z wyłączeniem ostatniej godziny w sobotę; * daty wejścia w życie przypadającej na ostatni dzień roku 2022; * planowanego przestoju trwającego 1 godzinę i 15 minut, rozpoczynającego się o północy 4 lipca; * włączonego obliczania tygodniowego raportu SLA; * minimalnego akceptowalnego SLO na poziomie 99,9995%.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.create",
    "params": [
        {
            "name": "Czas działania bazy danych",
            "slo": "99.9995",
            "period": "1",
            "timezone": "America/Toronto",
            "description": "Zapewnienie doskonałej dostępności głównych silników baz danych.",
            "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": "Wdrożenie aktualizacji wersji oprogramowania",
                    "period_from": "1648760400",
                    "period_to": "1648764900"
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
```

```
}
```

Źródło

CSla::create() w *ui/include/classes/api/services/CSla.php*.

sla.delete

Opis

`object sla.delete(array slaids)`

Ta metoda umożliwia usuwanie wpisów SLA.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(array)` ID SLA do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych SLA we właściwości `slaids`.

Przykłady

Usuwanie wielu SLA

Usuń dwa wpisy SLA.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CSla::delete() w *ui/include/classes/api/services/CSla.php*.

sla.get

Opis

`integer/array sla.get(object parameters)`

Metoda umożliwia pobieranie obiektów SLA zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
slaids	ID/array	Zwraca tylko SLA o podanych identyfikatorach.
serviceids	ID/array	Zwraca tylko SLA pasujące do określonych usług.
selectSchedule	query	Zwraca właściwość `schedule` z harmonogramami SLA. Obsługuje `count`.
selectExcludedDowntimes	query	Zwraca właściwość `excluded_downtimes` z wykluczonymi przestojami SLA. Obsługuje `count`.
selectServiceTags	query	Zwraca właściwość `service_tags` z tagami usług SLA. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `slaid`, `name`, `period`, `slo`, `effective_date`, `timezone`, `status`, `description`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobieranie wszystkich danych o SLA

Pobieranie danych o wszystkich SLA i ich właściwościach.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "sla.get",
"parametry": {
"wyjście": "rozszerz",
"selectSchedule": ["period_from", "period_to"],
"selectExcludedDowntimes": ["nazwa", "period_from", "period_to"],
"selectServiceTags": ["tag", "operator", "wartość"],
"preservekeys": prawda
},
```

```
"id": 1
}
```


Odpowiedź:


```
{
"jsonrpc": "2.0",
"wynik": {
"1": {
"slaid": "1",
"nazwa": "Czas działania bazy danych",
„period”: „1”,

slo”: „99.9995”,

effective_date”: „1672444800”,

timezone”: „America/Toronto”,

status”: „1”,

description”: „Zapewnij doskonały czas sprawności głównych silników baz danych SQL.”,

service_tags”: [
{
„tag”: „database”,

„operator”: „0”,

„value”: „mysql”
},
{
„tag”: „database”,

operator”: „0”,

„value”: „postgresql”
}
],

schedule”: [
{
„period_from”: „0”,

„period_to”: „601200”
}
],

excluded_downtimes”: [
{
„name”: „Wdrażanie aktualizacji wersji oprogramowania”,
„okres_od”: „1648760400”,
„okres_do”: „1648764900”
}
]
}
},
„id”: 1
}
```


Źródło

CSla:get() w *ui/include/classes/api/services/CSla.php*.

sla.getsli

Opis

`object sla.getsli(object parameters)`

Ta metoda umożliwia obliczenie danych wskaźnika poziomu usługi (SLI) dla umowy o gwarantowanym poziomie usług (SLA).

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikator SLA, okresy raportowania oraz opcjonalnie identyfikatory usług, dla których ma zostać obliczony wskaźnik SLI.

Parametr	Typ	Opis
slaid	ID	Identyfikator SLA, dla którego mają zostać zwrócone informacje o dostępności. Zachowanie parametru: - *wymagany*
period_from	timestamp	Początkowy znacznik czasu (włącznie), dla którego ma zostać zaraportowany wskaźnik SLI. Możliwe wartości: znacznik czasu Unix.
period_to	timestamp	Końcowy znacznik czasu (włącznie), dla którego ma zostać zaraportowany wskaźnik SLI. Możliwe wartości: znacznik czasu Unix.
periods	integer	Liczba okresów do zaraportowania. Możliwe wartości: 1-100
serviceids	ID/array	Identyfikatory usług, dla których ma zostać zwrócony wskaźnik SLI.

Podział na okresy

Poniższa tabela przedstawia układ zwracanych wycinków okresów na podstawie kombinacji parametrów.

Note:
Zwracane okresy nie poprzedzają pierwszego dostępnego okresu (na podstawie daty wejścia SLA w życie) i nie wykraczają poza bieżący okres.

Parametry			Zwracane okresy
period_from	**period_to**	**periods**	
-	-	-	Ostatnie 20 okresów, w tym bieżący.
-	-	N	Ostatnie N okresów.
-	specified	-	Ostatnie 20 okresów przed `period_to`.
-	specified	N	Ostatnie N okresów przed `period_to`.
specified	-	-	Pierwsze 20 okresów zaczynając od `period_from`.
specified	-	N	Pierwsze N okresów zaczynając od `period_from`.
specified	specified	-	Do 100 okresów w określonym zakresie.
specified	specified	N	N okresów w określonym zakresie.

Zwracane wartości

`(object)` Zwraca wyniki obliczenia.

Właściwość	Type	Opis
periods	array	Lista raportowanych okresów. Każdy raportowany okres jest reprezentowany jako obiekt składający się z: - `period_from` - (`timestamp`) Początkowy znacznik czasu (włącznie) raportowanego okresu. - `period_to` - (`timestamp`) Końcowy znacznik czasu (wyłącznie) raportowanego okresu. Okresy są sortowane według `period_from`, przy czym najwcześniejsze okresy pojawiają się jako pierwsze.
serviceids	array	Lista identyfikatorów usług w raportowanych okresach. Kolejność sortowania listy nie jest określona. Nawet jeśli parametr `serviceids` został przekazany do metody `sla.getsli`.
sli	array	Dane SLI (jako **tablica dwuwymiarowa**) dla każdego raportowanego okresu i usługi. Indeks właściwości `periods` jest używany jako **pierwszy** wymiar właściwości `sli`. Indeks właściwości `serviceids` jest używany jako **drugi** wymiar właściwości `sli`.

Dane SLI

Dane SLI zwracane dla każdego raportowanego okresu i serwisu obejmują:

Właściwość	Type	Opis
uptime	integer	Czas, przez jaki serwis znajdował się w stanie *OK* podczas zaplanowanego czasu działania, pomniejszony o wykluczone przestoje.
downtime	integer	Czas, przez jaki serwis znajdował się w stanie *not OK* podczas zaplanowanego czasu działania, pomniejszony o wykluczone przestoje.
sli	float	SLI (procent całkowitego czasu działania), na podstawie czasu działania i przestoju.
error_budget	integer	Budżet błędów (w sekundach), na podstawie SLI i SLO.
excluded_downtimes	array	Tablica wykluczonych przestojów w tym okresie raportowania. Każdy obiekt będzie zawierał następujące parametry: - `name` - (`string`) Nazwa wykluczonego przestoju. - `period_from` - (`timestamp`) Znacznik czasu rozpoczęcia (włącznie) wykluczonego przestoju. - `period_to` - (`timestamp`) Znacznik czasu zakończenia (wyłącznie) wykluczonego przestoju. Wykluczone przestoje są sortowane według `period_from`, przy czym najwcześniejsze okresy są wyświetlane jako pierwsze.

Przykłady

Obliczanie SLI dla dziennego SLA

Pobierz dane SLI dla usług o identyfikatorach „1” i „4”, które są powiązane ze SLA o identyfikatorze „1”. Pobierz dane dla jednego okresu do „1761861599” (30 paź 2025 23:59:59 GMT+0200). Ponieważ okres raportowania SLA jest dzienny, dane SLI są pobierane od „1761775200” (30 paź 2025 00:00:00 GMT+0200) do „1761861600” (31 paź 2025 00:00:00 GMT+0200).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
```

```
        "slaid": "1",
        "serviceids": [
            1,
            4
        ],
        "periods": 1,
        "period_to": 1761861599
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "periods": [
            {
                "period_from": 1761775200,
                "period_to": 1761861600
            }
        ],
        "serviceids": [
            1,
            4
        ],
        "sli": [
            [
                {
                    "uptime": 43843,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance OCT",
                            "period_from": 1761825600,
                            "period_to": 1761829200
                        }
                    ]
                },
                {
                    "uptime": 32225,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                }
            ]
        ]
    },
    "id": 1
}
```

Obliczanie SLI dla miesięcznego SLA

Pobierz dane SLI o usługach o identyfikatorach „50”, „60” i „70”, które są powiązane ze SLA o identyfikatorze „5”. Pobierz dane dla trzech okresów rozpoczynających się od „1635724800” (01 lis 2021 00:00:00 UTC). Ponieważ okres raportowania SLA jest miesięczny, dane SLI są pobierane dla następujących trzech miesięcy:

- Od „1635724800” (01 lis 2021 00:00:00 UTC) do „1638316800” (01 gru 2021 00:00:00 UTC)
- Od „1638316800” (01 gru 2021 00:00:00 UTC) do „1640995200” (01 sty 2022 00:00:00 UTC)
- Od „1640995200” (01 sty 2022 00:00:00 UTC) do „1643673600” (01 lut 2022 00:00:00 UTC)

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "5",
        "serviceids": [
            50,
            60,
            70
        ],
        "periods": 3,
        "period_from": 1635724800
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "Maintenance Nov 25 - Dec 01",
                            "period_from": 1637836212,
                            "period_to": 1638316800
                        }
                    ]
                },
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Nov 25 - Dec 01",
```

```
                    "period_from": 1637836212,
                    "period_to": 1638316800
                }
            ]
        },
        {
            "uptime": 1186212,
            "downtime": 0,
            "sli": 100,
            "error_budget": 0,
            "excluded_downtimes": [
                {
                    "name": "Maintenance Nov 25 - Dec 01",
                    "period_from": 1637836212,
                    "period_to": 1638316800
                }
            ]
        }
    ],
    [
        {
            "uptime": 1147548,
            "downtime": 0,
            "sli": 100,
            "error_budget": 0,
            "excluded_downtimes": [
                {
                    "name": "Maintenance Dec 02 - Dec 10",
                    "period_from": 1638439200,
                    "period_to": 1639109652
                }
            ]
        },
        {
            "uptime": 1147548,
            "downtime": 0,
            "sli": 100,
            "error_budget": 0,
            "excluded_downtimes": [
                {
                    "name": "Maintenance Dec 02 - Dec 10",
                    "period_from": 1638439200,
                    "period_to": 1639109652
                }
            ]
        },
        {
            "uptime": 1147548,
            "downtime": 0,
            "sli": 100,
            "error_budget": 0,
            "excluded_downtimes": [
                {
                    "name": "Maintenance Dec 02 - Dec 10",
                    "period_from": 1638439200,
                    "period_to": 1639109652
                }
            ]
        },
        {
            "uptime": 1147548,
            "downtime": 0,
            "sli": 100,
            "error_budget": 0,
            "excluded_downtimes": [
                {
                    "name": "Maintenance Dec 02 - Dec 10",
                    "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
}
```


Źródło

CSla::getSli() w *ui/include/classes/api/services/CSla.php*

sla.update

Opis

`object sla.update(object/array slaids)`

Ta metoda umożliwia aktualizację istniejących wpisów SLA.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object/array)` Właściwości SLA do zaktualizowania.

Właściwość `slaid` musi być zdefiniowana dla każdego SLA, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości SLA, metoda akceptuje następujące parametry.

Parametr	Type	Opis
service_tags	array	Tagi usług SLA zastępujące bieżące tagi usług SLA.
schedule	array	Harmonogram SLA zastępujący bieżący. Podanie pustego parametru będzie interpretowane jako harmonogram 24x7.
excluded_downtimes	array	Wykluczone przestoje SLA zastępujące bieżące.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych SLA we właściwości `slaids`.

Przykłady

Aktualizacja tagów usługi

Ustaw SLA o ID „5” tak, aby było obliczane w interwałach miesięcznych dla usług związanych z NoSQL, bez zmiany jego harmonogramu ani wykluczonych przestojów; ustaw SLO na 95%.

Żądanie:

```
{
    "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"
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Zmiana harmonogramu SLA

Przełącz SLA o ID „5” na harmonogram 24x7.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": {
        "slaid": "5",
        "schedule": []
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
```

```
    },
    "id": 1
}
```

Zmiana wykluczonych przestojów dla SLA

Dodaj planowany 4-godzinny przestój związany z rozbudową pamięci RAM w dniu 6 kwietnia 2022 r., zachowując jednocześnie (należy go zdefiniować ponownie) wcześniej istniejący plan aktualizacji oprogramowania z 4 lipca dla SLA o ID „5”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.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"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CSla::update() w *ui/include/classes/api/services/CSla.php*.

Szablon

Ta klasa jest przeznaczona do pracy z szablonami.

Odwołania do obiektów:

- Template
- Template tag

Dostępne metody:

- template.create - tworzenie nowych szablonów
- template.delete - usuwanie szablonów
- template.get - pobieranie szablonów
- template.massadd - dodawanie powiązanych obiektów do szablonów
- template.massremove - usuwanie powiązanych obiektów z szablonów
- template.massupdate - zastępowanie lub usuwanie powiązanych obiektów z szablonów
- template.update - aktualizowanie szablonów

Obiekt szablonu

Następujące obiekty są bezpośrednio powiązane z API `template`.

Szablon

Obiekt szablonu ma następujące właściwości.

Właściwość	Typ	Opis
templateid	ID	ID szablonu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	text	Opis szablonu.
host	string	Nazwa techniczna szablonu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Widoczna nazwa szablonu. Domyślnie: wartość właściwości `host`.
readme	text	Instrukcje konfiguracji specyficzne dla szablonu, wyświetlane w Kreatorze hosta. Obsługuje formatowanie Markdown.
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych szablonów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.
vendor_name	string	Nazwa dostawcy szablonu. W przypadku operacji tworzenia zarówno `vendor_name`, jak i `vendor_version` powinny być ustawione albo pozostawione puste. W przypadku operacji aktualizacji `vendor_version` może pozostać puste, jeśli ma wartość w bazie danych.
vendor_version	string	Wersja dostawcy szablonu. W przypadku operacji tworzenia zarówno `vendor_name`, jak i `vendor_version` powinny być ustawione albo pozostawione puste. W przypadku operacji aktualizacji `vendor_name` może pozostać puste, jeśli ma wartość w bazie danych.
wizard_ready	integer	Określa, czy szablon jest dostępny do wyboru w Kreatorze hosta. Możliwe wartości: 0 - *(domyślnie)* Niedostępny. 1 - Dostępny.

Znacznik szablonu

Obiekt znacznika szablonu ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa znacznika szablonu. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika szablonu.

template.create

Opis

```
object template.create(object/array templates)
```

Ta metoda umożliwia tworzenie nowych szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Szablony do utworzenia.

Oprócz standardowych właściwości szablonu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
groups	object/array	Grupy szablonów, do których należy dodać szablon. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametrów: - *wymagane*
tags	object/array	Tagi szablonu.
templates	object/array	Szablony, które mają zostać podlinkowane do szablonu. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla szablonu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych szablonów we właściwości `templateids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych szablonów.

Przykłady

Tworzenie szablonu

Utwórz szablon z tagami i podłącz dwa szablony do tego szablonu.

Żądanie:

```
{
    "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}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "11117"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::create() w *ui/include/classes/api/services/CTemplate.php*.

template.delete

Opis

`object template.delete(array templateIds)`

Ta metoda umożliwia usuwanie szablonów.

Usunięcie szablonu spowoduje usunięcie wszystkich encji szablonu (pozycje, wyzwalacze, wykresy itd.). Aby pozostawić encje szablonu na hostach, ale usunąć sam szablon, najpierw odłącz szablon od wymaganych hostów za pomocą jednej z tych metod: template.update, template.massupdate, host.update, host.massupdate.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji znajdziesz w sekcji Role użytkowników.

Parametry

`(array)` ID szablonów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych szablonów we właściwości `templateids`.

Przykłady

Usuwanie wielu szablonów

Usuń dwa szablony.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::delete() w *ui/include/classes/api/services/CTemplate.php*.

template.get

Opis

`integer/array template.get(object parameters)`

Metoda umożliwia pobieranie szablonów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
templateids	ID/array	Zwracaj tylko szablony o podanych ID szablonów.
groupids	ID/array	Zwracaj tylko szablony należące do podanych grup szablonów.
parentTemplateids	ID/array	Zwracaj tylko szablony, z którymi podany szablon jest powiązany.
hostids	ID/array	Zwracaj tylko szablony powiązane z podanymi hostami/szablonami.
graphids	ID/array	Zwracaj tylko szablony zawierające podane wykresy.
itemids	ID/array	Zwracaj tylko szablony zawierające podane pozycje.
triggerids	ID/array	Zwracaj tylko szablony zawierające podane wyzwalacze.
with_items	flag	Zwracaj tylko szablony, które mają pozycje.
with_triggers	flag	Zwracaj tylko szablony, które mają wyzwalacze.
with_graphs	flag	Zwracaj tylko szablony, które mają wykresy.
with_httptests	flag	Zwracaj tylko szablony, które mają scenariusze web.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* I/Lub; 2 - Lub.
tags	object/array	Zwracaj tylko szablony z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie szablony. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
selectTags	query	Zwraca tagi szablonu w właściwości `tags`.
selectDiscoveryRules	query	Zwraca właściwość `discoveryRules` z regułami LLD szablonu. Obsługuje `count`.
selectHosts	query	Zwraca hosty powiązane z szablonem w właściwości `hosts`. Obsługuje `count`.
selectTemplateGroups	query	Zwraca grupy szablonów, do których należy szablon, w właściwości `templategroups`.
selectTemplates	query	Zwraca szablony, z którymi podany szablon jest powiązany, w właściwości `templates`. Obsługuje `count`.

Parameter	Type	Description
selectParentTemplates	query	Zwraca szablony powiązane z podanym szablonem w właściwości `parentTemplates`. Obsługuje `count`.
selectHttpTests	query	Zwraca scenariusze web z szablonu w właściwości `httpTests`. Obsługuje `count`.
selectItems	query	Zwraca pozycje z szablonu w właściwości `items`. Obsługuje `count`.
selectTriggers	query	Zwraca wyzwalacze z szablonu w właściwości `triggers`. Obsługuje `count`.
selectGraphs	query	Zwraca wykresy z szablonu w właściwości `graphs`. Obsługuje `count`.
selectMacros	query	Zwraca makra z szablonu w właściwości `macros`.
selectDashboards	query	Zwraca dashboardy z szablonu w właściwości `dashboards`. Obsługuje `count`.
selectValueMaps	query	Zwraca właściwość `valuemaps` z mapami wartości szablonu.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectTemplates` - wyniki będą sortowane według `name`; `selectHosts` - sortowane według `host`; `selectParentTemplates` - sortowane według `host`; `selectItems` - sortowane według `name`; `selectDiscoveryRules` - sortowane według `name`; `selectTriggers` - sortowane według `description`; `selectGraphs` - sortowane według `name`; `selectDashboards` - sortowane według `name`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectDiscoveries	query	Zwraca odkrycia niskiego poziomu z szablonu w właściwości `discoveries`. Obsługuje `count`. To zapytanie jest **przestarzałe**, zamiast niego użyj `selectDiscoveryRules`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie szablonów według nazwy

Pobierz wszystkie dane dotyczące dwóch szablonów o nazwach ”Linux” i ”Windows”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent",
                "Windows by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "proxyid": "0",
            "host": "Linux by Zabbix agent",
            "status": "3",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Linux by Zabbix agent",
            "flags": "0",
            "templateid": "10001",
            "description": "To jest oficjalny szablon dla systemu Linux. Wymaga agent Zabbix 7.4 lub nowsz
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "tls_psk_identity": "",
            "tls_psk": "",
            "custom_interfaces": "0",
            "uuid": "f8f7908280354f2abeed07dc788c3747",
            "vendor_name": "Zabbix",
            "vendor_version": "7.4-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "## Omówienie\r\n\r\nTo jest oficjalny szablon dla systemu Linux. Wymaga agent Zabbi
        },
        {
            "proxyid": "0",
            "host": "Windows by Zabbix agent",
            "status": "3",
            "ipmi_authtype": "-1",
            "ipmi_privilege": "2",
            "ipmi_username": "",
```

```
            "ipmi_password": "",
            "maintenanceid": "0",
            "maintenance_status": "0",
            "maintenance_type": "0",
            "maintenance_from": "0",
            "name": "Windows by Zabbix agent",
            "flags": "0",
            "templateid": "10081",
            "description": "To jest oficjalny szablon dla systemu Windows. Wymaga agent Zabbix 7.4 lub now
            "tls_connect": "1",
            "tls_accept": "1",
            "tls_issuer": "",
            "tls_subject": "",
            "tls_psk_identity": "",
            "tls_psk": "",
            "custom_interfaces": "0",
            "uuid": "13b06904a6bf41cbb795e3193d896340",
            "vendor_name": "Zabbix",
            "vendor_version": "7.4-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "## Omówienie\r\n\r\nTo jest oficjalny szablon dla systemu Windows. Wymaga agent Zab
        }
    ],
    "id": 1
}
```


Pobieranie grup szablonów

Pobierz grupy szablonów, do których należy szablon „Linux by Zabbix agent”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTemplateGroups": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "templategroups": [
                {
                    "groupid": "10",
                    "name": "Templates/Operating systems",
                    "uuid": "846977d1dfed4968bc5f8bdb363285bc"
                }
            ]
        }
    ],
```

```
    "id": 1
}
```

Pobieranie hostów według szablonu

Pobierz hosty, które mają podłączony szablon „10001” (*Linux by Zabbix agent*).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "templateid",
        "templateids": "10001",
        "selectHosts": ["hostid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie według tagów szablonu

Pobierz szablony, które mają tag ”host-name” równy ”{HOST.NAME}”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}",
                "operator": 1
            }
        ]
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10402",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "{HOST.NAME}"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Grupa szablonów
- Szablon
- Makro użytkownika
- Interfejs hosta

Źródło

CTemplate::get() w *ui/include/classes/api/services/CTemplate.php*.

template.massadd

Opis

`object template.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do podanych szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory szablonów do zaktualizowania oraz obiekty do dodania do szablonów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
templates	object/array	Szablony do zaktualizowania. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`. Zachowanie parametrów: - *wymagane*
groups	object/array	Grupy szablonów do dodania do podanych szablonów. Grupy szablonów muszą mieć zdefiniowaną tylko właściwość `groupid`.
macros	object/array	Makra użytkownika do utworzenia dla podanych szablonów.
templates_link	object/array	Szablony do podlinkowania do podanych szablonów. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości templateids.

Przykłady

Powiąż grupę z szablonami

Dodaj grupę szablonów „2” do dwóch szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Podłącz dwa szablony do szablonu

Podłącz szablony „10106” i „10104” do szablonu „10073”.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10073"
            }
        ],
        "templates_link": [
            {
                "templateid": "10106"
            },
            {
                "templateid": "10104"
            }
        ]
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10073"
        ]
    },
    "id": 1
}
```

Zobacz także

- template.update
- Host
- Grupa szablonów
- Makro użytkownika

Źródło

CTemplate::massAdd() w *ui/include/classes/api/services/CTemplate.php*.

template.massremove

Opis

`object template.massremove(object parameters)`

Ta metoda umożliwia usunięcie powiązanych obiektów z wielu szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji User roles.

Parametry

`(object)` Parametry zawierające identyfikatory szablonów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Typ	Opis
templateids	ID/array	Identyfikatory szablonów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
groupids	ID/array	Identyfikatory grup szablonów, z których należy usunąć podane szablony.
macros	string/array	Identyfikatory makr użytkownika, które mają zostać usunięte z podanych szablonów.
templateids_clear	ID/array	Identyfikatory szablonów, które mają zostać odłączone i wyczyszczone z podanych szablonów (nadrzędnych).
templateids_link	ID/array	Identyfikatory szablonów, które mają zostać odłączone od podanych szablonów (nadrzędnych).

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości `templateids`.

Przykłady

Usuwanie szablonów z grupy

Usuń dwa szablony z grupy „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": [
            "10085",
            "10086"
        ],
        "groupids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Odłączanie szablonów od hosta

Odłącz szablony „10106” i „10104” od szablonu „10085”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": "10085",
        "templateids_link": [
            "10106",
            "10104"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085"
        ]
    },
    "id": 1
}
```

Zobacz także

- template.update
- Makro użytkownika

Źródło

CTemplate::massRemove() w *ui/include/classes/api/services/CTemplate.php*.

template.massupdate

Opis

`object template.massupdate(object parameters)`

Ta metoda umożliwia jednoczesne zastępowanie lub usuwanie powiązanych obiektów oraz aktualizowanie właściwości wielu szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory szablonów do aktualizacji oraz obiekty, które mają zostać zastąpione dla szablonów.

Metoda przyjmuje następujące parametry.

Parameter	Type	Description
templates	object/array	Szablony do zaktualizowania. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`. Zachowanie parametru: - *required*
groups	object/array	Grupy szablonów do zastąpienia bieżących grup szablonów, do których należą szablony. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
macros	object/array	Makra użytkownika do zastąpienia wszystkich bieżących makr użytkownika w podanych szablonach.
templates_clear	object/array	Szablony do odłączenia i wyczyszczenia z podanych szablonów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.
templates_link	object/array	Szablony do zastąpienia aktualnie podłączonych szablonów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów w właściwości `templateids`.

Przykłady

Odłączanie szablonu

Odłącz i wyczyść szablon ”10091” od podanych szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massupdate",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
```

```
            }
        ],
        "templates_clear": [
            {
                "templateid": "10091"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Zastępowanie makr użytkownika

Zastąp wszystkie makra użytkownika podanym makrem użytkownika w wielu szablonach.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "template.massupdate",
    "params": {
        "templates": [
            {
                "templateid": "10074"
            },
            {
                "templateid": "10075"
            },
            {
                "templateid": "10076"
            },
            {
                "templateid": "10077"
            }
        ],
        "macros": [
            {
                "macro": "{$AGENT.TIMEOUT}",
                "value": "5m",
                "description": "Timeout after which agent is considered unavailable. Works only for agents
            }
        ]
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10074",
```

```
            "10075",
            "10076",
            "10077"
        ]
    },
    "id": 1
}
```

Zobacz także

- template.update
- template.massadd
- Grupa szablonów
- Makro użytkownika

Źródło

CTemplate::massUpdate() w *ui/include/classes/api/services/CTemplate.php*.

template.update

Opis

`object template.update(object/array templates)`

Ta metoda umożliwia aktualizację istniejących szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości szablonu do zaktualizowania.

Właściwość `templateid` musi być zdefiniowana dla każdego szablonu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości szablonu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy szablonów zastępujące bieżące grupy szablonów, do których należą szablony. Dla grup szablonów musi być zdefiniowana tylko właściwość `groupid`.
tags	object/array	Tagi szablonu zastępujące bieżące tagi szablonu.
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika w podanych szablonach.
templates	object/array	Szablony zastępujące aktualnie podlinkowane szablony. Szablony, które nie zostaną przekazane, zostaną jedynie odlinkowane. Dla szablonów musi być zdefiniowana tylko właściwość `templateid`.
templates_clear	object/array	Szablony do odlinkowania i wyczyszczenia z podanych szablonów. Dla szablonów musi być zdefiniowana tylko właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości `templateids`.

Przykłady

Zmiana standardowych właściwości szablonu

Zmień nazwę techniczną szablonu na „Linux by Zabbix agent Custom”, nazwę widoczną na „My template” oraz zaktualizuj opis szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "host": "Linux by Zabbix agent Custom",
        "name": "My template",
        "description": "This is a custom Linux template."
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie grup szablonów

Zastąp wszystkie grupy szablonów dla danego szablonu inną grupą.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "groups": [
            {
                "groupid": "24"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów szablonu

Zastąp wszystkie tagi szablonu innym tagiem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
```

```
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr szablonu

Zastąp wszystkie makra szablonu innymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "macros": [
            {
                "macro": "{$MY_MACRO}",
                "value": "new_value"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizacja powiązanych szablonów szablonu

Odłącz (bez czyszczenia) wszystkie szablony od podanego szablonu i powiąż z nim inny.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates": [
            {
```

```
                "templateid": "10087"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Odłączanie i czyszczenie powiązanych szablonów

Odłącz i wyczyść podany szablon od określonego powiązanego szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates_clear": [
            {
                "templateid": "10087"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::update() w *ui/include/classes/api/services/CTemplate.php*.

Token

Ta klasa jest przeznaczona do pracy z tokenami.

Odwołania do obiektów:

- Token

Dostępne metody:

- token.create - tworzenie nowych tokenów
- token.delete - usuwanie tokenów
- token.get - pobieranie tokenów

- token.update - aktualizowanie tokenów
- token.generate - generowanie tokenów

Obiekt tokena

Następujące obiekty są bezpośrednio powiązane z API `token`.

Token

Obiekt tokenu ma następujące właściwości.

Właściwość	Typ	Opis
tokenid	ID	ID tokenu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa tokenu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	text	Opis tokenu.
userid	ID	ID użytkownika, do którego przypisano token. Domyślnie: *bieżący użytkownik*. Zachowanie właściwości: - *stałe*
lastaccess	timestamp	Data i godzina ostatniego uwierzytelnienia tokenu. ”0”, jeśli token nigdy nie został uwierzytelniony. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Status tokenu. Możliwe wartości: 0 - *(domyślnie)* token włączony; 1 - token wyłączony.
expires_at	timestamp	Data i godzina wygaśnięcia tokenu. ”0” dla tokenów, które nigdy nie wygasają.
created_at	timestamp	Data i godzina utworzenia tokenu. Zachowanie właściwości: - *tylko do odczytu*
creator_userid	ID	ID użytkownika, który utworzył token. Zachowanie właściwości: - *tylko do odczytu*

token.create

Opis

`object token.create(object/array tokens)`

Ta metoda umożliwia tworzenie nowych tokenów.

Note:
Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Attention:
Token utworzony tą metodą musi również zostać wygenerowany, zanim będzie można go użyć.

Parametry

`(object/array)` Tokeny do utworzenia.

Metoda akceptuje tokeny ze standardowymi właściwościami tokenu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych tokenów we właściwości `tokenids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych tokenów.

Przykłady

Utwórz token

Utwórz włączony token, który nigdy nie wygasa i uwierzytelnia użytkownika o ID 2.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "userid": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188"
        ]
    },
    "id": 1
}
```

Utwórz wyłączony token, który wygasa 21 stycznia 2021 roku. Ten token będzie uwierzytelniał bieżącego użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "status": "1",
        "expires_at": "1611238072"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "189"
        ]
    },
    "id": 1
}
```

Źródło

CToken::create() w *ui/include/classes/api/services/CToken.php*.

token.delete

Opis

`object token.delete(array tokenids)`

Ta metoda umożliwia usuwanie tokenów.

> **Note:**
> Uprawnienie *Zarządzanie Tokenami API* permission jest wymagane dla roli użytkownika, aby zarządzać tokenami dla innych użytkowników.

Parametry

`(array)` ID tokenów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych tokenów we właściwości `tokenids`.

Przykłady

Usuń wiele tokenów

Usuń dwa tokeny.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "token.delete",
"params": [
"188",
"192"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"tokenids": [
"188",
"192"
]
},
"id": 1
}
```

Źródło

CToken::delete() w *ui/include/classes/api/services/CToken.php*.

token.generate

Opis

`object token.generate(array tokenids)`

Ta metoda umożliwia generowanie tokenów.

Note:
Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Attention:
Token może zostać wygenerowany tą metodą tylko wtedy, gdy został utworzony.

Parametry

`(array)` ID tokenów do wygenerowania.

Zwracane wartości

`(array)` Zwraca tablicę obiektów zawierających identyfikator wygenerowanego tokenu we właściwości `tokenid` oraz wygenerowany ciąg autoryzacyjny we właściwości `token`.

Właściwość	Type	Opis
tokenid	ID	Identyfikator tokenu.
token	string	Wygenerowany ciąg autoryzacyjny dla tego tokenu.

Przykłady

Wygeneruj wiele tokenów

Wygeneruj dwa tokeny.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.generate",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "tokenid": "1",
            "token": "bbcfce79a2d95037502f7e9a534906d3466c9a1484beb6ea0f4e7be28e8b8ce2"
        },
        {
            "tokenid": "2",
            "token": "fa1258a83d518eabd87698a96bd7f07e5a6ae8aeb8463cae33d50b91dd21bd6d"
        }
    ],
    "id": 1
}
```

Źródło

CToken::generate() w *ui/include/classes/api/services/CToken.php*.

token.get

Opis

`integer/array token.get(object parameters)`

Ta metoda umożliwia pobieranie tokenów zgodnie z podanymi parametrami.

Note:
Tylko użytkownik typu *Super admin* może wyświetlać tokeny innych użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
tokenids	ID/array	Zwracaj tylko tokeny o podanych identyfikatorach.
userids	ID/array	Zwracaj tylko tokeny utworzone dla podanych użytkowników.
token	string	Zwracaj tylko tokeny utworzone dla podanego *tokenu uwierzytelniającego*.
valid_at	timestamp	Zwracaj tylko tokeny, które są ważne (nie wygasły) w podanej dacie i czasie.
expired_at	timestamp	Zwracaj tylko tokeny, które są wygasłe (nieważne) w podanej dacie i czasie.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `tokenid`, `name`, `lastaccess`, `status`, `expires_at`, `created_at`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie tokena

Pobierz wszystkie dane tokena o ID „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.get",
    "params": {
        "output": "extend",
        "tokenids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CToken::get() w *ui/include/classes/api/services/CToken.php*.

token.update

Opis

`object token.update(object/array tokens)`

Ta metoda umożliwia aktualizację istniejących tokenów.

> **Note:**
> Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Parametry

`(object/array)` Właściwości tokenów do zaktualizowania.

Właściwość `tokenid` musi być zdefiniowana dla każdego tokenu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje tokeny ze standardowymi właściwościami tokenu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych tokenów we właściwości `tokenids`.

Przykłady

Usuń datę wygaśnięcia tokena

Usuń datę wygaśnięcia z tokena.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.update",
    "params": {
        "tokenid": "2",
        "expires_at": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "2"
        ]
    },
```

```
    "id": 1
}
```

Źródło

CToken::update() w *ui/include/classes/api/services/CToken.php*.

Trend

Ta klasa została zaprojektowana do pracy z danymi trendów.

Odwołania do obiektów:

- Trend zmiennoprzecinkowy
- Trend całkowitoliczbowy

Dostępne metody:

- trend.get — pobieranie trendów

Obiekt trendu

Następujące obiekty są bezpośrednio powiązane z API `trend`.

Note:
Obiekty trendów różnią się w zależności od typu informacji elementu. Są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Trend float

Obiekt trendu float ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Znacznik czasu godziny, dla której obliczono wartość. Na przykład znacznik czasu „04:00:00” oznacza wartości obliczone dla okresu „04:00:00-04:59:59”.
itemid	ID	ID powiązanej pozycji.
num	integer	Liczba wartości, które były dostępne dla danej godziny.
value_min	float	Minimalna wartość godzinowa.
value_avg	float	Średnia wartość godzinowa.
value_max	float	Maksymalna wartość godzinowa.

Trend całkowitoliczbowy

Obiekt trendu całkowitoliczbowego ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Znacznik czasu godziny, dla której obliczono wartość. Na przykład znacznik czasu „04:00:00” oznacza wartości obliczone dla okresu „04:00:00-04:59:59”.
itemid	ID	ID powiązanej pozycji.
num	integer	Liczba wartości, które były dostępne dla danej godziny.
value_min	integer	Minimalna wartość godzinowa.
value_avg	integer	Średnia wartość godzinowa.
value_max	integer	Maksymalna wartość godzinowa.

trend.get

Opis

`integer/array trend.get(object parameters)`

Ta metoda umożliwia pobieranie danych trendów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
itemids	ID/array	Zwracaj tylko trendy dla podanych identyfikatorów pozycji.
time_from	timestamp	Zwracaj tylko wartości, które zostały zebrane po podanym czasie lub o podanym czasie.
time_till	timestamp	Zwracaj tylko wartości, które zostały zebrane przed podanym czasem lub o podanym czasie.
countOutput	boolean	Policz liczbę pobranych obiektów.
limit	integer	Ogranicz liczbę pobranych obiektów.
output	query	Ustaw właściwości obiektu trendu, które mają zostać zwrócone.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych trendów pozycji

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trend.get",
    "params": {
        "output": [
            "itemid",
            "clock",
            "num",
            "value_min",
            "value_avg",
            "value_max"
        ],
        "itemids": [
            "23715"
        ],
        "limit": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23715",
            "clock": "1446199200",
            "num": "60",
            "value_min": "0.165",
            "value_avg": "0.2168",
            "value_max": "0.35"
        }
```

```
    ],
    "id": 1
}
```

Źródło

CTrend::get() w *ui/include/classes/api/services/CTrend.php*.

Typ nośnika

Ta klasa jest przeznaczona do pracy z typami nośników.

Odwołania do obiektów:

- Typ nośnika
  - Parametry webhooka
  - Parametry skryptu
- Szablon wiadomości

Dostępne metody:

- mediatype.create - utwórz nowe typy nośników
- mediatype.delete - usuń typy nośników
- mediatype.get - pobierz typy nośników
- mediatype.update - zaktualizuj typy nośników

Obiekt typu mediów

Następujące obiekty są bezpośrednio powiązane z API `mediatype`.

Typ nośnika

Obiekt typu nośnika ma następujące właściwości.

Property	Type	Description
mediatypeid	ID	ID typu nośnika. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa typu nośnika. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Transport używany przez typ nośnika. Możliwe wartości: 0 - E-mail; 1 - Skrypt; 2 - SMS; 4 - Webhook. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
exec_path	string	Nazwa pliku skryptu (np. notification.sh), który znajduje się w katalogu określonym w parametrze konfiguracji serwera AlertScriptsPath. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Script”
gsm_modem	string	Nazwa urządzenia szeregowego modemu GSM. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SMS”

Property	Type	Description
passwd	string	Hasło uwierzytelniające. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_authentication` jest ustawione na „Normal password”
provider	integer	Dostawca poczty e-mail. Możliwe wartości: 0 - *(domyślnie)* Generic SMTP; 1 - Gmail; 2 - Gmail relay; 3 - Office365; 4 - Office365 relay.
smtp_email	string	Adres e-mail, z którego będą wysyłane powiadomienia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Email”
smtp_helo	string	SMTP HELO. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Email”
smtp_server	string	Serwer SMTP. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Email”
smtp_port	integer	Port serwera SMTP, z którym ma zostać nawiązane połączenie. Domyślnie: 25. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Email”
smtp_security	integer	Poziom zabezpieczeń połączenia SMTP, który ma być używany. Możliwe wartości: 0 - *(domyślnie)* Brak; 1 - STARTTLS; 2 - SSL/TLS. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Email”
smtp_verify_host	integer	Weryfikacja hosta SSL dla SMTP. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_security` jest ustawione na „STARTTLS” lub „SSL/TLS”
smtp_verify_peer	integer	Weryfikacja peera SSL dla SMTP. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_security` jest ustawione na „STARTTLS” lub „SSL/TLS”

Property	Type	Description
smtp_authentication	integer	Metoda uwierzytelniania SMTP, która ma być używana. Możliwe wartości: 0 - *(domyślnie)* Brak; 1 - Normal password; 2 - OAuth token. Uwierzytelnianie OAuth nie jest dozwolone dla dostawcy poczty e-mail *Office365 relay*. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Email”
redirection_url	string	Adres URL frontendu Zabbix, na który nastąpi przekierowanie zwrotne po autoryzacji OAuth. Domyślnie: Wartość właściwości ustawień API `url` z częścią `zabbix.php?action=oauth.authorize` Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
client_id	string	Identyfikator klienta zarejestrowany na serwerze autoryzacji OAuth. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
client_secret	string	Sekret klienta zarejestrowany na serwerze autoryzacji OAuth. Dostępny tylko dla użytkownika typu Super Admin. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
authorization_url	string	Adres URL OAuth wraz z parametrami, służący do uzyskania tokenów dostępu i odświeżania. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
token_url	string	Adres URL OAuth służący do wymiany tokenu autoryzacyjnego na tokeny dostępu i odświeżania. Ten adres URL jest również używany przez serwer do odświeżania nieważnego tokenu dostępu. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
tokens_status	integer	Maska bitowa stanu tokenów. Możliwe wartości: 0 - (domyślnie) Oba tokeny zawierają nieprawidłową wartość 1 - Token dostępu zawiera prawidłową wartość 2 - Token odświeżania zawiera prawidłową wartość 3 - Oba tokeny zawierają prawidłową wartość. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
access_token	string	Wartość tokenu dostępu OAuth. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”

Property	Type	Description
access_token_updated	timestamp	Znacznik czasu ostatniej modyfikacji `access_token` wykonanej przez serwer podczas odświeżania przy użyciu `refresh_token` lub przez API przy zmianach tokenu. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
access_expires_in	integer	Czas w sekundach, po którym `access_token` stanie się nieaktualny i będzie wymagał wysłania żądania do `refresh_url`. Jest ustawiany przez serwer Zabbix podczas odświeżania `access_token` lub przez API przy zmianach tokenu. Znacznik czasu jest obliczany przez dodanie wartości `access_token_updated`. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
refresh_token	string	Wartość tokenu odświeżania OAuth. Zachowanie właściwości: - *wymagane*, jeśli `smtp_authentication` jest ustawione na „OAuth token”
status	integer	Czy typ nośnika jest włączony. Możliwe wartości: 0 - *(domyślnie)* Włączony; 1 - Wyłączony.
username	string	Nazwa użytkownika. Zachowanie właściwości: - *obsługiwane*, jeśli `smtp_authentication` jest ustawione na „Normal password”
maxsessions	integer	Maksymalna liczba alertów, które mogą być przetwarzane równolegle. Możliwe wartości, jeśli `type` jest ustawione na „SMS”: 1. Możliwe wartości, jeśli `type` jest ustawione na „Email”, „Script” lub „Webhook”: 0-100. Domyślnie: 1.
maxattempts	integer	Maksymalna liczba prób wysłania alertu. Możliwe wartości: 1-100. Domyślnie: 3.
attempt_interval	string	Interwał między kolejnymi próbami ponowienia. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Możliwe wartości: 0-1h. Domyślnie: 10s.
message_format	integer	Format wiadomości. Możliwe wartości: 0 - Zwykły tekst; 1 - *(domyślnie)* HTML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Email”

Property	Type	Description
script	text	Treść skryptu webhook (JavaScript). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Webhook”
timeout	string	Limit czasu skryptu webhook. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Możliwe wartości: 1-60s. Domyślnie: 30s. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Webhook”
process_tags	integer	Przetwarzaj wartości właściwości JSON w odpowiedzi skryptu webhook jako tagi. Tagi te są dodawane do wszystkich istniejących tagów problemu. Możliwe wartości: 0 - *(domyślnie)* Ignoruj odpowiedź skryptu webhook; 1 - Przetwarzaj odpowiedź skryptu webhook jako tagi. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Webhook”
show_event_menu	integer	Uwzględnij wpis w menu zdarzenia, który prowadzi do niestandardowego adresu URL. Dodaje również właściwość `urls` do danych wyjściowych problem.get i event.get. Możliwe wartości: 0 - *(domyślnie)* Nie uwzględniaj wpisu menu zdarzenia ani właściwości `urls`; 1 - Uwzględnij wpis menu zdarzenia i właściwość `urls`. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Webhook”
event_menu_url	string	Adres URL używany we wpisie menu zdarzenia oraz we właściwości `urls` zwracanej przez problem.get i event.get. Zachowanie właściwości: - *obsługiwane*, jeśli `show_event_menu` jest ustawione na „Include event menu entry and `urls` property”
event_menu_name	string	Nazwa używana dla wpisu menu zdarzenia oraz we właściwości `urls` zwracanej przez problem.get i event.get. Zachowanie właściwości: - *obsługiwane*, jeśli `show_event_menu` jest ustawione na „Include event menu entry and `urls` property”
parameters	array	Parametry webhooka lub skryptu. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Webhook” lub „Script”
description	text	Opis typu nośnika.

Parametry webhooka

Parametry webhooka mają następujące właściwości.

Właściwość	Type	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru, obsługuje makra. Obsługiwane makra są opisane na stronie Supported macros.

Parametry skryptu

Parametry skryptu mają następujące właściwości.

Właściwość	Type	Opis
sortorder	integer	Kolejność, w jakiej wartości parametrów będą przekazywane do skryptu jako argumenty wiersza poleceń, zaczynając od 0 jako pierwszego argumentu. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru, obsługuje makra. Obsługiwane makra są opisane na stronie Supported macros.

Szablon wiadomości

Obiekt szablonu wiadomości definiuje szablon, który będzie używany jako domyślna wiadomość dla operacji akcji wysyłających powiadomienie. Ma on następujące właściwości.

Właściwość	Type	Opis
eventsource	integer	Źródło zdarzenia. Możliwe wartości: 0 - Wyzwalacze; 1 - Wykrywanie; 2 - Autorejestracja; 3 - Wewnętrzne; 4 - Usługi. Property behavior: - *required*
recovery	integer	Tryb operacji. Możliwe wartości: 0 - Operacje; 1 - Operacje odzyskiwania; 2 - Operacje aktualizacji. Property behavior: - *required*
subject	string	Temat wiadomości.
message	string	Treść wiadomości.

mediatype.create

Opis

`object mediatype.create(object/array mediaTypes)`

Ta metoda umożliwia tworzenie nowych typów mediów.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Typy mediów do utworzenia.

Oprócz standardowych właściwości typu mediów metoda akceptuje następujące parametry.

Parametr	Type	Opis
message_templates	array	Szablony wiadomości, które mają zostać utworzone dla typu mediów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych typów mediów we właściwości `mediatypeids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych typów mediów.

Przykłady

Tworzenie typu nośnika e-mail

Utwórz nowy typ nośnika e-mail z niestandardowym portem SMTP i szablonami wiadomości.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
        "type": "0",
        "name": "Email",
        "smtp_server": "mail.example.com",
        "smtp_helo": "example.com",
        "smtp_email": "zabbix@example.com",
        "smtp_port": "587",
        "message_format": "1",
        "message_templates": [
            {
                "eventsource": "0",
                "recovery": "0",
                "subject": "Problem: {EVENT.NAME}",
                "message": "Problem \"{EVENT.NAME}\" na hoście \"{HOST.NAME}\" rozpoczął się o {EVENT.TIME
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Rozwiązano w {EVENT.DURATION}: {EVENT.NAME}",
                "message": "Problem \"{EVENT.NAME}\" na hoście \"{HOST.NAME}\" został rozwiązany o {EVENT.
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Zaktualizowany problem w {EVENT.AGE}: {EVENT.NAME}",
                "message": "{USER.FULLNAME} wykonał(a) akcję {EVENT.UPDATE.ACTION} dla problemu \"{EVENT.N
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "mediatypeids": [
            "7"
        ]
    },
    "id": 1
}
```

Tworzenie typu nośnika skryptu

Utwórz nowy typ nośnika skryptu z niestandardową wartością liczby prób oraz interwału między nimi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
        "type": "1",
        "name": "Push notifications",
        "exec_path": "push-notification.sh",
        "maxattempts": "5",
        "attempt_interval": "11s",
        "parameters": [
            {
                "sortorder": "0",
                "value": "{ALERT.SENDTO}"
            },
            {
                "sortorder": "1",
                "value": "{ALERT.SUBJECT}"
            },
            {
                "sortorder": "2",
                "value": "{ALERT.MESSAGE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "8"
        ]
    },
    "id": 1
}
```

Tworzenie typu nośnika webhook

Utwórz nowy typ nośnika webhook.

Żądanie:

```
{
    "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>"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "9"
        ]
    },
    "id": 1
}
```

Źródło

CMediaType::create() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.delete

Opis

`object mediatype.delete(array mediaTypeIds)`

Ta metoda pozwala na usuwanie typów mediów.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID typów mediów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych typów mediów we właściwości `mediatypeids`.

Przykłady

Usuwanie wielu typów mediów

Usuń dwa typy mediów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "mediatype.delete",
"params": [
```

```
"3",
"5"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"mediatypeids": [
"3",
"5"
]
},
"id": 1
}
```

Źródło

CMediaType::delete() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.get

Opis

`integer/array mediatype.get(object parameters)`

Ta metoda umożliwia pobieranie typów mediów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Note:
Podczas żądania informacji o typach mediów powiązanych z użytkownikiem, użytkownicy typu *Admin* i *User* mogą pobierać dane tylko o własnym użytkowniku. Przykład znajduje się w sekcji Pobieranie typów mediów jako *Admin*.

Parametr	Typ	Opis
mediatypeids	ID/array	Zwracaj tylko typy mediów o podanych ID.
mediaids	ID/array	Zwracaj tylko typy mediów używane przez podane media.
userids	ID/array	Zwracaj tylko typy mediów używane przez podanych użytkowników.
selectActions	query	Zwracaj właściwość `actions` z akcjami używającymi typu mediów.
selectMessageTemplates	query	Zwracaj właściwość `message_templates` z tablicą komunikatów typu mediów. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*
selectUsers	query	Zwracaj właściwość `users` z użytkownikami używającymi typu mediów. Ograniczenia zależne od typu użytkownika opisano w `user.get`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `mediatypeid`.

Parametr	Typ	Opis
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Możliwe właściwości obiektu typu mediów dla użytkowników typu *Admin* i *User*: `mediatypeid`, `name`, `type`, `status`, `maxattempts`.
output	query	Właściwości obiektu typu mediów, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: `mediatypeid`, `name`, `type`, `status`, `maxattempts`, `description`. Przykład znajduje się w sekcji Pobieranie typów mediów jako *Admin*. Domyślnie: `extend`.
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Możliwe właściwości obiektu typu mediów dla użytkowników typu *Admin* i *User*: `name`, `description`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie typów mediów

Pobiera wszystkie skonfigurowane typy mediów. Poniższy przykład zwraca dwa typy mediów:

- typ mediów e-mail;
- typ mediów SMS.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "selectMessageTemplates": "extend"
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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",
            "maxsessions": "1",
            "maxattempts": "3",
            "attempt_interval": "10s",
            "message_format": "0",
            "script": "",
            "timeout": "30s",
            "process_tags": "0",
            "show_event_menu": "1",
            "event_menu_url": "",
            "event_menu_name": "",
            "description": "",
            "provider": "0",
            "message_templates": [
                {
                    "eventsource": "0",
                    "recovery": "0",
                    "subject": "Problem: {EVENT.NAME}",
                    "message": "Problem rozpoczął się o {EVENT.TIME} dnia {EVENT.DATE}\r\nNazwa problemu:
                },
                {
                    "eventsource": "0",
                    "recovery": "1",
                    "subject": "Rozwiązano: {EVENT.NAME}",
                    "message": "Problem został rozwiązany o {EVENT.RECOVERY.TIME} dnia {EVENT.RECOVERY.DAT
                },
                {
                    "eventsource": "0",
                    "recovery": "2",
                    "subject": "Zaktualizowany problem: {EVENT.NAME}",
                    "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem o {EVENT.UPDATE.DATE} {EVENT
                },
                {
                    "eventsource": "1",
                    "recovery": "0",
                    "subject": "Wykrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}",
                    "message": "Reguła wykrywania: {DISCOVERY.RULE.NAME}\r\n\r\nAdres IP urządzenia: {DISC
                },
```

```
            {
                "eventsource": "2",
                "recovery": "0",
                "subject": "Autorejestracja: {HOST.HOST}",
                "message": "Nazwa hosta: {HOST.HOST}\r\nIP hosta: {HOST.IP}\r\nPort agenta: {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",
        "maxsessions": "1",
        "maxattempts": "3",
        "attempt_interval": "10s",
        "message_format": "1",
        "script": "",
        "timeout": "30s",
        "process_tags": "0",
        "show_event_menu": "1",
        "event_menu_url": "",
        "event_menu_name": "",
        "description": "",
        "provider": "0",
        "message_templates": [
            {
                "eventsource": "0",
                "recovery": "0",
                "subject": "",
                "message": "{EVENT.SEVERITY}: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "",
                "message": "ROZWIĄZANO: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT.TIME}
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "",
                "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem o {EVENT.UPDATE.DATE} {EVENT
            },
            {
                "eventsource": "1",
                "recovery": "0",
                "subject": "",
                "message": "Wykrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}"
```

```
                },
                {
                    "eventsource": "2",
                    "recovery": "0",
                    "subject": "",
                    "message": "Autorejestracja: {HOST.HOST}\r\nIP hosta: {HOST.IP}\r\nPort agenta: {HOST.
                }
            ],
            "parameters": []
        }
    ],
    "id": 1
}
```

Pobieranie typów mediów jako *Admin*

Jako użytkownik typu *Admin* pobierz wszystkie typy mediów, które są włączone, wraz z użytkownikami korzystającymi z tych typów mediów. Poniższy przykład zwraca dwa typy mediów:

- typ mediów e-mail z jednym użytkownikiem (tylko własny użytkownik użytkownika typu *Admin*);
- typ mediów SMS bez użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "filter": {
            "status": 0
        },
        "selectUsers": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mediatypeid": "1",
            "type": "0",
            "name": "Email",
            "status": "0",
            "description": "",
            "maxattempts": "3",
            "users": [
                {
                    "userid": "3",
                    "username": "database-admin",
                    "name": "John",
                    "surname": "Doe",
                    "url": "",
                    "autologin": "0",
                    "autologout": "0",
                    "lang": "default",
                    "refresh": "30s",
                    "theme": "default",
                    "attempt_failed": "0",
                    "attempt_ip": "",
                    "attempt_clock": "0",
                    "rows_per_page": "50",
                    "timezone": "default",
```

```
                    "roleid": "2",
                    "provisioned": "0"
                }
            ]
        },
        {
            "mediatypeid": "3",
            "type": "2",
            "name": "SMS",
            "status": "0",
            "description": "",
            "maxattempts": "3",
            "users": []
        }
    ],
    "id": 1
}
```

Pobieranie typów nośników skryptu i webhooka

Poniższy przykład zwraca trzy typy nośników:

- typ nośnika skryptu z parametrami;
- typ nośnika skryptu bez parametrów;
- typ nośnika webhooka z parametrami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": ["mediatypeid", "name", "parameters"],
        "filter": {
            "type": [1, 4]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "mediatypeid": "10",
            "name": "Script with parameters",
            "parameters": [
                {
                    "sortorder": "0",
                    "value": "{ALERT.SENDTO}"
                },
                {
                    "sortorder": "1",
                    "value": "{EVENT.NAME}"
                },
                {
                    "sortorder": "2",
                    "value": "{ALERT.MESSAGE}"
                },
                {
                    "sortorder": "3",
                    "value": "Zabbix alert"
                }
            ]
```

```
        },
        {
            "mediatypeid": "13",
            "name": "Script without parameters",
            "parameters": []
        },
        {
            "mediatypeid": "11",
            "name": "Webhook",
            "parameters": [
                {
                    "name": "alert_message",
                    "value": "{ALERT.MESSAGE}"
                },
                {
                    "name": "event_update_message",
                    "value": "{EVENT.UPDATE.MESSAGE}"
                },
                {
                    "name": "host_name",
                    "value": "{HOST.NAME}"
                },
                {
                    "name": "trigger_description",
                    "value": "{TRIGGER.DESCRIPTION}"
                },
                {
                    "name": "trigger_id",
                    "value": "{TRIGGER.ID}"
                },
                {
                    "name": "alert_source",
                    "value": "Zabbix"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Użytkownik

Źródło

CMediaType::get() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.update

Opis

`object mediatype.update(object/array mediaTypes)`

Ta metoda umożliwia aktualizację istniejących typów mediów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości typu mediów do zaktualizowania.

Właściwość `mediatypeid` musi być zdefiniowana dla każdego typu mediów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości typu mediów, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
message_templates	array	Szablony wiadomości zastępujące bieżące szablony wiadomości.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych typów mediów we właściwości `mediatypeids`.

Przykłady

Włączanie typu mediów

Włącz typ mediów, czyli ustaw jego status na ”0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.update",
    "params": {
        "mediatypeid": "6",
        "status": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CMediaType::update() w *ui/include/classes/api/services/CMediaType.php*.

Ustawienia

Ta klasa została zaprojektowana do pracy z typowymi ustawieniami administracyjnymi.

Odwołania do obiektu:

- Settings

Dostępne metody:

- settings.get - pobieranie ustawień
- settings.update - aktualizacja ustawień

Obiekt ustawień

Następujące obiekty są bezpośrednio powiązane z API `settings`.

Ustawienia

Obiekt settings ma następujące właściwości.

Property	Type	Description
default_lang	string	Domyślny język systemu. Domyślnie: `en_US`.
default_timezone	string	Domyślna strefa czasowa systemu. Domyślnie: `system` - domyślna strefa systemu. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP.
default_theme	string	Domyślny motyw. Możliwe wartości: `blue-theme` - *(domyślny)* Niebieski; `dark-theme` - Ciemny; `hc-light` - Jasny o wysokim kontraście; `hc-dark` - Ciemny o wysokim kontraście.
search_limit	integer	Limit wyników wyszukiwania i filtrowania. Domyślnie: 1000.
max_overview_table_size	integer	Maksymalna liczba kolumn i wierszy w widżetach pulpitu Data overview i Trigger overview. Domyślnie: 50.
max_in_table	integer	Maksymalna liczba elementów wyświetlanych w komórce tabeli. Domyślnie: 50.
server_check_interval	integer	Wyświetl ostrzeżenie, jeśli serwer Zabbix jest niedostępny. Możliwe wartości: 0 - Nie wyświetlaj ostrzeżenia; 10 - *(domyślnie)* Wyświetl ostrzeżenie.
work_period	string	Czas pracy. Domyślnie: 1-5,09:00-18:00.
show_technical_errors	integer	Wyświetlaj błędy techniczne (PHP/SQL) użytkownikom innym niż Super admin oraz użytkownikom, którzy nie należą do grup użytkowników z włączonym trybem debugowania. Możliwe wartości: 0 - *(domyślnie)* Nie wyświetlaj błędów technicznych; 1 - Wyświetlaj błędy techniczne.
history_period	string	Maksymalny okres wyświetlania danych historycznych w widżetach Latest data, Web i Data overview. Akceptuje sekundy oraz jednostki czasu z sufiksem. Domyślnie: 24h.
period_default	string	Domyślny okres filtra czasu. Akceptuje sekundy oraz jednostki czasu z sufiksem, z obsługą miesięcy i lat (30s, 1m, 2h, 1d, 1M, 1y). Domyślnie: 1h.
max_period	string	Maksymalny okres dla filtra czasu. Akceptuje sekundy oraz jednostki czasu z sufiksem, z obsługą miesięcy i lat (30s, 1m, 2h, 1d, 1M, 1y). Domyślnie: 2y.
severity_color_0	string	Kolor dla poziomu ”Not classified” jako kod koloru szesnastkowego. Domyślnie: 97AAB3.
severity_color_1	string	Kolor dla poziomu ”Information” jako kod koloru szesnastkowego. Domyślnie: 7499FF.

Property	Type	Description
severity_color_2	string	Kolor dla poziomu ”Warning” jako kod koloru szesnastkowego. Domyślnie: FFC859.
severity_color_3	string	Kolor dla poziomu ”Average” jako kod koloru szesnastkowego. Domyślnie: FFA059.
severity_color_4	string	Kolor dla poziomu ”High” jako kod koloru szesnastkowego. Domyślnie: E97659.
severity_color_5	string	Kolor dla poziomu ”Disaster” jako kod koloru szesnastkowego. Domyślnie: E45959.
severity_name_0	string	Nazwa dla poziomu ”Not classified”. Domyślnie: Not classified.
severity_name_1	string	Nazwa dla poziomu ”Information”. Domyślnie: Information.
severity_name_2	string	Nazwa dla poziomu ”Warning”. Domyślnie: Warning.
severity_name_3	string	Nazwa dla poziomu ”Average”. Domyślnie: Average.
severity_name_4	string	Nazwa dla poziomu ”High”. Domyślnie: High.
severity_name_5	string	Nazwa dla poziomu ”Disaster”. Domyślnie: Disaster.
custom_color	integer	Używaj niestandardowych kolorów statusu zdarzeń. Możliwe wartości: 0 - *(domyślnie)* Nie używaj niestandardowych kolorów statusu zdarzeń; 1 - Używaj niestandardowych kolorów statusu zdarzeń.
ok_period	string	Wyświetlaj okres dla wyzwalaczy OK. Akceptuje sekundy oraz jednostki czasu z sufiksem. Domyślnie: 5m.
blink_period	string	Okres migania dla wyzwalaczy po zmianie statusu. Akceptuje sekundy oraz jednostki czasu z sufiksem. Domyślnie: 2m.
problem_unack_color	string	Kolor dla niepotwierdzonych zdarzeń PROBLEM jako kod koloru szesnastkowego. Domyślnie: CC0000.
problem_ack_color	string	Kolor dla potwierdzonych zdarzeń PROBLEM jako kod koloru szesnastkowego. Domyślnie: CC0000.
ok_unack_color	string	Kolor dla niepotwierdzonych zdarzeń RESOLVED jako kod koloru szesnastkowego. Domyślnie: 009900.
ok_ack_color	string	Kolor dla potwierdzonych zdarzeń RESOLVED jako kod koloru szesnastkowego. Domyślnie: 009900.

Property	Type	Description
problem_unack_style	integer	Miganie dla niepotwierdzonych zdarzeń PROBLEM. Możliwe wartości: 0 - Nie wyświetlaj migania; 1 - *(domyślnie)* Wyświetlaj miganie.
problem_ack_style	integer	Miganie dla potwierdzonych zdarzeń PROBLEM. Możliwe wartości: 0 - Nie wyświetlaj migania; 1 - *(domyślnie)* Wyświetlaj miganie.
ok_unack_style	integer	Miganie dla niepotwierdzonych zdarzeń RESOLVED. Możliwe wartości: 0 - Nie wyświetlaj migania; 1 - *(domyślnie)* Wyświetlaj miganie.
ok_ack_style	integer	Miganie dla potwierdzonych zdarzeń RESOLVED. Możliwe wartości: 0 - Nie wyświetlaj migania; 1 - *(domyślnie)* Wyświetlaj miganie.
url	string	URL frontendu.
discovery_groupid	ID	ID grupy hostów, do której będą automatycznie przypisywane wykryte hosty.
default_inventory_mode	integer	Domyślny tryb inwentarza hosta. Możliwe wartości: -1 - *(domyślnie)* Wyłączony; 0 - Ręczny; 1 - Automatyczny.
alert_usrgrpid	ID	ID grupy użytkowników, do której zostanie wysłana wiadomość alarmowa o niedostępności bazy danych. Jeśli ustawiono wartość ”0”, wiadomość alarmowa nie zostanie wysłana.
snmptrap_logging	integer	Rejestruj niepasujące pułapki SNMP. Możliwe wartości: 0 - Nie rejestruj niepasujących pułapek SNMP; 1 - *(domyślnie)* Rejestruj niepasujące pułapki SNMP.
login_attempts	integer	Liczba nieudanych prób logowania, po której formularz logowania zostanie zablokowany. Domyślnie: 5.
login_block	string	Przedział czasu, w którym formularz logowania będzie zablokowany, jeśli liczba nieudanych prób logowania przekroczy wartość zdefiniowaną w polu login_attempts. Akceptuje sekundy oraz jednostki czasu z sufiksem. Domyślnie: 30s.
validate_uri_schemes	integer	Sprawdzaj schematy URI. Możliwe wartości: 0 - Nie sprawdzaj; 1 - *(domyślnie)* Sprawdzaj.
uri_valid_schemes	string	Prawidłowe schematy URI. Domyślnie: http,https,ftp,file,mailto,tel,ssh.
x_frame_options	string	Nagłówek HTTP X-Frame-Options. Domyślnie: SAMEORIGIN.

Property	Type	Description
iframe_sandboxing_enabled	integer	Używaj piaskownicy iframe. Możliwe wartości: 0 - Nie używaj; 1 - *(domyślnie)* Używaj.
iframe_sandboxing_exceptions	string	Wyjątki dla piaskownicy iframe.
connect_timeout	string	Limit czasu połączenia z serwerem Zabbix. Zakres możliwych wartości: 1-30s. Domyślnie: 3s. Zachowanie właściwości: - *wymagane*
socket_timeout	string	Domyślny limit czasu sieciowego. Zakres możliwych wartości: 1-300s. Domyślnie: 3s. Zachowanie właściwości: - *wymagane*
media_type_test_timeout	string	Limit czasu sieciowego dla testu typu mediów. Zakres możliwych wartości: 1-300s. Domyślnie: 65s. Zachowanie właściwości: - *wymagane*
item_test_timeout	string	Limit czasu sieciowego dla testów pozycji. Zakres możliwych wartości: 1-600s. Domyślnie: 60s. Zachowanie właściwości: - *wymagane*
script_timeout	string	Limit czasu sieciowego dla wykonywania skryptu. Zakres możliwych wartości: 1-300s. Domyślnie: 60s. Zachowanie właściwości: - *wymagane*
report_test_timeout	string	Limit czasu sieciowego dla testu zaplanowanego raportu. Zakres możliwych wartości: 1-300s. Domyślnie: 60s. Zachowanie właściwości: - *wymagane*
auditlog_enabled	integer	Czy włączyć rejestrowanie audytu. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.

Property	Type	Description
auditlog_mode	integer	Czy włączyć rejestrowanie audytu działań low-level discovery, network discovery i autorejestracji wykonywanych przez serwer (użytkownik System). Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
ha_failover_delay	string	Opóźnienie przełączenia awaryjnego w sekundach. Domyślnie: 1m. Zachowanie właściwości: - *tylko do odczytu*
geomaps_tile_provider	string	Dostawca kafelków geomapy. Możliwe wartości: `OpenStreetMap.Mapnik` - *(domyślnie)* OpenStreetMap Mapnik; `OpenTopoMap` - OpenTopoMap; `Stamen.TonerLite` - Stamen Toner Lite; `Stamen.Terrain` - Stamen Terrain; `USGS.USTopo` - USGS US Topo; `USGS.USImagery` - USGS US Imagery. Obsługuje pusty ciąg znaków, aby określić niestandardowe wartości `geomaps_tile_url`, `geomaps_max_zoom` i `geomaps_attribution`.
geomaps_tile_url	string	URL kafelka geomapy. Zachowanie właściwości: - *obsługiwane* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg znaków
geomaps_max_zoom	integer	Maksymalny poziom powiększenia geomapy. Możliwe wartości: 0-30. Zachowanie właściwości: - *obsługiwane* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg znaków
geomaps_attribution	string	Tekst atrybucji geomapy. Zachowanie właściwości: - *obsługiwane* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg znaków
vault_provider	integer	Dostawca vault. Możliwe wartości: 0 - *(domyślnie)* HashiCorp Vault; 1 - CyberArk Vault.
proxy_secrets_provider	integer	Rozwiązuj wartości makr sekretów przez: Możliwe wartości: 0 - *(domyślnie)* tylko serwer; 1 - serwer i proxy niezależnie.

Property	Type	Description
timeout_zabbix_agent	string	Nie poświęcaj więcej niż `timeout_*` sekund na przetwarzanie. Akceptuje sekundy lub jednostki czasu z sufiksem (np. 30s, 1m). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-600s. Domyślnie: 3s. Domyślnie dla `timeout_browser`: 60s. Zachowanie właściwości: - *wymagane*
timeout_simple_check		
timeout_snmp_agent		
timeout_external_check		
timeout_db_monitor		
timeout_http_agent		
timeout_ssh_agent		
timeout_telnet_agent		
timeout_script		
timeout_browser		

settings.get

Opis

`object settings.get(object parameters)`

Ta metoda umożliwia pobranie obiektu ustawień zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

`(object)` Zwraca obiekt ustawień.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "settings.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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_US",
        "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",
        "auditlog_mode": "1",
        "ha_failover_delay": "1m",
        "geomaps_tile_provider": "OpenStreetMap.Mapnik",
        "geomaps_tile_url": "",
        "geomaps_max_zoom": "0",
```

```
        "geomaps_attribution": "",
        "vault_provider": "0",
        "timeout_zabbix_agent": "3s",
        "timeout_simple_check": "3s",
        "timeout_snmp_agent": "3s",
        "timeout_external_check": "3s",
        "timeout_db_monitor": "3s",
        "timeout_http_agent": "3s",
        "timeout_ssh_agent": "3s",
        "timeout_telnet_agent": "3s",
        "timeout_script": "3s"
    },
    "id": 1
}
```

Źródło

CSettings::get() w *ui/include/classes/api/services/CSettings.php*.

settings.update

Opis

`object settings.update(object settings)`

Ta metoda umożliwia aktualizację istniejących wspólnych ustawień.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Właściwości ustawień do zaktualizowania.

Zwracane wartości

`(array)` Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "settings.update",
    "params": {
        "login_attempts": "1",
        "login_block": "1m"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "login_attempts",
        "login_block"
    ],
    "id": 1
}
```

Źródło

CSettings::update() w *ui/include/classes/api/services/CSettings.php*.

Usługa

Ta klasa jest przeznaczona do pracy z usługami biznesowymi/infrastruktury IT.

Odwołania do obiektów:

- Usługa
- Reguła statusu
- Tag usługi
- Alarm usługi
- Tag problemu

Dostępne metody:

- service.create - utwórz nowe usługi
- service.delete - usuń usługi
- service.get - pobierz usługi
- service.update - zaktualizuj usługi

Obiekt usługi

Następujące obiekty są bezpośrednio powiązane z API `usługi`.

Usługa

Obiekt usługi ma następujące właściwości.

Właściwość	Typ	Opis
serviceid	ID	ID usługi. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
algorithm	integer	Reguła obliczania statusu. Ma zastosowanie tylko wtedy, gdy istnieją usługi podrzędne. Możliwe wartości: 0 - ustaw status na OK; 1 - najbardziej krytyczny, jeśli wszystkie elementy podrzędne mają problemy; 2 - najbardziej krytyczny spośród usług podrzędnych. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa usługi. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
sortorder	integer	Pozycja usługi używana do sortowania. Możliwe wartości: 0-999. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
weight	integer	Waga usługi. Możliwe wartości: 0-1000000. Domyślnie: 0.

Właściwość	Typ	Opis
propagation_rule	integer	Reguła propagacji statusu. Możliwe wartości: 0 - *(domyślnie)* propaguj status usługi bez zmian; 1 - zwiększ propagowany status o podaną wartość `propagation_value` (o 1 do 5 poziomów ważności); 2 - zmniejsz propagowany status o podaną wartość `propagation_value` (o 1 do 5 poziomów ważności); 3 - ignoruj tę usługę - status w ogóle nie jest propagowany do usługi nadrzędnej; 4 - ustaw stały status usługi przy użyciu podanej wartości `propagation_value`. Zachowanie właściwości: - *wymagane*, jeśli ustawiono `propagation_value`
propagation_value	integer	Wartość propagacji statusu. Możliwe wartości, jeśli `propagation_rule` jest ustawione na „0” lub „3”: 0 - Niesklasyfikowane. Możliwe wartości, jeśli `propagation_rule` jest ustawione na „1” lub „2”: 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Możliwe wartości, jeśli `propagation_rule` jest ustawione na „4”: -1 - OK; 0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane*, jeśli ustawiono `propagation_rule`
status	integer	Określa, czy usługa jest w stanie OK, czy w stanie problemu. Jeśli usługa jest w stanie problemu, `status` jest równy jednej z następujących wartości: - ważności najbardziej krytycznego problemu; - najwyższemu statusowi usługi podrzędnej będącej w stanie problemu. Jeśli usługa jest w stanie OK, `status` jest równy: -1. Zachowanie właściwości: - *tylko do odczytu*
description	string	Opis usługi.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych usług z już istniejącymi. Generowany automatycznie, jeśli nie został podany.
created_at	integer	Znacznik czasu Unix określający moment utworzenia usługi.

Właściwość	Typ	Opis
readonly	integer	Dostęp do usługi. Możliwe wartości: 0 - Odczyt i zapis; 1 - Tylko do odczytu. Zachowanie właściwości: - *tylko do odczytu*

Reguła statusu

Obiekt reguły statusu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Warunek ustawienia statusu (Nowy status). Możliwe wartości: 0 - jeśli co najmniej (N) usług podrzędnych ma status (Status) lub wyższy; 1 - jeśli co najmniej (N%) usług podrzędnych ma status (Status) lub wyższy; 2 - jeśli mniej niż (N) usług podrzędnych ma status (Status) lub niższy; 3 - jeśli mniej niż (N%) usług podrzędnych ma status (Status) lub niższy; 4 - jeśli waga usług podrzędnych o statusie (Status) lub wyższym wynosi co najmniej (W); 5 - jeśli waga usług podrzędnych o statusie (Status) lub wyższym wynosi co najmniej (N%); 6 - jeśli waga usług podrzędnych o statusie (Status) lub niższym jest mniejsza niż (W); 7 - jeśli waga usług podrzędnych o statusie (Status) lub niższym jest mniejsza niż (N%). Gdzie: - N (W) to `limit_value`; - (Status) to `limit_status`; - (Nowy status) to `new_status`. Zachowanie właściwości: - *wymagane*
limit_value	integer	Wartość graniczna. Możliwe wartości: - dla N i W: 1-100000; - dla N%: 1-100. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
limit_status	integer	Status graniczny. Możliwe wartości: -1 - OK; 0 - Niesklasyfikowany; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane*
new_status	integer	Wartość nowego statusu. Możliwe wartości: 0 - Niesklasyfikowany; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane*

Znacznik usługi

Obiekt znacznika usługi ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa znacznika usługi. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika usługi.

Alarm usługi

Note:
Alarmów usługi nie można bezpośrednio tworzyć, aktualizować ani usuwać za pomocą Zabbix API.

Obiekty alarmu usługi reprezentują zmianę stanu usługi. Mają następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Czas wystąpienia zmiany stanu usługi.
value	integer	Status usługi. Listę możliwych wartości można znaleźć we właściwości `status` usługi.

Znacznik problemu

Znaczniki problemów umożliwiają powiązanie usług ze zdarzeniami problemów. Obiekt znacznika problemu ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa znacznika problemu. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek mapowania operator. Możliwe wartości: 0 - *(domyślnie)* Równa się; 2 - Zawiera.
value	string	Wartość znacznika problemu.

service.create

Opis

`object service.create(object/array services)`

Ta metoda umożliwia tworzenie nowych usług.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` usługi do utworzenia.

Oprócz standardowych właściwości usługi, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
children	array	Podrzędne usługi, które mają zostać powiązane z usługą. Podrzędne usługi muszą mieć zdefiniowaną tylko właściwość `serviceid`.
parents	array	Nadrzędne usługi, które mają zostać powiązane z usługą. Nadrzędne usługi muszą mieć zdefiniowaną tylko właściwość `serviceid`.
tags	array	Tagi usługi, które mają zostać utworzone dla usługi.
problem_tags	array	Tagi problemów, które mają zostać utworzone dla usługi.
status_rules	array	Reguły statusu, które mają zostać utworzone dla usługi.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych usług we właściwości `serviceids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych usług.

Przykłady

Tworzenie usługi

Utwórz usługę, która zostanie przełączona w stan problemu, jeśli co najmniej jedno dziecko ma problem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.create",
    "params": {
        "name": "Server 1",
        "algorithm": 1,
        "sortorder": 1
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CService::create() w *ui/include/classes/api/services/CService.php*.

service.delete

Opis

`object service.delete(array serviceIds)`

Ta metoda umożliwia usuwanie usług.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID usług do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych usług we właściwości `serviceids`.

Przykłady

Usuwanie wielu usług

Usuń dwie usługi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CService::delete() w *ui/include/classes/api/services/CService.php*.

service.get

Opis

`integer/array service.get(object parameters)`

Ta metoda umożliwia pobieranie usług zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
serviceids	ID/array	Zwracaj tylko usługi o podanych identyfikatorach.
parentids	ID/array	Zwracaj tylko usługi, które są powiązane z podanymi usługami nadrzędnymi.
deep_parentids	flag	Zwracaj wszystkie bezpośrednie i pośrednie usługi podrzędne. Używane razem z `parentids`.
childids	ID/array	Zwracaj tylko usługi, które są powiązane z podanymi usługami podrzędnymi.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	object/array	Zwracaj tylko usługi z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie usługi. Możliwe wartości operator: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
problem_tags	object/array	Zwracaj tylko usługi z podanymi tagami problemów. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie usługi. Możliwe wartości operator: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
without_problem_tags	flag	Zwracaj tylko usługi bez tagów problemów.
slaids	ID/array	Zwracaj tylko usługi, które są powiązane z określonymi SLA.
selectChildren	query	Zwracaj właściwość `children` z usługami podrzędnymi. Obsługuje `count`.

Parametr	Typ	Opis
selectParents	query	Zwracaj właściwość `parents` z usługami nadrzędnymi. Obsługuje `count`.
selectTags	query	Zwracaj właściwość `tags` z tagami usług. Obsługuje `count`.
selectProblemEvents	query	Zwracaj właściwość `problem_events` z tablicą obiektów zdarzeń problemów. Obiekt zdarzenia problemu ma następujące właściwości: `eventid` - `(ID)` ID zdarzenia; `severity` - `(string)` Bieżący poziom ważności zdarzenia; `name` - `(string)` Nazwa rozwiązanego zdarzenia. Obsługuje `count`.
selectProblemTags	query	Zwracaj właściwość `problem_tags` z tagami problemów. Obsługuje `count`.
selectStatusRules	query	Zwracaj właściwość `status_rules` z regułami statusu. Obsługuje `count`.
selectStatusTimeline	object/array	Zwracaj właściwość `status_timeline` zawierającą zmiany stanu usługi dla podanych okresów. Format `[{"period_from": "<period_from>", "period_to": "<period_to>"}, ...]` - gdzie `period_from` jest datą początkową (włącznie; znacznik czasu integer), a `period_to` jest datą końcową (wyłącznie; znacznik czasu integer) dla interesującego Cię okresu. Zwraca tablicę wpisów zawierających właściwość `start_value` oraz tablicę `alarms` dla zmian stanu w określonych okresach.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `serviceid`, `name`, `status`, `sortorder`, `created_at`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich usług

Pobierz wszystkie dane o wszystkich usługach i ich zależnościach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.get",
    "params": {
        "output": "extend",
        "selectChildren": "extend",
        "selectParents": ["serviceid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "serviceid": "1",
            "name": "Zabbix cluster",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "8d77bd91b62347e4b79382912eb5df95",
            "created_at": "1761742392",
            "readonly": false,
            "parents": [],
            "children": [
                {
                    "serviceid": "2",
                    "name": "Zabbix server node 1",
                    "status": "-1",
                    "algorithm": "2",
                    "sortorder": "0",
                    "weight": "0",
                    "propagation_rule": "0",
                    "propagation_value": "0",
                    "description": "",
                    "uuid": "195911d26d7f4e218d6217079bcd5929",
                    "created_at": "1761742623",
                    "readonly": false
                },
                {
                    "serviceid": "3",
                    "name": "Zabbix server node 2",
                    "status": "-1",
                    "algorithm": "2",
                    "sortorder": "0",
                    "weight": "0",
                    "propagation_rule": "0",
                    "propagation_value": "0",
                    "description": "",
                    "uuid": "9fc659a30fe244f690dff25fc2a9db5c",
                    "created_at": "1761742654",
                    "readonly": false
                }
            ]
        },
        {
            "serviceid": "2",
```

```
            "name": "Zabbix server node 1",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "195911d26d7f4e218d6217079bcd5929",
            "created_at": "1761742623",
            "readonly": false,
            "parents": [
                {
                    "serviceid": "1",
                    "name": "Zabbix cluster"
                }
            ],
            "children": []
        },
        {
            "serviceid": "3",
            "name": "Zabbix server node 2",
            "status": "-1",
            "algorithm": "2",
            "sortorder": "0",
            "weight": "0",
            "propagation_rule": "0",
            "propagation_value": "0",
            "description": "",
            "uuid": "9fc659a30fe244f690dff25fc2a9db5c",
            "created_at": "1761742654",
            "readonly": false,
            "parents": [
                {
                    "serviceid": "1",
                    "name": "Zabbix cluster"
                }
            ],
            "children": []
        }
    ],
    "id": 1
}
```


Źródło

CService::get() w *ui/include/classes/api/services/CService.php*.

service.update

Opis

`object service.update(object/array services)`

Ta metoda umożliwia aktualizację istniejących usług.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` właściwości usługi do zaktualizowania.

Właściwość `serviceid` musi być zdefiniowana dla każdej usługi, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości usługi, metoda akceptuje następujące parametry.

Parametr	Type	Opis
children	array	Podrzędne usługi, które zastąpią bieżące usługi podrzędne. Usługi podrzędne muszą mieć zdefiniowaną wyłącznie właściwość `serviceid`.
parents	array	Nadrzędne usługi, które zastąpią bieżące usługi nadrzędne. Usługi nadrzędne muszą mieć zdefiniowaną wyłącznie właściwość `serviceid`.
tags	array	Tagi usługi, które zastąpią bieżące tagi usługi.
problem_tags	array	Tagi problemów, które zastąpią bieżące tagi problemów.
status_rules	array	Reguły statusu, które zastąpią bieżące reguły statusu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych usług we właściwości `serviceids`.

Przykłady

Ustawianie usługi nadrzędnej dla usługi

Ustaw usługę o ID „3” jako usługę nadrzędną dla usługi o ID „5”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "5",
        "parents": [
            {
                "serviceid": "3"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Dodawanie zaplanowanego przestoju

Dodaj przestój dla usługi o ID „4”, zaplanowany co tydzień od poniedziałku 22:00 do wtorku 10:00.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "4",
        "times": [
```

```
            {
                "type": "1",
                "ts_from": "165600",
                "ts_to": "201600"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4"
        ]
    },
    "id": 1
}
```

Źródło

CService::update() w *ui/include/classes/api/services/CService.php*.

Uwierzytelnianie

Ta klasa została zaprojektowana do pracy z ustawieniami uwierzytelniania.

Odwołania do obiektów:

- Authentication

Dostępne metody:

- authentication.get - pobieranie ustawień uwierzytelniania
- authentication.update - aktualizowanie ustawień uwierzytelniania

Obiekt uwierzytelniania

Następujące obiekty są bezpośrednio powiązane z API `authentication`.

Uwierzytelnianie

Obiekt uwierzytelniania ma następujące właściwości.

Właściwość	Typ	Opis
authentication_type	integer	Domyślne uwierzytelnianie. Możliwe wartości: 0 - *(domyślnie)* Wewnętrzne; 1 - LDAP.
http_auth_enabled	integer	Uwierzytelnianie HTTP. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `$ALLOW_HTTP_AUTH` jest włączone w pliku konfiguracyjnym frontend (*zabbix.conf.php*).

Właściwość	Typ	Opis
http_login_form	integer	Domyślny formularz logowania. Możliwe wartości: 0 - *(domyślnie)* Formularz logowania Zabbix; 1 - Formularz logowania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `$ALLOW_HTTP_AUTH` jest włączone w pliku konfiguracyjnym frontend (*zabbix.conf.php*).
http_strip_domains	string	Nazwa domeny do usunięcia. Zachowanie właściwości: - *obsługiwane*, jeśli `$ALLOW_HTTP_AUTH` jest włączone w pliku konfiguracyjnym frontend (*zabbix.conf.php*).
http_case_sensitive	integer	Logowanie HTTP z rozróżnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `$ALLOW_HTTP_AUTH` jest włączone w pliku konfiguracyjnym frontend (*zabbix.conf.php*).
ldap_auth_enabled	integer	Uwierzytelnianie LDAP. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone.
ldap_case_sensitive	integer	Logowanie LDAP z rozróżnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
ldap_userdirectoryid	ID	ID domyślnego katalogu użytkowników dla uwierzytelniania LDAP. Używane dla grup użytkowników z `gui_access` ustawionym na LDAP lub Domyślne systemowe. Zachowanie właściwości: - *wymagane*, jeśli `ldap_auth_enabled` jest ustawione na ”Enabled”
saml_auth_enabled	integer	Uwierzytelnianie SAML. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone.
saml_case_sensitive	integer	Logowanie SAML z rozróżnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
passwd_min_length	integer	Minimalna wymagana długość hasła. Możliwe wartości mieszczą się w zakresie od 1 do 70. Domyślnie: 8.

Właściwość	Typ	Opis
passwd_check_rules	integer	Reguły sprawdzania hasła. Możliwe wartości bitmapy: 0 - Sprawdź długość hasła; 1 - Sprawdź, czy hasło zawiera wielkie i małe litery alfabetu łacińskiego; 2 - Sprawdź, czy hasło zawiera cyfry; 4 - Sprawdź, czy hasło zawiera znaki specjalne; 8 - *(domyślnie)* Sprawdź, czy hasło nie znajduje się na liście powszechnie używanych haseł oraz nie zawiera pochodnych słowa ”Zabbix” ani imienia, nazwiska lub nazwy użytkownika. To pole jest maską bitową; akceptowana jest dowolna suma możliwych wartości bitmapy (na przykład 15, aby sprawdzać wszystkie reguły).
ldap_jit_status	integer	Status aprowizacji LDAP. Możliwe wartości: 0 - Wyłączona dla skonfigurowanych dostawców tożsamości LDAP; 1 - Włączona dla skonfigurowanych dostawców tożsamości LDAP.
saml_jit_status	integer	Status aprowizacji SAML. Możliwe wartości: 0 - Wyłączona dla skonfigurowanych dostawców tożsamości SAML; 1 - Włączona dla skonfigurowanych dostawców tożsamości SAML.
jit_provision_interval	string	Przedział czasu między żądaniami aprowizacji JIT dla zalogowanego użytkownika. Akceptuje sekundy oraz jednostkę czasu z sufiksem, z obsługą miesięcy i lat (3600s,60m,1h,1d,1M,1y). Wartość minimalna: 1h. Domyślnie: 1h.
disabled_usrgrpid	ID	Dostępne tylko dla aprowizacji LDAP. ID grupy użytkowników, do której ma zostać przypisany użytkownik po deprowizacji. Grupa użytkowników musi być wyłączona i nie może zostać włączona ani usunięta podczas konfiguracji. Zachowanie właściwości: - *wymagane*, jeśli `ldap_jit_status` jest ustawione na ”Enabled for configured LDAP IdPs” lub `saml_jit_status` jest ustawione na ”Enabled for configured SAML IdPs”
mfa_status	integer	Uwierzytelnianie wieloskładnikowe. Możliwe wartości: 0 - Wyłączone (dla wszystkich skonfigurowanych metod MFA); 1 - Włączone (dla wszystkich skonfigurowanych metod MFA).
mfaid	ID	Domyślna metoda MFA dla grup użytkowników z włączonym MFA. Zachowanie właściwości: - *wymagane*, jeśli `mfa_status` jest ustawione na ”Enabled”

authentication.get

Opis

`object authentication.get(object parameters)`

Ta metoda umożliwia pobranie obiektu uwierzytelniania zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w Rolach użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Type	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

`(object)` Zwraca obiekt uwierzytelniania.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "authentication_type": "0",
        "http_auth_enabled": "0",
        "http_login_form": "0",
        "http_strip_domains": "",
        "http_case_sensitive": "1",
        "ldap_auth_enabled": "0",
        "ldap_case_sensitive": "1",
        "ldap_userdirectoryid": "0",
        "saml_auth_enabled": "0",
        "saml_case_sensitive": "0",
        "passwd_min_length": "8",
        "passwd_check_rules": "15",
        "jit_provision_interval": "1h",
        "saml_jit_status": "0",
        "ldap_jit_status": "0",
        "disabled_usrgrpid": "9",
        "mfa_status": "0",
        "mfaid": "0"
    },
    "id": 1
}
```

Źródło

CAuthentication::get() w *ui/include/classes/api/services/CAuthentication.php*.

authentication.update

Opis

`object authentication.update(object authentication)`

Ta metoda umożliwia aktualizację istniejących ustawień uwierzytelniania.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Właściwości uwierzytelniania, które mają zostać zaktualizowane.

Zwracane wartości

`(array)` Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.update",
    "params": {
        "http_auth_enabled": 1,
        "http_case_sensitive": 0,
        "http_login_form": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "http_auth_enabled",
        "http_case_sensitive",
        "http_login_form"
    ],
    "id": 1
}
```

Źródło

CAuthentication::update() w *ui/include/classes/api/services/CAuthentication.php*.

Użytkownik

Ta klasa została zaprojektowana do pracy z użytkownikami.

Odwołania do obiektów:

- Użytkownik
- Media

Dostępne metody:

- user.checkauthentication - sprawdzanie i przedłużanie sesji użytkownika
- user.create - tworzenie nowych użytkowników
- user.delete - usuwanie użytkowników
- user.get - pobieranie użytkowników
- user.login - logowanie do API
- user.logout - wylogowanie z API
- user.provision - provisionowanie użytkowników LDAP
- user.resettotp - resetowanie sekretów TOTP użytkownika
- user.unblock - odblokowywanie użytkowników
- user.update - aktualizowanie użytkowników

Obiekt użytkownika

Następujące obiekty są bezpośrednio powiązane z API `user`.

User

Obiekt user ma następujące właściwości.

Property	Type	Description
userid	ID	ID użytkownika. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
username	string	Nazwa użytkownika. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla użytkowników aprowizowanych, jeśli użytkownik jest powiązany z user directory (`userdirectoryid` ma ustawioną prawidłową wartość inną niż ”0”), a stan aprowizacji user directory jest włączony (`provision_status` obiektu User directory object ma wartość ”1”), oraz stan uwierzytelniania dla całej aprowizacji LDAP lub SAML jest włączony (`ldap_jit_status` obiektu Authentication object ma wartość ”Enabled for configured LDAP IdPs” lub `saml_jit_status` obiektu Authentication object ma wartość ”Enabled for configured SAML IdPs”)
passwd	string	Hasło użytkownika. Wartość tego parametru może być pustym ciągiem, jeśli użytkownik jest powiązany z user directory. Zachowanie właściwości: - *tylko do zapisu*
roleid	ID	ID roli użytkownika. Należy pamiętać, że użytkownicy bez roli mogą logować się do Zabbixa tylko przy użyciu uwierzytelniania LDAP lub SAML, pod warunkiem że ich informacje LDAP/SAML odpowiadają mapowaniom grup użytkowników skonfigurowanym w Zabbixie.
attempt_clock	timestamp	Czas ostatniej nieudanej próby logowania. Zachowanie właściwości: - *tylko do odczytu*
attempt_failed	integer	Liczba ostatnich nieudanych prób logowania. Zachowanie właściwości: - *tylko do odczytu*
attempt_ip	string	Adres IP, z którego pochodziła ostatnia nieudana próba logowania. Zachowanie właściwości: - *tylko do odczytu*
autologin	integer	Czy włączyć automatyczne logowanie. Możliwe wartości: 0 - *(domyślnie)* automatyczne logowanie wyłączone; 1 - automatyczne logowanie włączone.
autologout	string	Czas życia sesji użytkownika. Akceptuje sekundy oraz jednostkę czasu z sufiksem. Jeśli ustawiono 0s, sesja nigdy nie wygaśnie. Domyślnie: 15m.

Property	Type	Description
lang	string	Kod języka użytkownika, na przykład `en_US`. Domyślnie: `default` - domyślny systemowy.
name	string	Imię użytkownika.
provisioned	integer	Czy użytkownik został zaprovisionowany. Możliwe wartości: 0 - niezaprovisionowany; 1 - zaprovisionowany. Zachowanie właściwości: - *tylko do odczytu*
refresh	string	Automatyczny okres odświeżania. Akceptuje sekundy lub jednostkę czasu z sufiksem (np. 30s, 90s, 1m, 1h). Domyślnie: 30s.
rows_per_page	integer	Liczba wierszy obiektów wyświetlanych na stronie. Domyślnie: 50.
surname	string	Nazwisko użytkownika.
theme	string	Motyw użytkownika. Możliwe wartości: `default` - *(domyślnie)* domyślny systemowy; `blue-theme` - niebieski; `dark-theme` - ciemny; `hc-light` - wysoki kontrast, jasny; `hc-dark` - wysoki kontrast, ciemny.
ts_provisioned	timestamp	Czas wykonania ostatniej operacji provisioning. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwane* dla użytkowników typu *Super admin*
url	string	Adres URL strony, na którą użytkownik ma zostać przekierowany po zalogowaniu.
userdirectoryid	ID	ID user directory, z którym użytkownik jest powiązany. Używane do aprowizacji (tworzenia lub aktualizacji), a także do logowania użytkownika powiązanego z user directory. W przypadku operacji logowania wartość tej właściwości ma pierwszeństwo przed właściwością `userdirectoryid` grup użytkowników, do których należy użytkownik. Domyślnie: 0. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwane* dla użytkowników typu *Super admin*
timezone	string	Strefa czasowa użytkownika, na przykład `Europe/London`, `UTC`. Domyślnie: `default` - domyślna systemowa. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP.

Media

Obiekt media ma następujące właściwości.

Property	Type	Description
mediaid	ID	ID media użytkownika. Zachowanie właściwości: - *tylko do odczytu*
mediatypeid	ID	ID typu mediów używanego przez media użytkownika. Zachowanie właściwości: - *wymagane*
sendto	string/array	Adres, nazwa użytkownika lub inny identyfikator odbiorcy. Jeśli `type` typu mediów jest ustawiony na „Email”, wartości są reprezentowane jako tablica. Dla innych typów mediów wartość jest reprezentowana jako ciąg znaków. Zachowanie właściwości: - *wymagane*
active	integer	Określa, czy media są włączone. Możliwe wartości: 0 - *(domyślnie)* włączone; 1 - wyłączone.
severity	integer	Poziomy ważności wyzwalaczy, dla których mają być wysyłane powiadomienia. Możliwe wartości bitmapy: 1 - Niesklasyfikowane; 2 - Informacja; 4 - Ostrzeżenie; 8 - Średni; 16 - Wysoki; 32 - Katastrofa. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 48 dla Średni, Wysoki i Katastrofa). Domyślnie: 63.
period	string	Czas, w którym mogą być wysyłane powiadomienia, określony jako okres czasu lub makra użytkownika oddzielone średnikiem. Domyślnie: 1-7,00:00-24:00.
provisioned	integer	Określa, czy użytkownik został provisioned. Możliwe wartości: 0 - nieprovisioned; 1 - provisioned.
userdirectory_mediaid	ID	ID mapowania mediów katalogu użytkowników dla provisioned media. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwane* dla użytkowników typu *Super admin*

user.checkAuthentication

Opis

`object user.checkAuthentication`

Ta metoda sprawdza i przedłuża sesję użytkownika.

Attention:
Wywołanie metody `user.checkAuthentication` z użyciem parametru `sessionid` domyślnie przedłuża sesję użytkownika.

Parametry

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
extend	boolean	Czy przedłużyć sesję użytkownika. Wartość domyślna: ”true”. Ustawienie wartości na ”false” pozwala sprawdzić sesję użytkownika bez jej przedłużania.
sessionid	string	Zachowanie parametru: - *obsługiwane* jeśli ustawiono `sessionid` Token uwierzytelniania użytkownika.
token	string	Zachowanie parametru: - *wymagane* jeśli `token` nie jest ustawiony Token API użytkownika. Zachowanie parametru: - *wymagane* jeśli `sessionid` nie jest ustawiony

Zwracane wartości

`(object)` Zwraca obiekt zawierający informacje o użytkowniku.

Oprócz standardowych właściwości użytkownika zwracane są następujące informacje.

Właściwość	Type	Opis
auth_type	integer	Domyślne uwierzytelnianie dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `authentication_type` obiektu Authentication object.
debug_mode	integer	Określa, czy tryb debugowania jest włączony czy wyłączony dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `debug_mode` obiektu User group object.
deprovisioned	boolean	Określa, czy użytkownik należy do grupy wycofanych użytkowników.
gui_access	string	Metoda uwierzytelniania użytkownika do frontend. Listę możliwych wartości można znaleźć we właściwości `gui_access` obiektu User group object.
secret	string	Losowy ciąg 32 znaków. Jest generowany podczas logowania użytkownika. Właściwość `secret` nie jest zwracana, jeśli sesja użytkownika jest sprawdzana przy użyciu tokena API.
sessionid	string	Token uwierzytelniający, który musi być używany w kolejnych żądaniach API. Właściwość `sessionid` nie jest zwracana, jeśli sesja użytkownika jest sprawdzana przy użyciu tokena API.
type	integer	Typ użytkownika. Listę możliwych wartości można znaleźć we właściwości `type` obiektu Role object.

Właściwość	Type	Opis
userip	string	Adres IP użytkownika.

Przykłady

Sprawdzenie uwierzytelnienia przy użyciu tokenu uwierzytelniającego

Sprawdź i przedłuż sesję użytkownika przy użyciu tokenu uwierzytelniającego użytkownika oraz zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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",
        "userdirectoryid": "0",
        "ts_provisioned": "0",
        "type": 3,
        "userip": "127.0.0.1",
        "debug_mode": 0,
        "gui_access": "0",
        "deprovisioned": false,
        "auth_type": 0,
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2",
        "secret": "0e329b933e46984e49a5c1051ecd0751"
    },
    "id": 1
}
```

Sprawdzenie uwierzytelnienia przy użyciu tokena API

Sprawdź sesję użytkownika przy użyciu tokena API użytkownika i zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
```

```
        "token": "00aff470e07c12d707e50d98cfe39edef9e6ec349c14728dbdfbc8ddc5ea3eae"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "1355919338",
        "rows_per_page": "50",
        "timezone": "Europe/Riga",
        "roleid": "3",
        "userdirectoryid": "0",
        "ts_provisioned": "0",
        "debug_mode": 0,
        "deprovisioned": false,
        "gui_access": "1",
        "mfaid": 0,
        "auth_type": 0,
        "type": 3,
        "userip": "127.0.0.1"
    },
    "id": 1
}
```

Źródło

CUser::checkAuthentication() w *ui/include/classes/api/services/CUser.php*.

user.create

Opis

`object user.create(object/array users)`

Ta metoda umożliwia tworzenie nowych użytkowników.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Note:
Siła hasła użytkownika jest weryfikowana zgodnie z regułami polityki haseł zdefiniowanymi przez Authentication API. Więcej informacji można znaleźć w sekcji Authentication API.

Parametry

`(object/array)` Użytkownicy do utworzenia.

Oprócz standardowych właściwości użytkownika metoda akceptuje następujące parametry.

Parametr	Type	Opis
usrgrps	array	Grupy użytkowników, do których użytkownik ma zostać dodany. Grupy użytkowników muszą mieć zdefiniowaną tylko właściwość `usrgrpid`.
medias	array	Media użytkownika, które mają zostać utworzone.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych użytkowników we właściwości `userids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych użytkowników.

Przykłady

Tworzenie użytkownika

Utwórz nowego użytkownika, dodaj go do grupy użytkowników i utwórz dla niego nowe media.

Żądanie:

```
{
    "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"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uwierzytelnianie
- Media
- Grupa użytkowników
- Rola

Źródło

CUser::create() w *ui/include/classes/api/services/CUser.php*.

user.delete

Opis

`object user.delete(array users)`

Ta metoda umożliwia usuwanie użytkowników.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych użytkowników w we właściwości `userids`.

Przykłady

Usunięcie wielu użytkowników

Usunięcie dwóch użytkowników

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.delete",
    "params": [
        "1",
        "5"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::delete() w *ui/include/classes/api/services/CUser.php*.

user.get

Opis

`integer/array user.get(object parameters)`

Ta metoda umożliwia pobieranie użytkowników zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Note:
Podczas żądania informacji o mediach użytkownika, uprawnieniach lub roli, użytkownicy typu *Admin* i *User* mogą pobierać dane tylko o swoim własnym użytkowniku. Przykład znajduje się w sekcji Pobieranie użytkowników jako *Admin*.

Parametr	Typ	Opis
mediaids	ID/array	Zwracaj tylko użytkowników, którzy używają podanych mediów.
mediatypeids	ID/array	Zwracaj tylko użytkowników, którzy używają podanych typów mediów.
userids	ID/array	Zwracaj tylko użytkowników o podanych ID.
usrgrpids	ID/array	Zwracaj tylko użytkowników należących do podanych grup użytkowników.
getAccess	flag	Dodaje dodatkowe informacje o uprawnieniach użytkownika. Dodaje następujące właściwości dla każdego użytkownika: `gui_access` - `(integer)` metoda uwierzytelniania użytkownika we frontendzie. Listę możliwych wartości można znaleźć we właściwości `gui_access` obiektu grupy użytkowników. `debug_mode` - `(integer)` wskazuje, czy debugowanie jest włączone dla użytkownika. Możliwe wartości: 0 - debugowanie wyłączone, 1 - debugowanie włączone. `users_status` - `(integer)` wskazuje, czy użytkownik jest wyłączony. Możliwe wartości: 0 - użytkownik włączony, 1 - użytkownik wyłączony.
selectMedias	query	Zwracaj media używane przez użytkownika we właściwości `medias`.
selectMediatypes	query	Zwracaj typy mediów używane przez użytkownika we właściwości `mediatypes`. Zobacz `mediatype.get`, aby poznać ograniczenia zależne od typu użytkownika.
selectUsrgrps	query	Zwracaj grupy użytkowników, do których należy użytkownik, we właściwości `usrgrps`. Zobacz `usergroup.get`, aby poznać ograniczenia zależne od typu użytkownika.
selectRole	query	Zwracaj rolę użytkownika we właściwości `role`.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Możliwe właściwości obiektu User dla użytkowników typu *Admin* i *User* podczas żądania danych o użytkownikach z ich grupy użytkowników: `userid`, `name`, `surname`, `username`.

Parametr	Typ	Opis
output	query	Właściwości obiektu User, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: - Dla własnego użytkownika: `userid`, `attempt_clock`, `attempt_failed`, `attempt_ip`, `autologin`, `autologout`, `lang`, `name`, `provisioned`, `refresh`, `roleid`, `rows_per_page`, `surname`, `theme`, `timezone`, `url`, `username`. - Dla użytkowników z ich grupy użytkowników: `userid`, `name`, `surname`, `username`. Domyślnie: `extend`.
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Możliwe właściwości obiektu User dla użytkowników typu *Admin* i *User* podczas żądania danych o użytkownikach z ich grupy użytkowników: `name`, `surname`, `username`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `userid`, `username`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie użytkowników

Pobierz wszystkich skonfigurowanych użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": [
        {
            "userid": "1",
            "username": "Admin",
            "name": "Zabbix",
            "surname": "Administrator",
            "url": "",
            "autologin": "1",
            "autologout": "0",
            "lang": "en_US",
            "refresh": "0s",
            "theme": "default",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "50",
            "timezone": "default",
            "roleid": "3",
            "userdirectoryid": "0",
            "ts_provisioned": "0"
        },
        {
            "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",
            "userdirectoryid": "0",
            "ts_provisioned": "0"
        },
        {
            "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",
            "userdirectoryid": "0",
            "ts_provisioned": "0"
        }
    ],
```

```
    "id": 1
}
```

Pobieranie użytkowników jako *Admin*

Jako użytkownik typu *Admin* pobierz szczegółowe dane o własnym użytkowniku oraz ograniczone dane użytkowników z własnej grupy użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend",
        "getAccess": true,
        "selectMedias": "extend",
        "selectMediatypes": "extend",
        "selectUsrgrps": "extend",
        "selectRole": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "userid": "1",
            "username": "Admin",
            "name": "Zabbix",
            "surname": "Administrator",
            "usrgrps": [
                {
                    "usrgrpid": "7",
                    "name": "Zabbix administrators",
                    "gui_access": "0",
                    "users_status": "0",
                    "debug_mode": "0",
                    "mfa_status": "0"
                }
            ]
        },
        {
            "userid": "3",
            "username": "database-admin",
            "name": "John",
            "surname": "Doe",
            "url": "",
            "autologin": "0",
            "autologout": "0",
            "lang": "default",
            "refresh": "30s",
            "theme": "default",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "50",
            "timezone": "default",
            "roleid": "2",
            "provisioned": "0",
            "gui_access": "0",
            "debug_mode": "0",
```

```
            "users_status": "0",
            "usrgrps": [
                {
                    "usrgrpid": "7",
                    "name": "Zabbix administrators",
                    "gui_access": "0",
                    "users_status": "0",
                    "debug_mode": "0",
                    "mfa_status": "0"
                }
            ],
            "medias": [
                {
                    "mediaid": "2",
                    "mediatypeid": "1",
                    "sendto": [
                        "john.doe@example.com"
                    ],
                    "active": "0",
                    "severity": "63",
                    "period": "1-7,00:00-24:00",
                    "provisioned": 0
                }
            ],
            "mediatypes": [
                {
                    "mediatypeid": "1",
                    "type": "0",
                    "name": "Email",
                    "status": "0",
                    "description": "",
                    "maxattempts": "3"
                }
            ],
            "role": {
                "roleid": "2",
                "name": "Admin role",
                "type": "2",
                "readonly": "0"
            }
        }
    ],
    "id": 1
}
```

Pobieranie danych użytkownika

Pobierz dane użytkownika o ID „12”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": ["userid", "username"],
        "selectRole": "extend",
        "userids": "12"
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
      {
          "userid": "12",
          "username": "John",
          "role": {
              "roleid": "5",
              "name": "Operator",
              "type": "1",
              "readonly": "0"
          }
      }
  ],
  "id": 1
}
```


Zobacz także

- Media
- Typ nośnika
- Grupa użytkowników
- Rola

Źródło

CUser::get() w *ui/include/classes/api/services/CUser.php*.

user.login

Opis

`string/object user.login(object parameters)`

Ta metoda umożliwia zalogowanie się do API i wygenerowanie tokenu uwierzytelniającego.

Warning:
Podczas korzystania z tej metody należy również wykonać user.logout, aby zapobiec generowaniu dużej liczby rekordów otwartych sesji.

Attention:
Ta metoda jest dostępna tylko dla nieuwierzytelnionych użytkowników, którzy nie należą do żadnej grupy użytkowników z włączonym uwierzytelnianiem wieloskładnikowym.

Parametry

`(object)` Parametry zawierające nazwę użytkownika i hasło.

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
password	string	Hasło użytkownika. Zachowanie parametru: - *wymagany*
username	string	Nazwa użytkownika. Zachowanie parametru: - *wymagany*
userData	flag	Zwraca informacje o uwierzytelnionym użytkowniku.

Zwracane wartości

(string/object) Jeśli używany jest parametr userData, zwracany jest obiekt zawierający informacje o uwierzytelnionym użytkowniku.

Oprócz standardowych właściwości użytkownika zwracane są następujące informacje:

Właściwość	Typ	Opis
auth_type	integer	Domyślna metoda uwierzytelniania dla użytkownika. Listę możliwych wartości można znaleźć we właściwości authentication_type obiektu Authentication object.
debug_mode	integer	Określa, czy tryb debugowania jest włączony czy wyłączony dla użytkownika. Listę możliwych wartości można znaleźć we właściwości debug_mode obiektu User group object.
deprovisioned	boolean	Określa, czy użytkownik należy do grupy wycofanych użytkowników.
gui_access	string	Metoda uwierzytelniania użytkownika do frontendu. Listę możliwych wartości można znaleźć we właściwości gui_access obiektu User group object.
mfaid	integer	ID metody MFA, która ma być używana dla użytkownika podczas logowania. Zwraca "0", jeśli MFA jest globalnie wyłączone lub wyłączone dla wszystkich grup użytkowników, do których należy użytkownik.
secret	string	Losowy ciąg 32 znaków. Jest generowany podczas logowania użytkownika.
sessionid	string	Token uwierzytelniający, który musi być używany w kolejnych żądaniach API.
type	integer	Typ użytkownika. Listę możliwych wartości można znaleźć we właściwości type obiektu Role object.
userip	string	Adres IP użytkownika.

Note:
Jeśli użytkownik został pomyślnie uwierzytelniony po jednej lub większej liczbie nieudanych prób, metoda zwróci bieżące wartości właściwości attempt_clock, attempt_failed i attempt_ip, a następnie je zresetuje.

Jeśli parametr userData nie jest używany, metoda zwraca token uwierzytelniający wymagany do uwierzytelniania.

Przykłady

Uwierzytelnianie użytkownika

Uwierzytelnij użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
```

```
    "id": 1
}
```

Żądanie informacji o uwierzytelnionym użytkowniku

Uwierzytelnij i zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix",
        "userData": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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",
        "userdirectoryid": "0",
        "type": 3,
        "userip": "127.0.0.1",
        "debug_mode": 0,
        "gui_access": "0",
        "mfaid": "1",
        "deprovisioned": false,
        "auth_type": 0,
        "sessionid": "5b56eee8be445e98f0bd42b435736e42",
        "secret": "cd0ba923319741c6586f3d866423a8f4"
    },
    "id": 1
}
```

Zobacz także

- user.logout

Źródło

CUser::login() w *ui/include/classes/api/services/CUser.php*.

user.logout

Opis

`string/object user.logout(array)`

Ta metoda umożliwia wylogowanie z API i unieważnia bieżący token uwierzytelniający.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Metoda akceptuje pustą tablicę.

Zwracane wartości

`(boolean)` Zwraca `true`, jeśli użytkownik został pomyślnie wylogowany.

Przykłady

Wylogowanie

Wyloguj się z API.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.logout",
    "params": [],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Zobacz także

- user.login

Źródło

CUser::login() w *ui/include/classes/api/services/CUser.php*.

user.provision

Opis

`object user.provision(object/array users)`

Ta metoda umożliwia provisionowanie użytkowników LDAP.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w Rolach użytkowników.

Parametry

`(array)` ID użytkowników do provisioningu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory provisioned users we właściwości `userids`.

Przykłady

Provisionowanie wielu użytkowników

Provisionuj dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.provision",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::provision() w *ui/include/classes/api/services/CUser.php*.

user.resettotp

Opis

`object user.resettotp(object/array users)`

Ta metoda umożliwia resetowanie sekretów TOTP użytkowników.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników, dla których należy zresetować sekrety TOTP.

Note:
Sesje użytkowników dla wskazanych użytkowników również zostaną usunięte (z wyjątkiem użytkownika wysyłającego żądanie).

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory użytkowników, dla których sekrety TOTP zostały zresetowane, we właściwości `userids`.

Przykłady

Resetowanie sekretów TOTP dla wielu użytkowników

Zresetuj sekrety TOTP dla dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.resettotp",
    "params": [
        "1",
        "5"
```

```
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Obiekt MFA

Źródło

CUser::resettotp() w *ui/include/classes/api/services/CUser.php*.

user.unblock

Opis

`object user.unblock(array userids)`

Ta metoda umożliwia odblokowanie użytkowników.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników do odblokowania.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory odblokowanych użytkowników we właściwości `userids`.

Przykłady

Odblokowywanie wielu użytkowników

Odblokuj dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.unblock",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
```

```
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::unblock() w *ui/include/classes/api/services/CUser.php*.

user.update

Opis

`object user.update(object/array users)`

Ta metoda umożliwia aktualizację istniejących użytkowników.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Note:
Siła hasła użytkownika jest weryfikowana zgodnie z regułami polityki haseł zdefiniowanymi przez Authentication API. Więcej informacji można znaleźć w sekcji Authentication API.

Parametry

`(object/array)` Właściwości użytkownika do zaktualizowania.

Właściwość `userid` musi być zdefiniowana dla każdego użytkownika, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości użytkownika metoda akceptuje następujące parametry.

Parametr	Typ	Opis
current_passwd	string	Bieżące hasło użytkownika. Wartość tego parametru może być pustym ciągiem znaków, jeśli użytkownik jest powiązany z katalogiem użytkowników. Zachowanie parametru: - *tylko do zapisu* - *wymagany*, jeśli `passwd` obiektu User object jest ustawione i użytkownik zmienia własne hasło
usrgrps	array	Grupy użytkowników zastępujące istniejące grupy użytkownika. Dla grup użytkowników musi być zdefiniowana tylko właściwość `usrgrpid`.
medias	array	Media użytkownika zastępujące istniejące, nieudostępnione media. Udostępnione media można pominąć podczas aktualizacji mediów.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych użytkowników we właściwości `userids`.

Przykłady

Zmiana nazwy użytkownika

Zmień nazwę użytkownika na John Doe.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "1",
        "name": "John",
        "surname": "Doe"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1"
        ]
    },
    "id": 1
}
```

Zmiana roli użytkownika

Zmień rolę użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "12",
        "roleid": "6"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uwierzytelnianie

Źródło

CUser::update() w *ui/include/classes/api/services/CUser.php*.

Wykres

Ta klasa jest przeznaczona do pracy z wykresami.

Odniesienia do obiektów:

- Wykres

Dostępne metody:

- graph.create - tworzenie nowego wykresu
- graph.delete - usuwanie wykresu
- graph.get - pobieranie wykresu
- graph.update - aktualizacja wykresu

Obiekt Graph

Następujące obiekty są bezpośrednio powiązane z API `graph`.

Wykres

Obiekt wykresu ma następujące właściwości.

Właściwość	Typ	Opis
graphid	ID	ID wykresu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
height	integer	Wysokość wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa wykresu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie wykresu. Możliwe wartości: 0 - *(domyślnie)* zwykły wykres; 4 - wykres przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*
graphtype	integer	Typ układu wykresu. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - skumulowany; 2 - kołowy; 3 - rozstrzelony.
percent_left	float	Lewy percentyl. Domyślnie: 0.
percent_right	float	Prawy percentyl. Domyślnie: 0.
show_3d	integer	Czy wykresy kołowe i rozstrzelone mają być wyświetlane w 3D. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj w 2D; 1 - wyświetlaj w 3D.
show_legend	integer	Czy wyświetlać legendę na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.

Właściwość	Typ	Opis
show_work_period	integer	Czy wyświetlać czas pracy na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
show_triggers	integer	Czy wyświetlać linię wyzwalacza na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
templateid	ID	ID wykresu nadrzędnego szablonu. Zachowanie właściwości: - *tylko do odczytu*
yaxismax	float	Stała maksymalna wartość osi Y. Domyślnie: 100.
yaxismin	float	Stała minimalna wartość osi Y. Domyślnie: 0.
ymax_itemid	ID	ID pozycji używanej jako maksymalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymax_type` było ustawione na „calculated”.
ymax_type	integer	Metoda obliczania wartości maksymalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
ymin_itemid	ID	ID pozycji używanej jako minimalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymin_type` było ustawione na „calculated”.
ymin_type	integer	Metoda obliczania wartości minimalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych wykresów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli wykres należy do szablonu

graph.create

Opis

`object graph.create(object/array graphs)`

Ta metoda umożliwia tworzenie nowych wykresów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

(object/array) Wykresy do utworzenia.

Oprócz standardowych właściwości wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Pozycje wykresu, które mają zostać utworzone dla wykresu. Zachowanie parametru: - *wymagane*

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych wykresów we właściwości graphids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych wykresów.

Przykłady

Tworzenie wykresu

Utwórz wykres z dwiema pozycjami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graph.create",
    "params": {
        "name": "MySQL bandwidth",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pozycja wykresu

Źródło

CGraph::create() w *ui/include/classes/api/services/CGraph.php*.

graph.delete

Opis

`object graph.delete(array graphIds)`

Ta metoda umożliwia usuwanie grafów.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID wykresów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wykresów we właściwości `graphids`.

Przykłady

Usuwanie wielu wykresów

Usunięcie dwóch wykresów.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "graph.delete",
    "params": [
        "652",
        "653"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Źródło

CGraph::delete() w *ui/include/classes/api/services/CGraph.php*.

graph.get

Opis

`integer/array graph.get(object parameters)`

Ta metoda umożliwia pobieranie wykresów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwracaj tylko wykresy o podanych ID.
groupids	ID/array	Zwracaj tylko wykresy należące do hostów lub szablonów w podanych grupach hostów lub grupach szablonów.
templateids	ID/array	Zwracaj tylko wykresy należące do podanych szablonów.
hostids	ID/array	Zwracaj tylko wykresy należące do podanych hostów.
itemids	ID/array	Zwracaj tylko wykresy zawierające podane pozycje.
templated	boolean	Jeśli ustawiono `true`, zwracaj tylko wykresy należące do szablonów.
inherited	boolean	Jeśli ustawiono `true`, zwracaj tylko wykresy odziedziczone z szablonu.
expandName	flag	Rozwiń makra w nazwie wykresu.
selectDiscoveryData	query	Zwróć właściwość `discoveryData` z danymi obiektu wykrywania wykresu. Obiekt wykrywania wykresu łączy wykryty wykres z prototypem wykresu, z którego został wykryty. Ma następujące właściwości: `parent_graphid` - `(ID)` ID prototypu wykresu, z którego wykres został utworzony; `status` - `(int)` status wykrywania wykresu: 0 - *(domyślnie)* wykres jest wykryty, 1 - wykres nie jest już wykryty; `ts_delete` - `(timestamp)` czas, kiedy wykres, który nie jest już wykrywany, zostanie usunięty.
selectDiscoveryRule	query	Zwróć właściwość `discoveryRule` z regułą wykrywania niskiego poziomu, która utworzyła wykres.
selectHostGroups	query	Zwróć właściwość `hostgroups` z grupami hostów, do których należy wykres.
selectTemplateGroups	query	Zwróć właściwość `templategroups` z grupami szablonów, do których należy wykres.
selectTemplates	query	Zwróć właściwość `templates` z szablonami, do których należy wykres.
selectHosts	query	Zwróć właściwość `hosts` z hostami, do których należy wykres.
selectItems	query	Zwróć właściwość `items` z pozycjami użytymi na wykresie.
selectGraphItems	query	Zwróć właściwość `gitems` z pozycjami użytymi na wykresie.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy wykres; `hostid` - ID hosta, do którego należy wykres.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `graphid`, `name`, `graphtype`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Parametr	Typ	Opis
selectGraphDiscovery	query	Zwróć właściwość `graphDiscovery` z obiektem wykrywania wykresu. Obiekty wykrywania wykresu łączą wykres z prototypem wykresu, z którego został utworzony. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wykresów z hostów

Pobierz wszystkie wykresy z hosta „10107” i posortuj je według nazwy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graph.get",
    "params": {
        "output": "extend",
        "hostids": 10107,
        "sortfield": "name"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "612",
            "name": "CPU jumps",
            "width": "900",
            "height": "200",
            "yaxismin": "0",
            "yaxismax": "100",
            "templateid": "439",
            "show_work_period": "1",
            "show_triggers": "1",
            "graphtype": "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",
            "flags": "0"
        },
        {
            "graphid": "613",
            "name": "CPU load",
            "width": "900",
            "height": "200",
            "yaxismin": "0",
```

```
            "yaxismax": "100",
            "templateid": "433",
            "show_work_period": "1",
            "show_triggers": "1",
            "graphtype": "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",
            "flags": "0"
        },
        {
            "graphid": "614",
            "name": "CPU utilization",
            "width": "900",
            "height": "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",
            "flags": "0"
        },
        {
            "graphid": "645",
            "name": "Disk space usage /",
            "width": "600",
            "height": "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",
            "flags": "4"
        }
    ],
    "id": 1
}
```


Zobacz także

- Reguła wykrywania
- Pozycja wykresu
- Pozycja
- Host
- Grupa hostów
- Szablon
- Grupa szablonów

Źródło

CGraph::get() w *ui/include/classes/api/services/CGraph.php*.

graph.update

Opis

`object graph.update(object/array graphs)`

Ta metoda służy do aktualizacji istniejących wykresów.

> **Note:**
> Ta metoda jest dostępna dla użytkowników typu *Admin* i *Super admin*. Uprawnienia mogą zostać zostać cofnięte w konfiguracji roli użytkownika. Więcej informacji Rola użytkownika

Parametry

`(object/array)` Właściwości wykresu do zaktualizowania.

Właściwość `graphid` musi być zdefiniowana dla każdego wykresu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości wykresu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
gitems	array	Pozycje wykresu służące do zastąpienia istniejących pozycji wykresu. Jeśli pozycja wykresu ma zdefiniowaną właściwość `gitemid`, zostanie zaktualizowana; w przeciwnym razie zostanie utworzona nowa pozycja wykresu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wykresów we właściwości `graphids`.

Przykłady

Ustawianie maksimum skali Y

Ustaw maksimum skali Y na stałą wartość 100.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graph.update",
    "params": {
        "graphid": "439",
        "ymax_type": 1,
        "yaxismax": 100
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
```

```
            "439"
        ]
    },
    "id": 1
}
```

Źródło

CGraph::update() w *ui/include/classes/api/services/CGraph.php*.

Wykryta usługa

Ta klasa została zaprojektowana do pracy z wykrytymi usługami.

Odwołania do obiektów:

- Wykryta usługa

Dostępne metody:

- dservice.get - pobieranie wykrytych usług

Obiekt odkrywania usługi

Następujące obiekty są bezpośrednio powiązane z API `dservice`.

Wykryta usługa

Note:
Wykryte usługi są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pomocą API.

Obiekt wykrytej usługi zawiera informacje o usłudze wykrytej przez regułę wykrywania sieci na hoście. Ma następujące właściwości.

Właściwość	Typ	Opis
dserviceid	ID	ID wykrytej usługi.
dcheckid	ID	ID kontroli wykrywania użytej do wykrycia usługi.
dhostid	ID	ID wykrytego hosta, na którym działa usługa.
dns	string	DNS hosta, na którym działa usługa.
ip	string	Adres IP hosta, na którym działa usługa.
lastdown	timestamp	Czas, kiedy wykryta usługa ostatnio przestała działać.
lastup	timestamp	Czas, kiedy wykryta usługa ostatnio zaczęła działać.
port	integer	Numer portu usługi.
status	integer	Status usługi. Możliwe wartości: 0 - usługa działa; 1 - usługa nie działa.
value	string	Wartość zwrócona przez usługę podczas wykonywania kontroli wykrywania agent, SNMPv1, SNMPv2 lub SNMPv3.

dservice.get

Opis

`integer/array dservice.get(object parameters)`

Ta metoda umożliwia pobranie wykrytych usług zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

(object) Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dserviceids	ID/array	Zwraca tylko wykryte usługi o podanych identyfikatorach.
dhostids	ID/array	Zwraca tylko wykryte usługi należące do podanych wykrytych hostów.
dcheckids	ID/array	Zwraca tylko wykryte usługi, które zostały wykryte przez podane kontrole wykrywania.
druleids	ID/array	Zwraca tylko wykryte usługi, które zostały wykryte przez podane reguły wykrywania.
selectDRules	query	Zwraca właściwość `drules` zawierającą tablicę reguł wykrywania, które wykryły usługę.
selectDHosts	query	Zwraca właściwość `dhosts` zawierającą tablicę wykrytych hostów, do których należy usługa.
selectHosts	query	Zwraca właściwość `hosts` zawierającą hosty z tym samym adresem IP i proxy co usługa. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wynik zostanie posortowany według `hostid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dserviceid`, `dhostid`, `ip`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

(integer/array) Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie usług wykrytych na hoście

Pobierz wszystkie wykryte usługi znalezione na wykrytym hoście „11”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dservice.get",
    "params": {
        "output": "extend",
        "dhostids": "11"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "dserviceid": "12",
            "dhostid": "11",
            "value": "",
            "port": "80",
            "status": "1",
            "lastup": "0",
            "lastdown": "1348650607",
            "dcheckid": "5",
            "ip": "192.168.1.134",
            "dns": "john.local"
        },
        {
            "dserviceid": "13",
            "dhostid": "11",
            "value": "",
            "port": "21",
            "status": "1",
            "lastup": "0",
            "lastdown": "1348650610",
            "dcheckid": "6",
            "ip": "192.168.1.134",
            "dns": "john.local"
        }
    ],
    "id": 1
}
```


Zobacz także

- Wykryty host
- Kontrola wykrywania
- Host

Źródło

CDService::get() w *ui/include/classes/api/services/CDService.php*.

Wykryty host

Ta klasa została zaprojektowana do pracy z wykrytymi hostami.

Odwołania do obiektów:

- Wykryty host

Dostępne metody:

- dhost.get - pobieranie wykrytych hostów

Obiekt odkrywania hosta

Następujące obiekty są bezpośrednio powiązane z API `dhost`.

Wykryty host

> **Note:**
> Wykryte hosty są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Obiekt wykrytego hosta zawiera informacje o hoście wykrytym przez regułę wykrywania sieci. Ma następujące właściwości.

Właściwość	Typ	Opis
dhostid	ID	ID wykrytego hosta.
druleid	ID	ID reguły wykrywania, która wykryła host.
lastdown	timestamp	Czas, kiedy wykryty host ostatnio przestał być dostępny.
lastup	timestamp	Czas, kiedy wykryty host ostatnio stał się dostępny.
status	integer	Czy wykryty host jest dostępny, czy niedostępny. Host jest dostępny, jeśli ma co najmniej jedną aktywną wykrytą usługę. Możliwe wartości: 0 - host dostępny; 1 - host niedostępny.

dhost.get

Opis

`integer/array dhost.get(object parameters)`

Ta metoda umożliwia pobieranie wykrytych hostów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dhostids	ID/array	Zwraca tylko wykryte hosty o podanych identyfikatorach.
druleids	ID/array	Zwraca tylko wykryte hosty, które zostały utworzone przez podane reguły wykrywania.
dserviceids	ID/array	Zwraca tylko wykryte hosty, na których działają podane usługi.
selectDRules	query	Zwraca właściwość `drules` zawierającą tablicę reguł wykrywania, które wykryły host.
selectDServices	query	Zwraca właściwość `dservices` zawierającą wykryte usługi działające na hoście. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Ma zastosowanie do następujących podzapytań: `selectDServices` - wyniki zostaną posortowane według `dserviceid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dhostid`, `druleid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobierz odkryte hosty według reguły wykrywania

Pobierz wszystkie hosty i odkryte usługi, które uruchamiają, które zostały wykryte przez regułę wykrywania „4”.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "dhost.get",
  "params": {
    "output": "extend",
    "selectDServices": "extend",
    "druleids": "4"
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "dservices": [
        {
          "dserviceid": "1",
          "dhostid": "1",
          "type": "4",
          "key_": "",
          "value": "",
          "port": "80",
          "status": "0",
          "lastup": "1337697227",
          "lastdown": "0",
          "dcheckid": "5",
          "ip": "192.168.1.1",
          "dns": "station.company.lan"
        }
      ],
      "dhostid": "1",
      "druleid": "4",
      "status": "0",
      "lastup": "1337697227",
      "lastdown": "0"
    },
    {
      "dservices": [
        {
          "dserviceid": "2",
          "dhostid": "2",
          "type": "4",
          "key_": "",
          "value": "",
          "port": "80",
          "status": "0",
          "lastup": "1337697234",
          "lastdown": "0",
          "dcheckid": "5",
```

```
          "ip": "192.168.1.4",
          "dns": "john.company.lan"
        }
      ],
      "dhostid": "2",
      "druleid": "4",
      "status": "0",
      "lastup": "1337697234",
      "lastdown": "0"
    },
    {
      "dservices": [
        {
          "dserviceid": "3",
          "dhostid": "3",
          "type": "4",
          "key_": "",
          "value": "",
          "port": "80",
          "status": "0",
          "lastup": "1337697234",
          "lastdown": "0",
          "dcheckid": "5",
          "ip": "192.168.1.26",
          "dns": "printer.company.lan"
        }
      ],
      "dhostid": "3",
      "druleid": "4",
      "status": "0",
      "lastup": "1337697234",
      "lastdown": "0"
    },
    {
      "dservices": [
        {
          "dserviceid": "4",
          "dhostid": "4",
          "type": "4",
          "key_": "",
          "value": "",
          "port": "80",
          "status": "0",
          "lastup": "1337697234",
          "lastdown": "0",
          "dcheckid": "5",
          "ip": "192.168.1.7",
          "dns": "mail.company.lan"
        }
      ],
      "dhostid": "4",
      "druleid": "4",
      "status": "0",
      "lastup": "1337697234",
      "lastdown": "0"
    }
  ],
  "id": 1
}
```


Zobacz także

- Wykryta usługa

- Reguła wykrywania

Źródło

CDHost::get() w *ui/include/classes/api/services/CDHost.php*.

Wyrażenie regularne

Ta klasa jest przeznaczona do pracy z globalnymi wyrażeniami regularnymi.

Odwołania do obiektów:

- Wyrażenie regularne
- Wyrażenia

Dostępne metody:

- regexp.create - tworzenie nowych wyrażeń regularnych
- regexp.delete - usuwanie wyrażeń regularnych
- regexp.get - pobieranie wyrażeń regularnych
- regexp.update - aktualizowanie wyrażeń regularnych

Obiekt wyrażenia regularnego

Następujące obiekty są bezpośrednio powiązane z API `regexp`.

Wyrażenie regularne

Globalny obiekt wyrażenia regularnego ma następujące właściwości.

Właściwość	Typ	Opis
regexpid	ID	ID wyrażenia regularnego. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa wyrażenia regularnego. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
test_string	string	Ciąg testowy.

Wyrażenia

Obiekt expressions ma następujące właściwości.

Właściwość	Typ	Opis
expression	string	Wyrażenie regularne. Zachowanie właściwości: - *wymagane*
expression_type	integer	Typ wyrażenia regularnego. Możliwe wartości: 0 - Zawiera ciąg znaków; 1 - Zawiera dowolny ciąg znaków; 2 - Nie zawiera ciągu znaków; 3 - Wynik to TRUE; 4 - Wynik to FALSE. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
exp_delimiter	string	Separator wyrażenia. Wartość domyślna: ",". Możliwe wartości: "," lub ".", lub "/". Zachowanie właściwości: - *obsługiwane* jeśli `expression_type` jest ustawione na „Zawiera dowolny ciąg znaków”
case_sensitive	integer	Rozróżnianie wielkości liter. Wartość domyślna: 0. Możliwe wartości: 0 - Bez rozróżniania wielkości liter; 1 - Z rozróżnianiem wielkości liter.

regexp.create

Opis

`object regexp.create(object/array regularExpressions)`

Ta metoda umożliwia tworzenie nowych globalnych wyrażeń regularnych.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Wyrażenia regularne do utworzenia.

Oprócz standardowych właściwości metoda akceptuje następujące parametry.

Parametr	Typ	Opis
expressions	array	Opcje wyrażeń. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Tworzenie nowego globalnego wyrażenia regularnego.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.create",
    "params": {
      "name": "Urządzenia pamięci masowej do wykrywania SNMP",
      "test_string": "/boot",
      "expressions": [
        {
          "expression": "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$",
          "expression_type": "4",
          "case_sensitive": "1"
        }
```

```
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::create() w *ui/include/classes/api/services/CRegexp.php*.

regexp.delete

Opis

`object regexp.delete(array regexpids)`

Ta metoda umożliwia usuwanie globalnych wyrażeń regularnych.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID wyrażeń regularnych do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Usuwanie wielu globalnych wyrażeń regularnych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.delete",
    "params": [
        "16",
        "17"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16",
            "17"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::delete() w *ui/include/classes/api/services/CRegexp.php*.

regexp.get

Opis

`integer/array regexp.get(object parameters)`

Ta metoda umożliwia pobieranie globalnych wyrażeń regularnych zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla *Super Admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
regexpids	ID/array	Zwraca tylko wyrażenia regularne o podanych identyfikatorach.
selectExpressions	query	Zwraca właściwość `expressions`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `regexpid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie globalnych wyrażeń regularnych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.get",
    "params": {
        "output": ["regexpid", "name"],
        "selectExpressions": ["expression", "expression_type"],
        "regexpids": [1, 2],
        "preservekeys": true
    },
    "id": 1
}
```

Odpowiedź:


```
{
  "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
}
```


Źródło

CRegexp::get() w *ui/include/classes/api/services/CRegexp.php*.

regexp.update

Opis

`object regexp.update(object/array regularExpressions)`

Ta metoda umożliwia aktualizację istniejących globalnych wyrażeń regularnych.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości wyrażenia regularnego do zaktualizowania.

Właściwość `regexpid` musi być zdefiniowana dla każdego obiektu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz właściwości standardowych metoda akceptuje następujące parametry.

Parametr	Type	Opis
expressions	array	Opcje wyrażeń.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Aktualizacja globalnego wyrażenia regularnego dla wykrywania systemów plików.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.update",
    "params": {
      "regexpid": "1",
      "name": "Systemy plików do wykrywania",
      "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"
        }
      ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::update() w *ui/include/classes/api/services/CRegexp.php*.

Wyzwalacz

Ta klasa jest przeznaczona do pracy z wyzwalaczami.

Odwołania do obiektów:

- Wyzwalacz
- Tag wyzwalacza

Dostępne metody:

- trigger.create - tworzenie nowych wyzwalaczy
- trigger.delete - usuwanie wyzwalaczy
- trigger.get - pobieranie wyzwalaczy
- trigger.update - aktualizacja wyzwalaczy

Obiekt wyzwalacza

Następujące obiekty są bezpośrednio powiązane z API `trigger`.

Wyzwalacz

Obiekt wyzwalacza ma następujące właściwości.

Property	Type	Description
triggerid	ID	ID wyzwalacza. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Nazwa wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
expression	string	Wyrażenie wyzwalacza. Pozycje z `value_type` ustawionym na 5 (binary) nie mogą być używane w wyrażeniach wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
event_name	string	Nazwa zdarzenia generowanego przez wyzwalacz.
opdata	string	Dane operacyjne.
comments	string	Dodatkowy opis wyzwalacza.
error	string	Tekst błędu, jeśli podczas aktualizacji stanu wyzwalacza wystąpiły jakiekolwiek problemy. Zachowanie właściwości: - *tylko do odczytu*
flags	integer	Pochodzenie wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* zwykły wyzwalacz; 4 - wyzwalacz przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*
lastchange	timestamp	Czas ostatniej zmiany stanu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*
priority	integer	Istotność wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* nie sklasyfikowano; 1 - informacja; 2 - ostrzeżenie; 3 - średnia; 4 - wysoka; 5 - katastrofa.

Property	Type	Description
state	integer	Stan wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* stan wyzwalacza jest aktualny; 1 - bieżący stan wyzwalacza jest nieznany. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Określa, czy wyzwalacz jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
templateid	ID	ID nadrzędnego wyzwalacza szablonu. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Określa, czy wyzwalacz może generować wiele zdarzeń problemu. Możliwe wartości: 0 - *(domyślnie)* nie generuj wielu zdarzeń; 1 - generuj wiele zdarzeń.
url	string	URL powiązany z wyzwalaczem.
url_name	string	Etykieta dla URL powiązanego z wyzwalaczem.
value	integer	Określa, czy wyzwalacz znajduje się w stanie OK, czy problemu. Możliwe wartości: 0 - *(domyślnie)* OK; 1 - problem. Zachowanie właściwości: - *tylko do odczytu*
recovery_mode	integer	Tryb generowania zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* wyrażenie; 1 - wyrażenie odzyskiwania; 2 - brak.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza. Pozycje z `value_type` ustawionym na 5 (binary) nie mogą być używane w wyrażeniach odzyskiwania wyzwalacza.
correlation_mode	integer	Zamykanie zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* wszystkie problemy; 1 - wszystkie problemy, jeśli wartości tagów są zgodne.
correlation_tag	string	Tag do dopasowania.
manual_close	integer	Zezwalaj na ręczne zamknięcie. Możliwe wartości: 0 - *(domyślnie)* nie; 1 - tak.
uuid	string	Uniwersalny unikalny identyfikator używany do powiązania zaimportowanych wyzwalaczy z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane* jeśli wyzwalacz należy do szablonu

Znacznik wyzwalacza

Obiekt znacznika wyzwalacza ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa znacznika wyzwalacza. Property behavior: - *required*
value	string	Wartość znacznika wyzwalacza.

trigger.create

Opis

`object trigger.create(object/array triggers)`

Ta metoda umożliwia tworzenie nowych wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Wyzwalacze do utworzenia.

Oprócz standardowych właściwości wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze, od których zależy wyzwalacz. Wyzwalacze muszą mieć zdefiniowaną tylko właściwość `triggerid`.
tags	array	Tagi wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych wyzwalaczy we właściwości `triggerids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych wyzwalaczy.

Przykłady

Tworzenie wyzwalacza

Utwórz dwa wyzwalacze, z których każdy zależy od jednego innego wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.create",
    "params": [
        {
            "description": "Obciążenie procesora jest zbyt wysokie na {HOST.NAME}",
            "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
            "dependencies": [
                {
                    "triggerid": "17367"
                }
            ]
        },
        {
            "description": "Usługa została zatrzymana",
            "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": ""
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 1
}
```

Źródło

CTrigger::create() w *ui/include/classes/api/services/CTrigger.php*.

trigger.delete

Opis

`object trigger.delete(array triggerIds)`

Ta metoda umożliwia usuwanie wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania metody można odwołać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w User roles.

Parametry

`(array)` ID wyzwalaczy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wyzwalaczy we właściwości `triggerids`.

Przykłady

Usuń wiele wyzwalaczy

Usuń dwa wyzwalacze.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "trigger.delete",
"params": [
"12002",
```

```
"12003"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"triggerids": [
"12002",
"12003"
]
},
"id": 1
}
```

Źródło

CTrigger::delete() w *ui/include/classes/api/services/CTrigger.php*.

trigger.get

Opis

`integer/array trigger.get(object parameters)`

Metoda umożliwia pobieranie wyzwalaczy zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
triggerids	ID/array	Zwracaj tylko wyzwalacze o podanych ID.
groupids	ID/array	Zwracaj tylko wyzwalacze należące do hostów lub szablonów z podanych grup hostów lub grup szablonów.
templateids	ID/array	Zwracaj tylko wyzwalacze należące do podanych szablonów.
hostids	ID/array	Zwracaj tylko wyzwalacze należące do podanych hostów.
itemids	ID/array	Zwracaj tylko wyzwalacze zawierające podane pozycje.
functions	string/array	Zwracaj tylko wyzwalacze, które używają podanych funkcji. Listę obsługiwanych funkcji znajdziesz na stronie supported function.
group	string	Zwracaj tylko wyzwalacze należące do hostów lub szablonów z grupy hostów lub grupy szablonów o podanej nazwie.
host	string	Zwracaj tylko wyzwalacze należące do hosta o podanej nazwie technicznej.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko wyzwalacze odziedziczone z szablonu.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko wyzwalacze należące do szablonów.
dependent	boolean	Jeśli ustawione na `true`, zwracaj tylko wyzwalacze, które mają zależności. Jeśli ustawione na `false`, zwracaj tylko wyzwalacze, które nie mają zależności.
monitored	flag	Zwracaj tylko włączone wyzwalacze należące do monitorowanych hostów i zawierające wyłącznie włączone pozycje.
active	flag	Zwracaj tylko włączone wyzwalacze należące do monitorowanych hostów.

Parameter	Type	Description
maintenance	boolean	Jeśli ustawione na `true`, zwracaj tylko włączone wyzwalacze należące do hostów w konserwacji.
withUnacknowledgedEvents	flag	Zwracaj tylko wyzwalacze z niepotwierdzonymi zdarzeniami.
withAcknowledgedEvents	flag	Zwracaj tylko wyzwalacze, dla których wszystkie zdarzenia zostały potwierdzone.
withLastEventUnacknowledged	flag	Zwracaj tylko wyzwalacze, dla których ostatnie zdarzenie jest niepotwierdzone.
skipDependent	flag	Pomiń wyzwalacze w stanie problemu, które są zależne od innych wyzwalaczy. Zwróć uwagę, że inne wyzwalacze są ignorowane, jeśli są wyłączone, mają wyłączone pozycje lub wyłączonych hostów pozycji.
lastChangeSince	timestamp	Zwracaj tylko wyzwalacze, które zmieniły swój stan po podanym czasie.
lastChangeTill	timestamp	Zwracaj tylko wyzwalacze, które zmieniły swój stan przed podanym czasem.
only_true	flag	Zwracaj tylko wyzwalacze, które ostatnio były w stanie problemu.
min_severity	integer	Zwracaj tylko wyzwalacze o ważności większej lub równej podanej ważności.
evaltype	integer	Tag evaluation method. Possible values: 0 - *(default)* And/Or; 2 - Or.
tags	array	Zwracaj tylko wyzwalacze z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie wyzwalacze. Możliwe wartości operator: 0 - *(default)* Contains; 1 - Equals; 2 - Does not contain; 3 - Does not equal; 4 - Exists; 5 - Does not exist.
expandComment	flag	Rozwiń makra w opisie wyzwalacza.
expandDescription	flag	Rozwiń makra w nazwie wyzwalacza.
expandExpression	flag	Rozwiń funkcje i makra w wyrażeniu wyzwalacza.
selectHostGroups	query	Zwracaj grupy hostów, do których należy wyzwalacz, w właściwości `hostgroups`.
selectHosts	query	Zwracaj hosty, do których należy wyzwalacz, w właściwości `hosts`.
selectItems	query	Zwracaj pozycje zawarte w wyzwalaczu w właściwości `items`.
selectFunctions	query	Zwracaj funkcje używane w wyzwalaczu w właściwości `functions`. Obiekty funkcji reprezentują funkcje używane w wyrażeniu wyzwalacza i mają następujące właściwości: `functionid` - `(ID)` ID funkcji; `itemid` - `(ID)` ID pozycji używanej w funkcji; `function` - `(string)` nazwa funkcji; `parameter` - `(string)` parametr przekazany do funkcji. Parametr zapytania jest zastępowany symbolem $ w zwracanym ciągu.
selectDependencies	query	Zwracaj wyzwalacze, od których zależy wyzwalacz, w właściwości `dependencies`.

Parameter	Type	Description
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania wyzwalacza. Obiekt wykrywania wyzwalacza łączy wykryty wyzwalacz z prototypem wyzwalacza, z którego został wykryty. Ma następujące właściwości: `parent_triggerid` - `(ID)` ID prototypu wyzwalacza, z którego został utworzony wyzwalacz; `status` - `(int)` status wykrywania wyzwalacza: 0 - *(default)* wyzwalacz został wykryty, 1 - wyzwalacz nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, po którym wyzwalacz, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, po którym wyzwalacz, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik, czy wyzwalacz został wyłączony przez regułę LLD, czy ręcznie: 0 - *(default)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj regułę wykrywania niskopoziomowego, która utworzyła wyzwalacz, w właściwości `discoveryRule`.
selectLastEvent	query	Zwracaj ostatnie istotne zdarzenie wyzwalacza w właściwości `lastEvent`.
selectTags	query	Zwracaj tagi wyzwalacza w właściwości `tags`.
selectTemplateGroups	query	Zwracaj grupy szablonów, do których należy wyzwalacz, w właściwości `templategroups`.
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma nastąpić dopasowanie. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy wyzwalacz; `hostid` - ID hosta, do którego należy wyzwalacz.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki będą sortowane według `host`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `triggerid`, `description`, `status`, `priority`, `lastchange`, `hostname`.
countOutput	boolean	Te parametry są opisane w reference commentary.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectTriggerDiscovery	query	Zwracaj obiekt wykrywania wyzwalacza w właściwości `triggerDiscovery`. Obiekty wykrywania wyzwalacza łączą wyzwalacz z prototypem wyzwalacza, z którego został utworzony. To zapytanie jest **przestarzałe**, zamiast niego użyj `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych według ID wyzwalacza

Pobierz wszystkie dane oraz funkcje użyte w wyzwalaczu ”14062”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "triggerids": "14062",
        "output": "extend",
        "selectFunctions": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "",
            "event_name": "",
            "uuid": "",
            "url_name": "",
            "functions": [
                {
                    "functionid": "13513",
                    "itemid": "24350",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie wyzwalaczy w stanie problemu

Pobierz identyfikator, nazwę i ważność wszystkich wyzwalaczy w stanie problemu i posortuj je według ważności malejąco.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description",
            "priority"
        ],
        "filter": {
            "value": 1
        },
        "sortfield": "priority",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13907",
            "description": "Procesy samomonitorowania Zabbix zajęte w mniej niż 100%",
            "priority": "4"
        },
        {
            "triggerid": "13824",
            "description": "Procesy discoverera Zabbix zajęte w więcej niż 75%",
            "priority": "3"
        }
    ],
    "id": 1
}
```

Pobieranie konkretnego wyzwalacza z tagami

Pobierz konkretny wyzwalacz z tagami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17578"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Reguła wykrywania
- Pozycja
- Host
- Grupa hostów
- Grupa szablonów

Źródło

CTrigger::get() w *ui/include/classes/api/services/CTrigger.php*.

trigger.update

Opis

`object trigger.update(object/array triggers)`

Ta metoda umożliwia aktualizację istniejących wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości wyzwalacza do zaktualizowania.

Właściwość `triggerid` musi być zdefiniowana dla każdego wyzwalacza, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości wyzwalacza metoda akceptuje następujące parametry.

Parametr	Type	Opis
dependencies	array	Wyzwalacze, od których zależy wyzwalacz. W wyzwalaczach musi być zdefiniowana tylko właściwość `triggerid`.
tags	array	Tagi wyzwalacza.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wyzwalaczy we właściwości `triggerids`.

Przykłady

Włączanie wyzwalacza

Włącz wyzwalacz, czyli ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie tagów wyzwalacza

Zastąp tagi dla wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "tags": [
            {
                "tag": "service",
                "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}"
            },
            {
                "tag": "error",
                "value": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie zależności

Zastąp zależności dla wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "22713",
        "dependencies": [
            {
                "triggerid": "22712"
            },
            {
                "triggerid": "22772"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "22713"
        ]
    },
    "id": 1
}
```

Źródło

CTrigger::update() w *ui/include/classes/api/services/CTrigger.php*.

Węzeł wysokiej dostępności

Ta klasa została zaprojektowana do pracy z węzłami serwera, które są częścią klastra wysokiej dostępności lub samodzielnej instancji serwera.

Odwołania do obiektów:

- Węzeł wysokiej dostępności

Dostępne metody:

- hanode.get - pobieranie węzłów

Obiekt węzła wysokiej dostępności

Poniższy obiekt jest powiązany z działaniem klastra wysokiej dostępności serwerów Zabbix.

Węzeł wysokiej dostępności

> **Note:**
> Węzły są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt węzła wysokiej dostępności ma następujące właściwości.

Właściwość	Typ	Opis
ha_nodeid	ID	ID węzła.
name	string	Nazwa przypisana do węzła przy użyciu parametru konfiguracyjnego HANodeName w pliku zabbix_server.conf. Puste dla serwera działającego w trybie samodzielnym.
address	string	Nazwa IP lub DNS, z której łączy się węzeł.
port	integer	Port, na którym działa węzeł.

Właściwość	Typ	Opis
lastaccess	integer	Czas heartbeat, czyli czas ostatniej aktualizacji z węzła. Znacznik czasu UTC.
status	integer	Stan węzła. Możliwe wartości: 0 - zapasowy; 1 - zatrzymany ręcznie; 2 - niedostępny; 3 - aktywny.

hanode.get

Opis

`integer/array hanode.get(object parameters)`

Ta metoda umożliwia pobranie listy węzłów klastra wysokiej dostępności zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
ha_nodeids	ID/array	Zwraca tylko węzły o podanych identyfikatorach węzłów.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Obsługiwane właściwości: `name`, `address`, `status`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`, `lastaccess`, `status`.
countOutput	flag	Te parametry opisano w komentarzu referencyjnym.
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz listę węzłów uporządkowaną według statusu

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "preservekeys": true,
```

```
        "sortfield": "status",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": {
    "ckuo7i1nw000h0sajj3l3hh8u": {
      "ha_nodeid": "ckuo7i1nw000h0sajj3l3hh8u",
      "name": "node-active",
      "address": "192.168.1.13",
      "port": "10051",
      "lastaccess": "1635335704",
      "status": "3"
    },
    "ckuo7i1nw000e0sajwfttc1mp": {
      "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
      "name": "node6",
      "address": "192.168.1.10",
      "port": "10053",
      "lastaccess": "1635332902",
      "status": "2"
    },
    "ckuo7i1nv000c0sajz85xcrtt": {
      "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
      "name": "node4",
      "address": "192.168.1.8",
      "port": "10052",
      "lastaccess": "1635334214",
      "status": "1"
    },
    "ckuo7i1nv000a0saj1fcdkeu4": {
      "ha_nodeid": "ckuo7i1nv000a0saj1fcdkeu4",
      "name": "node2",
      "address": "192.168.1.6",
      "port": "10051",
      "lastaccess": "1635335705",
      "status": "0"
    }
  },
  "id": 1
}
```

Pobierz listę określonych węzłów według ich identyfikatorów

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "ha_nodeids": ["ckuo7i1nw000e0sajwfttc1mp", "ckuo7i1nv000c0sajz85xcrtt"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
```

```
            "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
            "name": "node4",
            "address": "192.168.1.8",
            "port": "10052",
            "lastaccess": "1635334214",
            "status": "1"
        },
        {
            "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
            "name": "node6",
            "address": "192.168.1.10",
            "port": "10053",
            "lastaccess": "1635332902",
            "status": "2"
        }
    ],
    "id": 1
}
```

Pobierz listę zatrzymanych węzłów

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["ha_nodeid", "address", "port"],
        "filter": {
            "status": 1
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "ha_nodeid": "ckuo7i1nw000g0sajjsjre7e3",
            "address": "192.168.1.12",
            "port": "10051"
        },
        {
            "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
            "address": "192.168.1.8",
            "port": "10052"
        },
        {
            "ha_nodeid": "ckuo7i1nv000d0sajd95y1b6x",
            "address": "192.168.1.9",
            "port": "10053"
        }
    ],
    "id": 1
}
```

Pobierz liczbę węzłów zapasowych

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
```

```
    "params": {
        "countOutput": true,
        "filter": {
            "status": 0
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "3",
    "id": 1
}
```

Sprawdzenie statusu węzłów pod określonymi adresami IP

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["name", "status"],
        "filter": {
            "address": ["192.168.1.7", "192.168.1.13"]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "node3",
            "status": "0"
        },
        {
            "name": "node-active",
            "status": "3"
        }
    ],
    "id": 1
}
```

Źródło

CHaNode::get() w *ui/include/classes/api/services/CHaNode.php*.

Zadanie

Ta klasa jest przeznaczona do pracy z zadaniami (takimi jak sprawdzanie pozycji lub reguł wykrywania niskiego poziomu bez przeładowania konfiguracji).

Odwołania do obiektów:

- Task
  - Obiekt żądania „Wykonaj teraz”
  - Obiekt żądania „Odśwież konfigurację proxy”
  - Obiekt żądania „Informacje diagnostyczne”
    * Obiekt żądania statystyk
    * Obiekt wyniku statystyk

Dostępne metody:

- task.create - utwórz nowe zadania
- task.get - pobierz zadania

Obiekt task

Poniższe obiekty są bezpośrednio powiązane z API `task`.

Task

Obiekt task ma następujące właściwości:

Property	Type	Description
taskid	ID	ID task. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Typ task. Możliwe wartości: 1 - Informacje diagnostyczne; 2 - Odśwież konfigurację proxy; 6 - Wykonaj teraz. Od Zabbix 7.4.3 użytkownicy typu *Admin* i *User* mogą tworzyć taski „Wykonaj teraz". Należy pamiętać, że `task.get` zawsze zwraca "7" (podsumowanie wykonania task). Zachowanie właściwości: - *wymagane*
status	integer	Status task. Możliwe wartości: 1 - nowy task; 2 - task w trakcie realizacji; 3 - task został zakończony; 4 - task wygasł. Zachowanie właściwości: - *tylko do odczytu*
clock	timestamp	Czas utworzenia task. Zachowanie właściwości: - *tylko do odczytu*
ttl	integer	Czas w sekundach, po którym task wygasa. Zachowanie właściwości: - *tylko do odczytu*
proxyid	ID	ID proxy, o którym zbierane są statystyki informacji diagnostycznych. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na "Informacje diagnostyczne" lub "Odśwież konfigurację proxy"

Property	Type	Description
request	object	Obiekt żądania task zgodnie z typem task: Obiekt task „Wykonaj teraz” jest opisany szczegółowo poniżej; Obiekt task „Odśwież konfigurację proxy” jest opisany szczegółowo poniżej; Obiekt task „Informacje diagnostyczne” jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*
result	object	Obiekt wyniku task informacji diagnostycznych. Może zawierać NULL, jeśli wynik nie jest jeszcze gotowy. Obiekt wyniku jest opisany szczegółowo poniżej. Zachowanie właściwości: - *tylko do odczytu*

Obiekt żądania „Execute now”

Obiekt żądania zadania „Execute now” ma następujące właściwości.

Property	Type	Description
itemid	ID	ID pozycji i reguł niskopoziomowego wykrywania. Od Zabbix 7.4.3 użytkownicy typu *Admin* i *User* mogą wykonywać „Execute now” dla pozycji na hostach, dla których mają uprawnienie *read-write* permission albo uprawnienie *read* oraz włączoną dla swojej roli akcję `invoke_execute_now` action. To samo dotyczy użytkowników typu *Admin* w przypadku reguł niskopoziomowego wykrywania (LLD).

Obiekt żądania „Refresh proxy configuration”

Obiekt żądania zadania „Refresh proxy configuration” ma następujące właściwości.

Właściwość	Type	Opis
proxyids	array	ID proxy.

Obiekt żądania „Diagnostic information”

Obiekt żądania zadania informacji diagnostycznych ma następujące właściwości. Obiekt żądania statystyk dla wszystkich typów właściwości jest opisany szczegółowo poniżej.

Właściwość	Typ	Opis
historycache	object	Żądanie statystyk pamięci podręcznej historii. Dostępne na serwer i proxy.
valuecache	object	Żądanie statystyk pamięci podręcznej pozycji. Dostępne na serwer.
preprocessing	object	Żądanie statystyk menedżera przetwarzania wstępnego. Dostępne na serwer i proxy.
alerting	object	Żądanie statystyk menedżera alertów. Dostępne na serwer.
lld	object	Żądanie statystyk menedżera LLD. Dostępne na serwer.

Obiekt żądania statystyk

Obiekt żądania statystyk służy do określenia, jaki typ informacji powinien być zbierany o wewnętrznych procesach serwera/proxy. Ma on następujące właściwości.

Właściwość	Typ	Opis
stats	query	Właściwości obiektu statystyk, które mają zostać zwrócone. Lista dostępnych pól dla każdego typu statystyk informacji diagnostycznych jest opisana szczegółowo poniżej. Domyślnie: `extend` zwróci wszystkie dostępne pola statystyk.
top	object	Obiekt służący do sortowania i ograniczania zwracanych wartości statystyk. Lista dostępnych pól dla każdego typu statystyk informacji diagnostycznych jest opisana szczegółowo poniżej. Przykład: { “source.alerts”: 10 }

Lista pól statystyk dostępnych dla każdego typu żądania informacji diagnostycznych

Dla każdej właściwości typu żądania informacji diagnostycznych można żądać następujących pól statystyk.

Typ diagnostyczny	Dostępne pola	Opis
historycache	items	Liczba elementów w pamięci podręcznej.
	values	Liczba wartości w pamięci podręcznej.
	memory	Statystyki pamięci współdzielonej (wolne miejsce, liczba używanych fragmentów, liczba wolnych fragmentów, maksymalny rozmiar wolnego fragmentu).
	memory.data	Statystyki pamięci współdzielonej pamięci podręcznej danych historii.
	memory.index	Statystyki pamięci współdzielonej pamięci podręcznej indeksu historii.
valuecache	items	Liczba elementów w pamięci podręcznej.
	values	Liczba wartości w pamięci podręcznej.
	memory	Statystyki pamięci współdzielonej (wolne miejsce, liczba używanych fragmentów, liczba wolnych fragmentów, maksymalny rozmiar wolnego fragmentu).
	mode	Tryb pamięci podręcznej wartości.
preprocessing	values	Liczba wartości w kolejce.
	preproc.values	Liczba wartości w kolejce z krokami przetwarzania wstępnego.
alerting	alerts	Liczba alertów w kolejce.
lld	rules	Liczba reguł w kolejce.
	values	Liczba wartości w kolejce.

Lista pól sortowania dostępnych dla każdego typu żądania informacji diagnostycznych

Następujące pola statystyczne mogą być używane do sortowania i ograniczania żądanych informacji.

Typ diagnostyczny	Dostępne pola	Typ
historycache	values	integer
valuecache	values	integer
	request.values	integer
preprocessing	values	integer
alerting	media.alerts	integer
	source.alerts	integer
lld	values	integer

Obiekt wyniku statystyk

Obiekt wyniku statystyk jest pobierany w polu `result` obiektu zadania.

Właściwość	Typ	Opis
status	integer	Status wyniku zadania. Możliwe wartości: -1 - podczas wykonywania zadania wystąpił błąd; 0 - wynik zadania został utworzony. Zachowanie właściwości: *- tylko do odczytu*
data	string/object	Wyniki zgodne z obiektem żądania statystyk dla określonego zadania informacji diagnostycznych. Zawiera ciąg komunikatu o błędzie, jeśli podczas wykonywania zadania wystąpił błąd.

task.create

Opis

`object task.create(object/array tasks)`

Ta metoda umożliwia tworzenie zadań.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu (dla użytkowników typu *Admin* i *User* od Zabbix 7.4.3). Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Zadanie do utworzenia.

Metoda akceptuje zadania ze standardowymi właściwościami zadania.

Należy pamiętać, że zadania „Wykonaj teraz” można tworzyć tylko dla następujących typów pozycji/reguł wykrywania:

- agent Zabbix (pasywny)
- Proste sprawdzenie
- agent SNMP (v1/v2/v3)
- Wewnętrzny Zabbix
- Sprawdzenie zewnętrzne
- Monitor bazy danych
- agent HTTP
- agent IPMI
- agent SSH
- agent TELNET
- agent JMX
- Obliczana
- Zależna pozycja
- Skrypt
- Przeglądarka

Jeśli pozycja/reguła wykrywania jest typu „Zależna pozycja”, to jej pozycja nadrzędna również musi być jednym z powyższych typów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych zadań we właściwości `taskids`. Jedno zadanie jest tworzone dla każdej pozycji i reguły wykrywania niskiego poziomu. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych `itemids`.

Przykłady

Tworzenie zadania

Utwórz zadanie „Wykonaj teraz” dla pozycji i reguły wykrywania niskiego poziomu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 6,
            "request": {
                "itemid": "10092"
            }
        },
        {
            "type": 6,
            "request": {
                "itemid": "10093"
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Utwórz zadanie „Odśwież konfigurację proxy” dla dwóch proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 2,
            "request": {
                "proxyids": ["10459", "10460"]
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1"
        ]
    },
    "id": 1
}
```

Utwórz zadanie „Informacje diagnostyczne”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 1,
            "request": {
                "alerting": {
                    "stats": [
                        "alerts"
                    ],
                    "top": {
                        "media.alerts": 10
                    }
                },
                "lld": {
                    "stats": "extend",
                    "top": {
                        "values": 5
                    }
                }
            },
            "proxyid": 0
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Task
- Obiekt żądania „Wykonaj teraz”
- Obiekt żądania „Odśwież konfigurację proxy”
- Obiekt żądania „Informacje diagnostyczne”

Źródło

CTask::create() w *ui/include/classes/api/services/CTask.php*.

task.get

Opis

`integer/array task.get(object parameters)`

Ta metoda umożliwia pobieranie zadań zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna tylko dla użytkownika typu *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
taskids	ID/array	Zwraca tylko zadania o podanych identyfikatorach.
output	query	Te parametry są opisane w komentarzu referencyjnym.
preservekeys	boolean	

Zwrócone wartości

`(integer/array)` Zwraca tablicę obiektów.

Przykłady

Pobierz zadanie według ID

Pobierz wszystkie dane dotyczące zadania o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.get",
    "params": {
        "output": "extend",
        "taskids": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "taskid": "1",
            "type": "7",
            "status": "3",
            "clock": "1601039076",
            "ttl": "3600",
            "proxyid": 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
}
```


Zobacz także

- Task
- Obiekt żądania „Wykonaj teraz”
- Obiekt żądania „Odśwież konfigurację proxy”
- Obiekt żądania „Informacje diagnostyczne”

Źródło

CTask::get() w *ui/include/classes/api/services/CTask.php*.

Zdarzenie

Ta klasa jest przeznaczona do pracy ze zdarzeniami.

Odwołania do obiektów:

- Event
- Tag zdarzenia
- URL typu nośnika

Dostępne metody:

- event.get - pobierz zdarzenia
- event.acknowledge - potwierdź zdarzenia

Obiekt zdarzenia

Następujące obiekty są bezpośrednio powiązane z API `event`.

Zdarzenie

> **Note:**
> Zdarzenia są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt zdarzenia ma następujące właściwości.

Właściwość	Typ	Opis
eventid	ID	ID zdarzenia.
source	integer	Typ zdarzenia. Możliwe wartości: 0 - zdarzenie utworzone przez wyzwalacz; 1 - zdarzenie utworzone przez regułę wykrywania; 2 - zdarzenie utworzone przez aktywną autorejestrację agenta; 3 - zdarzenie wewnętrzne; 4 - zdarzenie utworzone przy aktualizacji stanu usługi.

Właściwość	Typ	Opis
object	integer	Typ obiektu powiązanego ze zdarzeniem. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”: 0 - wyzwalacz. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez regułę wykrywania”: 1 - wykryty host; 2 - wykryta usługa. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez aktywną autorejestrację agenta”: 3 - host z autorejestracji. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - wyzwalacz; 4 - pozycja; 5 - reguła LLD. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przy aktualizacji stanu usługi”: 6 - usługa.
objectid	ID	ID powiązanego obiektu.
acknowledged	integer	Czy zdarzenie zostało potwierdzone.
clock	timestamp	Czas utworzenia zdarzenia.
ns	integer	Nanosekundy w momencie utworzenia zdarzenia.
name	string	Rozwinięta nazwa zdarzenia.
value	integer	Stan powiązanego obiektu. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”: 0 - OK; 1 - problem. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez regułę wykrywania”: 0 - host lub usługa działa; 1 - host lub usługa nie działa; 2 - host lub usługa wykryte; 3 - host lub usługa utracone. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - stan „normalny”; 1 - stan „nieznany” lub „niewspierany”. Zachowanie właściwości: - *obsługiwane* jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie utworzone przez regułę wykrywania”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”

Właściwość	Typ	Opis
severity	integer	Bieżący poziom ważności zdarzenia. Możliwe wartości: 0 - niesklasyfikowane; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
r_eventid	ID	ID zdarzenia odzyskania.
c_eventid	ID	ID zdarzenia użytego do nadpisania (zamknięcia) bieżącego zdarzenia zgodnie z globalną regułą korelacji. Zobacz `correlationid`, aby zidentyfikować dokładną regułę korelacji. Ten parametr jest zdefiniowany tylko wtedy, gdy zdarzenie zostało zamknięte przez globalną regułę korelacji.
cause_eventid	ID	ID zdarzenia przyczyny.
correlationid	ID	ID reguły korelacji, która spowodowała zamknięcie problemu. Ten parametr jest zdefiniowany tylko wtedy, gdy zdarzenie zostało zamknięte przez globalną regułę korelacji.
userid	ID	ID użytkownika, który zamknął zdarzenie (jeśli zdarzenie zostało zamknięte ręcznie).
suppressed	integer	Czy zdarzenie jest wyciszone. Możliwe wartości: 0 - zdarzenie jest w stanie normalnym; 1 - zdarzenie jest wyciszone.
opdata	string	Dane operacyjne z rozwiniętymi makrami.
urls	array	Aktywne adresy URL typu nośnika.

Znacznik zdarzenia

Obiekt znacznika zdarzenia ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa znacznika zdarzenia.
value	string	Wartość znacznika zdarzenia.

URL typu nośnika

Obiekt URL typu nośnika ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa URL zdefiniowanego dla typu nośnika.
url	string	Wartość URL zdefiniowana dla typu nośnika.

Wyniki będą zawierać wpisy tylko dla aktywnych typów nośników z włączonym wpisem menu zdarzenia. Makra użyte we właściwościach zostaną rozwinięte, ale jeśli jedna z właściwości zawiera nierozwinięte makro, obie właściwości zostaną wykluczone z wyników. Obsługiwane makra opisano w sekcji *Obsługiwane makra*.

event.acknowledge

Opis

`object event.acknowledge(object/array parameters)`

Ta metoda umożliwia aktualizację zdarzeń. Można wykonać następujące akcje aktualizacji:

- Zamknij zdarzenie. Jeśli zdarzenie jest już rozwiązane, ta akcja zostanie pominięta.
- Potwierdź zdarzenie. Jeśli zdarzenie jest już potwierdzone, ta akcja zostanie pominięta.
- Cofnij potwierdzenie zdarzenia. Jeśli zdarzenie nie jest potwierdzone, ta akcja zostanie pominięta.

- Dodaj wiadomość.
- Zmień poziom ważności zdarzenia. Jeśli zdarzenie ma już ten sam poziom ważności, ta akcja zostanie pominięta.
- Wycisz zdarzenie. Jeśli zdarzenie jest już wyciszone, ta akcja zostanie pominięta.
- Przywróć wyciszenie zdarzenia. Jeśli zdarzenie nie jest wyciszone, ta akcja zostanie pominięta.
- Zmień rangę zdarzenia na przyczynę. Jeśli zdarzenie jest już oznaczone jako przyczyna, ta akcja zostanie pominięta.
- Zmień rangę zdarzenia na objaw. Jeśli zdarzenie jest już oznaczone jako objaw, ta akcja zostanie pominięta.

Attention:
Można aktualizować tylko zdarzenia wyzwalaczy.
 Można aktualizować tylko zdarzenia problemów.
 Do zamknięcia zdarzenia lub zmiany jego poziomu ważności wymagane są uprawnienia odczytu/zapisu dla wyzwalacza.
 Aby zamknąć zdarzenie, w wyzwalaczu musi być dozwolone ręczne zamknięcie.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Parametry zawierające identyfikatory zdarzeń oraz operacji aktualizacji, które należy wykonać.

Parameter	Type	Description
eventids	ID/array	Identyfikatory zdarzeń, które mają zostać potwierdzone. Zachowanie parametru: - *required*
action	integer	Akcja(-e) aktualizacji zdarzenia. Możliwe wartości bitowe: 1 - zamknij problem; 2 - potwierdź zdarzenie; 4 - dodaj wiadomość; 8 - zmień powagę; 16 - cofnij potwierdzenie zdarzenia; 32 - wycisz zdarzenie; 64 - wyłącz wyciszenie zdarzenia; 128 - zmień rangę zdarzenia na przyczynę; 256 - zmień rangę zdarzenia na objaw. To pole jest polem bitmaski; dopuszczalna jest dowolna suma możliwych wartości bitowych (na przykład 34 dla potwierdzenia i wyciszenia zdarzenia). Zachowanie parametru: - *required*
cause_eventid	ID	ID zdarzenia przyczyny. Zachowanie parametru: - *required* jeśli `action` zawiera bit ”change event rank to symptom”
message	string	Tekst wiadomości. Zachowanie parametru: - *required* jeśli `action` zawiera bit ”add message”

Parameter	Type	Description
severity	integer	Nowa powaga zdarzeń. Możliwe wartości: 0 - nie sklasyfikowano; 1 - informacja; 2 - ostrzeżenie; 3 - przeciętne; 4 - wysokie; 5 - katastrofa. Zachowanie parametru: - *required* jeśli `action` zawiera bit ”change severity”
suppress_until	integer	Znacznik czasu Unix, do którego zdarzenie musi być wyciszone. Jeśli ustawione na ”0”, wyciszenie będzie bezterminowe. Zachowanie parametru: - *required* jeśli `action` zawiera bit ”suppress event”

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych zdarzeń w właściwości `eventids`.

Przykłady

Potwierdzanie zdarzenia

Potwierdź pojedyncze zdarzenie i pozostaw wiadomość.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 6,
        "message": "Problem rozwiązany."
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427"
        ]
    },
    "id": 1
}
```

Zmiana ważności zdarzenia

Zmień ważność wielu zdarzeń i pozostaw wiadomość.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": ["20427", "20428"],
        "action": 12,
```

```
        "message": "Maintenance required to fix it.",
        "severity": 4
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427",
            "20428"
        ]
    },
    "id": 1
}
```

Źródło

CEvent::acknowledge() w *ui/include/classes/api/services/CEvent.php*.

event.get

Opis

`integer/array event.get(object parameters)`

Ta metoda umożliwia pobieranie zdarzeń zgodnie z podanymi parametrami.

Attention:
Ta metoda może zwracać zdarzenia usuniętej encji, jeśli zdarzenia te nie zostały jeszcze usunięte przez housekeeper.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
eventids	ID/array	Zwracaj tylko zdarzenia o podanych ID.
groupids	ID/array	Zwracaj tylko zdarzenia utworzone przez obiekty należące do podanych grup hostów.
hostids	ID/array	Zwracaj tylko zdarzenia utworzone przez obiekty należące do podanych hostów.
objectids	ID/array	Zwracaj tylko zdarzenia utworzone przez podane obiekty.
source	integer	Zwracaj tylko zdarzenia o podanym typie. Listę obsługiwanych typów zdarzeń można znaleźć na stronie obiektu event. Domyślnie: 0 - zdarzenia wyzwalaczy.
object	integer	Zwracaj tylko zdarzenia utworzone przez obiekty podanego typu. Listę obsługiwanych typów obiektów można znaleźć na stronie obiektu event. Domyślnie: 0 - wyzwalacz.
acknowledged	boolean	Jeśli ustawiono `true`, zwracaj tylko potwierdzone zdarzenia.

Parametr	Typ	Opis
action	integer	Zwracaj tylko zdarzenia, dla których wykonano podane akcje aktualizacji zdarzenia. W przypadku wielu akcji użyj sumy dowolnych akceptowalnych wartości bitmapy jako maski bitowej (na przykład 34 dla potwierdzenia i wyciszenia zdarzenia).
action_userids	ID/array	Zwracaj tylko zdarzenia z podanymi ID użytkowników, którzy wykonali akcje aktualizacji zdarzenia.
suppressed	boolean	Jeśli ustawiono `true`, zwracaj tylko wyciszone zdarzenia.
symptom	boolean	Jeśli ustawiono `true`, zwracaj tylko zdarzenia symptomów.
severities	integer/array	Zwracaj tylko zdarzenia o podanych poziomach ważności zdarzeń. Ma zastosowanie tylko wtedy, gdy object to wyzwalacz.
trigger_severities	integer/array	Zwracaj tylko zdarzenia o podanych poziomach ważności wyzwalaczy. Ma zastosowanie tylko wtedy, gdy object to wyzwalacz.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwracaj tylko zdarzenia z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie zdarzenia. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
eventid_from	string	Zwracaj tylko zdarzenia o ID większym lub równym podanemu ID.
eventid_till	string	Zwracaj tylko zdarzenia o ID mniejszym lub równym podanemu ID.
time_from	timestamp	Zwracaj tylko zdarzenia utworzone o podanym czasie lub później.
time_till	timestamp	Zwracaj tylko zdarzenia utworzone o podanym czasie lub wcześniej.
problem_time_from	timestamp	Zwraca tylko zdarzenia, które były w stanie problemu od `problem_time_from`, niezależnie od ich bieżącego statusu. Ma zastosowanie tylko wtedy, gdy source jest zdarzeniem wyzwalacza, a object jest wyzwalaczem. Ten parametr jest wymagany, jeśli określono `problem_time_till`.
problem_time_till	timestamp	Zwraca tylko zdarzenia, które były w stanie problemu do `problem_time_till`, niezależnie od ich bieżącego statusu. Ma zastosowanie tylko wtedy, gdy source jest zdarzeniem wyzwalacza, a object jest wyzwalaczem. Ten parametr jest wymagany, jeśli określono `problem_time_from`.
value	integer/array	Zwracaj tylko zdarzenia o podanych wartościach.

Parametr	Typ	Opis
selectAcknowledges	query	Zwracaj właściwość `acknowledges` z aktualizacjami zdarzenia. Aktualizacje zdarzenia są sortowane w odwrotnej kolejności chronologicznej. Obiekt aktualizacji zdarzenia ma następujące właściwości: `acknowledgeid` - `(ID)` ID potwierdzenia; `userid` - `(ID)` ID użytkownika, który zaktualizował zdarzenie; `clock` - `(timestamp)` czas aktualizacji zdarzenia; `message` - `(string)` treść wiadomości; `action` - `(integer)` wykonana akcja aktualizacji, zobacz `event.acknowledge`; `old_severity` - `(integer)` ważność zdarzenia przed tą akcją aktualizacji; `new_severity` - `(integer)` ważność zdarzenia po tej akcji aktualizacji; `suppress_until` - `(timestamp)` czas, do którego zdarzenie będzie wyciszone; `taskid` - `(ID)` ID zadania, jeśli bieżące zdarzenie przechodzi zmianę rangi; `username` - `(string)` nazwa użytkownika, który zaktualizował zdarzenie; `name` - `(string)` imię użytkownika, który zaktualizował zdarzenie; `surname` - `(string)` nazwisko użytkownika, który zaktualizował zdarzenie. Obsługuje `count`.
selectAlerts	query	Zwracaj właściwość `alerts` z alertami wygenerowanymi przez zdarzenie. Alerty są sortowane w odwrotnej kolejności chronologicznej.
selectHosts	query	Zwracaj właściwość `hosts` z hostami zawierającymi obiekt, który utworzył zdarzenie. Obsługiwane tylko dla zdarzeń wygenerowanych przez wyzwalacze, pozycje lub reguły LLD.
selectRelatedObject	query	Zwracaj właściwość `relatedObject` z obiektem, który utworzył zdarzenie. Typ zwracanego obiektu zależy od typu zdarzenia.
selectSuppressionData	query	Zwracaj właściwość `suppression_data` z listą aktywnych okresów utrzymaniowych i ręcznych wyciszeń: `maintenanceid` - `(ID)` ID okresu utrzymaniowego; `userid` - `(ID)` ID użytkownika, który wyciszył zdarzenie; `suppress_until` - `(integer)` czas, do którego zdarzenie jest wyciszone.
selectTags	query	Zwracaj właściwość `tags` z tagami zdarzenia.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartości są pojedynczą wartością lub tablicą wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `eventid`, `objectid`, `clock`. Możliwe wartości przy użyciu razem z `groupBy`: `objectid`. Możliwe wartości przy użyciu razem z `countOutput` i `groupBy`: `objectid`, `rowscount`.
groupBy	string/array	Grupuj wyniki według podanych właściwości. Określone właściwości zostaną zwrócone w wynikach. Możliwe wartości: `objectid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	

Parametr	Typ	Opis
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`, ale nie użyto parametru `groupBy`
- tablicę obiektów z wynikami agregacji, jeśli użyto parametru `groupBy`

Przykłady

Pobieranie zdarzeń wyzwalacza

Pobierz najnowsze zdarzenia z wyzwalacza „22395”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "selectAcknowledges": "extend",
        "selectSuppressionData": "extend",
        "selectTags": "extend",
        "objectids": "22395",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20",
            "source": "0",
            "object": "0",
            "objectid": "22395",
            "clock": "1728658089",
            "value": "0",
            "acknowledged": "0",
            "ns": "461809482",
            "name": "Średnie obciążenie jest zbyt wysokie (obciążenie na CPU powyżej 1.5 przez 5 min)",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "acknowledges": [],
            "opdata": "Średnie obciążenia(1m 5m 15m): (1.6328 3.0522 2.2515), liczba CPU: 2",
            "suppression_data": [],
```

```
            "suppressed": "0",
            "tags": [
                {
                    "tag": "scope",
                    "value": "capacity"
                },
                {
                    "tag": "scope",
                    "value": "performance"
                },
                {
                    "tag": "component",
                    "value": "cpu"
                },
                {
                    "tag": "class",
                    "value": "os"
                },
                {
                    "tag": "target",
                    "value": "linux"
                }
            ],
            "urls": []
        },
        {
            "eventid": "4",
            "source": "0",
            "object": "0",
            "objectid": "22395",
            "clock": "1728657737",
            "value": "1",
            "acknowledged": "1",
            "ns": "460759366",
            "name": "Średnie obciążenie jest zbyt wysokie (obciążenie na CPU powyżej 1.5 przez 5 min)",
            "severity": "3",
            "r_eventid": "20",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "acknowledges": [
                {
                    "acknowledgeid": "1",
                    "userid": "1",
                    "clock": "1728657938",
                    "message": "Środowisko testowe. Proszę zignorować ten alert.",
                    "action": "38",
                    "old_severity": "0",
                    "new_severity": "0",
                    "suppress_until": "1728744338",
                    "taskid": "0",
                    "username": "Admin",
                    "name": "Zabbix",
                    "surname": "Administrator"
                }
            ],
            "opdata": "Średnie obciążenia(1m 5m 15m): (1.6328 3.0522 2.2515), liczba CPU: 2",
            "suppression_data": [
                {
                    "maintenanceid": "0",
                    "suppress_until": "1728744338",
```

```
                    "userid": "1"
                }
            ],
            "suppressed": "1",
            "tags": [
                {
                    "tag": "scope",
                    "value": "capacity"
                },
                {
                    "tag": "scope",
                    "value": "performance"
                },
                {
                    "tag": "component",
                    "value": "cpu"
                },
                {
                    "tag": "class",
                    "value": "os"
                },
                {
                    "tag": "target",
                    "value": "linux"
                }
            ],
            "urls": []
        }
    ],
    "id": 1
}
```

Pobieranie zdarzeń według przedziału czasu

Pobierz wszystkie zdarzenia utworzone między 17 a 18 października 2012 roku, w odwrotnej kolejności chronologicznej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "time_from": "1350432000",
        "time_till": "1350518400",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20617",
            "source": "0",
            "object": "0",
            "objectid": "14282",
            "clock": "1350477816",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
```

```
            "name": "Mniej niż 25% wolnego miejsca w pamięci podręcznej historii",
            "severity": "3",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20616",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477814",
            "value": "0",
            "acknowledged": "0",
            "ns": "0",
            "name": "Procesy Zabbix trapper są zajęte w ponad 75%",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20615",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477541",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
            "name": "Procesy Zabbix trapper są zajęte w ponad 75%",
            "severity": "3",
            "r_eventid": "20616",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```


Pobieranie zdarzeń potwierdzonych przez określonego użytkownika

Pobieranie zdarzeń potwierdzonych przez użytkownika o ID=10

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "event.get",
```

```
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "503",
            "source": "0",
            "object": "0",
            "objectid": "23162",
            "clock": "1747212236",
            "value": "1",
            "acknowledged": "1",
            "ns": "413470863",
            "name": "Number of installed packages has been changed",
            "severity": "2",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "acknowledges": [
                {
                    "userid": "10",
                    "action": "2"
                }
            ],
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```

Pobieranie najważniejszych wyzwalaczy według liczby zdarzeń problemowych

Pobierz 5 najważniejszych wyzwalaczy o poziomach ważności „Warning”, „Average”, „High” lub „Disaster” wraz z liczbą zdarzeń problemowych w określonym przedziale czasu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "countOutput": true,
        "groupBy": "objectid",
        "source": 0,
        "object": 0,
        "value": 1,
        "time_from": 1672531200,
        "time_till": 1677628800,
        "trigger_severities": [2, 3, 4, 5],
```

```
        "sortfield": ["rowscount"],
        "sortorder": "DESC",
        "limit": 5
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "objectid": "232124",
            "rowscount": "27"
        },
        {
            "objectid": "29055",
            "rowscount": "23"
        },
        {
            "objectid": "253731",
            "rowscount": "18"
        },
        {
            "objectid": "254062",
            "rowscount": "11"
        },
        {
            "objectid": "23216",
            "rowscount": "7"
        }
    ],
    "id": 1
}
```

Zobacz także

- Alert
- Pozycja
- Host
- Reguła LLD
- Usługa
- Wyzwalacz

Źródło

CEvent::get() w *ui/include/classes/api/services/CEvent.php*.

Dodatek 1. Komentarz referencyjny

Notacja Typy danych

API Zabbix obsługuje następujące typy danych jako dane wejściowe:

Type	Description
ID	Unikalny identyfikator używany do odwoływania się do encji.
boolean	Wartość logiczna (`true` lub `false`).
flag	Wartość uznawana za `true`, jeśli została przekazana i nie jest równa `null`; w przeciwnym razie wartość jest uznawana za `false`.
integer	Liczba całkowita.
float	Liczba zmiennoprzecinkowa.

Type	Description
string	Ciąg tekstowy.
text	Dłuższy ciąg tekstowy.
timestamp	Znacznik czasu Unix.
array	Uporządkowana sekwencja wartości (zwykła tablica).
object	Tablica asocjacyjna.
query	Wartość definiująca dane, które mają zostać zwrócone. Wartość może być zdefiniowana jako tablica nazw właściwości (aby zwrócić tylko określone właściwości) lub jako jedna ze wstępnie zdefiniowanych wartości: `extend` - zwraca wszystkie właściwości obiektu; `count` - zwraca liczbę pobranych rekordów, obsługiwane tylko przez niektóre podwybory.

Attention:
API Zabbix zawsze zwraca wartości wyłącznie jako ciągi znaków lub tablice.

Zachowanie właściwości

Niektóre właściwości obiektu są oznaczone krótkimi etykietami opisującymi ich zachowanie. Używane są następujące etykiety:

- ***tylko do odczytu*** - wartość właściwości jest ustawiana automatycznie i nie może być definiowana ani zmieniana przez użytkownika, nawet w niektórych szczególnych warunkach (np. *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych);
- ***tylko do zapisu*** - wartość właściwości może zostać ustawiona, ale później nie można uzyskać do niej dostępu;
- ***stała*** - wartość właściwości może zostać ustawiona podczas tworzenia obiektu, ale później nie może być zmieniona;
- ***obsługiwana*** - ustawienie wartości właściwości nie jest wymagane, ale jest dozwolone w niektórych szczególnych warunkach (np. *obsługiwana*, jeśli `type` jest ustawione na ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent”); należy jednak pamiętać, że właściwości oznaczone jako *obsługiwana* mogą nadal być ustawiane na swoje wartości domyślne niezależnie od warunków;
- ***wymagana*** - ustawienie wartości właściwości jest wymagane dla wszystkich operacji (z wyjątkiem operacji pobierania) lub w niektórych szczególnych warunkach (np. *wymagana* dla operacji tworzenia; *wymagana*, jeśli `operationtype` jest ustawione na ”global script”, a `opcommand_hst` nie jest ustawione).

Note:
W przypadku operacji aktualizacji właściwość jest uznawana za „ustawioną”, gdy zostanie ustawiona podczas operacji aktualizacji.

Właściwości, które nie są oznaczone etykietami, są opcjonalne.

Zachowanie parametrów

Niektóre parametry operacji są oznaczone krótkimi etykietami opisującymi ich zachowanie dla danej operacji. Używane są następujące etykiety:

- ***tylko do odczytu*** - wartość parametru jest ustawiana automatycznie i nie może być definiowana ani zmieniana przez użytkownika, nawet w niektórych szczególnych warunkach (np. *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych);
- ***tylko do zapisu*** - wartość parametru może zostać ustawiona, ale później nie można uzyskać do niej dostępu;
- ***obsługiwany*** - ustawienie wartości parametru nie jest wymagane, ale jest dozwolone w niektórych szczególnych warunkach (np. *obsługiwany*, jeśli `operating_mode` obiektu Proxy jest ustawiony na ”passive proxy”); należy jednak pamiętać, że parametry oznaczone jako *obsługiwany* nadal mogą być ustawiane na wartości domyślne niezależnie od warunków;
- ***wymagany*** - wartość parametru musi zostać ustawiona.

Parametry, które nie są oznaczone etykietami, są opcjonalne.

Zastrzeżona wartość ID ”0” Zastrzeżona wartość ID ”0” może być używana do filtrowania elementów i usuwania obiektów , do których istnieją odniesienia. Na przykład, aby usunąć odwołanie do proxy z hosta , proxyid powinno być ustawione na 0 (”proxyid”: ”0”) lub aby filtrować hosty monitorowane przez opcję serwera proxyids powinno być ustawione na 0 (”proxyids”: ”0”).

Wspólne parametry metod ”get” Następujące parametry są obsługiwane przez wszystkie metody `get`:

Parametr	Typ	Opis
countOutput	boolean	Zwraca liczbę rekordów w wyniku zamiast rzeczywistych danych.

Parametr	Typ	Opis
editable	boolean	Jeśli ustawiono na `true`, zwraca tylko obiekty, do których użytkownik ma uprawnienia zapisu. Domyślnie: `false`.
excludeSearch	boolean	Zwraca wyniki, które nie pasują do kryteriów podanych w parametrze `search`.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości (np. właściwości obiektu Host w `host.get`, właściwości obiektu Item w `item.get` itd.), a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Należy pamiętać, że niektóre metody mają specyficzną funkcjonalność dla tego parametru, opisaną na stronie danej metody (np. parametr `filter` w host.get obsługuje również właściwości interfejsu hosta).
limit	integer	Ogranicza liczbę zwracanych rekordów.
output	query	Właściwości obiektu, które mają zostać zwrócone. Należy pamiętać, że identyfikator obiektu (tj. `hostid`, `itemid` itd.) jest zawsze uwzględniany w odpowiedzi, nawet jeśli nie został określony w parametrze `output`. Domyślnie: `extend`.
preservekeys	boolean	Używa identyfikatorów jako kluczy w wynikowej tablicy.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości (np. właściwości obiektu Host w `host.get`, właściwości obiektu Item w `item.get` itd.), a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Należy pamiętać, że niektóre metody mają specyficzną funkcjonalność dla tego parametru, opisaną na stronie danej metody (np. parametr `search` w host.get obsługuje również właściwości interfejsu hosta).
searchByAny	boolean	Jeśli ustawiono na `true`, zwraca wyniki pasujące do dowolnego z kryteriów podanych w parametrze `filter` lub `search`, zamiast do wszystkich. Domyślnie: `false`.
searchWildcardsEnabled	boolean	Jeśli ustawiono na `true`, włącza użycie znaku "*" jako znaku wieloznacznego w parametrze `search`. Domyślnie: `false`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Listę właściwości, które mogą być używane do sortowania, można znaleźć w opisie konkretnej metody API get. Makra nie są rozwijane przed sortowaniem. Jeśli nie podano wartości, dane zostaną zwrócone bez sortowania.
sortorder	string/array	Kolejność sortowania. Jeśli przekazano tablicę, każda wartość zostanie dopasowana do odpowiadającej jej właściwości podanej w parametrze `sortfield`. Możliwe wartości: `ASC` - *(domyślnie)* rosnąco; `DESC` - malejąco.
startSearch	boolean	Parametr `search` będzie porównywał początek pól, czyli zamiast tego wykona wyszukiwanie `LIKE "…%"`. Ignorowane, jeśli `searchWildcardsEnabled` jest ustawione na `true`.

Flagi pochodzenia encji Metody get zwracają właściwość `flags` dla encji powiązanych z wykrywaniem niskopoziomowym (reguła LLD/prototyp reguły LLD, pozycja/prototyp pozycji itd.). Ta właściwość jest przydatna do określenia, czy encja została

wykryta, czy nie, ponieważ możliwości edycji wykrytych encji są ograniczone.

Właściwość `flags` zwraca wynik oparty na kombinacji (operacja „+”) tych wartości:

Value	Description
0	Encja bazowa (pozycja, wyzwalacz, wykres, host)
1	Reguła wykrywania niskopoziomowego
2	Dowolny prototyp (prototyp pozycji, prototyp wyzwalacza, prototyp reguły LLD itd.)
4	Wykryta encja (wykryta pozycja, wyzwalacz, wykres, host, reguła LLD)

Łączna wartość zwracana przez właściwość `flags` może być następująca:

Value	Combination of	Description
0	0	Zwykła encja (pozycja, wyzwalacz, wykres, host).
2	2	Prototyp encji (prototyp pozycji, prototyp wyzwalacza itd.).
6	2+4	Wykryta pozycja, wyzwalacz, wykres, host (utworzone z prototypu).
1	1	Reguła wykrywania niskopoziomowego.
3	1+2	Prototyp reguły wykrywania niskopoziomowego.
5	1+4	Wykryta reguła wykrywania niskopoziomowego (utworzona z prototypu).
7	1+2+4	Wykryty prototyp reguły wykrywania niskopoziomowego.

Przykłady Sprawdzenie uprawnień użytkownika

Czy użytkownik ma uprawnienia do zapisu do hostów, których nazwy zaczynają się od ”MySQL” lub ”Linux” ?

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": ["MySQL", "Linux"]
        },
        "editable": true,
        "startSearch": true,
        "searchByAny": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "0",
    "id": 1
}
```

Note:
Wynik zero oznacza brak hostów z uprawnieniami do odczytu/zapisu.

Zliczanie niedopasowań

Policz liczbę hostów, których nazwy nie zawierają podciągu ”ubuntu”

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
```

```
        "search": {
            "host": "ubuntu"
        },
        "excludeSearch": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "44",
    "id": 1
}
```

Wyszukiwanie hostów przy użyciu symboli wieloznacznych

Znajdź hosty, których nazwa zawiera słowo „server” i które mają porty interfejsów „10050” lub „10071”. Posortuj wynik według nazwy hosta malejąco i ogranicz go do 5 hostów.

Żądanie:

```
{
    "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
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Wyszukiwanie hostów przy użyciu symboli wieloznacznych z parametrem ”preservekeys”

Jeśli dodasz parametr ”preservekeys” do poprzedniego żądania, wynik zostanie zwrócony jako tablica asocjacyjna, gdzie kluczami są identyfikatory obiektów.

Żądanie:

```
{
    "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
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Dodatek 2. Zmiany z 7.2 do 7.4

Zmiany niezgodne wstecznie mapa

ZBXNEXT-8784 Usunięto właściwości `linktriggerid` i `linkid` z obiektu wyzwalacz łącza mapy.

mediatype

ZBX-26258 `mediatype.get`: Dodano ścisłą walidację parametru `selectUsers`.

user

ZBX-26258 `user.get`: Dodano ścisłą walidację parametrów metody.
 ZBX-26258 `user.get`: Właściwość `userid` jest teraz zwracana tylko wtedy, gdy została jawnie zażądana lub gdy `output` jest ustawione na "extend".

usergroup

ZBX-26258 `usergroup.get`: Dodano ścisłą walidację parametrów `mfaids`, `selectUsers`.

Inne zmiany i poprawki błędów dashboard

ZBXNEXT-8919 Dodano nowe pole widżetu dashboardu `ds.0.color_palette` do widżetów `svggraph` i `piechart`.
 ZBXNEXT-9087 Dodano nowy typ widżetu dashboardu `itemcard`.

discoveryruleprototype

ZBXNEXT-1527 Dodano nowe API prototypu reguły LLD z metodami `discoveryruleprototype.create`, `discoveryruleprototype.upd` `discoveryruleprototype.get`, `discoveryruleprototype.delete`.

graph

ZBXNEXT-1527 Dodano nowy parametr `selectDiscoveryData` do metody `graph.get`; parametr `selectGraphDiscovery` został oznaczony jako przestarzały.

graphprototype

ZBXNEXT-1527 Dodano nowe parametry `selectDiscoveryRulePrototype` i `selectDiscoveryData` do metody `graphprototype.get`.

host

ZBXNEXT-1527 Dodano nowe parametry `selectDiscoveryData` i `selectDiscoveryRules` do metody `host.get`; parametry `selectHostDiscovery` i `selectDiscoveries` zostały oznaczone jako przestarzałe.

hostgroup

ZBXNEXT-1527 Dodano nowy parametr `selectDiscoveryData` do metody `hostgroup.get`; parametr `selectGroupDiscoveries` został oznaczony jako przestarzały.

hostprototype

ZBXNEXT-1527 Dodano nowe parametry `selectDiscoveryRulePrototype` i `selectDiscoveryData` do metody `hostprototype.get`.

pozycja

ZBXNEXT-1527 Dodano nowy parametr `selectDiscoveryData` do metody `item.get`; parametr `selectItemDiscovery` został oznaczony jako przestarzały.

itemprototype

ZBXNEXT-1527 Dodano nowe parametry `selectDiscoveryRulePrototype` i `selectDiscoveryData` do metody `itemprototype.get`.

mapa

ZBXNEXT-875 Dodano właściwość `zindex` do obiektu element mapy.
 ZBXNEXT-2122 Dodano właściwości `edit_own_media` i `edit_user_media` do obiektu action API roli.
 ZBXNEXT-8784 Dodano właściwości `background_scale`, `show_element_label`, `show_link_label` do obiektu mapa.
 ZBXNEXT-8784 Dodano właściwość `show_label` do obiektu element mapy.
 ZBXNEXT-8784 Dodano właściwości `show_label`, `indicator_type`, `itemid`, `thresholds`, `highlights` do obiektu łącze mapy.
 ZBXNEXT-8784 Dodano obiekt wskaźniki łącza mapy.

mediatype

ZBXNEXT-9805 Dodano opcję 2 ("OAuth token") do właściwości `smtp_authentication` obiektu mediatype.
 ZBXNEXT-9805 Dodano właściwości `redirection_url`, `client_id`, `client_secret`, `authorization_url`, `token_url`, `tokens_status`, `access_token`, `access_token_updated`, `access_expires_in`, `refresh_token` do obiektu mediatype.

szablon

ZBXNEXT-9872 Dodano nowe właściwości `readme` i `wizard_ready`.
 ZBXNEXT-1527 Dodano nowy parametr `selectDiscoveryRules` do metody `template.get`; parametr `selectDiscoveries` został oznaczony jako przestarzały.

pulpit szablonu

ZBXNEXT-9087 Dodano nowy typ widżetu pulpitu szablonu `itemcard`.

wyzwalacz

ZBXNEXT-1527 Dodano nowy parametr `selectDiscoveryData` do metody `trigger.get`; parametr `selectTriggerDiscovery` został oznaczony jako przestarzały.

triggerprototype

ZBXNEXT-1527 Dodano nowe parametry `selectDiscoveryRulePrototype` i `selectDiscoveryData` do metody `triggerprototype.ge`

usermacro

ZBXNEXT-9872 Dodano nową właściwość `config` do obiektu makra hosta.

Dodatek 3. Zmiany w wersji 7.4

7.4.3 dashboard

ZBXNEXT-10143 Dodano nowe pole widżetu dashboardu `highlight_row` w widżecie problems.

zadanie

ZBX-26202 `task.create`: użytkownicy typu *Admin* i *User* mogą teraz tworzyć zadania „Execute now".

21 Rozszerzenia

Przegląd Chociaż Zabbix oferuje wiele funkcji, zawsze istnieje miejsce na dodatkową funkcjonalność. Rozszerzenia są wygodnym sposobem modyfikowania i zwiększania możliwości monitorowania w Zabbix bez zmiany jego kodu źródłowego.

Możesz rozszerzyć funkcjonalność Zabbix, korzystając albo z wbudowanych opcji rozszerzeń (pozycje trapper, parametry użytkownika itp.), albo używając lub tworząc własne rozszerzenia (moduły ładowalne, wtyczki itp.).

Ta sekcja zawiera przegląd wraz z odwołaniami do wszystkich opcji rozszerzania Zabbix.

Zbieranie danych za pomocą niestandardowych poleceń Pozycje trapper

Pozycje trapper to pozycje, które przyjmują dane przychodzące zamiast je odpytywać. Pozycje trapper są przydatne do wysyłania określonych danych do serwera lub proxy Zabbix, na przykład okresowych danych o dostępności i wydajności w przypadku długotrwale działających skryptów użytkownika. Wysyłanie danych do serwera lub proxy Zabbix jest możliwe przy użyciu narzędzia Zabbix sender lub protokołu Zabbix sender. Wysyłanie danych do serwera Zabbix jest również możliwe przy użyciu metody API `history.push`.

Kontrole zewnętrzne

Kontrola zewnętrzna to pozycja służąca do wykonywania kontroli przez uruchomienie pliku wykonywalnego, na przykład skryptu powłoki lub pliku binarnego.

Kontrole zewnętrzne są wykonywane przez serwer Zabbix lub proxy (gdy host jest monitorowany przez proxy) i nie wymagają uruchomionego agenta na monitorowanym hoście.

Parametry użytkownika

Parametr użytkownika to polecenie zdefiniowane przez użytkownika (powiązane z kluczem zdefiniowanym przez użytkownika), które po wykonaniu może pobrać dane potrzebne z hosta, na którym działa agent Zabbix. Parametry użytkownika są przydatne do konfigurowania pozycji agenta lub agenta 2, które nie są predefiniowane w Zabbix.

Pozycje `system.run[]` Zabbix agent

`system.run[]` Zabbix agent item to pozycja dla polecenia zdefiniowanego przez użytkownika (powiązana ze wstępnie zdefiniowanym kluczem `system.run[]`, na przykład `system.run[myscript.sh]`), które może być wykonywane na hoście, na którym działa Zabbix agent.

Uwaga: pozycje `system.run[]` są domyślnie wyłączone i, jeśli są używane, muszą zostać włączone (dozwolone) oraz zdefiniowane w pliku konfiguracyjnym Zabbix agent lub agent 2 (parametr konfiguracyjny `AllowKey`).

> **Attention:**
> Polecenia zdefiniowane przez użytkownika w pozycjach, takich jak sprawdzenia zewnętrzne, parametry użytkownika i pozycje agenta Zabbix `system.run[]`, są wykonywane przez użytkownika systemu operacyjnego używanego do uruchamiania komponentów Zabbix. Aby wykonać te polecenia, ten użytkownik musi mieć niezbędne uprawnienia.

Pozycje HTTP agent

HTTP agent to pozycja służąca do wykonywania żądań danych przez HTTP/HTTPS. Pozycje HTTP agent są przydatne do wysyłania żądań do punktów końcowych HTTP w celu pobierania danych z usług takich jak *Elasticsearch* i *OpenWeatherMap*, sprawdzania stanu API Zabbix lub stanu serwera WWW Apache albo Nginx itp. Pozycje HTTP agent (z włączonym trapowaniem) mogą również działać jako pozycje trapper.

Pozycje skryptowe

Pozycja skryptowa to pozycja służąca do wykonywania zdefiniowanego przez użytkownika kodu JavaScript, który pobiera dane przez HTTP/HTTPS. Pozycje skryptowe są przydatne, gdy funkcjonalność zapewniana przez pozycje agenta HTTP nie jest wystarczająca. Na przykład w wymagających scenariuszach zbierania danych, które wymagają wielu kroków lub złożonej logiki, pozycję skryptową można skonfigurować tak, aby wykonała wywołanie HTTP, następnie przetworzyła otrzymane dane, a potem przekazała przekształconą wartość do drugiego wywołania HTTP.

> **Note:**
> Pozycje HTTP agent oraz pozycje skryptowe są obsługiwane przez serwer Zabbix i proxy i nie wymagają uruchomionego agenta na monitorowanym hoście.

Zaawansowane rozszerzenia Moduły ładowalne

Moduły ładowalne, napisane w C, są wszechstronną i wydajnościowo zoptymalizowaną opcją rozszerzania funkcjonalności komponentów Zabbix (serwer, proxy, agent) na platformach UNIX. Moduł ładowalny jest zasadniczo biblioteką współdzieloną używaną przez demon Zabbix i ładowaną podczas uruchamiania. Biblioteka powinna zawierać określone funkcje, aby proces Zabbix mógł wykryć, że plik jest rzeczywiście modułem, który może załadować i z którym może pracować.

Moduły ładowalne mają szereg zalet, w tym możliwość dodawania nowych metryk lub implementowania dowolnej innej logiki (na przykład eksportu danych historii Zabbix), wysoką wydajność oraz możliwość tworzenia, używania i udostępniania zapewnianej przez nie funkcjonalności. Przyczynia się to do bezproblemowego utrzymania i pomaga łatwiej dostarczać nową funkcjonalność niezależnie od bazowego kodu Zabbix.

Moduły ładowalne są szczególnie przydatne w złożonych środowiskach monitorowania. Podczas monitorowania systemów wbudowanych, przy dużej liczbie monitorowanych parametrów lub przy użyciu ciężkich skryptów ze złożoną logiką albo długim czasem uruchamiania, rozszerzenia takie jak parametry użytkownika, pozycje agenta Zabbix `system.run[]` oraz kontrole zewnętrzne będą miały wpływ na wydajność. Moduły ładowalne oferują sposób rozszerzania funkcjonalności Zabbix bez poświęcania wydajności.

Wtyczki

Wtyczki rozszerzają możliwości monitorowania agent Zabbix 2. Napisane w języku programowania Go stanowią alternatywę dla modułów ładowalnych (napisanych w C).

Wtyczka to pakiet Go, który definiuje strukturę i implementuje jeden lub kilka interfejsów wtyczek (*Exporter*, *Collector*, *Configurator*, *Runner*, *Watcher*). Obsługiwane są dwa typy wtyczek agent Zabbix 2:

- Wbudowane wtyczki (obsługiwane od Zabbix 4.4.0)
- Ładowalne wtyczki (obsługiwane od Zabbix 6.0.0)

Zobacz listę wtyczek dostarczanych przez Zabbix.

Instrukcje i samouczki dotyczące tworzenia własnych wtyczek znajdziesz w Developer center.

Dostosowywanie alertów Webhooki

Webhook to typ media type w Zabbix, który umożliwia rozszerzenie możliwości alertowania Zabbix na zewnętrzne oprogramowanie, takie jak systemy helpdesk, czaty lub komunikatory. Podobnie jak pozycje skryptowe, webhooki są przydatne do wykonywania wywołań HTTP przy użyciu niestandardowego kodu JavaScript, na przykład do wysyłania powiadomień do różnych platform, takich jak Microsoft Teams, Discord i Jira. Możliwe jest również zwracanie pewnych danych (na przykład dotyczących utworzonych zgłoszeń helpdesk), które są następnie wyświetlane w Zabbix.

Istniejące webhooki są dostępne w repozytorium Git Zabbix. Informacje dotyczące tworzenia własnych webhooków znajdziesz w Wytycznych dotyczących tworzenia webhooków.

Skrypty alertów

Skrypt alertu to typ mediów w Zabbix, który umożliwia utworzenie alternatywnego sposobu (skryptu) obsługi alertów Zabbix. Skrypty alertów są przydatne, jeśli istniejące typy mediów do wysyłania alertów w Zabbix nie spełniają Twoich oczekiwań.

Dostosowanie frontendu Własne motywy

Można zmienić wygląd wizualny frontendu Zabbix, używając własnych motywów. Zobacz instrukcje dotyczące tworzenia i stosowania własnych motywów.

Moduły frontend

Moduły frontend zapewniają możliwość rozszerzenia funkcjonalności frontend Zabbix przez dodawanie modułów innych firm lub tworzenie własnych. Za pomocą modułów frontend można dodawać nowe pozycje menu, odpowiadające im widoki, akcje itp.

Skrypty globalne Skrypt globalny to zdefiniowany przez użytkownika zestaw poleceń, które mogą być wykonywane na monitorowanym celu (przez interpreter powłoki (/bin/sh)), w zależności od skonfigurowanego zakresu i uprawnień użytkownika. Skrypty globalne można skonfigurować dla następujących działań:

- Operacja akcji
- Ręczne działanie na hoście
- Ręczne działanie na zdarzeniu

Skrypty globalne są przydatne w wielu przypadkach. Na przykład, jeśli są skonfigurowane dla operacji akcji lub ręcznych działań na hoście, można używać skryptów globalnych do automatycznego lub ręcznego wykonywania zdalnych poleceń, takich jak ponowne uruchamianie aplikacji (serwera WWW, middleware, CRM itp.) lub zwalnianie miejsca na dysku (usuwanie starszych plików, czyszczenie `/tmp` itp.). Inny przykład: jeśli są skonfigurowane dla ręcznych działań na zdarzeniach, można używać skryptów globalnych do zarządzania zgłoszeniami problemów w systemach zewnętrznych.

Skrypty globalne mogą być wykonywane przez serwer Zabbix, proxy lub agent.

> **Attention:**
> Polecenia zdefiniowane przez użytkownika są wykonywane z poziomu użytkownika systemu operacyjnego używanego do uruchamiania komponentów Zabbix. Aby wykonać te polecenia, ten użytkownik musi mieć niezbędne uprawnienia.

Zabbix API Zabbix API to API oparte na HTTP, będące częścią frontend Zabbix. Za pomocą Zabbix API można wykonywać następujące operacje:

- Programowo pobierać i modyfikować konfigurację Zabbix.
- Importować i eksportować konfigurację Zabbix.
- Uzyskiwać dostęp do danych historycznych i trendów Zabbix.
- Konfigurować aplikacje do współpracy z Zabbix.
- Integrować Zabbix z oprogramowaniem firm trzecich.
- Automatyzować rutynowe zadania.

Zabbix API składa się z wielu metod, które są umownie pogrupowane w oddzielne API. Każda metoda wykonuje określone zadanie. Informacje o dostępnych metodach, a także przegląd funkcji udostępnianych przez Zabbix API, można znaleźć w referencji metod Zabbix API.

1 Moduły ładowalne

Przegląd

Moduły ładowalne oferują rozwiązanie nastawione na wydajność do rozszerzania funkcjonalności Zabbix.

Funkcjonalność Zabbix można rozszerzać na wiele sposobów, na przykład za pomocą parametrów użytkownika, kontroli zewnętrznych oraz pozycji agenta Zabbix system.run. Rozwiązania te działają bardzo dobrze, ale mają jedną istotną wadę, a mianowicie fork(). Zabbix musi utworzyć nowy proces za każdym razem, gdy obsługuje metrykę użytkownika, co nie jest korzystne dla wydajności. Zwykle nie stanowi to dużego problemu, jednak może być poważnym wyzwaniem podczas monitorowania systemów wbudowanych, przy dużej liczbie monitorowanych parametrów lub w przypadku ciężkich skryptów ze złożoną logiką albo długim czasem uruchamiania.

Obsługa modułów ładowalnych umożliwia rozszerzanie agenta Zabbix, serwera i proxy bez poświęcania wydajności.

Moduł ładowalny jest zasadniczo biblioteką współdzieloną używaną przez demon Zabbix i ładowaną podczas uruchamiania. Biblioteka powinna zawierać określone funkcje, aby proces Zabbix mógł wykryć, że plik jest rzeczywiście modułem, który można załadować i z którym można pracować.

Moduły ładowalne mają szereg zalet. Bardzo ważne są wysoka wydajność i możliwość implementacji dowolnej logiki, ale być może najważniejszą zaletą jest możliwość tworzenia, używania i udostępniania modułów Zabbix. Przyczynia się to do bezproblemowego utrzymania i ułatwia dostarczanie nowej funkcjonalności w prostszy sposób oraz niezależnie od bazowego kodu Zabbix.

Licencjonowanie modułów i ich dystrybucja w postaci binarnej są regulowane licencją AGPL-3.0 (moduły są linkowane z Zabbix w czasie działania i używają nagłówków Zabbix; cały kod Zabbix jest objęty licencją AGPL-3.0 od wersji Zabbix 7.0). Zabbix nie gwarantuje zgodności binarnej.

Stabilność API modułów jest gwarantowana w trakcie jednego cyklu wydania Zabbix LTS (Long Term Support). Stabilność API Zabbix nie jest gwarantowana (technicznie możliwe jest wywoływanie wewnętrznych funkcji Zabbix z modułu, ale nie ma gwarancji, że takie moduły będą działać).

API modułu

Aby biblioteka współdzielona była traktowana jako moduł Zabbix, powinna implementować i eksportować kilka funkcji. Obecnie w API modułu Zabbix istnieje sześć funkcji, z których tylko jedna jest obowiązkowa, a pozostałe pięć jest opcjonalnych.

Wymagany interfejs

Jedyną obowiązkową funkcją jest **zbx_module_api_version()**:

```
int zbx_module_api_version(void);
```

Ta funkcja powinna zwracać wersję API zaimplementowaną przez ten moduł, a aby moduł mógł zostać załadowany, wersja ta musi być zgodna z wersją API modułu obsługiwaną przez Zabbix. Wersja API modułu obsługiwana przez Zabbix to ZBX_MODULE_API_VERSION. Dlatego funkcja ta powinna zwracać tę stałą. Stara stała ZBX_MODULE_API_VERSION_ONE, używana wcześniej do tego celu, jest obecnie zdefiniowana jako równa ZBX_MODULE_API_VERSION w celu zachowania zgodności ze źródłami, jednak jej użycie nie jest zalecane.

Interfejs opcjonalny

Funkcjami opcjonalnymi są **zbx_module_init()**, **zbx_module_item_list()**, **zbx_module_item_timeout()**, **zbx_module_history_write_cbs** oraz **zbx_module_uninit()**:

```
int zbx_module_init(void);
```

Ta funkcja powinna wykonać niezbędną inicjalizację modułu (jeśli jest wymagana). W przypadku powodzenia powinna zwrócić ZBX_MODULE_OK. W przeciwnym razie powinna zwrócić ZBX_MODULE_FAIL. W tym ostatnim przypadku Zabbix nie uruchomi się.

```
ZBX_METRIC  *zbx_module_item_list(void);
```

Ta funkcja powinna zwrócić listę pozycji obsługiwanych przez moduł. Każda pozycja jest zdefiniowana w strukturze ZBX_METRIC; szczegóły znajdują się w sekcji poniżej. Lista jest zakończona strukturą ZBX_METRIC, w której pole ”key” ma wartość NULL.

```
void    zbx_module_item_timeout(int timeout);
```

Jeśli moduł eksportuje **zbx_module_item_list()**, to ta funkcja jest używana przez Zabbix do określenia ustawień timeout z pliku konfiguracyjnego Zabbix, których powinny przestrzegać kontrole pozycji zaimplementowane przez moduł. Tutaj parametr ”timeout” jest podawany w sekundach.

```
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void);
```

Ta funkcja powinna zwrócić funkcje callback, których Zabbix server będzie używać do eksportowania historii różnych typów danych. Funkcje callback są udostępniane jako pola struktury ZBX_HISTORY_WRITE_CBS; pola mogą mieć wartość NULL, jeśli moduł nie jest zainteresowany historią określonego typu.

```
int zbx_module_uninit(void);
```

Ta funkcja powinna wykonać niezbędną deinicjalizację (jeśli jest wymagana), taką jak zwalnianie przydzielonych zasobów, zamykanie deskryptorów plików itp.

Wszystkie funkcje są wywoływane jednokrotnie podczas uruchamiania Zabbix, gdy moduł jest ładowany, z wyjątkiem zbx_module_uninit(), która jest wywoływana jednokrotnie podczas zamykania Zabbix, gdy moduł jest wyładowywany.

Definiowanie pozycji

Każda pozycja jest definiowana w strukturze ZBX_METRIC:

```
typedef struct
{
    char        *key;
    unsigned    flags;
    int     (*function)();
    char        *test_param;
}
ZBX_METRIC;
```

Tutaj **key** to klucz pozycji (np. ”dummy.random”), **flags** to albo CF_HAVEPARAMS, albo 0 (w zależności od tego, czy pozycja przyjmuje parametry), **function** to funkcja C implementująca pozycję (np. ”zbx_module_dummy_random”), a **test_param** to lista parametrów używana, gdy agent Zabbix jest uruchamiany z flagą ”-p” (np. ”1,1000”; może mieć wartość NULL). Przykładowa definicja może wyglądać następująco:

```
static ZBX_METRIC keys[] =
{
    { "dummy.random", CF_HAVEPARAMS, zbx_module_dummy_random, "1,1000" },
    { NULL }
}
```

Każda funkcja implementująca pozycję powinna przyjmować dwa parametry wskaźnikowe: pierwszy typu AGENT_REQUEST, a drugi typu AGENT_RESULT:

```
int zbx_module_dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    ...

    SET_UI64_RESULT(result, from + rand() % (to - from + 1));

    return SYSINFO_RET_OK;
}
```

Funkcje te powinny zwracać SYSINFO_RET_OK, jeśli wartość pozycji została pomyślnie pobrana. W przeciwnym razie powinny zwracać SYSINFO_RET_FAIL. Szczegółowe informacje o tym, jak pobierać informacje z AGENT_REQUEST i jak ustawiać informacje w AGENT_RESULT, można znaleźć w przykładzie modułu ”dummy” poniżej.

Udostępnianie callbacków eksportu historii

Attention:
Eksport historii przez moduł nie jest już obsługiwany przez Zabbix proxy.

Moduł może określać funkcje do eksportu danych historii według typu: numeryczne (float), numeryczne (unsigned), znakowe, tekstowe i log:

```
typedef struct
{
    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;
```

Każda z nich powinna przyjmować jako argumenty tablicę ”history” złożoną z ”history_num” elementów. W zależności od typu eksportowanych danych historii, ”history” jest odpowiednio tablicą następujących struktur:

```
typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    double      value;
}
ZBX_HISTORY_FLOAT;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    zbx_uint64_t    value;
}
ZBX_HISTORY_INTEGER;
```

```
typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
}
ZBX_HISTORY_STRING;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
}
ZBX_HISTORY_TEXT;

typedef struct
{
    zbx_uint64_t    itemid;
    int     clock;
    int     ns;
    const char  *value;
    const char  *source;
    int     timestamp;
    int     logeventid;
    int     severity;
}
ZBX_HISTORY_LOG;
```

Callbacki będą używane przez procesy synchronizacji historii Zabbix serwer na końcu procedury synchronizacji historii, po zapisaniu danych do bazy danych Zabbix i zachowaniu ich w pamięci podręcznej wartości.

> **Attention:**
> W przypadku błędu wewnętrznego w module eksportu historii zaleca się, aby moduł był napisany w taki sposób, by nie blokował całego monitoringu do czasu odzyskania sprawności, lecz zamiast tego odrzucał dane i pozwalał Zabbix serwer kontynuować działanie.

Budowanie modułów

Moduły są obecnie przeznaczone do budowania wewnątrz drzewa źródeł Zabbix, ponieważ API modułów zależy od niektórych struktur danych zdefiniowanych w nagłówkach Zabbix.

Najważniejszym nagłówkiem dla ładowalnych modułów jest **include/module.h**, który definiuje te struktury danych. Inne niezbędne nagłówki systemowe, które pomagają **include/module.h** działać poprawnie, to **stdlib.h** i **stdint.h**.

Mając te informacje na uwadze, wszystko jest gotowe do zbudowania modułu. Moduł powinien dołączać **stdlib.h**, **stdint.h** i **module.h**, a skrypt budowania powinien zapewnić, że pliki te znajdują się w ścieżce dołączania. Szczegóły można znaleźć w przykładzie modułu „dummy" poniżej.

Innym przydatnym nagłówkiem jest **include/zbxcommon.h**, który definiuje funkcję **zabbix_log()**, której można używać do celów rejestrowania i debugowania.

Parametry konfiguracyjne

Zabbix agent, serwer i proxy obsługują dwa parametry służące do obsługi modułów:

- LoadModulePath - pełna ścieżka do lokalizacji ładowalnych modułów
- LoadModule - moduł(y) do załadowania podczas uruchamiania. Moduły muszą znajdować się w katalogu określonym przez LoadModulePath lub ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od '/'), parametr LoadModulePath jest ignorowany. Dozwolone jest uwzględnienie wielu parametrów LoadModule.

Na przykład, aby rozszerzyć Zabbix agent, można dodać następujące parametry:

```
LoadModulePath=/usr/local/lib/zabbix/agent/
LoadModule=mariadb.so
```

```
LoadModule=apache.so
LoadModule=kernel.so
LoadModule=/usr/local/lib/zabbix/dummy.so
```

Podczas uruchamiania agent załaduje moduły mariadb.so, apache.so i kernel.so z katalogu /usr/local/lib/zabbix/agent, natomiast dummy.so zostanie załadowany z /usr/local/lib/zabbix. Agent nie uruchomi się, jeśli brakuje modułu, w przypadku nieprawidłowych uprawnień lub jeśli biblioteka współdzielona nie jest modułem Zabbix.

Konfiguracja frontendu

Ładowalne moduły są obsługiwane przez Zabbix agent, serwer i proxy. Dlatego typ pozycji w Zabbix frontend zależy od miejsca załadowania modułu. Jeśli moduł jest załadowany do agent, typ pozycji powinien być ustawiony na „Zabbix agent" lub „Zabbix agent (active)". Jeśli moduł jest załadowany do serwera lub proxy, typ pozycji powinien być ustawiony na „Simple check".

Eksport historii za pośrednictwem modułów Zabbix nie wymaga żadnej konfiguracji frontendu. Jeśli moduł zostanie pomyślnie załadowany przez serwer i udostępnia funkcję **zbx_module_history_write_cbs()**, która zwraca co najmniej jedną funkcję callback różną od NULL, eksport historii zostanie włączony automatycznie.

Moduł dummy

Zabbix zawiera przykładowy moduł napisany w języku C. Moduł znajduje się w katalogu src/modules/dummy:

```
alex@alex:~trunk/src/modules/dummy$ ls -l
-rw-rw-r-- 1 alex alex 9019 Apr 24 17:54 dummy.c
-rw-rw-r-- 1 alex alex   67 Apr 24 17:54 Makefile
-rw-rw-r-- 1 alex alex  245 Apr 24 17:54 README
```

Moduł jest dobrze udokumentowany i może być używany jako szablon dla własnych modułów.

Po uruchomieniu ./configure w katalogu głównym drzewa źródeł Zabbix, zgodnie z opisem powyżej, wystarczy uruchomić **make**, aby zbudować **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);
```

```c
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;
}
```

Moduł eksportuje trzy nowe pozycje:

- `dummy.ping` — zawsze zwraca „1”
- `dummy.echo[param1]` — zwraca pierwszy parametr bez zmian, na przykład `dummy.echo[ABC]` zwróci ABC
- `dummy.random[param1, param2]` — zwraca liczbę losową z zakresu param1-param2, na przykład `dummy.random[1,1000000]`

Ograniczenia

Obsługa modułów ładowalnych jest zaimplementowana wyłącznie dla platformy Unix. Oznacza to, że nie działa ona dla agentów Windows.

W niektórych przypadkach moduł może wymagać odczytu parametrów konfiguracyjnych związanych z modułem z pliku *zabbix_agentd.conf*. Obecnie nie jest to obsługiwane. Jeśli potrzebujesz, aby Twój moduł korzystał z parametrów konfiguracyjnych, prawdopodobnie powinieneś zaimplementować analizę pliku konfiguracyjnego specyficznego dla modułu.

2 Wtyczki

Omówienie

Wtyczki rozszerzają możliwości monitorowania Zabbix agent 2. Napisane w języku programowania Go, stanowią alternatywę dla ładowalnych modułów (napisanych w C) oraz innych metod rozszerzania funkcjonalności Zabbix.

Te funkcje są specyficzne dla agent 2 i jego wtyczek:

- Zaplanowane i elastyczne interwały zarówno dla kontroli pasywnych, jak i aktywnych
- Zarządzanie kolejką zadań z uwzględnieniem harmonogramu i współbieżności zadań
- Limity czasu na poziomie wtyczki
- Sprawdzanie zgodności Zabbix agent 2 i jego wtyczek podczas uruchamiania

Wszystkie metryki obsługiwane przez Zabbix agent 2 są zbierane przez wtyczki. Ta strona opisuje zasady konfiguracji wtyczek z perspektywy użytkownika oraz zawiera listę wbudowanych i ładowalnych wtyczek dostarczanych przez Zabbix.

> **Note:**
> Od Zabbix 6.0 wtyczki nie muszą już być wbudowane w agent 2. Można dodawać je jako ładowalne wtyczki, co upraszcza tworzenie wtyczek dla nowych metryk. Instrukcje dotyczące tworzenia własnych wtyczek znajdują się w Developer center oraz w Przykładowej wtyczce dla Zabbix agent 2.

Konfigurowanie wtyczek

Ta sekcja zawiera ogólne zasady konfiguracji wtyczek oraz najlepsze praktyki.

Wszystkie wtyczki są konfigurowane za pomocą parametru *Plugins.** , który może być częścią pliku konfiguracyjnego Zabbix agent 2 lub własnego pliku konfiguracyjnego danej wtyczki. Jeśli wtyczka używa oddzielnego pliku konfiguracyjnego, ścieżkę do tego pliku należy określić w parametrze Include pliku konfiguracyjnego Zabbix agent 2.

Typowy parametr wtyczki ma następującą strukturę:

Plugins.<PluginName>.<Parameter>=<Value>

Dodatkowo istnieją dwie specyficzne grupy parametrów:

- *Plugins.<PluginName>.Default.<Parameter>=<Value>* używane do definiowania domyślnych wartości parametrów.

- *Plugins.<PluginName>.<SessionName>.<Parameter>=<Value>* używane do definiowania oddzielnych zestawów parametrów dla różnych celów monitorowania za pomocą nazwanych sesji.

Wszystkie nazwy parametrów powinny spełniać następujące wymagania:

- zaleca się, aby nazwy wtyczek były zapisywane wielkimi literami
- parametr powinien być zapisany wielką literą
- znaki specjalne nie są dozwolone
- zagnieżdżenie nie jest ograniczone maksymalnym poziomem
- liczba parametrów nie jest ograniczona

Na przykład, aby wykonać aktywnych kontroli, które nie mają *Scheduling* interwału aktualizacji bezpośrednio po ponownym uruchomieniu agenta tylko dla wtyczki Uptime, ustaw `Plugins.Uptime.System.ForceActiveChecksOnStart=1` w pliku konfiguracyjnym. Podobnie, aby ustawić niestandardowy limit dla równoczesnych kontroli dla wtyczki CPU, ustaw `Plugins.CPU.System.Capacity=N` w pliku konfiguracyjnym.

Wartości domyślne

Możesz ustawić wartości domyślne dla parametrów związanych z połączeniem (URI, nazwa użytkownika, hasło itp.) w pliku konfiguracyjnym w formacie:

```
Plugins.<PluginName>.Default.<Parameter>=<Value>
```

Na przykład `Plugins.Mysql.Default.Username=zabbix`, `Plugins.MongoDB.Default.Uri=tcp://127.0.0.1:27017` itd.

Jeśli wartość takiego parametru nie zostanie podana w kluczu pozycji ani w parametrach nazwanej sesji, plugin użyje wartości domyślnej. Jeśli parametr domyślny również nie jest zdefiniowany, zostaną użyte wartości domyślne zapisane na stałe w kodzie.

> **Note:**
> Jeśli klucz pozycji nie ma żadnych parametrów, agent Zabbix 2 spróbuje zebrać metrykę, używając wartości zdefiniowanych w sekcji parametrów domyślnych.

Nazwane sesje

Nazwane sesje reprezentują dodatkowy poziom parametrów wtyczki i mogą być używane do określania oddzielnych zestawów parametrów uwierzytelniania dla każdej z monitorowanych instancji. Każdy parametr nazwanej sesji powinien mieć następującą strukturę:

```
Plugins.<PluginName>.Sessions.<SessionName>.<Parameter>=<Value>
```

Nazwa sesji może być użyta jako parametr klucza elementu connString zamiast osobnego podawania URI, nazwy użytkownika i/lub hasła.

W kluczach pozycji pierwszy parametr może być albo connString, albo URI. Jeśli pierwszy parametr klucza nie pasuje do żadnej nazwy sesji, zostanie potraktowany jako URI. Należy pamiętać, że przekazywanie osadzonych poświadczeń URI w kluczu pozycji nie jest obsługiwane; zamiast tego należy używać parametrów nazwanej sesji.

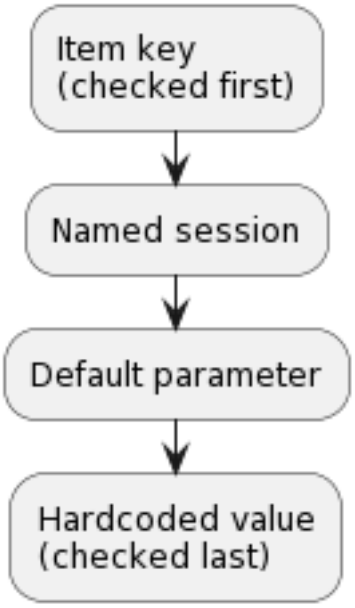
Lista dostępnych parametrów nazwanej sesji zależy od wtyczki.

Możliwe jest nadpisanie parametrów sesji przez określenie nowych wartości w parametrach klucza pozycji (zobacz przykład).

Jeśli parametr nie jest zdefiniowany dla nazwanej sesji, Zabbix agent 2 użyje wartości zdefiniowanej w domyślnym parametrze wtyczki.

Priorytet parametrów

Wtyczki Zabbix agent 2 wyszukują wartości parametrów związanych z połączeniem w następującej kolejności:



1. Pierwszy parametr klucza pozycji jest porównywany z nazwami sesji. Jeśli nie zostanie znalezione dopasowanie, jest on traktowany jako rzeczywista wartość; w takim przypadku krok 3 zostanie pominięty. Jeśli dopasowanie zostanie znalezione, wartość parametru (zwykle URI) musi być zdefiniowana w nazwanej sesji.
2. Pozostałe parametry zostaną pobrane z klucza pozycji, jeśli są zdefiniowane.
3. Jeśli parametr klucza pozycji (na przykład password) jest pusty, wtyczka wyszuka odpowiadający mu parametr nazwanej sesji.
4. Jeśli parametr sesji również nie został określony, zostanie użyta wartość zdefiniowana w odpowiednim domyślnym parametrze.
5. Jeśli wszystko inne zawiedzie, wtyczka użyje zakodowanej na stałe wartości domyślnej.

Przykład 1

Monitorowanie dwóch instancji "MySQL1" i "MySQL2".

Parametry konfiguracji:

```
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://192.0.2.1:3306
Plugins.Mysql.Sessions.MySQL2.User=mysql2_user
Plugins.Mysql.Sessions.MySQL2.Password=different_password
```

W wyniku tej konfiguracji każda nazwa sesji może być używana jako connString w kluczu pozycji, np. `mysql.ping[MySQL1]` lub `mysql.ping[MySQL2]`.

Przykład 2

Podanie części parametrów w kluczu pozycji.

Parametry konfiguracji:

```
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
```

Klucz pozycji: `pgsql.ping[session1,new_username,,postgres]`

W wyniku tej konfiguracji agent połączy się z PostgreSQL, używając następujących parametrów:

- URI z parametru sesji: `192.0.2.234:5432`
- Nazwa użytkownika z klucza pozycji: `new_username`
- Hasło z parametru sesji (ponieważ zostało pominięte w kluczu pozycji): `session_password`
- Nazwa bazy danych z klucza pozycji: `postgres`

Przykład 3

Zbieranie metryki przy użyciu domyślnych parametrów konfiguracji.

Parametry konfiguracji:

```
Plugins.PostgreSQL.Default.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Default.User=zabbix
Plugins.PostgreSQL.Default.Password=password
```

Klucz pozycji: `pgsql.ping[,,,postgres]`

W wyniku tej konfiguracji agent połączy się z PostgreSQL, używając następujących parametrów:

- Domyślny URI: `192.0.2.234:5432`
- Domyślna nazwa użytkownika: `zabbix`
- Domyślne hasło: `password`
- Nazwa bazy danych z klucza pozycji: `postgres`

Połączenia

Niektóre wtyczki obsługują zbieranie metryk z wielu instancji jednocześnie. Można monitorować zarówno instancje lokalne, jak i zdalne. Obsługiwane są połączenia TCP oraz Unix-socket.

Zaleca się skonfigurowanie wtyczek tak, aby utrzymywały połączenia z instancjami w stanie otwartym. Korzyści obejmują mniejsze przeciążenie sieci, niższe opóźnienia oraz mniejsze zużycie CPU i pamięci dzięki mniejszej liczbie połączeń. Biblioteka klienta zajmuje się tym automatycznie.

Note:
Okres, przez jaki nieużywane połączenia powinny pozostać otwarte, można określić za pomocą parametru *Plugins.<PluginName>.KeepAlive*. Przykład: *Plugins.Memcached.KeepAlive*

Wbudowane pluginy

Wbudowane pluginy Zabbix agent 2 są dostępne od razu po instalacji. Kliknij nazwę pluginu, aby przejść do repozytorium pluginu z dodatkowymi informacjami.

Nazwa pluginu	Opis	Obsługiwane klucze pozycji	Uwagi
Agent	Metryki używanego Zabbix agent.	agent.hostname, agent.ping, agent.version	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Ceph	Monitorowanie Ceph.	ceph.df.details, ceph.osd.stats, ceph.osd.discovery, ceph.osd.dump, ceph.ping, ceph.pool.discovery, ceph.status	
CPU	Monitorowanie CPU systemu (liczba CPU/rdzeni CPU, wykryte CPU, procent wykorzystania).	system.cpu.discovery, system.cpu.num, system.cpu.util	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Docker	Monitorowanie kontenerów Docker.	docker.container_info, docker.container_stats, docker.containers, docker.containers.discovery, docker.data_usage, docker.images, docker.images.discovery, docker.info, docker.ping	Zobacz także: Parametry konfiguracji

Nazwa pluginu	Opis	Obsługiwane klucze pozycji	Uwagi
File	Zbieranie metryk plików.	vfs.file.cksum, vfs.file.contents, vfs.file.exists, vfs.file.md5sum, vfs.file.regexp, vfs.file.regmatch, vfs.file.size, vfs.file.time	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Kernel	Monitorowanie jądra systemu.	kernel.maxfiles, kernel.maxproc	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Log	Monitorowanie plików dziennika.	log, log.count, logrt, logrt.count	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys. Zobacz także: Parametry konfiguracji pluginu (Unix/Windows)
Memcached	Monitorowanie serwera Memcached.	memcached.ping, memcached.stats	
Modbus	Odczytuje dane Modbus.	modbus.get	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
MQTT	Odbiera opublikowane wartości tematów MQTT.	mqtt.get	Aby skonfigurować szyfrowane połączenie z brokerem MQTT, określ parametry TLS w pliku konfiguracji agent jako nazwaną sesję lub parametry domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji.
MySQL	Monitorowanie MySQL i jego forków.	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	Aby skonfigurować szyfrowane połączenie z bazą danych, określ parametry TLS w pliku konfiguracji agent jako nazwaną sesję lub parametry domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji.
NetIf	Monitorowanie interfejsów sieciowych.	net.if.collisions, net.if.discovery, net.if.in, net.if.out, net.if.total	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Nazwa pluginu	Opis	Obsługiwane klucze pozycji	Uwagi
Oracle	Monitorowanie Oracle Database.	oracle.diskgroups.stats, oracle.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	Przed użyciem pluginu zainstaluj Oracle Instant Client.
Proc	Procent wykorzystania CPU procesu.	proc.cpu.util	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
Redis	Monitorowanie serwera Redis.	redis.config, redis.info, redis.ping, redis.slowlog.count	Aby skonfigurować szyfrowane połączenie z Redis, określ parametry TLS w pliku konfiguracji agent jako nazwaną sesję lub parametry domyślne. Parametrów TLS nie można przekazywać jako parametrów klucza pozycji. Należy pamiętać, że nieprawidłowa lub w inny sposób niepoprawna konfiguracja TLS może uniemożliwić uruchomienie Zabbix agent 2, dlatego przed włączeniem TLS sprawdź pliki certyfikatów, uprawnienia i ścieżki.

Smart	Monitorowanie S.M.A.R.T.	smart.attribute.discovery, smart.disk.discovery, smart.disk.get	Minimalna wymagana wersja smartctl to 7.1.
			Użytkownik uruchamiający Zabbix agent 2 musi mieć uprawnienia sudo/root do smartctl. Plugin używa tylko następujących poleceń:
			`/usr/sbin/smartctl -a *`
			`/usr/sbin/smartctl --scan *`
			`/usr/sbin/smartctl -j -V`
			
			Obsługiwane keys mogą być używane z Zabbix agent 2 tylko w systemach Linux/Windows, zarówno jako sprawdzenie pasywne, jak i aktywne.
			Zobacz także:
			Parametry konfiguracji

SW	Lista zainstalowanych pakietów.	system.sw.packages, system.sw.packages.get	Obsługiwane klucze mają te same parametry co klucz Zabbix agent key.
Swap	Rozmiar przestrzeni swap w bajtach/procentach.	system.swap.size	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.

Nazwa pluginu	Opis	Obsługiwane klucze pozycji	Uwagi
SystemRun	Uruchamia określone polecenie.	system.run	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key. Zobacz także: Parametry konfiguracji pluginu (Unix/Windows)
Systemd	Monitorowanie usług systemd.	systemd.unit.discovery, systemd.unit.get, systemd.unit.info	
TCP	Sprawdzanie dostępności połączenia TCP.	net.tcp.port	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
UDP	Monitorowanie dostępności i wydajności usług UDP.	net.udp.service, net.udp.service.perf	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Uname	Pobieranie informacji o systemie.	system.hostname, system.sw.arch, system.uname	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Uptime	Zbieranie metryk czasu działania systemu.	system.uptime	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
VFSDev	Zbieranie metryk VFS.	vfs.dev.discovery, vfs.dev.read, vfs.dev.write	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
WebCertificate	Monitorowanie certyfikatów TLS/SSL witryn internetowych.	web.certificate.get	
WebPage	Monitorowanie stron internetowych.	web.page.get, web.page.perf, web.page.regexp	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
ZabbixAsync	Asynchroniczne zbieranie metryk.	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	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
ZabbixStats	Wewnętrzne metryki serwer/proxy Zabbix lub liczba opóźnionych pozycji w kolejce.	zabbix.stats	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Nazwa pluginu	Opis	Obsługiwane klucze pozycji	Uwagi
ZabbixSync	Synchroniczne zbieranie metryk.	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.	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Wtyczki ładowalne

Wtyczki ładowalne Zabbix agent 2 nie są dostępne od razu po instalacji i trzeba je zainstalować osobno:

- W systemie Linux można użyć pakietów (np. zainstalować Ember+ na Ubuntu poleceniem `apt install zabbix-agent2-plugin-em` albo zbudować wtyczki
- W systemie Windows można zainstalować je z pliku MSI albo zbudować wtyczki

Attention:
Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

Kliknij nazwę wtyczki, aby przejść do repozytorium wtyczki, które zawiera plik README z dodatkowymi informacjami.

Plugin name	Description	Supported item keys	Comments
Ember+	Monitorowanie Ember+.	ember.get	Zobacz także parametry konfiguracji wtyczki Ember+.
MongoDB	Monitorowanie serwerów i klastrów MongoDB (rozproszona baza danych oparta na dokumentach).	mongodb.collection.stats, mongodb.collections.discovery, mongodb.collections.usage, mongodb.connpool.stats, mongodb.db.stats, mongodb.db.discovery, mongodb.jumbo_chunks.count, mongodb.oplog.stats, mongodb.ping, mongodb.rs.config, mongodb.rs.status, mongodb.server.status, mongodb.sh.discovery, mongodb.version	Aby skonfigurować szyfrowane połączenia z bazą danych, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycja. Zobacz także parametry konfiguracji wtyczki MongoDB.

Plugin name	Description	Supported item keys	Comments
MSSQL	Monitorowanie bazy danych 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	Aby skonfigurować szyfrowane połączenie z bazą danych, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycja. Zobacz także parametry konfiguracji wtyczki MSSQL.
NVIDIA GPU	Monitorowanie NVIDIA GPU.	nvml.device.count, nvml.device.decoder.utilization, nvml.device.ecc.mode, nvml.device.encoder.stats.get, nvml.device.encoder.utilization, nvml.device.energy.consumption, nvml.device.errors.memory, nvml.device.errors.register, nvml.device.fan.speed.avg, nvml.device.get, nvml.device.graphics.frequency, nvml.device.memory.bar1.get, nvml.device.memory.fb.get, nvml.device.memory.frequency, nvml.device.pci.utilization, nvml.device.performance.state, nvml.device.power.limit, nvml.device.power.usage, nvml.device.serial, nvml.device.sm.frequency, nvml.device.temperature, nvml.device.utilization, nvml.device.video.frequency, nvml.system.driver.version, nvml.version	Zobacz także parametry konfiguracji wtyczki NVIDIA GPU.

Plugin name	Description	Supported item keys	Comments
PostgreSQL	Monitorowanie PostgreSQL i jego forków.	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	Aby skonfigurować szyfrowane połączenia z bazą danych, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycja. Zobacz także parametry konfiguracji wtyczki PostgreSQL.

Note:
Wtyczki ładowalne, uruchamiane z opcjami:
 - `-V --version` - wyświetlają informacje o wersji wtyczki i licencji;
 - `-h --help` - wyświetlają informacje pomocy.

3 Moduły frontend

Przegląd

Możliwe jest rozszerzenie funkcjonalności frontend Zabbix przez dodanie modułów firm trzecich lub tworzenie własnych modułów bez konieczności zmiany kodu źródłowego Zabbix.

Należy pamiętać, że kod modułu będzie uruchamiany z takimi samymi uprawnieniami jak kod źródłowy Zabbix. Oznacza to, że:

- moduły firm trzecich mogą być szkodliwe. Musisz ufać modułom, które instalujesz;
- błędy w kodzie modułu firmy trzeciej mogą spowodować awarię frontend. Jeśli tak się stanie, po prostu usuń kod modułu z frontend. Gdy tylko ponownie załadujesz frontend Zabbix, zobaczysz informację, że niektóre moduły są nieobecne. Przejdź do Administracja modułami (w *Administracja* → *Ogólne* → *Moduły*) i kliknij ponownie *Skanuj katalog*, aby usunąć nieistniejące moduły z bazy danych.

Instalacja

Zawsze należy przeczytać instrukcję instalacji dla konkretnego modułu. Zaleca się instalowanie nowych modułów pojedynczo, aby łatwo wychwycić błędy.

Tuż przed instalacją modułu:

- Upewnij się, że moduł został pobrany z zaufanego źródła. Instalacja szkodliwego kodu może prowadzić do konsekwencji, takich jak utrata danych
- Różne wersje tego samego modułu (to samo ID) mogą być zainstalowane równolegle, ale jednocześnie może być włączona tylko jedna wersja

Kroki instalacji modułu:

- Rozpakuj moduł we własnym folderze w katalogu `modules` Zabbix frontend
- Upewnij się, że folder modułu zawiera co najmniej plik manifest.json

- Przejdź do Administracja modułami i kliknij przycisk *Scan directory*
- Nowy moduł pojawi się na liście wraz ze swoją wersją, autorem, opisem i statusem
- Włącz moduł, klikając jego status

Rozwiązywanie problemów:

Problem	Rozwiązanie
Moduł nie pojawił się na liście	Jeśli moduł nie pojawił się na liście, upewnij się, że plik manifest.json istnieje w modules/your-module/. Jeśli nie, moduł mógł zostać rozpakowany do niewłaściwego katalogu. Jeśli plik istnieje, moduł może nie być zgodny z używaną wersją Zabbix. Sprawdź również, czy użytkownik serwera WWW ma co najmniej uprawnienia odczytu i przeszukiwania (`r-x`) do folderu modułu oraz wszystkich podkatalogów, a także uprawnienia odczytu (`r--`) do wszystkich znajdujących się w nim plików.
Frontend uległ awarii	Kod modułu nie jest zgodny z bieżącą wersją Zabbix lub konfiguracją serwera. Usuń pliki modułu i ponownie załaduj frontend. Zobaczysz powiadomienie, że niektóre moduły są nieobecne. Przejdź do Administracja modułami i ponownie kliknij *Scan directory*, aby usunąć z bazy danych nieistniejące moduły.
Pojawia się komunikat o błędzie dotyczący identycznej przestrzeni nazw, ID lub akcji	Nowy moduł próbował zarejestrować przestrzeń nazw, ID lub akcje, które są już zarejestrowane przez inne włączone moduły. Wyłącz moduł powodujący konflikt (wspomniany w komunikacie o błędzie) przed włączeniem nowego modułu.
Pojawiają się techniczne komunikaty o błędach	Zgłoś błędy deweloperowi modułu.

Tworzenie modułów

Aby uzyskać informacje o tworzeniu niestandardowych modułów, zobacz Developer center.

22 Aneksy

Przegląd Ta sekcja zawiera materiały referencyjne, parametry konfiguracyjne, specyfikacje techniczne oraz przewodniki po procedurach, które uzupełniają główny podręcznik.

Podstawowe materiały referencyjne

- Instalacja i konfiguracja — tworzenie bazy danych, instalacja i konfiguracja serwera oraz agenta.
- Konfiguracja procesów — parametry konfiguracyjne dla serwera Zabbix, proxy, agenta i innych komponentów.
- Pozycje — informacje referencyjne o kluczach pozycji i obsługiwanych typach danych.
- Makra — składnia i użycie wbudowanych makr oraz makr użytkownika.

1 Instalacja i konfiguracja

Ta sekcja obejmuje tworzenie bazy danych, konfigurację zabezpieczeń, procedury konfiguracji specyficzne dla platformy oraz konfigurację funkcji.

Skorzystaj z paska bocznego, aby znaleźć procedurę odpowiednią dla Twojego środowiska.

1 Tworzenie bazy danych

Przegląd

Baza danych Zabbix musi zostać utworzona podczas instalacji serwera lub proxy Zabbix.

Ta sekcja zawiera instrukcje dotyczące tworzenia bazy danych Zabbix. Dla każdej obsługiwanej bazy danych dostępny jest osobny zestaw instrukcji.

Note:
Aby zwiększyć bezpieczeństwo bazy danych przez utworzenie ról/użytkowników bazy danych z minimalnymi uprawnieniami, zobacz najlepsze praktyki tworzenia bazy danych dla każdej obsługiwanej bazy danych:

- MySQL/MariaDB
- PostgreSQL/TimescaleDB

Aby skonfigurować bezpieczne połączenia TLS, zobacz Bezpieczne połączenie z bazą danych.

UTF-8 jest jedynym kodowaniem obsługiwanym przez Zabbix. Wiadomo, że działa bez żadnych luk bezpieczeństwa. Użytkownicy powinni mieć świadomość, że w przypadku używania niektórych innych kodowań istnieją znane problemy bezpieczeństwa. Informacje o przełączeniu na UTF-8 można znaleźć w sekcji Naprawa zestawu znaków i porównywania bazy danych Zabbix. Zobacz także Ograniczenia filtrowania przy użyciu porównań utf8mb4.

Note:
Jeśli instalujesz z repozytorium Git Zabbix, przed przejściem do następnych kroków musisz uruchomić następujące polecenie:

 `make dbschema`

MySQL/MariaDB

Zestawy znaków utf8 (znane także jako utf8mb3) oraz utf8mb4 są obsługiwane (odpowiednio z porównywaniem utf8_bin i utf8mb4_bin), aby serwer/proxy Zabbix działał poprawnie z bazą danych MySQL. W przypadku nowych instalacji zaleca się użycie utf8mb4.

Deterministyczne wyzwalacze muszą zostać utworzone podczas importu schematu. W MySQL i MariaDB wymaga to ustawienia `GLOBAL log_bin_trust_function_creators = 1`, jeśli logowanie binarne jest włączone, nie ma uprawnień superusera, a `log_bin_trust_function_creators = 1` nie jest ustawione w pliku konfiguracyjnym MySQL.

Jeśli instalujesz zabbix z **pakietów**, przejdź do instrukcji dla swojej platformy.

Jeśli instalujesz Zabbix ze źródeł:

- Utwórz i skonfiguruj bazę danych oraz użytkownika.

```
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;
```

- Zaimportuj dane do bazy danych i ustaw zestaw znaków utf8mb4 jako domyślny (zakładając, że znajdujesz się w katalogu głównym źródeł Zabbix). W przypadku bazy danych proxy Zabbix należy zaimportować tylko `schema.sql` (bez images.sql ani data.sql).

```
cd database/mysql
mysql -uzabbix -p<password> zabbix < schema.sql
#### zatrzymaj się tutaj, jeśli tworzysz bazę danych dla proxy Zabbix
mysql -uzabbix -p<password> zabbix < images.sql
mysql -uzabbix -p<password> --default-character-set=utf8mb4 zabbix < data.sql
```

Po pomyślnym zaimportowaniu schematu `log_bin_trust_function_creators` można wyłączyć:

```
mysql -uroot -p<password>

mysql> SET GLOBAL log_bin_trust_function_creators = 0;
mysql> quit;
```

PostgreSQL

Musisz mieć użytkownika bazy danych z uprawnieniami do tworzenia obiektów bazy danych.

Jeśli instalujesz zabbix z **pakietów**, przejdź do instrukcji dla swojej platformy.

Jeśli instalujesz Zabbix ze źródeł:

- Utwórz użytkownika bazy danych.

Poniższe polecenie powłoki utworzy użytkownika `zabbix`. Po wyświetleniu monitu podaj hasło i powtórz je (uwaga: najpierw możesz zostać poproszony o hasło sudo):

```
sudo -u postgres createuser --pwprompt zabbix
```

- Utwórz bazę danych.

Poniższe polecenie powłoki utworzy bazę danych zabbix (ostatni parametr), a wcześniej utworzony użytkownik będzie jej właścicielem (-O zabbix).

```
sudo -u postgres createdb -O zabbix -E Unicode -T template0 zabbix
```

- Zaimportuj początkowy schemat i dane (zakładając, że jesteś w katalogu głównym źródeł Zabbix). Dla bazy danych proxy Zabbix należy zaimportować tylko schema.sql (bez images.sql ani data.sql).

```
cd database/postgresql
cat schema.sql | sudo -u zabbix psql zabbix
#### zatrzymaj się tutaj, jeśli tworzysz bazę danych dla proxy Zabbix
cat images.sql | sudo -u zabbix psql zabbix
cat data.sql | sudo -u zabbix psql zabbix
```

Attention:
Powyższe polecenia podano jako przykład, który będzie działał w większości instalacji GNU/Linux. Możesz użyć innych poleceń w zależności od konfiguracji systemu/bazy danych, na przykład:

 psql -U <username>

 Jeśli masz jakiekolwiek problemy z konfiguracją bazy danych, skonsultuj się z administratorem bazy danych.

TimescaleDB

Instrukcje dotyczące tworzenia i konfiguracji TimescaleDB znajdują się w osobnej sekcji.

SQLite

Korzystanie z SQLite jest obsługiwane tylko dla **Zabbix proxy**!

Baza danych zostanie utworzona automatycznie, jeśli nie istnieje.

Wróć do sekcji instalacji.

2 Naprawa zestawu znaków i porównywania napisów bazy danych Zabbix

MySQL/MariaDB

Historycznie MySQL i jego pochodne używały „utf8” jako aliasu dla utf8mb3 — własnej 3-bajtowej implementacji standardu UTF8 w MySQL, który jest 4-bajtowy. Począwszy od MySQL 8.0.28 i MariaDB 10.6.1, zestaw znaków „utf8mb3” jest przestarzały i w pewnym momencie jego obsługa zostanie usunięta, podczas gdy „utf8” stanie się odniesieniem do „utf8mb4”. Od Zabbix 6.0 obsługiwany jest „utf8mb4”. Aby uniknąć problemów w przyszłości, zdecydowanie zaleca się używanie „utf8mb4”. Kolejną zaletą przejścia na „utf8mb4” jest obsługa dodatkowych znaków Unicode.

Warning:
Ponieważ wersje wcześniejsze niż Zabbix 6.0 nie obsługują utf8mb4, przed wykonaniem konwersji do utf8mb4 należy najpierw zaktualizować serwer Zabbix i schemat bazy danych do wersji 6.0.x lub nowszej.

1. Sprawdź zestaw znaków i porównywanie bazy danych.

Na przykład:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| latin2                   | latin2_general_ci    |
+--------------------------+----------------------+
```

Lub:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8                     | utf8_bin             |
+--------------------------+----------------------+
```

Jak widać, zestaw znaków nie jest tutaj ustawiony na „utf8mb4”, a porównywanie nie jest ustawione na „utf8mb4_bin”, więc należy to poprawić.

2. Zatrzymaj Zabbix.

3. Utwórz kopię zapasową bazy danych!

4. Popraw zestaw znaków i porównywanie na poziomie bazy danych:

```
alter database <your DB name> character set utf8mb4 collate utf8mb4_bin;
```

Poprawione wartości:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8mb4                  | utf8mb4_bin          |
+--------------------------+----------------------+
```

5. Załaduj skrypt w celu poprawienia zestawu znaków i porównywania na poziomie tabel i kolumn:

```
mysql <your DB name> < utf8mb4_convert.sql
```

6. Wykonaj skrypt:

```
                SET @ZABBIX_DATABASE = '<your DB name>';
If MariaDB →    set innodb_strict_mode = OFF;
                CALL zbx_convert_utf8();
If MariaDB →    set innodb_strict_mode = ON;
                drop procedure zbx_convert_utf8;
```

Należy pamiętać, że „utf8mb4” może zużywać nieco więcej miejsca na dysku.

7. Jeśli nie wystąpiły błędy, możesz utworzyć kopię zapasową poprawionej bazy danych.

8. Uruchom Zabbix.

3 Database upgrade to primary keys

Przegląd

Ta sekcja zawiera instrukcje dotyczące ręcznej aktualizacji tabel w istniejących instalacjach do kluczy głównych.

Aktualizacja do kluczy głównych optymalizuje sposób indeksowania danych i uzyskiwania do nich dostępu, co może przyspieszyć zapytania i zaoszczędzić miejsce. Poprawia także zarządzanie danymi i synchronizację w środowiskach klastrowych, pomagając w skalowaniu i zapewniając niezawodność systemu nawet w przypadku awarii niektórych serwerów.

> **Attention:**
> Instrukcje podane na tej stronie są przeznaczone dla zaawansowanych użytkowników i mogą wymagać dostosowania do konkretnej konfiguracji. Aktualizacja do kluczy głównych może być czasochłonna i zasobochłonna. Upewnij się, że dostępna jest wystarczająca ilość wolnego miejsca na dysku; w zależności od rozmiaru bazy danych i przechowywanych danych proces może wymagać nawet 2,5 razy więcej miejsca niż obecnie zajmują tabele historii.

Klucze główne są używane we wszystkich tabelach w nowych instalacjach od Zabbix 6.0.

Nie ma automatycznej aktualizacji bazy danych do kluczy głównych; jednak istniejące instalacje można zaktualizować ręcznie **po** aktualizacji serwera Zabbix do wersji 6.0 lub nowszej.

> **Attention:**
> Od Zabbix 7.0 aktualizacja tabel do kluczy głównych obejmuje również aktualizację tabel do używania typów danych double precision.

 Jeśli używasz Zabbix 7.0 (lub nowszego), tabele już korzystają z typu double precision. Jednak instrukcje na tej stronie nadal można zastosować, aby zaktualizować tabele do kluczy głównych bez wpływu na tabele, które już używają typu double precision.

 Jeśli używasz Zabbix 6.4 (lub wcześniejszego), rozważ najpierw aktualizację tabel do typu double precision. Więcej informacji można znaleźć w sekcji Upgrading to numeric values of extended range w dokumentacji Zabbix 7.0.

Dostępne są instrukcje dla:

- MySQL

- PostgreSQL
- TimescaleDB

Ważne uwagi

Aby przeprowadzić aktualizację bazy danych:

1. Zatrzymaj serwer Zabbix.

Zdecydowanie zaleca się zatrzymanie serwera Zabbix na czas aktualizacji. Jeśli jednak jest to absolutnie konieczne, możesz przeprowadzić aktualizację, gdy serwer działa (dotyczy tylko MySQL, MariaDB oraz PostgreSQL bez TimescaleDB).

2. Wykonaj kopię zapasową bazy danych.
3. Zainstaluj najnowszy pakiet zabbix-sql-scripts zgodny z używaną wersją Zabbix (np. dla RHEL: `dnf install zabbix-sql-scripts`).
4. Uruchom skrypty dla swojej bazy danych.
5. Uruchom serwer Zabbix.

Warning:
Uruchom skrypty tylko dla bazy danych serwera. proxy nie odniesie korzyści z tej aktualizacji.

Jeśli baza danych używa partycji, skontaktuj się z administratorem DB lub pomocą techniczną Zabbix.

Pliki CSV można usunąć po pomyślnej aktualizacji do kluczy głównych.

Opcjonalnie frontend Zabbix można przełączyć w tryb konserwacji.

MySQL

Eksport i import muszą być wykonywane w tmux/screen, aby zapewnić, że sesja nie zostanie przerwana.

Zobacz także: Ważne uwagi

MySQL 8.0+ z mysqlsh

Ta metoda może być używana przy działającym serwerze Zabbix, ale zaleca się zatrzymanie serwera na czas aktualizacji. MySQL Shell (*mysqlsh*) musi być zainstalowany i musi mieć możliwość połączenia z bazą danych.

- Zaloguj się do konsoli MySQL jako root (zalecane) lub jako dowolny użytkownik z uprawnieniami FILE.
- Uruchom MySQL z włączoną zmienną local_infile.
- Zmień nazwy starych tabel i utwórz nowe tabele, uruchamiając `history_upgrade_prepare.sql`.

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

- Wyeksportuj i zaimportuj dane.

Połącz się przez mysqlsh. Jeśli używasz połączenia socket, może być wymagane podanie ścieżki.

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Przełącz się w tryb JavaScript, używając:

```
\js
```

Następnie uruchom poniższy kod (CSVPATH można w razie potrzeby zmienić):

```
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" });
```

Jeśli pojawi się komunikat "JavaScript is not supported", oznacza to, że instalacja MySQL Shell nie obsługuje JS. W takim przypadku zainstaluj oficjalny pakiet MySQL Shell firmy Oracle (lub zbuduj go ze źródeł), aby włączyć tryb JavaScript.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

MariaDB/MySQL 8.0+ bez mysqlsh

Ta metoda aktualizacji trwa dłużej i powinna być używana tylko wtedy, gdy aktualizacja z użyciem *mysqlsh* nie jest możliwa.

Aktualizacja tabel

- Zaloguj się do konsoli MySQL jako root (zalecane) lub dowolny użytkownik z uprawnieniami FILE.
- Jeśli wykonujesz migrację przy działającym serwerze Zabbix, uruchom MySQL z włączoną zmienną local_infile.
- Zmień nazwy starych tabel i utwórz nowe tabele, uruchamiając `history_upgrade_prepare.sql`:

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

Migracja przy zatrzymanym serwerze

`max_execution_time` (w MySQL) lub `max_statement_time` (w MariaDB) musi zostać wyłączona przed migracją danych, aby uniknąć przekroczenia limitu czasu podczas migracji.

Dla MySQL:

```
SET @@max_execution_time=0;
```

Dla MariaDB:

```
SET @@max_statement_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;
```

Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Migracja przy działającym serwerze

Sprawdź, dla jakich ścieżek włączony jest import/eksport:

```
mysql> SELECT @@secure_file_priv;
+-----------------------+
| @@secure_file_priv    |
+-----------------------+
| /var/lib/mysql-files/ |
+-----------------------+
```

Jeśli wartość *secure_file_priv* jest ścieżką do katalogu, eksport/import będzie wykonywany dla plików w tym katalogu. W takim przypadku odpowiednio zmień ścieżki do plików w zapytaniach albo ustaw wartość *secure_file_priv* na pusty ciąg znaków na czas aktualizacji.

Jeśli wartość *secure_file_priv* jest pusta, eksport/import może być wykonywany z dowolnej lokalizacji.

Jeśli wartość *secure_file_priv* ma wartość NULL, ustaw ją na ścieżkę zawierającą wyeksportowane dane tabel ('/var/lib/mysql-files/' w powyższym przykładzie).

Więcej informacji znajdziesz w dokumentacji MySQL lub dokumentacji MariaDB.

`max_execution_time` (w MySQL) lub `max_statement_time` (w MariaDB) musi zostać wyłączona przed eksportem danych, aby uniknąć przekroczenia limitu czasu podczas eksportu.

Dla MySQL:

```
SET @@max_execution_time=0;
```

Dla MariaDB:

```
SET @@max_statement_time=0;
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TER
LOAD DATA INFILE '/var/lib/mysql-files/history.csv' IGNORE INTO TABLE history FIELDS TERMINATED BY ',' ESC
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history_uint.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINE
LOAD DATA INFILE '/var/lib/mysql-files/history_uint.csv' IGNORE INTO TABLE history_uint FIELDS TERMINATED

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_str.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
LOAD DATA INFILE '/var/lib/mysql-files/history_str.csv' IGNORE INTO TABLE history_str FIELDS TERMINATED BY

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_log.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
LOAD DATA INFILE '/var/lib/mysql-files/history_log.csv' IGNORE INTO TABLE history_log FIELDS TERMINATED BY

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_text.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINE
LOAD DATA INFILE '/var/lib/mysql-files/history_text.csv' IGNORE INTO TABLE history_text FIELDS TERMINATED
```

Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

PostgreSQL

Eksport i import należy wykonać w tmux/screen, aby upewnić się, że sesja nie zostanie przerwana. W przypadku instalacji z TimescaleDB pomiń tę sekcję i przejdź do PostgreSQL + TimescaleDB.

Zobacz także: Ważne uwagi

Aktualizacja tabel

- Zmień nazwy tabel, używając `history_upgrade_prepare.sql`:

```
sudo -u zabbix psql zabbix < /usr/share/zabbix/sql-scripts/postgresql/option-patches/history_upgrade_prepa
```

Migracja przy zatrzymanym serwerze

- Wyeksportuj bieżącą historię, zaimportuj ją do tabeli tymczasowej, a następnie wstaw dane do nowych tabel, ignorując duplikaty:

```
INSERT INTO history SELECT * FROM history_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_uint SELECT * FROM history_uint_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_str SELECT * FROM history_str_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_log SELECT * FROM history_log_old ON CONFLICT (itemid,clock,ns) DO NOTHING;

INSERT INTO history_text SELECT * FROM history_text_old ON CONFLICT (itemid,clock,ns) DO NOTHING;
```

Zobacz wskazówki dotyczące poprawy wydajności INSERT: PostgreSQL: Bulk Loading Huge Amounts of Data, Checkpoint Distance and Amount of WAL.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Migracja z działającym serwerem

- Wyeksportuj bieżącą historię, zaimportuj ją do tabeli tymczasowej, a następnie wstaw dane do nowych tabel, ignorując duplikaty:

```
\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,n

\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;
```

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

PostgreSQL + TimescaleDB

Eksport i import muszą być wykonywane w tmux/screen, aby upewnić się, że sesja nie zostanie przerwana. Serwer Zabbix powinien być wyłączony podczas aktualizacji.

Zobacz także: Important notes

- Zmień nazwy tabel, używając `history_upgrade_prepare.sql`.
  - Jeśli kompresja jest włączona (w domyślnej instalacji), uruchom skrypt z /usr/share/zabbix/sql-scripts/postgresql/tim

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    ```

  - Jeśli kompresja jest wyłączona, uruchom skrypt z /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-p

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    ```

- Uruchom skrypty migracji hypertable TimescaleDB w zależności od ustawień kompresji:
  - Jeśli kompresja jest włączona (w domyślnej instalacji), uruchom skrypty z /usr/share/zabbix/sql-scripts/postgresql/ti

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/trends_u
    ```

  - Jeśli kompresja jest wyłączona, uruchom skrypty z /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
    ```

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/trend
```

Zobacz także: Tips dotyczące poprawy wydajności INSERT.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Po migracji

Dla wszystkich baz danych, po zakończeniu migracji wykonaj następujące czynności:

- Zweryfikuj, że wszystko działa zgodnie z oczekiwaniami.
- Usuń stare tabele:

```
DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;
```

- W przypadku TimescaleDB usuń również następującą starą tabelę:

```
DROP TABLE trends_old;
```

Zobacz także

- Przygotowanie tabeli auditlog do partycjonowania

4 Przygotowanie tabeli auditlog do partycjonowania

Przegląd

Niektóre bazy danych (na przykład MySQL) wymagają, aby kolumna partycjonowania była częścią unikalnego ograniczenia tabeli. Dlatego, aby partycjonować tabelę `auditlog` według czasu, klucz główny musi zostać zmieniony z `auditid` na klucz złożony `auditid` + `clock`.

Ta sekcja zawiera instrukcje dotyczące zmiany klucza głównego tabeli `auditlog`.

Attention:
Instrukcje podane na tej stronie są przeznaczone dla zaawansowanych użytkowników. Należy pamiętać, że instrukcje te mogą wymagać dostosowania do konkretnej konfiguracji. Zmiana klucza głównego może być również niezgodna z przyszłymi poprawkami aktualizacyjnymi, dlatego może być konieczna ręczna obsługa przyszłych aktualizacji.

 Zmiana klucza głównego może być operacją wymagającą dużych zasobów i zająć dużo czasu, w zależności od rozmiaru tabeli `auditlog`. Zaleca się zatrzymanie serwera Zabbix i przełączenie frontend Zabbix w tryb konserwacji na czas wprowadzania zmiany. Jeśli jednak jest to absolutnie konieczne, istnieje sposób na zmianę klucza głównego bez przestoju (patrz poniżej).

Partycjonowanie tabeli `auditlog` może poprawić na przykład housekeeping w dużych środowiskach. Chociaż housekeeping Zabbix obecnie nie może korzystać z tabel partycjonowanych (z wyjątkiem TimescaleDB), można wyłączyć housekeeping Zabbix i usuwać partycje za pomocą skryptów.

Od Zabbix 7.0 tabela `auditlog` dla TimescaleDB została przekształcona w hypertable, co pozwala housekeeperowi usuwać dane całymi chunkami. Aby zaktualizować istniejącą tabelę `auditlog` do hypertable, zobacz Upgrading TimescaleDB schema.

MySQL

Ważne uwagi dotyczące przebudowy indeksów

MySQL automatycznie przebudowuje indeksy dla klucza głównego podczas operacji `ALTER TABLE`. Jednak zdecydowanie zaleca się również ręczną przebudowę indeksów za pomocą polecenia `OPTIMIZE TABLE`, aby zapewnić optymalną wydajność bazy danych.

Przebudowa indeksów może tymczasowo wymagać dodatkowej ilości miejsca na dysku równej rozmiarowi samej tabeli. Aby uzyskać bieżący rozmiar danych i indeksów, można wykonać następujące polecenia:

```
ANALYZE TABLE auditlog;
SHOW TABLE STATUS LIKE 'auditlog';
```

Jeśli dostępne miejsce na dysku jest problemem, postępuj zgodnie z instrukcjami w sekcji Zmiana klucza głównego bez przestoju. Dostępne są również inne opcje:

- Zwiększenie parametru MySQL `sort_buffer_size` może pomóc zmniejszyć zużycie miejsca na dysku podczas ręcznej przebudowy indeksów. Należy jednak pamiętać, że modyfikacja tej zmiennej może wpłynąć na ogólne zużycie pamięci przez bazę danych.
- Rozważ zwolnienie miejsca poprzez usunięcie potencjalnie niepotrzebnych danych.
- Rozważ zmniejszenie parametru *Okres przechowywania danych* housekeeper przed uruchomieniem housekeepera.

Zmiana klucza głównego z przestojem

1. Usuń bieżący klucz główny tabeli `auditlog` i dodaj nowy klucz główny.

```
ALTER TABLE auditlog DROP PRIMARY KEY, ADD PRIMARY KEY (auditid, clock);
```

2. Odbuduj indeksy (opcjonalnie, ale zdecydowanie zalecane; zobacz Ważne uwagi dotyczące odbudowy indeksów).

```
OPTIMIZE TABLE auditlog;
```

Zmiana klucza głównego bez przestoju

Ręczna metoda zmiany klucza głównego jest opisana tutaj. Alternatywnie można użyć zestawu narzędzi pt-online-schema-change firmy Percona. Ten zestaw narzędzi automatycznie wykonuje następujące qnnonnności, jednocześnie minimalizując ilość miejsca używanego podczas zmiany tabeli `auditlog`.

1. Utwórz nową tabelę z nowym kluczem głównym i indeksami.

```
CREATE TABLE `auditlog_new` (
  `auditid`             varchar(25)                             NOT NULL,
  `userid`              bigint unsigned                         NULL,
  `username`            varchar(100)    DEFAULT ''              NOT NULL,
  `clock`               integer         DEFAULT '0'             NOT NULL,
  `ip`                  varchar(39)     DEFAULT ''              NOT NULL,
  `action`              integer         DEFAULT '0'             NOT NULL,
  `resourcetype`        integer         DEFAULT '0'             NOT NULL,
  `resourceid`          bigint unsigned                         NULL,
  `resource_cuid`       varchar(25)                             NULL,
  `resourcename`        varchar(255)    DEFAULT ''              NOT NULL,
  `recordsetid`         varchar(25)                             NOT NULL,
  `details`             longtext                                NOT NULL,
  PRIMARY KEY (auditid,clock)
) ENGINE=InnoDB;
CREATE INDEX `auditlog_1` ON `auditlog_new` (`userid`,`clock`);
CREATE INDEX `auditlog_2` ON `auditlog_new` (`clock`);
CREATE INDEX `auditlog_3` ON `auditlog_new` (`resourcetype`,`resourceid`);
```

2. Zamień tabele miejscami.

```
RENAME TABLE auditlog TO auditlog_old, auditlog_new TO auditlog;
```

3. Skopiuj dane ze starej tabeli do nowej tabeli.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Można to wykonać w częściach (za pomocą wielu instrukcji `INSERT INTO` z klauzulami `WHERE clock`, w razie potrzeby), aby uniknąć nadmiernego zużycia zasobów.

4. Usuń starą tabelę.

```
DROP TABLE auditlog_old;
```

PostgreSQL

Ważne uwagi dotyczące przebudowy indeksów

PostgreSQL automatycznie przebudowuje indeksy dla klucza głównego podczas operacji `ALTER TABLE`. Jednak zdecydowanie zaleca się również ręczną przebudowę indeksów za pomocą polecenia `REINDEX TABLE CONCURRENTLY`, aby zapewnić optymalną wydajność bazy danych.

Przebudowa indeksów może tymczasowo wymagać nawet trzykrotności miejsca na dysku aktualnie zajmowanego przez indeksy. Aby sprawdzić bieżący rozmiar indeksów, możesz wykonać następujące zapytanie:

```
SELECT pg_size_pretty(pg_indexes_size('auditlog'));
```

Jeśli dostępne miejsce na dysku jest problemem, postępuj zgodnie z instrukcjami dotyczącymi zmiany klucza głównego bez przestoju. Dostępne są również inne opcje:

- Zwiększenie parametru PostgreSQL `maintenance_work_mem` może pomóc zmniejszyć zużycie miejsca na dysku podczas ręcznej przebudowy indeksów. Jednak modyfikacja tej zmiennej może wpłynąć na ogólne zużycie pamięci przez bazę danych.
- Jeśli masz inny dysk lub tablespace z większą ilością dostępnego miejsca, możesz rozważyć zmianę tymczasowej lokalizacji przechowywania dla przebudowy indeksu. Możesz ustawić parametr PostgreSQL `temp_tablespaces`, aby wskazać inny tablespace dla obiektów tymczasowych.
- Rozważ zwolnienie miejsca przez usunięcie potencjalnie niepotrzebnych danych.
- Rozważ zmniejszenie parametru *Okres przechowywania danych* housekeeper przed uruchomieniem housekeeper.

Zmiana klucza głównego z przestojem

1. Usuń bieżący klucz główny tabeli `auditlog` i dodaj nowy klucz główny.

```
ALTER TABLE auditlog DROP CONSTRAINT auditlog_pkey;
ALTER TABLE auditlog ADD PRIMARY KEY (auditid,clock);
```

2. Odbuduj indeksy (opcjonalnie, ale zdecydowanie zalecane; zobacz Ważne uwagi dotyczące odbudowy indeksów).

```
REINDEX TABLE CONCURRENTLY auditlog;
```

Zmiana klucza głównego bez przestoju

Tutaj opisano ręczną metodę zmiany klucza głównego. Alternatywnie można rozważyć rozszerzenie `pg_repack` do utworzenia nowej tabeli, skopiowania danych i zamiany tabel.

1. Utwórz nową tabelę z nowym kluczem głównym i utwórz indeksy.

```
CREATE TABLE auditlog_new (
  auditid              varchar(25)                          NOT NULL,
  userid               bigint                               NULL,
  username             varchar(100)    DEFAULT ''           NOT NULL,
  clock                integer         DEFAULT '0'          NOT NULL,
  ip                   varchar(39)     DEFAULT ''           NOT NULL,
  action               integer         DEFAULT '0'          NOT NULL,
  resourcetype         integer         DEFAULT '0'          NOT NULL,
  resourceid           bigint                               NULL,
  resource_cuid        varchar(25)                          NULL,
  resourcename         varchar(255)    DEFAULT ''           NOT NULL,
  recordsetid          varchar(25)                          NOT NULL,
  details              text            DEFAULT ''           NOT NULL,
  PRIMARY KEY (auditid,clock)
);
CREATE INDEX auditlog_new_1 ON auditlog_new (userid,clock);
CREATE INDEX auditlog_new_2 ON auditlog_new (clock);
CREATE INDEX auditlog_new_3 ON auditlog_new (resourcetype,resourceid);
```

2. Zamień tabele.

```
ALTER TABLE auditlog RENAME TO auditlog_old;
ALTER TABLE auditlog_new RENAME TO auditlog;
```

3. Skopiuj dane ze starej tabeli do nowej tabeli.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Można to wykonać partiami (wiele instrukcji `INSERT INTO` z klauzulami `WHERE clock` w razie potrzeby), aby uniknąć nadmiernego zużycia zasobów.

4. Usuń starą tabelę.

```
DROP TABLE auditlog_old;
```

Zobacz także

- Aktualizacja bazy danych do kluczy głównych

5 Bezpieczne połączenie z bazą danych

Przegląd

Ta sekcja zawiera kroki konfiguracji Zabbix oraz przykłady konfiguracji dla bezpiecznych połączeń TLS pomiędzy:

Baza danych	Komponenty Zabbix
MySQL	Zabbix frontend, Zabbix serwer, Zabbix proxy
PostgreSQL	Zabbix frontend, Zabbix serwer, Zabbix proxy

Aby skonfigurować szyfrowanie połączeń w ramach systemu DBMS, szczegółowe informacje można znaleźć w oficjalnej dokumentacji producenta:

- MySQL: źródłowe i replikowane serwery baz danych.
- MySQL: serwery baz danych z replikacją grupową itp.
- PostgreSQL opcje szyfrowania.

Wszystkie przykłady opierają się na wydaniach GA MySQL CE (8.0) i PgSQL (13), dostępnych w oficjalnych repozytoriach dla CentOS 8.

Wymagania

Do skonfigurowania szyfrowania wymagane są następujące elementy:

- System operacyjny wspierany przez producenta, z OpenSSL >=1.1.X lub rozwiązaniem alternatywnym.

Note:
Zaleca się unikanie systemów operacyjnych, które osiągnęły status end-of-life, szczególnie w przypadku nowych instalacji

- Silnik bazy danych (RDBMS) zainstalowany i utrzymywany z oficjalnego repozytorium dostarczanego przez producenta. Systemy operacyjne są często dostarczane z nieaktualnymi wersjami oprogramowania bazodanowego, dla których obsługa szyfrowania nie jest zaimplementowana, na przykład systemy oparte na RHEL 7 oraz PostgreSQL 9.2, MariaDB 5.5 bez obsługi szyfrowania.

Terminologia

Ustawienie tej opcji wymusza użycie połączenia TLS z bazą danych przez serwer/proxy Zabbix oraz frontend do bazy danych:

- `required` — połącz przy użyciu TLS jako trybu transportu bez sprawdzania tożsamości
- `verify_ca` — połącz przy użyciu TLS i zweryfikuj certyfikat
- `verify_full` — połącz przy użyciu TLS, zweryfikuj certyfikat oraz sprawdź, czy tożsamość bazy danych (CN) określona przez DBHost jest zgodna z jej certyfikatem

Konfiguracja Zabbix

Frontend do bazy danych

Bezpieczne połączenie z bazą danych można skonfigurować podczas instalacji frontend:

- Zaznacz pole wyboru *Database TLS encryption* w kroku Configure DB connection, aby włączyć szyfrowanie transportu.
- Zaznacz pole wyboru *Verify database certificate*, które pojawia się po zaznaczeniu pola *TLS encryption*, aby włączyć szyfrowanie z użyciem certyfikatów.

Note:
Dla MySQL pole wyboru *Database TLS encryption* jest wyłączone, jeśli w polu *Database host* ustawiono localhost, ponieważ połączenie korzystające z pliku gniazda (w systemach Unix) lub pamięci współdzielonej (w systemie Windows) nie może być szyfrowane.
 Dla PostgreSQL pole wyboru *TLS encryption* jest wyłączone, jeśli wartość pola *Database host* zaczyna się od ukośnika lub pole jest puste.

Następujące parametry stają się dostępne w trybie szyfrowania TLS z certyfikatami (jeśli zaznaczone są oba pola wyboru):

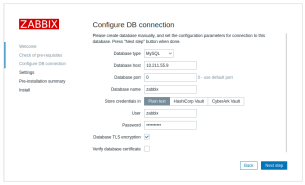
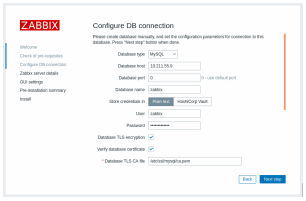
Parameter	Description
Database TLS CA file	Podaj pełną ścieżkę do prawidłowego pliku urzędu certyfikacji TLS (CA).
Database TLS key file	Podaj pełną ścieżkę do prawidłowego pliku klucza TLS.
Database TLS certificate file	Podaj pełną ścieżkę do prawidłowego pliku certyfikatu TLS.

Parameter	Description
Database host verification	Zaznacz to pole wyboru, aby włączyć weryfikację hosta. Wyłączone dla MYSQL, ponieważ biblioteka PHP MySQL nie pozwala pominąć kroku walidacji certyfikatu peera.
Database TLS cipher list	Podaj niestandardową listę prawidłowych szyfrów. Format listy szyfrów musi być zgodny ze standardem OpenSSL. Dostępne tylko dla MySQL.

Attention:
Parametry TLS muszą wskazywać prawidłowe pliki. Jeśli wskazują na nieistniejące lub nieprawidłowe pliki, spowoduje to błąd autoryzacji.
 Jeśli pliki certyfikatów mają uprawnienia do zapisu, frontend generuje ostrzeżenie w raporcie System information: "TLS certificate files must be read-only." (wyświetlane tylko wtedy, gdy użytkownik PHP jest właścicielem certyfikatu).
Certyfikaty chronione hasłami nie są obsługiwane.

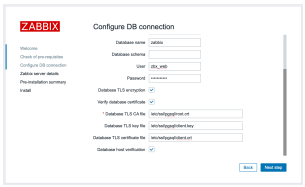
Przypadki użycia

Zabbix frontend używa interfejsu GUI do definiowania możliwych opcji: `required`, `verify_ca`, `verify_full`. Wymagane opcje należy określić w kroku kreatora instalacji *Configure DB connections*. Opcje te są mapowane do pliku konfiguracyjnego (zabbix.conf.php) w następujący sposób:

Ustawienia GUI	Plik konfiguracyjny	Opis	Wynik
	... // Used for TLS connection. $DB['ENCRYPTION'] = true; $DB['KEY_FILE'] = ''; $DB['CERT_FILE'] = ''; $DB['CA_FILE'] = ''; $DB['VERIFY_HOST'] = false; $DB['CIPHER_LIST'] = ''; ...	Zaznacz *Database TLS encryption* Pozostaw *Verify database certificate* niezaznaczone	Włącza tryb `required`.
	... $DB['ENCRYPTION'] = true; $DB['KEY_FILE'] = ''; $DB['CERT_FILE'] = ''; $DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem'; $DB['VERIFY_HOST'] = false; $DB['CIPHER_LIST'] = ''; ...	1. Zaznacz *Database TLS encryption* i *Verify database certificate* 2. Podaj ścieżkę do *Database TLS CA file*	Włącza tryb `verify_ca`.

Ustawienia GUI	Plik konfiguracyjny	Opis	Wynik

	...
	// 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'] = '';
	...
		1. Zaznacz *Database TLS encryption* i *Verify database certificate*
		2. Podaj ścieżkę do *Database TLS key file*
		3. Podaj ścieżkę do *Database TLS CA file*
		4. Podaj ścieżkę do *Database TLS certificate file*
		5. Podaj *Database TLS cipher list* (opcjonalnie)
			Włącza tryb `verify_full` dla MySQL.

	... $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'] = ' '; ...	1. Zaznacz *Database TLS encryption* i *Verify database certificate* 2. Podaj ścieżkę do *Database TLS key file* 3. Podaj ścieżkę do *Database TLS CA file* 4. Podaj ścieżkę do *Database TLS certificate file* 5. Zaznacz *Database host verification*	Włącza tryb `verify_full` dla PostgreSQL.

Zobacz także: Przykłady konfiguracji szyfrowania dla MySQL, Przykłady konfiguracji szyfrowania dla PostgreSQL.

Konfiguracja serwera/proxy Zabbix

Bezpieczne połączenia z bazą danych można skonfigurować za pomocą odpowiednich parametrów w pliku konfiguracyjnym Zabbix serwer i/lub proxy.

Konfiguracja	Wynik
Brak	Połączenie z bazą danych bez szyfrowania.
1. Ustaw `DBTLSConnect=required`	Serwer/proxy nawiązują połączenie TLS z bazą danych. Nieszyfrowane połączenie nie jest dozwolone.
1. Ustaw `DBTLSConnect=verify_ca` 2. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS	Serwer/proxy nawiązują połączenie TLS z bazą danych po zweryfikowaniu certyfikatu bazy danych.
1. Ustaw `DBTLSConnect=verify_full` 2. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS	Serwer/proxy nawiązują połączenie TLS z bazą danych po zweryfikowaniu certyfikatu bazy danych oraz tożsamości hosta bazy danych.
1. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS 2. Ustaw `DBTLSCertFile` - określ plik certyfikatu klucza publicznego klienta 3. Ustaw `DBTLSKeyFile` - określ plik klucza prywatnego klienta	Serwer/proxy przedstawiają certyfikat klienta podczas łączenia z bazą danych.
1. Ustaw `DBTLSCipher` - lista szyfrów, które klient dopuszcza dla połączeń używających protokołów TLS do wersji TLS 1.2 włącznie lub `DBTLSCipher13` - lista szyfrów, które klient dopuszcza dla połączeń używających protokołu TLS 1.3	(MySQL) Połączenie TLS jest nawiązywane z użyciem szyfru z podanej listy. (PostgreSQL) Ustawienie tej opcji będzie traktowane jako błąd.

1 Konfiguracja szyfrowania MySQL

Omówienie

Ta sekcja zawiera kilka przykładów konfiguracji szyfrowania dla CentOS 8.2 i MySQL 8.0.30 i może służyć jako przewodnik szybkiego startu do szyfrowania połączenia z bazą danych.

Attention:
Jeśli host MySQL jest ustawiony na localhost, opcje szyfrowania nie będą dostępne. W takim przypadku połączenie między frontend Zabbix a bazą danych korzysta z pliku gniazda (w systemach Unix) lub pamięci współdzielonej (w systemach Windows) i nie może być szyfrowane.

Note:
Lista kombinacji szyfrowania nie ogranicza się do tych wymienionych na tej stronie. Dostępnych jest znacznie więcej kombinacji.

Wymagania wstępne

Zainstaluj bazę danych MySQL z oficjalnego repozytorium.

Szczegółowe informacje o korzystaniu z repozytorium MySQL znajdują się w dokumentacji MySQL.

Serwer MySQL jest gotowy do akceptowania bezpiecznych połączeń z użyciem certyfikatu z podpisem własnym.

Aby sprawdzić, którzy użytkownicy korzystają z szyfrowanego połączenia, uruchom następujące zapytanie (Performance Schema musi być włączone):

```
SELECT sbt.variable_value AS tls_version, t2.variable_value AS cipher, processlist_user AS user, processli
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;
```

Szyfrowanie wyłącznie transportu

Konfiguracja MySQL

Nowoczesne wersje bazy danych są gotowe do użycia od razu w trybie `required` trybu szyfrowania. Po początkowej konfiguracji i uruchomieniu zostanie utworzony certyfikat po stronie serwera.

Utwórz użytkowników i role dla głównych komponentów:

Attention:
Dla wersji MySQL 8.4+ należy używać caching_sha2_password zamiast mysql_native_password.

```
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'@'%';
```

Należy pamiętać, że protokół X.509 nie jest używany do weryfikacji tożsamości, ale użytkownik jest skonfigurowany tak, aby używał wyłącznie szyfrowanych połączeń. Więcej informacji o konfiguracji użytkowników można znaleźć w dokumentacji MySQL.

Uruchom, aby sprawdzić połączenie (połączenie socket nie może być użyte do testowania bezpiecznych połączeń):

```
mysql -u zbx_srv -p -h 10.211.55.9 --ssl-mode=REQUIRED
```

Sprawdź bieżący stan i dostępne zestawy szyfrów:

```
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@bfdb.local
SSL: Cipher in use is TLS_AES_256_GCM_SHA384

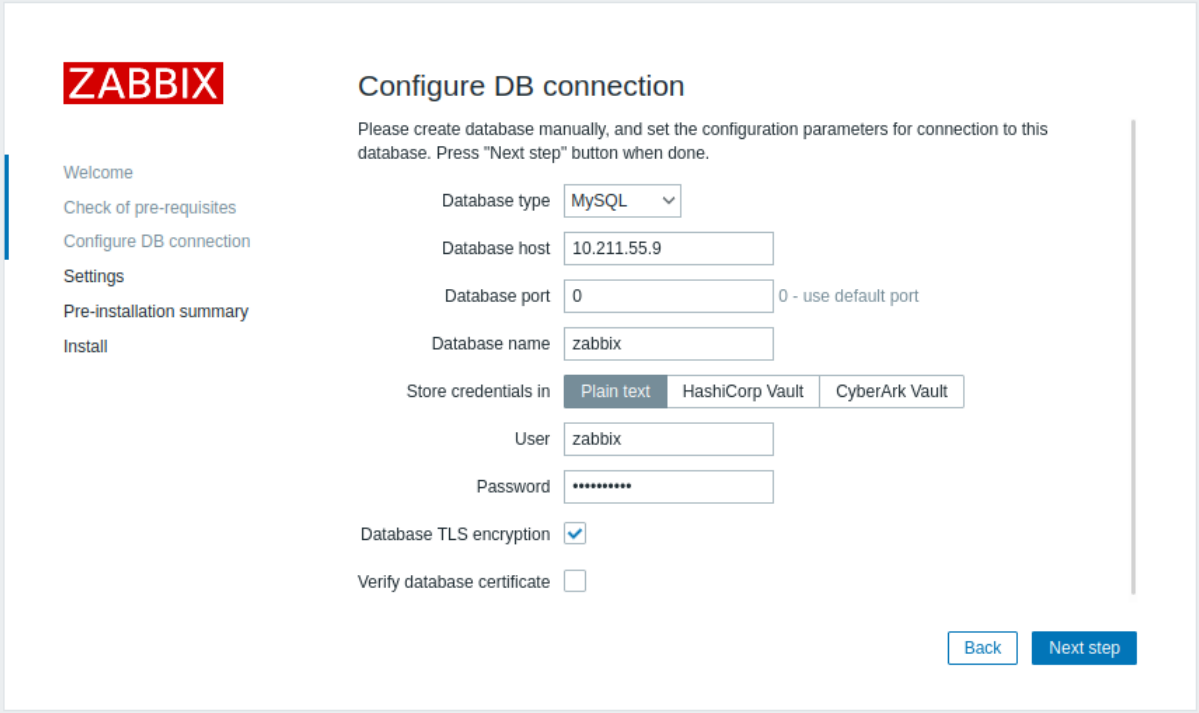

mysql> SHOW SESSION STATUS LIKE 'Ssl_cipher_list'\G;
*************************** 1. row ***************************
Variable_name: Ssl_cipher_list
Value: TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:E
1 row in set (0.00 sec)

ERROR:
No query specified
```

Frontend

Aby włączyć szyfrowanie tylko transportu dla połączeń między frontend a bazą danych:

- Zaznacz *Database TLS encryption*
- Pozostaw *Verify database certificate* niezaznaczone



Serwer

Aby włączyć szyfrowanie wyłącznie transportu dla połączeń między serwerem a bazą danych, skonfiguruj */etc/zabbix/zabbix_server.conf*:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Szyfrowanie z weryfikacją urzędu certyfikacji

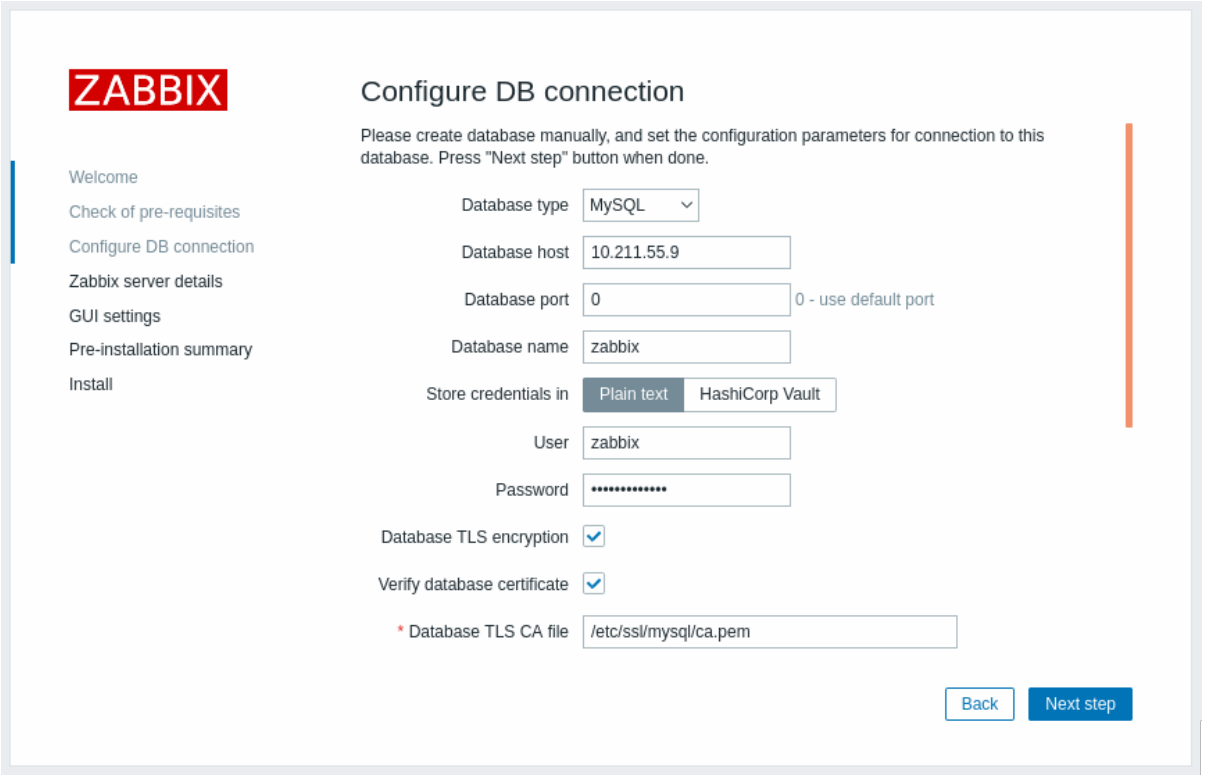
Skopiuj wymagany urząd certyfikacji MySQL na serwer frontend Zabbix i ustaw odpowiednie uprawnienia, aby serwer WWW mógł odczytać ten plik.

Note:
Ten tryb nie działa w RHEL 7 z powodu starszych bibliotek MySQL.

Frontend

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między frontend Zabbix a bazą danych:

- Zaznacz *Database TLS encryption* i *Verify database certificate*
- Określ ścieżkę do pliku Database TLS CA



Alternatywnie można to ustawić w pliku */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'] = '';
...
```

Aby rozwiązać problemy, użyj narzędzia wiersza poleceń, aby sprawdzić, czy połączenie jest możliwe dla wymaganego użytkownika:

```
    mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=REQUIRED --ssl-ca=/var/lib/mysql/ca.pem
```

Serwer

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między serwerem Zabbix a bazą danych, skonfiguruj */etc/zabbix/zabbix_server.c*

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/mysql/ca.pem
...
```

Szyfrowanie z pełną weryfikacją

Konfiguracja MySQL

Ustaw opcję konfiguracji serwera MySQL CE (*/etc/my.cnf.d/server-tls.cnf*) na:

```
[mysqld]
...
##### w tym przykładzie klucze znajdują się w katalogu datadir MySQL CE
ssl_ca=ca.pem
ssl_cert=server-cert.pem
ssl_key=server-key.pem

require_secure_transport=ON
tls_version=TLSv1.3
...
```

Klucze dla serwera i klienta MySQL CE (frontend Zabbix) należy utworzyć ręcznie zgodnie z dokumentacją MySQL CE: Creating SSL and RSA certificates and keys using MySQL lubCreating SSL certificates and keys using openssl

Attention:
Certyfikat serwera MySQL powinien zawierać pole Common Name ustawione na nazwę FQDN, ponieważ frontend Zabbix będzie używać nazwy DNS do komunikacji z bazą danych lub adresu IP hosta bazy danych.

Utwórz użytkownika MySQL:

Attention:
Dla wersji MySQL 8.4+ należy używać `caching_sha2_password` zamiast `mysql_native_password`.

```
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;
```

Sprawdź, czy można zalogować się przy użyciu tego użytkownika:

```
mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=VERIFY_IDENTITY --ssl-ca=/var/lib/mysql/ca.pem --ssl-cert=/v
```

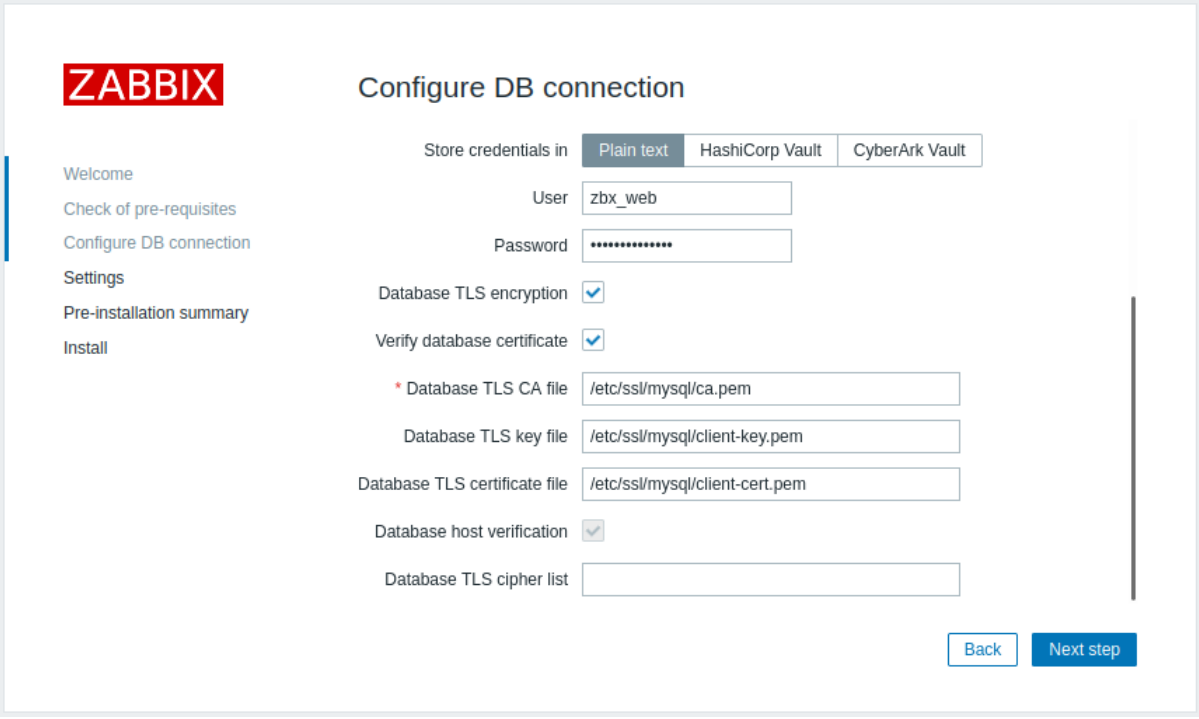
Frontend

Aby włączyć szyfrowanie z pełną weryfikacją dla połączeń między frontendem Zabbixa a bazą danych:

- Zaznacz *Database TLS encryption* oraz *Verify database certificate*
- Podaj ścieżkę do *Database TLS key file*
- Podaj ścieżkę do *Database TLS CA file*
- Podaj ścieżkę do *Database TLS certificate file*

Zwróć uwagę, że *Database host verification* jest zaznaczone i wyszarzone - tego kroku nie można pominąć w przypadku MySQL.

Warning:
Jeśli pole *Database TLS cipher list* pozostanie puste, zostaną włączone wspólne szyfry dozwolone zarówno przez frontend (klienta), jak i serwer. Alternatywnie szyfry można ustawić jawnie, zgodnie z wymaganiami konfiguracji szyfrów.



Alternatywnie można to ustawić w */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_1
...
// 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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z pełną weryfikacją dla połączeń między serwerem Zabbix a bazą danych, skonfiguruj */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 Konfiguracja szyfrowania PostgreSQL

Przegląd

Ta sekcja zawiera kilka przykładów konfiguracji szyfrowania dla CentOS 8.2 i PostgreSQL 13.

> **Note:**
> Połączenie między frontend Zabbix a PostgreSQL nie może być szyfrowane (parametry w GUI są wyłączone), jeśli wartość pola *Database host* zaczyna się od ukośnika lub pole jest puste.

Wymagania wstępne

Zainstaluj bazę danych PostgreSQL, korzystając z oficjalnego repozytorium.

PostgreSQL nie jest domyślnie skonfigurowany do akceptowania połączeń TLS. Postępuj zgodnie z instrukcjami w dokumentacji PostgreSQL dotyczącymi przygotowania certyfikatów z użyciem postgresql.conf, a także kontroli dostępu użytkowników za pośrednictwem ph_hba.conf.

Domyślnie gniazdo PostgreSQL jest powiązane z localhost; aby umożliwić zdalne połączenia sieciowe, należy włączyć nasłuchiwanie na rzeczywistym interfejsie sieciowym.

Ustawienia PostgreSQL dla wszystkich trybów mogą wyglądać następująco:

/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'
...
```

Aby skonfigurować kontrolę dostępu, dostosuj */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

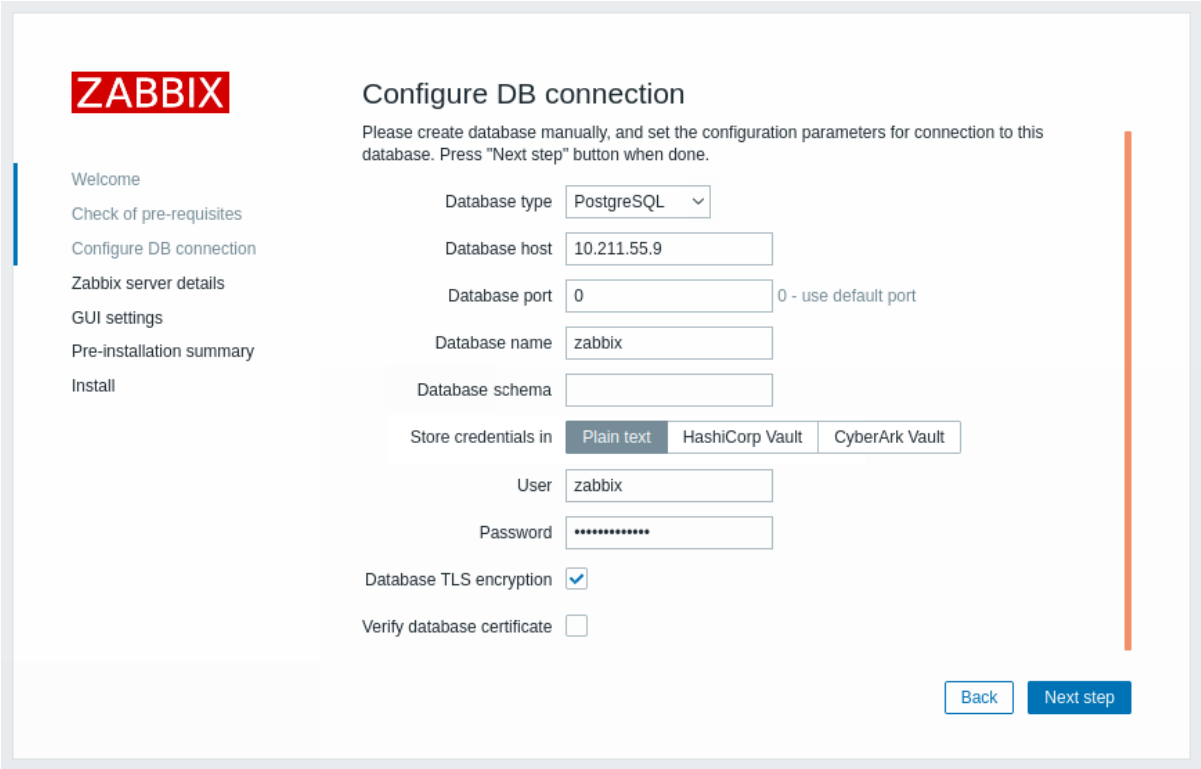
### verify full
hostssl all all 0.0.0.0/0 md5 clientcert=verify-full
...
```

Szyfrowanie wyłącznie transportu

Frontend

Aby włączyć szyfrowanie tylko transportu dla połączeń między frontend a bazą danych:

- Zaznacz *Database TLS encryption*
- Pozostaw *Verify database certificate* niezaznaczone



Serwer

Aby włączyć szyfrowanie wyłącznie transportu dla połączeń między serwerem a bazą danych, skonfiguruj plik */etc/zabbix/zabbix_server.conf*:

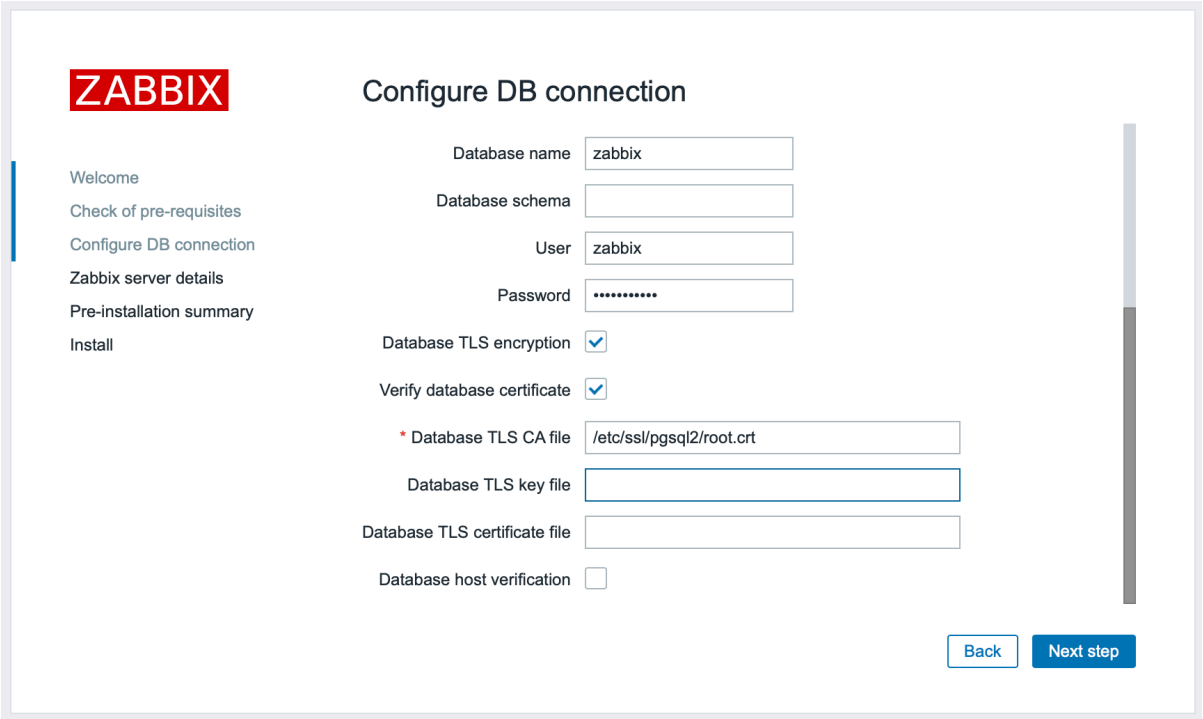
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Szyfrowanie z weryfikacją urzędu certyfikacji

Frontend

Aby włączyć szyfrowanie z weryfikacją urzędu certyfikacji dla połączeń między frontendem Zabbix a bazą danych:

- Zaznacz *Database TLS encryption* i *Verify database certificate*
- Określ ścieżkę do *Database TLS CA file*



Alternatywnie można to ustawić w */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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między serwer Zabbix a bazą danych, skonfiguruj */etc/zabbix/zabbix_server.con*

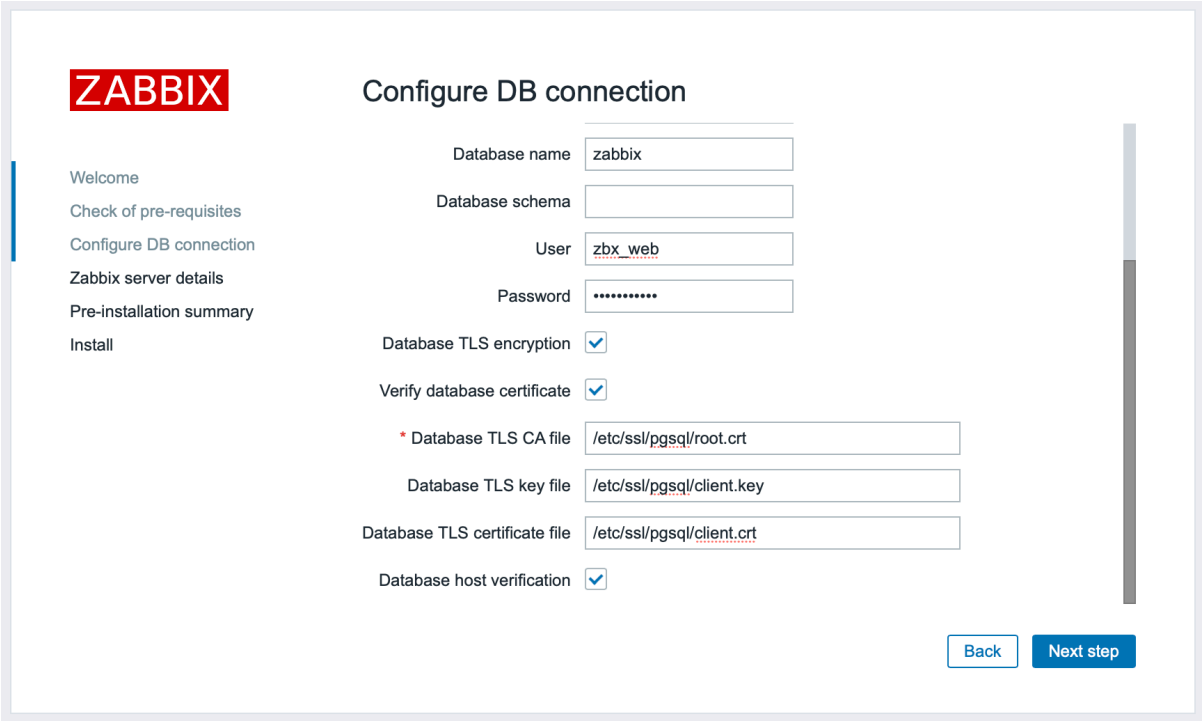
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/pgsql/root.crt
...
```

Szyfrowanie z pełną weryfikacją

Frontend

Aby włączyć szyfrowanie z użyciem certyfikatu oraz weryfikację tożsamości hosta bazy danych dla połączeń między frontend a bazą danych:

- Zaznacz *Database TLS encryption* i *Verify database certificate*
- Określ ścieżkę do *Database TLS key file*
- Określ ścieżkę do *Database TLS CA file*
- Określ ścieżkę do *Database TLS certificate file*
- Zaznacz *Database host verification*



Alternatywnie można to ustawić w */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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z użyciem certyfikatu oraz weryfikację tożsamości hosta bazy danych dla połączeń między serwerem Zabbix a bazą danych, skonfiguruj */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
...
```

6 Bezpieczne połączenie z frontendem

Przegląd

Ta sekcja zawiera kroki konfiguracji Zabbix oraz przykłady konfiguracji dla bezpiecznych połączeń TLS między frontend Zabbix a serwer Zabbix.

Konfiguracja

Domyślnie komunikacja między frontendem Zabbix a serwerem Zabbix nie jest szyfrowana. Aby zwiększyć bezpieczeństwo, włącz TLS po obu stronach. Poniżej znajduje się przykład najprostszego sposobu, aby to zrobić.

1. Wygeneruj certyfikaty i klucze.

Utwórz katalog roboczy:

```
sudo mkdir -p /etc/zabbix/ssl && cd /etc/zabbix/ssl
```

Utwórz certyfikat CA (dostosuj wartość `MyZabbixCA`, aby odpowiadała rzeczywistej nazwie pospolitej):

```
sudo openssl genrsa -out ca.key 4096
sudo openssl req -new -x509 -days 3650 -key ca.key -out ca.crt -subj "/CN=MyZabbixCA/"
```

Wygeneruj klucz prywatny i certyfikat dla serwera Zabbix (dostosuj wartość `zabbix-server.example.com`, aby odpowiadała rzeczywistej nazwie pospolitej):

```
sudo openssl genrsa -out server.key 2048
sudo openssl req -new -key server.key -out server.csr -subj "/CN=zabbix-server.example.com/"
sudo openssl x509 -req -days 365 -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial -sha256 -out serv
```

Wygeneruj klucz prywatny i certyfikat dla frontendu Zabbix (dostosuj wartość `zabbix-frontend-node`, aby odpowiadała rzeczywistej nazwie pospolitej):

```
sudo openssl genrsa -out frontend.key 2048
sudo openssl req -new -key frontend.key -out frontend.csr -subj "/CN=zabbix-frontend-node/"
sudo openssl x509 -req -days 365 -in frontend.csr -CA ca.crt -CAkey ca.key -CAcreateserial -sha256 -out fr
```

2. Ustaw prawidłowe uprawnienia.

Dla serwera Zabbix (dostosuj właściciela/grupę zgodnie z użytkownikiem demona serwera Zabbix w twojej dystrybucji):

```
sudo chown root:zabbix /etc/zabbix/ssl/server.{crt,key} /etc/zabbix/ssl/ca.crt
sudo chmod 640 /etc/zabbix/ssl/server.key
sudo chmod 644 /etc/zabbix/ssl/server.crt /etc/zabbix/ssl/ca.crt
```

Dla frontendu (dostosuj właściciela/grupę zgodnie z użytkownikiem serwera WWW w twojej dystrybucji):

```
sudo chown root:www-data /etc/zabbix/ssl/frontend.{crt,key}
sudo chmod 640 /etc/zabbix/ssl/frontend.key
sudo chmod 644 /etc/zabbix/ssl/frontend.crt
```

3. Skonfiguruj serwer Zabbix.

W pliku `zabbix_server.conf` dodaj:

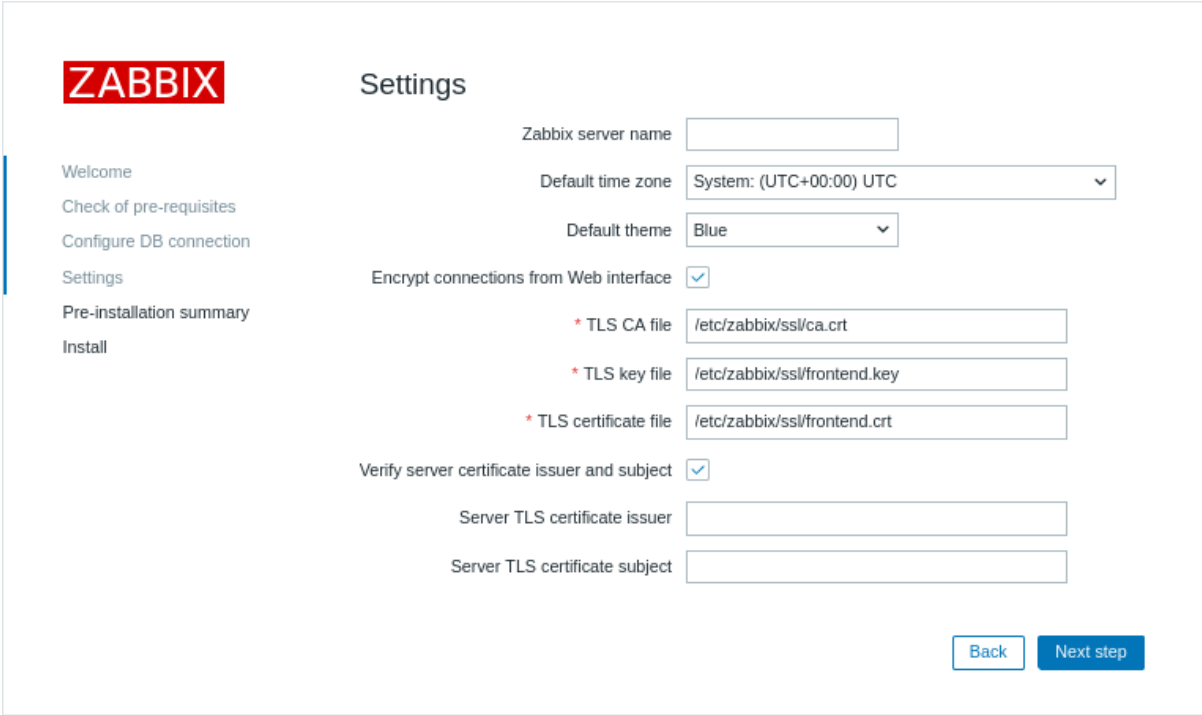
```
TLSFrontendAccept=cert
TLSCertFile=/etc/zabbix/ssl/server.crt
TLSKeyFile=/etc/zabbix/ssl/server.key
TLSCAFile=/etc/zabbix/ssl/ca.crt
#### Opcjonalnie:
#### TLSFrontendCertIssuer=/CN=MyZabbixCA/
#### TLSFrontendCertSubject=/CN=zabbix-frontend-node/
```

Następnie uruchom ponownie serwer:

```
sudo systemctl restart zabbix-server
```

4. Skonfiguruj frontend Zabbix.

Podczas instalacji interfejsu webowego włącz opcję *Encrypt connections from Web interface* (oraz opcję *Verify server certificate issuer and subject*, jeśli to konieczne) i wypełnij pola *TLS CA file*, *TLS key file*, *TLS certificate file* (oraz pola *Server TLS certificate issuer* i *Server TLS certificate subject*, jeśli to konieczne):



Parametr	Opis
TLS CA file	Podaj pełną ścieżkę do pliku certyfikatu urzędu certyfikacji (CA) używanego do weryfikacji certyfikatu serwera.
TLS key file	Podaj pełną ścieżkę do pliku klucza prywatnego klienta odpowiadającego certyfikatowi klienta.
TLS certificate file	Podaj pełną ścieżkę do pliku certyfikatu klienta, jeśli wymagana jest wzajemna autentykacja TLS.
Server TLS certificate issuer	Podaj nazwę wyróżniającą (DN) wystawcy, która ma zostać dopasowana do certyfikatu serwera.
Server TLS certificate subject	Podaj nazwę wyróżniającą (DN) podmiotu, która ma zostać dopasowana do certyfikatu serwera.

W istniejących instalacjach edytuj następujące pola w pliku `zabbix.conf.php`:

```
$ZBX_SERVER_TLS['ACTIVE'] = '1';
$ZBX_SERVER_TLS['CA_FILE'] = '/etc/zabbix/ssl/ca.crt';
$ZBX_SERVER_TLS['KEY_FILE'] = '/etc/zabbix/ssl/frontend.key';
$ZBX_SERVER_TLS['CERT_FILE'] = '/etc/zabbix/ssl/frontend.crt';
// Opcjonalnie:
// $ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = '/CN=MyZabbixCA/';
// $ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = '/CN=zabbix-server.example.com/';
```

5. Zweryfikuj szyfrowanie, potwierdzając, że we frontendzie Zabbix ani w pliku dziennika serwera Zabbix nie ma komunikatów o błędach:

```
tail -f /var/log/zabbix/zabbix_server.log
```

7 Konfiguracja TimescaleDB

Przegląd

Zabbix obsługuje TimescaleDB, rozwiązanie bazodanowe oparte na PostgreSQL, które automatycznie partycjonuje dane na fragmenty oparte na czasie, aby zapewnić wyższą wydajność przy dużej skali.

> **Warning:**
> Obecnie TimescaleDB nie jest obsługiwany przez Zabbix proxy.

Instrukcje na tej stronie można wykorzystać w następujących scenariuszach:

- Tworzenie bazy danych TimescaleDB lub migracja z istniejących tabel PostgreSQL do TimescaleDB (zobacz Konfiguracja).
- Aktualizacja schematu istniejącej bazy danych TimescaleDB podczas aktualizacji Zabbix (zobacz Aktualizacja schematu TimescaleDB).

Konfiguracja

Wymagania wstępne: Rozszerzenie TimescaleDB w obsługiwanej wersji zainstalowane na serwerze bazy danych. Instrukcje instalacji znajdują się w dokumentacji TimescaleDB.

> **Warning:**
> Przed instalacją TimescaleDB zainstaluj obsługiwaną wersję PostgreSQL z oficjalnego repozytorium PostgreSQL.

Włącz rozszerzenie TimescaleDB dla określonej bazy danych, wykonując:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;" | sudo -u postgres psql zabbix
```

Uruchomienie tego polecenia wymaga uprawnień administratora bazy danych.

> **Note:**
> Jeśli używasz schematu bazy danych innego niż 'public', musisz dodać klauzulę SCHEMA do powyższego polecenia. Np.:
 `echo "CREATE EXTENSION IF NOT EXISTS timescaledb SCHEMA yourschema CASCADE;" | sudo -u postgres psql zabbix`

Następnie uruchom skrypt `postgresql/timescaledb/schema.sql`. W przypadku nowych instalacji skrypt należy uruchomić po utworzeniu standardowej bazy danych PostgreSQL z początkowym schematem/danymi (zobacz tworzenie bazy danych).

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/schema.sql | sudo -u zabbix psql zabbix
```

> **Attention:**
> Zignoruj komunikaty ostrzegawcze informujące o nieprzestrzeganiu najlepszych praktyk podczas uruchamiania skryptu `schema.sql` w TimescaleDB w wersji 2.9.0 i nowszej. Niezależnie od tego ostrzeżenia konfiguracja zostanie ukończona pomyślnie.

Migracja istniejących danych historii, trendów i dziennika audytu może zająć dużo czasu. serwer Zabbix i frontend muszą być wyłączone na czas migracji.

Skrypt `schema.sql` ustawia następujące parametry housekeeping:

- Zastąp okres przechowywania historii pozycji
- Zastąp okres przechowywania trendów pozycji

Aby używać partycjonowanego housekeeping dla historii i trendów, obie te opcje muszą być włączone. Możliwe jest również włączenie nadpisania indywidualnie tylko dla historii albo tylko dla trendów.

Dla PostgreSQL i TimescaleDB skrypt `postgresql/timescaledb/schema.sql` ustawia dwa dodatkowe parametry:

- Włącz kompresję
- Kompresuj rekordy starsze niż 7 dni

Aby housekeeper mógł pomyślnie usuwać skompresowane dane, muszą być włączone obie opcje: *Override item history period* oraz *Override item trend period*. Jeśli nadpisywanie jest wyłączone, a tabele zawierają skompresowane fragmenty, housekeeper nie będzie usuwał danych z tych tabel, a ostrzeżenia o nieprawidłowej konfiguracji będą wyświetlane w sekcjach *Housekeeping* oraz *System information*.

Wszystkie te parametry można zmienić w *Administracja* > *Housekeeping* po instalacji.

> **Note:**
> Możesz uruchomić narzędzie timescaledb-tune dostarczane przez TimescaleDB, aby zoptymalizować parametry konfiguracji PostgreSQL w pliku `postgresql.conf`.

Aktualizacja schematu TimescaleDB

Podczas aktualizacji Zabbixa do wersji zawierającej nowe hypertabele TimescaleDB, serwer Zabbix nie konfiguruje ich automatycznie (na przykład podczas aktualizacji z Zabbix 6.4 do 7.4, ponieważ wersje 7.0.0 i 7.0.2 wprowadziły nowe hypertabele).

Aby skonfigurować nowe hypertabele TimescaleDB, wykonaj następujące kroki:

1. Uruchom serwer Zabbix; spowoduje to aktualizację istniejącej bazy danych.
2. Sprawdź w pliku dziennika serwera, czy aktualizacja bazy danych została zakończona; po zakończeniu zatrzymaj serwer Zabbix. Zwróć uwagę, że serwer zapisuje ostrzeżenie, jeśli spróbuje włączyć kompresję dla tabeli, która nie jest hypertabelą.
3. Uruchom skrypt `postgresql/timescaledb/schema.sql`; skonfiguruje on nowe hypertabele TimescaleDB. Zwróć uwagę, że od Zabbix 7.0.0 lokalizacja i nazwa skryptu zmieniły się z `postgresql/timescaledb.sql` na `postgresql/timescaledb/schema`

Attention:
Zignoruj komunikaty ostrzegawcze informujące, że podczas uruchamiania skryptu `schema.sql` w wersji TimescaleDB 2.9.0 i nowszych nie są przestrzegane najlepsze praktyki. Niezależnie od tego ostrzeżenia konfiguracja zostanie pomyślnie zakończona.

Kompresja TimescaleDB

Natywna kompresja TimescaleDB jest obsługiwana dla wszystkich tabel Zabbix, które są hypertabelami TimescaleDB. Podczas aktualizacji lub migracji do TimescaleDB początkowa kompresja dużych tabel może zająć dużo czasu.

Należy pamiętać, że kompresja jest obsługiwana w ramach licencji Timescale Community ”timescale”, natomiast nie jest obsługiwana w ramach licencji Apache 2.0 ”apache”. Jeśli Zabbix wykryje, że kompresja nie jest obsługiwana, do logu serwera Zabbix zostanie zapisany komunikat ostrzegawczy, a użytkownicy nie będą mogli włączyć kompresji we frontendzie.

Note:
Przed użyciem kompresji zaleca się użytkownikom zapoznanie się z mechanizmem kompresji w dokumentacji TimescaleDB.

Należy pamiętać, że istnieją pewne ograniczenia wynikające z kompresji, w szczególności:

- Modyfikacje skompresowanych chunków (wstawianie, usuwanie, aktualizacje) nie są dozwolone
- Zmiany schematu dla skompresowanych tabel nie są dozwolone

Ustawienia kompresji można zmienić w bloku *History, trends and audit log compression* w sekcji *Administration* > *Housekeeping* frontendu Zabbix.

Parametr	Domyślnie	Uwagi
Enable compression	Włączone	Zaznaczenie lub odznaczenie pola wyboru nie aktywuje/dezaktywuje kompresji natychmiast. Ponieważ kompresja jest obsługiwana przez Housekeeper, zmiany zaczną obowiązywać maksymalnie po 2 × `HousekeepingFrequency` godzinach (ustawianych w zabbix_server.conf) Po wyłączeniu kompresji nowe chunki, które mieszczą się w okresie kompresji, nie będą kompresowane. Jednak wszystkie wcześniej skompresowane dane pozostaną skompresowane. Aby zdekompresować wcześniej skompresowane chunki, postępuj zgodnie z instrukcjami w dokumentacji TimescaleDB.
Compress records older than	7d	W przypadku aktualizacji ze starszych wersji Zabbix z obsługą TimescaleDB kompresja nie będzie domyślnie włączona. Ten parametr nie może być mniejszy niż 7 dni. Ze względu na niezmienność skompresowanych chunków wszystkie opóźnione dane (np. dane opóźnione przez proxy), które są starsze niż ta wartość, zostaną odrzucone.

Aby uzyskać lepszą wydajność aktualizacji trendów, można obniżyć wartość ”chunk_time_interval” dla tabel `trends` i `trends_uint` z 30 dni do 7 dni lub mniej, w zależności od tego, ile pozycji korzysta z trendów. Celem tego ustawienia jest stosowanie się do najlepszych praktyk TimescaleDB oraz zapewnienie, że rozmiar chunków pozostaje w granicach zasobów dostępnych w systemie.

8 Konfiguracja Elasticsearch

::: uwaga ważna Obsługa Elasticsearch jest eksperymentalna!

:::

Zabbix obsługuje przechowywanie danych historycznych za pomocą Elasticsearch zamiast bazy danych. Użytkownicy mogą wybrać miejsce przechowywania danych historycznych pomiędzy kompatybilną bazą danych a Elasticsearch. Procedura konfiguracji opisana w tej sekcji dotyczy Elasticsearch wersji 7.X. W przypadku korzystania ze starszej lub nowszej wersji Elasticsearch niektóre funkcje mogą nie działać zgodnie z przeznaczeniem.

::: uwaga ostrzeżenie 1. Jeśli wszystkie dane historyczne są przechowywane w Elasticsearch, trendy **nie** są obliczane ani przechowywane w bazie danych. Bez obliczania i zapisywania trendów okres przechowywania historii może wymagać wydłużenia.

2. W przypadku korzystania z Elasticsearch, zapytania o zakresy pobierające wartości z bazy danych są ograniczone znacznikiem czasu okresu przechowywania danych. :::

Konfiguracja

Aby zapewnić prawidłową komunikację między wszystkimi zaangażowanymi elementami, upewnij się, że parametry pliku konfiguracyjnego serwera i pliku konfiguracyjnego front-endu są poprawnie skonfigurowane.

Serwer Zabbix i front-end

Wersja robocza pliku konfiguracyjnego serwera Zabbix z parametrami do aktualizacji:

Opcja: HistoryStorageURL

Adres URL HTTP[S] magazynu historii.

Obowiązkowe: nie

Domyślne:

HistoryStorageURL= Opcja: HistoryStorageTypes

Lista typów wartości oddzielonych przecinkami, które mają zostać wysłane do magazynu historii.

Obowiązkowe: nie

Domyślne:

HistoryStorageTypes=uint,dbl,str,log,text Przykładowe wartości parametrów do wypełnienia pliku konfiguracyjnego serwera Zabbix za pomocą:

HistoryStorageURL=http://test.elasticsearch.lan:9200 HistoryStorageTypes=str,log,text

Ta konfiguracja wymusza na serwerze Zabbix przechowywanie wartości historycznych typów liczbowych w odpowiedniej bazie danych oraz tekstowych danych historycznych w Elasticsearch.

Elasticsearch obsługuje następujące typy elementów:

uint, dbl, str, log, text

Objaśnienie obsługiwanych typów elementów:

Typ wartości elementu	Tabela bazy danych	Typ Elasticsearch
Liczba (bez znaku)	history_uint	uint
Liczba (zmiennoprzecinkowa)	history	dbl
Znak	history_str	str
Log	history_log	log
Tekst	history_text	text

Plik konfiguracyjny front-endu Zabbix (`conf/zabbix.conf.php`) z parametrami do aktualizacji:

// Adres URL Elasticsearch (może być ciągiem znaków, jeśli dla wszystkich typów używany jest ten sam adres URL). $HISTORY['url'] = ['uint' => 'http://localhost:9200', 'text' => 'http://localhost:9200']; // Typy wartości przechowywane w Elasticsearch. $HISTORY['types'] = ['uint', 'text'];

Przykładowe wartości parametrów do wypełnienia pliku konfiguracyjnego front-endu Zabbix tym:

$HISTORY['url'] = 'http://test.elasticsearch.lan:9200'; $HISTORY['types'] = ['str', 'text', 'log'];

Ta konfiguracja wymusza przechowywanie wartości historii `Text`, `Character` i `Log` w Elasticsearch.

Wymagane jest również ustawienie zmiennej $HISTORY jako globalnej w pliku `conf/zabbix.conf.php`, aby zapewnić prawidłowe działanie (zobacz `conf/zabbix.conf.php.example`, aby dowiedzieć się, jak to zrobić):

// Plik konfiguracyjny graficznego interfejsu użytkownika Zabbix. global $DB, $HISTORY;

Instalacja Elasticsearch i tworzenie mapowania

Ostatnie dwa kroki, które należy wykonać, to instalacja samego Elasticsearch i utworzenie procesu mapowania.

Aby zainstalować Elasticsearch, zapoznaj się z przewodnikiem instalacji Elasticsearch.

Note:
Mapowanie to struktura danych w Elasticsearch (podobna do tabeli w bazie danych). Mapowanie dla wszystkich typów danych historycznych jest dostępne tutaj: `database/elasticsearch/elasticsearch.map`.

Warning:
Utworzenie mapowania jest obowiązkowe. Niektóre funkcje mogą nie działać, jeśli mapowanie nie zostanie utworzone zgodnie z instrukcją.

Aby utworzyć mapowanie dla typu `text`, wyślij następujące żądanie do 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"
}
}
}
}'
```

Podobne żądanie jest wymagane do utworzenia mapowania wartości historii `Character` i Log z odpowiednią korektą typu.

Note:
Aby korzystać z Elasticsearch, zapoznaj się z Wymaganiami w celu uzyskania dodatkowych informacji.

Note:
Housekeeper nie usuwa żadnych danych z Elasticsearch.

Przechowywanie danych historycznych w wielu indeksach opartych na datach

W tej sekcji opisano dodatkowe kroki wymagane do pracy z potokami i węzłami przetwarzania.

Na początek należy utworzyć szablony dla indeksów.

Poniższy przykład przedstawia żądanie utworzenia szablonu uint:

```
curl -X PUT \
http://your-elasticsearch.here:9200/_template/uint_template \
-H 'content-type:application/json' \
-d '{
"index_patterns": [
"uint*"
],
"settings": {
"index": {
"number_of_replicas": 1,
"number_of_shards": 5
}
},
"mappings": {
"properties": {
"itemid": {
"type": "long"
},
"clock": {
"format": "epoch_second",
"type": "date"
},
"value": {
"type": "long"
}
}
}
}'
```

Aby utworzyć inne szablony, użytkownik powinien zmienić adres URL (ostatnia część to nazwa szablonu), zmienić pole `"index_patterns"` tak, aby odpowiadało nazwie indeksu oraz ustawić prawidłowe mapowanie, które można pobrać z pliku `database/elasticsearch/elasticsearch.map`.

Na przykład, poniższe polecenie może zostać użyte do utworzenia szablonu dla indeksu tekstowego:

```
curl -X PUT \
http://your-elasticsearch.here:9200/_template/text_template \
-H 'content-type:application/json' \
-d '{
"index_patterns": [
"text*"
],
"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"
}
}
}
}'
```

Jest to wymagane, aby Elasticsearch mógł ustawić prawidłowe mapowanie dla indeksów tworzonych automatycznie. Następnie wymagane jest utworzenie definicji potoku. Potok to swego rodzaju wstępne przetwarzanie danych przed umieszczeniem ich w indeksach. Do utworzenia potoku dla indeksu uint można użyć następującego polecenia:

```
curl -X PUT \
http://your-elasticsearch.here:9200/_ingest/pipeline/uint-pipeline \
-H 'content-type:application/json' \
-d '{
"description": "codzienne nazewnictwo indeksów uint",
"processors": [
{
"date_index_name": {
"field": "clock",
"date_formats": [
"UNIX"
],
"index_name_prefix": "uint-",
"date_rounding": "d"
}
}
]
}'
```

Użytkownik może zmienić parametr zaokrąglania („date_rounding”), aby ustawić konkretny okres rotacji indeksu. Aby utworzyć inne potoki, użytkownik powinien zmienić adres URL (ostatnia część to nazwa potoku) i zmienić pole „index_name_prefix”, aby odpowiadało nazwie indeksu.

Zobacz także dokumentację Elasticsearch.

Dodatkowo, przechowywanie danych historycznych w wielu indeksach opartych na dacie powinno być również włączone w nowym parametrze w konfiguracji serwera Zabbix:

Opcja: HistoryStorageDateIndex

Włącz wstępne przetwarzanie wartości historycznych w pamięci historycznej, aby przechowywać wartości w różnych indeksach w zależności od daty.

0 - wyłącz

1 - włącz

Obowiązkowe: nie

Domyślne:

HistoryStorageDateIndex=0 Rozwiązywanie problemów

Poniższe kroki mogą pomóc w rozwiązaniu problemów z konfiguracją Elasticsearch:

1. Sprawdź, czy mapowanie jest poprawne (żądanie GET do wymaganego adresu URL indeksu, np. `http://localhost:9200/uint`).
2. Sprawdź, czy shardy nie są w stanie awarii (ponowne uruchomienie Elasticsearch powinno pomóc).

3. Sprawdź konfigurację Elasticsearch. Konfiguracja powinna umożliwiać dostęp z hosta frontendowego Zabbix i hosta serwera Zabbix.
4. Sprawdź logi Elasticsearch.
5. `LogSlowQueries` może być użyte do sprawdzenia wolnych zapytań w bazie danych Elasticsearch.

Jeśli nadal występują problemy z instalacją, utwórz raport o błędzie ze wszystkimi informacjami z tej listy (mapowanie, logi błędów, konfiguracja, wersja itp.).

9 Uwagi dotyczące konfiguracji Nginx dla Zabbix w zależności od dystrybucji

RHEL

W RHEL 8, 9 i 10 Nginx jest dołączony do AppStream — nie są potrzebne żadne dodatkowe repozytoria. Wystarczy włączyć i zainstalować go za pomocą:

```
dnf module list nginx
dnf -y install nginx
```

W RHEL 7 Nginx można zainstalować z repozytorium CentOS 7 Extras (w przypadku używania CentOS) za pomocą:

```
yum -y install nginx
```

Jeśli preferujesz najnowsze kompilacje bezpośrednio od upstreamu, skonfiguruj oficjalne repozytorium NGINX w `/etc/yum.repos.d/nginx.r`

```
cat > /etc/yum.repos.d/nginx.repo << 'EOF'
[nginx-stable]
name=nginx stable repo
baseurl=https://nginx.org/packages/rhel/$releasever/$basearch/
gpgcheck=1
enabled=1
gpgkey=https://nginx.org/keys/nginx_signing.key
EOF
dnf -y install nginx
```

SLES 15

W SUSE Linux Enterprise Server 15 należy skonfigurować `php-fpm` (ścieżka do pliku konfiguracyjnego może się nieznacznie różnić w zależności od dodatku Service Pack):

```
cp /etc/php8/fpm/php-fpm.conf{.default,}
cp /etc/php8/fpm/php-fpm.d/www.conf{.default,}
sed -i 's/user = nobody/user = wwwrun/; s/group = nobody/group = www/' /etc/php8/fpm/php-fpm.d/www.conf
```

10 Uruchamianie agent jako root

Od wersji Zabbix **5.0.0** plik usługi systemd dla Zabbix agent w oficjalnych pakietach jawnie zawiera dyrektywy `User` i `Group`. Obie są ustawione na `zabbix`.

Nie ma już możliwości skonfigurowania, jako który użytkownik działa Zabbix agent, za pomocą pliku `zabbix_agentd.conf`, ponieważ agent pominie tę konfigurację i uruchomi się jako użytkownik określony w pliku usługi systemd. Aby uruchomić Zabbix agent jako root, należy wprowadzić modyfikacje opisane poniżej.

Zabbix agent

Aby zastąpić domyślnego użytkownika i grupę dla Zabbix agent, uruchom:

```
systemctl edit zabbix-agent
```

Następnie dodaj następującą zawartość:

```
[Service]
User=root
Group=root
```

Przeładuj daemony i uruchom ponownie usługę zabbix-agent:

```
systemctl daemon-reload
systemctl restart zabbix-agent
```

W przypadku **Zabbix agent** spowoduje to ponowne włączenie funkcji konfigurowania użytkownika w pliku `zabbix_agentd.conf`. Teraz należy ustawić parametry konfiguracyjne `User=root` i `AllowRoot=1` w pliku konfiguracyjnym agent.

Zabbix agent 2

Aby zastąpić domyślnego użytkownika i grupę dla Zabbix agent 2, uruchom:

```
systemctl edit zabbix-agent2
```

Następnie dodaj następującą zawartość:

```
[Service]
User=root
Group=root
```

Przeładuj demony i uruchom ponownie usługę zabbix-agent2:

```
systemctl daemon-reload
systemctl restart zabbix-agent2
```

W przypadku **Zabbix agent2** to całkowicie określa użytkownika, jako który jest uruchamiany. Nie są wymagane żadne dodatkowe modyfikacje.

11 Zabbix agent w systemie Microsoft Windows

Konfigurowanie agent

Obie generacje agentów Zabbix działają jako usługa systemu Windows. W przypadku Zabbix agent 2 w poniższych instrukcjach należy zastąpić *agentd* przez *agent2*.

Można uruchomić pojedynczą instancję agent Zabbix lub wiele instancji agent na hoście Microsoft Windows. Pojedyncza instancja może używać:

- domyślnego pliku konfiguracyjnego znajdującego się w tym samym katalogu co plik binarny agent;
- pliku konfiguracyjnego określonego w wierszu poleceń.

W przypadku wielu instancji każda instancja agent musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Przykładowy plik konfiguracyjny jest dostępny w archiwum źródłowym Zabbix jako:

- `conf/zabbix_agentd.conf` dla Zabbix agent;
- `conf/zabbix_agent2.conf` dla Zabbix agent2.

Jeśli chcesz zainstalować Zabbix agent/agent 2 dla systemu Windows jako usługę z archiwum bez jawnego określania pliku konfiguracyjnego, to przed instalacją agent:

- `conf/zabbix_agentd.conf` należy ręcznie skopiować do katalogu, w którym zostanie zainstalowany plik zabbix_agentd.exe;
- `conf/zabbix_agent2.conf` oraz katalog `conf/zabbix_agent2.d` należy ręcznie skopiować do katalogu, w którym zostanie zainstalowany plik zabbix_agent2.exe.

Szczegółowe informacje na temat konfiguracji agenta Zabbix dla systemu Windows można znaleźć w opcjach pliku konfiguracyjnego.

Parametr Hostname

Aby wykonywać aktywne kontrole na hoście, agent Zabbix musi mieć zdefiniowaną nazwę hosta. Ponadto wartość nazwy hosta ustawiona po stronie agenta powinna dokładnie odpowiadać wartości "Host name" skonfigurowanej dla hosta we frontendzie.

Wartość nazwy hosta po stronie agenta może być zdefiniowana za pomocą parametru **Hostname** lub **HostnameItem** w pliku konfiguracyjnym agenta — albo zostaną użyte wartości domyślne, jeśli którykolwiek z tych parametrów nie zostanie określony.

Domyślną wartością parametru **HostnameItem** jest wartość zwracana przez klucz agenta "system.hostname". W systemie Windows zwraca on wynik funkcji gethostname(), która odpytuje dostawców przestrzeni nazw w celu określenia lokalnej nazwy hosta. Jeśli żaden dostawca przestrzeni nazw nie odpowie, zwracana jest nazwa NetBIOS.

Domyślną wartością parametru **Hostname** jest wartość zwracana przez parametr HostnameItem. W praktyce oznacza to, że jeśli oba te parametry nie są określone, rzeczywistą nazwą hosta będzie nazwa NetBIOS hosta; agent Zabbix użyje nazwy hosta NetBIOS do pobrania listy aktywnych kontroli z serwera Zabbix i wysłania do niego wyników.

Klucz "system.hostname" obsługuje dwa opcjonalne parametry — *type* i *transform*.

Type określa typ nazwy, którą pozycja ma zwracać:

- *netbios* (domyślnie) — zwraca nazwę hosta NetBIOS, która jest ograniczona do 15 znaków i występuje wyłącznie WIELKIMI LITERAMI;
- *host* — z rozróżnianiem wielkości liter, zwraca pełną, rzeczywistą nazwę hosta Windows (bez domeny);
- *shorthost* — zwraca część nazwy hosta przed pierwszą kropką. Zwróci pełny ciąg, jeśli nazwa nie zawiera kropki.
- *fqdn* — zwraca Fully Qualified Domain Name (bez końcowej kropki).

Transform pozwala określić dodatkową regułę przekształcenia nazwy hosta:

- *none* (domyślnie) — użyj oryginalnej wielkości liter;
- *lower* — zamień tekst na małe litery.

Aby więc uprościć konfigurację pliku zabbix_agentd.conf i ją ujednolicić, można zastosować trzy różne podejścia:

1. Pozostawić parametry **Hostname** lub **HostnameItem** niezdefiniowane, a agent Zabbix użyje nazwy hosta NetBIOS jako nazwy hosta.
2. Pozostawić parametr **Hostname** niezdefiniowany i zdefiniować **HostnameItem** w następujący sposób:
 HostnameItem=system.hostname[host] — aby agent Zabbix używał pełnej, rzeczywistej (z rozróżnianiem wielkości liter) nazwy hosta Windows jako nazwy hosta.
 HostnameItem=system.hostname[shorthost,lower] — aby agent Zabbix używał tylko części nazwy hosta przed pierwszą kropką, zamienionej na małe litery.
 HostnameItem=system.hostname[fqdn] — aby agent Zabbix używał Fully Qualified Domain Name jako nazwy hosta.

Nazwa hosta jest również używana jako część nazwy usługi Windows, która służy do instalowania, uruchamiania, zatrzymywania i odinstalowywania usługi Windows. Na przykład, jeśli plik konfiguracyjny agenta Zabbix określa `Hostname=Windows_db_server`, to agent zostanie zainstalowany jako usługa Windows "`Zabbix Agent [Windows_db_server]`". Dlatego, aby każda instancja agenta Zabbix miała inną nazwę usługi Windows, każda instancja musi używać innej nazwy hosta.

Instalacja agent jako usługi Windows

Przed zainstalowaniem agent skopiuj ręcznie plik `conf/zabbix_agentd.conf` do katalogu, w którym zostanie zainstalowany plik zabbix_agentd.exe.

Aby zainstalować pojedynczą instancję Zabbix agent z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --install
```

Attention:
W systemie 64-bitowym wymagana jest 64-bitowa wersja Zabbix agent, aby wszystkie kontrole związane z uruchomionymi procesami 64-bitowymi działały prawidłowo.

Jeśli chcesz użyć pliku konfiguracyjnego innego niż domyślny, powinieneś użyć następującego polecenia do instalacji usługi:

```
zabbix_agentd.exe --config <your_configuration_file> --install
```

Należy podać pełną ścieżkę do pliku konfiguracyjnego.

Wiele instancji Zabbix agent można zainstalować jako usługi w następujący sposób:

```
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
```

Zainstalowana usługa powinna być teraz widoczna w Panelu sterowania.

Uruchamianie agent

Aby uruchomić usługę agent, możesz użyć Panelu sterowania lub zrobić to z wiersza poleceń.

Aby uruchomić pojedynczą instancję Zabbix agent z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --start
```

Aby uruchomić pojedynczą instancję Zabbix agent z innym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --config <your_configuration_file> --start
```

Aby uruchomić jedną z wielu instancji Zabbix agent:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --start --multiple-agents
```

Zatrzymywanie agent

Aby zatrzymać usługę agent, możesz użyć Panelu sterowania lub zrobić to z wiersza poleceń.

Aby zatrzymać pojedynczą instancję Zabbix agent uruchomioną z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --stop
```

Aby zatrzymać pojedynczą instancję Zabbix agent uruchomioną z innym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --config <your_configuration_file> --stop
```

Aby zatrzymać jedną z wielu instancji Zabbix agent:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --stop --multiple-agents
```

Odinstalowywanie usługi agenta Windows

Aby odinstalować pojedynczą instancję Zabbix agent przy użyciu domyślnego pliku konfiguracyjnego:

```
zabbix_agentd.exe --uninstall
```

Aby odinstalować pojedynczą instancję Zabbix agent przy użyciu niedomyślnego pliku konfiguracyjnego:

```
zabbix_agentd.exe --config <your_configuration_file> --uninstall
```

Aby odinstalować wiele instancji Zabbix agent z usług Windows:

```
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
```

Ograniczenia

Zabbix agent dla systemu Windows nie obsługuje niestandardowych konfiguracji systemu Windows, w których procesory CPU są rozmieszczone nierównomiernie pomiędzy węzłami NUMA. Jeśli logiczne procesory CPU są rozmieszczone nierównomiernie, metryki wydajności CPU mogą nie być dostępne dla niektórych procesorów CPU. Na przykład, jeśli istnieją 72 logiczne procesory CPU i 2 węzły NUMA, oba węzły muszą mieć po 36 procesorów CPU.

12 Konfiguracja SAML z Microsoft Entra ID

Przegląd

Ta sekcja zawiera wytyczne dotyczące konfigurowania logowania jednokrotnego i konfiguracja kont użytkowników w Zabbix z Microsoft Entra ID (dawniej Microsoft Azure Active Directory) przy użyciu uwierzytelniania SAML 2.0.

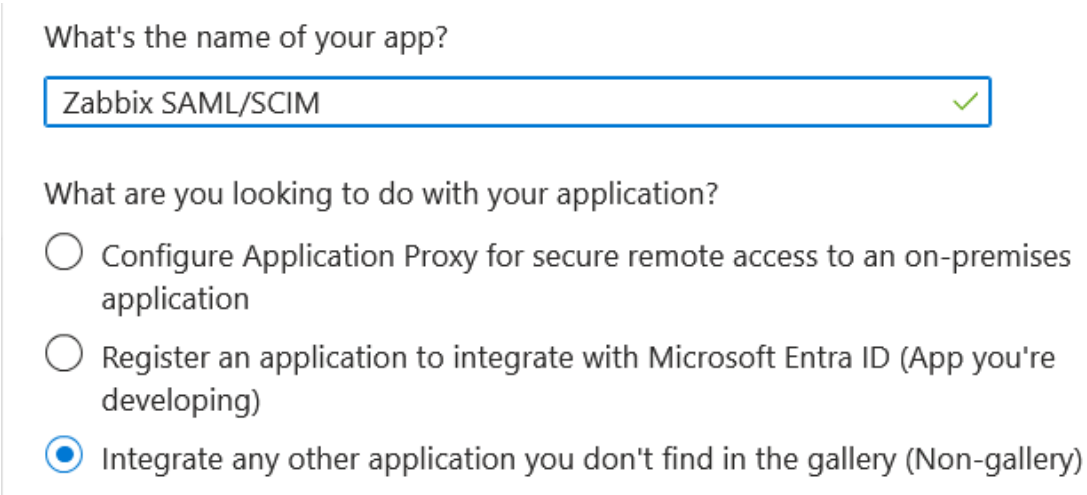
Konfiguracja ID Microsoft Entra

Tworzenie aplikacji

1. Zaloguj się do centrum administracyjnego Microsoft Entra w Microsoft Entra ID. Do celów testowych możesz utworzyć bezpłatne konto próbne w Microsoft Entra ID.

2. W centrum administracyjnym Microsoft Entra wybierz *Applications -> Enterprise applications -> New application -> Create your own application*.

3. Dodaj nazwę swojej aplikacji i wybierz opcję *Integrate any other application...*. Następnie kliknij *Create*.

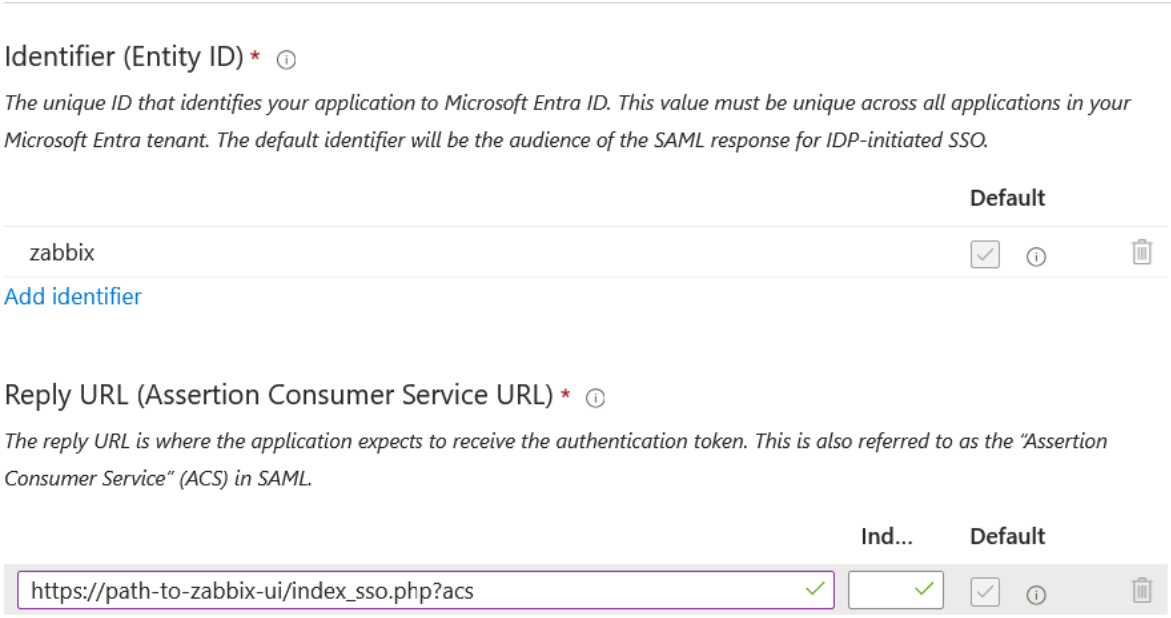


Konfigurowanie pojedynczego logowania

1. Na stronie aplikacji przejdź do *Set up single sign on* i kliknij *Get started*. Następnie wybierz *SAML*.

2. Edytuj *Basic SAML Configuration*:

- W *Identifier (Entity ID)* ustaw unikalną nazwę identyfikującą Twoją aplikację w Microsoft Entra ID, na przykład `zabbix`;
- W *Reply URL (Assertion Consumer Service URL)* ustaw punkt końcowy pojedynczego logowania Zabbix: `https://<path-to-zabbix-u`



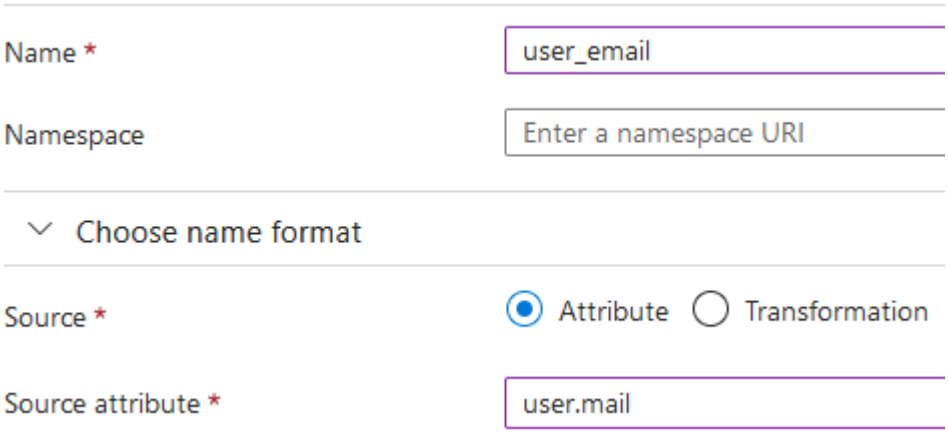
Pamiętaj, że wymagany jest protokół "https". Aby to działało z Zabbix, należy dodać do `conf/zabbix.conf.php` następującą linię:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

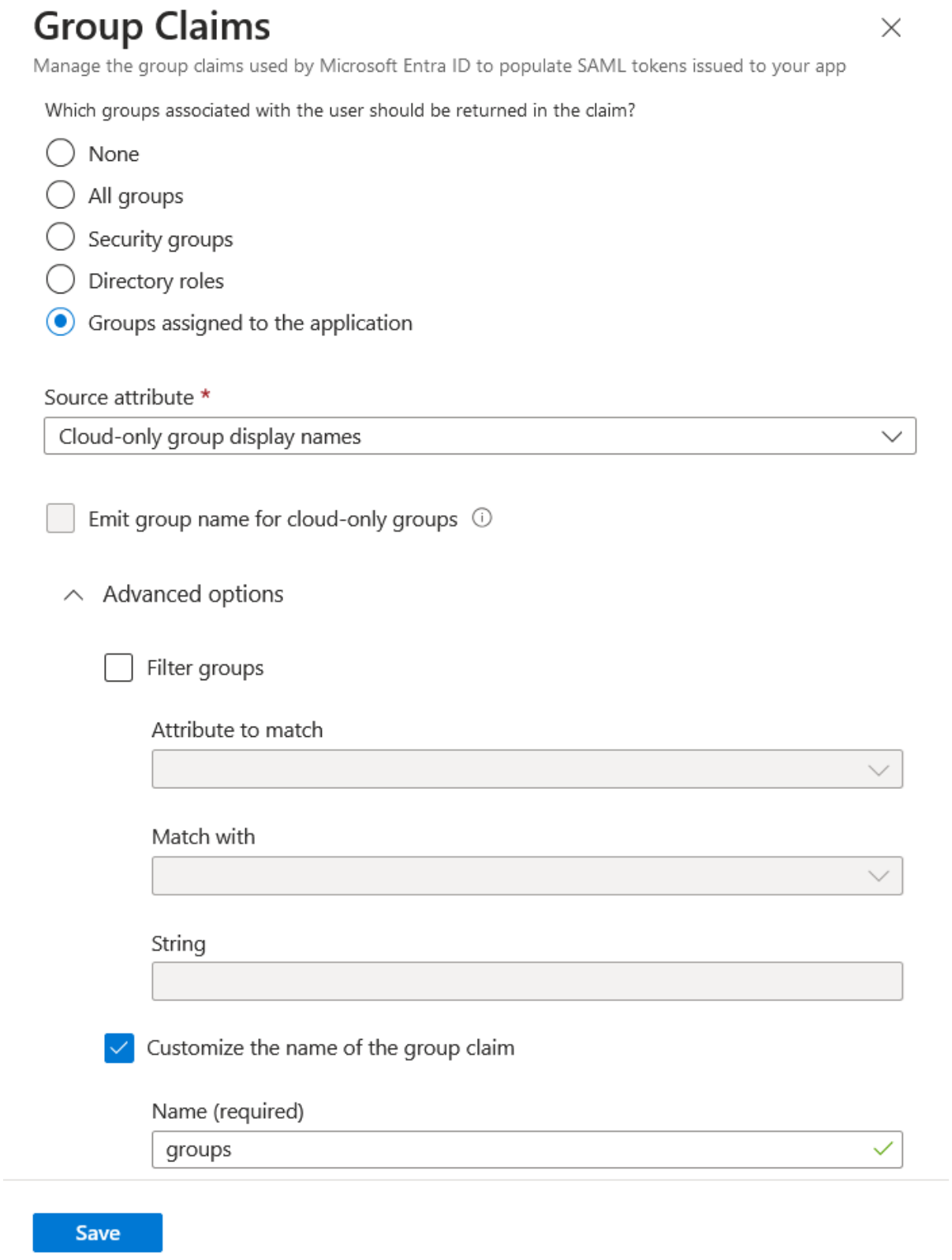
3. Edytuj *Attributes & Claims*. Musisz dodać wszystkie atrybuty, które chcesz przekazać do Zabbix (user_name, user_lastname, user_email, user_mobile, groups).

Nazwy atrybutów są dowolne. Można użyć różnych nazw atrybutów, jednak wymagane jest, aby odpowiadały one odpowiedniej wartości pola w ustawieniach SAML Zabbix.

- Kliknij *Add new claim*, aby dodać atrybut:



- Kliknij *Add a group claim*, aby dodać atrybut służący do przekazywania grup do Zabbix:



W tym roszczeniu ważne jest, aby do Zabbix były przekazywane nazwy grup (a nie identyfikatory grup) przez wybrany *Source attribute*. W przeciwnym razie aprowizacja użytkowników JIT nie będzie działać poprawnie.

4. W *SAML Certificates* pobierz certyfikat Base64 udostępniony przez Entra ID i umieść go w `conf/certs` instalacji frontend Zabbix.

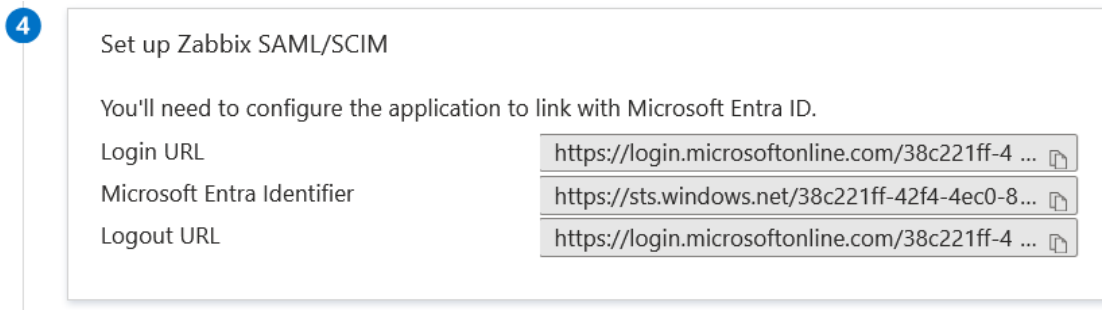
Nadaj mu uprawnienia 644, uruchamiając:

```
chmod 644 entra.cer
```

Upewnij się, że `conf/zabbix.conf.php` zawiera linię:

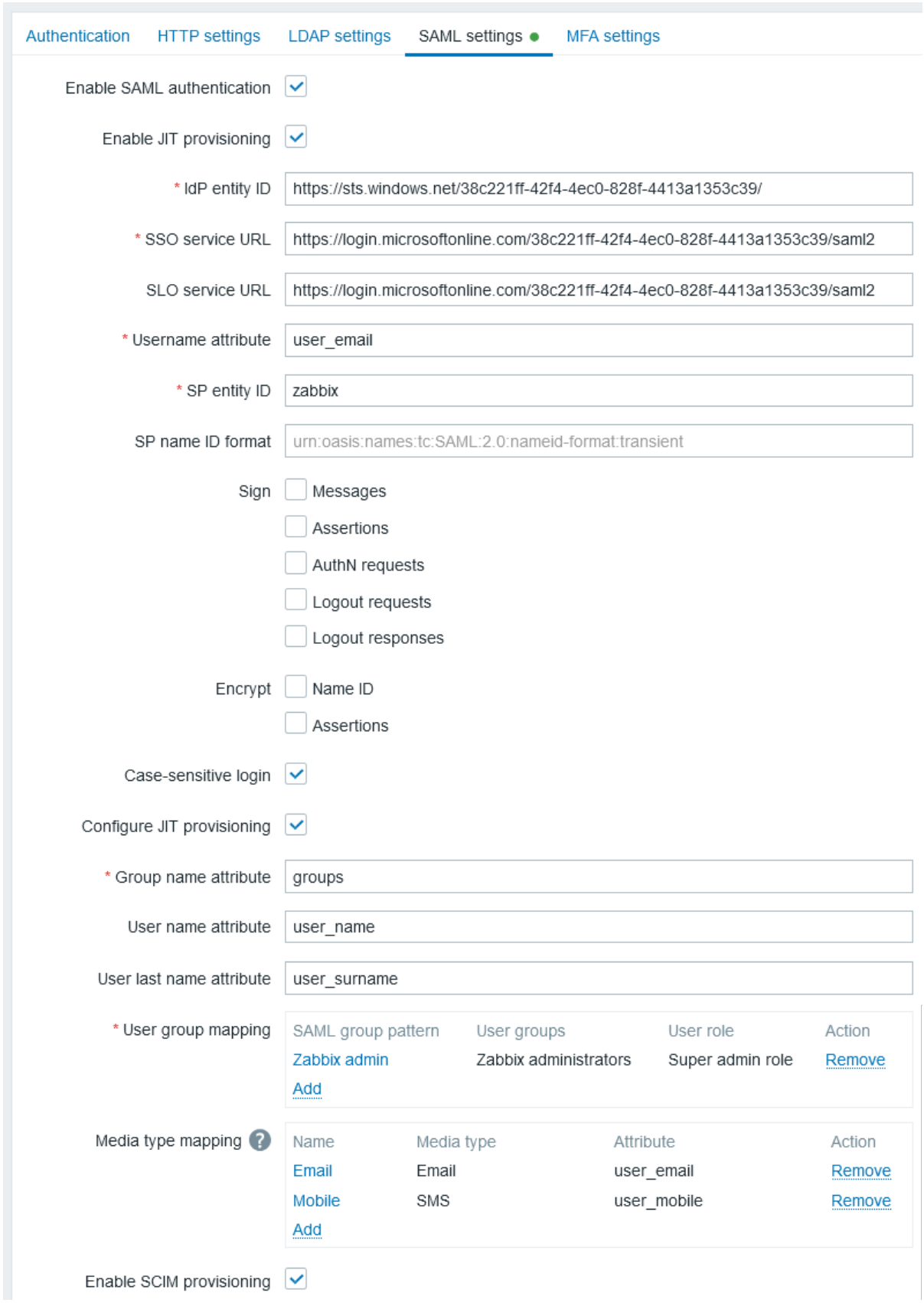
```
$SSO['IDP_CERT'] = 'conf/certs/entra.cer';
```

5. Użyj wartości z *Set up <your app name>* w Entra ID, aby skonfigurować uwierzytelnianie SAML Zabbix (zobacz następna sekcja):



Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i wypełnij opcje konfiguracji na podstawie konfiguracji Entra ID:



Pole Zabbix	Pole konfiguracji w Entra ID	Przykładowa wartość
IdP entity ID	Microsoft Entra identifier	
SSO service URL	Login URL	
SLO service URL	Logout URL	
SP entity ID	Identifier (Entity ID)	
Username attribute	Custom attribute (claim)	`user_email`
Group name attribute	Custom attribute (claim)	`groups`
User name attribute	Custom attribute (claim)	`user_name`

Pole Zabbix	Pole konfiguracji w Entra ID	Przykładowa wartość
User last name attribute	Custom attribute (claim)	`user_lastname`

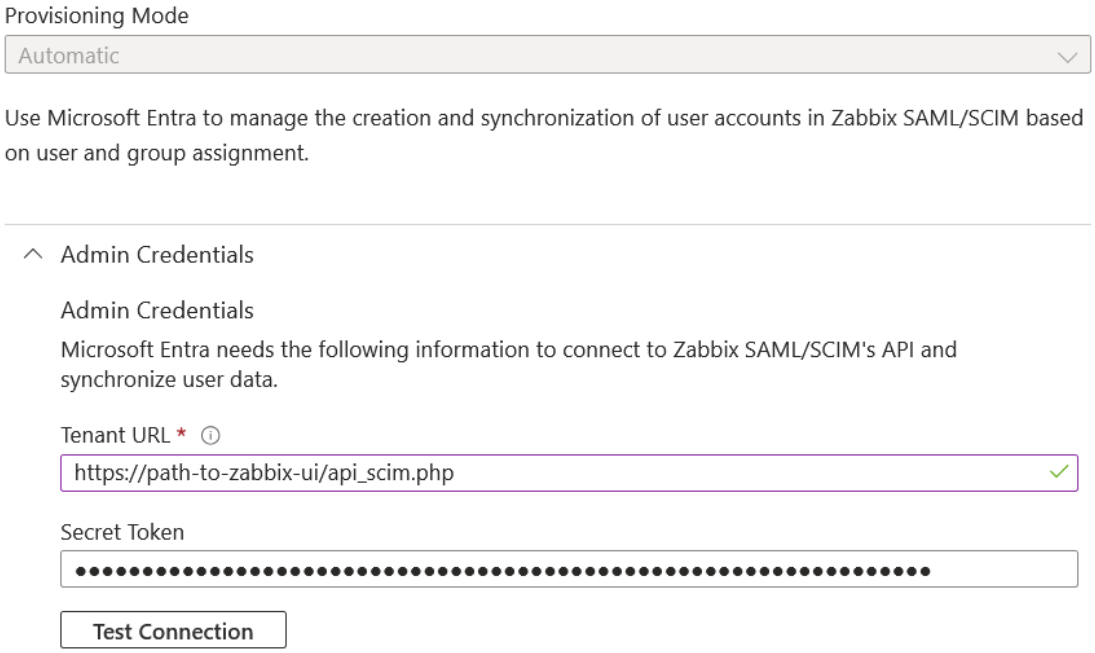
Wymagane jest również skonfigurowanie mapowania grup użytkowników. Mapowanie mediów jest opcjonalne.

Kliknij *Update*, aby zapisać te ustawienia.

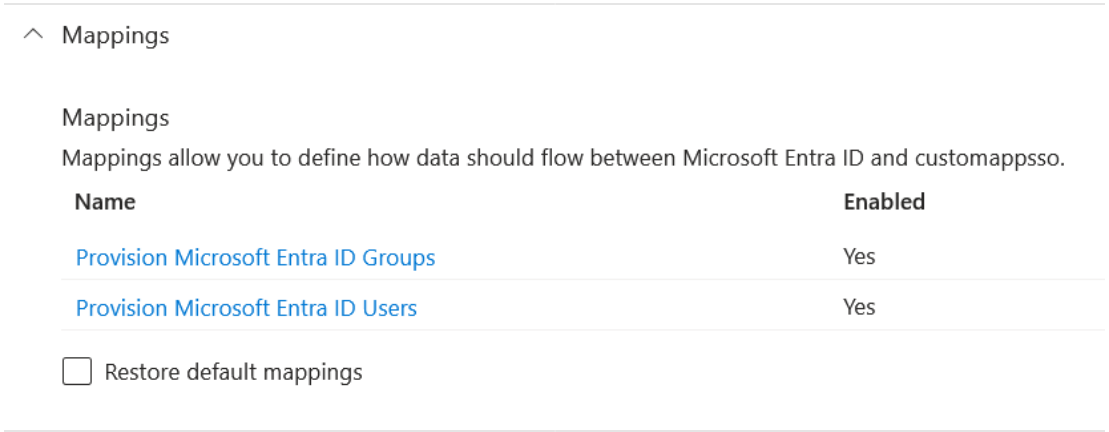
Provisioning użytkowników SCIM

1. Na stronie aplikacji w Entra ID, z głównego menu otwórz stronę Provisioning. Kliknij *Get started*, a następnie wybierz tryb Automatic provisioning:

- W *Tenant URL* ustaw następującą wartość: `https://<path-to-zabbix-ui>/api_scim.php`
- W *Secret token* wprowadź token API Zabbix z uprawnieniami Super admin.
- Kliknij *Test connection*, aby sprawdzić, czy połączenie zostało nawiązane.

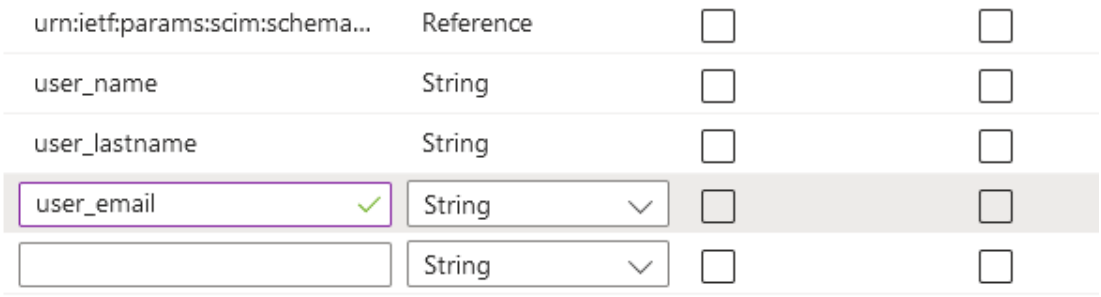


2. Teraz możesz dodać wszystkie atrybuty, które będą przekazywane przez SCIM do Zabbix. Aby to zrobić, kliknij *Mappings*, a następnie *Provision Microsoft Entra ID Users*.



Na dole listy Attribute Mapping włącz *Show advanced options*, a następnie kliknij *Edit attribute list for customappsso*.

Na dole listy atrybutów dodaj własne atrybuty typu 'String':



Zapisz listę.

3. Teraz możesz dodać mapowania dla dodanych atrybutów. Na dole listy Attribute Mapping kliknij *Add New Mapping* i utwórz mapowania pokazane poniżej:



Gdy wszystkie mapowania zostaną dodane, zapisz listę mapowań.



4. Jako warunek wstępny provisioning użytkowników do Zabbix, musisz mieć skonfigurowanych użytkowników i grupy w Entra ID.

Aby to zrobić, przejdź do *Microsoft Entra admin center*, a następnie dodaj użytkowników/grupy na odpowiednich stronach Users i Groups.

5. Gdy użytkownicy i grupy zostaną utworzeni w Entra ID, możesz przejść do menu *Users and groups* swojej aplikacji i dodać ich do aplikacji.

6. Przejdź do menu *Provisioning* swojej aplikacji i kliknij *Start provisioning*, aby rozpocząć provisioning użytkowników do Zabbix.

Należy pamiętać, że żądanie PATCH Users w Entra ID nie obsługuje zmian w media.

Podpisywanie żądań uwierzytelniania

Można skonfigurować Entra ID tak, aby weryfikował podpis podpisanych żądań uwierzytelniania.

Aby to działało, utwórz klucze publiczny/prywatny:

```
openssl req -x509 -newkey rsa:4096 -keyout /usr/share/zabbix/conf/certs/request-sign.key -out /usr/share/z
```

Przypisz uprawnienia:

```
chown apache /usr/share/zabbix/conf/certs/request-sign.key
chmod 400 /usr/share/zabbix/conf/certs/request-sign.key
```

Zaktualizuj konfigurację frontendu Zabbix, dodając:

```
$SSO['SP_KEY'] = 'conf/certs/request-sign.key';
$SSO['SP_CERT'] = 'conf/certs/request-sign.crt';
```

Rozwiązywanie problemów

Problemy z uwierzytelnianiem mogą wystąpić w przeglądarkach MicroSoft Edge, gdy użytkownik próbujący zalogować się do Zabbix za pomocą SAML jest już zalogowany w profilu MicroSoft Edge. Oznaką takiego problemu może być możliwość zalogowania się do Zabbix przy użyciu MicroSoft Edge w trybie prywatnym.

Aby uniknąć problemów z uwierzytelnianiem w tym przypadku, może być konieczne ustawienie `requestedAuthnContext` na "false" w pliku konfiguracyjnym frontend Zabbix (*zabbix.conf.php*).

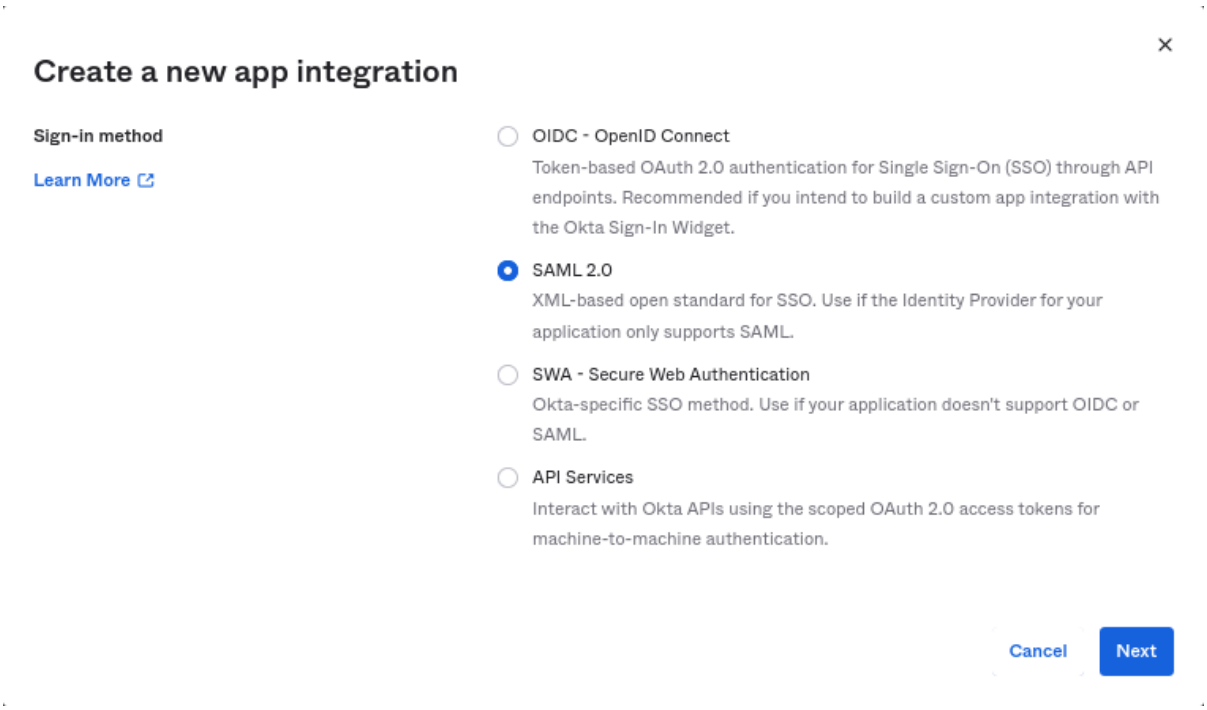
```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => false
    ]
];
```

13 Konfiguracja SAML z Okta

Ta sekcja zawiera wskazówki dotyczące konfiguracji Okta, aby włączyć uwierzytelnianie SAML 2.0 oraz provisioning użytkowników dla Zabbix.

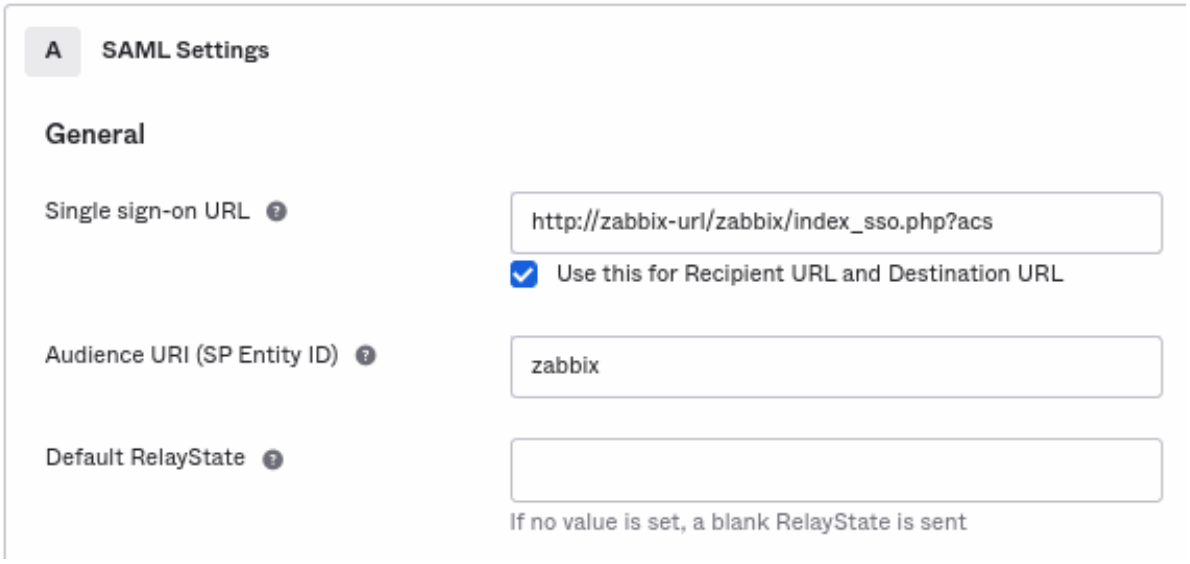
Konfiguracja Okta

1. Przejdź do https://developer.okta.com/signup/ i zarejestruj się / zaloguj na swoje konto.
2. W interfejsie webowym Okta przejdź do *Applications* → *Applications*.
3. Kliknij *Create App Integration*.

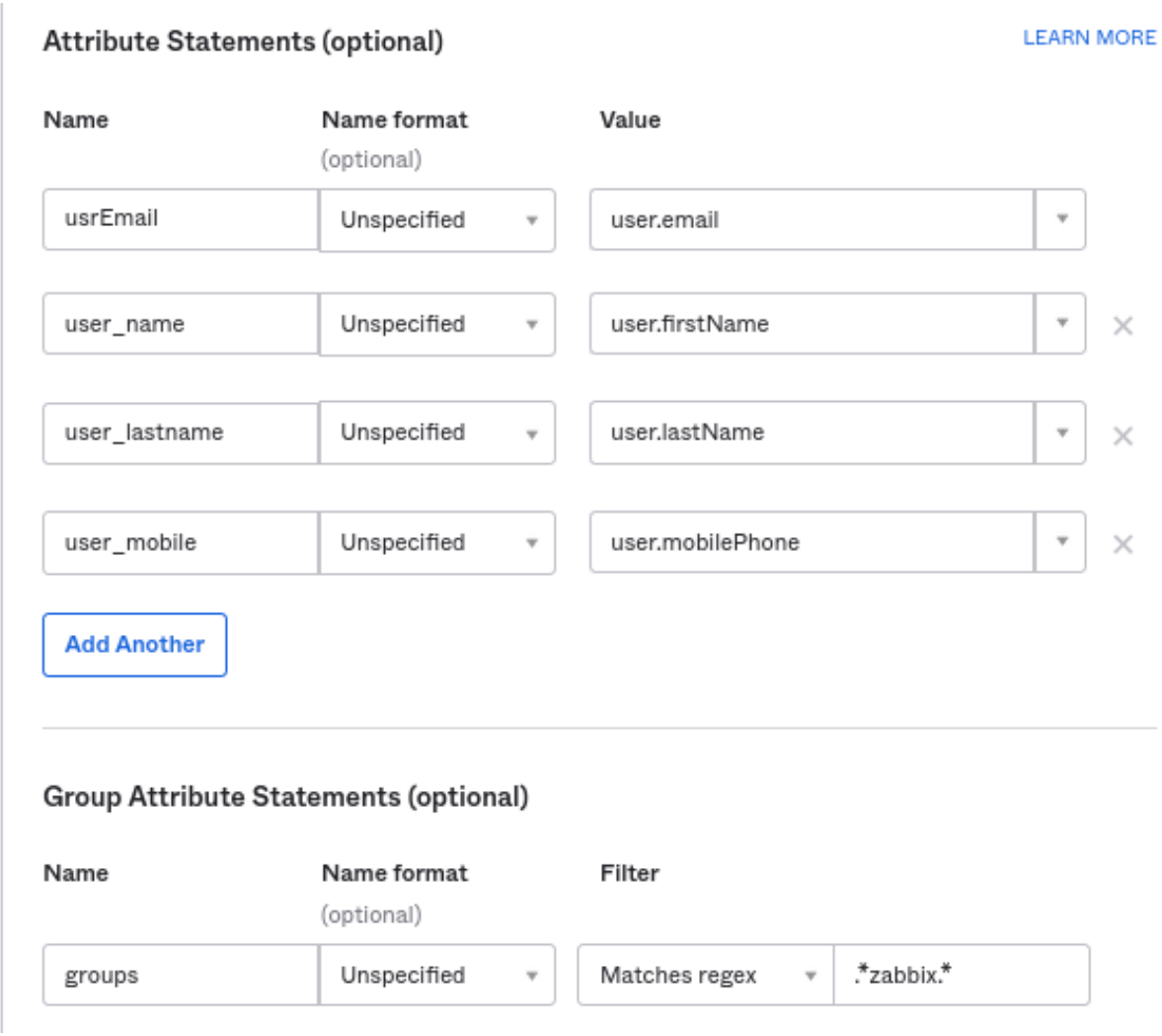


Wybierz "SAML 2.0" jako metodę logowania i kliknij *Next*.

4. W ustawieniach ogólnych wpisz nazwę aplikacji i kliknij *Next*.
5. W konfiguracji SAML wprowadź poniższe wartości, a następnie kliknij *Next*.



- W sekcji **General** dodaj:
  - *Single sign-on URL*: `http://<your-zabbix-url>/zabbix/index_sso.php?acs<br>` Zwróć uwagę na użycie "http", a nie "https", aby parametr `acs` nie został obcięty w żądaniu. Pole wyboru *Use this for Recipient URL and Destination URL* również powinno być zaznaczone.
  - *Audience URI (SP Entity ID)*: `zabbix<br>` Zwróć uwagę, że ta wartość będzie używana w asercji SAML jako unikalny identyfikator dostawcy usług (jeśli nie będzie zgodna, operacja zostanie odrzucona). W tym polu można podać adres URL lub dowolny ciąg znaków.
  - *Default RelayState*:
 Pozostaw to pole puste; jeśli wymagane jest niestandardowe przekierowanie, można je dodać w Zabbix w ustawieniach *Users* → *Users*.
  - Uzupełnij pozostałe pola zgodnie z własnymi preferencjami.
- W sekcji **Attribute Statements/Group Attribute Statements** dodaj:



Te deklaracje atrybutów są wstawiane do asercji SAML udostępnianych Zabbix.

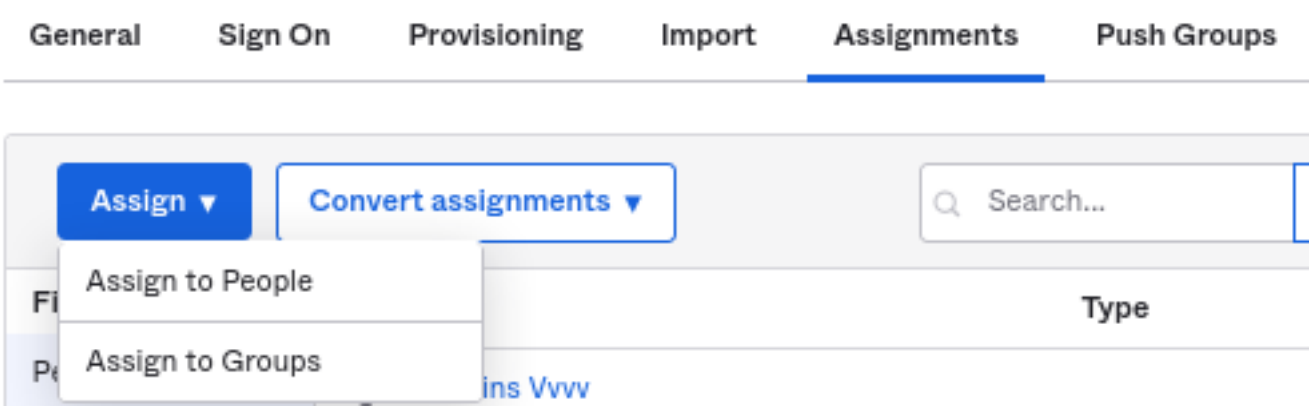
Nazwy atrybutów użyte tutaj są przykładowe i mogą być dowolne. Możesz użyć innych nazw atrybutów, jednak muszą one odpowiadać odpowiednim wartościom pól w ustawieniach SAML Zabbix.

Jeśli chcesz skonfigurować logowanie SAML do Zabbix *bez* JIT provisioning użytkowników, wymagany jest tylko atrybut email.

Note:
Jeśli planujesz używać szyfrowanego połączenia, wygeneruj prywatny i publiczny certyfikat szyfrowania, a następnie prześlij publiczny certyfikat do Okta. Formularz przesyłania certyfikatu pojawi się, gdy *Assertion Encryption* zostanie ustawione na "Encrypted" (kliknij *Show Advanced Settings*, aby znaleźć ten parametr).

6. W następnej karcie wybierz "I'm a software vendor. I'd like to integrate my app with Okta" i naciśnij "Finish".

7. Przejdź do karty "Assignments" nowo utworzonej aplikacji i kliknij przycisk *Assign*, a następnie wybierz z listy rozwijanej *Assign to People*.



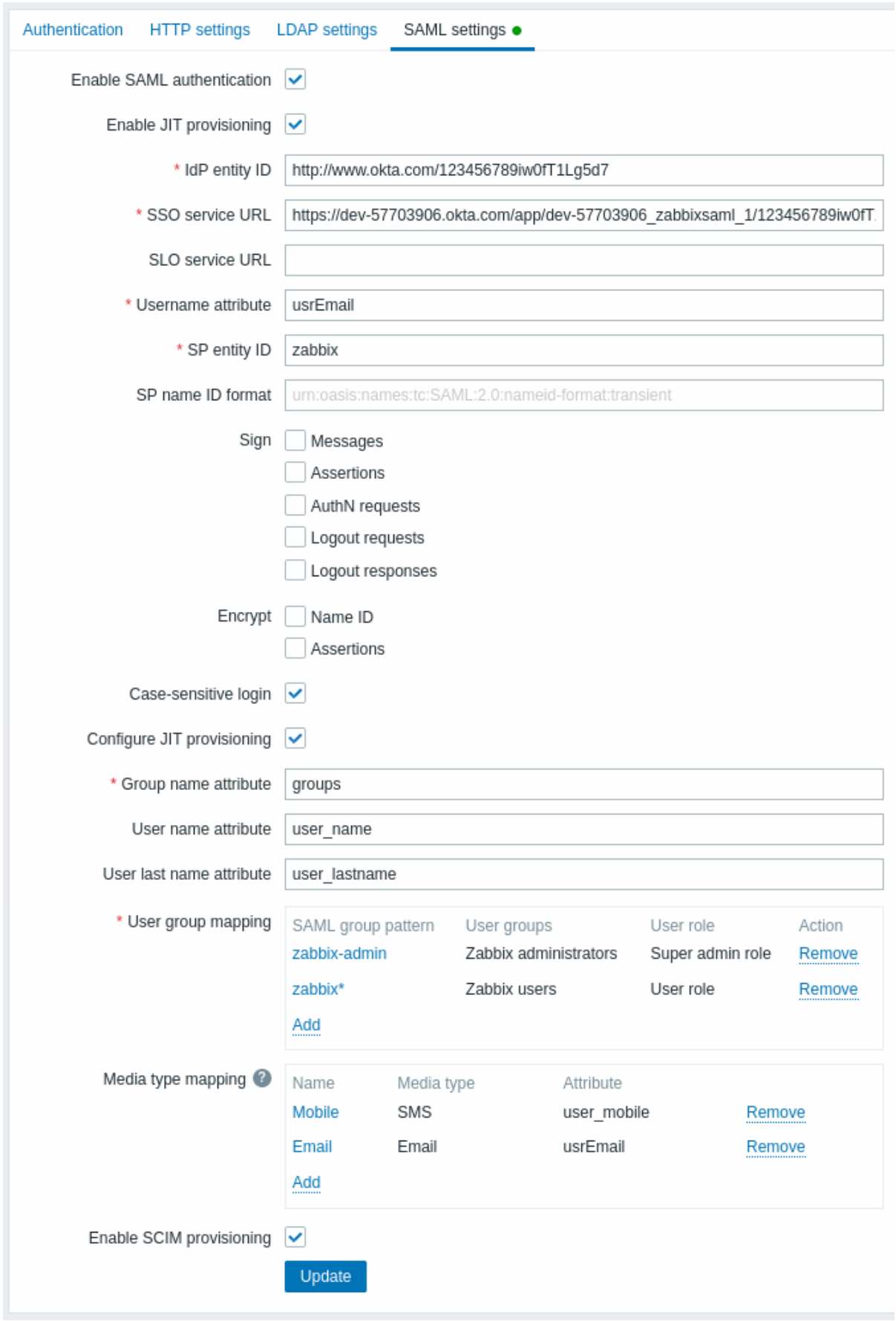
8. W oknie podręcznym, które się pojawi, przypisz aplikację osobom, które będą używać SAML 2.0 do uwierzytelniania w Zabbix, a następnie kliknij *Save and go back*.

9. Przejdź do karty "Sign On" i kliknij przycisk *View Setup Instructions*.

Instrukcje konfiguracji zostaną otwarte w nowej karcie; pozostaw tę kartę otwartą podczas konfigurowania Zabbix.

Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i wypełnij opcje konfiguracji na podstawie instrukcji konfiguracji z Okta:



Pole Zabbix	Pole konfiguracji w Okta	Przykładowa wartość
IdP entity ID	Identity Provider Issuer	
SSO service URL	Identity Provider Single Sign-On URL	
Username attribute	Attribute name	`usrEmail`
SP entity ID	Audience URI	`zabbix`
Group name attribute	Attribute name	`groups`

Pole Zabbix	Pole konfiguracji w Okta	Przykładowa wartość
User name attribute	Attribute name	`user_name`
User last name attribute	Attribute name	`user_lastname`

Wymagana jest również konfiguracja mapowania grup użytkowników i mediów.

2. Pobierz certyfikat udostępniony w instrukcjach konfiguracji SAML w Okta i zapisz go w folderze *ui/conf/certs* jako idp.crt.

Nadaj mu uprawnienia 644, uruchamiając:

```
chmod 644 idp.crt
```

3. Jeśli w Okta ustawiono *Assertion Encryption* na "Encrypted", w Zabbix należy również zaznaczyć pole wyboru "Assertions" dla parametru *Encrypt*.

4. Naciśnij przycisk "Update", aby zapisać te ustawienia.

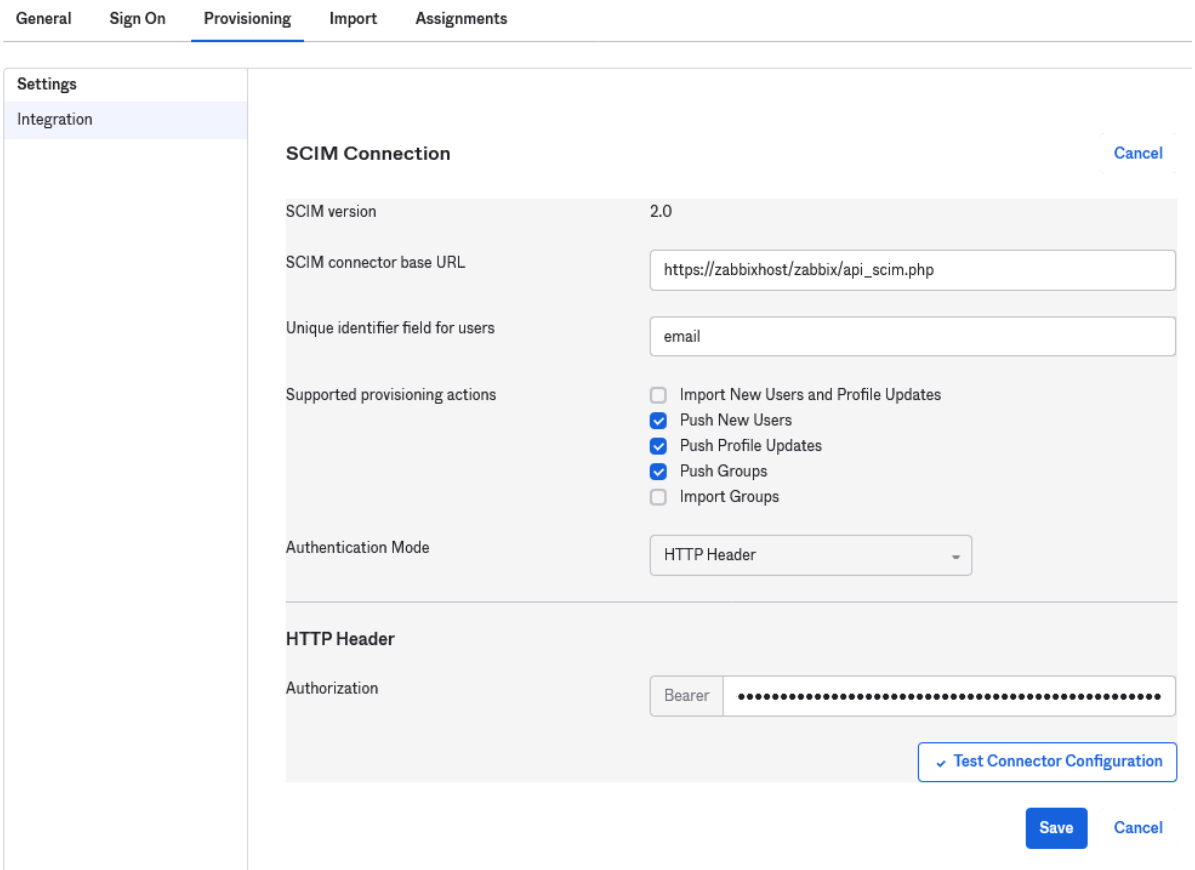
Provisioning SCIM

1. Aby włączyć provisioning SCIM, przejdź do "General" -> "App Settings" aplikacji w Okta.

Zaznacz pole wyboru *Enable SCIM provisioning*. W rezultacie pojawi się nowa karta *Provisioning*.

2. Przejdź do karty "Provisioning", aby skonfigurować połączenie SCIM:

- W *SCIM connector base URL* określ ścieżkę do frontend Zabbix i dodaj do niej `api_scim.php`, tj.:
 `https://<your-zabbix-url>/`
- *Unique identifier field for users*: `email`
- *Authentication mode*: `HTTP header`
- W *Authorization* wprowadź prawidłowy token API z uprawnieniami Super admin



Attention:
Jeśli występują problemy z uwierzytelnianiem, zobacz Authorization header forwarding.

3. Kliknij *Test Connector Configuration*, aby przetestować połączenie. Jeśli wszystko jest poprawne, zostanie wyświetlony komunikat o powodzeniu.

4. W "Provisioning" -> "To App" upewnij się, że zaznaczono następujące pola wyboru:

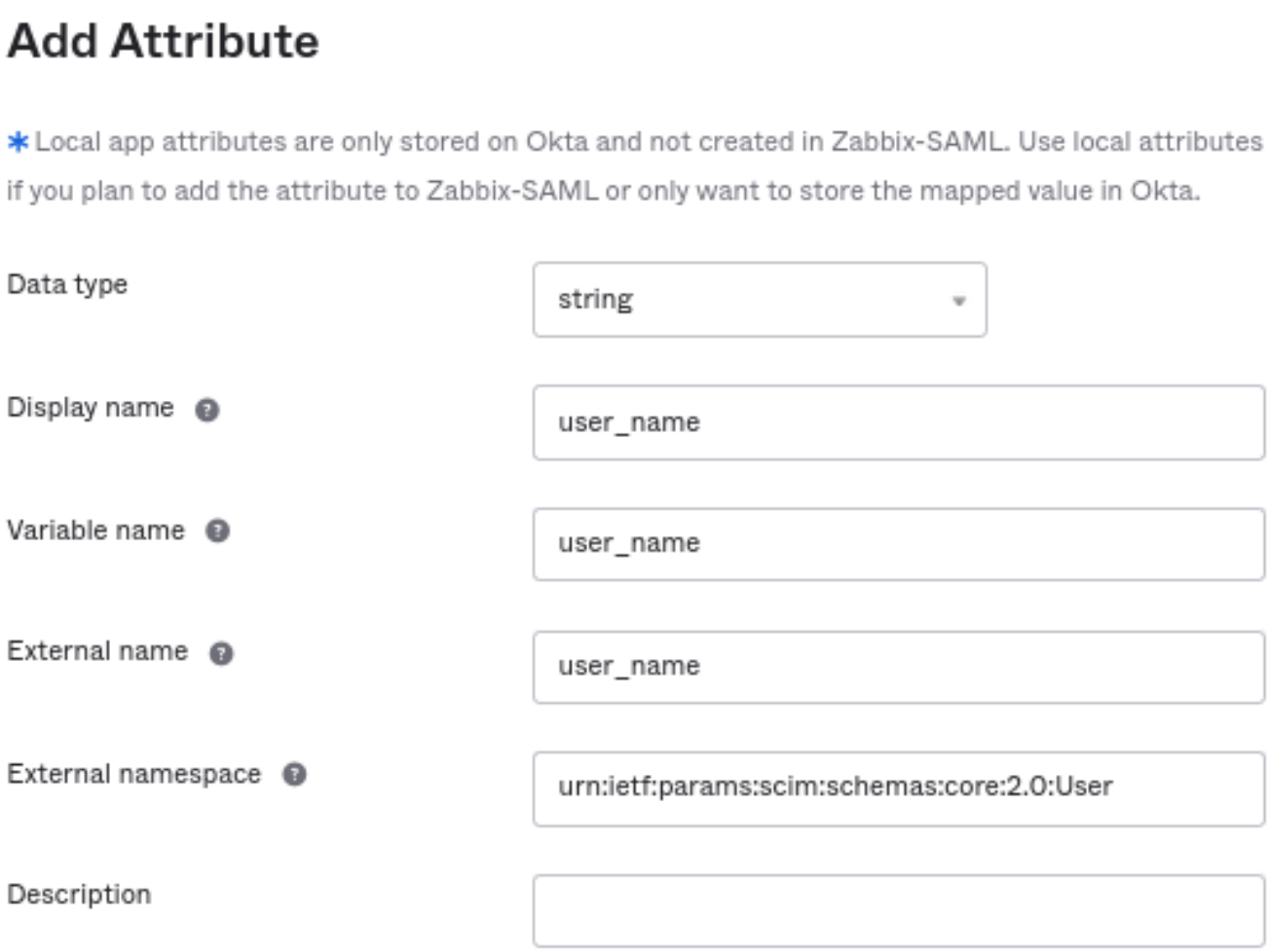
- Create Users

- Update User Attributes
- Deactivate Users

Zapewni to, że te typy żądań będą wysyłane do Zabbix.

5. Upewnij się, że wszystkie atrybuty zdefiniowane w SAML są zdefiniowane w SCIM. Edytor profilu dla aplikacji jest dostępny w ”Provisioning” -> ”To App” po kliknięciu *Go to Profile Editor*.

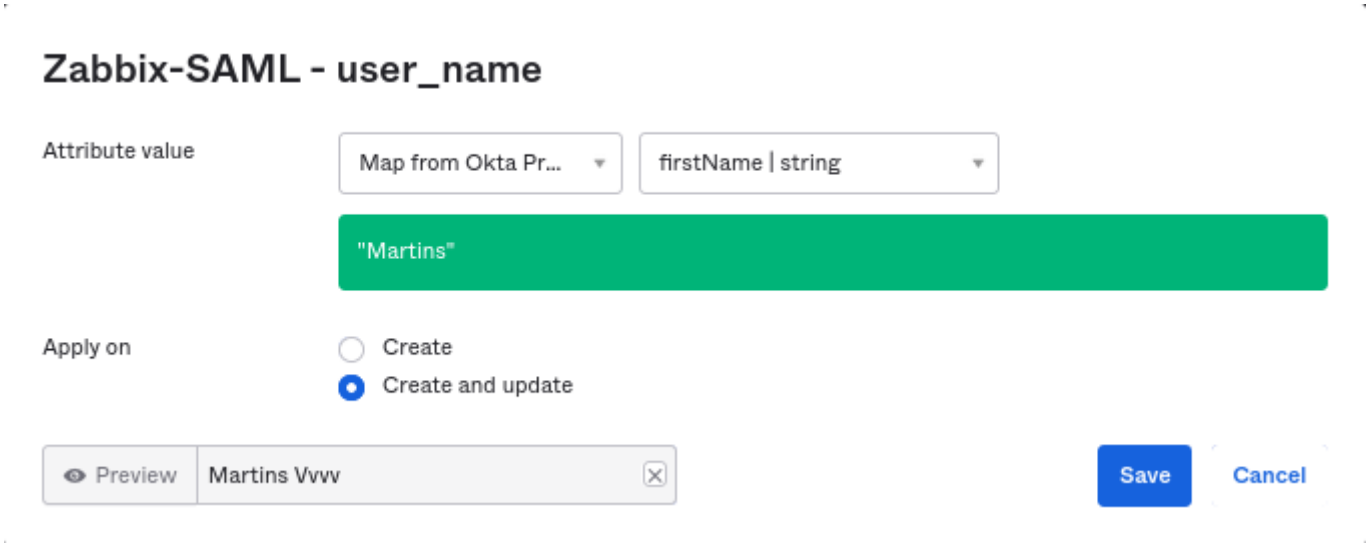
Kliknij *Add Attribute*. Wypełnij wartości *Display name*, *Variable name*, *External name* nazwą atrybutu SAML, na przykład `user_name`.



External namespace powinien być taki sam jak schemat użytkownika: `urn:ietf:params:scim:schemas:core:2.0:User`

6. Przejdź do ”Provisioning” -> ”To App” -> ”Attribute Mappings” swojej aplikacji. Kliknij na dole *Show Unmapped Attributes*. Pojawią się nowo dodane atrybuty.

7. Przypisz każdy dodany atrybut.



8. Dodaj użytkowników na karcie ”Assignments”. Użytkownicy muszą wcześniej zostać dodani w *Directory -> People*. Wszystkie te przypisania zostaną wysłane jako żądania do Zabbix.

9. Dodaj grupy na karcie ”Push Groups”. Wzorzec mapowania grup użytkowników w ustawieniach SAML Zabbix musi odpowiadać grupie określonej tutaj. Jeśli nie ma zgodności, użytkownik nie może zostać utworzony w Zabbix.

Informacje o członkach grupy są wysyłane za każdym razem, gdy zostanie wprowadzona jakaś zmiana.

14 Konfiguracja SAML z OneLogin

Przegląd

Ta sekcja zawiera wytyczne dotyczące konfigurowania logowania jednokrotnego oraz aprowizacji użytkowników w Zabbix z OneLogin przy użyciu uwierzytelniania SAML 2.0.

Konfiguracja OneLogin

Tworzenie aplikacji

1. Zaloguj się na swoje konto w OneLogin. Do celów testowych możesz utworzyć bezpłatne konto deweloperskie w OneLogin.

2. W interfejsie webowym OneLogin przejdź do *Applications → Applications*.

3. Kliknij ”Add App” i wyszukaj odpowiednią aplikację. Wytyczne na tej stronie opierają się na przykładzie aplikacji *SCIM Provisioner with SAML (SCIM v2 Enterprise, full SAML)*.

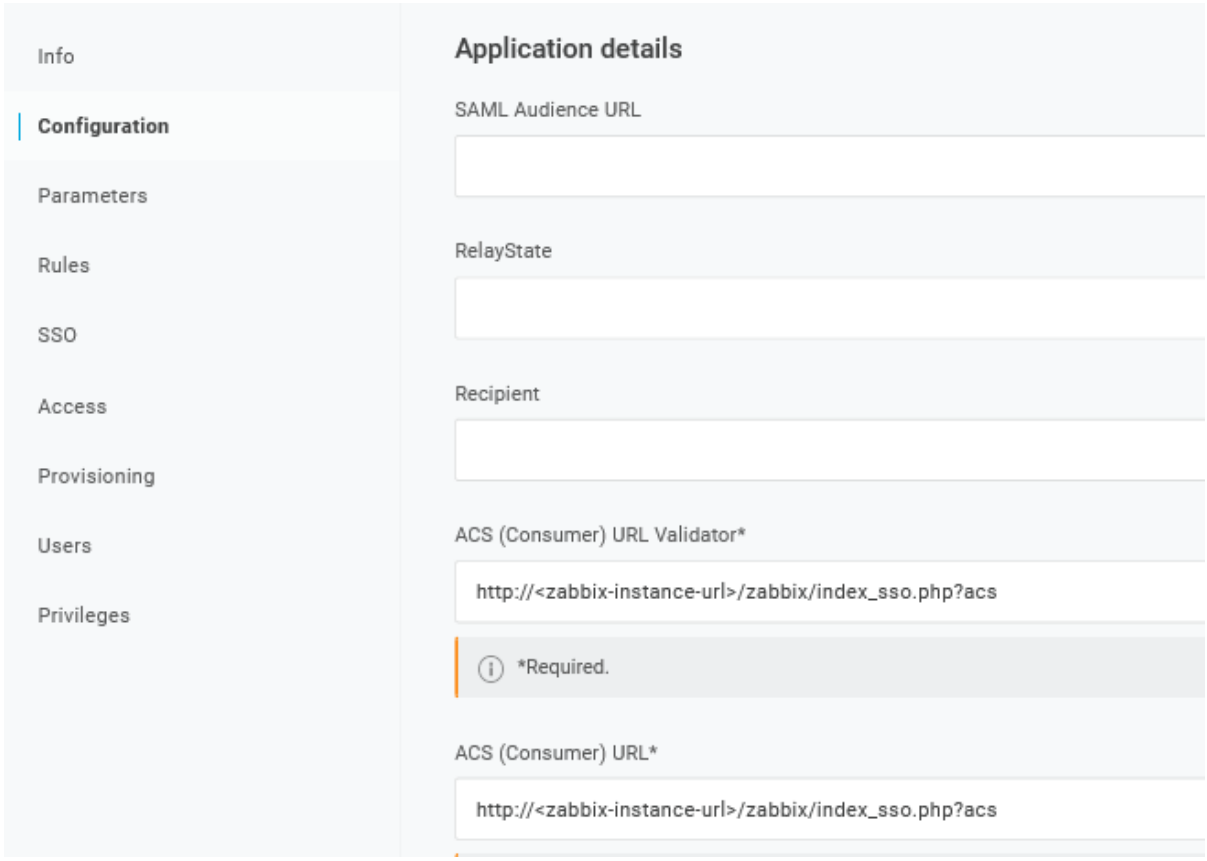
4. Na początek możesz dostosować nazwę wyświetlaną swojej aplikacji. Możesz też dodać ikonę i szczegóły aplikacji. Następnie kliknij *Save*.

Konfiguracja SSO/SCIM provisioning

1. W *Configuration -> Application details* ustaw punkt końcowy logowania jednokrotnego Zabbix `http://<zabbix-instance-url>/zabbix` jako wartość tych pól:

- *ACS (Consumer) URL Validator*
- *ACS (Consumer) URL*

Zwróć uwagę na użycie `http`, a nie `https`, aby parametr `acs` nie został obcięty w żądaniu.



Możliwe jest również użycie `https`. Aby to działało z Zabbix, należy dodać do `conf/zabbix.conf.php` następującą linię:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

Pozostałe opcje pozostaw z wartościami domyślnymi.

2. W *Configuration -> API connection* ustaw następujące wartości:

- *SCIM Base URL*: `https://<zabbix-instance-url>/zabbix/api_scim.php`
- *SCIM JSON Template*: powinien zawierać wszystkie niestandardowe atrybuty, które chcesz przekazać do Zabbix przez SCIM, takie jak `user_name`, `user_lastname`, `user_email` i `user_mobile`:

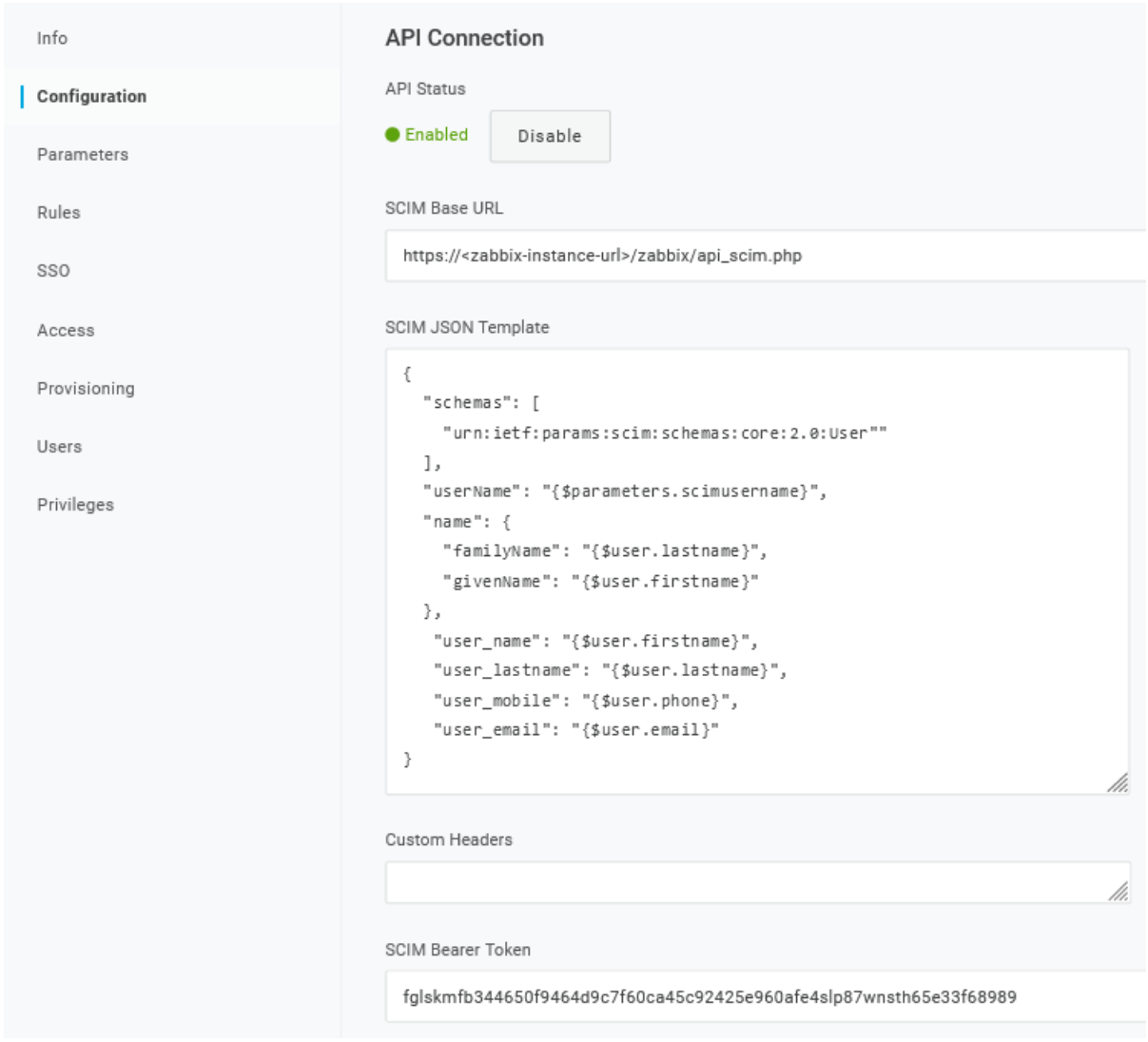
```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User"
  ],
  "userName": "{$parameters.scimusername}",
  "name": {
    "familyName": "{$user.lastname}",
    "givenName": "{$user.firstname}"
  },
   "user_name": "{$user.firstname}",
   "user_lastname": "{$user.lastname}",
   "user_mobile": "{$user.phone}",
   "user_email": "{$user.email}"
}
```

Nazwy atrybutów są dowolne. Można użyć innych nazw atrybutów, jednak muszą one odpowiadać odpowiedniej wartości pola w ustawieniach Zabbix SAML.

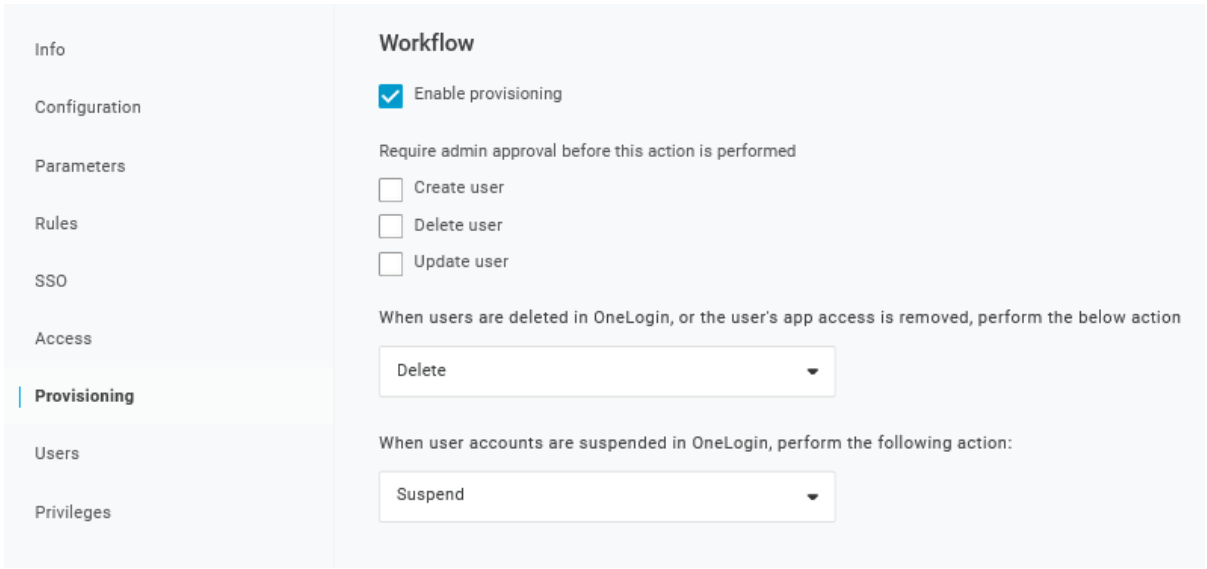
Zwróć uwagę, że aby provisioning użytkowników działał, OneLogin musi otrzymać w odpowiedzi atrybut `name` z `givenName` i `familyName`, nawet jeśli nie był on wymagany przez dostawcę usług. Dlatego konieczne jest określenie tego w schemacie w części konfiguracji aplikacji.

- *SCIM Bearer Token*: wprowadź token API Zabbix z uprawnieniami Super admin.

Kliknij *Enable*, aby aktywować połączenie.

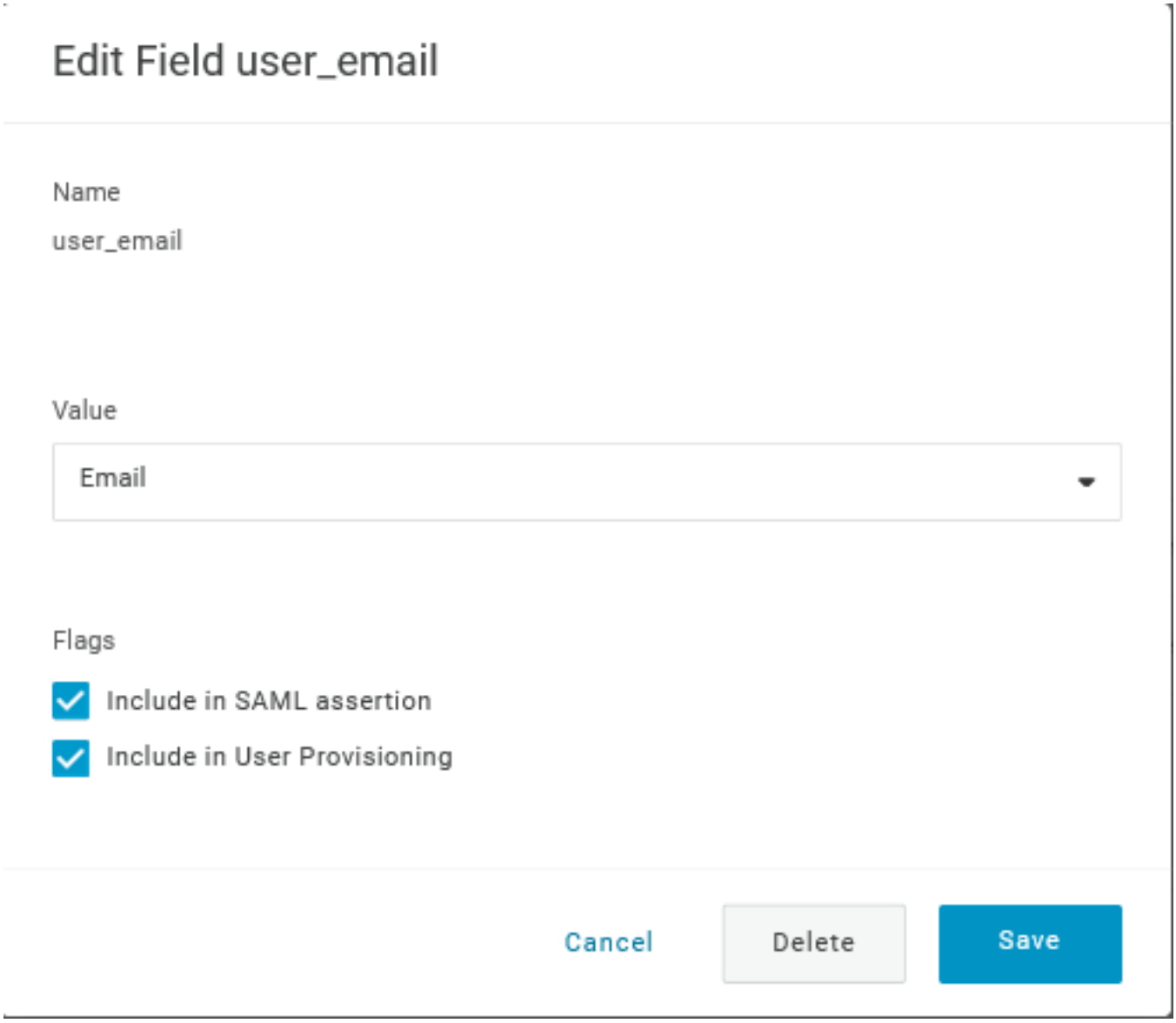


3. Na stronie *Provisioning* włącz opcję Provisioning:



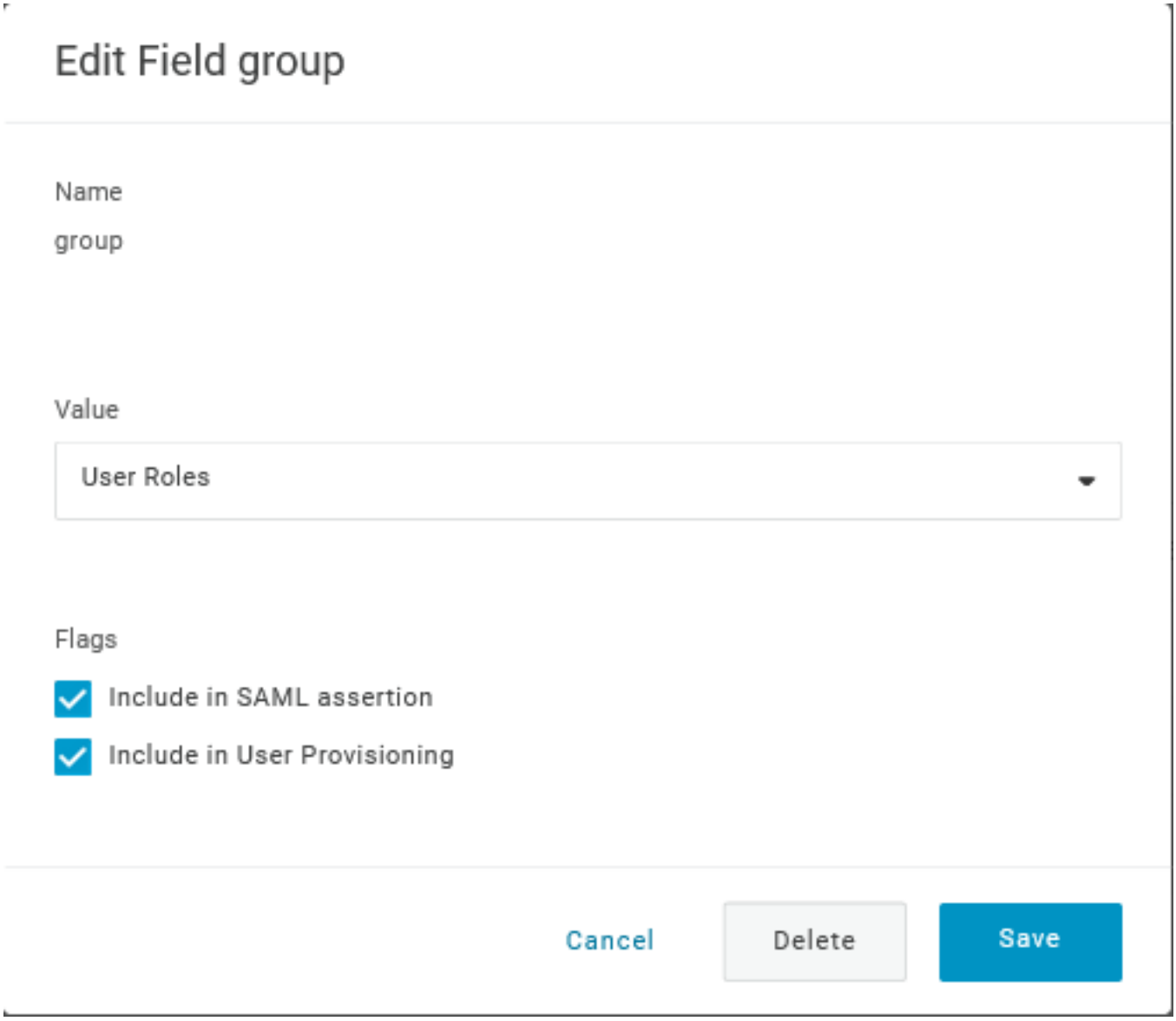
4. Strona *Parameters* zawiera listę domyślnych parametrów:

- Upewnij się, że `scimusername` odpowiada wartości logowania użytkownika w OneLogin (np. adresowi e-mail);
- Zaznacz opcję *Include in User Provisioning* dla parametru `Groups`;
- Kliknij "+" , aby utworzyć niestandardowe parametry wymagane dla asercji SAML i provisioning użytkowników, takie jak `user_name`, `user_lastname`, `user_email` i `user_mobile`:

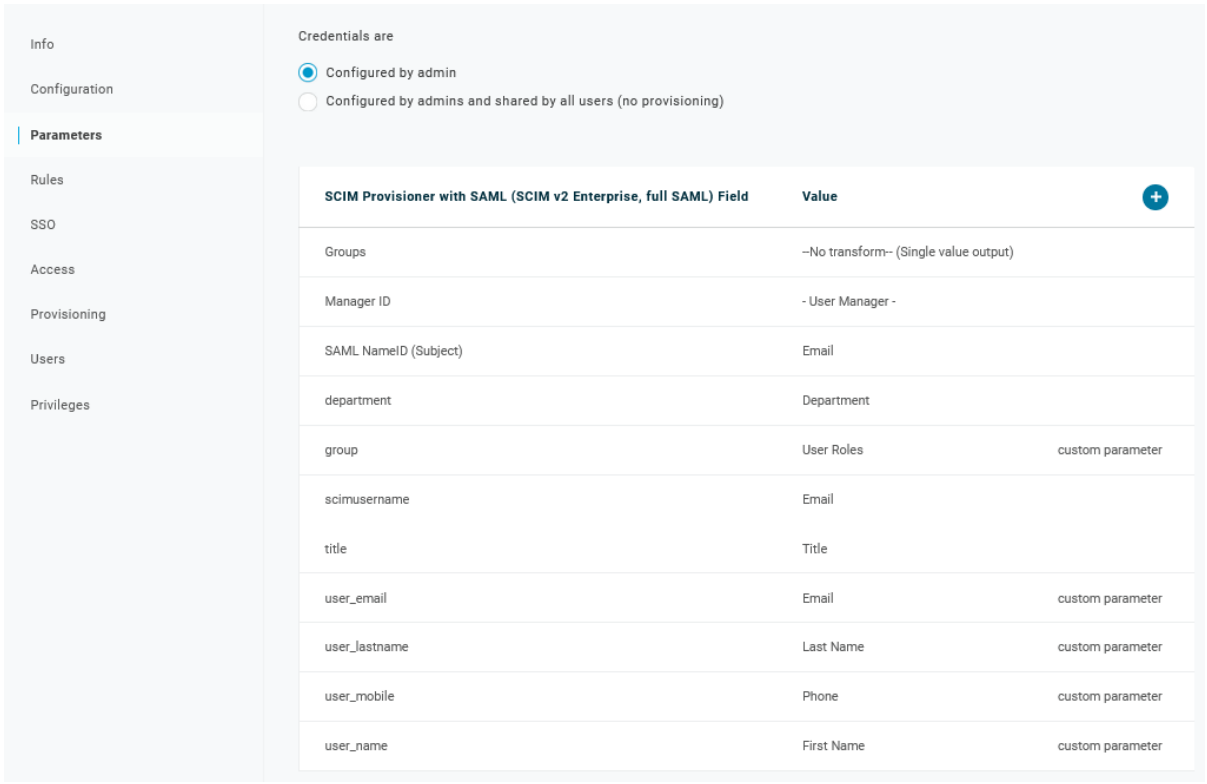


Podczas dodawania parametru upewnij się, że zaznaczone są zarówno opcje *Include in SAML assertion*, jak i *Include in User Provisioning*.

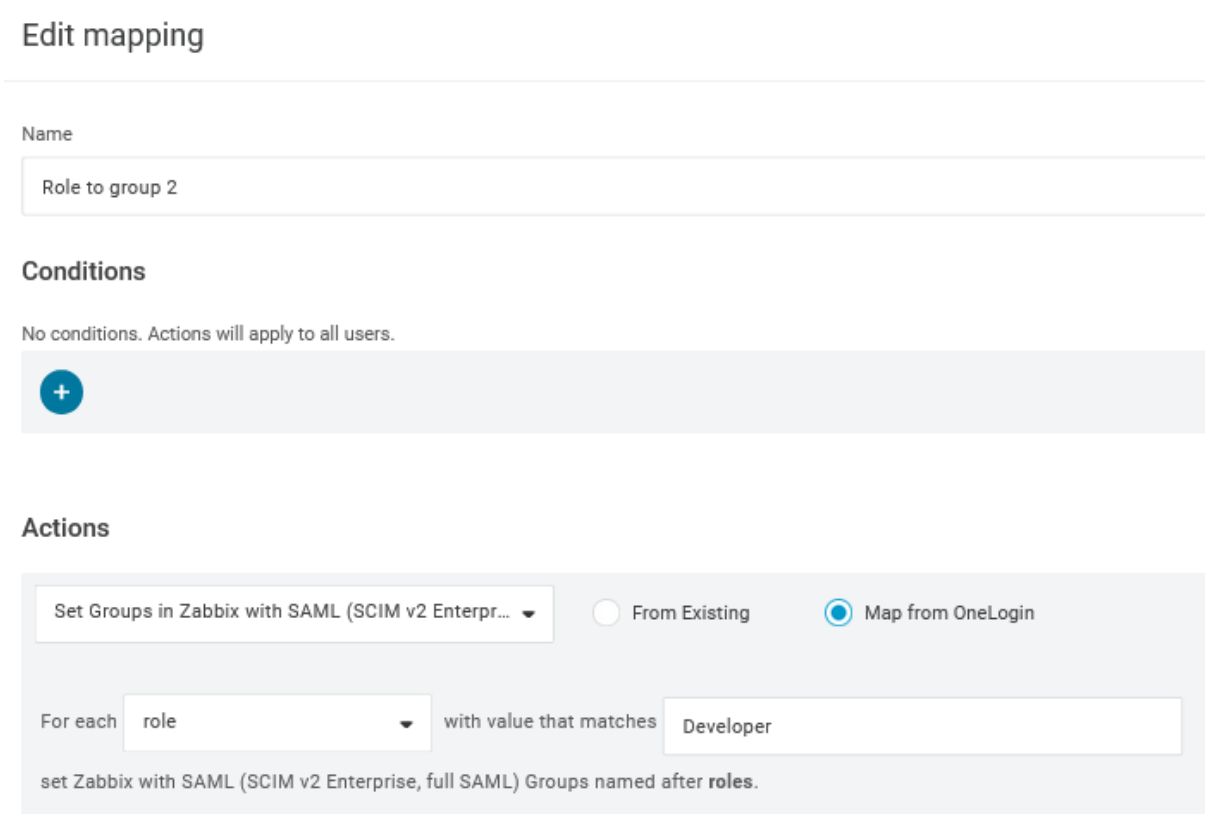
- Dodaj parametr `group`, który odpowiada rolom użytkowników w OneLogin. Role użytkowników będą przekazywane jako ciąg znaków, rozdzielony średnikiem `;`. Role użytkowników OneLogin będą używane do tworzenia grup użytkowników w Zabbix:



Sprawdź listę parametrów:



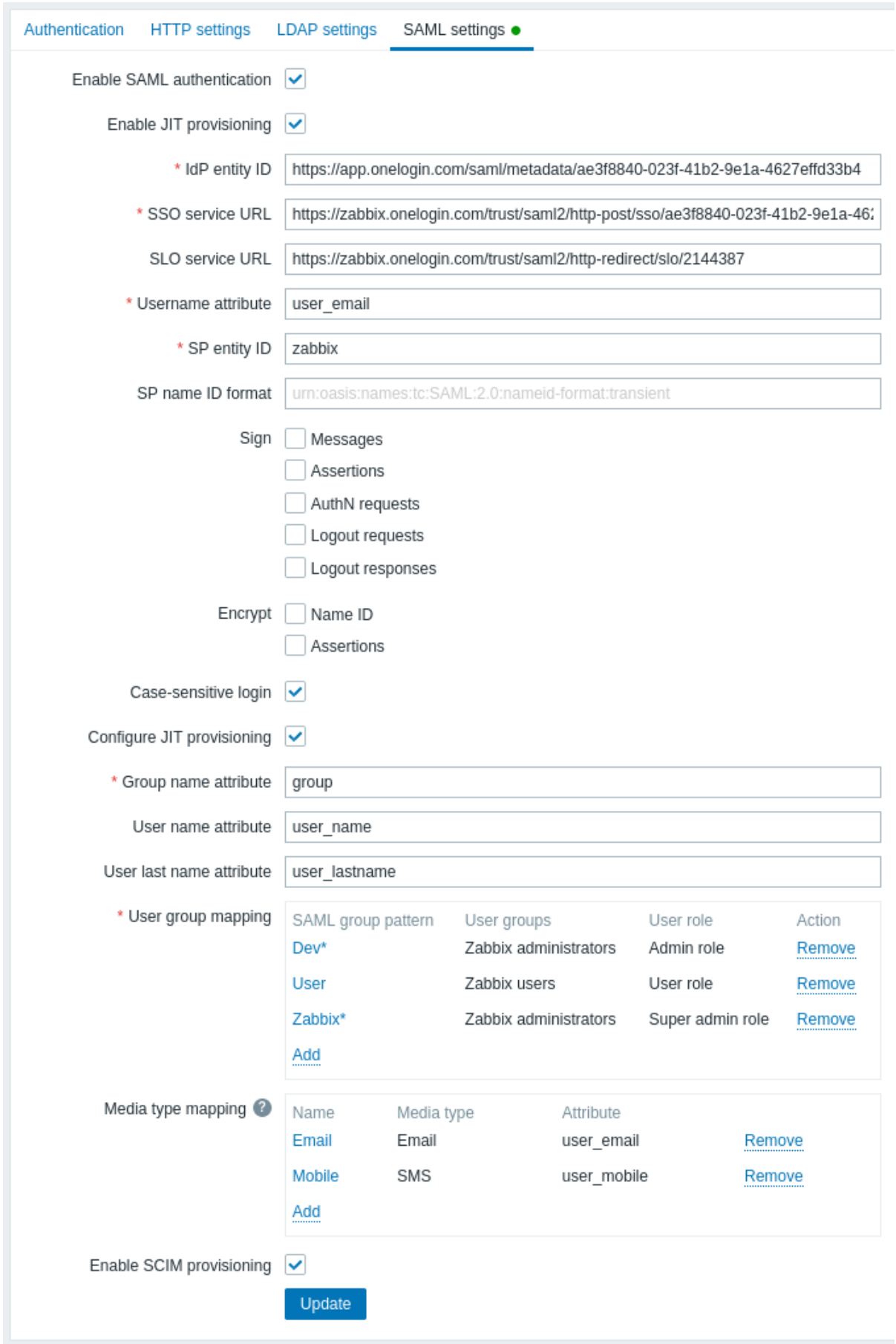
5. Na stronie *Rules* utwórz mapowania ról użytkowników do domyślnego parametru Groups.



Możesz użyć wyrażenia regularnego, aby przekazać określone role jako grupy. Nazwy ról nie powinny zawierać `;`, ponieważ OneLogin używa tego znaku jako separatora podczas wysyłania atrybutu z wieloma rolami.

Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i wypełnij opcje konfiguracji na podstawie konfiguracji OneLogin:



Pole Zabbix	Pole konfiguracji w OneLogin	Przykładowa wartość
IdP entity ID	Issuer URL (zobacz kartę SSO swojej aplikacji w OneLogin)	
SSO service URL	SAML 2.0 Endpoint (HTTP) (zobacz kartę SSO swojej aplikacji w OneLogin)	

Pole Zabbix	Pole konfiguracji w OneLogin	Przykładowa wartość
SLO service URL	SLO Endpoint (HTTP) (zobacz kartę SSO swojej aplikacji w OneLogin)	
Username attribute	Custom parameter	`user_email`
Group name attribute	Custom parameter	`group`
User name attribute	Custom parameter	`user_name`
User last name attribute	Custom parameter	`user_lastname`

Wymagana jest również konfiguracja mapowania grup użytkowników. Mapowanie mediów jest opcjonalne. Kliknij *Update*, aby zapisać te ustawienia.

2. Pobierz certyfikat udostępniony przez OneLogin i umieść go w `conf/certs` instalacji frontend Zabbix jako idp.crt.

Nadaj mu uprawnienia 644, uruchamiając:

```
chmod 644 idp.crt
```

Certyfikat można pobrać w OneLogin w sekcji *Applications* -> *SSO* -> kliknij *View details* pod bieżącym certyfikatem.

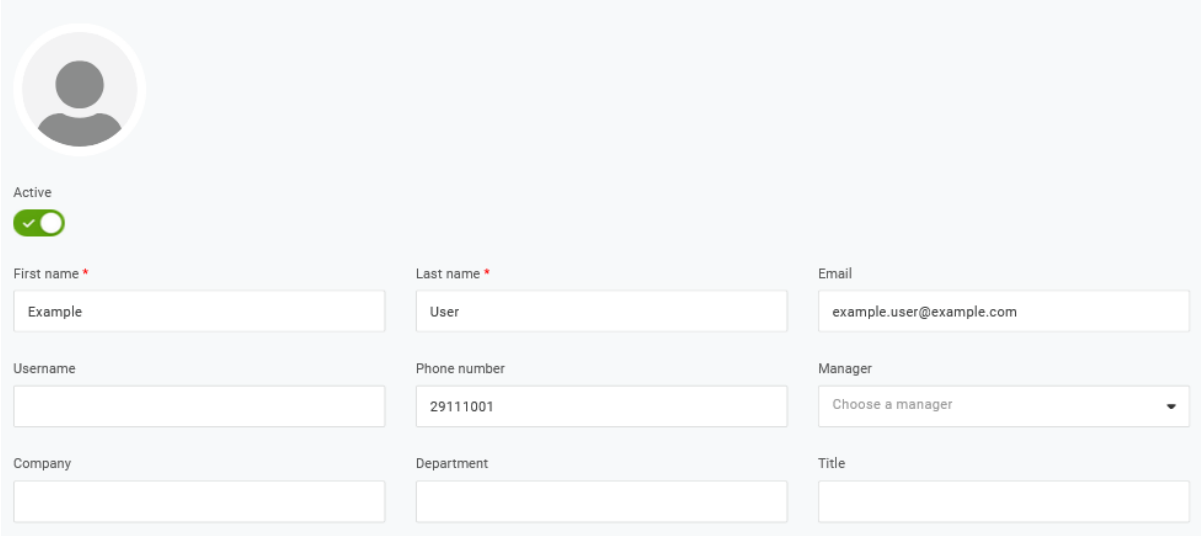
Można użyć innej nazwy i lokalizacji certyfikatu. W takim przypadku upewnij się, że do `conf/zabbix.conf.php` dodano następującą linię:

```
$SSO['IDP_CERT'] = 'path/to/certname.crt';
```

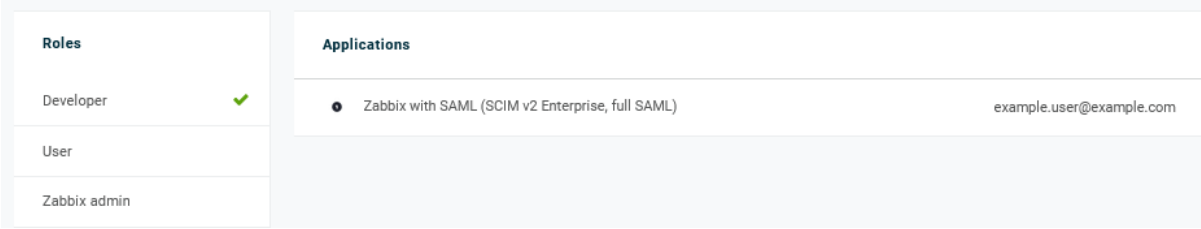
Provisioning użytkowników SCIM

Po włączeniu provisioning użytkowników możliwe jest teraz dodawanie/aktualizowanie użytkowników oraz ich ról w OneLogin i natychmiastowe provisionowanie ich do Zabbix.

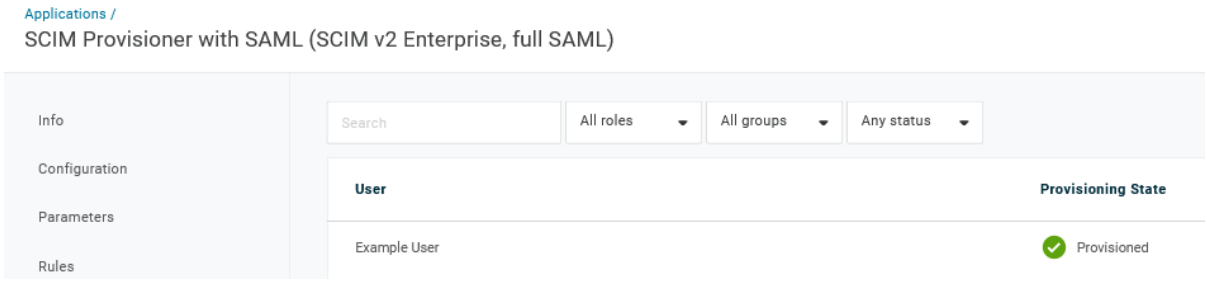
Na przykład możesz utworzyć nowego użytkownika:



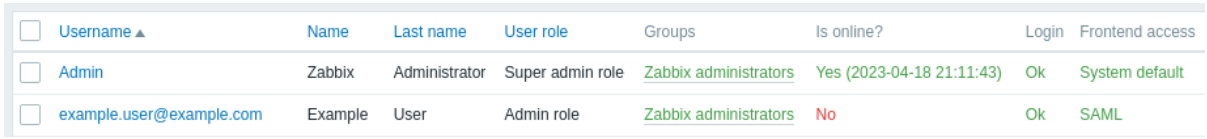
Dodaj go do roli użytkownika oraz aplikacji, która będzie provisionować użytkownika:



Po zapisaniu użytkownika zostanie on provisionowany do Zabbix. W Application -> Users możesz sprawdzić status provisioningu bieżących użytkowników aplikacji:



Jeśli provisioning zakończy się powodzeniem, użytkownik będzie widoczny na liście użytkowników Zabbix.



15 Konfigurowanie zaplanowanych raportów

Przegląd

Ta sekcja zawiera instrukcje dotyczące instalacji usługi internetowej Zabbix oraz konfiguracji Zabbix w celu umożliwienia generowania zaplanowanych raportów.

Instalacja

Nowy proces Zabbix web service oraz przeglądarka Google Chrome powinny zostać zainstalowane, aby umożliwić generowanie zaplanowanych raportów. Usługa web service może zostać zainstalowana na tej samej maszynie, na której zainstalowany jest serwer Zabbix, lub na innej maszynie. Przeglądarka Google Chrome powinna być zainstalowana na tej samej maszynie, na której zainstalowana jest usługa web service.

Oficjalny pakiet zabbix-web-service jest dostępny w repozytorium Zabbix. Przeglądarka Google Chrome nie jest dołączona do tych pakietów i musi zostać zainstalowana osobno.

Aby skompilować usługę web Zabbix ze źródeł, zobacz Installing Zabbix web service.

Po instalacji uruchom zabbix_web_service na maszynie, na której usługa web jest zainstalowana:

```
zabbix_web_service
```

Konfiguracja

Aby zapewnić prawidłową komunikację między wszystkimi zaangażowanymi elementami, upewnij się, że plik konfiguracyjny serwera oraz parametry konfiguracyjne frontend są poprawnie skonfigurowane.

Serwer Zabbix

Należy zaktualizować następujące parametry w pliku konfiguracyjnym serwera Zabbix: *WebServiceURL* i *StartReportWriters*.

WebServiceURL

Ten parametr jest wymagany do włączenia komunikacji z usługą web service. Adres URL powinien mieć format `http[s]://host:port/repo`

- Domyślnie usługa web service nasłuchuje na porcie 10053. Inny port można określić w pliku konfiguracyjnym usługi web service.
- Podanie ścieżki `/report` jest obowiązkowe (ścieżka jest zakodowana na stałe i nie można jej zmienić).

Przykład:

```
WebServiceURL=http://localhost:10053/report
```

Attention:
Zdecydowanie zaleca się skonfigurowanie szyfrowania między serwerem Zabbix a usługą web service przy użyciu certyfikatów. Domyślnie dane przesyłane między serwerem Zabbix a usługą web service nie są szyfrowane, co może prowadzić do nieautoryzowanego dostępu.

StartReportWriters

Ten parametr określa, ile procesów zapisu raportów powinno zostać uruchomionych. Jeśli nie jest ustawiony lub ma wartość 0, generowanie raportów jest wyłączone. W zależności od wymaganej liczby i częstotliwości raportów można włączyć od 1 do 100 procesów zapisu raportów.

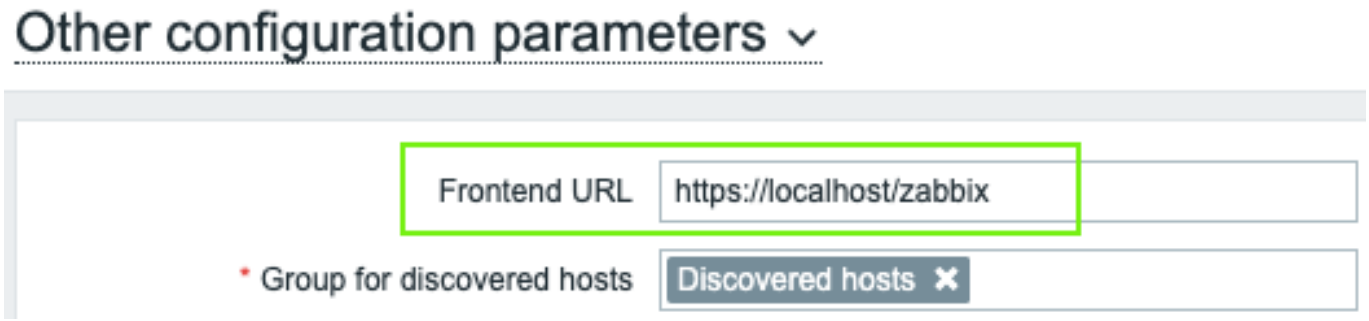
Przykład:

```
StartReportWriters=3
```

Zabbix frontend

Parametr *Frontend URL* powinien być ustawiony, aby umożliwić komunikację między Zabbix frontend a usługą internetową Zabbix:

- Przejdź do sekcji menu frontend *Administration* > *General* > *Other*
- Określ pełny adres URL interfejsu webowego Zabbix w parametrze *Frontend URL*



Note:
Po zakończeniu procedury konfiguracji możesz skonfigurować i wysłać raport testowy, aby upewnić się, że wszystko działa poprawnie.

16 Dodatkowe języki frontendu

Przegląd

Aby korzystać z języka innego niż angielski w interfejsie sieciowym Zabbix, jego ustawienia regionalne muszą być ustawione na serwerze WWW. Dodatkowo, wymagane jest rozszerzenie PHP gettext, aby tłumaczenia działały.

Installing locales

To list all installed languages, run:

```
locale -a
```

If some languages that are needed are not listed, open the */etc/locale.gen* file and uncomment the required locales. Since Zabbix uses UTF-8 encoding, you need to select locales with UTF-8 charset.

Now, run:

```
locale-gen
```

Restart the web server.

The locales should now be installed. It may be required to reload Zabbix frontend page in browser using Ctrl + F5 for new languages to appear.

Instalowanie Zabbix

Jeśli instalujesz Zabbix bezpośrednio z repozytorium git Zabbix, pliki tłumaczeń należy wygenerować ręcznie. Aby wygenerować pliki tłumaczeń, uruchom:

```
make gettext
locale/make_mo.sh
```

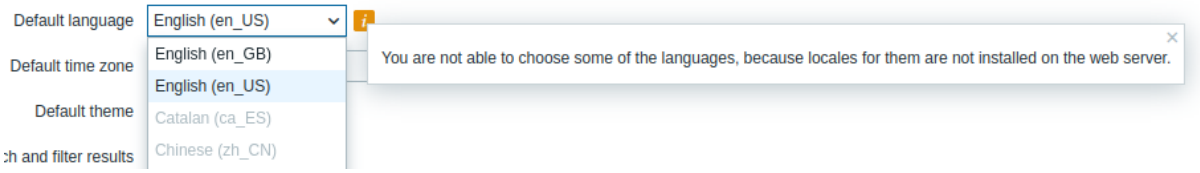
Ten krok nie jest potrzebny podczas instalowania Zabbix z pakietów lub plików tar.gz źródła.

Wybór języka

Istnieje kilka sposobów wyboru języka w interfejsie webowym Zabbix:

- Podczas instalacji interfejsu webowego — w kreatorze instalacji frontend. Wybrany język zostanie ustawiony jako domyślny dla systemu.
- Po instalacji domyślny język systemu można zmienić w sekcji menu *Administracja→Ogólne→GUI*.
- Język dla konkretnego użytkownika można zmienić w profilu użytkownika.

Jeśli ustawienia regionalne dla danego języka nie są zainstalowane na maszynie, ten język będzie wyszarzony w selektorze języka Zabbix. Obok selektora języka wyświetlana jest pomarańczowa ikona, jeśli brakuje co najmniej jednych ustawień regionalnych. Po kliknięciu tej ikony zostanie wyświetlony następujący komunikat: „Nie możesz wybrać niektórych języków, ponieważ ustawienia regionalne dla nich nie są zainstalowane na serwerze webowym.”



2 Konfiguracja procesów

Ta sekcja opisuje parametry plików konfiguracyjnych dla procesów Zabbix serwer, proxy, agent, agent 2 oraz innych procesów.

Aby przejść do odpowiedniej podsekcji, skorzystaj z paska bocznego.

1 Serwer Zabbix

Przegląd

Parametry obsługiwane przez plik konfiguracyjny serwera Zabbix (zabbix_server.conf) są wymienione w tej sekcji.

Parametry są podane bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
AlertScriptsPath	Lokalizacja niestandardowych skryptów alertów.
AllowRoot	Zezwala serwerowi działać jako ’root’.
AllowSoftwareUpdateCheck	Zezwala frontendowi Zabbix na otrzymywanie informacji o aktualizacjach oprogramowania z zabbix.com.
AllowUnsupportedDBVersions	Zezwala serwerowi na pracę z niewspieranymi wersjami bazy danych.
CacheSize	Rozmiar pamięci podręcznej konfiguracji.
CacheUpdateFrequency	Ten parametr określa, jak często Zabbix będzie wykonywać aktualizację pamięci podręcznej konfiguracji, w sekundach.
DBHost	Host (lub katalog gniazda) bazy danych Zabbix.
DBName	Nazwa bazy danych.
DBPassword	Hasło bazy danych.
DBPort	Port hosta bazy danych Zabbix.
DBSchema	Nazwa schematu bazy danych. Używane dla PostgreSQL.
DBSocket	Ścieżka do pliku gniazda MySQL.
DBUser	Użytkownik bazy danych.
DBTLSConnect	Ustawienie tej opcji na określoną wartość wymusza użycie połączenia TLS z bazą danych.
DBTLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu bazy danych.
DBTLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat serwera Zabbix do uwierzytelniania w bazie danych.
DBTLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny do uwierzytelniania w bazie danych.
DBTLSCipher	Lista szyfrów szyfrowania, które serwer Zabbix dopuszcza dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.
DBTLSCipher13	Lista zestawów szyfrów szyfrowania, które serwer Zabbix dopuszcza dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, od wersji 8.0.16.
DebugLevel	Określa poziom debugowania.
EnableGlobalScripts	Włącza globalne skrypty na serwerze Zabbix.
ExportDir	Katalog do eksportu zdarzeń, historii i trendów w czasie rzeczywistym w formacie JSON rozdzielanym znakami nowej linii. Jeśli ustawiony, włącza eksport w czasie rzeczywistym.
ExportFileSize	Maksymalny rozmiar pojedynczego pliku eksportu w bajtach.
ExportType	Lista typów encji rozdzielonych przecinkami (zdarzenia, historia, trendy) do eksportu w czasie rzeczywistym (domyślnie wszystkie typy).
ExternalScripts	Lokalizacja zewnętrznych skryptów.
Fping6Location	Lokalizacja fping6.
FpingLocation	Lokalizacja fping.

Parameter	Description
FrontendAllowedIP	Lista adresów IP lub zakresów CIDR rozdzielonych przecinkami, z których dozwolone jest łączenie się z frontendem.
HANodeName	Nazwa węzła klastra wysokiej dostępności.
HistoryCacheSize	Rozmiar pamięci podręcznej historii.
HistoryIndexCacheSize	Rozmiar pamięci podręcznej indeksu historii.
HistoryStorageDateIndex	Włącza wstępne przetwarzanie wartości historii w magazynie historii, aby przechowywać wartości w różnych indeksach na podstawie daty.
HistoryStorageURL	Adres URL HTTP[S] magazynu historii.
HistoryStorageTypes	Lista typów wartości rozdzielonych przecinkami, które mają być wysyłane do magazynu historii.
HousekeepingFrequency	Ten parametr określa, jak często Zabbix będzie wykonywać procedurę housekeeping, w godzinach.
Include	W pliku konfiguracyjnym można dołączać pojedyncze pliki lub wszystkie pliki z katalogu.
JavaGateway	Adres IP (lub nazwa hosta) bramy Java gateway Zabbix.
JavaGatewayPort	Port, na którym nasłuchuje brama Java gateway Zabbix.
ListenBacklog	Maksymalna liczba oczekujących połączeń w kolejce TCP.
ListenIP	Lista adresów IP rozdzielonych przecinkami, na których trapper powinien nasłuchiwać.
ListenPort	Port nasłuchiwania dla trappera.
LoadModule	Moduł do załadowania podczas uruchamiania serwera.
LoadModulePath	Pełna ścieżka do lokalizacji modułów serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogSlowQueries	Określa, jak długo zapytanie do bazy danych może być wykonywane, zanim zostanie zapisane w dzienniku, w milisekundach.
LogType	Typ wyjścia dziennika.
MaxConcurrentChecksPerPoller	Maksymalna liczba asynchronicznych sprawdzeń, które mogą być jednocześnie wykonywane przez każdego pollera HTTP agent, pollera agent lub pollera SNMP.
MaxHousekeeperDelete	Nie więcej niż 'MaxHousekeeperDelete' wierszy (odpowiadających [tablename], [field], [value]) zostanie usuniętych w ramach jednego zadania w jednym cyklu housekeeping.
NodeAddress	Adres IP lub nazwa hosta z opcjonalnym portem, służące do nadpisania sposobu, w jaki frontend powinien łączyć się z serwerem.
PidFile	Nazwa pliku PID.
ProblemHousekeepingFrequency	Określa, jak często Zabbix będzie usuwać problemy dla usuniętych wyzwalaczy.
ProxyConfigFrequency	Określa, jak często serwer Zabbix wysyła dane konfiguracyjne do proxy Zabbix.
ProxyDataFrequency	Określa, jak często serwer Zabbix pobiera dane historii z proxy Zabbix.
ServiceManagerSyncFrequency	Określa, jak często Zabbix będzie synchronizować konfigurację menedżera usług.
SMSDevices	Lista plików modemów rozdzielonych przecinkami, które mogą być używane przez serwer Zabbix.
SNMPTrapperFile	Plik tymczasowy używany do przekazywania danych z demona SNMP trap do serwera.
SocketDir	Katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.
SourceIP	Źródłowy adres IP.
SSHKeyLocation	Lokalizacja kluczy publicznych i prywatnych do sprawdzeń i działań SSH.
SSLCertLocation	Lokalizacja plików certyfikatu klienta SSL do uwierzytelniania klienta.
SSLKeyLocation	Lokalizacja plików prywatnego klucza SSL do uwierzytelniania klienta.
SSLCALocation	Nadpisuje lokalizację plików urzędu certyfikacji (CA) do weryfikacji certyfikatu serwera SSL.
StartAgentPollers	Liczba wstępnie utworzonych instancji asynchronicznych pollerów agent.
StartAlerters	Liczba wstępnie utworzonych instancji alerterów.
StartBrowserPollers	Liczba wstępnie utworzonych instancji pollerów pozycji przeglądarki.
StartConnectors	Liczba wstępnie utworzonych instancji procesów roboczych konektorów.
StartDBSyncers	Liczba wstępnie utworzonych instancji synchronizatorów historii.
StartDiscoverers	Liczba wstępnie utworzonych instancji procesów roboczych wykrywania.
StartEscalators	Liczba wstępnie utworzonych instancji eskalatorów.
StartHistoryPollers	Liczba wstępnie utworzonych instancji pollerów historii.
StartHTTPAgentPollers	Liczba wstępnie utworzonych instancji asynchronicznych pollerów HTTP agent.
StartHTTPPollers	Liczba wstępnie utworzonych instancji pollerów HTTP.
StartIPMIPollers	Liczba wstępnie utworzonych instancji pollerów IPMI.
StartJavaPollers	Liczba wstępnie utworzonych instancji pollerów Java.
StartLLDProcessors	Liczba wstępnie utworzonych instancji procesów roboczych wykrywania niskiego poziomu (LLD).
StartODBCPollers	Liczba wstępnie utworzonych instancji pollerów ODBC.
StartPingers	Liczba wstępnie utworzonych instancji pingerów ICMP.
StartPollersUnreachable	Liczba wstępnie utworzonych instancji pollerów dla niedostępnych hostów (w tym IPMI i Java).
StartPollers	Liczba wstępnie utworzonych instancji pollerów.
StartPreprocessors	Liczba wcześniej uruchomionych instancji procesów roboczych wstępnego przetwarzania.

Parameter	Description
StartProxyPollers	Liczba wstępnie utworzonych instancji pollerów dla pasywnych proxy.
StartReportWriters	Liczba wstępnie utworzonych instancji generatorów raportów.
StartSNMPPollers	Liczba wstępnie utworzonych instancji asynchronicznych pollerów SNMP.
StartSNMPTrapper	Jeśli ustawione na 1, zostanie uruchomiony proces SNMP trapper.
StartTimers	Liczba wstępnie utworzonych instancji timerów.
StartTrappers	Liczba wstępnie utworzonych instancji trapperów.
StartVMwareCollectors	Liczba wstępnie utworzonych instancji kolektorów VMware.
StatsAllowedIP	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS zewnętrznych instancji Zabbix. Żądanie statystyk zostanie zaakceptowane tylko z adresów wymienionych tutaj.
Timeout	Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy Zabbix, agent, usługą web oraz dla sprawdzeń SNMP (z wyjątkiem pozycji SNMP `walk[OID]` i `get[OID]`).
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu peer, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCRLFile	Pełna ścieżka do pliku zawierającego certyfikaty unieważnione. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSFrontendAccept	Jakie połączenia przychodzące akceptować z frontend.
TLSFrontendCertIssuer	Dozwolony wystawca certyfikatu frontend.
TLSFrontendCertSubject	Dozwolony podmiot certyfikatu frontend.
TLSKeyFile	Pełna ścieżka do pliku zawierającego prywatny klucz serwera, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSListen	Steruje TLS na gnieździe trappera.
TmpDir	Katalog tymczasowy.
TrapperTimeout	Określa limit czasu w sekundach dla: - pobierania danych historycznych z proxy Zabbix; - wysyłania danych konfiguracyjnych do proxy Zabbix; - wykonywania globalnego skryptu lub zdalnego polecenia na serwerze Zabbix.
TrendCacheSize	Rozmiar pamięci podręcznej trendów.
TrendFunctionCacheSize	Rozmiar pamięci podręcznej funkcji trendów.
UnavailableDelay	Określa, jak często host jest sprawdzany pod kątem dostępności w okresie niedostępności.
UnreachableDelay	Określa, jak często host jest sprawdzany pod kątem dostępności w okresie niedostępności sieciowej.
UnreachablePeriod	Określa, po ilu sekundach niedostępności host jest traktowany jako niedostępny.
User	Odrzuca uprawnienia do określonego, istniejącego użytkownika w systemie.
ValueCacheSize	Rozmiar pamięci podręcznej wartości historii.
Vault	Określa dostawcę vault.
VaultDBPath	Określa lokalizację, z której poświadczenia bazy danych powinny być pobierane za pomocą kluczy.
VaultPrefix	Niestandardowy prefiks dla ścieżki lub zapytania vault.
VaultTLSCertFile	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta.
VaultTLSKeyFile	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta.
VaultToken	Token uwierzytelniania HashiCorp vault.
VaultURL	Adres URL HTTP[S] serwera vault.
VMwareCacheSize	Rozmiar pamięci współdzielonej do przechowywania danych VMware.
VMwareFrequency	Opóźnienie w sekundach między pobieraniem danych z pojedynczej usługi VMware.
VMwarePerfFrequency	Opóźnienie w sekundach między pobieraniem statystyk liczników wydajności z pojedynczej usługi VMware.

Parameter	Description
VMwareTimeout	Maksymalna liczba sekund, przez którą kolektor vmware będzie czekał na odpowiedź od usługi VMware.
WebDriverURL	Adres URL HTTP[S] interfejsu WebDriver.
WebServiceURL	Adres URL HTTP[S] do usługi web Zabbix w formacie <host:port>/report.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że parametr jest obowiązkowy.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia demona, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

AlertScriptsPath

Lokalizacja niestandardowych skryptów alertów (zależy od zmiennej instalacyjnej *datadir* ustawianej podczas kompilacji).

Wartość domyślna: `/usr/local/share/zabbix/alertscripts`

AllowRoot

Zezwala serwerowi na działanie jako „root”. Jeśli opcja jest wyłączona, a serwer zostanie uruchomiony przez „root”, serwer spróbuje zamiast tego przełączyć się na użytkownika „zabbix”. Nie ma wpływu, jeśli został uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowSoftwareUpdateCheck

Zezwala interfejsowi użytkownika Zabbix na otrzymywanie informacji o aktualizacjach oprogramowania z zabbix.com.

Domyślnie: 1
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowUnsupportedDBVersions

Zezwala serwerowi działać z nieobsługiwanymi wersjami bazy danych.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

CacheSize

Rozmiar pamięci podręcznej konfiguracji, w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych hostów, pozycji i wyzwalaczy.

Domyślnie: 32M
 Zakres: 128K-64G

CacheUpdateFrequency

Ten parametr określa, jak często Zabbix będzie aktualizował pamięć podręczną konfiguracji, w sekundach. Zobacz także opcje runtime control.

Wartość domyślna: 10
 Zakres: 1-3600

DBHost

Host (lub katalog gniazda) bazy danych Zabbix.

Dla MySQL:

- `localhost` lub pusty ciąg używa domyślnego gniazda domeny UNIX.

Dla PostgreSQL:

- `localhost` jest rozwiązywany przez DNS (zwykle do 127.0.0.1).
- Pusty ciąg używa domyślnego gniazda domeny UNIX.
- Ścieżka (na przykład `/var/run/pgbouncer`) powoduje użycie gniazda UNIX w tej ścieżce.
- Lista rozdzielona przecinkami może zawierać wiele wartości `host:port`. Każdy host jest sprawdzany po kolei, aż zostanie ustanowione połączenie do odczytu i zapisu. Przykład: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[::1]:3005`

Domyślnie: `localhost`

DBName

Nazwa bazy danych.

Wymagane: Tak

DBPassword

Hasło bazy danych. Zakomentuj ten wiersz, jeśli hasło nie jest używane.

DBPort

Port hosta bazy danych Zabbix (zobacz DBHost).[3]

Domyślnie dla MySQL: 3306

Domyślnie dla PostgreSQL: 5432

Zakres: 1024-65535

DBSchema

Nazwa schematu bazy danych. Używane dla PostgreSQL.

DBSocket

Ścieżka do pliku socket MySQL.[3]

DBUser

Użytkownik bazy danych.

DBTLSConnect

Ustawienie tej opcji na jedną z poniższych wartości wymusza użycie połączenia TLS z bazą danych:
required - połącz przy użyciu TLS
verify_ca - połącz przy użyciu TLS i zweryfikuj certyfikat
verify_full - połącz przy użyciu TLS, zweryfikuj certyfikat oraz sprawdź, czy tożsamość bazy danych określona przez DBHost odpowiada jej certyfikatowi

W przypadku MySQL, od wersji 5.7.11, oraz PostgreSQL obsługiwane są następujące wartości: `required`, `verify_ca`, `verify_full`.
W przypadku MariaDB, od wersji 10.2.6, obsługiwane są wartości `required` i `verify_full`.
Domyślnie opcja nie jest ustawiona, a zachowanie zależy od konfiguracji bazy danych.

DBTLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnego urzędu certyfikacji (CA) używane do weryfikacji certyfikatu bazy danych.

Obowiązkowe: nie (tak, jeśli DBTLSConnect ustawiono na *verify_ca* lub *verify_full*)

DBTLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat serwer Zabbix używany do uwierzytelniania w bazie danych.

DBTLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.

DBTLSCipher

Lista szyfrów szyfrowania, które serwer Zabbix dopuszcza dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.

DBTLSCipher13

Lista szyfrów szyfrowania, na które serwer Zabbix zezwala dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, począwszy od wersji 8.0.16.

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).
Zobacz także opcje kontroli w czasie działania.

Domyślnie: 3
 Zakres: 0-5

EnableGlobalScripts

Włącz skrypty globalne na serwerze Zabbix.
 Uwaga: wykonywanie skryptów globalnych jest domyślnie włączone. Jednak w przypadku nowych instalacji począwszy od Zabbix 7.0 parametr EnableGlobalScripts jest jawnie ustawiony na 0 (wyłączony).

Domyślnie: 1
 Wartości: 0 - wyłącz; 1 - włącz

ExportDir

Katalog do eksportu w czasie rzeczywistym zdarzeń, historii i trendów w formacie JSON rozdzielanym znakami nowej linii. Jeśli jest ustawiony, włącza eksport w czasie rzeczywistym.

ExportFileSize

Maksymalny rozmiar pojedynczego pliku eksportu w bajtach. Używane do rotacji, jeśli ustawiono `ExportDir`.

Wartość domyślna: 1G
 Zakres: 1M-1G

ExportType

Lista typów encji rozdzielonych przecinkami (zdarzenia, historia, trendy) dla eksportu w czasie rzeczywistym (domyślnie wszystkie typy). Prawidłowe tylko wtedy, gdy ustawiono ExportDir.
Uwaga: jeśli określono ExportType, ale nie określono ExportDir, jest to błąd konfiguracji i serwer nie uruchomi się.

Przykład eksportu historii i trendów:

```
ExportType=history,trends
```

Przykład tylko dla eksportu zdarzeń:

```
ExportType=events
```

ExternalScripts

Lokalizacja skryptów zewnętrznych (zależy od zmiennej instalacyjnej `datadir` ustawianej w czasie kompilacji).

Domyślnie: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Lokalizacja fping6. Upewnij się, że plik binarny fping6 ma właściciela root oraz ustawioną flagę SUID. Pozostaw puste (”Fping6Location=”), jeśli narzędzie fping potrafi przetwarzać adresy IPv6.

Domyślnie: `/usr/sbin/fping6`

FpingLocation

Lokalizacja fping. Upewnij się, że plik binarny fping ma właściciela root oraz ustawioną flagę SUID.

Domyślnie: `/usr/sbin/fping`

FrontendAllowedIP

Lista adresów IP lub zakresów CIDR, z których dozwolone jest łączenie się z frontend. Jeśli ten parametr jest ustawiony, połączenie z frontend zostanie zaakceptowane tylko z adresów wymienionych tutaj. Domyślnie dla żądań frontend akceptowane są wszystkie połączenia. Jeśli obsługa IPv6 jest włączona, wtedy ’127.0.0.1’, ’::127.0.0.1’, ’::ffff:127.0.0.1’ są traktowane jednakowo, a ’::/0’ zezwoli na dowolny adres IPv4 lub IPv6. ’0.0.0.0/0’ może być użyte do zezwolenia na dowolny adres IPv4.

Przykład:

```
FrontendAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

HANodeName

Nazwa węzła klastra wysokiej dostępności. Jeśli pole jest puste, serwer działa w trybie samodzielnym i tworzony jest węzeł z pustą nazwą.

HistoryCacheSize

Rozmiar pamięci podręcznej historii, w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych historii.

Domyślnie: 16M
 Zakres: 128K-16G

HistoryIndexCacheSize

Rozmiar pamięci podręcznej indeksu historii, w bajtach. Rozmiar pamięci współdzielonej do indeksowania danych historii przechowywanych w pamięci podręcznej historii. Rozmiar pamięci podręcznej indeksu wymaga około 100 bajtów do przechowywania w pamięci podręcznej jednej pozycji.

Domyślnie: 4M
 Zakres: 128K-16G

HistoryStorageDateIndex

Włącz przetwarzanie wstępne wartości historii w magazynie historii, aby przechowywać wartości w różnych indeksach na podstawie daty.

Domyślnie: 0
 Wartości: 0 - wyłączone; 1 - włączone

HistoryStorageURL

Adres URL HTTP[S] magazynu historii. Ten parametr jest używany do konfiguracji Elasticsearch.

HistoryStorageTypes

Lista typów wartości rozdzielona przecinkami, które mają być wysyłane do magazynu historii. Ten parametr jest używany podczas konfiguracji Elasticsearch.

Domyślnie: `uint,dbl,str,log,text`

HousekeepingFrequency

Ten parametr określa, jak często serwer będzie wykonywał procedurę housekeeping, w godzinach. Housekeeping polega na usuwaniu nieaktualnych informacji z bazy danych.
Uwaga: Aby zapobiec przeciążeniu housekeepera (na przykład, gdy okresy historii i trendów są znacznie skrócone), w jednym cyklu housekeeping usuwanych jest nie więcej niż 4 razy HousekeepingFrequency godzin nieaktualnych informacji dla każdej pozycji. Zatem jeśli HousekeepingFrequency wynosi 1, w jednym cyklu zostanie usuniętych nie więcej niż 4 godziny nieaktualnych informacji (licząc od najstarszego wpisu).
Uwaga: Aby zmniejszyć obciążenie podczas uruchamiania serwera, housekeeping jest odraczany o 30 minut po starcie serwera. Zatem jeśli HousekeepingFrequency wynosi 1, pierwsza procedura housekeeping po uruchomieniu serwera zostanie wykonana po 30 minutach, a następnie będzie powtarzana z godzinnym opóźnieniem.
Można wyłączyć automatyczny housekeeping, ustawiając HousekeepingFrequency na 0. W takim przypadku procedura housekeeping może być uruchomiona wyłącznie za pomocą opcji sterowania w czasie działania *housekeeper_execute*, a okres nieaktualnych informacji usuwanych w jednym cyklu housekeeping wynosi 4 razy okres od ostatniego cyklu housekeeping, ale nie mniej niż 4 godziny i nie więcej niż 4 dni.
Zobacz także opcje runtime control.

Domyślnie: 1
 Zakres: 0-24

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym. Aby uwzględnić tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców. Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Adres IP (lub nazwa hosta) bramy Java gateway Zabbix. Wymagane tylko wtedy, gdy uruchomione są odpytywacze Java.

JavaGatewayPort

Port, na którym nasłuchuje Zabbix Java gateway.

Domyślnie: 10052
 Zakres: 1024-32767

ListenBacklog

Maksymalna liczba oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na stałe i zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać po cichu obcięte do „maksimum określonego przez implementację”.

Domyślnie: `SOMAXCONN`
 Zakres: 0 - INT_MAX

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których trapper powinien nasłuchiwać.
Jeśli ten parametr nie jest określony, trapper będzie nasłuchiwał na wszystkich interfejsach sieciowych.

Wartość domyślna: `0.0.0.0`

ListenPort

Port nasłuchiwania dla trappera.

Domyślnie: 10051
 Zakres: 1024-32767

LoadModule

Moduł do załadowania podczas uruchamiania serwera. Moduły są używane do rozszerzania funkcjonalności serwera. Moduł musi znajdować się w katalogu określonym przez LoadModulePath lub ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od ’/’), wtedy LoadModulePath jest ignorowane.
Formaty:
LoadModule=<module.so>
Loa
jest uwzględnienie wielu parametrów LoadModule.

LoadModulePath

Pełna ścieżka do lokalizacji modułów serwera. Wartość domyślna zależy od opcji kompilacji.

LogFile

Nazwa pliku dziennika.

Obowiązkowe: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpocznie się jego zapis od nowa.

Wartość domyślna: 1
 Zakres: 0-1024
 Wymagane: Tak, jeśli LogType jest ustawiony na *file*; w przeciwnym razie nie

LogSlowQueries

Określa, jak długo może trwać zapytanie do bazy danych, zanim zostanie zapisane w logu, w milisekundach.
0 - nie zapisuj wolnych zapytań w logu.
Ta opcja jest włączana przy ustawieniu DebugLevel=3.

Domyślnie: 0
 Zakres: 0-3600000

LogType

Typ wyjścia logu:
file — zapisuje log do pliku określonego parametrem LogFile;
system — zapisuje log do syslog;
console — zapisuje log na standardowe wyjście.

Wartość domyślna: `file`

MaxConcurrentChecksPerPoller

Maksymalna liczba asynchronicznych kontroli, które mogą być wykonywane jednocześnie przez każdy proces odpytywania HTTP agent, agent lub SNMP. Zobacz StartHTTPAgentPollers, StartAgentPollers oraz StartSNMPPollers.

Wartość domyślna: 1000
 Zakres: 1-1000

MaxHousekeeperDelete

Nie więcej niż `MaxHousekeeperDelete` wierszy (odpowiadających [tablename], [field], [value]) zostanie usuniętych w ramach jednego zadania w jednym cyklu housekeeping.
Jeśli ustawiono wartość 0, nie będzie stosowany żaden limit. W takim przypadku musisz dokładnie wiedzieć, co robisz, aby nie przeciążyć bazy danych.[2]
Ten parametr ma zastosowanie tylko do usuwania danych pozostawionych przez już usunięte pozycje.

Domyślnie: 5000
 Zakres: 0-1000000

NodeAddress

Adres IP lub nazwa hosta z opcjonalnym portem, aby zastąpić sposób, w jaki frontend powinien łączyć się z serwerem.
Format: <adres>[:<port>]

Jeśli adres IP lub nazwa hosta nie są ustawione, zostanie użyta wartość ListenIP. Jeśli ListenIP nie jest ustawione, zostanie użyta wartość `localhost`.
Jeśli port nie jest ustawiony, zostanie użyta wartość ListenPort. Jeśli ListenPort nie jest ustawione, zostanie użyta wartość 10051.

Tę opcję można zastąpić adresem określonym w konfiguracji frontendu.

Zobacz także: parametr HANodeName; Włączanie wysokiej dostępności.

Wartość domyślna: `localhost:10051`

PidFile

Nazwa pliku PID.

Wartość domyślna: `/tmp/zabbix_server.pid`

ProblemHousekeepingFrequency

Określa, jak często Zabbix będzie usuwać problemy dla usuniętych wyzwalaczy, w sekundach.

Domyślnie: 60
 Zakres: 1-3600

ProxyConfigFrequency

Określa, jak często serwer Zabbix wysyła dane konfiguracyjne do proxy Zabbix, w sekundach. Używane tylko dla proxy w trybie pasywnym.

Domyślnie: 10
 Zakres: 1-604800

ProxyDataFrequency

Określa, jak często serwer Zabbix żąda danych historii od proxy Zabbix, w sekundach. Używane tylko dla proxy w trybie pasywnym.

Domyślnie: 1
 Zakres: 1-3600

ServiceManagerSyncFrequency

Określa, jak często Zabbix będzie synchronizować konfigurację menedżera usług, w sekundach.

Domyślnie: 60
 Zakres: 1-3600

SMSDevices

Lista plików modemów oddzielonych przecinkami, które mogą być używane przez serwer Zabbix.
Wysyłanie wiadomości SMS nie jest możliwe, jeśli ten parametr nie jest ustawiony.

Przykład:

```
SMSDevices=/dev/ttyUSB0,/dev/ttyUSB1
```

SNMPTrapperFile

Plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do serwera.
Musi być taki sam jak w pliku konfiguracyjnym zabbix_trap_receiver.pl lub SNMPTT.

Domyślnie: `/tmp/zabbix_traps.tmp`

SocketDir

Katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.

Wartość domyślna: `/tmp`

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do Zabbix proxy i Zabbix agent
- połączeń bezagentowych (VMware, SSH, JMX, SNMP, Telnet i prostych testów)
- połączeń HTTP agent
- żądań HTTP JavaScript dla pozycji skryptowych
- żądań HTTP JavaScript dla preprocessing
- wysyłania e-maili z powiadomieniami (połączenia do serwera SMTP)
- powiadomień webhook (połączenia HTTP JavaScript)
- połączeń do Vault

SSHKeyLocation

Lokalizacja kluczy publicznych i prywatnych dla kontroli SSH oraz akcji.

SSLCertLocation

Lokalizacja plików certyfikatów klienta SSL do uwierzytelniania klienta.
Ten parametr jest używany tylko w monitorowaniu WWW.

SSLKeyLocation

Lokalizacja plików klucza prywatnego SSL do uwierzytelniania klienta.
Ten parametr jest używany tylko w monitorowaniu WWW.

SSLCALocation

Nadpisuje lokalizację plików urzędu certyfikacji (CA) używanych do weryfikacji certyfikatu SSL serwera. Jeśli nie jest ustawiony, użyty zostanie katalog systemowy.
Zwróć uwagę, że wartość tego parametru zostanie ustawiona jako opcja libcurl CURLOPT_CAPATH. W przypadku wersji libcurl wcześniejszych niż 7.42.0 ma to skutek tylko wtedy, gdy libcurl został skompilowany do użycia OpenSSL. Więcej informacji można znaleźć na stronie cURL.
Ten parametr jest używany w monitorowaniu WWW oraz przy uwierzytelnianiu SMTP.

StartAgentPollers

Liczba wstępnie rozwidlonych instancji pollerów Zabbix agent. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartAlerters

Liczba wstępnie rozwidlonych instancji alerterów.

Domyślnie: 3
 Zakres: 1-100

StartBrowserPollers

Liczba wstępnie rozwidlonych instancji pollerów pozycji przeglądarki.

Domyślnie: 1
 Zakres: 0-1000

StartConnectors

Liczba wstępnie rozwidlonych instancji procesów roboczych connectora. Proces menedżera connectora jest uruchamiany automatycznie po uruchomieniu procesu roboczego connectora.

Domyślnie: 0
 Zakres: 0-1000

StartDBSyncers

Liczba wstępnie rozwidlonych instancji synchronizatorów historii.
Uwaga: Zachowaj ostrożność podczas zmiany tej wartości, jej zwiększenie może przynieść więcej szkody niż pożytku. Orientacyjnie, wartość domyślna powinna wystarczyć do obsługi do 4000 NVPS.

Domyślnie: 4
 Zakres: 1-100

StartDiscoverers

Liczba wstępnie utworzonych instancji workerów discovery[1].

Domyślnie: 5
 Zakres: 0-1000

StartEscalators

Liczba wstępnie rozwidlonych instancji escalators.

Domyślnie: 1
 Zakres: 1-100

StartHistoryPollers

Liczba wstępnie utworzonych instancji history pollers.
Wymagane tylko dla obliczanych kontroli.

Domyślnie: 5
 Zakres: 0-1000

StartHTTPAgentPollers

Liczba wstępnie rozwidlonych instancji HTTP agent pollerów. Zobacz MaxConcurrentChecksPerPoller.

Wartość domyślna: 1
 Zakres: 0-1000

StartHTTPPollers

Liczba wstępnie rozwidlonych instancji HTTP pollers[1].

Domyślnie: 1
 Zakres: 0-1000

StartIPMIPollers

Liczba wstępnie rozwidlonych instancji modułów odpytujących IPMI.

Domyślnie: 0
 Zakres: 0-1000

StartJavaPollers

Liczba wstępnie rozwidlonych instancji Java pollers[1].

Domyślnie: 0
 Zakres: 0-1000

StartLLDProcessors

Liczba wstępnie rozwidlonych instancji workerów niskopoziomowego wykrywania (LLD)[1].
Proces menedżera LLD jest uruchamiany automatycznie po uruchomieniu workera LLD.

Domyślnie: 2
 Zakres: 1-100

StartODBCPollers

Liczba wstępnie rozwidlonych instancji ODBC pollers[1].

Domyślnie: 1
 Zakres: 0-1000

StartPingers

Liczba wstępnie rozwidlonych instancji ICMP pingers[1].

Domyślnie: 1
 Zakres: 0-1000

StartPollersUnreachable

Liczba wstępnie rozwidlonych instancji pollerów dla nieosiągalnych hostów (w tym IPMI i Java)[1].
Jeśli uruchomione są zwykłe pollery, pollery IPMI lub Java, musi działać co najmniej jeden poller dla nieosiągalnych hostów.

Domyślnie: 1
 Zakres: 0-1000

StartPollers

Liczba wstępnie rozwidlonych instancji pollerów[1].

Domyślnie: 5
 Zakres: 0-1000

StartPreprocessors

Liczba wstępnie uruchomionych instancji wątków worker[1] przetwarzania wstępnego powinna być ustawiona na wartość nie mniejszą niż liczba dostępnych rdzeni CPU. Należy ustawić więcej workerów, jeśli przetwarzanie wstępne nie jest ograniczone przez CPU i obejmuje wiele żądań sieciowych.

Domyślnie: 16
 Zakres: 1-1000

StartProxyPollers

Liczba wstępnie utworzonych instancji procesów odpytywania dla pasywnych proxy[1].

Domyślnie: 1
 Zakres: 0-250

StartReportWriters

Liczba wstępnie rozwidlonych instancji modułów zapisujących raporty.
Jeśli ustawiono 0, generowanie raportów według harmonogramu jest wyłączone.
Proces menedżera raportów jest uruchamiany automatycznie po uruchomieniu modułu zapisującego raporty.

Wartość domyślna: 0
 Zakres: 0-100

StartSNMPPollers

Liczba wstępnie rozwidlonych instancji SNMP pollers. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartSNMPTrapper

Jeśli ustawione na 1, zostanie uruchomiony proces SNMP trapper.

Domyślnie: 0
 Zakres: 0-1

StartTimers

Liczba wstępnie rozwidlonych instancji timerów.
Timery przetwarzają okresy utrzymania.

Domyślnie: 1
 Zakres: 1-1000

StartTrappers

Liczba wstępnie rozwidlonych instancji trapperów[1].
Trappery akceptują połączenia przychodzące z Zabbix sender, aktywnych agentów i aktywnych proxy.

Domyślnie: 5
 Zakres: 0-1000

StartVMwareCollectors

Liczba wstępnie rozwidlonych instancji kolektora VMware.

Domyślnie: 0
 Zakres: 0-250

StatsAllowedIP

Lista adresów IP instancji zewnętrznych Zabbix, rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS. Żądanie statystyk będzie akceptowane tylko z adresów wymienionych tutaj. Jeśli ten parametr nie jest ustawiony, żadne żądania statystyk nie będą akceptowane.
Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' zezwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może być użyte do zezwolenia na dowolny adres IPv4.

Przykład:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Timeout

Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z Zabbix proxy, agent, usługą web oraz starszymi kontrolami SNMP (pojedynczy numer OID lub ciąg znaków).

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych:

- zdalne wykonywanie poleceń na Zabbix agent

- wykonywanie poleceń SSH/Telnet
- żądania do usługi web Zabbix
- limit czasu komunikacji dla żądania testowego typu nośnika oraz metody `history.push`
- ponowne planowanie pozycji, gdy interfejs IPMI staje się niedostępny
- wysyłanie odpowiedzi do Zabbix proxy w przypadku niepowodzenia wymiany danych z powodu problemów z uprawnieniami lub szyfrowaniem
- limit czasu dla asynchronicznych gniazd IPC oraz opcji kontroli w czasie działania
- połączenia JMX
- pobieranie statystyk ze zdalnego Zabbix proxy lub serwer
- wysyłanie odpowiedzi do Zabbix frontend
- asynchroniczne żądania DNS pollera
- odpowiedź dla heartbeat aktywnej kontroli
- pobieranie danych Zabbix agent (wartości) od aktywnych agentów
- pobieranie danych z Zabbix sender
- odpowiedź, gdy trapper nie może przetworzyć żądania
- wysyłanie listy aktywnych kontroli do Zabbix agent

Ten limit czasu **nie** będzie używany dla tych kontroli, które mają skonfigurowane ustawienia elastycznego limitu czasu we frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).
Na przykład pozycje SNMP `walk[OID]` i `get[OID]` używają limitu czasu skonfigurowanego we frontendzie; starsze kontrole SNMP nadal używają wartości limitu czasu serwera.

Domyślnie: 3
 Zakres: 1-30

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSCipherAll

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSFrontendAccept

Jakie połączenia przychodzące akceptować z frontend.

Można określić wiele wartości, oddzielonych przecinkami:

- unencrypted - akceptuj połączenia bez szyfrowania.
- cert - akceptuj połączenia zabezpieczone TLS i certyfikatem.

Wartość domyślna: `unencrypted`

TLSFrontendCertIssuer

Dozwolony wystawca certyfikatu frontend.

TLSFrontendCertSubject

Dozwolony temat certyfikatu frontend.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny serwera, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSListen

Steruje TLS na gnieździe trappera.

Obsługiwane wartości:

- required - akceptuj tylko połączenia TLS

TmpDir

Katalog tymczasowy.

Domyślnie: `/tmp`

TrapperTimeout

Określa limit czasu w sekundach dla:

- pobierania danych historycznych z proxy Zabbix
- wysyłania danych konfiguracyjnych do proxy Zabbix
- wykonywania skryptów globalnych lub poleceń zdalnych na serwerze Zabbix

Domyślnie: 300
 Zakres: 1-300

TrendCacheSize

Rozmiar pamięci podręcznej trendów, w bajtach.
Rozmiar pamięci współdzielonej do przechowywania danych trendów.

Domyślnie: `4M`
 Zakres: 128K-16G

TrendFunctionCacheSize

Rozmiar pamięci podręcznej funkcji trendu, w bajtach.
Rozmiar pamięci współdzielonej do buforowania obliczonych danych funkcji trendu.

Domyślnie: `4M`
 Zakres: 128K-2G

UnavailableDelay

Określa, jak często sprawdzana jest dostępność hosta w okresie niedostępności, w sekundach.

Domyślnie: 60
 Zakres: 1-3600

UnreachableDelay

Określa, jak często sprawdzana jest dostępność hosta w okresie nieosiągalności, w sekundach.

Domyślnie: 15
 Zakres: 1-3600

UnreachablePeriod

Określa, po ilu sekundach nieosiągalności host jest traktowany jako niedostępny.

Domyślnie: 45
 Zakres: 1-3600

Użytkownik

Obniż uprawnienia do określonego, istniejącego użytkownika w systemie.
Ma to skutek tylko w przypadku uruchomienia jako 'root' i przy wyłączonym AllowRoot.

Domyślnie: `zabbix`

ValueCacheSize

Rozmiar pamięci podręcznej wartości historii, w bajtach.
Rozmiar pamięci współdzielonej do buforowania żądań danych historii pozycji.
Ustawienie wartości 0 wyłącza pamięć podręczną wartości (niezalecane).
Gdy pamięć podręczna wartości wyczerpie pamięć współdzieloną, co 5 minut do dziennika serwera zapisywany jest komunikat ostrzegawczy.

Domyślnie: `8M`
 Zakres: 0,128K-64G

Vault

Określa dostawcę vault:
HashiCorp - HashiCorp KV Secrets Engine wersja 2
CyberArk - CyberArk Central Credential Provider
Musi odpowiadać dostawcy vault ustawionemu we frontend.

Wartość domyślna: `HashiCorp`

VaultDBPath

Ścieżka lub zapytanie Vault, zależnie od używanego Vault, skąd po kluczach zostaną pobrane poświadczenia do bazy danych.

Kluczami używanymi dla HashiCorp są 'password' i 'username'.

Przykładowa ścieżka z VaultPrefix=/v1/secret/data/zabbix/:

```
database
```

Przykładowa ścieżka bez VaultPrefix:

```
secret/zabbix/database
```

Kluczami używanymi dla CyberArk są 'Content' i 'UserName'.

Przykład:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
```

Ta opcja może być używana tylko wtedy, gdy DBUser i DBPassword nie są określone.

VaultPrefix

Niestandardowy prefiks dla ścieżki lub zapytania Vault, zależnie od Vault. Jeśli nie zostanie określony, zostaną użyte najbardziej odpowiednie wartości domyślne.
Należy pamiętać, że w przypadku HashiCorp po mountpoint automatycznie dodawane jest „data", jeśli VaultPrefix nie jest określony.

Przykładowy prefiks dla Hashicorp:

```
v1/secret/data/zabbix/
```

Przykładowy prefiks dla Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta.
 Plik certyfikatu musi być w formacie PEM1.
 Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole pliku klucza SSL puste.
 Katalog zawierający ten plik jest określony przez parametr konfiguracji SSLCertLocation.
 Tę opcję można pominąć, ale jest zalecana dla sejfu CyberArkCCP.

VaultTLSKeyFile

Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta.
 Plik prywatnego klucza musi być w formacie PEM1.
 Katalog zawierający ten plik jest określany przez parametr konfiguracji SSLKeyLocation.
 Tę opcję można pominąć, ale jest zalecana dla skarbca CyberArkCCP.

VaultToken

Token uwierzytelniający HashiCorp Vault, który powinien zostać wygenerowany wyłącznie dla serwera Zabbix z uprawnieniami tylko do odczytu dla ścieżek określonych w makrach Vault oraz uprawnieniami tylko do odczytu dla ścieżki określonej w opcjonalnym parametrze konfiguracyjnym VaultDBPath.
Zdefiniowanie jednocześnie VaultToken i zmiennej środowiskowej VAULT_TOKEN jest błędem.

Obowiązkowe: Tak, jeśli Vault jest ustawiony na *HashiCorp*; w przeciwnym razie nie

VaultURL

Adres URL HTTP[S] serwera vault. Jeśli SSLCALocation nie jest określone, zostanie użyty katalog certyfikatów CA dla całego systemu.

Domyślnie: `https://127.0.0.1:8200`

VMwareCacheSize

Rozmiar pamięci współdzielonej do przechowywania danych VMware.
Do monitorowania użycia pamięci podręcznej VMware można użyć wewnętrznej kontroli VMware zabbix[vmware,buffer,...] (zobacz Kontrole wewnętrzne).
Należy pamiętać, że pamięć współdzielona nie jest przydzielana, jeśli nie skonfigurowano do uruchomienia żadnych instancji kolektora vmware.

Wartość domyślna: 8M
 Zakres: 256K-2G

VMwareFrequency

Opóźnienie w sekundach między pobieraniem danych z pojedynczej usługi VMware.
To opóźnienie powinno być ustawione na najmniejszy interwał aktualizacji dowolnej pozycji monitorowania VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwarePerfFrequency

Opóźnienie w sekundach pomiędzy pobieraniem statystyk liczników wydajności z pojedynczej usługi VMware. To opóźnienie powinno być ustawione na najmniejszy interwał aktualizacji dowolnej monitorującej VMware pozycji, która używa liczników wydajności VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwareTimeout

Maksymalna liczba sekund, przez którą kolektor vmware będzie oczekiwać na odpowiedź od usługi VMware (vCenter lub hipernadzorcy ESX).

Domyślnie: 10
 Zakres: 1-300

WebServiceURL

Adres URL HTTP[S] do usługi internetowej Zabbix w formacie `http[s]://host:port/report`.

Przykład:

```
WebServiceURL=http://localhost:10053/report
```

Uwaga: schemat (`http://`) można pominąć tylko w przypadku połączeń bez TLS (HTTP); jeśli skonfigurowano TLS, należy użyć `https://`.

WebDriverURL

Adres URL HTTP[S] interfejsu WebDriver.

Przykład (używany z samodzielnym serwerem Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

Przypisy

[1] Należy pamiętać, że zbyt duża liczba procesów zbierania danych (pollerów, pollerów niedostępnych hostów, pollerów ODBC, pollerów HTTP, pollerów Java, pingerów, trapperów, proxypollerów) wraz z menedżerem IPMI, trapperem SNMP, workerami przetwarzania wstępnego oraz workerami wykrywania może wyczerpać limit deskryptorów plików na proces dla menedżera przetwarzania wstępnego.

Warning:
Wyczerpanie limitu deskryptorów plików spowoduje zatrzymanie serwera Zabbix, zwykle krótko po uruchomieniu, choć czasami może to nastąpić później. Aby uniknąć takich problemów, przejrzyj plik konfiguracyjny serwera Zabbix, aby zoptymalizować liczbę równoległych kontroli i procesów. Dodatkowo, jeśli to konieczne, upewnij się, że limit deskryptorów plików jest ustawiony wystarczająco wysoko, sprawdzając i dostosowując limity systemowe.

[2] Gdy usuwanych jest wiele pozycji, zwiększa to obciążenie bazy danych, ponieważ housekeeper będzie musiał usunąć wszystkie dane historyczne, które te pozycje posiadały. Na przykład, jeśli trzeba usunąć tylko 1 prototyp pozycji z szablonu, ale ten szablon jest podłączony do 50 hostów, a dla każdego hosta prototyp jest rozwijany do 100 rzeczywistych pozycji, łącznie trzeba usunąć 5000 pozycji (1*50*100). Jeśli dla MaxHousekeeperDelete ustawiono 500 (MaxHousekeeperDelete=500), proces housekeeper będzie musiał usunąć do 2500000 wartości (5000*500) dla usuniętych pozycji z tabel historii i trendów w jednym cyklu.

[3] DBSocket i DBPort wzajemnie się wykluczają w konfiguracji serwera. Należy określić tylko jeden z nich albo pozostawić oba niezdefiniowane.

2 Zabbix proxy

Przegląd

Parametry obsługiwane przez plik konfiguracyjny Zabbix proxy (zabbix_proxy.conf) są wymienione w tej sekcji.

Parametry są wymienione bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
AllowRoot	Zezwala na uruchamianie proxy jako „root”.
AllowUnsupportedDBVersions	Zezwala proxy na pracę z nieobsługiwanymi wersjami bazy danych.
CacheSize	Rozmiar pamięci podręcznej konfiguracji.
ConfigFrequency	Ten parametr jest **przestarzały** (zamiast niego użyj ProxyConfigFrequency). Określa, jak często proxy pobiera dane konfiguracyjne z serwera Zabbix, w sekundach.
DataSenderFrequency	Proxy będzie wysyłać zebrane dane do serwera co N sekund.
DBHost	host (lub katalog gniazda) bazy danych Zabbix.
DBName	Nazwa bazy danych lub ścieżka do pliku bazy danych dla SQLite3.
DBPassword	Hasło bazy danych.
DBPort	Port hosta bazy danych Zabbix.
DBSchema	Nazwa schematu bazy danych. Używane dla PostgreSQL.
DBSocket	Ścieżka do pliku gniazda MySQL.
DBUser	Użytkownik bazy danych.
DBTLSConnect	Ustawienie tej opcji na określoną wartość wymusza użycie połączenia TLS z bazą danych.
DBTLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) do weryfikacji certyfikatu bazy danych.
DBTLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat Zabbix proxy do uwierzytelniania w bazie danych.
DBTLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny do uwierzytelniania w bazie danych.
DBTLSCipher	Lista szyfrów, na które Zabbix proxy zezwala dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.
DBTLSCipher13	Lista pakietów szyfrów, na które Zabbix proxy zezwala dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, począwszy od wersji 8.0.16.
DebugLevel	Poziom debugowania.
EnableRemoteCommands	Czy zdalne polecenia z serwera Zabbix są dozwolone.
ExternalScripts	Lokalizacja zewnętrznych skryptów.
Fping6Location	Lokalizacja fping6.
FpingLocation	Lokalizacja fping.
HistoryCacheSize	Rozmiar pamięci podręcznej historii.
HistoryIndexCacheSize	Rozmiar pamięci podręcznej indeksu historii.
Hostname	Unikalna nazwa proxy z rozróżnianiem wielkości liter.

Parameter	Description
HostnameItem	pozycja używana do ustawienia Hostname, jeśli nie jest zdefiniowana.
HousekeepingFrequency	Określa, jak często Zabbix będzie wykonywać procedurę housekeeping, w godzinach.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki z katalogu do pliku konfiguracyjnego.
JavaGateway	Adres IP (lub nazwa hosta) Zabbix Java gateway.
JavaGatewayPort	Port, na którym nasłuchuje Zabbix Java gateway.
ListenBacklog	Maksymalna liczba oczekujących połączeń w kolejce TCP.
ListenIP	Lista adresów IP oddzielonych przecinkami, na których trapper powinien nasłuchiwać.
ListenPort	Port nasłuchiwania dla trappera.
LoadModule	Moduł do załadowania przy uruchamianiu proxy.
LoadModulePath	Pełna ścieżka do lokalizacji modułów proxy.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogSlowQueries	Jak długo może trwać zapytanie do bazy danych, zanim zostanie zarejestrowane.
LogType	Typ wyjścia dziennika.
MaxConcurrentChecksPerPoller	Maksymalna liczba asynchronicznych kontroli, które mogą być wykonywane jednocześnie przez każdy poller HTTP agent, poller agent lub poller SNMP.
PidFile	Nazwa pliku PID.
ProxyBufferMode	Określa mechanizm przechowywania danych historii, wykrywania i autorejestracji (dysk/pamięć/hybrydowy).
ProxyConfigFrequency	Określa, jak często proxy pobiera dane konfiguracyjne z serwera Zabbix, w sekundach.
ProxyLocalBuffer	Proxy będzie przechowywać dane lokalnie przez N godzin, nawet jeśli dane zostały już zsynchronizowane z serwerem.
ProxyMemoryBufferAge	Maksymalny wiek danych w buforze pamięci proxy, w sekundach.
ProxyMemoryBufferSize	Rozmiar współdzielonej pamięci podręcznej dla zebranych danych historii, wykrywania i autorejestracji.
ProxyMode	Tryb działania proxy (aktywny/pasywny).
ProxyOfflineBuffer	Proxy będzie przechowywać dane przez N godzin w przypadku braku łączności z serwerem Zabbix.
Server	Jeśli ProxyMode jest ustawiony na tryb aktywny: adres IP lub nazwa DNS serwera Zabbix (adres:port) albo klaster (adres:port;adres2:port), z którego pobierane są dane konfiguracyjne i do którego wysyłane są dane. Jeśli ProxyMode jest ustawiony na tryb pasywny: lista adresów IP serwera Zabbix oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS.
SNMPTrapperFile	Plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do proxy.
SocketDir	Katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.
SourceIP	Źródłowy adres IP.
SSHKeyLocation	Lokalizacja kluczy publicznych i prywatnych dla kontroli SSH i akcji.
SSLCertLocation	Lokalizacja plików certyfikatów klienta SSL do uwierzytelniania klienta.
SSLKeyLocation	Lokalizacja plików kluczy prywatnych SSL do uwierzytelniania klienta.
SSLCALocation	Nadpisuje lokalizację plików urzędu certyfikacji (CA) do weryfikacji certyfikatu serwera SSL.
StartAgentPollers	Liczba wstępnie rozwidlonych instancji asynchronicznych pollerów Zabbix agent.
StartBrowserPollers	Liczba wstępnie rozwidlonych instancji pollerów pozycji przeglądarkowych.
StartDBSyncers	Liczba wstępnie rozwidlonych instancji synchronizatorów historii.
StartDiscoverers	Liczba wstępnie rozwidlonych instancji workerów wykrywania.
StartHTTPAgentPollers	Liczba wstępnie rozwidlonych instancji asynchronicznych pollerów HTTP agent.
StartHTTPPollers	Liczba wstępnie rozwidlonych instancji pollerów HTTP.
StartIPMIPollers	Liczba wstępnie rozwidlonych instancji pollerów IPMI.
StartJavaPollers	Liczba wstępnie rozwidlonych instancji pollerów Java.
StartODBCPollers	Liczba wstępnie rozwidlonych instancji pollerów ODBC.
StartPingers	Liczba wstępnie rozwidlonych instancji pingerów ICMP.
StartPollersUnreachable	Liczba wstępnie rozwidlonych instancji pollerów dla nieosiągalnych hostów (w tym IPMI i Java).
StartPollers	Liczba wstępnie rozwidlonych instancji pollerów.
StartPreprocessors	Liczba wstępnie uruchomionych instancji workerów przetwarzania wstępnego.
StartSNMPPollers	Liczba wstępnie rozwidlonych instancji asynchronicznych pollerów SNMP.
StartSNMPTrapper	Jeśli ustawione na 1, zostanie uruchomiony proces trappera SNMP.
StartTrappers	Liczba wstępnie rozwidlonych instancji trapperów.
StartVMwareCollectors	Liczba wstępnie rozwidlonych instancji kolektorów VMware.
StatsAllowedIP	Lista adresów IP zewnętrznych instancji Zabbix oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS. Żądanie statystyk będzie akceptowane tylko z adresów wymienionych tutaj.

Parameter	Description
Timeout	Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z Zabbix proxy, agent, usługą web oraz dla kontroli SNMP (z wyjątkiem pozycji SNMP `walk[OID]` i `get[OID]`).
TLSAccept	Jakie połączenia przychodzące od serwera Zabbix mają być akceptowane.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używanego do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Nadpisuje domyślne kryteria wyboru pakietów szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Jak proxy powinno łączyć się z serwerem Zabbix.
TLSCRLFile	Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny proxy, używanego do szyfrowanej komunikacji między komponentami Zabbix.
TLSListen	Steruje TLS na gnieździe trappera.
TLSPSKFile	Pełna ścieżka do pliku zawierającego klucz współdzielony proxy, używanego do szyfrowanej komunikacji z serwerem Zabbix.
TLSPSKIdentity	Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.
TLSServerCertIssuer	Dozwolony wystawca certyfikatu serwera.
TLSServerCertSubject	Dozwolony podmiot certyfikatu serwera.
TmpDir	Katalog tymczasowy.
TrapperTimeout	Określa limit czasu w sekundach dla: - pobierania danych konfiguracyjnych z serwera Zabbix; - wykonywania skryptu globalnego lub zdalnego polecenia.
UnavailableDelay	Jak często sprawdzana jest dostępność hosta w okresie niedostępności.
UnreachableDelay	Jak często sprawdzana jest dostępność hosta w okresie nieosiągalności.
UnreachablePeriod	Po ilu sekundach nieosiągalności host ma być traktowany jako niedostępny.
User	Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
Vault	Dostawca vault.
VaultDBPath	Lokalizacja, z której poświadczenia bazy danych powinny być pobierane za pomocą kluczy.
VaultPrefix	Niestandardowy prefiks dla ścieżki lub zapytania vault.
VaultTLSCertFile	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta.
VaultTLSKeyFile	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta.
VaultToken	Token uwierzytelniający HashiCorp vault.
VaultURL	Adres URL HTTP[S] serwera vault.
VMwareCacheSize	Rozmiar pamięci współdzielonej do przechowywania danych VMware.
VMwareFrequency	Opóźnienie w sekundach między zbieraniem danych z pojedynczej usługi VMware.
VMwarePerfFrequency	Opóźnienie w sekundach między pobieraniem statystyk liczników wydajności z pojedynczej usługi VMware.
VMwareTimeout	Maksymalna liczba sekund, przez którą kolektor vmware będzie czekać na odpowiedź z usługi VMware.
WebDriverURL	Adres URL HTTP[S] interfejsu WebDriver.

Wszystkie parametry są opcjonalne, chyba że wyraźnie wskazano, że dany parametr jest obowiązkowy.

Zwróć uwagę, że:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości w dostarczanych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.

- Zabbix obsługuje pliki konfiguracyjne tylko w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od "#" są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

AllowRoot

Zezwala na uruchamianie proxy jako „root". Jeśli wyłączone, a proxy zostanie uruchomiony przez „root", proxy spróbuje zamiast tego przełączyć się na użytkownika „zabbix". Nie ma wpływu, jeśli został uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowUnsupportedDBVersions

Zezwala proxy na pracę z nieobsługiwanymi wersjami bazy danych.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

CacheSize

Rozmiar pamięci podręcznej konfiguracji, w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych hostów i pozycji.

Domyślnie: 32M
 Zakres: 128K-64G

ConfigFrequency

Ten parametr jest **przestarzały** (zamiast niego użyj ProxyConfigFrequency).
Określa, jak często proxy pobiera dane konfiguracyjne z serwera Zabbix, w sekundach.
Parametr aktywnego proxy. Ignorowany dla pasywnych proxy (zobacz parametr ProxyMode).

Wartość domyślna: 3600
 Zakres: 1-604800

DataSenderFrequency

proxy będzie wysyłał zebrane dane do serwer co N sekund. Należy pamiętać, że aktywny proxy będzie nadal odpytywać serwer Zabbix co sekundę w celu sprawdzenia zadań zdalnych poleceń.
Parametr aktywnego proxy. Ignorowany dla pasywnych proxy (zobacz parametr ProxyMode).

Wartość domyślna: 1
 Zakres: 1-3600

DBHost

Host (lub katalog gniazda) bazy danych Zabbix.

Dla MySQL:

- `localhost` lub pusty ciąg używa domyślnego gniazda domeny UNIX.

Dla PostgreSQL:

- `localhost` jest rozwiązywany przez DNS (zwykle do 127.0.0.1).
- Pusty ciąg używa domyślnego gniazda domeny UNIX.
- Ścieżka (na przykład `/var/run/pgbouncer`) powoduje użycie gniazda UNIX w tej ścieżce.
- Lista rozdzielona przecinkami może zawierać wiele wartości `host:port`. Każdy host jest wypróbowywany po kolei, aż zostanie ustanowione połączenie do odczytu i zapisu. Przykład: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[:`

Domyślnie: `localhost`

DBName

Nazwa bazy danych lub ścieżka do pliku bazy danych dla SQLite3 (architektura wieloprocesowa Zabbix nie pozwala na użycie bazy danych w pamięci, np. `:memory:`, `file::memory:?cache=shared` lub `file:memdb1?mode=memory&cache=shared`).
Ostrzeżenie: Nie próbuj używać tej samej bazy danych, której używa serwer Zabbix.

Wymagane: Tak

DBPassword

Hasło bazy danych. Zakomentuj tę linię, jeśli hasło nie jest używane. Ignorowane dla SQLite.

DBPort

Port hosta bazy danych Zabbix (zobacz DBHost).[1] Ignorowane dla SQLite.

Domyślnie dla MySQL: 3306

Domyślnie dla PostgreSQL: 5432

Zakres: 1024-65535

DBSchema

Nazwa schematu bazy danych. Używane dla PostgreSQL.

DBSocket

Ścieżka do pliku gniazda MySQL.[1]

DBUser

Użytkownik bazy danych. Ignorowane dla SQLite.

DBTLSConnect

Ustawienie tej opcji wymusza użycie połączenia TLS z bazą danych:
required - połącz przy użyciu TLS
verify_ca - połącz przy użyciu TLS i zweryfikuj certyfikat
verify_full - połącz przy użyciu TLS, zweryfikuj certyfikat oraz sprawdź, czy tożsamość bazy danych określona przez DBHost odpowiada jej certyfikatowi
W MySQL od wersji 5.7.11 oraz w PostgreSQL obsługiwane są następujące wartości: ”required”, ”verify”, ”verify_full”.
W MariaDB od wersji 10.2.6 obsługiwane są wartości ”required” i ”verify_full”.
Domyślnie nie jest ustawiona żadna opcja, a zachowanie zależy od konfiguracji bazy danych.

DBTLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnego urzędu certyfikacji (CA) używane do weryfikacji certyfikatu bazy danych.

Obowiązkowe: nie (tak, jeśli DBTLSConnect ustawiono na *verify_ca* lub *verify_full*)

DBTLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat proxy Zabbix używany do uwierzytelniania w bazie danych.

DBTLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.

DBTLSCipher

Lista szyfrów szyfrowania, na które proxy Zabbix zezwala dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.

DBTLSCipher13

Lista szyfrów pakietów szyfrujących, na które proxy Zabbix zezwala dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, począwszy od wersji 8.0.16.

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

EnableRemoteCommands

Czy zdalne polecenia z serwer Zabbix są dozwolone.

Domyślnie: 0
 Wartości: 0 - niedozwolone; 1 - dozwolone

ExternalScripts

Lokalizacja skryptów zewnętrznych (zależy od zmiennej instalacyjnej `datadir` ustawianej podczas kompilacji).

Wartość domyślna: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Lokalizacja fping6. Upewnij się, że plik binarny fping6 ma właściciela root oraz ustawioną flagę SUID. Pozostaw puste (`Fping6Location=`), jeśli narzędzie fping potrafi przetwarzać adresy IPv6.

Domyślnie: `/usr/sbin/fping6`

FpingLocation

Lokalizacja fping. Upewnij się, że plik binarny fping ma właściciela root oraz ustawioną flagę SUID.

Domyślnie: `/usr/sbin/fping`

HistoryCacheSize

Rozmiar pamięci podręcznej historii w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych historii.

Domyślnie: 16M
 Zakres: 128K-16G

HistoryIndexCacheSize

Rozmiar pamięci podręcznej indeksu historii, w bajtach. Rozmiar pamięci współdzielonej do indeksowania danych historii przechowywanych w pamięci podręcznej historii. Rozmiar pamięci podręcznej indeksu wymaga około 100 bajtów do buforowania jednej pozycji.

Domyślnie: 4M
 Zakres: 128K-16G

Hostname

Unikalna nazwa proxy z rozróżnianiem wielkości liter. Upewnij się, że nazwa proxy jest znana serwerowi.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128

Wartość domyślna: ustawiana przez HostnameItem

HostnameItem

Pozycja używana do ustawienia Hostname, jeśli nie jest on zdefiniowany (zostanie uruchomiona na proxy podobnie jak na agencie). Ignorowane, jeśli Hostname jest ustawiony.
Nie obsługuje UserParameters, liczników wydajności ani aliasów, ale obsługuje system.run[].

Wartość domyślna: system.hostname

HousekeepingFrequency

Jak często Zabbix będzie wykonywać procedurę housekeeping (w godzinach). Housekeeping polega na usuwaniu nieaktualnych informacji z bazy danych.
Uwaga: Aby zmniejszyć obciążenie podczas uruchamiania proxy, housekeeping jest opóźniany o 30 minut po starcie proxy. W związku z tym, jeśli HousekeepingFrequency ma wartość 1, pierwsza procedura housekeeping po uruchomieniu proxy zostanie wykonana po 30 minutach, a następnie będzie uruchamiana co godzinę.
Możliwe jest wyłączenie automatycznego housekeeping przez ustawienie HousekeepingFrequency na 0. W takim przypadku procedura housekeeping może zostać uruchomiona wyłącznie za pomocą opcji sterowania w czasie działania *housekeeper_execute*.
Zobacz także opcje runtime control.

Domyślnie: 1
 Zakres: 0-24

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym.
Aby dołączać tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Adres IP (lub nazwa hosta) bramy Java gateway Zabbix. Wymagane tylko wtedy, gdy uruchomione są odpytywacze Java.

JavaGatewayPort

Port, na którym nasłuchuje Zabbix Java gateway.

Domyślnie: 10052
 Zakres: 1024-32767

ListenBacklog

Maksymalna liczba oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na stałe i zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać po cichu obcięte do „maksimum określonego przez implementację".

Domyślnie: SOMAXCONN
 Zakres: 0 - INT_MAX

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których trapper powinien nasłuchiwać.
Jeśli ten parametr nie zostanie określony, trapper będzie nasłuchiwał na wszystkich interfejsach sieciowych.

Wartość domyślna: 0.0.0.0

ListenPort

Port nasłuchiwania dla trappera.

Domyślnie: 10051
 Zakres: 1024-32767

LoadModule

Moduł do załadowania podczas uruchamiania proxy. Moduły są używane do rozszerzania funkcjonalności proxy. Moduł musi znajdować się w katalogu określonym przez LoadModulePath lub ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od '/'), wówczas LoadModulePath jest ignorowane.
Formaty:
LoadModule=<module.so>
LoadMod jest uwzględnienie wielu parametrów LoadModule.

LoadModulePath

Pełna ścieżka do lokalizacji modułów proxy. Wartość domyślna zależy od opcji kompilacji.

LogFile

Nazwa pliku dziennika.

Wymagane: Tak, jeśli parametr LogType jest ustawiony na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację plików dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty od nowa.

Wartość domyślna: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogType

Typ wyjścia logu:
file - zapisuj log do pliku określonego przez parametr LogFile;
system - zapisuj log do syslog;
console - zapisuj log na standardowe wyjście.

Wartość domyślna: `file`

LogSlowQueries

Jak długo zapytanie do bazy danych może trwać, zanim zostanie zapisane w logu (w milisekundach).
0 - nie zapisuj wolnych zapytań w logu.
Ta opcja jest włączana od DebugLevel=3.

Wartość domyślna: 0
 Zakres: 0-3600000

MaxConcurrentChecksPerPoller

Maksymalna liczba asynchronicznych kontroli, które mogą być wykonywane jednocześnie przez każdy moduł odpytywania HTTP agent, agent lub SNMP. Zobacz StartHTTPAgentPollers, StartAgentPollers oraz StartSNMPPollers.

Wartość domyślna: 1000
 Zakres: 1-1000

PidFile

Nazwa pliku PID.

Domyślnie: `/tmp/zabbix_proxy.pid`

ProxyBufferMode

Określa mechanizm przechowywania danych historii, wykrywania sieci i autorejestracji: *disk* — dane są przechowywane w bazie danych i wysyłane z bazy danych; *memory* — dane są przechowywane w pamięci i wysyłane z pamięci. Jeśli buforowi zabraknie pamięci, stare dane zostaną odrzucone. Przy zamykaniu bufor jest odrzucany. *hybrid* — bufor proxy zwykle działa jak w trybie pamięci, dopóki nie zabraknie mu pamięci lub najstarszy rekord nie przekroczy skonfigurowanego wieku. Jeśli tak się stanie, bufor jest opróżniany do bazy danych i działa jak w trybie dyskowym, dopóki wszystkie dane nie zostaną wysłane, po czym ponownie zaczyna działać w pamięci. Przy zamykaniu bufor pamięci jest opróżniany do bazy danych.

Zobacz także: Bufor pamięci proxy.

Wartość domyślna: `disk`
 Wartości: disk; memory; hybrid

ProxyConfigFrequency

Jak często proxy pobiera dane konfiguracyjne z serwera Zabbix, w sekundach.
Parametr aktywnego proxy. Ignorowany dla proxy pasywnych (zobacz parametr ProxyMode).

Domyślnie: 10
 Zakres: 1-604800

ProxyLocalBuffer

Proxy będzie przechowywać dane lokalnie przez N godzin, nawet jeśli dane zostały już zsynchronizowane z serwerem.
Tego parametru można użyć, jeśli lokalne dane będą wykorzystywane przez aplikacje innych firm.

Wartość domyślna: 0
 Zakres: 0-720

ProxyMemoryBufferAge

Maksymalny wiek danych w buforze pamięci proxy, w sekundach. Gdy jest włączony (wartość różna od zera) i rekordy w buforze pamięci proxy są starsze, wymusza to przełączenie bufora proxy w tryb bazy danych do momentu przesłania wszystkich rekordów do serwer. Ten parametr musi być mniejszy lub równy parametrowi ProxyOfflineBuffer.

Wartość domyślna: 0
 Zakres: 0;600-864000

ProxyMemoryBufferSize

Rozmiar pamięci współdzielonej cache dla zebranych danych historii, wykrywania i autorejestracji, w bajtach. Jeśli włączone (wartość różna od zera), proxy będzie przechowywać dane historii, wykrywania i autorejestracji w pamięci, chyba że cache będzie pełny lub zapisane rekordy będą starsze niż zdefiniowany parametr ProxyMemoryBufferAge. Ten parametr nie może być używany razem z parametrem ProxyLocalBuffer.

Domyślnie: 0
 Zakres: 0;128K-2G

ProxyMode

Tryb pracy proxy.
0 - proxy w trybie aktywnym
1 - proxy w trybie pasywnym
Uwaga: poufne dane konfiguracyjne proxy mogą stać się dostępne dla stron mających dostęp do portu trappera serwera Zabbix podczas korzystania z aktywnego proxy. Jest to możliwe, ponieważ każdy może podszyć się pod aktywny proxy i zażądać danych konfiguracyjnych; uwierzytelnianie nie jest przeprowadzane.

Wartość domyślna: 0
 Zakres: 0-1

ProxyOfflineBuffer

Proxy będzie przechowywać dane przez N godzin w przypadku braku łączności z serwerem Zabbix.
Starsze dane zostaną utracone.

Wartość domyślna: 1
 Zakres: 1-720

Serwer

Jeśli ProxyMode jest ustawiony na *tryb aktywny*:
Adres IP lub nazwa DNS serwera Zabbix (adres:port) albo klaster (adres:port;adres2:port), z którego pobierane są dane konfiguracyjne i do którego wysyłane są dane.
Jeśli port nie jest określony, używany jest port domyślny.
Węzły klastra muszą być oddzielone średnikiem.

Jeśli ProxyMode jest ustawiony na *tryb pasywny*:
Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwera Zabbix. Połączenia przychodzące będą akceptowane tylko z adresów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo.
'::/0' zezwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może być użyte do zezwolenia na dowolny adres IPv4.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak

SNMPTrapperFile

Plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do proxy.
Musi być taki sam jak w pliku konfiguracyjnym zabbix_trap_receiver.pl lub SNMPTT.

Wartość domyślna: `/tmp/zabbix_traps.tmp`

SocketDir

Katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.

Domyślnie: `/tmp`

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix
- połączeń bez agenta (VMware, SSH, JMX, SNMP, Telnet oraz prostych kontroli)
- połączeń HTTP agent
- żądań HTTP JavaScript dla pozycji skryptowych

- żądań HTTP JavaScript dla preprocessingu
- połączeń z Vault

SSHKeyLocation

Lokalizacja kluczy publicznych i prywatnych dla kontroli SSH oraz akcji.

SSLCertLocation

Lokalizacja plików certyfikatów klienta SSL używanych do uwierzytelniania klienta.
Ten parametr jest używany tylko w monitorowaniu WWW.

SSLKeyLocation

Lokalizacja plików prywatnych kluczy SSL do uwierzytelniania klienta.
Ten parametr jest używany tylko w monitorowaniu WWW.

SSLCALocation

Lokalizacja plików urzędu certyfikacji (CA) do weryfikacji certyfikatu SSL serwera.
Należy pamiętać, że wartość tego parametru zostanie ustawiona jako opcja libcurl CURLOPT_CAPATH. W przypadku wersji libcurl wcześniejszych niż 7.42.0 ma to skutek tylko wtedy, gdy libcurl został skompilowany do użycia OpenSSL. Więcej informacji można znaleźć na stronie cURL.
Ten parametr jest używany w monitorowaniu WWW oraz w uwierzytelnianiu SMTP.

StartAgentPollers

Liczba wstępnie rozwidlonych instancji pollerów Zabbix agent. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartBrowserPollers

Liczba wstępnie utworzonych instancji pollerów pozycji przeglądarki.

Domyślnie: 1
 Zakres: 0-1000

StartDBSyncers

Liczba wstępnie rozwidlonych instancji synchronizatorów historii.
Uwaga: Zachowaj ostrożność podczas zmiany tej wartości, jej zwiększenie może przynieść więcej szkody niż pożytku.

Domyślnie: 4
 Zakres: 1-100

StartDiscoverers

Liczba wstępnie rozwidlonych instancji workerów wykrywania.

Domyślnie: 5
 Zakres: 0-1000

StartHTTPAgentPollers

Liczba wstępnie rozwidlonych instancji HTTP agent pollerów. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartHTTPPollers

Liczba wstępnie rozwidlonych instancji modułów odpytujących HTTP.

Domyślnie: 1
 Zakres: 0-1000

StartIPMIPollers

Liczba wstępnie rozwidlonych instancji modułów odpytywania IPMI.

Domyślnie: 0
 Zakres: 0-1000

StartJavaPollers

Liczba wstępnie rozwidlonych instancji sond Java.

Domyślnie: 0
 Zakres: 0-1000

StartODBCPollers

Liczba wstępnie rozwidlonych instancji ODBC pollers.

Domyślnie: 1
 Zakres: 0-1000

StartPingers

Liczba wstępnie rozwidlonych instancji ICMP pingers.

Domyślnie: 1
 Zakres: 0-1000

StartPollersUnreachable

Liczba wstępnie rozwidlonych instancji pollerów dla nieosiągalnych hostów (w tym IPMI i Java). Jeśli uruchomione są zwykłe pollery, pollery IPMI lub Java, musi działać co najmniej jeden poller dla nieosiągalnych hostów.

Domyślnie: 1
 Zakres: 0-1000

StartPollers

Liczba wstępnie rozwidlonych instancji pollerów.

Domyślnie: 5
 Zakres: 0-1000

StartPreprocessors

Liczba wstępnie uruchomionych instancji wątków worker przetwarzania wstępnego powinna być ustawiona na wartość nie mniejszą niż liczba dostępnych rdzeni CPU. Należy ustawić więcej workerów, jeśli przetwarzanie wstępne nie jest ograniczone wydajnością CPU i obejmuje wiele żądań sieciowych.

Domyślnie: 16
 Zakres: 1-1000

StartSNMPPollers

Liczba wstępnie rozwidlonych instancji SNMP pollerów. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartSNMPTrapper

Jeśli ustawione na 1, zostanie uruchomiony proces SNMP trapper.

Domyślnie: 0
 Zakres: 0-1

StartTrappers

Liczba wstępnie rozwidlonych instancji trapperów.
Trappery akceptują połączenia przychodzące od Zabbix sender oraz aktywnych agentów.

Domyślnie: 5
 Zakres: 0-1000

StartVMwareCollectors

Liczba wstępnie rozwidlonych instancji kolektora VMware.

Domyślnie: 0
 Zakres: 0-250

StatsAllowedIP

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS zewnętrznych instancji Zabbix. Żądanie statystyk będzie akceptowane wyłącznie z adresów wymienionych tutaj. Jeśli ten parametr nie jest ustawiony, żadne żądania statystyk nie będą akceptowane.
Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' zezwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może zostać użyte do zezwolenia na dowolny adres IPv4.

Przykład:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Timeout

Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z serwerem Zabbix, agent, usługą web oraz starszymi kontrolami SNMP (pojedynczy numer OID lub ciąg znaków).

Ten parametr określa czas trwania różnych operacji komunikacyjnych:

- Zdalne wykonywanie poleceń na Zabbix agent
- Wykonywanie poleceń SSH / Telnet
- Ponowne planowanie pozycji, gdy interfejs IPMI staje się niedostępny
- Wysyłanie odpowiedzi do serwera Zabbix w przypadku niepowodzenia wymiany danych z powodu problemów z uprawnieniami lub szyfrowaniem
- Limit czasu dla asynchronicznych gniazd IPC i opcji kontroli w czasie działania
- Żądania DNS asynchronicznego pollera
- Odpowiedź dla heartbeat aktywnej kontroli
- Pobieranie danych Zabbix agent (wartości) od aktywnych agentów
- Pobieranie danych z Zabbix sender

- Wysyłanie listy aktywnych kontroli do Zabbix agent

Ten limit czasu **nie** będzie używany dla tych kontroli, które mają skonfigurowane ustawienia elastycznego limitu czasu we frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).
Na przykład pozycje SNMP `walk[OID]` i `get[OID]` używają limitu czasu skonfigurowanego we frontendzie; starsze kontrole SNMP nadal używają wartości limitu czasu serwera.

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Jakie połączenia przychodzące od serwera Zabbix mają być akceptowane. Używane dla proxy pasywnego, ignorowane dla proxy aktywnego. Można określić wiele wartości, oddzielonych przecinkami:
unencrypted — akceptuj połączenia bez szyfrowania (domyślnie)
psk — akceptuj połączenia z TLS i kluczem współdzielonym (PSK)
cert — akceptuj połączenia z TLS i certyfikatem

Wymagane: tak dla proxy pasywnego, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) używane do weryfikacji certyfikatu peera, wykorzystywane w szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat proxy lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSCipherAll

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Sposób, w jaki proxy powinien łączyć się z serwerem Zabbix. Używane dla aktywnego proxy, ignorowane dla pasywnego proxy. Można określić tylko jedną wartość:
unencrypted - połączenie bez szyfrowania (domyślnie)
psk - połączenie z użyciem TLS i klucza współdzielonego (PSK)
cert - połączenie z użyciem TLS i certyfikatu

Wymagane: tak dla aktywnego proxy, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny proxy, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSListen

Steruje TLS na gnieździe trappera.

Obsługiwane wartości:

- required - akceptuj tylko połączenia TLS

TLSPSKFile

Pełna ścieżka do pliku zawierającego klucz współdzielony proxy, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSPSKIdentity

Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Dozwolony wystawca certyfikatu serwera.

TLSServerCertSubject

Dozwolony temat certyfikatu serwera.

TmpDir

Katalog tymczasowy.

Domyślnie: `/tmp`

TrapperTimeout

Określa limit czasu w sekundach dla:
 - pobierania danych konfiguracyjnych z serwera Zabbix;
 - wykonywania skryptów globalnych lub poleceń zdalnych.

Wartość domyślna: 300
 Zakres: 1-300

UnavailableDelay

Jak często host jest sprawdzany pod kątem dostępności w okresie niedostępności, w sekundach.

Wartość domyślna: 60
 Zakres: 1-3600

UnreachableDelay

Jak często sprawdzana jest dostępność hosta w okresie niedostępności, w sekundach.

Domyślnie: 15
 Zakres: 1-3600

UnreachablePeriod

Po ilu sekundach nieosiągalności traktować host jako niedostępny.

Domyślnie: 45
 Zakres: 1-3600

Użytkownik

Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
Ma zastosowanie tylko w przypadku uruchomienia jako `root` i gdy opcja AllowRoot jest wyłączona.

Domyślnie: `zabbix`

Dostawca vault

Dostawca vault:
HashiCorp - HashiCorp KV Secrets Engine w wersji 2
CyberArk - CyberArk Central Credential Provider
Musi odpowiadać dostawcy vault ustawionemu we frontend.

Wartość domyślna: `HashiCorp`

VaultDBPath

Ścieżka lub zapytanie Vault, zależnie od używanego Vault, skąd będą pobierane poświadczenia do bazy danych za pomocą kluczy.

Klucze używane dla HashiCorp to 'password' i 'username'.

Przykładowa ścieżka z VaultPrefix=/v1/secret/data/zabbix/:

```
database
```

Przykładowa ścieżka bez VaultPrefix:

```
secret/zabbix/database
```

Klucze używane dla CyberArk to 'Content' i 'UserName'.

Przykład:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_proxy_database
```

Ta opcja może być używana tylko wtedy, gdy DBUser i DBPassword nie są określone.

VaultPrefix

Niestandardowy prefiks dla ścieżki lub zapytania Vault, w zależności od Vault. Jeśli nie zostanie określony, zostaną użyte najbardziej odpowiednie wartości domyślne.
Zwróć uwagę, że dla HashiCorp po mountpoint automatycznie dodawane jest `data`, jeśli VaultPrefix nie jest określony.
Zwróć uwagę, że dla HashiCorp po mountpoint automatycznie dodawane jest `data`, jeśli VaultPrefix nie jest określony.

Przykładowy prefiks dla Hashicorp:

```
v1/secret/data/zabbix/
```

Przykładowy prefiks dla Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM1.
Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole pliku klucza SSL puste.
Katalog zawierający ten plik jest określany przez parametr konfiguracyjny SSLCertLocation.
Tę opcję można pominąć, ale jest zalecana dla skarbca CyberArkCCP.

VaultTLSKeyFile

Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik prywatnego klucza musi być w formacie PEM1.
Katalog zawierający ten plik jest określany przez parametr konfiguracyjny SSLKeyLocation.
Tę opcję można pominąć, ale jest zalecana w przypadku sejfu CyberArkCCP vault.

VaultToken

Token uwierzytelniania HashiCorp vault, który powinien zostać wygenerowany wyłącznie dla proxy Zabbix z uprawnieniami tylko do odczytu dla ścieżki określonej w opcjonalnym parametrze konfiguracyjnym VaultDBPath.
Błędem jest jednoczesne zdefiniowanie VaultToken i zmiennej środowiskowej VAULT_TOKEN.

Obowiązkowe: Tak, jeśli Vault jest ustawiony na *HashiCorp*; w przeciwnym razie nie

VaultURL

Adres URL HTTP[S] serwera vault. Jeśli SSLCALocation nie jest określone, zostanie użyty katalog certyfikatów CA dla całego systemu.

Domyślnie: `https://127.0.0.1:8200`

VMwareCacheSize

Rozmiar pamięci współdzielonej do przechowywania danych VMware.
Wewnętrzna kontrola VMware `zabbix[vmware,buffer,...]` może być używana do monitorowania wykorzystania pamięci podręcznej VMware (zobacz Kontrole wewnętrzne).
Zwróć uwagę, że pamięć współdzielona nie jest przydzielana, jeśli nie skonfigurowano do uruchomienia żadnych instancji kolektora VMware.

Wartość domyślna: 8M
 Zakres: 256K-2G

VMwareFrequency

Opóźnienie w sekundach między pobieraniem danych z pojedynczej usługi VMware.
To opóźnienie powinno być ustawione na najmniejszy interwał aktualizacji dowolnej pozycji monitorowania VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwarePerfFrequency

Opóźnienie w sekundach między pobieraniem statystyk liczników wydajności z pojedynczej usługi VMware.
To opóźnienie powinno być ustawione na najmniejszy interwał aktualizacji dowolnej monitorującej VMware pozycji, która używa liczników wydajności VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwareTimeout

Maksymalna liczba sekund, przez którą kolektor vmware będzie oczekiwał na odpowiedź od usługi VMware (vCenter lub hiperwizora ESX).

Domyślnie: 10
 Zakres: 1-300

WebDriverURL

Adres URL HTTP[S] interfejsu WebDriver.

Przykład (używany z samodzielnym serwerem Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

Przypisy

[1] DBSocket i DBPort wzajemnie się wykluczają w konfiguracji proxy. Należy określić tylko jeden z nich lub pozostawić oba niezdefiniowane.

3 Zabbix agent (UNIX)

Przegląd

Parametry obsługiwane przez plik konfiguracyjny Zabbix agent (zabbix_agentd.conf) są wymienione w tej sekcji.

Parametry są podane bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
Alias	Ustawia alias dla klucza pozycji.
AllowKey	Zezwala na wykonywanie tych kluczy pozycji, które pasują do wzorca.
AllowRoot	Zezwala na uruchamianie agent jako 'root'.
BufferSend	Nie przechowuje danych w buforze dłużej niż N sekund.
BufferSize	Maksymalna liczba wartości w buforze pamięci.
DebugLevel	Poziom debugowania.
DenyKey	Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca.
EnableRemoteCommands	Określa, czy zdalne polecenia z Zabbix serwer są dozwolone.
HeartbeatFrequency	Częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Opcjonalny parametr definiujący interfejs hosta.
HostInterfaceItem	Opcjonalny parametr definiujący pozycję używaną do pobierania interfejsu hosta.
HostMetadata	Opcjonalny parametr definiujący metadane hosta.
HostMetadataItem	Opcjonalny parametr definiujący pozycję Zabbix agent używaną do pobierania metadanych hosta.
Hostname	Opcjonalny parametr definiujący nazwę hosta.
HostnameItem	Opcjonalny parametr definiujący pozycję Zabbix agent używaną do pobierania nazwy hosta.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki z katalogu do pliku konfiguracyjnego.
ListenBacklog	Maksymalna liczba oczekujących połączeń w kolejce TCP.
ListenIP	Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać.

Parameter	Description
ListenPort	agent będzie nasłuchiwał na tym porcie połączeń od serwer.
LoadModule	Moduł do załadowania podczas uruchamiania agent.
LoadModulePath	Pełna ścieżka do lokalizacji modułów agent.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogType	Typ wyjścia dziennika.
MaxLinesPerSecond	Maksymalna liczba nowych wierszy, które agent wyśle na sekundę do Zabbix serwer lub proxy podczas przetwarzania aktywnych kontroli ’log’ i ’logrt’.
PidFile	Nazwa pliku PID.
RefreshActiveChecks	Jak często odświeżana jest lista aktywnych kontroli.
Server	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix.
ServerActive	Adres Zabbix serwer/proxy lub konfiguracja klastra, z którego pobierane są aktywne kontrole.
SourceIP	Źródłowy adres IP.
StartAgents	Liczba wstępnie rozwidlonych instancji zabbix_agentd przetwarzających kontrole pasywne.
Timeout	Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z Zabbix proxy lub serwer.
TLSAccept	Jakie połączenia przychodzące akceptować.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatów partnera, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Określa, w jaki sposób agent powinien łączyć się z Zabbix serwer lub proxy.
TLSCRLFile	Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny agent, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Pełna ścieżka do pliku zawierającego klucz współdzielony agent, używana do szyfrowanej komunikacji z Zabbix serwer.
TLSPSKIdentity	Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z Zabbix serwer.
TLSServerCertIssuer	Dozwolony wystawca certyfikatu serwer (proxy).
TLSServerCertSubject	Dozwolony podmiot certyfikatu serwer (proxy).
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
User	Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
UserParameter	Parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Domyślna ścieżka wyszukiwania dla poleceń UserParameter.

Wszystkie parametry są opcjonalne, chyba że wyraźnie stwierdzono, że dany parametr jest obowiązkowy.

Zwróć uwagę, że:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości w dostarczanych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne tylko w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

Alias

Ustawia alias dla klucza pozycji. Może być używany do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów *Alias*. Wiele parametrów z tym samym kluczem *Alias* jest niedozwolonych.
 Różne klucze *Alias* mogą wskazywać ten sam klucz pozycji.
 Aliasy mogą być używane w parametrze *HostMetadataItem*, ale nie w parametrze *HostnameItem*.

Przykład 1: Pobieranie identyfikatora użytkownika „zabbix".

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Teraz skrócony klucz **zabbix.userid** może być używany do pobierania danych.

Przykład 2: Pobieranie wykorzystania CPU z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Pozwala to używać klucza **cpu.util** do pobierania procentowego wykorzystania CPU z parametrami domyślnymi, a także **cpu.util[all, idle, avg15]** do pobierania szczegółowych danych o wykorzystaniu CPU.

Przykład 3: Uruchamianie wielu reguł odkrywania niskiego poziomu przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz możliwe jest skonfigurowanie kilku reguł wykrywania używających **vfs.fs.discovery** z różnymi parametrami dla każdej reguły, np. **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak "*", który dopasowuje dowolną liczbę dowolnych znaków.
Można zdefiniować wiele reguł dopasowywania kluczy w połączeniu z DenyKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

AllowRoot

Zezwala agentowi na działanie jako „root". Jeśli opcja jest wyłączona, a agent zostanie uruchomiony przez użytkownika „root", agent spróbuje zamiast tego przełączyć się na użytkownika „zabbix". Nie ma to wpływu, jeśli agent został uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

BufferSend

Nie przechowuj danych w buforze dłużej niż N sekund.

Wartość domyślna: 5
 Zakres: 1-3600

BufferSize

Maksymalna liczba wartości w buforze pamięci. agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor będzie pełny.

Domyślnie: 100
 Zakres: 2-65535

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak "*", który dopasowuje dowolną liczbę dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z AllowKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

EnableRemoteCommands

Czy zdalne polecenia z serwera Zabbix są dozwolone. Ten parametr jest **przestarzały**; zamiast niego użyj AllowKey=system.run[*] lub DenyKey=system.run[*].
Jest to wewnętrzny alias parametrów AllowKey/DenyKey zależnie od wartości:
0 - DenyKey=system.run[*]
1 - AllowKey=system.run[*]

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

HeartbeatFrequency

Częstotliwość komunikatów heartbeat w sekundach. Używana do monitorowania dostępności aktywnych kontroli.
0 - komunikaty heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Opcjonalny parametr definiujący interfejs hosta (adres IP lub nazwę DNS) używany podczas procesu autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawną konfigurację adresu IP lub DNS. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Opcjonalny parametr definiujący pozycję używaną do określenia interfejsu hosta (adresu IP lub nazwy DNS) podczas procesu autorejestracji hosta. Ta wartość jest używana tylko wtedy, gdy HostInterface nie jest zdefiniowany. Więcej szczegółów można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekracza limit 255 znaków.

Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

HostMetadata

Opcjonalny parametr określający metadane używane do identyfikacji lub rozróżniania hosta podczas procesu autorejestracji hosta (aktywny agent). HostMetadata umożliwia rozróżnianie hostów niezależnie od nazwy hosta.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostMetadataItem.

agent zgłosi błąd i nie uruchomi się, jeśli podana wartość przekracza limit 2034 bajtów lub jest ciągiem w formacie innym niż UTF-8. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Zakres: 0-2034 bajtów

HostMetadataItem

Opcjonalny parametr, który definiuje pozycję agenta Zabbix używaną do pobierania metadanych hosta. Ta opcja jest używana tylko wtedy, gdy parametr HostMetadata nie jest zdefiniowany.

Wartość HostMetadataItem jest pobierana przy każdej próbie autorejestracji i jest używana wyłącznie w procesie autorejestracji hosta (aktywny agent). HostMetadataItem umożliwia rozróżnianie hostów poza nazwą hosta.

Obsługiwane są parametry użytkownika i aliasy. Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekracza limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem UTF-8, w przeciwnym razie zostanie zignorowana. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości, które są poprawnym UTF-8, ale nie są poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Hostname

Lista unikalnych nazw hostów rozdzielonych przecinkami, z rozróżnianiem wielkości liter. Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z HostnameItem, jeśli nie jest zdefiniowana.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania nazwy hosta. Ta opcja jest używana tylko wtedy, gdy parametr Hostname nie jest zdefiniowany. Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja system.run[] jest obsługiwana niezależnie od wartości AllowKey/DenyKey.

Domyślnie: `system.hostname`

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym. Aby uwzględnić tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenBacklog

Maksymalna liczba oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na sztywno i zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać po cichu obcięte do „maksimum określonego przez implementację”.

Domyślnie: SOMAXCONN
 Zakres: 0 - INT_MAX

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać.

Wartość domyślna: `0.0.0.0`

ListenPort

agent będzie nasłuchiwał na tym porcie połączeń z serwera.

Domyślnie: 10050
 Zakres: 1024-32767

LoadModule

Moduł do załadowania podczas uruchamiania agenta. Moduły są używane do rozszerzania funkcjonalności agenta. Moduł musi znajdować się w katalogu określonym przez LoadModulePath lub ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od ’/’), wtedy LoadModulePath jest ignorowane.
Formaty:
LoadModule=<module.so>
LoadModule jest uwzględnienie wielu parametrów LoadModule.

LoadModulePath

Pełna ścieżka do lokalizacji modułów agenta. Wartość domyślna zależy od opcji kompilacji.

LogFile

Nazwa pliku dziennika.

Wymagane: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 — wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty od nowa.

Wartość domyślna: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń. Polecenia będą rejestrowane tylko wtedy, gdy są wykonywane zdalnie. Wpisy dziennika nie będą tworzone, jeśli system.run[] jest uruchamiane lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogType

Typ wyjścia logu:
file - zapisuje log do pliku określonego parametrem LogFile;
system - zapisuje log do syslog;
console - zapisuje log na standardowe wyjście.

Wartość domyślna: `file`

MaxLinesPerSecond

Maksymalna liczba nowych linii, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych kontroli „log" i „logrt". Podana wartość zostanie nadpisana przez parametr „maxlines" podany w kluczu pozycji „log" lub „logrt".
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w *MaxLinesPerSecond*, aby wyszukać wymagany ciąg znaków w pozycjach logów.

Wartość domyślna: 20
 Zakres: 1-1000

PidFile

Nazwa pliku PID.

Domyślnie: `/tmp/zabbix_agentd.pid`

RefreshActiveChecks

Jak często odświeżana jest lista aktywnych kontroli, w sekundach. Należy pamiętać, że po nieudanej próbie odświeżenia aktywnych kontroli kolejna próba odświeżenia zostanie podjęta za 60 sekund.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix. Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' pozwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może być użyte do zezwolenia na dowolny adres IPv4. Należy pamiętać, że „adresy IPv6 zgodne z IPv4" (prefiks 0000::/96) są obsługiwane, ale przestarzałe zgodnie z RFC4291. Spacje są dozwolone.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak, jeśli StartAgents nie jest jawnie ustawione na 0

ServerActive

Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego mają być pobierane aktywne kontrole. Adres serwera/proxy jest adresem IP lub nazwą DNS oraz opcjonalnym portem oddzielonym dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwerów lub proxy oddzielonych średnikiem. Można określić wiele serwerów/klastrów Zabbix i proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy określać więcej niż jednego proxy Zabbix dla każdego serwera/klastra Zabbix. Jeśli określono proxy Zabbix, nie należy określać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby używać kilku niezależnych serwerów Zabbix równolegle. Dozwolone są spacje.
Jeśli port nie zostanie określony, używany jest domyślny port.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie jest określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne kontrole są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.).

StartAgents

Liczba wstępnie utworzonych instancji zabbix_agentd, które przetwarzają kontrole pasywne. Jeśli ustawiono 0, kontrole pasywne są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Wartość domyślna: 10
 Zakres: 0-100

Timeout

Określa, jak długo czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr określa czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwania na odpowiedź od serwera Zabbix;
- wysyłania żądań do serwera Zabbix, w tym żądań konfiguracji pozycji oraz danych pozycji w aktywnych testach;
- pobierania danych logów przez logfile;
- wysyłania komunikatów heartbeat;
- maksymalnego czasu trwania testów `vfs.*`;
- użycia przez moduły agenta Zabbix;
- użycia jako wartości zapasowej w scenariuszach, gdy serwer lub proxy starszy niż wersja 7.0 wysyła testy bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych testów agenta, które mają konfigurowalne ustawienia limitu czasu we frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).

Wartość domyślna: 3
 Zakres: 1-30

TLSAccept

Jakie połączenia przychodzące akceptować. Używane dla pasywnych kontroli. Można określić wiele wartości, oddzielonych przecinkami:
unencrypted - akceptuj połączenia bez szyfrowania (domyślnie)
psk - akceptuj połączenia z TLS i kluczem współdzielonym (PSK)
cert - akceptuj połączenia z TLS i certyfikatem

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) używane do weryfikacji certyfikatu peera, wykorzystywana do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Jak agent powinien łączyć się z serwerem lub proxy Zabbix. Używane dla aktywnych testów. Można określić tylko jedną wartość:
unencrypted - połączenie bez szyfrowania (domyślnie)
psk - połączenie z użyciem TLS i klucza współdzielonego (PSK)
cert - połączenie z użyciem TLS i certyfikatu

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Pełna ścieżka do pliku zawierającego klucz współdzielony agenta, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSPSKIdentity

Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Dozwolony wystawca certyfikatu serwera (proxy).

TLSServerCertSubject

Dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

Użytkownik

Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
Ma zastosowanie tylko w przypadku uruchomienia jako `root`, gdy opcja AllowRoot jest wyłączona.

Wartość domyślna: `zabbix`

UserParameter

Parametr zdefiniowany przez użytkownika do monitorowania. Może istnieć kilka parametrów zdefiniowanych przez użytkownika.
Format: UserParameter=<key>,<shell command>
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr UserParameterDir.

Przykład:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Domyślna ścieżka wyszukiwania dla poleceń UserParameter. Jeśli jest używana, agent zmieni swój katalog roboczy na katalog określony tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą mieć względny prefiks `./` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=/opt/myscripts
```

Zobacz także

1. Różnice w konfiguracji agenta Zabbix dla aktywnych i pasywnych kontroli począwszy od wersji 2.0.0

4 Zabbix agent 2 (UNIX)

Przegląd

Zabbix agent 2 to nowa generacja agenta Zabbix i może być używany zamiast agenta Zabbix.

Parametry obsługiwane przez plik konfiguracyjny Zabbix agent 2 (zabbix_agent2.conf) są wymienione w tej sekcji.

Parametry są wymienione bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
Alias	Ustawia alias dla klucza pozycji.
AllowKey	Zezwala na wykonywanie tych kluczy pozycji, które pasują do wzorca.
BufferSend	Nie przechowuje danych w buforze dłużej niż N sekund.
BufferSize	Maksymalna liczba wartości w buforze pamięci.
ControlSocket	Gniazdo sterujące używane do wysyłania poleceń w czasie działania za pomocą opcji '-R'.
DebugLevel	Poziom debugowania.
DenyKey	Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca.
EnablePersistentBuffer	Włącza użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji.
ForceActiveChecksOnStart	Wykonuje aktywne sprawdzenia natychmiast po restarcie dla pierwszej otrzymanej konfiguracji.
HeartbeatFrequency	Częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Opcjonalny parametr definiujący interfejs hosta.
HostInterfaceItem	Opcjonalny parametr definiujący pozycję używaną do pobierania interfejsu hosta.
HostMetadata	Opcjonalny parametr definiujący metadane hosta.
HostMetadataItem	Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania metadanych hosta.
Hostname	Opcjonalny parametr definiujący nazwę hosta.
HostnameItem	Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania nazwy hosta.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki z katalogu do pliku konfiguracyjnego.
ListenIP	Lista adresów IP oddzielonych przecinkami, na których agent powinien nasłuchiwać.
ListenPort	Agent będzie nasłuchiwał na tym porcie połączeń z serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogType	Typ wyjścia dziennika.
PersistentBufferFile	Plik, w którym Zabbix agent 2 powinien przechowywać bazę danych SQLite.
PersistentBufferPeriod	Okres, przez który dane powinny być przechowywane, gdy nie ma połączenia z serwerem lub proxy.
PidFile	Nazwa pliku PID.
Plugins.<PluginName>.System.Capacity	Limit sprawdzeń na plugin, które mogą być wykonywane jednocześnie.
Plugins.Log.MaxLinesPerSecond	Maksymalna liczba nowych wierszy, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych sprawdzeń 'log' i 'logrt'.
Plugins.SystemRun.LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
PluginSocket	Ścieżka do gniazda UNIX dla komunikacji z ładowalnymi pluginami.
PluginTimeout	Limit czasu połączeń z ładowalnymi pluginami, w sekundach.
RefreshActiveChecks	Jak często odświeżana jest lista aktywnych sprawdzeń.
Server	Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix.

Parameter	Description
ServerActive	Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego pobierane są aktywne sprawdzenia.
SourceIP	Źródłowy adres IP.
StatusPort	Jeśli ustawiony, agent będzie nasłuchiwał na tym porcie żądań statusu HTTP (http://localhost:<port>/status).
Timeout	Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z proxy Zabbix lub serwerem.
TLSAccept	Jakie połączenia przychodzące akceptować.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	W jaki sposób agent powinien łączyć się z serwerem Zabbix lub proxy.
TLSCRLFile	Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Pełna ścieżka do pliku zawierającego klucz współdzielony agenta, używana do szyfrowanej komunikacji z serwerem Zabbix.
TLSPSKIdentity	Ciąg identyfikatora klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.
TLSServerCertIssuer	Dozwolony wystawca certyfikatu serwera (proxy).
TLSServerCertSubject	Dozwolony podmiot certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Domyślna ścieżka wyszukiwania dla poleceń UserParameter.

Wszystkie parametry są opcjonalne, chyba że wyraźnie stwierdzono, że parametr jest obowiązkowy.

Uwaga:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne tylko w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

Alias

Ustawia alias dla klucza pozycji. Może być używany do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów *Alias*. Wiele parametrów z tym samym kluczem *Alias* jest niedozwolonych.
 Różne klucze *Alias* mogą wskazywać ten sam klucz pozycji.
 Aliasy mogą być używane w parametrze *HostMetadataItem*, ale nie w parametrze *HostnameItem*.

Przykład 1: Pobieranie identyfikatora użytkownika „zabbix”.

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Teraz skrócony klucz **zabbix.userid** może być używany do pobierania danych.

Przykład 2: Pobieranie wykorzystania CPU z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Pozwala to używać klucza **cpu.util** do uzyskania procentowego wykorzystania CPU z parametrami domyślnymi, a także używać **cpu.util[all, idle, avg15]** do uzyskania szczegółowych danych o wykorzystaniu CPU.

Przykład 3: Uruchamianie wielu reguł odkrywania niskiego poziomu przetwarzających te same pozycje odkrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz możliwe jest skonfigurowanie kilku reguł odkrywania używających **vfs.fs.discovery** z różnymi parametrami dla każdej reguły, np. **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak ”*”, który dopasowuje dowolną liczbę dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z DenyKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

BufferSend

Interwał czasu w sekundach określający, jak często wartości są wysyłane z bufora do serwera Zabbix. Należy pamiętać, że jeśli bufor jest pełny, dane zostaną wysłane wcześniej.

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Maksymalna liczba wartości w buforze pamięci. agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor jest pełny. Ten parametr powinien być używany tylko wtedy, gdy trwały bufor jest wyłączony (*EnablePersistentBuffer=0*).

Domyślnie: 1000
 Zakres: 2-65535

ControlSocket

Gniazdo sterujące, używane do wysyłania poleceń w czasie działania za pomocą opcji ’-R’.

Domyślnie: `/tmp/agent.sock`

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak ”*”, który dopasowuje dowolną liczbę dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z AllowKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

EnablePersistentBuffer

Włącz użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji. Jeśli trwała pamięć masowa jest wyłączona, użyty zostanie bufor pamięci.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

ForceActiveChecksOnStart

Wykonuj aktywne sprawdzenia natychmiast po restarcie dla pierwszej otrzymanej konfiguracji. Dostępne również jako parametr konfiguracji dla każdej wtyczki, na przykład: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

HeartbeatFrequency

Częstotliwość komunikatów heartbeat w sekundach. Używana do monitorowania dostępności aktywnych testów.
0 - komunikaty heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Opcjonalny parametr definiujący interfejs hosta (adres IP lub nazwę DNS) używany podczas procesu autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawne skonfigurowanie adresu IP lub DNS. Aby uzyskać więcej informacji, zobacz Używanie DNS jako domyślnego interfejsu.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Opcjonalny parametr definiujący pozycję używaną do określenia interfejsu hosta (adresu IP lub nazwy DNS) podczas procesu autorejestracji. Ta wartość jest używana tylko wtedy, gdy HostInterface nie jest zdefiniowany. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

HostMetadata

Opcjonalny parametr określający metadane używane do identyfikacji lub rozróżniania hosta podczas procesu autorejestracji hosta. HostMetadata umożliwia rozróżnianie hostów na podstawie czegoś więcej niż tylko nazwy hosta.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostMetadataItem.

agent zgłosi błąd i nie uruchomi się, jeśli określona wartość przekracza limit 2034 bajtów lub jest ciągiem w formacie innym niż UTF-8. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Zakres: 0-2034 bajtów

HostMetadataItem

Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania metadanych hosta. Ta opcja jest używana tylko wtedy, gdy parametr HostMetadata nie jest zdefiniowany.

Wartość HostMetadataItem jest pobierana przy każdej próbie autorejestracji i jest używana wyłącznie w procesie autorejestracji hosta. HostMetadataItem umożliwia rozróżnianie hostów nie tylko na podstawie nazwy hosta.

Parametry użytkownika i aliasy są obsługiwane. Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem UTF-8, w przeciwnym razie zostanie zignorowana. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Hostname

Lista unikalnych nazw hostów rozdzielonych przecinkami, z rozróżnianiem wielkości liter. Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z HostnameItem, jeśli nie jest zdefiniowana.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Opcjonalny parametr definiujący pozycję używaną do pobierania nazwy hosta. Ta opcja jest używana tylko wtedy, gdy Hostname nie jest zdefiniowany. Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja system.run[] jest obsługiwana niezależnie od wartości AllowKey/DenyKey.

Wartość domyślna: `system.hostname`

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym. Podczas instalacji Zabbix utworzy katalog include w /usr/local/etc, chyba że zostanie to zmienione podczas kompilacji. Ścieżka może być względna względem lokalizacji

pliku *zabbix_agent2.conf*.
Aby dołączyć tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać. Pierwszy adres IP jest wysyłany do serwera Zabbix, jeśli agent łączy się z nim, aby pobrać listę aktywnych kontroli.

Wartość domyślna: `0.0.0.0`

ListenPort

agent będzie nasłuchiwał na tym porcie połączeń z serwera.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika.

Domyślnie: `/tmp/zabbix_agent2.log`
 Wymagane: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty od nowa.

Wartość domyślna: 1
 Zakres: 0-1024

LogType

Typ wyjścia logu:
file - zapisuj log do pliku określonego parametrem LogFile;
system - zapisuj log do syslog;
console - zapisuj log na standardowe wyjście

Domyślnie: `file`

PersistentBufferFile

Plik, w którym agent Zabbix 2 powinien przechowywać bazę danych SQLite. Musi to być pełna nazwa pliku. Ten parametr jest używany tylko wtedy, gdy trwały bufor jest włączony (*EnablePersistentBuffer=1*).

PersistentBufferPeriod

Okres, przez który dane powinny być przechowywane, gdy nie ma połączenia z serwerem lub proxy. Starsze dane zostaną utracone. Dane dziennika zostaną zachowane. Ten parametr jest używany tylko wtedy, gdy trwały bufor jest włączony (*EnablePersistentBuffer=1*).

Wartość domyślna: `1h`
 Zakres: 1m-365d

PidFile

Nazwa pliku PID.

Domyślnie: `/tmp/zabbix_agent2.pid`

Plugins.<PluginName>.System.Capacity

Limit kontroli na plugin `<PluginName>`, które mogą być wykonywane jednocześnie.

Domyślnie: 1000 Zakres: 1-1000

Plugins.Log.MaxLinesPerSecond

Maksymalna liczba nowych linii, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych kontroli 'log' i 'logrt'. Podana wartość zostanie nadpisana przez parametr 'maxlines' podany w kluczu pozycji 'log' i 'logrt'.
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w *MaxLinesPerSecond*, aby wyszukać wymagany ciąg znaków w pozycjach logów.

Wartość domyślna: 20
 Zakres: 1-1000

Plugins.SystemRun.LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń. Polecenia będą rejestrowane tylko wtedy, gdy są wykonywane zdalnie. Wpisy dziennika nie będą tworzone, jeśli system.run[] jest uruchamiane lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

PluginSocket

Ścieżka do gniazda UNIX używanego do komunikacji z ładowalnymi wtyczkami.

Domyślnie: `/tmp/agent.plugin.sock`

PluginTimeout

Limit czasu połączeń z ładowalnymi wtyczkami, w sekundach.

Wartość domyślna: `Timeout`
 Zakres: 1-30

RefreshActiveChecks

Jak często odświeżana jest lista aktywnych testów, w sekundach. Należy pamiętać, że po nieudanej próbie odświeżenia aktywnych testów kolejna próba odświeżenia zostanie podjęta za 60 sekund.

Wartość domyślna: 5
 Zakres: 1-86400

Serwer

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' zezwoli na dowolny adres IPv4 lub IPv6. Aby zezwolić na dowolny adres IPv4, można użyć '0.0.0.0/0'. Spacje są dozwolone. Jeśli ten parametr nie jest określony, wyłącza kontrole pasywne, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego pobierane są aktywne kontrole. Adres serwera/proxy to adres IP lub nazwa DNS oraz opcjonalny port oddzielony dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwera lub proxy, oddzielonych średnikiem. Można określić wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy określać więcej niż jednego proxy Zabbix z każdego serwera/klastra Zabbix. Jeśli określono proxy Zabbix, nie należy określać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby równolegle używać kilku niezależnych serwerów Zabbix. Spacje są dozwolone.
Jeśli port nie został określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla danego hosta określono port. Jeśli port nie został określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne kontrole są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix.
- nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.).

StatusPort

Jeśli ustawione, agent będzie nasłuchiwał na tym porcie żądań statusu HTTP (http://localhost:<port>/status).

Zakres: 1024-32767

Timeout

Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź od serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji oraz danych pozycji w active checks;
- pobieranie danych logów przez logfile;
- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania kontroli `vfs.*`;
- użycie jako wartości zapasowej w sytuacjach, gdy serwer lub proxy starszy niż wersja 7.0 wysyła kontrole bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych kontroli agenta, które mają konfigurowalne ustawienia limitu czasu w frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).

Wartość domyślna: 3
 Zakres: 1-30

TLSAccept

Przychodzące połączenia do zaakceptowania. Używane dla pasywnych kontroli. Można określić wiele wartości, oddzielonych przecinkami:
unencrypted - akceptuj połączenia bez szyfrowania (domyślnie)
psk - akceptuj połączenia z TLS i kluczem współdzielonym (PSK)
cert - akceptuj połączenia z TLS i certyfikatem

Obowiązkowe: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu lub urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll

Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherAll13

Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert

Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

Należy pamiętać, że ten parametr nie może być używany razem z `TLSAccept=cert,psk`; w przypadku połączenia certyfikatowego (`TLSConnect=cert`) należy zamiast tego użyć TLSCipherAll.

TLSCipherCert13

Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Należy pamiętać, że tego parametru nie można używać razem z `TLSAccept=cert,psk`; w przypadku połączenia certyfikatowego (`TLSConnect=cert`) należy zamiast tego użyć TLSCipherAll13.

TLSCipherPSK

Ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Jak agent powinien łączyć się z serwerem Zabbix lub proxy. Używane dla aktywnych testów. Można określić tylko jedną wartość:
unencrypted - połącz bez szyfrowania (domyślnie)
psk - połącz przy użyciu TLS i klucza współdzielonego (PSK)
cert - połącz przy użyciu TLS i certyfikatu

Obowiązkowe: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Pełna ścieżka do pliku zawierającego klucz współdzielony agent, używany do szyfrowanej komunikacji z serwer Zabbix.

TLSPSKIdentity

Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Dozwolony wystawca certyfikatu serwera (proxy).

TLSServerCertSubject

Dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Parametr zdefiniowany przez użytkownika do monitorowania. Może istnieć kilka parametrów zdefiniowanych przez użytkownika.
Format: UserParameter=<key>,<polecenie powłoki>
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr UserParameterDir.

Przykład:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Domyślna ścieżka wyszukiwania dla poleceń UserParameter. Jeśli jest używana, agent zmieni swój katalog roboczy na określony tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą mieć względny prefiks `./` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=/opt/myscripts
```

5 Zabbix agent (Windows)

Przegląd

Parametry obsługiwane przez plik konfiguracyjny agenta Zabbix dla systemu Windows (zabbix_agentd.conf) są wymienione w tej sekcji.

Parametry są wymienione bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
Alias	Ustawia alias dla klucza pozycji.
AllowKey	Zezwala na wykonywanie tych kluczy pozycji, które pasują do wzorca.
BufferSend	Nie przechowuje danych w buforze dłużej niż N sekund.
BufferSize	Maksymalna liczba wartości w buforze pamięci.
DebugLevel	Poziom debugowania.
DenyKey	Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca.
EnableRemoteCommands	Określa, czy zdalne polecenia z serwera Zabbix są dozwolone.
HeartbeatFrequency	Częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Opcjonalny parametr definiujący interfejs hosta.
HostInterfaceItem	Opcjonalny parametr definiujący pozycję używaną do pobierania interfejsu hosta.
HostMetadata	Opcjonalny parametr definiujący metadane hosta.
HostMetadataItem	Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania metadanych hosta.
Hostname	Opcjonalny parametr definiujący nazwę hosta.
HostnameItem	Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania nazwy hosta.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki z katalogu do pliku konfiguracyjnego.
ListenBacklog	Maksymalna liczba oczekujących połączeń w kolejce TCP.
ListenIP	Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać.
ListenPort	Agent będzie nasłuchiwał na tym porcie połączeń z serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogType	Typ danych wyjściowych dziennika.
MaxLinesPerSecond	Maksymalna liczba nowych wierszy, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych kontroli 'log' i 'logrt'.
PerfCounter	Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
PerfCounterEn	Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach). W porównaniu do PerfCounter ścieżki perfcounter muszą być w języku angielskim.
RefreshActiveChecks	Jak często odświeżana jest lista aktywnych kontroli.
Server	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix.
ServerActive	Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego pobierane są aktywne kontrole.
SourceIP	Źródłowy adres IP.
StartAgents	Liczba wstępnie utworzonych instancji zabbix_agentd przetwarzających kontrole pasywne.
Timeout	Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z proxy lub serwerem Zabbix.
TLSAccept	Jakie połączenia przychodzące akceptować.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSConnect	Jak agent powinien łączyć się z serwerem Zabbix lub proxy.
TLSCRLFile	Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używana do szyfrowanej komunikacji między komponentami Zabbix.

Parameter	Description
TLSPSKFile	Pełna ścieżka do pliku zawierającego klucz współdzielony agenta, używana do szyfrowanej komunikacji z serwerem Zabbix.
TLSPSKIdentity	Ciąg identyfikujący klucz współdzielony, używany do szyfrowanej komunikacji z serwerem Zabbix.
TLSServerCertIssuer	Dozwolony wystawca certyfikatu serwera (proxy).
TLSServerCertSubject	Dozwolony temat certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Domyślna ścieżka wyszukiwania dla poleceń UserParameter.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Uwaga:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

Alias

Ustawia alias dla klucza pozycji. Może być używany do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów *Alias*. Wiele parametrów z tym samym kluczem *Alias* jest niedozwolonych.
 Różne klucze *Alias* mogą wskazywać ten sam klucz pozycji.
 Aliasów można używać w *HostMetadataItem*, ale nie w parametrze *HostnameItem* ani *PerfCounter*.

Przykład 1: Pobieranie użycia pliku stronicowania w procentach z serwera.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Teraz skrócony klucz **pg_usage** może być używany do pobierania danych.

Przykład 2: Pobieranie obciążenia CPU z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Pozwala to używać klucza **cpu.load** do pobierania obciążenia CPU z parametrami domyślnymi, a także **cpu.load[percpu,avg15]** do pobierania szczegółowych danych o obciążeniu CPU.

Przykład 3: Uruchamianie wielu reguł odkrywania niskiego poziomu przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz możliwe jest skonfigurowanie kilku reguł wykrywania używających **vfs.fs.discovery** z różnymi parametrami dla każdej reguły, np. **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym, które obsługuje znak ”*” do dopasowania dowolnej liczby dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z DenyKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

BufferSend

Nie przechowuj danych w buforze dłużej niż N sekund.

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Maksymalna liczba wartości w buforze pamięci. agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor będzie pełny.

Domyślnie: 100
 Zakres: 2-65535

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak "*" do dopasowania dowolnej liczby dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z AllowKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

EnableRemoteCommands

Czy zdalne polecenia z serwera Zabbix są dozwolone. Ten parametr jest **przestarzały**; zamiast niego użyj AllowKey=system.run[*] lub DenyKey=system.run[*].
Jest to wewnętrzny alias parametrów AllowKey/DenyKey, zależnie od wartości:
0 - DenyKey=system.run[*]
1 - AllowKey=system.run[*]

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

HeartbeatFrequency

Częstotliwość komunikatów heartbeat w sekundach. Używana do monitorowania dostępności aktywnych testów.
0 - komunikaty heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Opcjonalny parametr definiujący interfejs hosta (adres IP lub nazwę DNS) używany podczas procesu autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawną konfigurację adresu IP lub DNS. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Opcjonalny parametr definiujący pozycję używaną do określenia interfejsu hosta (adresu IP lub nazwy DNS) podczas procesu autorejestracji hosta. Ta wartość jest używana tylko wtedy, gdy HostInterface nie jest zdefiniowany. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

HostMetadata

Opcjonalny parametr określający metadane używane do identyfikacji lub rozróżniania hosta podczas procesu autorejestracji hosta (aktywny agent). HostMetadata umożliwia rozróżnianie hostów niezależnie od nazwy hosta.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostMetadataItem.

agent zgłosi błąd i nie uruchomi się, jeśli podana wartość przekracza limit 2034 bajtów lub jest ciągiem w formacie innym niż UTF-8. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Zakres: 0-2034 bajtów

HostMetadataItem

Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania metadanych hosta. Ta opcja jest używana tylko wtedy, gdy HostMetadata nie jest zdefiniowany.

Wartość HostMetadataItem jest pobierana przy każdej próbie autorejestracji i jest używana wyłącznie w procesie autorejestracji hosta (agent aktywny). HostMetadataItem umożliwia rozróżnianie hostów na podstawie czegoś więcej niż tylko nazwy hosta.

Obsługiwane są parametry użytkownika, liczniki wydajności i aliasy. Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

Podczas żądania autorejestracji agent zapisze ostrzeżenie w logu, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem UTF-8, w przeciwnym razie zostanie zignorowana. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Hostname

Lista unikalnych nazw hostów rozdzielonych przecinkami, z rozróżnianiem wielkości liter. Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z HostnameItem, jeśli nie jest zdefiniowana.
Dozwolone znaki: znaki alfanumeryczne, '.', ' ', '_' oraz '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Opcjonalny parametr, który definiuje pozycję agenta Zabbix używaną do pobierania nazwy hosta. Ta opcja jest używana tylko wtedy, gdy Hostname nie jest zdefiniowany. Parametry użytkownika, liczniki wydajności ani aliasy nie są obsługiwane, ale pozycja system.run[] jest obsługiwana niezależnie od wartości AllowKey/DenyKey.
Zobacz także bardziej szczegółowy opis.

Wartość domyślna: `system.hostname`

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym (domyślnie znajdującym się w `C:\Program Files\Zabbix Agent`, jeśli Zabbix agent jest zainstalowany przy użyciu pakietów instalatora Windows MSI; znajdującym się w folderze określonym podczas instalacji, jeśli Zabbix agent jest zainstalowany jako archiwum zip). Wszystkie dołączone pliki muszą mieć poprawną składnię, w przeciwnym razie agent nie uruchomi się.
Aby uwzględnić tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=C:\Program Files\Zabbix Agent\zabbix_agentd.d\*.conf
```

ListenBacklog

Maksymalna liczba oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na stałe, która zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać po cichu obcięte do „maksimum określonego przez implementację".

Domyślnie: `SOMAXCONN`
 Zakres: 0 - INT_MAX

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać.

Wartość domyślna: `0.0.0.0`

ListenPort

agent będzie nasłuchiwał na tym porcie połączeń z serwera.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika agenta.

Domyślnie: `c:\zabbix_agentd.log`
 Wymagane: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń. Polecenia będą rejestrowane tylko wtedy, gdy są wykonywane zdalnie. Wpisy dziennika nie będą tworzone, jeśli `system.run[]` jest uruchamiane lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogType

Typ wyjścia logu:
file - zapisuj log do pliku określonego parametrem LogFile;
system - zapisuj log do dziennika zdarzeń systemu Windows;
console - zapisuj log na standardowe wyjście.

Domyślnie: `file`

MaxLinesPerSecond

Maksymalna liczba nowych linii, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych kontroli 'log', 'logrt' i 'eventlog'. Podana wartość zostanie nadpisana przez parametr 'maxlines' podany w kluczu pozycji 'log', 'logrt' lub 'eventlog'.
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w *MaxLinesPerSecond*, aby wyszukać wymagany ciąg znaków w pozycjach logów.

Domyślnie: 20
 Zakres: 1-1000

PerfCounter

Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
Składnia: <parameter_name>,"<perf_counter_path>",<period>

Na przykład, jeśli chcesz otrzymać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr "interrupts" w następujący sposób:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania wartości średniej będą pobierane co sekundę.
Możesz uruchomić "typeperf -qx", aby uzyskać listę wszystkich liczników wydajności dostępnych w systemie Windows.

PerfCounterEn

Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach). W porównaniu do PerfCounter ścieżki liczników wydajności muszą być w języku angielskim. Obsługiwane tylko w **Windows Server 2008/Vista** i nowszych.
Składnia: <parameter_name>,"<perf_counter_path>",<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr „interrupts" w następujący sposób:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania wartości średniej będą pobierane co sekundę.
Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Jak często odświeżana jest lista aktywnych kontroli, w sekundach. Należy pamiętać, że po nieudanej próbie odświeżenia aktywnych kontroli kolejna próba zostanie podjęta za 60 sekund.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Połączenia przychodzące będą akceptowane wyłącznie od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' zezwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może zostać użyte do zezwolenia na dowolny adres IPv4. Należy pamiętać, że „adresy IPv6 zgodne z IPv4" (prefiks 0000::/96) są obsługiwane, ale przestarzałe zgodnie z RFC4291. Spacje są dozwolone.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak, jeśli StartAgents nie jest jawnie ustawione na 0

ServerActive

Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego mają być pobierane aktywne testy. Adres serwera/proxy to adres IP lub nazwa DNS oraz opcjonalny port oddzielony dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwera lub proxy, oddzielonych średnikiem. Można określić wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy określać więcej niż jednego proxy Zabbix z każdego serwera/klastra

Zabbix. Jeśli określono proxy Zabbix, to nie należy określać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby równolegle używać kilku niezależnych serwerów Zabbix. Spacje są dozwolone.
Jeśli port nie został określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla danego hosta określono port. Jeśli port nie został określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne testy są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

Zakres: (*)

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.).

StartAgents

Liczba wstępnie utworzonych instancji zabbix_agentd, które przetwarzają pasywne testy. Jeśli ustawiono 0, pasywne testy są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Domyślnie: 10
 Zakres: 0-100 (*)

Timeout

Określa, jak długo czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr określa czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź od serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji oraz danych pozycji w active checks;
- pobieranie danych logów za pomocą monitorowania logfile lub dziennika zdarzeń Windows;
- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania kontroli `vfs.*`;
- używany przez moduły agenta Zabbix;
- stosowany jako ustawienie zapasowe w sytuacjach, gdy serwer lub proxy starszy niż wersja 7.0 wysyła kontrole bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych kontroli agenta, które mają konfigurowalne ustawienia limitu czasu we frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Przychodzące połączenia do akceptowania. Używane dla pasywnych testów. Można określić wiele wartości, oddzielonych przecinkami:
unencrypted - akceptuj połączenia bez szyfrowania (domyślnie)
psk - akceptuj połączenia z TLS i kluczem współdzielonym (PSK)
cert - akceptuj połączenia z TLS i certyfikatem

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji z komponentami Zabbix.

TLSConnect

Jak agent powinien łączyć się z serwerem Zabbix lub proxy. Używane dla aktywnych testów. Można określić tylko jedną wartość:
unencrypted - połączenie bez szyfrowania (domyślnie)
psk - połączenie z użyciem TLS i klucza współdzielonego (PSK)
cert - połączenie z użyciem TLS i certyfikatu

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Pełna ścieżka do pliku zawierającego klucz współdzielony agenta, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSPSKIdentity

Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Dozwolony wystawca certyfikatu serwera (proxy).

TLSServerCertSubject

Dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @
Ponadto znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Parametr zdefiniowany przez użytkownika do monitorowania. Może istnieć kilka parametrów zdefiniowanych przez użytkownika.
Format: UserParameter=<key>,<shell command>
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą mieć ścieżki względne, jeśli określono parametr UserParameterDir.

Przykład:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Domyślna ścieżka wyszukiwania dla poleceń UserParameter. Jeśli jest używana, agent zmieni swój katalog roboczy na określony tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą mieć względny prefiks `.\` zamiast pełnej ścieżki. Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=C:\opt\myscripts
```

Note:
(*) Liczba aktywnych serwerów wymienionych w ServerActive oraz liczba wstępnie utworzonych instancji dla pasywnych kontroli określona w StartAgents musi być mniejsza niż 64.

6 Zabbix agent 2 (Windows)

Przegląd

Zabbix agent 2 to nowa generacja Zabbix agent i może być używany zamiast Zabbix agent.

Parametry obsługiwane przez plik konfiguracyjny Zabbix agent 2 dla systemu Windows (zabbix_agent2.conf) są wymienione w tej sekcji.

Parametry są wymienione bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
Alias	Ustawia alias dla klucza pozycja.
AllowKey	Zezwala na wykonywanie tych kluczy pozycja, które pasują do wzorca.
BufferSend	Nie przechowuje danych w buforze dłużej niż N sekund.
BufferSize	Maksymalna liczba wartości w buforze pamięci.
ControlSocket	Gniazdo sterujące, używane do wysyłania poleceń w czasie działania za pomocą opcji '-R'.
DebugLevel	Poziom debugowania.
DenyKey	Odmowa wykonywania tych kluczy pozycja, które pasują do wzorca.
EnablePersistentBuffer	Włącza użycie lokalnej trwałej pamięci dla aktywnych pozycji.
ForceActiveChecksOnStart	Wykonuje aktywne kontrole natychmiast po restarcie, dla pierwszej otrzymanej konfiguracji.
HeartbeatFrequency	Częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Opcjonalny parametr definiujący interfejs hosta.
HostInterfaceItem	Opcjonalny parametr definiujący pozycja używaną do pobierania interfejsu hosta.
HostMetadata	Opcjonalny parametr definiujący metadane hosta.
HostMetadataItem	Opcjonalny parametr definiujący pozycja Zabbix agent używaną do pobierania metadanych hosta.
Hostname	Opcjonalny parametr definiujący nazwę hosta.
HostnameItem	Opcjonalny parametr definiujący pozycja Zabbix agent używaną do pobierania nazwy hosta.
Include	W pliku konfiguracyjnym można dołączyć pojedyncze pliki lub wszystkie pliki z katalogu.
ListenIP	Lista adresów IP rozdzielonych przecinkami, na których agent ma nasłuchiwać.
ListenPort	Agent będzie nasłuchiwał na tym porcie połączeń z serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogType	Typ wyjścia dziennika.
PersistentBufferFile	Plik, w którym Zabbix agent 2 powinien przechowywać bazę danych SQLite.
PersistentBufferPeriod	Okres, przez który dane powinny być przechowywane, gdy nie ma połączenia z serwerem lub proxy.
Plugins.<PluginName>.System.Capacity	Limit kontroli na plugin, które mogą być wykonywane jednocześnie.
Plugins.Log.MaxLinesPerSecond	Maksymalna liczba nowych linii, które agent będzie wysyłał na sekundę do Zabbix server lub proxy podczas przetwarzania aktywnych kontroli 'log' i 'logrt'.
Plugins.SystemRun.LogRemoteCommands	Włącza rejestrowanie wykonanych poleceń powłoki jako ostrzeżeń.
PluginSocket	Ścieżka do nazwanego potoku systemu Windows dla komunikacji z ładowalnymi pluginami.
PluginTimeout	Limit czasu połączeń z ładowalnymi pluginami, w sekundach.
PerfCounter	Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
PerfCounterEn	Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach). W porównaniu z PerfCounter, ścieżki perfcounter muszą być w języku angielskim.
RefreshActiveChecks	Jak często odświeżana jest lista aktywnych kontroli.
Server	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS Zabbix server i Zabbix proxy.
ServerActive	Adres Zabbix server/proxy lub konfiguracja klastra, z którego mają być pobierane aktywne kontrole.
SourceIP	Źródłowy adres IP.
StatusPort	Jeśli ustawione, agent będzie nasłuchiwał na tym porcie żądań stanu HTTP (http://localhost:<port>/status).
Timeout	Określa, jak długo czekać (w sekundach) na nawiązanie połączenia i wymianę danych z Zabbix proxy lub serwer.
TLSAccept	Jakie połączenia przychodzące akceptować.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównych CA, używane do weryfikacji certyfikatu peer w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używane do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll13	Łańcuch szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert13	Łańcuch szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Parameter	Description
TLSCipherPSK13	Łańcuch szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Sposób, w jaki agent ma łączyć się z Zabbix server lub proxy.
TLSCRLFile	Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego prywatny klucz agenta, używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Pełna ścieżka do pliku zawierającego klucz wstępnie współdzielony agenta, używany do szyfrowanej komunikacji z Zabbix server.
TLSPSKIdentity	Identyfikator klucza wstępnie współdzielonego, używany do szyfrowanej komunikacji z Zabbix server.
TLSServerCertIssuer	Dozwolony wystawca certyfikatu serwera (proxy).
TLSServerCertSubject	Dozwolony podmiot certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Domyślna ścieżka wyszukiwania dla poleceń UserParameter.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że parametr jest obowiązkowy.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

Alias

Ustawia alias dla klucza pozycji. Może być używany do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów *Alias*. Wiele parametrów z tym samym kluczem *Alias* jest niedozwolonych.
 Różne klucze *Alias* mogą wskazywać ten sam klucz pozycji.
 Aliasy mogą być używane w *HostMetadataItem*, ale nie w parametrze *HostnameItem*.

Przykład 1: Pobieranie użycia pliku stronicowania w procentach z serwera.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Teraz skrócony klucz **pg_usage** może być używany do pobierania danych.

Przykład 2: Pobieranie obciążenia CPU z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Pozwala to używać klucza **cpu.load** do pobierania obciążenia CPU z parametrami domyślnymi, a także klucza **cpu.load[percpu,avg15]** do pobierania szczegółowych danych o obciążeniu CPU.

Przykład 3: Uruchamianie wielu reguł low-level discovery przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz możliwe jest skonfigurowanie kilku reguł wykrywania używających **vfs.fs.discovery** z różnymi parametrami dla każdej reguły, np. **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak ”*” do dopasowania dowolnej liczby dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z DenyKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

BufferSend

The time interval in seconds which determines how often values are sent from the buffer to Zabbix server.
Note, that if the buffer is full, the data will be sent sooner.

Default: 5
 Range: 1-3600

BufferSize

Maksymalna liczba wartości w buforze pamięci. agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor będzie pełny.
Tego parametru należy używać tylko wtedy, gdy trwały bufor jest wyłączony (*EnablePersistentBuffer=0*).

Domyślnie: 1000
 Zakres: 2-65535

ControlSocket

Gniazdo sterujące, używane do wysyłania poleceń w czasie działania za pomocą opcji '-R'.

Domyślnie: `\\.\pipe\agent.sock`

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak "*", który dopasowuje dowolną liczbę dowolnych znaków.
Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z AllowKey. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Zobacz także: Ograniczanie kontroli agenta.

EnablePersistentBuffer

Włącz użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji. Jeśli trwała pamięć masowa jest wyłączona, użyty zostanie bufor pamięci.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

ForceActiveChecksOnStart

Wykonuj aktywne kontrole natychmiast po restarcie dla pierwszej otrzymanej konfiguracji. Dostępne również jako parametr konfiguracji dla poszczególnych wtyczek, na przykład: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Wartość domyślna: 0
 Wartości: 0 - wyłączone, 1 - włączone

HeartbeatFrequency

Częstotliwość komunikatów heartbeat w sekundach. Używana do monitorowania dostępności aktywnych testów.
0 - komunikaty heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Opcjonalny parametr definiujący interfejs hosta (adres IP lub nazwę DNS) używany podczas procesu autorejestracji. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawną konfigurację adresu IP lub DNS. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Jeśli parametr nie jest zdefiniowany, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Opcjonalny parametr definiujący pozycję używaną do określenia interfejsu hosta (adresu IP lub nazwy DNS) podczas procesu autorejestracji hosta. Ta wartość jest używana tylko wtedy, gdy HostInterface nie jest zdefiniowany. Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

HostMetadata

Opcjonalny parametr określający metadane używane do identyfikacji lub rozróżniania hosta podczas procesu autorejestracji hosta (aktywny agent). HostMetadata umożliwia rozróżnianie hostów nie tylko na podstawie nazwy hosta.

Jeśli nie jest zdefiniowany, wartość zostanie pobrana z HostMetadataItem.

agent zgłosi błąd i nie uruchomi się, jeśli podana wartość przekracza limit 2034 bajtów lub jest ciągiem znaków innym niż UTF-8. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca linii.

Zakres: 0-2034 bajtów

HostMetadataItem

Opcjonalny parametr definiujący pozycję agenta Zabbix używaną do pobierania metadanych hosta. Ta opcja jest używana tylko wtedy, gdy HostMetadata nie jest zdefiniowany.

Wartość HostMetadataItem jest pobierana przy każdej próbie autorejestracji i jest używana wyłącznie w procesie autorejestracji hosta (agent aktywny). HostMetadataItem umożliwia rozróżnianie hostów poza nazwą hosta.

Parametry użytkownika i aliasy są obsługiwane. Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey/DenyKey.

Podczas żądania autorejestracji agent zapisze komunikat ostrzegawczy w logu, jeśli wartość zwrócona przez określoną pozycję przekracza limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem UTF-8, w przeciwnym razie zostanie zignorowana. Gdy parametr oczekuje adresu IP lub nazwy DNS, wartości będące poprawnym UTF-8, ale niebędące poprawnymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Hostname

Lista unikalnych nazw hostów rozdzielonych przecinkami, z rozróżnianiem wielkości liter. Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z HostnameItem, jeśli nie jest zdefiniowana.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Opcjonalny parametr definiujący pozycję używaną do pobierania nazwy hosta. Ta opcja jest używana tylko wtedy, gdy parametr Hostname nie jest zdefiniowany. Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja system.run[] jest obsługiwana niezależnie od wartości AllowKey/DenyKey.

Wartość domyślna: `system.hostname`

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym (domyślnie znajdującym się w `C:\Program Files\Zabbix Agent 2`, jeśli Zabbix agent jest zainstalowany przy użyciu pakietów instalatora Windows MSI; znajdującym się w folderze określonym podczas instalacji, jeśli Zabbix agent jest zainstalowany jako archiwum zip). Wszystkie dołączane pliki muszą mieć poprawną składnię, w przeciwnym razie agent nie uruchomi się. Ścieżka może być względna względem lokalizacji pliku *zabbix_agent2.conf* (np. `Include=.\zabbix_agent2.d\plugins.d\*.conf`).
Aby dołączyć tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=C:\Program Files\Zabbix Agent2\zabbix_agent2.d\*.conf
```

ListenIP

Lista adresów IP rozdzielonych przecinkami, na których agent powinien nasłuchiwać. Pierwszy adres IP jest wysyłany do serwera Zabbix, jeśli nawiązuje z nim połączenie, w celu pobrania listy aktywnych kontroli.

Wartość domyślna: `0.0.0.0`

ListenPort

agent będzie nasłuchiwał na tym porcie połączeń z serwera.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika agenta.

Domyślnie: `c:\zabbix_agent2.log`
 Wymagane: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty na nowo.

Wartość domyślna: 1
 Zakres: 0-1024

LogType

Typ wyjścia logu:
file - zapisuj log do pliku określonego parametrem LogFile;
console - zapisuj log na standardowe wyjście.

Wartość domyślna: `file`

PersistentBufferFile

Plik, w którym agent Zabbix 2 powinien przechowywać bazę danych SQLite. Musi to być pełna nazwa pliku. Ten parametr jest używany tylko wtedy, gdy trwały bufor jest włączony (*EnablePersistentBuffer=1*).

PersistentBufferPeriod

Okres, przez jaki dane powinny być przechowywane, gdy nie ma połączenia z serwerem lub proxy. Starsze dane zostaną utracone. Dane dziennika zostaną zachowane. Ten parametr jest używany tylko wtedy, gdy trwały bufor jest włączony (*EnablePersistent-Buffer=1*).

Wartość domyślna: `1h`
 Zakres: 1m-365d

Plugins.<PluginName>.System.Capacity

Limit sprawdzeń na plugin `<PluginName>`, które mogą być wykonywane jednocześnie.

Domyślnie: 1000 Zakres: 1-1000

Plugins.Log.MaxLinesPerSecond

Maksymalna liczba nowych wierszy na sekundę, które agent wyśle do serwera lub proxy Zabbix podczas przetwarzania aktywnych kontroli „log”, „logrt” i „eventlog”. Podana wartość zostanie nadpisana przez parametr „maxlines” podany w kluczu pozycji „log”, „logrt” lub „eventlog”.
Uwaga: Zabbix przetworzy 10 razy więcej nowych wierszy niż ustawiono w *MaxLinesPerSecond*, aby wyszukać wymagany ciąg znaków w pozycjach logów.

Domyślnie: 20
 Zakres: 1-1000

Plugins.SystemRun.LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń. Polecenia będą rejestrowane tylko wtedy, gdy są wykonywane zdalnie. Wpisy dziennika nie będą tworzone, jeśli system.run[] jest uruchamiane lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

PluginSocket

Ścieżka do nazwanego potoku systemu Windows używanego do komunikacji z ładowalnymi wtyczkami.

Wartość domyślna: `\\.\pipe\agent.plugin.sock`

PluginTimeout

Limit czasu połączeń z ładowalnymi wtyczkami, w sekundach.

Wartość domyślna: `Timeout`
 Zakres: 1-30

PerfCounter

Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
Składnia: <parameter_name>,”<perf_counter_path>”,<period>

Na przykład, jeśli chcesz otrzymać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr ”interrupts” w następujący sposób:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania wartości średniej będą pobierane co sekundę.
Możesz uruchomić ”typeperf -qx”, aby uzyskać listę wszystkich liczników wydajności dostępnych w systemie Windows.

PerfCounterEn

Definiuje nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego przedziału czasu <period> (w sekundach). W porównaniu z PerfCounter ścieżki perfcounter muszą być w języku angielskim. Obsługiwane tylko w systemach **Windows Server 2008/Vista** i nowszych.
Składnia: <parameter_name>,"<perf_counter_path>",<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr „interrupts" w następujący sposób:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania wartości średniej będą pobierane co sekundę.
Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Jak często odświeżana jest lista aktywnych testów, w sekundach. Należy pamiętać, że po nieudanej próbie odświeżenia aktywnych testów kolejna próba odświeżenia zostanie podjęta za 60 sekund.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, wtedy '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' są traktowane jednakowo, a '::/0' zezwoli na dowolny adres IPv4 lub IPv6. '0.0.0.0/0' może zostać użyte do zezwolenia na dowolny adres IPv4. Spacje są dozwolone. Jeśli ten parametr nie jest określony, wyłącza sprawdzenia pasywne, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Adres serwera/proxy Zabbix lub konfiguracja klastra, z którego mają być pobierane aktywne testy. Adres serwera/proxy to adres IP lub nazwa DNS oraz opcjonalny port oddzielony dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwera lub proxy oddzielonych średnikiem. Można podać wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy podawać więcej niż jednego proxy Zabbix z każdego serwera/klastra Zabbix. Jeśli podano proxy Zabbix, nie należy podawać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby równolegle używać kilku niezależnych serwerów Zabbix. Spacje są dozwolone.
Jeśli port nie został określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie został określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne testy są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,\[::1\]:30051,::1,\[12fc::1\]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.).

StatusPort

Jeśli ustawione, agent będzie nasłuchiwał na tym porcie żądań statusu HTTP (http://localhost:<port>/status).

Zakres: 1024-32767

Timeout

Określa, jak długo czekać (w sekundach) na ustanowienie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr określa czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwania na odpowiedź od serwera Zabbix;
- wysyłania żądań do serwera Zabbix, w tym żądań konfiguracji pozycji oraz danych pozycji w active checks;
- pobierania danych logów za pomocą monitorowania logfile lub dziennika zdarzeń Windows;
- wysyłania komunikatów heartbeat;
- maksymalnego czasu trwania kontroli `vfs.*`;
- użycia jako wartości zapasowej w sytuacjach, gdy serwer lub proxy starszy niż wersja 7.0 wysyła kontrole bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych kontroli agenta, które mają konfigurowalne ustawienia limitu czasu we frontendzie (na poziomie globalnym, proxy lub pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Przychodzące połączenia do akceptowania. Używane dla pasywnych kontroli. Można określić wiele wartości, oddzielonych przecinkami:
unencrypted - akceptuj połączenia bez szyfrowania (domyślnie)
psk - akceptuj połączenia z TLS i kluczem współdzielonym (PSK)
cert - akceptuj połączenia z TLS i certyfikatem

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu lub urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll13

Ciąg szyfrów OpenSSL (TLS 1.3).
Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert13

Ciąg szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Należy pamiętać, że tego parametru nie można używać razem z `TLSAccept=cert,psk`; w przypadku połączenia certyfikatowego (`TLSConnect=cert`) należy zamiast tego użyć TLSCipherAll13.

TLSCipherPSK13

Ciąg szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Jak agent powinien łączyć się z serwerem lub proxy Zabbix. Używane dla aktywnych testów. Można określić tylko jedną wartość:
unencrypted - połączenie bez szyfrowania (domyślnie)
psk - połączenie z użyciem TLS i klucza współdzielonego (PSK)
cert - połączenie z użyciem TLS i certyfikatu

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Pełna ścieżka do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny agenta, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Pełna ścieżka do pliku zawierającego klucz współdzielony agenta, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSPSKIdentity

Ciąg tożsamości klucza współdzielonego, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Dozwolony wystawca certyfikatu serwera (proxy).

TLSServerCertSubject

Dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @
Ponadto znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Parametr zdefiniowany przez użytkownika do monitorowania. Może istnieć kilka parametrów zdefiniowanych przez użytkownika.
Format: UserParameter=<key>,<shell command>
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą zawierać ścieżki względne, jeśli określono parametr UserParameterDir.

Przykład:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Domyślna ścieżka wyszukiwania dla poleceń UserParameter. Jeśli jest używana, agent zmieni swój katalog roboczy na określony tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą mieć względny prefiks `.\` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=C:\opt\myscripts
```

7 Pluginy Zabbix agent 2

Przegląd

Sekcja zawiera opisy parametrów zawartych w pliku konfiguracyjnym dla Zabbix agent 2. Aby uzyskać informację o konkretnym pluginie, skorzystaj z paska bocznego.

1 Wtyczka Ceph

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki Ceph Zabbix agent 2 (ceph.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne tylko w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od "#" są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Ceph.Default.ApiKey				Domyślny klucz API do łączenia z Ceph; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.Ceph.Default.User				Domyślna nazwa użytkownika do łączenia z Ceph; używana, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.Ceph.Default.Uri			https://localhost:8003	Domyślny URI do łączenia z Ceph; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `https`; schemat można pominąć. Port można pominąć (domyślnie=8003). Przykłady: `https://127.0.0.1:8003` `localhost`
Plugins.Ceph.InsecureSkipVerify		false / true	false	Określa, czy klient HTTP powinien weryfikować łańcuch certyfikatów serwera i nazwę hosta. Jeśli *true*, TLS akceptuje dowolny certyfikat przedstawiony przez serwer oraz dowolną nazwę hosta w tym certyfikacie. W tym trybie TLS jest podatny na ataki typu man-in-the-middle (powinien być używany wyłącznie do testów).
Plugins.Ceph.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Ceph.Sessions.<SessionName>.ApiKey				Klucz API nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Ceph.Sessions.<SessionName>.User				Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Ceph.Sessions.<SessionName>.Uri				Ciąg połączenia nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `https`; schemat można pominąć. Port można pominąć (domyślnie=8003). Przykłady: `https://127.0.0.1:8003` `localhost`
Plugins.Ceph.Timeout		1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

2 Wtyczka Docker

Omówienie

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracji wtyczki Docker dla agent 2 Zabbix (docker.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;

- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Docker.Endpoint			unix:///var/run/docker.sock	Lokalizacja unix-socket demona Docker. Musi zawierać schemat (obsługiwany jest tylko `unix://`).
Plugins.Docker.Timeout		1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji plugins

3 Wtyczka Ember+

Omówienie

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki Ember+ dla agent 2 Zabbix (ember.conf).

Wtyczka Ember+ jest wtyczką ładowaną dynamicznie i jest dostępna oraz w pełni opisana w repozytorium wtyczki Ember+.

Obecnie tę wtyczkę można zbudować wyłącznie z kodu źródłowego (zarówno dla Unix, jak i Windows).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki oraz informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.EmberPlus.Default.Uri			tcp://localhost:9998	Domyślny URI do połączenia. Jedynym obsługiwanym schematem jest `tcp://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane.
Plugins.EmberPlus.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń pluginu.
Plugins.EmberPlus.Sessions.<SessionName>.Uri			tcp://localhost:9998	URI do połączenia dla nazwanej sesji. Jedynym obsługiwanym schematem jest `tcp://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.EmberPlus.System.Path				Ścieżka do pliku wykonywalnego pluginu Ember+. Przykład użycia: `Plugins.EmberPlus.System.Path=/usr/libexec/zabbix/zabbix-`
Plugins.EmberPlus.Timeout		1-30	global timeout	Czas oczekiwania, w sekundach, na odpowiedź serwera podczas pierwszego nawiązywania połączenia oraz podczas kolejnych operacji w sesji.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

4 Wtyczka Memcached

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracji wtyczki Memcached dla Zabbix agent 2 (memcached.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Memcached.Default.Password				Domyślne hasło do łączenia z Memcached; używane, jeśli w kluczu pozycja lub nazwanej sesji nie określono żadnej wartości.

Plugins.Memcached.Default.Uri			tcp://localhost:11211	Domyślny URI do łączenia z Memcached; używany, jeśli w kluczu pozycja lub nazwanej sesji nie określono żadnej wartości.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=11211).
				Przykłady: `tcp://localhost:11211`
				`localhost`
				`unix:/var/run/memcached.sock`

Plugins.Memcached.Default.User				Domyślna nazwa użytkownika do łączenia z Memcached; używana, jeśli w kluczu pozycja lub nazwanej sesji nie określono żadnej wartości.
Plugins.Memcached.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Memcached.Sessions.<SessionName>.Password				Hasło nazwanej sesji. **<SessionName>** - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Memcached.Sessions.<SessionName>.Uri				Ciąg połączenia nazwanej sesji.
				<SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=11211).
				Przykłady: `tcp://localhost:11211`
				`localhost`
				`unix:/var/run/memcached.sock`

Plugins.Memcached.Sessions.<SessionName>.User				Nazwa użytkownika nazwanej sesji. **<SessionName>** - określa nazwę sesji do użycia w kluczach pozycji.
Plugins.Memcached.Timeout		1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

5 Wtyczka Modbus

Omówienie

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki Modbus dla agent 2 Zabbix (modbus.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Modbus.Sessions.<SessionName>.Endpoint				Endpoint to ciąg połączenia składający się ze schematu protokołu, adresu hosta oraz portu lub nazwy portu szeregowego i atrybutów. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Modbus.Sessions.<SessionName>.SlaveID				Slave ID nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Przykład: `Plugins.Modbus.Sessions.MB1.SlaveID=20` *Uwaga*: ten parametr nazwanej sesji jest sprawdzany tylko wtedy, gdy wartość podana w parametrze slave ID klucza pozycji jest pusta.
Plugins.Modbus.Sessions.<SessionName>.Timeout				Limit czasu nazwanej sesji w sekundach. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Przykład: `Plugins.Modbus.Sessions.MB1.Timeout=2`

Jeśli musisz ustawić limit czasu wykonania żądania (jak długo czekać na zakończenie żądania przed jego przerwaniem), użyj formularza konfiguracji pozycji.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

6 Wtyczka MongoDB

Omówienie

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki MongoDB dla agent 2 Zabbix (mongo.conf).

Wtyczka MongoDB jest wtyczką ładowaną dynamicznie i jest dostępna oraz w pełni opisana w repozytorium wtyczki MongoDB.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki oraz informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.MongoDB.Default.Password				Domyślne hasło do łączenia z MongoDB; używane, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.MongoDB.Default.Uri				Domyślny URI do łączenia z MongoDB; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `tcp`; schemat można pominąć. Port można pominąć (domyślnie=27017). Przykłady: `tcp://127.0.0.1:27017`, `tcp:localhost`, `localhost`
Plugins.MongoDB.Default.User				Domyślna nazwa użytkownika do łączenia z MongoDB; używana, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.MongoDB.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń pluginu.
Plugins.MongoDB.Sessions.<SessionName>.Password				Hasło nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile	(yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSConnect is set to one of: verify_ca, verify_full)			Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używana w szyfrowanej komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile	(yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile is specified			Pełna ścieżka do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używana w szyfrowanej komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MongoDB.Sessions.<SessionName>.TLSConnect				Typ szyfrowania dla komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Obsługiwane wartości: *required* - wymagaj połączenia TLS; *verify_ca* - weryfikuj certyfikaty; *verify_full* - weryfikuj certyfikaty i adres IP. Obsługiwane od wersji pluginu 1.2.1
Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile	(yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile is specified			Pełna ścieżka do pliku zawierającego prywatny klucz bazy danych używany w szyfrowanej komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MongoDB.Sessions.<SessionName>.Uri				Ciąg połączenia nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `tcp`; schemat można pominąć. Port można pominąć (domyślnie=27017). Przykłady: `tcp://127.0.0.1:27017`, `tcp:localhost`, `localhost`

Parameter	Mandatory	Range	Default	Description
Plugins.MongoDB.Sessions.<SessionName>.User	No			Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MongoDB.System.Path	No			Ścieżka do pliku wykonywalnego pluginu MongoDB. Przykład użycia: `Plugins.MongoDB.System.Path=/usr/libexec/zabbix/zabbix-ag`
Plugins.MongoDB.Timeout	No	1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

7 Wtyczka MQTT

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracji wtyczki MQTT dla agent 2 Zabbixa (mqtt.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.MQTT.Default.Password	No			Domyślne hasło do łączenia z MQTT; używane, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
Plugins.MQTT.Default.TLSCAFile	No			Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu peer dla szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT; używane, jeśli nie określono żadnej wartości w nazwanej sesji.
Plugins.MQTT.Default.TLSCertFile	No			Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT; używane, jeśli nie określono żadnej wartości w nazwanej sesji.
Plugins.MQTT.Default.TLSKeyFile	No			Pełna ścieżka do pliku zawierającego prywatny klucz MQTT do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT; używane, jeśli nie określono żadnej wartości w nazwanej sesji.

Plugins.MQTT.Default.Topic	No			Domyślny temat subskrypcji MQTT; używany, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
				
				Temat może zawierać symbole wieloznaczne (”+”,”#”)
				Przykłady: `path/to/file`
				`path/to/#`
				`path/+/topic`

Parameter	Mandatory	Range	Default	Description

Plugins.MQTT.Default.Url	no		tcp://localhost:1883	Domyślny ciąg połączenia z brokerem MQTT; używany, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
				
				Nie powinien zawierać parametrów zapytania.
				Musi odpowiadać formatowi URL.
				Obsługiwane schematy: `tcp` (domyślny), `ws`, `tls`; schemat można pominąć.
				Port można pominąć (domyślnie=1883).
				Przykłady: `tcp://host:1883`
				`localhost`
				`ws://host:8080`

Plugins.MQTT.Default.User	no			Domyślna nazwa użytkownika do łączenia z MQTT; używana, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
Plugins.MQTT.Sessions.<SessionName>.Password	no			Hasło nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MQTT.Sessions.<SessionName>.TLSCAFile	no			Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu peer, używana do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MQTT.Sessions.<SessionName>.TLSCertFile	no			Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MQTT.Sessions.<SessionName>.TLSKeyFile	no			Pełna ścieżka do pliku zawierającego prywatny klucz MQTT, używana do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Plugins.MQTT.Sessions.<SessionName>.Topic	no			Temat nazwanej sesji dla subskrypcji MQTT.
				<SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
				
				Temat może zawierać symbole wieloznaczne (”+”,”#”)
				Przykłady: `path/to/file`
				`path/to/#`
				`path/+/topic`

Plugins.MQTT.Sessions.<SessionName>.Url	no			Ciąg połączenia nazwanej sesji.
				<SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
				
				Nie powinien zawierać parametrów zapytania.
				Musi odpowiadać formatowi URL.
				Obsługiwane schematy: `tcp` (domyślny), `ws`, `tls`; schemat można pominąć.
				Port można pominąć (domyślnie=1883).
				Przykłady: `tcp://host:1883`
				`localhost`
				`ws://host:8080`

Plugins.MQTT.Sessions.<SessionName>.User	no			Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Jeśli trzeba ustawić limit czasu wykonania żądania (jak długo czekać na zakończenie żądania przed jego przerwaniem), użyj formularza konfiguracji pozycji.

Zobacz też:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

8 Wtyczka MSSQL

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki MSSQL dla agent Zabbix 2 (mssql.conf).

Wtyczka MSSQL jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki MSSQL.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki i informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.MSSQL.CustomQueriesDir			`/usr/local/share/zabbix/mssql.queries` dla systemów Unix `*:\Program Files\Zabbix Agent 2\Custom Queries\MSSQL` dla systemów Windows, gdzie * jest nazwą dysku z zmiennej środowiskowej `ProgramFiles`	Ścieżka do katalogu zawierającego zdefiniowane przez użytkownika pliki .sql z niestandardowymi zapytaniami, które plugin może wykonywać. Plugin wczytuje wszystkie dostępne pliki .sql w skonfigurowanym katalogu podczas uruchamiania. Oznacza to, że wszelkie zmiany w plikach niestandardowych zapytań nie zostaną uwzględnione do czasu ponownego uruchomienia pluginu. Plugin jest uruchamiany i zatrzymywany razem z Zabbix agent 2.
Plugins.MSSQL.CustomQueriesEnabled			false	Jeśli ustawione, włącza wykonywanie klucza pozycji `mssql.custom.query`. Jeśli wyłączone, żadne zapytania nie będą wczytywane z katalogu niestandardowych zapytań.
Plugins.MSSQL.Default.CACertPath				Domyślna ścieżka do pliku certyfikatu klucza publicznego urzędu certyfikacji (CA), który wydał certyfikat serwera MSSQL. Certyfikat musi być w formacie PEM.
Plugins.MSSQL.Default.Database				Domyślna nazwa bazy danych, z którą należy się połączyć.

Parameter	Mandatory	Range	Default	Description
Plugins.MSSQL.Default.Encrypt				Określa domyślny typ szyfrowania połączenia. Dostępne wartości: *true* - przesyłanie danych między pluginem a serwerem jest szyfrowane; *false* - przesyłanie danych między pluginem a serwerem nie jest szyfrowane poza pakietem logowania; *strict* - przesyłanie danych między pluginem a serwerem jest szyfrowane E2E przy użyciu TDS8; *disable* - przesyłanie danych między pluginem a serwerem nie jest szyfrowane.
Plugins.MSSQL.Default.HostNameInCertificate				Domyślna nazwa pospolita (CN) certyfikatu serwera MSSQL.
Plugins.MSSQL.Default.Password				Domyślne hasło przekazywane do chronionego serwera MSSQL.
Plugins.MSSQL.Default.TLSMinVersion				Domyślna minimalna wersja TLS do użycia. Dostępne wartości: `1.0`, `1.1`, `1.2`, `1.3`.
Plugins.MSSQL.Default.TrustServerCertificate				Określa, czy plugin ma domyślnie ufać certyfikatowi serwera bez jego weryfikacji. Dostępne wartości: `true`, `false`.
Plugins.MSSQL.Default.Uri			sqlserver://localhost:1433	Domyślny URI do połączenia. Jedynym obsługiwanym schematem jest `sqlserver://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane.
Plugins.MSSQL.Default.User				Domyślna nazwa użytkownika przekazywana do chronionego serwera MSSQL.
Plugins.MSSQL.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń pluginu.
Plugins.MSSQL.Sessions.<SessionName>.CACertPath				Ścieżka do pliku certyfikatu klucza publicznego urzędu certyfikacji (CA), który wydał certyfikat serwera MSSQL dla nazwanej sesji. Certyfikat musi być w formacie PEM. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.Database				Nazwa bazy danych, z którą należy się połączyć dla nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.Encrypt				Określa typ szyfrowania połączenia dla nazwanej sesji. Dostępne wartości: *true* - przesyłanie danych między pluginem a serwerem jest szyfrowane; *false* - przesyłanie danych między pluginem a serwerem nie jest szyfrowane poza pakietem logowania; *strict* - przesyłanie danych między pluginem a serwerem jest szyfrowane E2E przy użyciu TDS8; *disable* - przesyłanie danych między pluginem a serwerem nie jest szyfrowane. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate				Domyślna nazwa pospolita (CN) certyfikatu serwera MSSQL dla nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.Password				Hasło przekazywane do chronionego serwera MSSQL dla nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion				Minimalna wersja TLS do użycia dla nazwanej sesji. Dostępne wartości: `1.0`, `1.1`, `1.2`, `1.3`. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate				Określa, czy plugin ma ufać certyfikatowi serwera bez jego weryfikacji dla nazwanej sesji. Dostępne wartości: `true`, `false`. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Parameter	Mandatory	Range	Default	Description
Plugins.MSSQL.Sessions.<SessionName>.Uri	no		sqlserver://localhost:1433	URI połączenia dla nazwanej sesji. Jedynym obsługiwanym schematem jest `sqlserver://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.Sessions.<SessionName>.User	no			Nazwa użytkownika przekazywana do chronionego serwera MSSQL dla nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.MSSQL.System.Path	no			Ścieżka do pliku wykonywalnego pluginu MSSQL. Ustawienie globalne dla pluginu MSSQL. Stosowane do wszystkich połączeń. Przykład użycia: `Plugins.MSSQL.System.Path=/usr/libexec/zabbix/zabbix-agen`
Plugins.MSSQL.Timeout	no	1-30	global timeout	Czas oczekiwania, w sekundach, na odpowiedź serwera przy pierwszym nawiązywaniu połączenia oraz podczas kolejnych operacji w sesji.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

9 Wtyczka MySQL

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracji wtyczki MySQL dla agent 2 Zabbix (mysql.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Mysql.CallTimeout	no	1-30	global timeout	Maksymalny czas w sekundach oczekiwania na wykonanie żądania.
Plugins.Mysql.CustomQueriesPath	no		`/usr/local/sbin/zabbix/custom-queries/mysql` dla systemów Unix `*:\Program Files\Zabbix Agent 2\Custom Queries\Mysql` dla systemów Windows, gdzie `*` to nazwa dysku z zmiennej środowiskowej `ProgramFiles`	Pełna ścieżka do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.
Plugins.Mysql.CustomQueriesEnabled	no		false	Jeśli ustawione, włącza wykonywanie klucza pozycji `mysql.custom.query`. Jeśli wyłączone, żadne zapytania nie będą ładowane z katalogu niestandardowych zapytań.

Parameter	Mandatory	Range	Default	Description
Plugins.Mysql.Default.Password	no			Domyślne hasło do łączenia z MySQL; używane, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
Plugins.Mysql.Default.TLSCAFile	no (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu partnera dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych; używane, jeśli nie określono żadnej wartości w nazwanej sesji.
Plugins.Mysql.Default.TLSCertFile	no (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych; używane, jeśli nie określono żadnej wartości w nazwanej sesji.
Plugins.Mysql.Default.TLSConnect	no			Typ szyfrowania dla komunikacji między Zabbix agent 2 a monitorowanymi bazami danych; używane, jeśli nie określono żadnej wartości w nazwanej sesji. Obsługiwane wartości: *required* - wymagaj połączenia TLS; *verify_ca* - weryfikuj certyfikaty; *verify_full* - weryfikuj certyfikaty i adres IP.
Plugins.Mysql.Default.TLSKeyFile	no (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego prywatny klucz bazy danych dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych; używane, jeśli nie określono żadnej wartości w nazwanej sesji.

Plugins.Mysql.Default.Uri	no		tcp://localhost:3306	Domyślny URI do łączenia z MySQL; używane, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=3306).
				Przykłady: `tcp://localhost:3306`
				`localhost`
				`unix:/var/run/mysql.sock`

Plugins.Mysql.Default.User	no			Domyślna nazwa użytkownika do łączenia z MySQL; używana, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.
Plugins.Mysql.KeepAlive	no	60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Mysql.Sessions.<SessionName>.Password	no			Hasło nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Mysql.Sessions.<SessionName>.TLSCAFile	no (yes, if Plugins.Mysql.Sessions.<SessionName>.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu partnera, używana dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	yes, if Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile is specified			Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używana dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Parameter	Mandatory	Range	Default	Description
Plugins.Mysql.Sessions.<SessionName>.TLSConnect				Typ szyfrowania dla komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Obsługiwane wartości: *required* - wymagaj połączenia TLS; *verify_ca* - weryfikuj certyfikaty; *verify_full* - weryfikuj certyfikaty i adres IP.
Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	yes, if Plugins.Mysql.Sessions.<SessionName>.TLSCertFile is specified			Pełna ścieżka do pliku zawierającego prywatny klucz bazy danych używany dla szyfrowanych komunikacji między Zabbix agent 2 a monitorowanymi bazami danych. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Plugins.Mysql.Sessions.<SessionName>.Uri				Ciąg połączenia nazwanej sesji.
				<SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=3306).
				Przykłady: `tcp://localhost:3306`
				`localhost`
				`unix:/var/run/mysql.sock`

Plugins.Mysql.Sessions.<SessionName>.User				Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Mysql.Timeout		1-30	global timeout	Maksymalny czas w sekundach oczekiwania na ustanowienie połączenia.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

10 Wtyczka NVIDIA GPU

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki NVIDIA GPU dla agent 2 (nvidia.conf).

Wtyczka NVIDIA GPU jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki NVIDIA GPU.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.NVIDIA.System.Path				Ścieżka do pliku wykonywalnego wtyczki NVIDIA GPU. Przykład użycia: `Plugins.NVIDIA.System.Path=/path/to/executable/nvidia`

Parameter	Mandatory	Range	Default	Description
Plugins.NVIDIA.Timeout		1-30	global timeout	Określa maksymalny czas (w sekundach) oczekiwania na odpowiedź serwera podczas prób połączenia oraz kolejnych operacji w sesji. Globalny limit czasu dla pozycji lub indywidualny limit czasu pozycji zastąpi tę wartość, jeśli będzie większy.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

11 Wtyczka Oracle

Omówienie

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracji wtyczki Oracle dla agent 2 Zabbix (oracle.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Oracle.CallTimeout		1-30	global timeout	Maksymalny czas oczekiwania w sekundach na zakończenie żądania.
Plugins.Oracle.ConnectTimeout		1-30	global timeout	Maksymalny czas oczekiwania w sekundach na nawiązanie połączenia.
Plugins.Oracle.CustomQueriesPath			`/usr/local/share/zabbix/oracle/custom_queries` dla systemów Unix `*:\Program Files\Zabbix Agent 2\Custom Queries\Oracle` dla systemów Windows, gdzie * to nazwa dysku z zmiennej środowiskowej `ProgramFiles`	Pełna ścieżka do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami. Przykład: `/etc/zabbix/oracle/sql`
Plugins.Oracle.CustomQueriesEnabled			false	Jeśli ustawione, włącza wykonywanie klucza pozycji `oracle.custom.query`. Jeśli wyłączone, żadne zapytania nie będą ładowane z katalogu niestandardowych zapytań.
Plugins.Oracle.Default.Password				Domyślne hasło do łączenia z Oracle; używane, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.
Plugins.Oracle.Default.Service				Domyślna nazwa usługi do łączenia z Oracle (SID nie jest obsługiwany); używana, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Parameter	Mandatory	Range	Default	Description
Plugins.Oracle.Default.Uri			tcp://localhost:1521	Domyślny URI do łączenia z Oracle; używany, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `tcp`; schemat można pominąć. Port można pominąć (domyślnie=1521). Od **Zabbix 7.4.3** można również określić klucz TNS lub wartość TNS jako ciąg połączenia. Wartość TNS musi być zapisana bez spacji. Przykłady: `tcp://127.0.0.1:1521` `localhost` `zbx_tns_example` (klucz TNS) `(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(POR` (wartość TNS)
Plugins.Oracle.Default.User				Domyślna nazwa użytkownika do łączenia z Oracle; używana, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.
Plugins.Oracle.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Oracle.ResolveTNS		true / false	false	Opcja określa, jak interpretować ciąg połączenia (ConnString) w kluczu metryki `commonParams` podczas łączenia z serwerem Oracle. Jeśli ma wartość false, do ConnString zostaną dodane domyślny schemat i port (jeśli ich brakuje). Jeśli opcja ma wartość true, domyślny schemat i port zostaną pominięte (chyba że zostały jawnie określone w ConnString), a ConnString zostanie przekazany do klienta Oracle bez zmian. Jeśli klient Oracle znajdzie ConnString w pliku tnsnames.ora, do połączenia z serwerem Oracle zostanie użyty znaleziony opis połączenia. Ten parametr jest obsługiwany od Zabbix 7.4.3.
Plugins.Oracle.Sessions.<SessionName>.Password				Hasło nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Oracle.Sessions.<SessionName>.Service				Nazwa usługi nazwanej sesji używana do połączenia (SID nie jest obsługiwany). **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.Oracle.Sessions.<SessionName>.Uri				Ciąg połączenia dla nazwanej sesji Oracle. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane). Musi odpowiadać formatowi URI. Obsługiwany jest tylko schemat `tcp`; schemat można pominąć. Port można pominąć (domyślnie=1521). Od **Zabbix 7.4.3** można również określić klucz TNS lub wartość TNS jako ciąg połączenia. Wartość TNS musi być zapisana bez spacji. Przykłady: `tcp://127.0.0.1:1521` `localhost` `zbx_tns_example` (klucz TNS) `(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(POR` (wartość TNS)

Parameter	Mandatory	Range	Default	Description
Plugins.Oracle.Sessions.<SessionName>.User				Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

12 Wtyczka PostgreSQL

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki PostgreSQL dla agent 2 (postgresql.conf).

Wtyczka PostgreSQL jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki PostgreSQL.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki oraz informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.PostgreSQL.Default.CacheMode			prepare	Tryb pamięci podręcznej dla połączenia PostgreSQL. Obsługiwane wartości: *prepare* (domyślnie) - utworzy przygotowane instrukcje na serwerze PostgreSQL; *describe* - użyje anonimowej przygotowanej instrukcji do opisania instrukcji bez tworzenia jej na serwerze. Należy pamiętać, że ”describe” jest przydatne głównie wtedy, gdy środowisko nie pozwala na przygotowane instrukcje, na przykład podczas korzystania z poolera połączeń, takiego jak PgBouncer.
Plugins.PostgreSQL.CallTimeout		1-30	global timeout	Maksymalny czas oczekiwania (w sekundach) na zakończenie żądania.

Parameter	Mandatory	Range	Default	Description
Plugins.PostgreSQL.CustomQueriesPath			`/usr/local/share/zabbix/custom-queries/postgresql` dla systemów Unix `*:\Program Files\Zabbix Agent 2\Custom Queries\PostgreSQL` dla systemów Windows, gdzie `*` jest nazwą dysku z zmiennej środowiskowej `ProgramFiles`	Pełna ścieżka do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.
Plugins.PostgreSQL.CustomQueriesEnabled			false	Jeśli ustawione, włącza wykonywanie klucza pozycja `postgresql.custom.query`. Jeśli wyłączone, żadne zapytania nie będą ładowane z katalogu niestandardowych zapytań.
Plugins.PostgreSQL.Default.Database				Domyślna baza danych do łączenia z PostgreSQL; używana, jeśli w kluczu pozycji lub nazwanej sesji nie określono wartości.
Plugins.PostgreSQL.Default.Password				Domyślne hasło do łączenia z PostgreSQL; używane, jeśli w kluczu pozycji lub nazwanej sesji nie określono wartości.
Plugins.PostgreSQL.Default.TLSCAFile	(yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer dla szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych; używana, jeśli w nazwanej sesji nie określono wartości.
Plugins.PostgreSQL.Default.TLSCertFile	(yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych; używana, jeśli w nazwanej sesji nie określono wartości.
Plugins.PostgreSQL.Default.TLSConnect				Typ szyfrowania komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych; używany, jeśli w nazwanej sesji nie określono wartości. Obsługiwane wartości: *required* - połączenie z użyciem TLS jako trybu transportu bez sprawdzania tożsamości; *verify_ca* - połączenie z użyciem TLS i weryfikacją certyfikatu; *verify_full* - połączenie z użyciem TLS, weryfikacją certyfikatu oraz sprawdzeniem, czy tożsamość bazy danych (CN) określona przez DBHost jest zgodna z jej certyfikatem. Nieokreślony typ szyfrowania oznacza połączenie nieszyfrowane.
Plugins.PostgreSQL.Default.TLSKeyFile	(yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego klucz prywatny PostgreSQL do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych; używana, jeśli w nazwanej sesji nie określono wartości.

Parameter	Mandatory	Range	Default	Description

Plugins.PostgreSQL.Default.Uri				Domyślny URI do łączenia z PostgreSQL; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono wartości.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi być zgodny z formatem URI.
				Obsługiwane schematy: `tcp`, `unix`.
				Przykłady: `tcp://127.0.0.1:5432`
				`tcp://localhost`
				`unix:/var/run/postgresql/.s.PGSQL.5432`

Plugins.PostgreSQL.Default.User				Domyślna nazwa użytkownika do łączenia z PostgreSQL; używana, jeśli w kluczu pozycji lub nazwanej sesji nie określono wartości.
Plugins.PostgreSQL.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.PostgreSQL.Sessions.<SessionName>.CachePrepareMode			prepare	Tryb pamięci podręcznej dla połączenia PostgreSQL. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Obsługiwane wartości: *prepare* (domyślnie) - utworzy przygotowane instrukcje na serwerze PostgreSQL; *describe* - użyje anonimowej przygotowanej instrukcji do opisania instrukcji bez tworzenia jej na serwerze. Należy pamiętać, że ”describe” jest przydatne głównie wtedy, gdy środowisko nie pozwala na przygotowane instrukcje, na przykład podczas korzystania z poolera połączeń, takiego jak PgBouncer.
Plugins.PostgreSQL.Sessions.<SessionName>.Database				Baza danych dla połączenia sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.PostgreSQL.Sessions.<SessionName>.Password		Must match the password format.		Hasło dla połączenia sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile	(yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect is set to *verify_ca* or *verify_full*)			Pełna ścieżka do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile	(yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile is specified			Pełna ścieżka do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect				Typ szyfrowania dla połączenia PostgreSQL. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji. Obsługiwane wartości: *required* - połączenie z użyciem TLS jako trybu transportu bez sprawdzania tożsamości; *verify_ca* - połączenie z użyciem TLS i weryfikacją certyfikatu; *verify_full* - połączenie z użyciem TLS, weryfikacją certyfikatu oraz sprawdzeniem, czy tożsamość bazy danych (CN) określona przez DBHost jest zgodna z jej certyfikatem. Nieokreślony typ szyfrowania oznacza połączenie nieszyfrowane.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile	(yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile is specified			Pełna ścieżka do pliku zawierającego klucz prywatny PostgreSQL. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Parameter	Mandatory	Range	Default	Description

Plugins.PostgreSQL.Sessions.<SessionName>.Uri				Ciąg połączenia dla nazwanej sesji.
				<SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi być zgodny z formatem URI.
				Obsługiwane schematy: `tcp`, `unix`.
				Przykłady: `tcp://127.0.0.1:5432`
				`tcp://localhost`
				`unix:/var/run/postgresql/.s.PGSQL.5432`

Plugins.PostgreSQL.Sessions.<SessionName>.User				Nazwa użytkownika nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
Plugins.PostgreSQL.System.Path				Ścieżka do pliku wykonywalnego wtyczki PostgreSQL. Przykład użycia: `Plugins.PostgreSQL.System.Path=/usr/libexec/zabbix/zabbi`
Plugins.PostgreSQL.Timeout		1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

13 Wtyczka Redis

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki Redis dla agent 2 (redis.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Redis.Default.Password				Domyślne hasło do łączenia z Redis; używane, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.Redis.Default.TLSConnect				Wymaganie weryfikacji TLS po stronie klienta (Zabbix agent 2). Obsługiwane wartości: `required` - szyfrowane, nieweryfikowane (tylko do testów) `verify_ca` - szyfrowane, certyfikat serwera `verify_full` - szyfrowane, certyfikat serwera i nazwa serwera zweryfikowana za pomocą SAN Pusta lub nieustawiona wartość jest traktowana jako brak tls.
Plugins.Redis.Default.TLSCAFile				Domyślna pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA.
Plugins.Redis.Default.TLSCertFile				Domyślna pełna ścieżka do pliku zawierającego certyfikat agenta.
Plugins.Redis.Default.TLSKeyFile				Domyślna pełna ścieżka do pliku zawierającego prywatny klucz agenta.

Parameter	Mandatory	Range	Default	Description

Plugins.Redis.Default.Uri			tcp://localhost:6379	Domyślny URI do łączenia z Redis; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=6379).
				Przykłady: `tcp://localhost:6379`
				`localhost`
				`unix:/var/run/redis.sock`

Plugins.Redis.Default.User			default	Domyślny użytkownik przekazywany do chronionego serwera Redis; używany, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.
Plugins.Redis.KeepAlive		60-900	300	Maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Redis.Sessions.<SessionName>.Password				Hasło nazwanej sesji. **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.
Plugins.Redis.Sessions.<SessionName>.TLSConnect				Wymaganie weryfikacji TLS po stronie klienta (Zabbix agent 2). **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji. Obsługiwane wartości: `required` - szyfrowane, nieweryfikowane (tylko do testów) `verify_ca` - szyfrowane, certyfikat serwera `verify_full` - szyfrowane, certyfikat serwera i nazwa serwera zweryfikowana za pomocą SAN Pusta lub nieustawiona wartość jest traktowana jako brak tls.
Plugins.Redis.Sessions.<SessionName>.TLSCAFile	(yes, if TLSConnect is `verify_ca` or `verify_full`)			Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu serwera Redis. **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.
Plugins.Redis.Sessions.<SessionName>.TLSCertFile				Pełna ścieżka do pliku zawierającego certyfikat agenta do uwierzytelniania klienta. **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.
Plugins.Redis.Sessions.<SessionName>.TLSKeyFile				Pełna ścieżka do pliku zawierającego prywatny klucz agenta do uwierzytelniania klienta. **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.

Plugins.Redis.Sessions.<SessionName>.Uri			localhost:6379	Ciąg połączenia nazwanej sesji.
				<SessionName> - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.
				
				Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
				Musi odpowiadać formatowi URI.
				Obsługiwane schematy: `tcp`, `unix`; schemat może zostać pominięty.
				Port może zostać pominięty (domyślnie=6379).
				Przykłady: `tcp://localhost:6379`
				`localhost`
				`unix:/var/run/redis.sock`

Plugins.Redis.Sessions.<SessionName>.User			default	Użytkownik przekazywany do chronionego serwera Redis. **<SessionName>** - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.

Parameter	Mandatory	Range	Default	Description
Plugins.Redis.Timeout	no	1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach oczekiwania na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

14 Wtyczka SMART

Przegląd

Ta sekcja zawiera listę parametrów obsługiwanych w pliku konfiguracyjnym wtyczki SMART dla agent Zabbix 2 (smart.conf).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Ścieżka do pliku wykonywalnego `smartctl` musi być ustawiona albo przez dodanie jego katalogu do systemowej zmiennej środowiskowej `PATH`, albo przez skonfigurowanie `Plugins.Smart.Path`; dotyczy to zarówno systemu Linux, jak i Windows;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Parametry

Parameter	Mandatory	Range	Default	Description
Plugins.Smart.Path	no		smartctl	Ścieżka do pliku wykonywalnego smartctl.
Plugins.Smart.Timeout	no	1-30	global timeout	Limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Zobacz także:

- Opis ogólnych parametrów konfiguracji agent 2 Zabbix: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji pluginów

8 Zabbix Java gateway

Jeśli do uruchamiania Zabbix Java gateway używasz skryptów `startup.sh` i `shutdown.sh`, możesz określić wymagane parametry konfiguracji w pliku `settings.sh`. Skrypty uruchamiania i zatrzymywania wczytują plik ustawień i odpowiadają za konwersję zmiennych powłoki (wymienionych w pierwszej kolumnie) na właściwości Java (wymienione w drugiej kolumnie).

Jeśli uruchamiasz Zabbix Java gateway ręcznie, wykonując bezpośrednio polecenie `java`, wówczas odpowiednie właściwości Java określasz w wierszu poleceń.

Variable	Property	Mandatory	Range	Default	Description
LISTEN_IP	zabbix.listenIP	no		0.0.0.0	Adres IP, na którym nasłuchuje usługa.
LISTEN_PORT	zabbix.listenPort	no	1024-32767	10052	Port, na którym usługa nasłuchuje.
PID_FILE	zabbix.pidFile	no		/tmp/zabbix_java.pid	Nazwa pliku PID. Jeśli zostanie pominięty, Zabbix Java Gateway zostanie uruchomiony jako aplikacja konsolowa.

Variable	Property	Mandatory	Range	Default	Description
PROPERTIES_FILE	zabbix.propertiesFile	no			Nazwa pliku właściwości. Może być używany do ustawiania dodatkowych właściwości w formacie klucz-wartość, tak aby nie były widoczne w wierszu poleceń, lub do nadpisywania istniejących. Na przykład: ”javax.net.ssl.trustStorePassword=<p
START_POLLERS	zabbix.startPollers	no	1-1000	5	Liczba wątków roboczych do uruchomienia.
TIMEOUT	zabbix.timeout	no	1-30	3	Jak długo czekać na operacje sieciowe (w sekundach).

Warning:
Port 10052 nie jest zarejestrowany przez IANA registered.

9 Usługa internetowa Zabbix

Przegląd

Usługa internetowa Zabbix to proces używany do komunikacji z zewnętrznymi usługami internetowymi.

Parametry obsługiwane przez plik konfiguracyjny usługi internetowej Zabbix (zabbix_web_service.conf) są wymienione w tej sekcji.

Parametry są podane bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
AllowedIP	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix.
DebugLevel	Poziom debugowania.
IgnoreURLCertErrors	Określa obsługę błędów walidacji certyfikatu TLS podczas uzyskiwania dostępu do adresu URL frontend.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki z katalogu do pliku konfiguracyjnego.
ListenPort	Usługa będzie nasłuchiwać na tym porcie połączeń z serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogType	Typ wyjścia dziennika.
Timeout	Maksymalny czas (w sekundach) poświęcony na formatowanie raportu PDF pulpitu.
TLSAccept	Jakie połączenia przychodzące mają być akceptowane.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatu peera, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat usługi lub łańcuch certyfikatów, używana do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny usługi, używana do szyfrowanej komunikacji między komponentami Zabbix.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Zwróć uwagę, że:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości w dostarczanych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne tylko w kodowaniu UTF-8 bez BOM;

- Komentarze rozpoczynające się od "#" są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

AllowedIP

Lista adresów IP oddzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix. Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj.
Jeśli obsługa IPv6 jest włączona, wtedy `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` pozwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` może być użyte do zezwolenia na dowolny adres IPv4.

Przykład:

```
127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak

DebugLevel

Określ poziom debugowania:
0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
1 - informacje krytyczne;
2 - informacje o błędach;
3 - ostrzeżenia;
4 - do debugowania (generuje dużo informacji);
5 - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

IgnoreURLCertErrors

Określa obsługę błędów walidacji certyfikatu TLS podczas uzyskiwania dostępu do adresu URL frontendu:
0 - nie ignoruj błędów certyfikatu;
1 - ignoruj błędy certyfikatu.

Wartość domyślna: 0
 Zakres: 0-1

Include

Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu w pliku konfiguracyjnym. Podczas instalacji Zabbix utworzy katalog include w /usr/local/etc, chyba że zostanie to zmienione podczas kompilacji. Ścieżka może być względna względem lokalizacji pliku *zabbix_web_service.conf*.
Aby dołączyć tylko odpowiednie pliki w określonym katalogu, obsługiwany jest znak wieloznaczny gwiazdki do dopasowywania wzorców.
Zobacz uwagi specjalne dotyczące ograniczeń.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenPort

Usługa będzie nasłuchiwać na tym porcie połączeń z serwera.

Domyślnie: 10053
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika.

Przykład:

```
/tmp/zabbix_web_service.log
```

Wymagane: Tak, jeśli LogType jest ustawione na *file*; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację logów.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i rozpoczęty na nowo.

Wartość domyślna: 1
 Zakres: 0-1024

LogType

Typ wyjścia logu:
file - zapisuj log do pliku określonego przez parametr LogFile;
system - zapisuj log do syslog;
console - zapisuj log na standardowe wyjście.

Domyślnie: `file`

Timeout

Maksymalny czas (w sekundach) poświęcony na formatowanie raportu PDF pulpitu.

Domyślnie: 10
 Zakres: 1-30

TLSAccept

Jakie połączenia przychodzące akceptować:
unencrypted - akceptuj połączenia bez szyfrowania (domyślnie)
cert - akceptuj połączenia z TLS i certyfikatem

Wartość domyślna: `unencrypted`

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnych urzędów certyfikacji (CA) do weryfikacji certyfikatu strony komunikacji, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat usługi lub łańcuch certyfikatów, używanego do szyfrowanej komunikacji z komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego prywatny klucz usługi, używany do szyfrowanej komunikacji między komponentami Zabbix.

10 Zmienne środowiskowe

Przegląd

Zmienne środowiskowe umożliwiają konfigurowanie komponentów Zabbix bez wpisywania na stałe wartości w plikach konfiguracyjnych. Ułatwia to zarządzanie konfiguracją w dynamicznych środowiskach, takich jak Docker, gdzie zmienne mogą być przekazywane w czasie uruchamiania, aby dostosować ustawienia do różnych konfiguracji.

W najprostszym przypadku, ustawiając wartość parametru konfiguracyjnego DebugLevel serwera Zabbix na zmienną środowiskową, można następnie użyć jej do skonfigurowania serwera podczas uruchamiania:

```
#### Plik konfiguracyjny serwera Zabbix:
DebugLevel=${NEW_DEBUG_LEVEL}

#### Uruchamianie serwera Zabbix:
NEW_DEBUG_LEVEL=5 /usr/sbin/zabbix_server
```

Zmienne środowiskowe są obsługiwane przez następujące komponenty Zabbix:

- Serwer
- Proxy
- Agent (UNIX lub Windows)
- Agent 2 (UNIX lub Windows), w tym wtyczki
- Usługa web
- Zabbix sender (przy użyciu opcji -c, --config)

Ważne uwagi

- Gdy parametr konfiguracyjny jest ustawiony na zmienną środowiskową, która nie została określona podczas uruchamiania komponentu, użyta zostanie wartość domyślna parametru.
- W przypadku używania poleceń runtime (np. w celu zwiększenia poziomu logowania agent), należy określić wszystkie wcześniej używane zmienne środowiskowe. Wynika to z faktu, że komponenty Zabbix używają swojego pliku konfiguracyjnego do wykonywania poleceń runtime; jeśli zmienne środowiskowe zostaną pominięte, użyte zostaną domyślne wartości parametrów konfiguracyjnych. Zobacz Przykłady.
- Polecenie `userparameter_reload` runtime nie obsługuje ponownego wczytywania zmiennych środowiskowych. Podczas przeładowania zmienne są ignorowane i ponownie wczytywane są tylko parametry ze zwykłymi wartościami.
- Bieżące zmienne środowiskowe procesu, które były używane w plikach konfiguracyjnych, są czyszczone po uruchomieniu komponentu Zabbix. Zapewnia to, że procesy potomne (np. zdalne skrypty wykonywane przez Zabbix) nie mogą uzyskać dostępu do tych zmiennych. Należy jednak pamiętać, że początkowe zmienne procesu nadal mogą zostać odczytane (np. za pośrednictwem pliku `/proc/<PID>/environ`).

Składnia

Zmienne środowiskowe muszą używać następującej składni: `${alphanumerics/underscores}`.

Nazwa zmiennej może zawierać wyłącznie litery (a-z, A-Z), podkreślenia (_), oraz cyfry (0-9) i nie może zaczynać się od cyfry.

Zmienne, które nie są zgodne z wymaganą składnią lub są połączone ze zwykłą wartością, będą traktowane jako zwykłe wartości, co może powodować błędy.

Poprawna składnia zmiennych:

```
DebugLevel=${NEW_DEBUG_LEVEL}
Hostname=${ZBX_HOSTNAME}
LogFile=${LogFile_001}
```

Niepoprawna składnia zmiennych:

```
DebugLevel=${5_DebugLevel}
Hostname=${ZBX.HOSTNAME 1}
LogFile=/${HOME}/zabbix/zabbix_server.log
```

Note:
W systemie Windows nazwy zmiennych środowiskowych nie rozróżniają wielkości liter.

Przykłady

Poniższe przykłady pokazują, jak skonfigurować i używać zmiennych środowiskowych z komponentami Zabbix.

Przykład 1: Konfigurowanie i testowanie agenta Zabbix

1. Ustaw zmienne środowiskowe w pliku konfiguracyjnym agenta:

```
Hostname=${ZBX_HOSTNAME}
ServerActive=${ServerActive}
```

2. Przetestuj plik konfiguracyjny:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

3. Uruchom agenta ze zmiennymi środowiskowymi:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Podczas używania poleceń runtime (na przykład w celu zwiększenia poziomu logowania agenta) należy podać wszystkie wcześniej użyte zmienne środowiskowe:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Wynika to z tego, że agent używa swojego pliku konfiguracyjnego do wykonywania poleceń runtime; jeśli zmienne środowiskowe zostaną pominięte, zostaną użyte domyślne wartości parametrów konfiguracyjnych.

Alternatywnie, po ustawieniu zmiennych środowiskowych w pliku konfiguracyjnym agenta, możesz udostępnić je procesom (na przykład za pomocą polecenia `export`). Zmniejsza to ryzyko nieoczekiwanego zachowania spowodowanego brakiem zmiennych lub ich nieprawidłowym ustawieniem.

```
export ZBX_HOSTNAME="New Zabbix agent"
export ServerActive=127.0.0.1
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf --test-config
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf
/usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf -R log_level_increase
```

Przykład 2: Konfigurowanie agenta Zabbix dla kontenera

Jeśli tworzysz i konfigurujesz własny niestandardowy obraz dla komponentów Zabbix (np. agenta Zabbix), możesz zdefiniować parametry konfiguracyjne za pomocą zmiennych środowiskowych, a następnie uruchomić kontener z tymi zmiennymi.

1. Podczas przygotowywania obrazu ustaw zmienne środowiskowe w pliku konfiguracyjnym agenta:

```
Hostname=${ZBX_HOSTNAME}
BufferSize=${BUFSZ}
ListenPort=${LISTENPORT}
UserParameter=${_UsrPar01}
UserParameter=${_UsrPar02}
```

2. Po zbudowaniu obrazu kontenera uruchom kontener agenta (np. Docker) ze zmiennymi środowiskowymi:

```
docker run --name my-zabbix-agent -e ZBX_HOSTNAME="new-hostname" -e BUFSZ=1000 -e LISTENPORT=20050 -e _Usr
```

3. Podczas używania poleceń runtime (np. w celu zwiększenia poziomu logowania agenta) uzyskaj dostęp do powłoki kontenera i wykonaj polecenie runtime:

```
docker exec -it <containerid> sh
/usr/sbin/zabbix_agentd -R log_level_increase
```

Attention:
Polecenie runtime `userparameter_reload` nie obsługuje ponownego wczytywania zmiennych środowiskowych. Podczas ponownego wczytywania zmienne są ignorowane, a ponownie wczytywane są tylko parametry ze zwykłymi wartościami.

11 Dołączanie

Przegląd

Dodatkowe pliki lub katalogi mogą być dołączane do konfiguracji serwera, proxy, agenta i usługi web service przy użyciu parametru `Include`.

Uwagi dotyczące dołączania

Jeśli parametr `Include` jest używany do dołączania pliku, plik musi być dostępny do odczytu.

Jeśli parametr `Include` jest używany do dołączania katalogu:

- Wszystkie pliki w katalogu muszą być dostępne do odczytu.
- Nie należy zakładać żadnej konkretnej kolejności dołączania (np. pliki nie są dołączane w kolejności alfabetycznej). Dlatego nie należy definiować jednego parametru w kilku plikach ”Include” (np. aby nadpisać ustawienie ogólne ustawieniem szczegółowym).
- Wszystkie pliki w katalogu są dołączane do konfiguracji.
- Należy uważać na kopie zapasowe plików tworzone automatycznie przez niektóre edytory tekstu. Na przykład, jeśli edycja pliku ”include/my_specific.conf” powoduje utworzenie kopii zapasowej ”include/my_specific_conf.BAK”, wówczas oba pliki zostaną dołączone. Przenieś ”include/my_specific.conf.BAK” poza katalog ”Include”. W systemie Linux zawartość katalogu ”Include” można sprawdzić poleceniem ”ls -al”, aby wykryć zbędne pliki.

Jeśli parametr `Include` jest używany do dołączania plików przy użyciu wzorca:

- Wszystkie pliki pasujące do wzorca muszą być dostępne do odczytu.
- Nie należy zakładać żadnej konkretnej kolejności dołączania (np. pliki nie są dołączane w kolejności alfabetycznej). Dlatego nie należy definiować jednego parametru w kilku plikach ”Include” (np. aby nadpisać ustawienie ogólne ustawieniem szczegółowym).

3 Protokoły

Ta sekcja opisuje protokoły komunikacyjne używane przez komponenty Zabbix, w tym wymianę danych serwer-proxy, komunikację z agent oraz formaty eksportu danych.

Skorzystaj z paska bocznego, aby znaleźć potrzebną specyfikację protokołu.

1 Protokół wymiany danych serwer-proxy

Przegląd

Wymiana danych serwer - proxy opiera się na formacie JSON.

Komunikaty żądania i odpowiedzi muszą zaczynać się od nagłówka i długości danych.

Pasywny proxy

Żądanie konfiguracji

Serwer najpierw wyśle puste żądanie `proxy config`. To żądanie jest wysyłane co `ProxyConfigFrequency` (sekunda parametru konfiguracji serwera).

Proxy odpowiada bieżącą wersją proxy, tokenem sesji i rewizją konfiguracji. Serwer odpowiada danymi konfiguracji, które wymagają aktualizacji.

name	value type	description
server→proxy: **request**	*string*	’proxy config’

name	value type	description
proxy→server:		
version	*string*	Wersja proxy (<major>.<minor>.<build>).
session	*string*	Token sesji konfiguracji proxy.
config_revision	*number*	Rewizja pamięci podręcznej konfiguracji proxy.
server→proxy:		
full_sync	*number*	1 - jeśli wysyłane są pełne dane konfiguracji; w przeciwnym razie brak (opcjonalne).
data	*array*	Obiekt danych tabeli. Brak, jeśli konfiguracja nie została zmieniona (opcjonalne).
<table>	*object*	Jeden lub więcej obiektów z danymi <table> (opcjonalne, zależnie od zmian).
fields	*array*	Tablica nazw pól.
-	*string*	Nazwa pola.
data	*array*	Tablica wierszy.
-	*array*	Tablica kolumn.
-	*string*,*number*	Wartość kolumny, której typ zależy od typu kolumny w schemacie bazy danych.
macro.secrets	*object*	Informacje o tajnych makrach, brak jeśli nie ma zmian w makrach vault (opcjonalne).
config_revision	*number*	Rewizja pamięci podręcznej konfiguracji - wysyłana wraz z danymi konfiguracji (opcjonalne).
del_hostids	*array*	Tablica usuniętych hostidów (opcjonalne).
-	*number*	Identyfikator hosta.
del_macro_hostids	*array*	Tablica hostidów, z których usunięto wszystkie makra (opcjonalne).
-	*number*	Identyfikator hosta.
proxy→server:		
response	*string*	Informacja o powodzeniu żądania ('success' lub 'failed').
version	*string*	Wersja proxy (<major>.<minor>.<build>).

Przykład:

server→proxy:

```
{
  "request":"proxy config"
}
```

proxy→server:

```
{
  "version": "7.4.0",
  "session": "00331249498008l1e5686dbfd9bcea98",
  "config_revision": 0
}
```

server→proxy:

```
{
"full_sync": 1,
"data": {
"hosts": {
"fields": ["hostid", "host", "status", "ipmi_authtype", "ipmi_privilege", "ipmi_username", "ipmi_password"
"data": [
[10084, "Zabbix server", 0, -1, 2, "", "", "Zabbix server", 1, 1, "", "", "", ""]
]
},
"interface": {
"fields": ["interfaceid", "hostid", "main", "type", "useip", "ip", "dns", "port", "available"],
"data": [
```

```
[1, 10084, 1, 1, 1, "127.0.0.1", "", "10053", 1]
]
},
"interface_snmp": {
"fields": ["interfaceid", "version", "bulk", "community", "securityname", "securitylevel", "authpassphrase
"data": []
},
"host_inventory": {
"fields": ["hostid", "type", "type_full", "name", "alias", "os", "os_full", "os_short", "serialno_a", "ser
"data": [
[10084, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "5
]
},
"items": {
"fields": ["itemid", "type", "snmp_oid", "hostid", "key_", "delay", "history", "status", "value_type", "tr
"data": [
[44161, 7, "", 10084, "agent.hostmetadata", "10s", "90d", 0, 1, "", "", "", "", 0, "", "", "", "", 0, null
[44162, 0, "", 10084, "agent.ping", "10s", "90d", 0, 3, "", "", "", "", 0, "", "", "", "", 0, 1, 0, "", nu
]
},
"item_rtdata": {
"fields": ["itemid", "lastlogsize", "mtime"],
"data": [
[44161, 0, 0],
[44162, 0, 0]
]
},
"item_preproc": {
"fields": ["item_preprocid", "itemid", "step", "type", "params", "error_handler", "error_handler_params"],
"data": []
},
"item_parameter": {
"fields": ["item_parameterid", "itemid", "name", "value"],
"data": []
},
"globalmacro": {
"fields": ["globalmacroid", "macro", "value", "type"],
"data": [
[2, "{$SNMP_COMMUNITY}", "public", 0]
]
},
"hosts_templates": {
"fields": ["hosttemplateid", "hostid", "templateid", "link_type"],
"data": []
},
"hostmacro": {
"fields": ["hostmacroid", "hostid", "macro", "value", "type", "automatic"],
"data": [
[5676, 10084, "{$M}", "AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix:Content", 2, 0]
]
},
"drules": {
"fields": ["druleid", "name", "iprange", "delay"],
"data": [
[2, "Local network", "127.0.0.1", "10s"]
]
},
"dchecks": {
"fields": ["dcheckid", "druleid", "type", "key_", "snmp_community", "ports", "snmpv3_securityname", "snmpv
"data": [
[2, 2, 9, "system.uname", "", "10052", "", 0, "", "", 0, 0, 0, "", 1, 0]
]
```

```
},
"regexps": {
"fields": ["regexpid", "name"],
"data": [
[1, "File systems for discovery"],
[2, "Network interfaces for discovery"],
[3, "Storage devices for SNMP discovery"],
[4, "Windows service names for discovery"],
[5, "Windows service startup states for discovery"]
]
},
"expressions": {
"fields": ["expressionid", "regexpid", "expression", "expression_type", "exp_delimiter", "case_sensitive"]
"data": [
[1, 1, "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs)$", 3, ",", 0
[3, 3, "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$", 4, ",", 1],
[5, 4, "^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$", 4, ","
[6, 5, "^(automatic|automatic delayed)$", 3, ",", 1],
[7, 2, "^Software Loopback Interface", 4, ",", 1],
[8, 2, "^(In)?[Ll]oop[Bb]ack[0-9._]*$", 4, ",", 1],
[9, 2, "^NULL[0-9.]*$", 4, ",", 1],
[10, 2, "^[Ll]o[0-9.]*$", 4, ",", 1],
[11, 2, "^[Ss]ystem$", 4, ",", 1],
[12, 2, "^Nu[0-9.]*$", 4, ",", 1]
]
},
"settings": {
"fields": ["name", "type", "value_str", "value_int"],
"data": [
["autoreg_tls_accept", 2, "", 1],
["hk_history_global", 2, "", 0],
["snmptrap_logging", 2, "", 1],
["proxy_secrets_provider", 2, "", 0],
["hk_history", 1, "31d", 0],
["timeout_db_monitor", 1, "3s", 0],
["timeout_external_check", 1, "3s", 0],
["timeout_http_agent", 1, "3s", 0],
["timeout_simple_check", 1, "3s", 0],
["timeout_snmp_agent", 1, "3s", 0],
["timeout_ssh_agent", 1, "3s", 0],
["timeout_telnet_agent", 1, "3s", 0],
["timeout_zabbix_agent", 1, "3s", 0],
["timeout_browser", 1, "30s", 0]
]
},
"httptest": {
"fields": ["httptestid", "name", "delay", "agent", "authentication", "http_user", "http_password", "hostid
"data": []
},
"httptestitem": {
"fields": ["httptestitemid", "httptestid", "itemid", "type"],
"data": []
},
"httptest_field": {
"fields": ["httptest_fieldid", "httptestid", "type", "name", "value"],
"data": []
},
"httpstep": {
"fields": ["httpstepid", "httptestid", "name", "no", "url", "timeout", "posts", "required", "status_codes"
"data": []
},
"httpstepitem": {
```

```
"fields": ["httpstepitemid", "httpstepid", "itemid", "type"],
"data": []
},
"httpstep_field": {
"fields": ["httpstep_fieldid", "httpstepid", "type", "name", "value"],
"data": []
},
"config_autoreg_tls": {
"fields": ["autoreg_tlsid", "tls_psk_identity", "tls_psk"],
"data": [
[1, "", ""]
]
}
},
"macro.secrets": {
"AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix": {
"Content": "738"
}
},
"config_revision": 2
}
```


proxy→server:


```
{
  "response": "success",
  "version": "7.4.0"
}
```


Żądanie danych proxy

Żądanie `proxy data` służy do pobierania dostępności interfejsów hosta, danych historycznych, danych wykrywania oraz danych autorejestracji z proxy. To żądanie jest wysyłane co `ProxyDataFrequency` (parametr konfiguracji serwera) sekund.

name	value type	description
server→proxy:		
request	*string*	'proxy data'
proxy→server:		
session	*string*	Token sesji danych.
interface availability	*array*	*(optional)* Tablica obiektów dostępności interfejsów.
interfaceid	*number*	Identyfikator interfejsu.
available	*number*	Dostępność interfejsu: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - nieznana **1**, *INTERFACE_AVAILABLE_TRUE* - dostępny **2**, *INTERFACE_AVAILABLE_FALSE* - niedostępny
error	*string*	Komunikat błędu interfejsu lub pusty ciąg znaków.
history data	*array*	*(optional)* Tablica obiektów danych historycznych.
itemid	*number*	Identyfikator pozycji.
clock	*number*	Znacznik czasu wartości pozycji (sekundy).
ns	*number*	Znacznik czasu wartości pozycji (nanosekundy).
value	*string*	*(optional)* Wartość pozycji.
id	*number*	Identyfikator wartości (rosnący licznik, unikalny w ramach jednej sesji danych).
timestamp	*number*	*(optional)* Znacznik czasu dla pozycji typu log.
source	*string*	*(optional)* Wartość źródła pozycji eventlog.
severity	*number*	*(optional)* Wartość ważności pozycji eventlog.
eventid	*number*	*(optional)* Wartość eventid pozycji eventlog.

name	value type	description
state	*string*	*(optional)* Stan pozycji: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*
lastlogsize	*number*	*(optional)* Ostatni rozmiar logu dla pozycji typu log.
mtime	*number*	*(optional)* Czas modyfikacji dla pozycji typu log.
discovery data	*array*	*(optional)* Tablica obiektów danych wykrywania.
clock	*number*	Znacznik czasu danych wykrywania.
druleid	*number*	Identyfikator reguły wykrywania.
dcheckid	*number*	Identyfikator sprawdzenia wykrywania lub null dla danych reguły wykrywania.
type	*number*	Typ sprawdzenia wykrywania: **-1** dane reguły wykrywania **0**, *SVC_SSH* - sprawdzenie usługi SSH **1**, *SVC_LDAP* - sprawdzenie usługi LDAP **2**, *SVC_SMTP* - sprawdzenie usługi SMTP **3**, *SVC_FTP* - sprawdzenie usługi FTP **4**, *SVC_HTTP* - sprawdzenie usługi HTTP **5**, *SVC_POP* - sprawdzenie usługi POP **6**, *SVC_NNTP* - sprawdzenie usługi NNTP **7**, *SVC_IMAP* - sprawdzenie usługi IMAP **8**, *SVC_TCP* - sprawdzenie dostępności portu TCP **9**, *SVC_AGENT* - agent Zabbixa **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - sprawdzenie usługi HTTPS **15**, *SVC_TELNET* - sprawdzenie dostępności Telnet
ip	*string*	Adres IP hosta.
dns	*string*	Nazwa DNS hosta.
port	*number*	*(optional)* Numer portu usługi.
key_	*string*	*(optional)* Klucz pozycji dla sprawdzenia wykrywania typu **9** *SVC_AGENT*
value	*string*	*(optional)* Wartość otrzymana z usługi, dla większości usług może być pusta.
status	*number*	*(optional)* Status usługi: **0**, *DOBJECT_STATUS_UP* - usługa UP **1**, *DOBJECT_STATUS_DOWN* - usługa DOWN
auto registration	*array*	*(optional)* Tablica obiektów danych autorejestracji.
clock	*number*	Znacznik czasu danych autorejestracji.
host	*string*	Nazwa hosta.
ip	*string*	*(optional)* Adres IP hosta.
dns	*string*	*(optional)* Rozwiązana nazwa DNS z adresu IP.
port	*string*	*(optional)* Port hosta.
host_metadata	*string*	*(optional)* Metadane hosta wysłane przez agent (na podstawie parametru konfiguracji agenta HostMetadata lub HostMetadataItem).
tasks	*array*	*(optional)* Tablica zadań.
type	*number*	Typ zadania: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - wynik zdalnego polecenia
status	*number*	Status wykonania zdalnego polecenia: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - zdalne polecenie wykonane pomyślnie **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - zdalne polecenie nie powiodło się

name	value type	description
error	*string*	*(optional)* Komunikat błędu.
parent_taskid	*number*	ID zadania nadrzędnego.
more	*number*	*(optional)* 1 - do wysłania pozostały kolejne dane historyczne.
clock	*number*	*(optional)* Znacznik czasu przesyłania danych (sekundy).
ns	*number*	*(optional)* Znacznik czasu przesyłania danych (nanosekundy).
version	*string*	Wersja proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Informacja o powodzeniu żądania (’success’ lub ’failed’).
tasks	*array*	*(optional)* Tablica zadań.
type	*number*	Typ zadania: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - zdalne polecenie
clock	*number*	Czas utworzenia zadania.
ttl	*number*	Czas w sekundach, po którym zadanie wygasa.
commandtype	*number*	Typ zdalnego polecenia: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - użyj niestandardowego skryptu **1**, *ZBX_SCRIPT_TYPE_IPMI* - użyj IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - użyj SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - użyj Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - użyj globalnego skryptu (obecnie funkcjonalnie równoważny niestandardowemu skryptowi)
command	*string*	Zdalne polecenie do wykonania.
execute_on	*number*	Cel wykonania dla niestandardowych skryptów: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - wykonaj skrypt na agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - wykonaj skrypt na serwer **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - wykonaj skrypt na proxy
port	*number*	*(optional)* Port dla poleceń Telnet i SSH.
authtype	*number*	*(optional)* Typ uwierzytelniania dla poleceń SSH.
username	*string*	*(optional)* Nazwa użytkownika dla poleceń Telnet i SSH.
password	*string*	*(optional)* Hasło dla poleceń Telnet i SSH.
publickey	*string*	*(optional)* Klucz publiczny dla poleceń SSH.
privatekey	*string*	*(optional)* Klucz prywatny dla poleceń SSH.
parent_taskid	*number*	ID zadania nadrzędnego.
hostid	*number*	ID docelowego hosta.

Przykład:

server→proxy:

```
{
  "request": "proxy data"
}
```

proxy→server:

```
{
    "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": 0,
            "status": 0,
            "parent_taskid": 10
        },
        {
            "type": 0,
            "status": 1,
            "error": "No permissions to execute task.",
            "parent_taskid": 20
        }
    ],
    "version":"7.4.0"
}
```

server→proxy:

```
{
  "response": "success",
  "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": "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCqGKukO1De7zhZj6+H0qtjTkVxwTCpvKe",
         "privatekey": "lsuusFncCzWBQ7RKNUSesmQRMSGkVb1/3j+skZ6UtW+5u09lHNsj6tQ5QCqGKukO1De7zhd",
         "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
      }
  ]
}
```

Żądanie zadań

Wymiana `proxy tasks` obsługuje dostarczanie i potwierdzanie zadań zdalnych poleceń między serwerem a pasywnym proxy. Gdy serwer odpytuje proxy o wyniki zadań, wysyła puste żądanie `proxy tasks` w interwale zdefiniowanym przez `ProxyDataFrequency`. Proxy odpowiada swoją bieżącą wersją oraz wszystkimi oczekującymi wynikami zadań (w tym znacznikami czasu). Następnie serwer potwierdza odbiór, zwracając `response`, i może dołączyć nowe zadania do wykonania przez proxy.

name	value type	description
serwer→proxy:		
request	*string*	'proxy tasks'
proxy→serwer:		
version	*string*	Wersja proxy (<major>.<minor>.<build>).
clock	*number*	*(opcjonalnie)* Znacznik czasu transferu danych (sekundy).
ns	*number*	*(opcjonalnie)* Znacznik czasu transferu danych (nanosekundy).
serwer→proxy:		
response	*string*	Informacja o powodzeniu żądania ('success' lub 'failed').

Przykład:

serwer→proxy:

```
{
  "request":"proxy tasks"
}
```

proxy→serwer:

```
{
  "version":"7.0.0",
  "clock":1721059872,
  "ns":848141522
}
```

serwer→proxy:

```
{
  "response":"success"
}
```

Aktywny proxy

Żądanie konfiguracji

Żądanie `proxy config` jest wysyłane przez aktywny proxy w celu pobrania danych konfiguracji proxy. To żądanie jest wysyłane co `ProxyConfigFrequency` (parametr konfiguracji proxy) sekund.

name	value type	description
proxy→server:		
request	*string*	'proxy config'
host	*string *	Nazwa proxy.
version	*string*	Wersja proxy (<major>.<minor>.<build>).
session	*string*	Token sesji konfiguracji proxy.
config_revision	*number*	Rewizja konfiguracji proxy.
hostmap_revision	*number*	Bieżąca rewizja mapowania hostów do proxy.
server→proxy:		
full_sync	*number*	1 - jeśli wysyłane są pełne dane konfiguracji, w przeciwnym razie brak (opcjonalne).
data	*array*	Obiekt danych tabeli. Brak, jeśli konfiguracja nie została zmieniona (opcjonalne).
<table>	*object*	Jeden lub więcej obiektów z danymi <table> (opcjonalne, zależnie od zmian).
fields	*array*	Tablica nazw pól.
-	*string*	Nazwa pola.
data	*array*	Tablica wierszy.
-	*array*	Tablica kolumn.
-	*string,number*	Wartość kolumny, której typ zależy od typu kolumny w schemacie bazy danych.
macro.secrets	*object*	Informacje o tajnych makrach, brak jeśli nie ma zmian w makrach vault (opcjonalne).
proxy_group	*string*	Nazwa grupy proxy, do której należy proxy.

name	value type	description
config_revision	*number*	Rewizja pamięci podręcznej konfiguracji - wysyłana z danymi konfiguracji (opcjonalne).
del_hostids	*array*	Tablica usuniętych hostidów (opcjonalne).
-	*number*	Identyfikator hosta.
del_macro_hostids	*array*	Tablica hostidów, z których usunięto wszystkie makra (opcjonalne).
-	*number*	Identyfikator hosta.

Przykład:

proxy→server:

```
{
  "request": "proxy config",
  "host": "Zabbix proxy",
  "version":"7.4.0",
  "session": "fd59a09ff4e9d1fb447de1f04599bcf6",
  "config_revision": 0
}
```

server→proxy:

```
{
"full_sync": 1,
"data": {
"hosts": {
"fields": ["hostid", "host", "status", "ipmi_authtype", "ipmi_privilege", "ipmi_username", "ipmi_password"
"data": [
[10084, "Zabbix server", 0, -1, 2, "", "", "Zabbix server", 1, 1, "", "", "", ""]
]
},
"interface": {
"fields": ["interfaceid", "hostid", "main", "type", "useip", "ip", "dns", "port", "available"],
"data": [
[1, 10084, 1, 1, 1, "127.0.0.1", "", "10053", 1]
]
},
"interface_snmp": {
"fields": ["interfaceid", "version", "bulk", "community", "securityname", "securitylevel", "authpassphrase
"data": []
},
"host_inventory": {
"fields": ["hostid", "type", "type_full", "name", "alias", "os", "os_full", "os_short", "serialno_a", "ser
"data": [
[10084, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "5
]
},
"items": {
"fields": ["itemid", "type", "snmp_oid", "hostid", "key_", "delay", "history", "status", "value_type", "tr
"data": [
[44161, 7, "", 10084, "agent.hostmetadata", "10s", "90d", 0, 1, "", "", "", "", 0, "", "", "", "", 0, null
[44162, 0, "", 10084, "agent.ping", "10s", "90d", 0, 3, "", "", "", "", 0, "", "", "", "", 0, 1, 0, "", nu
]
},
"item_rtdata": {
"fields": ["itemid", "lastlogsize", "mtime"],
"data": [
[44161, 0, 0],
[44162, 0, 0]
]
},
"item_preproc": {
```

```
"fields": ["item_preprocid", "itemid", "step", "type", "params", "error_handler", "error_handler_params"],
"data": []
},
"item_parameter": {
"fields": ["item_parameterid", "itemid", "name", "value"],
"data": []
},
"globalmacro": {
"fields": ["globalmacroid", "macro", "value", "type"],
"data": [
[2, "{$SNMP_COMMUNITY}", "public", 0]
]
},
"hosts_templates": {
"fields": ["hosttemplateid", "hostid", "templateid", "link_type"],
"data": []
},
"hostmacro": {
"fields": ["hostmacroid", "hostid", "macro", "value", "type", "automatic"],
"data": [
[5676, 10084, "{$M}", "AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix:Content", 2, 0]
]
},
"drules": {
"fields": ["druleid", "name", "iprange", "delay"],
"data": [
[2, "Local network", "127.0.0.1", "10s"]
]
},
"dchecks": {
"fields": ["dcheckid", "druleid", "type", "key_", "snmp_community", "ports", "snmpv3_securityname", "snmpv
"data": [
[2, 2, 9, "system.uname", "", "10052", "", 0, "", "", 0, 0, 0, "", 1, 0]
]
},
"regexps": {
"fields": ["regexpid", "name"],
"data": [
[1, "File systems for discovery"],
[2, "Network interfaces for discovery"],
[3, "Storage devices for SNMP discovery"],
[4, "Windows service names for discovery"],
[5, "Windows service startup states for discovery"]
]
},
"expressions": {
"fields": ["expressionid", "regexpid", "expression", "expression_type", "exp_delimiter", "case_sensitive"]
"data": [
[1, 1, "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs)$", 3, ",", 0
[3, 3, "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$", 4, ",", 1],
[5, 4, "^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$", 4, ","
[6, 5, "^(automatic|automatic delayed)$", 3, ",", 1],
[7, 2, "^Software Loopback Interface", 4, ",", 1],
[8, 2, "^(In)?[Ll]oop[Bb]ack[0-9._]*$", 4, ",", 1],
[9, 2, "^NULL[0-9.]*$", 4, ",", 1],
[10, 2, "^[Ll]o[0-9.]*$", 4, ",", 1],
[11, 2, "^[Ss]ystem$", 4, ",", 1],
[12, 2, "^Nu[0-9.]*$", 4, ",", 1]
]
},
"settings": {
"fields": ["name", "type", "value_str", "value_int"],
```

```
"data": [
["autoreg_tls_accept", 2, "", 1],
["hk_history_global", 2, "", 0],
["snmptrap_logging", 2, "", 1],
["proxy_secrets_provider", 2, "", 0],
["hk_history", 1, "31d", 0],
["timeout_db_monitor", 1, "3s", 0],
["timeout_external_check", 1, "3s", 0],
["timeout_http_agent", 1, "3s", 0],
["timeout_simple_check", 1, "3s", 0],
["timeout_snmp_agent", 1, "3s", 0],
["timeout_ssh_agent", 1, "3s", 0],
["timeout_telnet_agent", 1, "3s", 0],
["timeout_zabbix_agent", 1, "3s", 0],
["timeout_browser", 1, "30s", 0]
]
},
"httptest": {
"fields": ["httptestid", "name", "delay", "agent", "authentication", "http_user", "http_password", "hostid
"data": []
},
"httptestitem": {
"fields": ["httptestitemid", "httptestid", "itemid", "type"],
"data": []
},
"httptest_field": {
"fields": ["httptest_fieldid", "httptestid", "type", "name", "value"],
"data": []
},
"httpstep": {
"fields": ["httpstepid", "httptestid", "name", "no", "url", "timeout", "posts", "required", "status_codes"
"data": []
},
"httpstepitem": {
"fields": ["httpstepitemid", "httpstepid", "itemid", "type"],
"data": []
},
"httpstep_field": {
"fields": ["httpstep_fieldid", "httpstepid", "type", "name", "value"],
"data": []
},
"config_autoreg_tls": {
"fields": ["autoreg_tlsid", "tls_psk_identity", "tls_psk"],
"data": [
[1, "", ""]
]
}
},
"macro.secrets": {
"AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix": {
"Content": "738"
}
},
"config_revision": 2
}
```


Żądanie danych proxy

Żądanie `proxy data` jest wysyłane przez proxy w celu przekazania danych o dostępności interfejsów hosta, historii, wykrywania oraz autorejestracji. To żądanie jest wysyłane co `DataSenderFrequency` sekund (parametr konfiguracji proxy). Należy pamiętać, że aktywne proxy nadal będzie odpytywać serwer Zabbix co sekundę o zadania zdalnych poleceń (z pustym żądaniem `proxy data`).

name	value type	description
proxy→server:		
request	*string*	'proxy data'
host	*string*	Nazwa proxy.
session	*string*	Token sesji danych.
interface availability	*array*	*(opcjonalnie)* Tablica obiektów danych o dostępności interfejsów.
interfaceid	*number*	Identyfikator interfejsu.
available	*number*	Dostępność interfejsu: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - nieznana **1**, *INTERFACE_AVAILABLE_TRUE* - dostępny **2**, *INTERFACE_AVAILABLE_FALSE* - niedostępny
error	*string*	Komunikat błędu interfejsu lub pusty ciąg znaków.
history data	*array*	*(opcjonalnie)* Tablica obiektów danych historycznych.
itemid	*number*	Identyfikator pozycji.
clock	*number*	Znacznik czasu wartości pozycji (sekundy).
ns	*number*	Znacznik czasu wartości pozycji (nanosekundy).
value	*string*	*(opcjonalnie)* Wartość pozycji.
id	*number*	Identyfikator wartości (rosnący licznik, unikalny w ramach jednej sesji danych).
timestamp	*number*	*(opcjonalnie)* Znacznik czasu dla pozycji typu log.
source	*string*	*(opcjonalnie)* Wartość źródła pozycji eventlog.
severity	*number*	*(opcjonalnie)* Wartość ważności pozycji eventlog.
eventid	*number*	*(opcjonalnie)* Wartość eventid pozycji eventlog.
state	*string*	*(opcjonalnie)* Stan pozycji: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*
lastlogsize	*number*	*(opcjonalnie)* Ostatni rozmiar logu dla pozycji typu log.
mtime	*number*	*(opcjonalnie)* Czas modyfikacji dla pozycji typu log.
discovery data	*array*	*(opcjonalnie)* Tablica obiektów danych wykrywania.
clock	*number*	Znacznik czasu danych wykrywania.
druleid	*number*	Identyfikator reguły wykrywania.
dcheckid	*number*	Identyfikator sprawdzenia wykrywania lub null dla danych reguły wykrywania.
type	*number*	Typ sprawdzenia wykrywania: **-1** dane reguły wykrywania **0**, *SVC_SSH* - sprawdzenie usługi SSH **1**, *SVC_LDAP* - sprawdzenie usługi LDAP **2**, *SVC_SMTP* - sprawdzenie usługi SMTP **3**, *SVC_FTP* - sprawdzenie usługi FTP **4**, *SVC_HTTP* - sprawdzenie usługi HTTP **5**, *SVC_POP* - sprawdzenie usługi POP **6**, *SVC_NNTP* - sprawdzenie usługi NNTP **7**, *SVC_IMAP* - sprawdzenie usługi IMAP **8**, *SVC_TCP* - sprawdzenie dostępności portu TCP **9**, *SVC_AGENT* - agent Zabbix **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - sprawdzenie usługi HTTPS **15**, *SVC_TELNET* - sprawdzenie dostępności Telnet
ip	*string*	Adres IP hosta.
dns	*string*	Nazwa DNS hosta.
port	*number*	*(opcjonalnie)* Numer portu usługi.
key_	*string*	*(opcjonalnie)* Klucz pozycji dla sprawdzenia wykrywania typu **9** *SVC_AGENT*

name	value type	description
value	*string*	*(opcjonalnie)* Wartość otrzymana z usługi, dla większości usług może być pusta.
status	*number*	*(opcjonalnie)* Status usługi: **0**, *DOBJECT_STATUS_UP* - usługa działa **1**, *DOBJECT_STATUS_DOWN* - usługa nie działa
autoregistration	*array*	*(opcjonalnie)* Tablica obiektów danych autorejestracji.
clock	*number*	Znacznik czasu danych autorejestracji.
host	*string*	Nazwa hosta.
ip	*string*	*(opcjonalnie)* Adres IP hosta.
dns	*string*	*(opcjonalnie)* Rozwiązana nazwa DNS z adresu IP.
port	*string*	*(opcjonalnie)* Port hosta.
host_metadata	*string*	*(opcjonalnie)* Metadane hosta wysyłane przez agent (na podstawie parametru konfiguracji agenta HostMetadata lub HostMetadataItem).
tasks	*array*	*(opcjonalnie)* Tablica zadań.
type	*number*	Typ zadania: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - wynik zdalnego polecenia
status	*number*	Status wykonania zdalnego polecenia: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - zdalne polecenie wykonane pomyślnie **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - zdalne polecenie nie powiodło się
error	*string*	*(opcjonalnie)* Komunikat błędu.
parent_taskid	*number*	Identyfikator nadrzędnego zadania.
more	*number*	*(opcjonalnie)* 1 - do wysłania pozostały kolejne dane historyczne
clock	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (sekundy).
ns	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (nanosekundy).
version	*string*	Wersja proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Informacja o powodzeniu żądania (’success’ lub ’failed’).
upload	*string*	Kontrola wysyłania danych historycznych (historia, autorejestracja, dostępność hosta, wykrywanie sieci). Możliwe wartości: **enabled** - normalne działanie **disabled** - serwer nie przyjmuje danych (możliwe, że z powodu przekroczenia wewnętrznej pamięci podręcznej)
tasks	*array*	*(opcjonalnie)* Tablica zadań.
type	*number*	Typ zadania: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - zdalne polecenie
clock	*number*	Czas utworzenia zadania.
ttl	*number*	Czas w sekundach, po którym zadanie wygasa.
commandtype	*number*	Typ zdalnego polecenia: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - użyj własnego skryptu **1**, *ZBX_SCRIPT_TYPE_IPMI* - użyj IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - użyj SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - użyj Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - użyj skryptu globalnego (obecnie funkcjonalnie równoważny własnemu skryptowi)
command	*string*	Zdalne polecenie do wykonania.
execute_on	*number*	Cel wykonania dla własnych skryptów: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - wykonaj skrypt na agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - wykonaj skrypt na serwer **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - wykonaj skrypt na proxy
port	*number*	*(opcjonalnie)* Port dla poleceń Telnet i SSH.
authtype	*number*	*(opcjonalnie)* Typ uwierzytelniania dla poleceń SSH.

name	value type	description
username	*string*	*(opcjonalnie)* Nazwa użytkownika dla poleceń Telnet i SSH.
password	*string*	*(opcjonalnie)* Hasło dla poleceń Telnet i SSH.
publickey	*string*	*(opcjonalnie)* Klucz publiczny dla poleceń SSH.
privatekey	*string*	*(opcjonalnie)* Klucz prywatny dla poleceń SSH.
parent_taskid	*number*	Identyfikator nadrzędnego zadania.
hostid	*number*	Identyfikator docelowego hosta.

Przykład:

proxy→server:


```
{
"request": "proxy data",
"host": "Zabbix proxy",
"session": "818cdd1b537bdc5e50c09ed4969235b6",
"interface availability": [{
"interfaceid": 1,
"available": 1,
"error": ""
}],
"history data": [{
"id": 1114,
"itemid": 44162,
"clock": 1665730632,
"ns": 798953105,
"value": "1"
}, {
"id": 1115,
"itemid": 44161,
"clock": 1665730633,
"ns": 811684663,
"value": "58"
}],
"auto registration": [{
"clock": 1665730633,
"host": "Zabbix server",
"ip": "127.0.0.1",
"dns": "localhost",
"port": "10053",
"host_metadata": "58",
"tls_accepted": 1
}],
"discovery data": [{
"clock": 1665732232,
"drule": 2,
"dcheck": 2,
"ip": "127.0.0.1",
"dns": "localhost",
"port": 10052,
"status": 1
}, {
"clock": 1665732232,
"drule": 2,
"dcheck": null,
"ip": "127.0.0.1",
"dns": "localhost",
"status": 1
}],
"host data": [{
"hostid": 10084,
"active_status": 1
}],
```

```
"tasks": [{
"type": 3,
"clock": 1665730985,
"ttl": 0,
"status": -1,
"info": "Remote commands are not enabled",
"parent_taskid": 3
}],
"version": "7.4.0",
"clock": 1665730643,
"ns": 65389964
}
```


server→proxy:


```
{
"upload": "enabled",
"response": "success",
"tasks": [{
"type": 2,
"clock": 1665730986,
"ttl": 600,
"commandtype": 0,
"command": "ping -c 3 127.0.0.1; case $? in [01]) true;; *) false;; esac",
"execute_on": 2,
"port": 0,
"authtype": 0,
"username": "",
"password": "",
"publickey": "",
"privatekey": "",
"alertid": 0,
"parent_taskid": 4,
"hostid": 10084
}]
}
```


2 Protokół Zabbix agent/agent2

Więcej informacji o protokołach Zabbix agent i Zabbix agent 2 można znaleźć na stronie Passive and active agent checks.

4 Protokół pluginów Zabbix agent 2

Protokół Zabbix agent 2 opiera się na modelu kodu, rozmiaru i danych.

Kod

Typ	Rozmiar	Komentarze
Bajt	4	Typ ładunku, obecnie obsługiwany jest tylko JSON.

Rozmiar

Type	Size	Comments
Bajt	4	Rozmiar bieżącego ładunku w bajtach.

Dane ładunku

Typ	Rozmiar	Komentarze
Byte	Zdefiniowany przez pole *Rozmiar*	Dane w formacie JSON.

Definicja danych payloadu

Wspólne dane

Te parametry są obecne we wszystkich żądaniach/odpowiedziach:

Nazwa	Typ	Komentarze
id	uint32	Dla żądań — inkrementowany identyfikator używany do powiązania żądań z odpowiedziami. Unikalny w obrębie kierunku żądania (tj. od agent do wtyczki lub od wtyczki do agent). Dla odpowiedzi — ID odpowiadającego żądania.
type	uint32	Typ żądania.

Żądanie logowania

Żądanie wysyłane przez wtyczkę w celu zapisania komunikatu dziennika w pliku dziennika agenta.

direction	plugin → agent
response	no

Parametry specyficzne dla żądań logowania:

Name	Type	Comments
severity	uint32	Poziom ważności komunikatu (poziom logowania).
message	string	Komunikat do zapisania w dzienniku.

Przykład:

```
{"id":0,"type":1,"severity":3,"message":"message"}
```

Żądanie rejestracji

Żądanie wysyłane przez agent podczas fazy uruchamiania agent w celu uzyskania udostępnianych metryk do zarejestrowania wtyczki.

direction	agent → plugin
response	yes

Parametry specyficzne dla żądań rejestracji:

Name	Type	Comments
version	string	Wersja protokołu <major>.<minor>

Przykład:

```
{"id":1,"type":2,"version":"1.0"}
```

Odpowiedź rejestracji

Odpowiedź pluginu na żądanie rejestracji.

kierunek	plugin → agent
odpowiedź	n/d

Parametry specyficzne dla odpowiedzi rejestracji:

Nazwa	Typ	Uwagi
name	string	Nazwa pluginu.
metrics	array of strings (optional)	Metryki wraz z opisami używanymi w pluginie. Zwraca RegisterMetrics(). Nieobecne, jeśli zwracany jest błąd.
interfaces	uint32 (optional)	Maska bitowa interfejsów obsługiwanych przez plugin. Nieobecne, jeśli zwracany jest błąd.
error	string (optional)	Komunikat błędu zwracany, jeśli plugin nie może zostać uruchomiony. Nieobecne, jeśli zwracane są metryki.

Przykłady:

```
{"id":2,"type":3,"metrics":["external.test", "External exporter Test."], "interfaces": 4}
```

lub

```
{"id":2,"type":3,"error":"error message"}
```

Żądanie Start

Żądanie wykonania funkcji Start interfejsu Runner.

direction	agent → plugin
response	nie

Żądanie nie ma specyficznych parametrów, zawiera tylko parametry wspólnych danych.

Przykład:

```
{"id":3,"type":4}
```

Żądanie zakończenia

Żądanie wysyłane przez agent w celu zamknięcia wtyczki.

direction	agent → plugin
response	no

Żądanie nie ma specyficznych parametrów, zawiera jedynie parametry wspólnych danych.

Przykład:

```
{"id":3,"type":5}
```

Żądanie eksportu

Żądanie wykonania funkcji Export interfejsu Exporter.

direction	agent → plugin
response	nie

Parametry specyficzne dla żądań eksportu:

Name	Type	Comments
key	string	Klucz pluginu.
parameters	array of strings (optional)	Parametry funkcji Export.

Przykład:

```
{"id":4,"type":6,"key":"test.key","parameters":["foo","bar"]}
```

Odpowiedź eksportu

Odpowiedź z funkcji Export interfejsu Exporter.

kierunek	plugin → agent
odpowiedź	n/d

Parametry specyficzne dla odpowiedzi eksportu:

Nazwa	Typ	Komentarze
value	string (optional)	Wartość odpowiedzi z funkcji Export. Nieobecne, jeśli zwracany jest błąd.
error	string (optional)	Komunikat błędu, jeśli funkcja Export nie została wykonana pomyślnie. Nieobecne, jeśli zwracana jest wartość.

Przykłady:

```
{"id":5,"type":7,"value":"response"}
```

lub

```
{"id":5,"type":7,"error":"error message"}
```

Konfiguracja żądania

Żądanie wykonania funkcji *Configure* interfejsu *Configurator*.

direction	agent → plugin
response	n/a

Parametry specyficzne dla żądań *Configure*:

Name	Type	Comments
global_options	JSON object	Obiekt JSON zawierający globalne opcje konfiguracji agenta.
private_options	JSON object (optional)	Obiekt JSON zawierający prywatne opcje konfiguracji pluginu, jeśli zostały podane.

Przykład:

```
{"id":6,"type":8,"global_options":{...},"private_options":{...}}
```

Walidacja żądania

Żądanie wykonania funkcji *Validate* interfejsu *Configurator*.

direction	agent → plugin
response	yes

Parametry specyficzne dla żądań *Validate*:

Name	Type	Comments
private_options	JSON object (optional)	Obiekt JSON zawierający prywatne opcje konfiguracji pluginu, jeśli zostały podane.

Przykład:

```
{"id":7,"type":9,"private_options":{...}}
```

Walidacja odpowiedzi

Odpowiedź z funkcji *Validate* interfejsu *Configurator*.

direction	plugin → agent
response	n/a

Parametry specyficzne dla odpowiedzi *Validate*:

Name	Type	Comments
error	string (optional)	Komunikat o błędzie zwracany, jeśli funkcja Validate nie zostanie wykonana pomyślnie. Nieobecny, jeśli wykonanie zakończy się powodzeniem.

Przykład:

```
{"id":8,"type":10}
```

lub

```
{"id":8,"type":10,"error":"error message"}
```

5 Protokół Zabbix sender

Przegląd

serwer Zabbix i proxy Zabbix używają protokołu komunikacyjnego opartego na JSON do odbierania danych z Zabbix sender. Dane mogą być odbierane za pomocą pozycji trapper lub pozycji HTTP agent z włączonym przechwytywaniem.

Komunikaty żądania i odpowiedzi muszą zaczynać się od nagłówka i długości danych.

Żądanie Zabbix sender

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value"
        }
    ]
}
```

Odpowiedź serwera Zabbix

```
{
    "response": "success",
    "info": "processed: 1; failed: 0; total: 1; seconds spent: 0.060753"
}
```

Żądanie Zabbix sender ze znacznikiem czasu

Alternatywnie Zabbix sender może wysłać żądanie ze znacznikiem czasu i nanosekundami.

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value",
            "clock": 1516710794,
            "ns": 592397170
        },
```

```
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value",
            "clock": 1516710795,
            "ns": 192399456
        }
    ],
    "clock": 1516712029,
    "ns": 873386094
}
```

Odpowiedź serwera Zabbix

```
{
    "response": "success",
    "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.060904"
}
```

6 Nagłówek

Przegląd

Nagłówek jest obecny we wszystkich komunikatach żądania i odpowiedzi między komponentami Zabbix. Jest wymagany do określenia długości komunikatu, tego, czy jest on skompresowany, czy nie, oraz tego, czy jest to duży pakiet, czy nie.

Protokół komunikacyjny Zabbix ma limit rozmiaru pakietu wynoszący 1 GB na połączenie. Limit 1 GB ma zastosowanie zarówno do długości odebranych danych pakietu, jak i do długości danych po dekompresji.

Podczas wysyłania konfiguracji do Zabbix proxy limit rozmiaru pakietu jest zwiększany do 4 GB, aby umożliwić synchronizację dużych konfiguracji. Gdy długość danych przed kompresją przekracza 4 GB, Zabbix server automatycznie zaczyna używać formatu dużych pakietów (flaga 0x04), co zwiększa limit rozmiaru pakietu do 16 GB.

Należy pamiętać, że chociaż format dużych pakietów może być używany do wysyłania dowolnych danych, obecnie tylko synchronizator konfiguracji Zabbix proxy może obsługiwać pakiety większe niż 1 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

Przykłady

Poniżej znajdują się fragmenty kodu pokazujące, jak dodać nagłówek protokołu Zabbix do danych, które chcesz wysłać, aby uzyskać pakiet, który należy wysłać do Zabbix, tak aby został poprawnie zinterpretowany. Te fragmenty kodu zakładają, że dane nie są większe niż 1 GB, dlatego format dużych pakietów nie jest używany.

Python

```
packet = b"ZBXD\1" + struct.pack("<II", len(data), 0) + data
```

lub

```
def zbx_create_header(plain_data_size, compressed_data_size=None):
    protocol = b"ZBXD"
    flags = 0x01
    if compressed_data_size is None:
        datalen = plain_data_size
        reserved = 0
    else:
        flags |= 0x02
        datalen = compressed_data_size
        reserved = plain_data_size
    return protocol + struct.pack("<BII", flags, datalen, reserved)

packet = zbx_create_header(len(data)) + data
```

Perl

```
my $packet = "ZBXD\1" . pack("(II)<", length($data), 0) . $data;
```

lub

```
sub zbx_create_header($;$)
{
    my $plain_data_size = shift;
    my $compressed_data_size = shift;

    my $protocol = "ZBXD";
    my $flags = 0x01;
    my $datalen;
    my $reserved;

    if (!defined($compressed_data_size))
    {
        $datalen = $plain_data_size;
        $reserved = 0;
    }
    else
    {
        $flags |= 0x02;
        $datalen = $compressed_data_size;
        $reserved = $plain_data_size;
    }

    return $protocol . chr($flags) . pack("(II)<", $datalen, $reserved);
}

my $packet = zbx_create_header(length($data)) . $data;
```

PHP

```
$packet = "ZBXD\1" . pack("VV", strlen($data), 0) . $data;
```

lub

```
function zbx_create_header($plain_data_size, $compressed_data_size = null)
{
    $protocol = "ZBXD";
    $flags = 0x01;
    if (is_null($compressed_data_size))
    {
        $datalen = $plain_data_size;
        $reserved = 0;
    }
    else
    {
        $flags |= 0x02;
```

```
        $datalen = $compressed_data_size;
        $reserved = $plain_data_size;
    }
    return $protocol . chr($flags) . pack("VV", $datalen, $reserved);
}

$packet = zbx_create_header(strlen($data)) . $data;
```

Bash

```
datalen=$(printf "%08x" ${#data})
datalen="\\x${datalen:6:2}\\x${datalen:4:2}\\x${datalen:2:2}\\x${datalen:0:2}"
printf "ZBXD\1${datalen}\0\0\0\0%s" "$data"
```

7 Protokół eksportu JSON rozdzielanego znakami nowej linii

Ta sekcja przedstawia szczegóły protokołu eksportu w formacie JSON rozdzielanym znakami nowej linii, używanego w:

- eksporcie danych do plików
- strumieniowaniu do systemów zewnętrznych

Można eksportować następujące dane:

- zdarzenia wyzwalaczy
- wartości pozycji
- trendy (tylko eksport do plików)

Wszystkie pliki mają rozszerzenie .ndjson. Każda linia pliku eksportu jest obiektem JSON.

Zdarzenia wyzwalaczy

Następujące informacje są eksportowane dla zdarzenia problemu:

Field		Type	Description
clock		number	Liczba sekund od Epoch do momentu wykrycia problemu (część całkowita).
ns		number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas wykrycia problemu.
value		number	1 (zawsze).
eventid		number	ID zdarzenia problemu.
name		string	Nazwa zdarzenia problemu.
severity		number	Poziom ważności zdarzenia problemu (0 - Niesklasyfikowane, 1 - Informacja, 2 - Ostrzeżenie, 3 - Średni, 4 - Wysoki, 5 - Katastrofa).
hosts		array	Lista hostów uwzględnionych w wyrażeniu wyzwalacza; w tablicy powinien znajdować się co najmniej jeden element.
-		object	
	host	string	Nazwa hosta.
	name	string	Widoczna nazwa hosta.
groups		array	Lista grup hostów wszystkich hostów uwzględnionych w wyrażeniu wyzwalacza; w tablicy powinien znajdować się co najmniej jeden element.
-		string	Nazwa grupy hostów.
tags		array	Lista tagów problemu (może być pusta).
-		object	
	tag	string	Nazwa tagu.
	value	string	Wartość tagu (może być pusta).

Następujące informacje są eksportowane dla zdarzenia odzyskania:

Field	Type	Description
clock	number	Liczba sekund od Epoch do momentu rozwiązania problemu (część całkowita).

Field	Type	Description
ns	number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas rozwiązania problemu.
value	number	0 (zawsze).
eventid	number	ID zdarzenia odzyskania.
p_eventid	number	ID zdarzenia problemu.

Przykłady

Problem:

```
{"clock":1519304285,"ns":123456789,"value":1,"name":"Agent Zabbix jest niedostępny na hoście Host B lub po
```

Odzyskanie:

```
{"clock":1519304345,"ns":987654321,"value":0,"eventid":43,"p_eventid":42}
```

Problem (generowanie wielu zdarzeń problemów):

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"Agent Zabbix jest niedostępny na hoście
```

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"Agent Zabbix jest niedostępny na hoście
```

Odzyskanie:

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":43}
```

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":42}
```

Wartości pozycji

Poniższe informacje są eksportowane dla zebranej wartości pozycji:

Field	Type	Description
host	object	Nazwa hosta hosta pozycji.
host	string	Nazwa hosta.
name	string	Widoczna nazwa hosta.
groups	array	Lista grup hosta hosta pozycji; w tablicy powinien znajdować się co najmniej jeden element.
-	string	Nazwa grupy hosta.
item_tags	array	Lista tagów pozycji (może być pusta).
-	object	
tag	string	Nazwa tagu.
value	string	Wartość tagu (może być pusta).
itemid	number	ID pozycji.
name	string	Widoczna nazwa pozycji.
clock	number	Liczba sekund od Epoki do momentu, w którym wartość została zebrana (część całkowita).
ns	number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas zebrania wartości.
timestamp (*Log* only)	number	0, jeśli niedostępne.
source (*Log* only)	string	Pusty ciąg, jeśli niedostępne.
severity (*Log* only)	number	0, jeśli niedostępne.
eventid (*Log* only)	number	0, jeśli niedostępne.
value	number (for numeric items) or string (for text items)	Zebrana wartość pozycji.

Field	Type	Description
type	number	Typ zebranej wartości: 0 - liczba zmiennoprzecinkowa, 1 - znak, 2 - log, 3 - liczba całkowita bez znaku, 4 - tekst, 5 - binarny

Przykłady

Wartość numeryczna (unsigned):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość numeryczna (float):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość znakowa, tekstowa:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość logu:

```
{"host":{"host":"Host A","name":"Host A visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Trendy

Następujące informacje są eksportowane dla obliczonej wartości trendu:

Pole	Typ	Opis
host	object	Nazwa hosta hosta pozycji.
host	string	Nazwa hosta.
name	string	Widoczna nazwa hosta.
groups	array	Lista grup hostów hosta pozycji; w tablicy powinien znajdować się co najmniej jeden element.
-	string	Nazwa grupy hostów.
item_tags	array	Lista tagów pozycji (może być pusta).
-	object	
tag	string	Nazwa tagu.
value	string	Wartość tagu (może być pusta).
itemid	number	ID pozycji.
name	string	Widoczna nazwa pozycji.
clock	number	Liczba sekund od Epoki do momentu zebrania wartości (część całkowita).
count	number	Liczba wartości zebranych dla danej godziny.
min	number	Minimalna wartość pozycji dla danej godziny.
avg	number	Średnia wartość pozycji dla danej godziny.
max	number	Maksymalna wartość pozycji dla danej godziny.
type	number	Typ wartości: 0 - liczba zmiennoprzecinkowa, 3 - liczba całkowita bez znaku

Przykłady

Wartość numeryczna (unsigned):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość numeryczna (float):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

4 Pozycje

Ta sekcja zawiera informacje referencyjne i uwagi techniczne dotyczące pozycji, w tym szczegóły parametrów dla określonych kluczy pozycji, informacje specyficzne dla platform oraz uwagi dotyczące specjalnych konfiguracji pozycji.

Skorzystaj z paska bocznego, aby znaleźć informacje odpowiednie dla Twojego przypadku użycia.

1 parametry vm.memory.size

Przegląd

Ta sekcja zawiera szczegółowe informacje o niektórych parametrach pozycji agent vm.memory.size[<mode>].

Parametry

Dla tej pozycji dostępne są następujące parametry:

- **active** - pamięć aktualnie używana lub używana jeszcze bardzo niedawno, a więc znajdująca się w RAM
- **anon** - pamięć niezwiązana z plikiem (nie można jej z niego ponownie odczytać)
- **available** - dostępna pamięć, obliczana różnie w zależności od platformy (zobacz tabelę poniżej)
- **buffers** - pamięć podręczna dla takich elementów jak metadane systemu plików
- **cached** - pamięć podręczna dla różnych danych
- **exec** - kod wykonywalny, zazwyczaj pochodzący z pliku (programu)
- **file** - pamięć podręczna zawartości ostatnio używanych plików
- **free** - pamięć, która jest od razu dostępna dla każdego podmiotu żądającego pamięci
- **inactive** - pamięć oznaczona jako nieużywana
- **pavailable** - pamięć „available" jako procent wartości „total" (obliczana jako `available/total*100`)
- **pinned** - to samo co „wired"
- **pused** - pamięć „used" jako procent wartości „total" (obliczana jako `used/total*100`)
- **shared** - pamięć, do której jednocześnie może uzyskiwać dostęp wiele procesów
- **slab** - całkowita ilość pamięci używanej przez jądro do buforowania struktur danych na własne potrzeby
- **total** - całkowita dostępna pamięć fizyczna
- **used** - używana pamięć, obliczana różnie w zależności od platformy (zobacz tabelę poniżej)
- **wired** - pamięć oznaczona jako mająca zawsze pozostawać w RAM. Nigdy nie jest przenoszona na dysk.

Warning:
Niektóre z tych parametrów są specyficzne dla platformy i mogą nie być dostępne na Twojej platformie. Szczegóły znajdują się w pozycjach agenta Zabbix.

Obliczanie wartości **available** i **used** zależnie od platformy:

Platforma	**"available"**	**"used"**
AIX	free + cached	rzeczywiście używana pamięć
FreeBSD	inactive + cached + free	active + wired + cached
HP UX	free	total - free
Linux<3.14	free + buffers + cached	total - free
Linux 3.14+ (również przeniesione do wersji 3.10 w RHEL 7)	/proc/meminfo, szczegóły znajdują się w sekcji „MemAvailable" dokumentacji jądra Linux documentation. Zwróć uwagę, że free + buffers + cached nie jest już równe „available", ponieważ nie cała pamięć podręczna stron może zostać zwolniona, a w obliczeniach używany jest niski próg znaku wodnego.	total - free
NetBSD	inactive + execpages + file + free	total - free
OpenBSD	inactive + free + cached	active + wired
OSX	inactive + free	active + wired
Solaris	free	total - free
Win32	free	total - free

Attention:
Suma *vm.memory.size[used]* i *vm.memory.size[available]* nie musi być równa wartości total. Na przykład w FreeBSD:
 * Pamięć active, inactive, wired i cached jest uznawana za używaną, ponieważ przechowuje pewne użyteczne informacje.
 * Jednocześnie pamięć inactive, cached i free jest uznawana za dostępną, ponieważ te rodzaje pamięci mogą zostać natychmiast przydzielone procesom żądającym większej ilości pamięci.
Tak więc pamięć inactive jest jednocześnie używana i dostępna. Z tego powodu pozycja *vm.memory.size[used]* jest przeznaczona wyłącznie do celów informacyjnych, natomiast *vm.memory.size[available]* jest przeznaczona do użycia w wyzwalaczach.

Zobacz także

1. Dodatkowe szczegóły dotyczące obliczania pamięci w różnych Systemach Operacyjnych

2 Pasywne i aktywne kontrole agenta

Przegląd

Ta sekcja zawiera szczegółowe informacje o kontrolach pasywnych i aktywnych wykonywanych przez Zabbix agent i Zabbix agent 2.

Zabbix używa protokołu komunikacyjnego opartego na JSON do komunikacji z agentami.

Protokoły Zabbix agent i Zabbix agent 2 zostały ujednolicone od wersji Zabbix 7.0. Różnica między żądaniami/odpowiedziami Zabbix agent i Zabbix agent 2 jest wyrażona przez wartość znacznika „variant".

Pasywne sprawdzenia

Pasywne sprawdzenie to proste żądanie danych. Serwer Zabbix lub proxy pyta o pewne dane (na przykład obciążenie CPU), a agent Zabbix odsyła wynik do serwera.

Pasywne sprawdzenia są wykonywane asynchronicznie - nie jest wymagane otrzymanie odpowiedzi na jedno żądanie przed rozpoczęciem kolejnych sprawdzeń. Rozwiązywanie DNS również odbywa się asynchronicznie.

Poller agenta będzie próbował połączyć się ze wszystkimi adresami zwróconymi przez wyszukiwanie DNS. Zapewnia to, że jeśli jeden adres IP jest niedostępny, poller spróbuje następnego dostępnego adresu, zwiększając prawdopodobieństwo pomyślnego połączenia. To ulepszenie dotyczy zarówno serwera Zabbix, jak i proxy.

Maksymalna współbieżność asynchronicznych sprawdzeń wynosi 1000 (zdefiniowana przez MaxConcurrentChecksPerPoller).

Liczba asynchronicznych pollerów agenta jest definiowana przez parametr StartAgentPollers.

Żądanie serwera

Definicję nagłówka i długości danych można znaleźć w szczegółach protokołu.

```
{
  "request": "passive checks",
  "data": [
    {
      "key": "agent.version",
      "timeout": 3
    }
  ]
}
```

Field		Type	Mandatory	Value
request		*string*	yes	`"passive checks"`
data		*array of object*	yes	Element sprawdzenia pasywnego.
	key	*string*	yes	Klucz pozycji z rozwiniętymi makrami.
	timeout	*number*	yes	Limit czasu komunikacji.

Odpowiedź agenta

```
{
  "version": "7.4.0",
  "variant": 2,
  "data": [
    {
      "value": "7.4.0"
    }
  ]
}
```

Field	Type	Mandatory	Value
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).

Field	Type	Mandatory	Value
data	*array of object*	yes	Zawiera wynik sprawdzenia.
value	*string*	no	Wartość pozycji, jeśli sprawdzenie zakończyło się powodzeniem.
error	*string*	no	Komunikat błędu, jeśli sprawdzenie nie zakończyło się powodzeniem.

Na przykład dla obsługiwanych pozycji:

1. Serwer otwiera połączenie TCP
2. Serwer wysyła **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”agent.ping”,”timeout”:3}]}**
3. Agent odczytuje żądanie i odpowiada **<HEADER><DATALEN>{”version”:”7.4.0”,”variant”:2,”data”:[{”value”:1}]}**
4. Serwer przetwarza dane, aby uzyskać wartość, w naszym przypadku ’1’
5. Połączenie TCP zostaje zamknięte

Dla nieobsługiwanych pozycji:

1. Serwer otwiera połączenie TCP
2. Serwer wysyła **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”vfs.fs.size[/nono]”,”timeout”:3}]}**
3. Agent odczytuje żądanie i odpowiada **<HEADER><DATALEN>{”version”:”7.4.0”,”variant”:2,”data”:[{”error”:”Unsupported item key.”}]}**
4. Serwer przetwarza dane, zmienia stan pozycji na nieobsługiwany z podanym komunikatem błędu
5. Połączenie TCP zostaje zamknięte

Przełączenie awaryjne na stary protokół

Aby upewnić się, że serwer lub proxy Zabbix mogą współpracować z agentami z wersji wcześniejszych niż 7.2, które używają protokołu tekstowego, zaimplementowano przełączenie awaryjne na stary protokół.

Pasywne kontrole są wykonywane z użyciem protokołu JSON (7.0 i nowsze) po restarcie lub po zmianie konfiguracji interfejsu. Jeśli w odpowiedzi nie zostanie odebrany prawidłowy JSON (agent wysłał „ZBX_NOTSUPPORTED”), Zabbix zapisze interfejs w pamięci podręcznej jako używający starego protokołu i **ponowi** kontrolę, wysyłając tylko klucz pozycji.

Należy pamiętać, że co godzinę serwer/proxy Zabbix ponownie spróbuje pracować z nowym protokołem na wszystkich interfejsach, w razie potrzeby przełączając się z powrotem na stary protokół.

Kontrole aktywne

Kontrole aktywne wymagają bardziej złożonego przetwarzania. agent musi najpierw pobrać z serwera/proxy listę pozycji i/lub zdalnych poleceń do niezależnego przetwarzania.

Serwery/proxy, z których mają być pobierane kontrole aktywne, są wymienione w parametrze ’ServerActive’ w pliku konfiguracyjnym agenta. Częstotliwość odpytywania o te kontrole jest ustawiana przez parametr ’RefreshActiveChecks’ w tym samym pliku konfiguracyjnym. Jeśli jednak odświeżenie kontroli aktywnych się nie powiedzie, ponowna próba nastąpi po zakodowanych na stałe 60 sekundach.

Note:
Od wersji Zabbix 6.4 agent (w trybie aktywnym) nie otrzymuje już od serwera/proxy pełnej kopii konfiguracji co dwie minuty (domyślnie). Zamiast tego, aby zmniejszyć ruch sieciowy i zużycie zasobów, co 5 sekund (domyślnie) wykonywana jest przyrostowa synchronizacja konfiguracji, podczas której serwer/proxy dostarcza pełną kopię konfiguracji **tylko** wtedy, gdy agent jeszcze jej nie otrzymał lub gdy coś zmieniło się w konfiguracji hosta, makrach globalnych albo globalnych wyrażeniach regularnych.

Agent następnie okresowo wysyła nowe wartości do serwera(ów). Jeśli agent otrzymał jakiekolwiek zdalne polecenia do wykonania, wynik ich wykonania również zostanie wysłany. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0.

Note:
Jeśli agent znajduje się za zaporą sieciową, warto rozważyć używanie wyłącznie aktywnych kontroli, ponieważ w takim przypadku nie trzeba modyfikować zapory, aby zezwolić na początkowe połączenia przychodzące.

Pobieranie listy pozycji

Żądanie agenta

Żądanie aktywnych kontroli służy do pobrania aktywnych kontroli, które mają zostać przetworzone przez agenta.
To żądanie jest wysyłane przez agenta podczas uruchamiania, a następnie w odstępach RefreshActiveChecks.

```
{
  "request": "active checks",
  "host": "Zabbix server",
  "host_metadata": "mysql,nginx",
  "interface": "zabbix.server.lan",
  "ip": "159.168.1.1",
  "port": 12050,
  "version": "7.4.0",
  "variant": 2,
  "config_revision": 1,
  "session": "e3dcbd9ace2c9694e1d7bbd030eeef6e"
}
```

Field	Type	Mandatory	Value
request	*string*	yes	`active checks`
host	*string*	yes	Nazwa hosta.
host_metadata	*string*	no	Parametr konfiguracyjny HostMetadata lub wartość metryki HostMetadataItem.
interface	*string*	no	Parametr konfiguracyjny HostInterface lub wartość metryki HostInterfaceItem.
ip	*string*	no	Pierwszy adres IP z parametru konfiguracyjnego ListenIP, jeśli jest ustawiony.
port	*number*	no	Wartość parametru konfiguracyjnego ListenPort, jeśli jest ustawiona i nie jest domyślnym portem nasłuchu agenta.
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).
config_revision	*number*	no	Identyfikator konfiguracji dla inkrementalnej synchronizacji konfiguracji.
session	*string*	no	Identyfikator sesji dla inkrementalnej synchronizacji konfiguracji.

Odpowiedź serwera

Odpowiedź aktywnych kontroli jest wysyłana przez serwer z powrotem do agenta po przetworzeniu żądania aktywnych kontroli.

```
{
  "response": "success",
  "config_revision": 2,
  "data": [
    {
      "key": "system.uptime",
      "itemid": 1234,
      "delay": "10s",
      "lastlogsize": 0,
      "mtime": 0
    },
    {
      "key": "agent.version",
      "itemid": 5678,
      "delay": "10m",
      "lastlogsize": 0,
      "mtime": 0,
      "timeout": "30s"
    }
  ],
  "commands": [
    {
      "command": "df -h --output=source,size / | awk 'NR>1 {print $2}'",
      "id": 1324,
      "wait": 1
    }
  ]
}
```

Field	Type	Mandatory	Value
response	*string*	yes	`success` | `failed`

Field	Type	Mandatory	Value
info	*string*	no	Informacje o błędzie w przypadku niepowodzenia.
data	*array of objects*	no	Pozycje aktywnych kontroli. Pomijane, jeśli konfiguracja hosta nie uległa zmianie.
key	*string*	no	Klucz pozycji z rozwiniętymi makrami.
itemid	*number*	no	Identyfikator pozycji.
delay	*string*	no	Interwał aktualizacji pozycji. Elastyczne/interwały harmonogramu są obsługiwane zarówno przez Zabbix agent, jak i Zabbix agent 2 od Zabbix 7.0.
lastlogsize	*number*	no	lastlogsize pozycji.
mtime	*number*	no	mtime pozycji.
timeout	*string*	no	Limit czasu pozycji.
refresh_unsupported	*number*	no	Interwał odświeżania nieobsługiwanej pozycji.
regexp	*array of objects*	no	Globalne wyrażenia regularne.
name	*string*	no	Nazwa globalnego wyrażenia regularnego.
expression	*string*	no	Globalne wyrażenie regularne.
expression_type	*number*	no	Typ globalnego wyrażenia regularnego.
exp_delimiter	*string*	no	Separator globalnego wyrażenia regularnego.
case_sensitive	*number*	no	Ustawienie rozróżniania wielkości liter dla globalnego wyrażenia regularnego.
commands	*array of objects*	no	Zdalne polecenia do wykonania. Dołączane, jeśli wykonanie zdalnego polecenia zostało wyzwolone przez operację action lub ręczne wykonanie script. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0. Starsze aktywne agenty zignorują wszystkie zdalne polecenia zawarte w odpowiedzi serwera aktywnych kontroli.
command	*string*	no	Zdalne polecenie.
id	*number*	no	Identyfikator zdalnego polecenia.
wait	*number*	no	Tryb wykonania zdalnego polecenia (”0” (nowait) dla poleceń z operacji action; ”1” (wait) dla poleceń z ręcznego wykonania script).
timeout	*number*	no	Limit czasu wykonania zdalnego polecenia w konfiguracji serwer/proxy.
config_revision	*number*	no	Identyfikator konfiguracji dla inkrementalnej synchronizacji konfiguracji. Pomijane, jeśli konfiguracja hosta nie uległa zmianie. Zwiększane, jeśli konfiguracja hosta została zmieniona.

Serwer musi odpowiedzieć sukcesem.

Na przykład:

1. Agent otwiera połączenie TCP
2. Agent prosi o listę kontroli
3. Serwer odpowiada listą pozycji i zdalnych poleceń do wykonania
4. Agent analizuje odpowiedź
5. Połączenie TCP zostaje zamknięte
6. Agent rozpoczyna okresowe zbieranie danych i wykonuje zdalne polecenia (obsługiwane od Zabbix agent 7.0)

Attention:
Należy pamiętać, że (wrażliwe) dane konfiguracyjne mogą stać się dostępne dla stron mających dostęp do portu trappera serwera Zabbix podczas korzystania z aktywnej kontroli. Jest to możliwe, ponieważ każdy może podszyć się pod aktywnego agenta i zażądać danych konfiguracyjnych pozycji; uwierzytelnianie nie jest przeprowadzane, chyba że użyjesz opcji encryption.

Wysyłanie zebranych danych

Agent wysyła

Żądanie danych agenta zawiera zebrane wartości pozycji oraz wartości wykonanych zdalnych poleceń (jeśli występują).

```
{
  "request": "agent data",
  "data": [
    {
```

```
      "id": 1,
      "itemid": 5678,
      "value": "7.0.0",
      "clock": 1712830783,
      "ns": 76808644
    },
    {
      "id": 2,
      "itemid": 1234,
      "value": "69672",
      "clock": 1712830783,
      "ns": 77053975
    }
  ],
  "commands": [
    {
      "id": 1324,
      "value": "16G"
    }
  ],
  "session": "8495cd52070e6ca52b371f29c8574165",
  "host": "Zabbix server",
  "version": "7.4.0",
  "variant": 2
}
```


Field	Type	Mandatory	Value
request	*string*	yes	`agent data`
data	*array of objects*	yes	Wartości pozycji.
id	*number*	yes	Identyfikator wartości (licznik inkrementowany, używany do sprawdzania zduplikowanych wartości w przypadku problemów sieciowych).
itemid	*number*	yes	Identyfikator pozycji.
value	*string*	no	Wartość pozycji.
lastlogsize	*number*	no	lastlogsize pozycji.
mtime	*number*	no	mtime pozycji.
state	*number*	no	Stan pozycji.
source	*string*	no	Źródło dziennika zdarzeń wartości.
eventid	*number*	no	eventid dziennika zdarzeń wartości.
severity	*number*	no	severity dziennika zdarzeń wartości.
timestamp	*number*	no	timestamp dziennika zdarzeń wartości.
clock	*number*	yes	Znacznik czasu wartości (sekundy od początku ery Unix).
ns	*number*	yes	Nanosekundy znacznika czasu wartości.
commands	*array of objects*	no	Wynik wykonania zdalnych poleceń. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0. Starsze aktywne agenty zignorują wszystkie zdalne polecenia zawarte w odpowiedzi serwera dla aktywnych kontroli.
id	*number*	no	Identyfikator zdalnego polecenia.
value	*string*	no	Wynik wykonania zdalnego polecenia, jeśli wykonanie zakończyło się powodzeniem.
error	*string*	no	Komunikat błędu wykonania zdalnego polecenia, jeśli wykonanie nie powiodło się.
session	*string*	yes	Unikalny identyfikator sesji generowany za każdym razem po uruchomieniu agenta.
host	*string*	yes	Nazwa hosta.
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).

Każdej wartości przypisywany jest wirtualny identyfikator. Identyfikator wartości jest prostym rosnącym licznikiem, unikalnym w ramach jednej sesji danych (identyfikowanej przez token sesji). Ten identyfikator służy do odrzucania zduplikowanych wartości, które mogą być wysyłane w środowiskach o słabej łączności.

Odpowiedź serwera

Odpowiedź na dane agenta jest wysyłana przez serwer z powrotem do agenta po przetworzeniu żądania danych agenta.

```
{
  "response": "success",
  "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.003534"
}
```

Field	Type	Mandatory	Value
response	*string*	yes	`success` | `failed`
info	*string*	yes	Wyniki przetwarzania pozycji.

Attention:
Jeśli wysyłanie niektórych wartości nie powiedzie się na serwerze (na przykład dlatego, że host lub pozycja zostały wyłączone albo usunięte), agent nie będzie ponawiać wysyłania tych wartości.

Na przykład:

1. Agent otwiera połączenie TCP
2. Agent wysyła listę wartości
3. Serwer przetwarza dane i odsyła status
4. Połączenie TCP zostaje zamknięte

Attention:
Komunikat błędu zostanie po stronie serwera skrócony do 2048 znaków.

Komunikat heartbeat

Wysyłany przez agent

Komunikat heartbeat jest wysyłany przez aktywny agent do serwer/proxy Zabbix co HeartbeatFrequency sekund (skonfigurowane w pliku konfiguracyjnym Zabbix agent/ agent 2).

Służy do monitorowania dostępności aktywnych kontroli.

```
{
  "request": "active check heartbeat",
  "host": "Zabbix server",
  "heartbeat_freq": 60,
  "version": "7.4.0",
  "variant": 2
}
```

Field	Type	Mandatory	Value
request	*string*	yes	`active check heartbeat`
host	*string*	yes	Nazwa hosta.
heartbeat_freq	*number*	yes	Częstotliwość heartbeat agenta (parametr konfiguracyjny HeartbeatFrequency).
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).

Odpowiedź przekierowania

Gdy host został przypisany ponownie, serwer może polecić agentowi przekierowanie jego heartbeat (oraz kolejnych aktywnych kontroli) do innej instancji proxy lub serwer.

```
  {
    "response": "failed",
    "redirect": {
```

```
    "revision": 2,
    "address": "192.0.2.1:10055"
  }
}
```

Field		Type	Mandatory	Value
response		*string*	yes	`success` | `failed`
redirect		*object*	yes	Instrukcje przekierowania.
	revision	*number*	yes	Identyfikator rewizji konfiguracji.
	address	*string*	yes	Adres docelowego serwer/proxy.

Starszy protokół XML

Note:
Zabbix przyjmie do 16 MB danych XML zakodowanych w Base64, ale pojedyncza zdekodowana wartość nie powinna być dłuższa niż 64 KB, w przeciwnym razie zostanie obcięta do 64 KB podczas dekodowania.

3 Minimalny poziom uprawnień dla pozycji agenta Windows

Przegląd

Podczas monitorowania systemów przy użyciu agent, dobrą praktyką jest pozyskiwanie metryk z host, na którym agent jest zainstalowany. Aby stosować zasadę najmniejszych uprawnień, konieczne jest określenie, jakie metryki są pozyskiwane przez agent.

Tabela w tym dokumencie umożliwia wybór minimalnych uprawnień gwarantujących poprawne działanie Zabbix agent.

Jeśli do działania agent wybrano innego użytkownika niż „LocalSystem”, to aby agent działał jako usługa systemu Windows, nowy użytkownik musi mieć uprawnienie „Log on as a service” w „Local Policy→User Rights Assignment” oraz prawo do tworzenia, zapisywania i usuwania pliku dziennika Zabbix agent. Użytkownik Active Directory musi zostać dodany do grupy *Performance Monitor Users*.

Note:
W przypadku pracy z uprawnieniami agent opartymi na grupie „minimalnie technicznie akceptowalnej” wymagane jest wcześniejsze nadanie uprawnień do obiektów przeznaczonych do monitorowania.

Typowe pozycje agenta obsługiwane w systemie Windows

Klucz pozycji	Grupa użytkowników	
	Zalecane	Minimalne technicznie akceptowalne (funkcjonalność jest ograniczona)
agent.hostname	Goście	Goście
agent.ping	Goście	Goście
agent.variant	Goście	Goście
agent.version	Goście	Goście
log	Administratorzy	Goście
log.count	Administratorzy	Goście
logrt	Administratorzy	Goście
logrt.count	Administratorzy	Goście
net.dns	Goście	Goście
net.dns.perf	Goście	Goście
net.dns.record	Goście	Goście
net.if.discovery	Goście	Goście

Klucz pozycji	Grupa użytkowników	
net.if.in	Goście	Goście
net.if.out	Goście	Goście
net.if.total	Goście	Goście
net.tcp.listen	Goście	Goście
net.tcp.port	Goście	Goście
net.tcp.service	Goście	Goście
net.tcp.service.perf	Goście	Goście
net.udp.service	Goście	Goście
net.udp.service.perf	Goście	Goście
proc.num	Administratorzy	Goście
system.cpu.discovery	Użytkownicy monitora wydajności	Użytkownicy monitora wydajności
system.cpu.load	Użytkownicy monitora wydajności	Użytkownicy monitora wydajności
system.cpu.num	Goście	Goście
system.cpu.util	Użytkownicy monitora wydajności	Użytkownicy monitora wydajności
system.hostname	Goście	Goście
system.localtime	Goście	Goście
system.run	Administratorzy	Goście
system.sw.arch	Goście	Goście
system.swap.size	Goście	Goście
system.uname	Goście	Goście
system.uptime	Użytkownicy monitora wydajności	Użytkownicy monitora wydajności
vfs.dir.count	Administratorzy	Goście
vfs.dir.get	Administratorzy	Goście
vfs.dir.size	Administratorzy	Goście
vfs.file.cksum	Administratorzy	Goście
vfs.file.contents	Administratorzy	Goście
vfs.file.exists	Administratorzy	Goście
vfs.file.md5sum	Administratorzy	Goście
vfs.file.regexp	Administratorzy	Goście
vfs.file.regmatch	Administratorzy	Goście
vfs.file.size	Administratorzy	Goście
vfs.file.time	Administratorzy	Goście
vfs.fs.discovery	Administratorzy	Goście
vfs.fs.size	Administratorzy	Goście
vfs.fs.get	Administratorzy	Goście
vm.memory.size	Goście	Goście
web.page.get	Goście	Goście
web.page.perf	Goście	Goście
web.page.regexp	Goście	Goście
zabbix.stats	Goście	Goście

Klucze pozycji specyficzne dla systemu Windows

Klucz pozycji	Grupa użytkowników	
	Zalecane	Minimalne technicznie akceptowalne (funkcjonalność jest ograniczona)
eventlog	Czytelnicy dziennika zdarzeń	Goście
net.if.list	Goście	Goście
perf_counter	Użytkownicy monitora wydajności	Użytkownicy monitora wydajności
proc_info	Administratorzy	Goście
service.discovery	Goście	Goście
service.info	Goście	Goście
services	Goście	Goście
wmi.get	Administratorzy	Goście
vm.vmemory.size	Goście	Goście

4 Kodowanie zwracanych wartości

Serwer Zabbix oczekuje, że każda zwracana wartość tekstowa będzie w kodowaniu UTF8. Dotyczy to wszystkich typów kontroli: Zabbix agent, SSH, Telnet itd.

Różne monitorowane systemy/urządzenia i kontrole mogą zwracać w wartości znaki spoza ASCII. W takich przypadkach prawie wszystkie możliwe klucze **zabbix** zawierają dodatkowy parametr klucza pozycji - **<encoding>**. Ten parametr klucza jest opcjonalny, ale należy go określić, jeśli zwracana wartość nie jest w kodowaniu UTF8 i zawiera znaki spoza ASCII. W przeciwnym razie wynik może być nieoczekiwany i nieprzewidywalny.

Poniżej znajduje się opis zachowania w takich przypadkach dla różnych backendów baz danych.

MySQL

Jeśli wartość zawiera znak spoza ASCII w kodowaniu innym niż UTF8, ten znak i kolejne zostaną odrzucone podczas zapisywania tej wartości w bazie danych. Żadne komunikaty ostrzegawcze nie zostaną zapisane w pliku *zabbix_server.log*.
 Dotyczy co najmniej wersji MySQL 5.1.61.

PostgreSQL

Jeśli wartość zawiera znak spoza ASCII w kodowaniu innym niż UTF8, doprowadzi to do niepowodzenia zapytania SQL (PGRES_FATAL_ERROR:ERROR invalid byte sequence for encoding), a dane nie zostaną zapisane. Odpowiedni komunikat ostrzegawczy zostanie zapisany w pliku *zabbix_server.log*.
 Dotyczy co najmniej PostgreSQL w wersji 9.1.3.

5 Obsługa dużych plików

Obsługa dużych plików, często skracana do LFS, to termin odnoszący się do możliwości pracy z plikami większymi niż 2 GB w 32-bitowych systemach operacyjnych. Obsługa dużych plików wpływa co najmniej na monitorowanie plików dziennika oraz wszystkie pozycje vfs.file.*. Obsługa dużych plików zależy od możliwości systemu w momencie kompilacji Zabbix, ale jest całkowicie wyłączona w 32-bitowym Solarisie z powodu niezgodności z procfs i swapctl.

6 Czujnik

Przegląd

Każdy układ czujnika ma własny katalog w drzewie sysfs `/sys/devices`. Aby znaleźć wszystkie układy czujników, łatwiej jest śledzić dowiązania symboliczne urządzeń z `/sys/class/hwmon/hwmon*`, gdzie * jest liczbą rzeczywistą (0,1,2,...).

Odczyty czujników znajdują się albo w katalogu `/sys/class/hwmon/hwmon*/` dla urządzeń wirtualnych, albo w katalogu `/sys/class/hwmon/hwmon*/device` dla urządzeń niewirtualnych. Plik o nazwie `name`, znajdujący się w katalogach `hwmon*` lub `hwmon*/device`, zawiera nazwę układu, która odpowiada nazwie sterownika jądra używanego przez układ czujnika.

W każdym pliku znajduje się tylko jedna wartość odczytu czujnika. Wspólny schemat nazewnictwa plików zawierających odczyty czujników wewnątrz dowolnego z wyżej wymienionych katalogów to: `<type><number>_<item>`, gdzie

- **type** - dla układów czujników jest to ”in” (napięcie), ”temp” (temperatura), ”fan” (wentylator) itd.
- **item** - ”input” (wartość zmierzona), ”max” (górny próg), ”min” (dolny próg) itd.
- **number** - zawsze używany dla elementów, które mogą występować więcej niż raz (zwykle zaczyna się od 1, z wyjątkiem napięć, które zaczynają się od 0). Jeśli pliki nie odnoszą się do konkretnego elementu, mają prostą nazwę bez numeru.

Informacje dotyczące czujników dostępnych na hoście można uzyskać za pomocą narzędzi **sensors-detect** i **sensors** (pakiet lm-sensors). **Sensors-detect** pomaga określić, które moduły są wymagane dla dostępnych czujników. Po załadowaniu modułów program **sensors** może zostać użyty do wyświetlenia odczytów wszystkich układów czujników. Etykiety odczytów czujników używane przez ten program mogą różnić się od wspólnego schematu nazewnictwa (`<type><number>_<item>`):

- jeśli istnieje plik o nazwie `<type><number>_label`, wtedy etykieta z tego pliku zostanie użyta zamiast nazwy `<type><number><item>`;
- jeśli nie ma pliku `<type><number>_label`, program przeszukuje `/etc/sensors.conf` (może to być także `/etc/sensors3.conf` lub inny plik) pod kątem podstawienia nazwy.

To etykietowanie pozwala użytkownikowi określić, jaki rodzaj sprzętu jest używany. Jeśli nie ma ani pliku `<type><number>_label`, ani etykiety w pliku konfiguracyjnym, typ sprzętu można określić na podstawie atrybutu name (`hwmon*/device/name`). Rzeczywiste nazwy czujników akceptowane przez `zabbix_agent` można uzyskać, uruchamiając program **sensors** z parametrem `-u` (`sensors -u`).

W programie **sensors** dostępne czujniki są rozdzielone według typu magistrali (adapter ISA, adapter PCI, adapter SPI, urządzenie wirtualne, interfejs ACPI, adapter HID).

Uzyskiwanie identyfikatorów sensorów

Etykiety sensorów, wyświetlane przez polecenie `sensors`, nie zawsze mogą być używane bezpośrednio, ponieważ sposób etykietowania różni się w zależności od producenta układu. Na przykład dane wyjściowe `sensors` mogą zawierać następujące wiersze:

```
sensors
in0:          +2.24 V  (min =  +0.00 V, max =  +3.32 V)
Vcore:        +1.15 V  (min =  +0.00 V, max =  +2.99 V)
+3.3V:        +3.30 V  (min =  +2.97 V, max =  +3.63 V)
+12V:        +13.00 V  (min =  +0.00 V, max = +15.94 V)
M/B Temp:     +30.0°C  (low  = -127.0°C, high = +127.0°C)
```

Spośród nich tylko jedna nazwa wyświetlana odpowiada bezpośrednio identyfikatorowi wewnętrznemu:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in0]
2.240000
```

Próba użycia innych etykiet wyświetlanych (takich jak `Vcore` lub `+12V`) nie zadziała:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,Vcore]
ZBX_NOTSUPPORTED
```

Aby znaleźć wewnętrzny identyfikator sensora akceptowany przez Zabbix, uruchom `sensors -u`. W danych wyjściowych można zaobserwować następujące elementy:

```
sensors -u
...
Vcore:
  in1_input: 1.15
  in1_min: 0.00
  in1_max: 2.99
  in1_alarm: 0.00
...
+12V:
  in4_input: 13.00
  in4_min: 0.00
  in4_max: 15.94
  in4_alarm: 0.00
...
```

Zatem etykieta Vcore odpowiada identyfikatorowi in1, a +12V odpowiada in4. Zgodnie ze specyfikacją kernel hwmon sysfs są to napięcia na pinach układu i ogólnie rzecz biorąc mogą wymagać skalowania.

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in1]
1.301000
```

Zabbix może pobierać nie tylko odczyty napięcia (in), ale także natężenia prądu (curr), temperatury (temp) i prędkości wentylatora (fan).

Zgodność

Starsze dystrybucje

Odczyty czujników są pobierane z katalogu /proc/sys/dev/sensors:

- **device** - nazwa urządzenia (jeśli użyto <mode>, jest to wyrażenie regularne)
- **sensor** - nazwa czujnika (jeśli użyto <mode>, jest to wyrażenie regularne)
- **mode** - możliwe wartości: avg, max, min (jeśli ten parametr zostanie pominięty, device i sensor są traktowane dosłownie)

Przykładowy klucz: sensor[w83781d-i2c-0-2d,temp1].

Bieżące dystrybucje

Odczyty sensorów są pobierane z katalogu /sys/class/hwmon:

- **device** - nazwa urządzenia (bez wyrażeń regularnych). Nazwa urządzenia może być rzeczywistą nazwą urządzenia (np. 0000:00:18.3) lub nazwą uzyskaną za pomocą programu sensors (np. k8temp-pci-00c3). Wybór, której nazwy użyć, należy do użytkownika.
- **sensor** - nazwa sensora (bez wyrażeń regularnych).
- **mode** - możliwe wartości: avg, max, min (jeśli ten parametr zostanie pominięty, device i sensor są traktowane dosłownie).

Przykładowe klucze:

sensor[k8temp-pci-00c3,temp,max] lub sensor[0000:00:18.3,temp1].

sensor[smsc47b397-isa-0880,in,avg] lub sensor[smsc47b397.2176,in1].

7 Uwagi dotyczące parametru memtype w pozycjach proc.mem

Przegląd

Parametr **memtype** jest obsługiwany na platformach Linux, AIX, FreeBSD i Solaris.

Trzy typowe wartości parametru „memtype" są obsługiwane na wszystkich tych platformach: pmem, rss i vsize. Dodatkowo na niektórych platformach obsługiwane są specyficzne dla platformy wartości parametru „memtype".

AIX

Zobacz wartości obsługiwane dla parametru „memtype" w systemie AIX w tabeli.

Obsługiwana wartość	Opis	Źródło w strukturze procentry64	Próbuje zachować zgodność z
vsize [1]	Rozmiar pamięci wirtualnej	pi_size	
pmem	Procent pamięci rzeczywistej	pi_prm	ps -o pmem
rss	Rozmiar rezydentnego zestawu pamięci	pi_trss + pi_drss	ps -o rssize
size	Rozmiar procesu (kod + dane)	pi_dvm	"ps gvw" kolumna SIZE
dsize	Rozmiar danych	pi_dsize	
tsize	Rozmiar tekstu (kodu)	pi_tsize	"ps gvw" kolumna TSIZ
sdsize	Rozmiar danych z biblioteki współdzielonej	pi_sdsize	
drss	Rozmiar rezydentnego zestawu danych	pi_drss	
trss	Rozmiar rezydentnego zestawu tekstu	pi_trss	

Uwagi dotyczące AIX:

1. Wybierając parametry dla klucza pozycji proc.mem[] w AIX, postaraj się określić wąskie kryteria wyboru procesów. W przeciwnym razie istnieje ryzyko, że do wyniku proc.mem[] zostaną wliczone niepożądane procesy.

Przykład:

```
$ zabbix_agentd -t proc.mem[,,,NonExistingProcess,rss]
proc.mem[,,,NonExistingProcess,rss]           [u|2879488]
```

Ten przykład pokazuje, że podanie tylko parametru wiersza poleceń (wyrażenia regularnego do dopasowania) powoduje samouzględnianie przez Zabbix agent — prawdopodobnie nie jest to oczekiwane zachowanie.

2. Nie używaj `ps -ef` do przeglądania procesów — pokazuje ono tylko procesy niejądrowe. Użyj `ps -Af`, aby zobaczyć wszystkie procesy, które będą widoczne dla Zabbix agent.
3. Przejdźmy przez przykład `topasrec`, aby zobaczyć, jak Zabbix agent `proc.mem[]` wybiera procesy.

```
$ 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[]` ma argumenty:

```
proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]
```

Pierwszym kryterium jest nazwa procesu (argument <name>). W naszym przykładzie agent Zabbix zobaczy ją jako 'topasrec'. Aby dopasowanie zadziałało, należy albo podać 'topasrec', albo pozostawić to pole puste. Drugim kryterium jest nazwa użytkownika (argument <user>). Aby dopasowanie zadziałało, należy albo podać 'root', albo pozostawić to pole puste. Trzecim kryterium używanym przy wyborze procesu jest argument <cmdline>. agent Zabbix zobaczy jego wartość jako '/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'. Aby dopasowanie zadziałało, należy albo podać wyrażenie regularne pasujące do tego ciągu, albo pozostawić to pole puste.

Argumenty <mode> i <memtype> są stosowane po użyciu trzech kryteriów wymienionych powyżej.

FreeBSD

Zobacz wartości obsługiwane dla parametru 'memtype' w systemie FreeBSD w tabeli.

Obsługiwana wartość	Opis	Źródło w strukturze kinfo_proc	Próbuje zachować zgodność z
vsize	Rozmiar pamięci wirtualnej	kp_eproc.e_vm.vm_map_size lub ki_size	ps -o vsz
pmem	Procent pamięci rzeczywistej	obliczany na podstawie rss	ps -o pmem
rss	Rozmiar zestawu rezydentnego	kp_eproc.e_vm.vm_rssize lub ki_rssize	ps -o rss
size [1]	Rozmiar procesu (kod + dane + stos)	tsize + dsize + ssize	
tsize	Rozmiar tekstu (kodu)	kp_eproc.e_vm.vm_tsize lub ki_tsize	ps -o tsiz
dsize	Rozmiar danych	kp_eproc.e_vm.vm_dsize lub ki_dsize	ps -o dsiz
ssize	Rozmiar stosu	kp_eproc.e_vm.vm_ssize lub ki_ssize	ps -o ssiz

Linux

Obsługiwane wartości parametru „memtype" w systemie Linux przedstawiono w tabeli.

Obsługiwana wartość	Opis	Źródło w pliku /proc/<pid>/status
vsize [1]	Rozmiar pamięci wirtualnej	VmSize
pmem	Procent pamięci rzeczywistej	(VmRSS/total_memory) * 100
rss	Rozmiar zestawu rezydentnego	VmRSS
data	Rozmiar segmentu danych	VmData
exe	Rozmiar segmentu kodu	VmExe
hwm	Szczytowy rozmiar zestawu rezydentnego	VmHWM
lck	Rozmiar zablokowanej pamięci	VmLck
lib	Rozmiar bibliotek współdzielonych	VmLib
peak	Szczytowy rozmiar pamięci wirtualnej	VmPeak
pin	Rozmiar przypiętych stron	VmPin
pte	Rozmiar wpisów tablic stron	VmPTE

Obsługiwana wartość	Opis	Źródło w pliku /proc/<pid>/status
size	Rozmiar segmentów kodu + danych + stosu procesu	VmExe + VmData + VmStk
stk	Rozmiar segmentu stosu	VmStk
swap	Rozmiar używanej przestrzeni wymiany	VmSwap

Uwagi dotyczące systemu Linux:

1. Nie wszystkie wartości „memtype” są obsługiwane przez starsze jądra Linux. Na przykład jądra Linux 2.4 nie obsługują wartości `hwm`, `pin`, `peak`, `pte` i `swap`.
2. Zauważyliśmy, że samomonitorowanie procesu aktywnego sprawdzania agenta Zabbix za pomocą `proc.mem[...,...,...,...,dat` pokazuje wartość o 4 kB większą niż raportowana przez wiersz `VmData` w pliku /proc/<pid>/status agenta. W momencie samopomiaru segment danych agenta zwiększa się o 4 kB, a następnie wraca do poprzedniego rozmiaru.

Solaris

Zobacz wartości obsługiwane dla parametru „memtype” w systemie Solaris w tabeli.

Obsługiwana wartość	Opis	Źródło w strukturze psinfo	Próbuje zachować zgodność z
vsize [1]	Rozmiar obrazu procesu	pr_size	ps -o vsz
pmem	Procent pamięci rzeczywistej	pr_pctmem	ps -o pmem
rss	Rozmiar rezydentnego zestawu stron Może być zaniżony — zobacz opis rss w „man ps”.	pr_rssize	ps -o rss

Przypisy

[1] Wartość domyślna.

8 Uwagi dotyczące wybierania procesów w pozycjach proc.mem i proc.num

Procesy modyfikujące swoją linię poleceń

Niektóre programy wykorzystują modyfikowanie swojej linii poleceń jako metodę wyświetlania bieżącej aktywności. Użytkownik może zobaczyć tę aktywność, uruchamiając polecenia `ps` i `top`. Przykładami takich programów są *PostgreSQL*, *Sendmail*, *Zabbix*.

Zobaczmy przykład z Linuksa. Załóżmy, że chcemy monitorować liczbę procesów Zabbix agent.

Polecenie `ps` pokazuje interesujące nas procesy jako

```
$ 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/zabbix_agentd.conf
zabbix    6319  6318  0 12:01 ?        00:00:01 sbin/zabbix_agentd: collector [idle 1 sec]
zabbix    6320  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #1 [waiting for connection]
zabbix    6321  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #2 [waiting for connection]
zabbix    6322  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: listener #3 [waiting for connection]
zabbix    6323  6318  0 12:01 ?        00:00:00 sbin/zabbix_agentd: active checks #1 [idle 1 sec]
...
```

Wybieranie procesów według nazwy i użytkownika działa:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd,zabbix]'
6
```

Teraz zmieńmy nazwę pliku wykonywalnego `zabbix_agentd` na `zabbix_agentd_30` i uruchommy go ponownie.

`ps` pokazuje teraz

```
$ 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/zabbix_agentd.conf
zabbix    6716  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: collector [idle 1 sec]
```

```
zabbix    6717  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #1 [waiting for connection
zabbix    6718  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #2 [waiting for connection
zabbix    6719  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: listener #3 [waiting for connection
zabbix    6720  6715  0 12:53 ?        00:00:00 sbin/zabbix_agentd_30: active checks #1 [idle 1 sec]
...
```

Teraz wybieranie procesów według nazwy i użytkownika daje niepoprawny wynik:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd_30,zabbix]'
1
```

Dlaczego zwykła zmiana nazwy pliku wykonywalnego na dłuższą prowadzi do tak odmiennego wyniku?

Zabbix agent zaczyna od sprawdzenia nazwy procesu. Otwierany jest plik `/proc/<pid>/status` i sprawdzany jest wiersz `Name`. W naszym przypadku wiersze `Name` są następujące:

```
$ grep Name /proc/{6715,6716,6717,6718,6719,6720}/status
/proc/6715/status:Name:   zabbix_agentd_3
/proc/6716/status:Name:   zabbix_agentd_3
/proc/6717/status:Name:   zabbix_agentd_3
/proc/6718/status:Name:   zabbix_agentd_3
/proc/6719/status:Name:   zabbix_agentd_3
/proc/6720/status:Name:   zabbix_agentd_3
```

Nazwa procesu w pliku `status` jest obcinana do 15 znaków.

Podobny wynik można zobaczyć przy użyciu polecenia `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
 ...
```

Oczywiście nie jest to równe wartości parametru `name` `proc.num[]`, czyli `zabbix_agentd_30`. Po nieudanej próbie dopasowania nazwy procesu z pliku `status` Zabbix agent przechodzi do pliku `/proc/<pid>/cmdline`.

To, jak agent widzi plik „cmdline", można zilustrować, uruchamiając polecenie

```
$ for i in 6715 6716 6717 6718 6719 6720; do cat /proc/$i/cmdline | awk '{gsub(/\x0/,"<NUL>"); print};'; d
sbin/zabbix_agentd_30<NUL>-c<NUL>/home/zabbix/zabbix_agentd.conf<NUL>
sbin/zabbix_agentd_30: collector [idle 1 sec]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><
sbin/zabbix_agentd_30: listener #1 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: listener #2 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: listener #3 [waiting for connection]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><N
sbin/zabbix_agentd_30: active checks #1 [idle 1 sec]<NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL><NUL
```

Pliki `/proc/<pid>/cmdline` w naszym przypadku zawierają niewidoczne, niedrukowalne bajty null, używane do kończenia łańcuchów w języku *C*. W tym przykładzie bajty null są pokazane jako „<NUL>".

Zabbix agent sprawdza „cmdline" dla procesu głównego i pobiera `zabbix_agentd_30`, co odpowiada wartości naszego parametru `name` `zabbix_agentd_30`. Dlatego proces główny jest zliczany przez pozycja `proc.num[zabbix_agentd_30,zabbix]`.

Podczas sprawdzania kolejnego procesu agent pobiera `zabbix_agentd_30: collector [idle 1 sec]` z pliku `cmdline` i nie spełnia to warunku naszego parametru `name` `zabbix_agentd_30`. Zatem zliczany jest tylko proces główny, który nie modyfikuje swojej linii poleceń. Pozostałe procesy agent modyfikują swoją linię poleceń i są ignorowane.

Ten przykład pokazuje, że parametr `name` nie może być używany w `proc.mem[]` i `proc.num[]` do wybierania procesów w takim przypadku.

Note:
Dla pozycja `proc.get[]`, gdy Zabbix agent sprawdza „cmdline" pod kątem nazwy procesu, użyje tylko tej części nazwy, która zaczyna się od ostatniego ukośnika i kończy na pierwszej spacji lub dwukropku. Nazwa procesu pobrana z pliku cmdline zostanie użyta tylko wtedy, gdy jej początek całkowicie odpowiada skróconej nazwie procesu w pliku `status`. Algorytm jest taki sam zarówno dla nazwy procesu w filtrze, jak i w wyjściu JSON.

Użycie parametru cmdline z odpowiednim wyrażeniem regularnym daje poprawny wynik:

```
$ zabbix_get -s localhost -k 'proc.num[,zabbix,,zabbix_agentd_30[ :]]'
6
```

Zachowaj ostrożność podczas używania pozycji proc.get[], proc.mem[] i proc.num[] do monitorowania programów, które modyfikują swoje linie poleceń.

Przed wstawieniem parametrów name i cmdline do pozycji proc.get[], proc.mem[] i proc.num[] możesz chcieć przetestować te parametry przy użyciu pozycji proc.num[] i polecenia ps.

Wątki jądra Linux

W pozycjach proc.get[], proc.mem[] i proc.num[] nie można wybierać wątków za pomocą parametru cmdline

Jako przykład weźmy jeden z wątków jądra:

```
$ ps -ef| grep kthreadd
root         2     0  0 09:33 ?        00:00:00 [kthreadd]
```

Można go wybrać za pomocą parametru name procesu:

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd,root]'
1
```

Jednak wybieranie według parametru cmdline procesu nie działa:

```
$ zabbix_get -s localhost -k 'proc.num[,root,,kthreadd]'
0
```

Powodem jest to, że agent Zabbix pobiera wyrażenie regularne określone w parametrze cmdline i stosuje je do zawartości pliku procesu /proc/<pid>/cmdline. W przypadku wątków jądra ich pliki /proc/<pid>/cmdline są puste. Dlatego parametr cmdline nigdy nie pasuje.

Zliczanie wątków w pozycjach proc.mem[] i proc.num[]

Wątki jądra Linux są zliczane przez pozycję proc.num[], ale nie raportują pamięci w pozycji proc.mem[]. Na przykład:

```
$ ps -ef | grep kthreadd
root         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: Cannot get amount of "VmSize" memory.
```

Ale co się stanie, jeśli istnieje proces użytkownika o tej samej nazwie co wątek jądra? Wtedy może to wyglądać tak:

```
$ ps -ef | grep kthreadd
root         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[] zliczył zarówno wątek jądra, jak i proces użytkownika. proc.mem[] raportuje pamięć tylko dla procesu użytkownika i traktuje pamięć wątku jądra tak, jakby wynosiła 0. Różni się to od przypadku powyżej, gdy zgłaszane było ZBX_NOTSUPPORTED.

Zachowaj ostrożność podczas używania pozycji proc.mem[] i proc.num[], jeśli nazwa programu przypadkowo odpowiada nazwie jednego z wątków.

Przed wstawieniem parametrów do pozycji proc.mem[] i proc.num[] warto przetestować parametry za pomocą pozycji proc.num[] oraz polecenia ps.

9 Szczegóły implementacji kontroli net.tcp.service i net.udp.service

Implementacja kontroli net.tcp.service i net.udp.service została szczegółowo opisana na tej stronie dla różnych usług określonych w parametrze service.

Parametry pozycji net.tcp.service

ftp

Tworzy połączenie TCP i oczekuje, że pierwsze 4 znaki odpowiedzi będą równe ”220 ”, a następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 21.

http

Tworzy połączenie TCP bez oczekiwania i wysyłania czegokolwiek. Jeśli nie określono portu, używany jest domyślny port 80.

https

Używa biblioteki libcurl (i działa wyłącznie z nią), nie weryfikuje autentyczności certyfikatu, nie weryfikuje nazwy hosta w certyfikacie SSL, pobiera jedynie nagłówek odpowiedzi (żądanie HEAD). Jeśli nie określono portu, używany jest domyślny port 443.

imap

Tworzy połączenie TCP i oczekuje, że pierwsze 4 znaki odpowiedzi będą równe ”* OK”, a następnie wysyła ”a1 LOGOUT\r\n”. Jeśli nie określono portu, używany jest domyślny port 143.

ldap

Otwiera połączenie z serwerem LDAP i wykonuje operację wyszukiwania LDAP z filtrem ustawionym na (objectClass=*). Oczekuje pomyślnego pobrania pierwszego atrybutu pierwszego wpisu. Jeśli nie określono portu, używany jest domyślny port 389.

nntp

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”200” lub ”201”, a następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 119.

pop

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”+OK”, a następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 110.

smtp

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”220”, po których następuje spacja, znak końca linii lub myślnik. Linie zawierające myślnik należą do odpowiedzi wielowierszowej i odpowiedź będzie odczytywana ponownie aż do otrzymania linii bez myślnika. Następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 25.

ssh

Tworzy połączenie TCP. Jeśli połączenie zostało ustanowione, obie strony wymieniają ciąg identyfikacyjny (SSH-major.minor-XXXX), gdzie major i minor to wersje protokołu, a XXXX to ciąg znaków. Zabbix sprawdza, czy znaleziono ciąg zgodny ze specyfikacją, a następnie odsyła ciąg ”SSH-major.minor-zabbix_agent\r\n” lub ”0\n” w przypadku niezgodności. Jeśli nie określono portu, używany jest domyślny port 22.

tcp

Tworzy połączenie TCP bez oczekiwania i wysyłania czegokolwiek. W przeciwieństwie do innych kontroli wymaga określenia parametru portu.

telnet

Tworzy połączenie TCP i oczekuje monitu logowania (’:’ na końcu). Jeśli nie określono portu, używany jest domyślny port 23.

Parametry pozycji net.udp.service

ntp

Wysyła pakiet SNTP przez UDP i sprawdza poprawność odpowiedzi zgodnie z RFC 4330, sekcja 5. Jeśli nie określono portu, używany jest domyślny port 123.

10 parametrów proc.get

Przegląd

Pozycja **proc.get**[<name>,<user>,<cmdline>,<mode>] jest obsługiwana w systemach Linux, Windows, FreeBSD, OpenBSD i NetBSD.

Lista parametrów procesu zwracanych przez pozycję różni się w zależności od systemu operacyjnego i wartości argumentu „mode”.

Linux

Następujące parametry procesu są zwracane w systemie Linux dla każdego trybu:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nazwa procesu
ppid: PID procesu nadrzędnego	ppid: PID procesu nadrzędnego	processes: liczba procesów
name: nazwa procesu	name: nazwa procesu	vsize: rozmiar pamięci wirtualnej
cmdline: polecenie z argumentami	user: użytkownik (rzeczywisty), z uprawnieniami którego działa proces	pmem: procent pamięci rzeczywistej
user: użytkownik (rzeczywisty), z uprawnieniami którego działa proces	group: grupa (rzeczywista), z uprawnieniami której działa proces	rss: rozmiar rezydentnego zestawu pamięci
group: grupa (rzeczywista), z uprawnieniami której działa proces	uid: identyfikator użytkownika	data: rozmiar segmentu danych
uid: identyfikator użytkownika	gid: identyfikator grupy, z uprawnieniami której działa proces	exe: rozmiar segmentu kodu
gid: identyfikator grupy, z uprawnieniami której działa proces	tid: identyfikator wątku	lib: rozmiar bibliotek współdzielonych
vsize: rozmiar pamięci wirtualnej	tname: nazwa wątku	lck: rozmiar zablokowanej pamięci
pmem: procent pamięci rzeczywistej	cputime_user: całkowity czas CPU spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)	pin: rozmiar przypiętych stron
rss: rozmiar rezydentnego zestawu pamięci	cputime_system: całkowity czas CPU spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)	pte: rozmiar wpisów tablicy stron
data: rozmiar segmentu danych	state: stan wątku	size: rozmiar segmentów kodu + danych + stosu procesu
exe: rozmiar segmentu kodu	ctx_switches: liczba przełączeń kontekstu	stk: rozmiar segmentu stosu
hwm: szczytowy rozmiar rezydentnego zestawu pamięci	page_faults: liczba błędów stron	swap: rozmiar używanej przestrzeni wymiany
lck: rozmiar zablokowanej pamięci		cputime_user: całkowity czas CPU spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)
lib: rozmiar bibliotek współdzielonych		cputime_system: całkowity czas CPU spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)
peak: szczytowy rozmiar pamięci wirtualnej		ctx_switches: liczba przełączeń kontekstu
pin: rozmiar przypiętych stron		threads: liczba wątków
pte: rozmiar wpisów tablicy stron		page_faults: liczba błędów stron
size: rozmiar segmentów kodu + danych + stosu procesu		pss: pamięć o proporcjonalnym rozmiarze zestawu
stk: rozmiar segmentu stosu		
swap: rozmiar używanej przestrzeni wymiany		
cputime_user: całkowity czas CPU spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)		

mode=process	mode=thread	mode=summary
cputime_system: całkowity czas CPU spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy) state: stan procesu (pobierany bezpośrednio z procfs, forma długa) ctx_switches: liczba przełączeń kontekstu threads: liczba wątków page_faults: liczba błędów stron pss: pamięć o proporcjonalnym rozmiarze zestawu		

Systemy operacyjne oparte na BSD

Następujące parametry procesu są zwracane w systemach FreeBSD, OpenBSD i NetBSD dla każdego trybu:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nazwa procesu
ppid: PID procesu nadrzędnego	ppid: PID procesu nadrzędnego	processes: liczba procesów
jid: ID jaila (tylko FreeBSD)	jid: ID jaila (tylko FreeBSD)	vsize: rozmiar pamięci wirtualnej
jname: nazwa jaila (tylko FreeBSD)	jname: nazwa jaila (tylko FreeBSD)	pmem: procent pamięci rzeczywistej (tylko FreeBSD)
name: nazwa procesu	name: nazwa procesu	rss: rozmiar rezydentnego zestawu pamięci
cmdline: polecenie z argumentami	user: użytkownik (rzeczywisty), z którego uprawnieniami działa proces	size: rozmiar procesu (kod + dane + stos)
user: użytkownik (rzeczywisty), z którego uprawnieniami działa proces	group: grupa (rzeczywista), z której uprawnieniami działa proces	tsize: rozmiar tekstu (kodu)
group: grupa (rzeczywista), z której uprawnieniami działa proces	uid: ID użytkownika	dsize: rozmiar danych
uid: ID użytkownika	gid: ID grupy, z której uprawnieniami działa proces	ssize: rozmiar stosu
gid: ID grupy, z której uprawnieniami działa proces	tid: ID wątku	cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)
vsize: rozmiar pamięci wirtualnej	tname: nazwa wątku	cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach)
pmem: procent pamięci rzeczywistej (tylko FreeBSD)	cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)	ctx_switches: liczba przełączeń kontekstu
rss: rozmiar rezydentnego zestawu pamięci	cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach)	threads: liczba wątków (nieobsługiwane w NetBSD)
size: rozmiar procesu (kod + dane + stos)	state: stan wątku	stk: rozmiar segmentu stosu
tsize: rozmiar tekstu (kodu)	ctx_switches: liczba przełączeń kontekstu	page_faults: liczba błędów strony
dsize: rozmiar danych	io_read_op: liczba operacji wejścia wymaganych przez system	fds: liczba deskryptorów plików (tylko OpenBSD)
ssize: rozmiar stosu	io_write_op: liczba operacji wyjścia wymaganych przez system	swap: rozmiar używanej przestrzeni wymiany
cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)		io_read_op: liczba operacji wejścia wymaganych przez system

mode=process	mode=thread	mode=summary
cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach) state: stan procesu (uśpienie dyskowe/działający/uśpiony/zatrzymanie śledzenia/zombie/inny) ctx_switches: liczba przełączeń kontekstu threads: liczba wątków (nieobsługiwane w NetBSD) page_faults: liczba błędów strony fds: liczba deskryptorów plików (tylko OpenBSD) swap: rozmiar używanej przestrzeni wymiany io_read_op: liczba operacji wejścia wymaganych przez system io_write_op: liczba operacji wyjścia wymaganych przez system		io_write_op: liczba operacji wyjścia wymaganych przez system

Windows

Następujące parametry procesu są zwracane w systemie Windows dla każdego trybu:

mode=process	mode=thread	mode=summary
pid: PID ppid: PID procesu nadrzędnego name: nazwa procesu user: użytkownik, z uprawnieniami którego działa proces sid: SID użytkownika vmsize: rozmiar pamięci wirtualnej wkset: rozmiar zestawu roboczego procesu cputime_user: całkowity czas CPU spędzony w trybie użytkownika (raportowany w milisekundach) cputime_system: całkowity czas CPU spędzony w trybie systemowym (raportowany w milisekundach) threads: liczba wątków page_faults: liczba błędów strony handles: liczba uchwytów io_read_b: odczytane bajty IO io_write_b: zapisane bajty IO io_read_op: operacje odczytu IO io_write_op: operacje zapisu IO io_other_b: bajty IO przesłane w operacjach innych niż odczyt i zapis io_other_op: operacje IO inne niż odczyt i zapis	pid: PID ppid: PID procesu nadrzędnego name: nazwa procesu user: użytkownik, z uprawnieniami którego działa proces sid: SID użytkownika tid: ID wątku	name: nazwa procesu processes: liczba procesów vmsize: rozmiar pamięci wirtualnej wkset: rozmiar zestawu roboczego procesu cputime_user: całkowity czas CPU spędzony w trybie użytkownika (raportowany w milisekundach) cputime_system: całkowity czas CPU spędzony w trybie systemowym (raportowany w milisekundach) threads: liczba wątków page_faults: liczba błędów strony handles: liczba uchwytów io_read_b: odczytane bajty IO io_write_b: zapisane bajty IO io_read_op: operacje odczytu IO io_write_op: operacje zapisu IO io_other_b: bajty IO przesłane w operacjach innych niż odczyt i zapis io_other_op: operacje IO inne niż odczyt i zapis

11 Ustawienia nieosiągalnego/niedostępnego interfejsu hosta

Przegląd

Kilka parametrów konfiguracji określa, jak serwer Zabbix powinien się zachować, gdy sprawdzenie przez agent (Zabbix, SNMP, IPMI, JMX) zakończy się niepowodzeniem, a interfejs hosta stanie się nieosiągalny.

Nieosiągalny interfejs

Interfejs hosta jest traktowany jako nieosiągalny po nieudanym sprawdzeniu (błąd sieciowy, przekroczenie limitu czasu) przez agenty Zabbix, SNMP, IPMI lub JMX. Od wersji Zabbix 6.2.0 aktywne sprawdzenia agenta Zabbix również wpływają na dostępność interfejsu. Gdy aktywne sprawdzenia stają się niedostępne, mają wpływ na ogólny status dostępności interfejsu agenta.

Od momentu, gdy interfejs staje się nieosiągalny, **UnreachableDelay** określa, jak często jest on sprawdzany ponownie przy użyciu jednej z pozycji (w tym reguł LLD).
Te ponowne sprawdzenia są wykonywane przez unreachable pollers (lub IPMI pollers w przypadku sprawdzeń IPMI).
Domyślnie odstęp między kolejnymi sprawdzeniami osiągalności wynosi 15 sekund.

Attention:
Sprawdzenia wykonywane przez asynchroniczne pollery nie są przenoszone do unreachable pollers.

W logu serwera Zabbix nieosiągalność jest wskazywana przez komunikaty takie jak:

```
Zabbix agent item "system.cpu.load[percpu,avg1]" on host "New host" failed: first network error, wait for
Zabbix agent item "system.cpu.load[percpu,avg15]" on host "New host" failed: another network error, wait f
```

Komunikaty logu określają dokładną pozycję, która zakończyła się niepowodzeniem, oraz jej typ (agent Zabbix).

Note:
Parametr *Timeout* również wpływa na to, jak wcześnie interfejs jest sprawdzany ponownie w czasie nieosiągalności. Jeśli Timeout jest ustawiony na 20 sekund, a UnreachableDelay na 30 sekund, następne sprawdzenie nastąpi po 50 sekundach od pierwszej próby.

Parametr **UnreachablePeriod** określa całkowity czas trwania okresu nieosiągalności.
Domyślnie wartość UnreachablePeriod wynosi 45 sekund.
Ta wartość powinna być kilkukrotnie większa niż UnreachableDelay, aby zapewnić wielokrotne ponowne sprawdzenie interfejsu przed oznaczeniem go jako niedostępnego.

Wewnętrzna pozycja zabbix[host,active_agent,available] umożliwia monitorowanie dostępności aktywnych sprawdzeń w scenariuszach nieosiągalności.

Przełączanie interfejsu z powrotem na dostępny

Po zakończeniu okresu niedostępności interfejs jest ponownie odpytywany, ze zmniejszonym priorytetem dla pozycji, która spowodowała przejście interfejsu w stan niedostępności. Jeśli niedostępny interfejs pojawi się ponownie, monitorowanie automatycznie wraca do normy:

```
wznowienie kontroli Zabbix agent na hoście "New host": połączenie przywrócone
```

Note:
Gdy interfejs staje się dostępny, host nie odpytuje natychmiast wszystkich swoich pozycji z dwóch powodów:
- Mogłoby to przeciążyć host.
- Czas przywrócenia interfejsu nie zawsze pokrywa się z planowanym czasem harmonogramu odpytywania pozycji.

Dlatego po przywróceniu dostępności interfejsu pozycje nie są odpytywane natychmiast, lecz są przeplanowywane na następną rundę odpytywania.

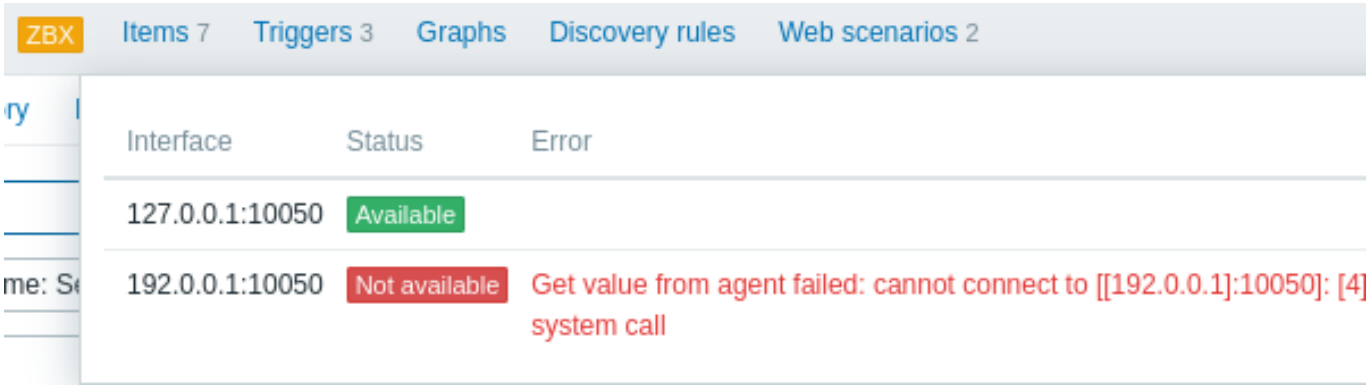
Niedostępny interfejs

Po zakończeniu okresu UnreachablePeriod, jeśli interfejs nie pojawi się ponownie, jest on traktowany jako niedostępny.

W logu serwera jest to sygnalizowane komunikatami takimi jak:

```
temporarily disabling Zabbix agent checks on host "New host": interface unavailable
```

a we frontend ikona dostępności hosta zmienia kolor z zielonego/szarego na żółty/czerwony (szczegóły niedostępnego interfejsu można zobaczyć w dymku podpowiedzi, który jest wyświetlany po najechaniu kursorem na ikonę dostępności hosta):



Parametr **UnavailableDelay** określa, jak często interfejs jest sprawdzany w czasie jego niedostępności.

Domyślnie jest to 60 sekund (więc w tym przypadku „temporarily disabling" z komunikatu logu powyżej będzie oznaczać wyłączenie sprawdzania na jedną minutę).

Gdy połączenie z interfejsem zostanie przywrócone, monitorowanie również automatycznie wraca do normy:

```
enabling Zabbix agent checks on host "New host": interface became available
```

12 Zdalne monitorowanie statystyk Zabbix

Przegląd

Możliwe jest udostępnienie zdalnie niektórych wewnętrznych metryk serwera i proxy Zabbix innej instancji Zabbix lub narzędziu zewnętrznemu. Może to być przydatne, aby osoby zapewniające wsparcie/dostawcy usług mogli zdalnie monitorować serwery/proxy Zabbix swoich klientów lub w organizacjach, w których Zabbix nie jest głównym narzędziem monitorującym, aby wewnętrzne metryki Zabbix mogły być monitorowane przez system zewnętrzny w ramach nadrzędnej konfiguracji monitoringu.

Wewnętrzne statystyki Zabbix są udostępniane dla konfigurowalnego zestawu adresów wymienionych w nowym parametrze 'StatsAllowedIP' server/proxy. Żądania będą akceptowane wyłącznie z tych adresów.

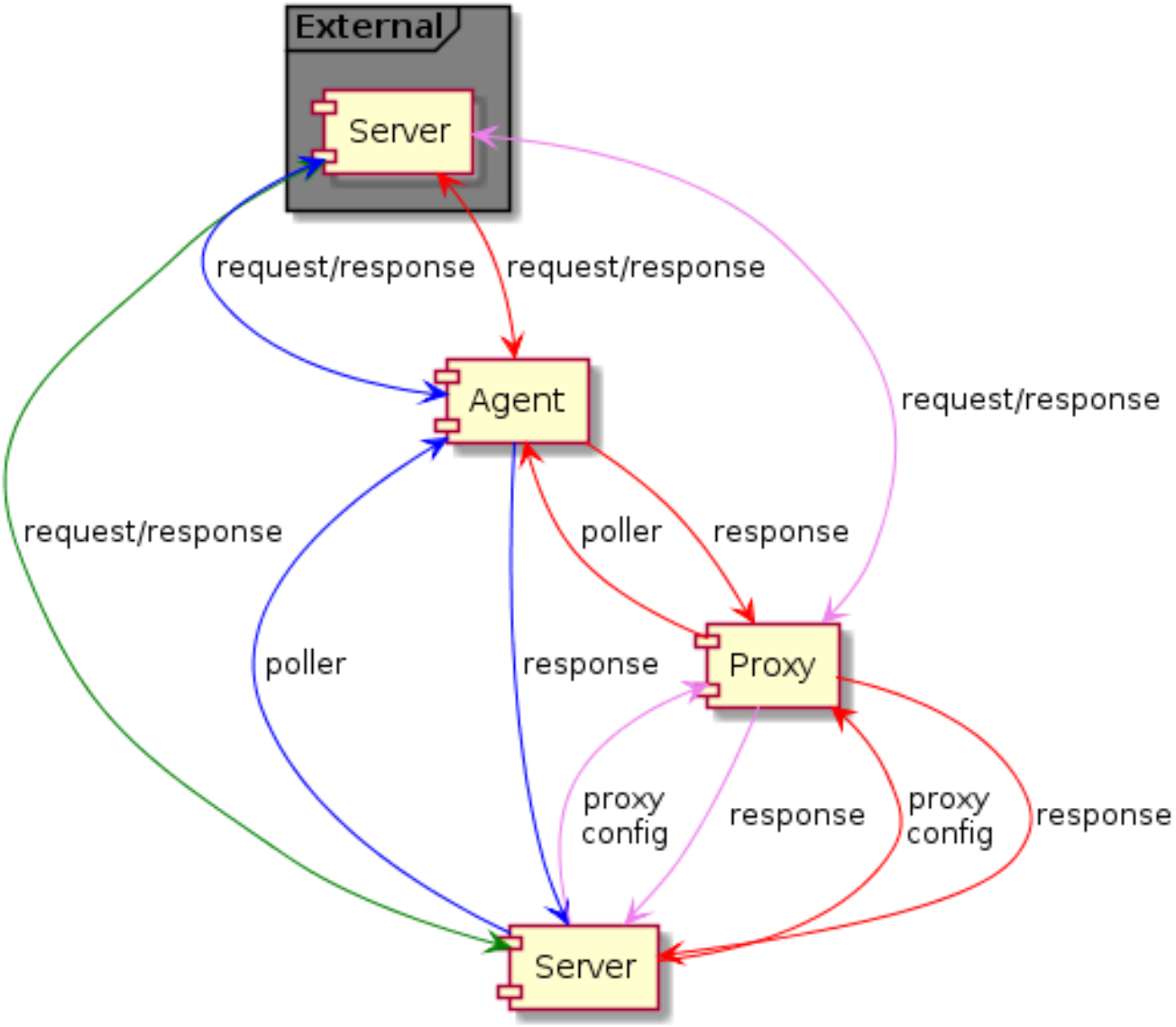
Pozycje

Aby skonfigurować odpytywanie wewnętrznych statystyk na innej instancji Zabbix, można użyć dwóch pozycji:

- wewnętrzna pozycja `zabbix[stats,<ip>,<port>]` — do bezpośrednich zdalnych zapytań do serwera/proxy Zabbix. `<ip>` i `<port>` są używane do identyfikacji instancji docelowej.
- pozycja agenta `zabbix.stats[<ip>,<port>]` — do zdalnych zapytań do serwera/proxy Zabbix opartych na agencie. `<ip>` i `<port>` są używane do identyfikacji instancji docelowej.

Zobacz także: Pozycje wewnętrzne, Pozycje agenta Zabbix

Poniższy diagram ilustruje użycie jednej z tych pozycji w zależności od kontekstu.



- — Serwer → zewnętrzna instancja Zabbix (`zabbix[stats,<ip>,<port>]`)
- — Serwer → proxy → zewnętrzna instancja Zabbix (`zabbix[stats,<ip>,<port>]`)
- — Serwer → agent → zewnętrzna instancja Zabbix (`zabbix.stats[<ip>,<port>]`)
- — Serwer → proxy → agent → zewnętrzna instancja Zabbix (`zabbix.stats[<ip>,<port>]`)

Aby upewnić się, że instancja docelowa zezwala na odpytywanie jej przez instancję zewnętrzną, podaj adres instancji zewnętrznej w parametrze `StatsAllowedIP` na instancji docelowej.

Udostępniane metryki

Pozycje statystyk zbierają statystyki zbiorczo i zwracają JSON, który stanowi podstawę dla pozycji zależnych do pobierania z niego danych. Następujące metryki wewnętrzne są zwracane przez dowolną z tych dwóch pozycji:

- `zabbix[boottime]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[preprocessing]` (tylko serwer)
- `zabbix[preprocessing_queue]` (tylko serwer)
- `zabbix[process,<type>,<mode>,<state>]` (tylko statystyki oparte na typie procesu)
- `zabbix[rcache,<cache>,<mode>]`
- `zabbix[requiredperformance]`
- `zabbix[triggers]` (tylko serwer)
- `zabbix[uptime]`
- `zabbix[vcache,buffer,<mode>]` (tylko serwer)

- zabbix[vcache,cache,<parameter>]
- zabbix[version]
- zabbix[vmware,buffer,<mode>]
- zabbix[wcache,<cache>,<mode>] (typ pamięci podręcznej „trends” tylko dla serwera)

Szablony

Szablony są dostępne do zdalnego monitorowania wewnętrznych metryk serwera Zabbix lub proxy z zewnętrznej instancji:

- Stan zdalnego serwera Zabbix
- Stan zdalnego proxy Zabbix

Należy pamiętać, że aby użyć szablonu do zdalnego monitorowania wielu zewnętrznych instancji, wymagany jest oddzielny host dla monitorowania każdej zewnętrznej instancji.

Proces trapper

Odbieranie żądań metryk wewnętrznych z zewnętrznej instancji Zabbix jest obsługiwane przez proces trapper, który weryfikuje żądanie, zbiera metryki, tworzy bufor danych JSON i odsyła przygotowany JSON z powrotem, na przykład z serwera:

```
{
  "response": "success",
  "data": {
    "boottime": N,
    "uptime": N,
    "hosts": N,
    "items": N,
    "items_unsupported": N,
    "preprocessing": {
        "queued": {
          "count": N,
          "size": N
        },
        "direct": {
          "count": N,
          "size": N
        },
        "queue": N
    },
    "preprocessing_queue": N,
    "process": {
      "alert manager": {
        "busy": {
          "avg": N,
          "max": N,
          "min": N
        },
        "idle": {
          "avg": N,
          "max": N,
          "min": N
        },
        "count": N
      },
...
    },
    "queue": N,
    "rcache": {
      "total": N,
      "free": N,
      "pfree": N,
      "used": N,
      "pused": N
    },
    "requiredperformance": N,
    "triggers": N,
```

```
    "uptime": N,
    "vcache": {
      "buffer": {
        "total": N,
        "free": N,
        "pfree": N,
        "used": N,
        "pused": N
      },
      "cache": {
        "requests": N,
        "hits": N,
        "misses": N,
        "mode": N
      }
    },
    "vmware": {
      "total": N,
      "free": N,
      "pfree": N,
      "used": N,
      "pused": N
    },
    "version": "N",
    "wcache": {
      "values": {
        "all": N,
        "float": N,
        "uint": N,
        "str": N,
        "log": N,
        "text": N,
        "not supported": N
      },
      "history": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      },
      "index": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      },
      "trend": {
        "pfree": N,
        "free": N,
        "total": N,
        "used": N,
        "pused": N
      }
    }
  }
}
```


Wewnętrzne pozycje kolejki

Istnieją także dwie dodatkowe pozycje, które umożliwiają zdalne odpytywanie statystyk wewnętrznej kolejki w innej instancji Zabbix:

- wewnętrzna pozycja `zabbix[stats,<ip>,<port>,queue,<from>,<to>]` — do bezpośrednich zapytań o wewnętrzną kolejkę do zdalnego serwera/proxy Zabbix
- pozycja agenta `zabbix.stats[<ip>,<port>,queue,<from>,<to>]` — do opartych na agencie zapytań o wewnętrzną kolejkę do zdalnego serwera/proxy Zabbix

Zobacz także: Wewnętrzne pozycje, Pozycje agenta Zabbix

13 Konfigurowanie Kerberos z Zabbix

Przegląd

Uwierzytelnianie Kerberos może być używane w monitorowaniu WWW oraz w pozycjach HTTP w Zabbix.

Ta strona opisuje przykład konfiguracji Kerberos dla serwera Zabbix, aby wykonywać monitorowanie WWW `www.example.com` z użyciem principal Kerberos dla procesu Zabbix w Debian/Ubuntu.

Konfiguracja

1. Zainstaluj KDC i narzędzia klienckie:

```
sudo apt update
sudo apt install krb5-kdc krb5-admin-server krb5-user
```

Podczas konfiguracji pakietów odpowiedz na wyświetlane pytania, np.:

```
Default Kerberos version 5 realm: EXAMPLE.COM
Kerberos servers for your realm: localhost (lub twój FQDN)
Administrative server for your Kerberos realm: localhost (lub twój FQDN)
```

2. Przypisz przyjazną nazwę hosta (opcjonalnie, do testów lokalnych).

Edytuj /etc/hosts i dodaj wpis dla swojego DC oraz serwera WWW, jeśli nie masz DNS:

```
sudo vi /etc/hosts
```

Przykładowy wiersz, który możesz dodać:

```
192.168.1.100  dc01.example.com dc01
```

3. Skonfiguruj klienta Kerberos i realm KDC:

```
sudo vi /etc/krb5.conf
```

Przykładowe ustawienia:

```
[libdefaults]
    default_realm = EXAMPLE.COM
    dns_lookup_realm = false
    dns_lookup_kdc = false
    rdns = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true

[realms]
    EXAMPLE.COM = {
        kdc = dc01.example.com
        admin_server = dc01.example.com
    }

[domain_realm]
    .example.com = EXAMPLE.COM
    example.com = EXAMPLE.COM
```

Jeśli planujesz używać `.localdomain` lub innych niepublicznych nazw, dodaj jawne mapowania domena→realm, aby mapowanie nazwa hosta→realm działało poprawnie.
Niezgodności w tym miejscu powodują błędy `Server not found in Kerberos database`.

4. Zainicjalizuj bazę danych Kerberos (jednorazowo, na hoście KDC).
Po wyświetleniu monitu ustaw bezpieczne hasło główne:

```
sudo krb5_newrealm
```

5. Utwórz principal `HTTP/host.fqdn@REALM`, używając dokładnej nazwy hosta, z której będą korzystać klienci; preferowane są małe litery (np. `HTTP/web.example.com@EXAMPLE.COM`).
Niezgodność wielkości liter/nazwy powoduje `Server not found in Kerberos database`.

```
sudo kadmin.local
```

Wewnątrz `kadmin.local`:

```
addprinc kerb-admin@EXAMPLE.COM      # principal administracyjny
addprinc -randkey HTTP/dc01.example.com@EXAMPLE.COM
ktadd -k /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
quit
```

Przenieś keytab na host WWW (lub pozostaw lokalnie, jeśli to ta sama maszyna) i ustaw uprawnienia umożliwiające użycie przez Apache:

```
chown www-data:www-data /etc/apache2/http.keytab
chmod 600 /etc/apache2/http.keytab
#### weryfikacja
sudo -u www-data -k /etc/apache2/http.keytab
```

6. Zainstaluj i włącz moduł Apache GSSAPI:

```
sudo apt install libapache2-mod-auth-gssapi
sudo a2enmod auth_gssapi
sudo a2enmod headers
sudo systemctl restart apache2
```

> **Attention:**
> Nie wszystkie wersje mod_auth_gssapi obsługują każdą dyrektywę `Gssapi*`.
> Jeśli Apache kończy działanie z błędem `Invalid command 'GssapiCredStore'`, usuń nieobsługiwaną dyrektywę lub zaktualizuj moduł.

7. Skonfiguruj VirtualHost (dostosuj `DocumentRoot` / ścieżkę do interfejsu Zabbix):

```
sudo vi /etc/apache2/sites-available/zabbix.conf
```

Wewnątrz `zabbix.conf`:

```
<VirtualHost *:80>
    ServerName dc01.example.com
    DocumentRoot /usr/share/zabbix/ui
    <Directory /usr/share/zabbix/ui>
        Options FollowSymLinks
        AllowOverride None
        Require all granted
        AuthType GSSAPI
        AuthName "Kerberos Login"
        GssapiCredStore keytab:/etc/apache2/http.keytab
        GssapiLocalName On
        Require valid-user
    </Directory>
    RequestHeader set X-Remote-User %{REMOTE_USER}s env=REMOTE_USER
    RequestHeader unset Authorization
</VirtualHost>
```

Uruchom ponownie Apache:

```
sudo systemctl restart apache2
```

8. Włącz/uruchom usługi KDC i zweryfikuj porty nasłuchujące (host KDC):

```
sudo systemctl enable --now krb5-kdc krb5-admin-server
ss -tnlp | grep :80    # lub: sudo netstat -tnlp | grep :80
```

9. Uzyskaj TGT do testów (uruchom jako użytkownik, który będzie używać ticketu).

Na liście ticketów powinien być widoczny krbtgt/EXAMPLE.COM@EXAMPLE.COM.
Uruchom kinit jako ten sam użytkownik systemu operacyjnego, który potrzebuje ticketu (np. zabbix dla testów WWW lub www-data/Apache dla interaktywnych testów SSO w przeglądarce).
Tickety wydane innemu użytkownikowi systemu operacyjnego nie będą widoczne, chyba że zostaną odpowiednio dostosowane KRB5CCNAME i uprawnienia.

```
kinit kerb-admin@EXAMPLE.COM
klist
```

10. Przetestuj wymianę SPNEGO za pomocą curl (z klienta z prawidłowym TGT).
200 OK (lub przekierowanie do aplikacji) oznacza, że SPNEGO zadziałało poprawnie:

```
curl -v --negotiate -u : http://dc01.example.com/
```

11. Opcjonalnie, jeśli interfejs Zabbix ma akceptować logowanie uwierzytelniane przez HTTP, włącz HTTP auth w frontendzie Zabbix (ui/conf/zabbix.conf.php):

```
$ALLOW_HTTP_AUTH = true;
```

W interfejsie WWW przejdź do *Users* > *Authentication* i otwórz kartę *HTTP settings*.
Zaznacz pole wyboru *Enable HTTP authentication* i kliknij *Ok* w wyskakującym oknie.
Wybierz „HTTP login form" z listy rozwijanej *Default login form*.
Zdecyduj, czy *Case-sensitive login* odpowiada polityce twojego katalogu.
Kliknij przycisk *Update*, aby zakończyć.

12. Konfiguracja przeglądarki (jako przykład użyto Firefox): ustaw network.negotiate-auth.trusted-uris na host(y) wykonujące Negotiate (dc01.example.com), aby przeglądarka automatycznie wysyłała tokeny Kerberos.

W about:config:

```
network.negotiate-auth.trusted-uris = dc01.example.com
```

Po wejściu na http://dc01.example.com powinieneś zostać od razu zalogowany do Zabbix bez formularza.

13. Utrzymuj klucze/tickety aktualne.
Domyślny czas życia ticketu Kerberos wynosi około 10 godzin.
Dodaj cron/systemd timer, aby uniknąć wygaśnięcia:

```
####dla usługi WWW
kinit -kt /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
####dla użytkownika monitorowania
kinit -kt /var/lib/zabbix/kerb.keytab kerb-admin@EXAMPLE.COM
```

14. Kontrole porządkowe:

- klist -k /etc/apache2/http.keytab — sprawdź, czy principal usługi jest obecny w keytab.
- sudo tail -f /var/log/apache2/error.log — obserwuj błędy GSSAPI (gss_acquire_cred[_from]() failed to get server creds oznacza problem z keytab/uprawnieniami lub brak principal).
- Zwracanie 401/403 przez curl --negotiate często oznacza nieprawidłowy principal, brak ticketu, niedopasowanie nagłówka hosta lub problem z uprawnieniami systemu plików; sprawdź logi i mapowania domen w /etc/krb5.conf.

Uwagi dotyczące bezpieczeństwa i uprawnień do plików

Pliki keytab muszą być dostępne do odczytu wyłącznie dla konta, które ich potrzebuje. Przykładowe uprawnienia: 0400 z właścicielem zabbix:zabbix dla pliku keytab użytkownika zabbix lub 0440 i root:www-data dla pliku keytab Apache.

Unikaj przechowywania na hoście długotrwałych haseł w postaci zwykłego tekstu. Tam, gdzie to możliwe, używaj plików keytab lub principali maszyn przyłączonych do domeny.

Podczas uruchamiania testów lub skryptów, które ustawiają KRB5CCNAME lub kopiują pliki keytab, po zakończeniu operacji dokładnie sprawdź właściciela i uprawnienia — odrzucanie poświadczeń przez serwer WWW często wynika z problemu z uprawnieniami do plików.

14 parametry modbus.get

Przegląd

Poniższa tabela przedstawia szczegóły parametrów pozycji modbus.get.

Parametry

Parameter	Opis	Defaults	Example
endpoint	Protokół i adres endpointu, zdefiniowane jako `protocol://connection_string` Możliwe wartości protokołu: *rtu*, *ascii* (tylko Agent 2), *tcp* Format connection string: dla *tcp* - `address:port` dla linii szeregowej: *rtu*, *ascii* - `port_name:speed:params` gdzie 'speed' - 1200, 9600 itd. 'params' - bity danych (5,6,7 lub 8), parzystość (n,e lub o dla brak/parzysta/nieparzysta), bity stopu (1 lub 2)	protocol: brak protokół *rtu/ascii*: port_name: brak speed: 115200 params: 8n1 protokół *tcp*: address: brak port: 502	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 rtu://ttyS0:9600 ascii://ttyS1
slave id	Adres Modbus urządzenia, do którego jest przeznaczony (1 do 247), zobacz MODBUS Messaging Implementation Guide (strona 23) urządzenie tcp (nie GW) zignoruje to pole	serial: 1 tcp: 255 (0xFF)	2
function	Puste lub wartość obsługiwanej funkcji: 1 - Odczyt cewki, 2 - Odczyt wejścia dyskretnego, 3 - Odczyt rejestrów przechowujących, 4 - Odczyt rejestrów wejściowych	puste	3
address	Adres pierwszego rejestru, cewki lub wejścia. Jeśli 'function' jest puste, wtedy 'address' powinien należeć do zakresu dla: Cewka - 00001 - 09999 Wejście dyskretne - 10001 - 19999 Rejestr wejściowy - 30001 - 39999 Rejestr przechowujący - 40001 - 49999 Jeśli 'function' nie jest puste, pole 'address' będzie miało zakres od 0 do 65535 i zostanie użyte bez modyfikacji (PDU)	pusta function: 00001 niepusta function: 0	9999
count	Liczba sekwencyjnych wartości 'type', które zostaną odczytane z urządzenia, gdzie: dla cewki lub wejścia dyskretnego 'type' = 1 bit dla pozostałych przypadków: (count*sizeof(type))/2 = rzeczywista liczba rejestrów do odczytu Jeśli 'offset' nie jest równe 0, wartość ta zostanie dodana do 'rzeczywistej liczby' Dopuszczalny zakres dla 'rzeczywistej liczby' to 1:65535	1	2

Parameter	Opis	Defaults	Example
type	Typ danych: dla Read Coil i Read Discrete Input - *bit* dla Read Holding Registers i Read Input Registers: *int8* - 8bit *uint8* - 8bit (bez znaku) *int16* - 16bit *uint16* - 16bit (bez znaku) *int32* - 32bit *uint32* - 32bit (bez znaku) *float* - 32bit *uint64* - 64bit (bez znaku) *double* - 64bit	bit uint16	uint64
endianness	Typ kolejności bajtów: *be* - Big Endian *le* - Little Endian *mbe* - Mid-Big Endian *mle* - Mid-Little Endian Ograniczenia: dla 1 bitu - be dla 8 bitów - be,le dla 16 bitów - be,le	be	le
offset	Liczba rejestrów, począwszy od 'address', których wynik zostanie odrzucony. Rozmiar każdego rejestru to 16bit (potrzebne do obsługi urządzeń, które nie obsługują losowego dostępu do odczytu).	0	4

15 Tworzenie niestandardowych nazw liczników wydajności dla VMware

Przegląd

Ścieżka licznika wydajności VMware ma format `group/counter[rollup]`, gdzie:

- `group` — grupa licznika wydajności, na przykład *cpu*
- `counter` — nazwa licznika wydajności, na przykład *usagemhz*
- `rollup` — typ agregacji licznika wydajności, na przykład *average*

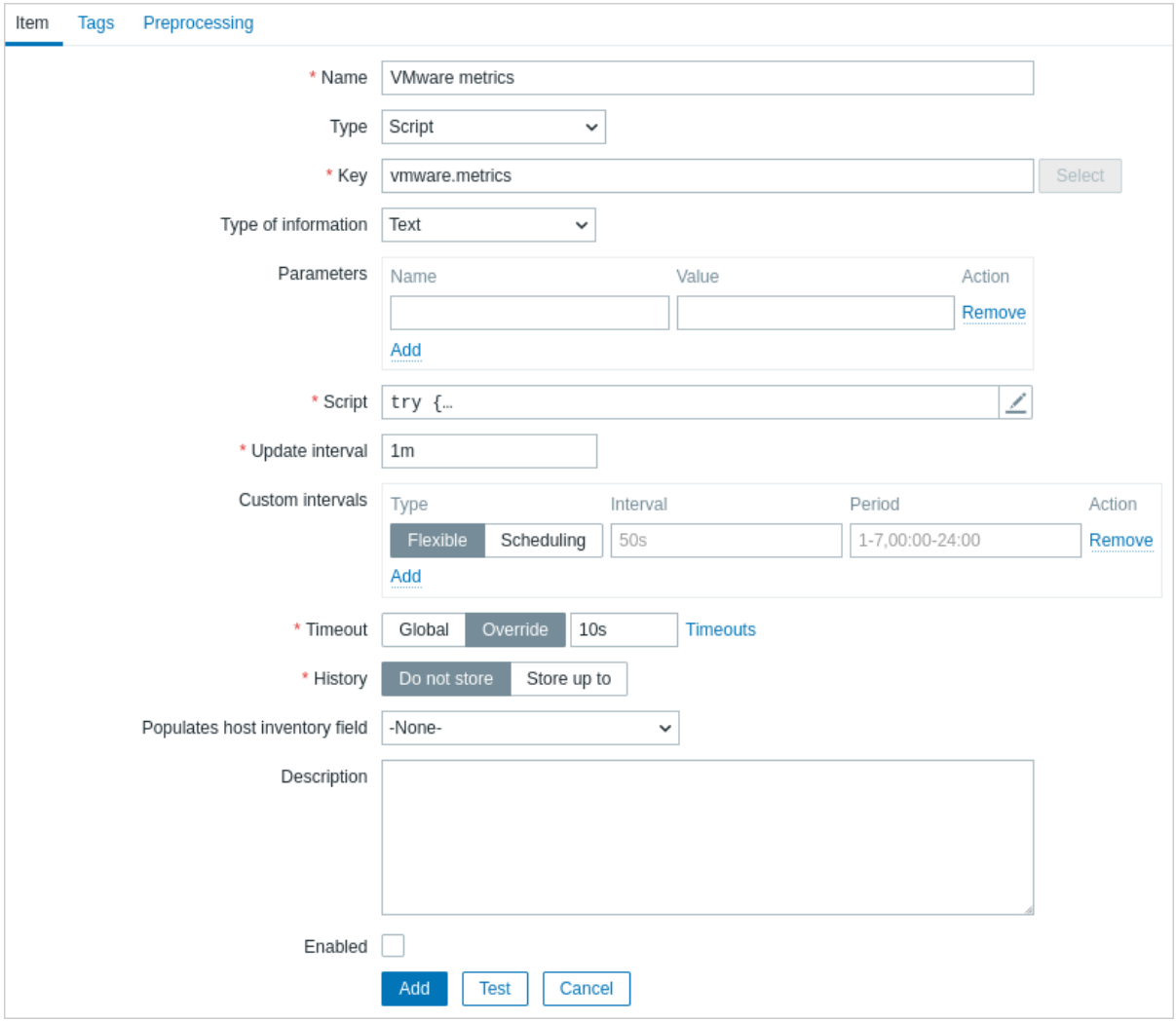
Tak więc powyższy przykład da następującą ścieżkę licznika: `cpu/usagemhz[average]`

Opisy grup liczników wydajności, nazwy liczników oraz typy agregacji można znaleźć w dokumentacji VMware.

Możliwe jest uzyskanie nazw wewnętrznych oraz tworzenie niestandardowych nazw liczników wydajności przy użyciu pozycji skryptowej w Zabbix.

Konfiguracja

1. Utwórz wyłączoną pozycję Script na głównym hoście VMware (tam, gdzie znajduje się pozycja **eventlog[]**) z następującymi parametrami:



- *Nazwa*: Metryki VMware
- *Typ*: Script
- *Klucz*: vmware.metrics
- *Typ informacji*: Tekst
- *Skrypt*: skopiuj i wklej skrypt podany poniżej
- *Limit czasu*: 10
- *Historia*: Nie przechowuj
- *Włączone*: niezaznaczone

Skrypt

```
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:vi
    <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:vi
<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:v
    <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;
```

Po skonfigurowaniu pozycji naciśnij przycisk *Test*, a następnie *Get value*.



Skopiuj otrzymany XML do dowolnego formatera XML i znajdź żądaną metrykę.

Przykład XML dla jednej metryki:

```
<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>
```

Użyj XPath, aby wyodrębnić ścieżkę licznika z otrzymanego XML. Dla powyższego przykładu XPath będzie następujący:

pole	xPath	wartość
group	//groupInfo[../key=6]/key	cpu
counter	//nameInfo[../key=6]/key	usagemhz
rollup	//rollupType[../key=6]	average

Wynikowa ścieżka licznika wydajności w tym przypadku to: `cpu/usagemhz[average]`

16 Wartości zwracane dla system.sw.packages.get

Przegląd

Ta sekcja zawiera szczegółowe informacje o wartościach zwracanych dla pozycji agenta Zabbix `system.sw.packages.get`.

Szczegóły

Wyjściem tej pozycji jest tablica obiektów, z których każdy zawiera następujące klucze:

- **name** - nazwa pakietu
- **manager** - menedżer pakietów, który zgłosił te dane (`rpm`, `dpkg`, `pacman`, `pkgtool` lub `portage`)
- **version** - wersja pakietu
- **size** - rozmiar nieskompresowanego pakietu w bajtach (jeśli niedostępny, ustaw na 0)
- **arch** - architektura pakietu
- **buildtime** - obiekt z 2 wpisami:
  - **timestamp** - znacznik czasu UNIX, kiedy pakiet został zbudowany (jeśli niedostępny, ustaw na 0)
  - **value** - czytelna dla człowieka data i godzina, kiedy pakiet został zbudowany (jeśli niedostępne, ustaw na pusty ciąg znaków)
- **installtime** - obiekt z 2 wpisami:
  - **timestamp** - znacznik czasu UNIX, kiedy pakiet został zainstalowany (jeśli niedostępny, ustaw na 0)
  - **value** - czytelna dla człowieka data i godzina, kiedy pakiet został zainstalowany (jeśli niedostępne, ustaw na pusty ciąg znaków)

Przykład:

```
[
    {
        "name": "util-linux-core",
        "manager": "rpm",
        "version": "2.37.4-3.el9",
        "size": 1296335,
        "arch": "x86_64",
        "buildtime": {
```

```
            "timestamp" : 1653552239,
            "value" : "Sep 20 01:39:40 2021 UTC"
        },
        "installtime": {
            "timestamp" : 1660780885,
            "value" : "Aug 18 00:01:25 2022 UTC"
        }
    },
    {
        "name": "xfonts-base",
        "manager": "dpkg",
        "version": "1:1.0.5",
        "size": 7337984,
        "arch": "all",
        "buildtime": {
            "timestamp": 0,
            "value": ""
        },
        "installtime": {
            "timestamp": 0,
            "value": ""
        }
    }
]
```

17 Wartości zwracane dla net.dns.get

Przegląd

Ta sekcja zawiera szczegółowe informacje o wartościach zwracanych dla pozycji Zabbix agent 2 `net.dns.get`.

Szczegóły

Wyjście tej pozycji jest obiektem zawierającym informacje o rekordach DNS na podstawie parametrów podanych w kluczu pozycji.

Na przykład pozycja `net.dns.get[,example.com]` może zwrócić następujący JSON dla odrzuconego zapytania:

```
{
    "flags": [
        "RA"
    ],
    "query_time": "0.019030",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "REFUSED",
    "zbx_error_code": 0
}
```

Po podaniu adresu IP serwera DNS pozycja `net.dns.get[192.0.2.1,example.com]` może zwrócić następujący JSON:

```
{
    "answer_section": [
        {
            "class": "IN",
            "name": "example.com.",
            "rdata": {
                "expire": 1209600,
                "mbox": "noc.dns.example.org.",
                "minttl": 3600,
                "ns": "ns.example.org.",
```

```
                "refresh": 7200,
                "retry": 3600,
                "serial": 2022091378
            },
            "rdlength": 44,
            "ttl": 1205,
            "type": "SOA"
        }
    ],
    "flags": [
        "RA"
    ],
    "query_time": "0.029556",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "NOERROR",
    "zbx_error_code": 0
}
```


Jeśli wystąpi problem z połączeniem, pozycja `net.dns.get[192.0.2.1,example.com]` może zwrócić błąd:


```
{
    "zbx_error_code": -1,
    "zbx_error_msg": "Communication error: read udp 192.0.2.1:12345->192.0.2.1:53: i/o timeout"
}
```


Możliwe są następujące typy kodów błędów:

Scenario	"zbx_error_code"	"zbx_error_msg"
Brak błędów, a odpowiedź DNS została odebrana i przeanalizowana.	0	
DNS jest niedostępny.	-1	"Communication error"
Błąd występuje podczas parsowania JSON	-2	"Received unexpected response"

Przy dodatkowych parametrach pozycja `net.dns.get[192.0.2.1,example.com,ANY,5,5,tcp,"cdflag,rdflag,dnssec,nsid,edn` może zwrócić następujący JSON:


```
{
    "additional_section": [
        {
            "extended_rcode": 32768,
            "name": ".",
            "rdata": {
                "options": [
                {
                    "code": 0,
                    "nsid": "67 70 64 6e 73 2d 6c 70 70"
                }
            ]
        },
            "rdlength": 13,
            "type": "OPT",
            "udp_payload": 512
        }
    ],
    "answer_section": [
        {
            "class": "IN",
            "name": "example.com.",
```

```
            "rdata": {
                "a": "192.0.2.1"
            },
            "rdlength": 4,
            "ttl": 19308,
            "type": "A"
        },
        {
            "class": "IN",
            "name": "example.com.",
            "rdata": {
                "algorithm": 13,
                "expiration": 1704715951,
                "inception": 1702910624,
                "key_tag": 21021,
                "labels": 2,
                "orig_ttl": 86400,
                "signature": "HVBOBcJJQy0S08J3f8kviPj8UkEUj7wmyiMyQqPSWgQIY9SCEJ5plq6KuxJmtAek1txZWXDo+6tp
                "signer_name": "example.com.",
                "type_covered": "A"
            },
            "rdlength": 95,
            "ttl": 19308,
            "type": "RRSIG"
        }
    ],
    "flags": [
        "RD",
        "RA",
        "AD",
        "CD"
    ],
    "query_time": "0.058221",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "ANY"
        }
    ],
    "response_code": "NOERROR",
    "zbx_error_code": 0
}
```


Zobacz także

Aby uzyskać więcej informacji o rekordach DNS, zobacz:

- Domain Names - Implementation and Specification
- Domain Name System (DNS) Parameters

18 Uwagi dotyczące pozycji system.cpu.util w systemie Windows

system.cpu.util pozycja dostarcza procent wykorzystania CPU.

Gdy w Zabbix agent dla Windows uruchamiany jest proces collectora, inicjalizowany jest bufor dla N pozycji dla N procesorów logicznych (wątków). Wartości są okresowo aktualizowane w buforze przy użyciu liczników wydajności Windows. Wartości te są zwracane, gdy żądane jest `system.cpu.util[n]`, gdzie 0 <= n < N.

Jeszcze jedna pozycja w buforze jest używana dla `system.cpu.util[total]`.

Zabbix automatycznie wybiera różne liczniki wydajności, aby obsługiwać zarówno systemy NUMA, jak i starsze wersje Windows bez obsługi grup procesorów.

Jeśli liczba procesorów logicznych (wątków) jest mniejsza lub równa 64, a liczba grup procesorów jest równa 1, wtedy:

```
N to liczba procesorów logicznych (wątków).

\Processor(_Total)\% Processor Time
\Processor(0)\% Processor Time
\Processor(1)\% Processor Time
\Processor(2)\% Processor Time
...
\Processor(N-1)\% Processor Time
```

W przeciwnym razie:

```
G to liczba grup procesorów, a N to liczba procesorów logicznych (wątków) w grupie.

\Processor Information(_Total)\% Processor Time
\Processor Information(0,0)\% Processor Time
\Processor Information(0,1)\% Processor Time
\Processor Information(0,2)\% Processor Time
...
\Processor Information(0,N-1)\% Processor Time
...
\Processor Information(G-1,0)\% Processor Time
\Processor Information(G-1,1)\% Processor Time
\Processor Information(G-1,2)\% Processor Time
...
\Processor Information(G-1,N-1)\% Processor Time
```

5 Obsługiwane funkcje

Kliknij odpowiednią grupę funkcji, aby zobaczyć więcej szczegółów.

Grupa funkcji	Funkcje
Funkcje agregujące	avg, bucket_percentile, count, histogram_quantile, item_count, kurtosis, mad, max, min, skewness, stddevpop, stddevsamp, sum, sumofsquares, varpop, varsamp
Funkcje foreach	avg_foreach, bucket_rate_foreach, count_foreach, exists_foreach, last_foreach, max_foreach, min_foreach, sum_foreach
Funkcje bitowe	bitand, bitlshift, bitnot, bitor, bitrshift, bitxor
Funkcje daty i czasu	date, dayofmonth, dayofweek, now, time
Funkcje historii	change, changecount, count, countunique, find, first, firstclock, fuzzytime, last, lastclock, logeventid, logseverity, logsource, logtimestamp, monodec, monoinc, nodata, percentile, rate
Funkcje trendów	baselinedev, baselinewma, trendavg, trendcount, trendmax, trendmin, trendstl, trendsum
Funkcje matematyczne	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
Funkcje operatorów	between, in
Funkcje predykcyjne	forecast, timeleft

Grupa funkcji	Funkcje
Funkcje tekstowe	ascii, bitlength, bytelength, char, concat, insert, jsonpath, left, length, ltrim, mid, repeat, replace, right, rtrim, trim, xmlxpath

O ile nie zaznaczono inaczej, te funkcje są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje foreach są obsługiwane tylko dla obliczeń agregujących.

1 Funkcje agregujące

O ile nie zaznaczono inaczej, wszystkie funkcje wymienione tutaj są obsługiwane w:

- wyrażeniach wyzwalaczy
- pozycjach obliczanych
- makrach wyrażeń

Funkcje agregujące mogą działać z:

- historią pozycji, na przykład `min(/host/key,1h)`
- funkcjami foreach jako jedynym parametrem, na przykład `min(last_foreach(/*/key))` (tylko w pozycjach obliczanych; nie można ich używać w wyzwalaczach)

Funkcje są wymienione bez dodatkowych informacji. Kliknij nazwę funkcji, aby zobaczyć pełne szczegóły.

Function	Description
avg	Średnia wartość pozycji w zdefiniowanym okresie oceny.
bucket_percentile	Oblicza percentyl na podstawie kubełków histogramu.
count	Liczba wartości w tablicy zwróconej przez funkcję foreach.
histogram_quantile	Oblicza kwantyl φ na podstawie kubełków histogramu.
item_count	Liczba istniejących pozycji w konfiguracji, które pasują do kryteriów filtra.
kurtosis	”Szczytowość” rozkładu prawdopodobieństwa w zebranych wartościach w zdefiniowanym okresie oceny.
mad	Mediana odchyleń bezwzględnych w zebranych wartościach w zdefiniowanym okresie oceny.
max	Najwyższa wartość pozycji w zdefiniowanym okresie oceny.
min	Najniższa wartość pozycji w zdefiniowanym okresie oceny.
skewness	Asymetria rozkładu prawdopodobieństwa w zebranych wartościach w zdefiniowanym okresie oceny.
stddevpop	Odchylenie standardowe populacji w zebranych wartościach w zdefiniowanym okresie oceny.
stddevsamp	Odchylenie standardowe próby w zebranych wartościach w zdefiniowanym okresie oceny.
sum	Suma zebranych wartości w zdefiniowanym okresie oceny.
sumofsquares	Suma kwadratów zebranych wartości w zdefiniowanym okresie oceny.
varpop	Wariancja populacji zebranych wartości w zdefiniowanym okresie oceny.
varsamp	Wariancja próby zebranych wartości w zdefiniowanym okresie oceny.

Parametry wspólne

- `/host/key` jest wspólnym obowiązkowym pierwszym parametrem funkcji odwołujących się do historii pozycji hosta
- `(sec|#num)<:time shift>` jest wspólnym drugim parametrem funkcji odwołujących się do historii pozycji hosta, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać przyrostków czasu), lub
  - **#num** - maksymalny zakres oceny w ostatnio zebranych wartościach (jeśli poprzedzony znakiem hash)
  - **time shift** (opcjonalnie) umożliwia przesunięcie punktu oceny wstecz w czasie. Zobacz więcej szczegółów na temat określania time shift.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >

- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

avg(/host/key,(sec|#num)<:time shift>)

Średnia wartość pozycji w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz parametry wspólne.

Przesunięcie czasu jest przydatne, gdy zachodzi potrzeba porównania bieżącej średniej wartości ze średnią wartością sprzed pewnego czasu.

Przykłady:

```
avg(/host/key,1h) #oblicz średnią wartość z ostatniej godziny do teraz
avg(/host/key,#5) #oblicz średnią wartość z pięciu najnowszych wartości
avg(/host/key,1h:now-1d) #oblicz średnią wartość dla godziny od 25 godzin temu do 24 godzin temu względem
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"])) #oblicz średnie obciążenie procesora
avg(/host/proc.num,5m)>300 #wyzwalacz, jeśli średnia liczba procesów w ostatnich 5 minutach była większa n
```

bucket_percentile(item filter,time period,percentage)

Oblicza percentyl na podstawie kubełków histogramu.

Parametry:

- **item filter** - zobacz item filter;
- **time period** - zobacz time period;
- **percentage** - procent (0-100).

Komentarze:

- Obsługiwane tylko w pozycjach obliczanych;
- Ta funkcja jest aliasem dla `histogram_quantile(percentage/100, bucket_rate_foreach(item filter, time period, 1))`.

Przykłady:

```
bucket_percentile(/*/http_request_duration_seconds_bucket[*],5m,95) #oblicz 95. percentyl czasu odpowiedzi
bucket_percentile(/*/apiserver_request_duration_seconds_bucket[*],10m,99) #oblicz 99. percentyl opóźnienia
```

count(func_foreach(item filter,<time period>),<operator>,<pattern>)

Liczba wartości w tablicy zwróconej przez funkcję foreach.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry:

- **func_foreach** - funkcja foreach, dla której należy policzyć liczbę zwróconych wartości. Szczegóły znajdują się w opisie funkcji foreach. Należy pamiętać, że count_foreach i bucket_rate_foreach obsługują dodatkowe parametry.
- **item filter** - zobacz składnię filtra pozycji;
- **time period** - zobacz okres czasu;
- **operator** (musi być ujęty w podwójne cudzysłowy). Obsługiwane `operators`:
eq - równe
ne - nierówne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like - dopasowuje, jeśli zawiera wzorzec (uwzględnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie wyrażenia regularnego podanego w `pattern` z uwzględnieniem wielkości liter
iregexp - dopasowanie wyrażenia regularnego podanego w `pattern` bez uwzględniania wielkości liter
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójne cudzysłowy); obsługiwany, jeśli w trzecim parametrze określono *operator*.

Uwagi:

- Użycie **count()** z funkcją foreach opartą na historii (max_foreach, avg_foreach itp.) może mieć wpływ na wydajność, natomiast użycie **exists_foreach()**, która działa wyłącznie na danych konfiguracyjnych, nie będzie miało takiego efektu.
- Opcjonalnych parametrów *operator* lub *pattern* nie można pozostawić pustych po przecinku — można je jedynie całkowicie pominąć.
- Przy *bitand* jako trzecim parametrze czwarty parametr `pattern` może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. count() oblicza operację "bitwise AND" dla wartości i *mask* oraz porównuje wynik z *number_to_compare_with*. Jeśli wynik "bitwise AND" jest równy *number_to_compare_with*, wartość jest liczona.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').

- Przy *regexp* lub *iregexp* jako trzecim parametrze czwarty parametr `pattern` może być zwykłym lub globalnym wyrażeniem regularnym (zaczynającym się od '@'). W przypadku globalnych wyrażeń regularnych uwzględnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego. Na potrzeby dopasowania regexp wartości zmiennoprzecinkowe będą zawsze reprezentowane z 4 cyframi po '.'. Należy też pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Przykłady:

```
count(max_foreach(/*/net.if.in[*],1h)) #oblicza liczbę pozycji net.if.in, które otrzymały dane w ostatniej
count(last_foreach(/*/vfs.fs.size[*,pused]),"gt",95) #oblicza liczbę systemów plików, w których wykorzysta
```

histogram_quantile(quantile,bucket1,value1,bucket2,value2,...)

Oblicza kwantyl φ na podstawie kubełków histogramu.
 Obsługiwane foreach function: *bucket_rate_foreach*.

Parametry:

- **quantile** - 0 ≤ φ ≤ 1;
- **bucketN, valueN** - ręcznie wprowadzone pary (>=2) parametrów lub odpowiedź funkcji bucket_rate_foreach.

Uwagi:

- Obsługiwane tylko w pozycjach obliczanych;
- Funkcjonalnie odpowiada 'histogram_quantile' z PromQL;
- Zwraca -1, jeśli wartości ostatniego kubełka 'Infinity' ("+*inf*") są równe 0.

Przykłady:

```
histogram_quantile(0.75,1.0,last(/host/rate_bucket[1.0]),"+Inf",last(/host/rate_bucket[Inf]))
histogram_quantile(0.5,bucket_rate_foreach(//item_key,30s))
```

item_count(item filter)

Liczba istniejących pozycji w konfiguracji, które spełniają kryteria filtra.
 Obsługiwany typ wartości: *Integer*.

Parametr:

- **item filter** - kryteria wyboru pozycji, umożliwiające odwołanie się przez grupę hostów, host, klucz pozycji i tagi. Obsługiwane są symbole wieloznaczne. Więcej informacji można znaleźć w sekcji item filter.

Uwagi:

- Obsługiwane tylko w pozycjach obliczanych;
- Działa jako alias funkcji *count(exists_foreach(item_filter))*.

Przykłady:

```
item_count(/*/agent.ping?[group="Host group 1"]) #oblicz liczbę hostów z pozycją agent.ping w "Host group
```

kurtosis(/host/key,(sec|#num)<:time shift>)

"Ogoniastość" rozkładu prawdopodobieństwa w zebranych wartościach w określonym okresie oceny. Zobacz także: Kurtosis.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz parametry wspólne.

Przykłady:

```
kurtosis(/host/system.cpu.util[,user],5m)>5 #trigger on sharp spikes of CPU utilization over 5 minutes
```

mad(/host/key,(sec|#num)<:time shift>)

Mediana odchyleń bezwzględnych w zebranych wartościach w zdefiniowanym okresie oceny. Zobacz także: Mediana odchyleń bezwzględnych.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz parametry wspólne.

Przykłady:

```
last(/host/system.cpu.util[,user])>avg(/host/system.cpu.util[,user],1h)+3*mad(/host/system.cpu.util[,user]
```

max(/host/key,(sec|#num)<:time shift>)

Najwyższa wartość pozycji w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
max(/host/key,1h) - min(/host/key,1h) #oblicza różnicę między maksymalną i minimalną wartością z ostatniej
max(last_foreach(/*/vfs.fs.size[*,pused]?[group="Linux servers"])) #zwraca najwyższe użycie dysku spośród
```

min(/host/key,(sec|#num)<:time shift>)

Najniższa wartość pozycji w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
max(/host/key,1h) - min(/host/key,1h) #oblicza różnicę między maksymalną i minimalną wartością w ciągu ost
min(last_foreach(/*/vfs.file.contents["/sys/class/net/enp0s3/operstate"]?[group="Linux servers"])) #zwraca
```

skewness(/host/key,(sec|#num)<:przesunięcie czasowe>)

Asymetria rozkładu prawdopodobieństwa w zebranych wartościach w określonym okresie oceny. Zobacz także: Skewness.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz parametry wspólne.

Przykłady:

```
skewness(/host/vm.memory.size[used],5m)<-20 #wyzwalacz na nagłe ujemne spadki wskazujące na awarie/ponowne
```

stddevpop(/host/key,(sec|#num)<:przesunięcie czasu>)

Odchylenie standardowe populacji w zebranych wartościach w zdefiniowanym okresie oceny. Zobacz także: Odchylenie standardowe.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
stddevpop(/host/system.cpu.util[,user],10m)>7 #wyzwalacz, jeśli użycie CPU mocno się waha (odchylenie stan
```

stddevsamp(/host/key,(sec|#num)<:time shift>)

Próbkowe odchylenie standardowe w zebranych wartościach w zdefiniowanym okresie oceny. Zobacz także: Odchylenie standardowe.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *last_foreach*.

Parametry: zobacz parametry wspólne.

Do działania tej funkcji wymagane są co najmniej dwie wartości danych.

Przykłady:

```
stddevsamp(/host/system.cpu.util[,user],10m)>7 #wyzwalacz, jeśli użycie CPU silnie się waha (próbkowe odch
```

sum(/host/key,(sec|#num)<:time shift>)

Suma zebranych wartości w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
sum(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #oblicza całkowity ruch przychodzący w sieci dl
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"])) #oblicza całkowitą przestrzeń dyskową d
sum(last_foreach(/*/net.dns[*,*,*])) #oblicza całkowitą liczbę pomyślnych sprawdzeń DNS
```

sumofsquares(/host/key,(sec|#num)<:time shift>)

Suma kwadratów w zebranych wartościach w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *last_foreach*.

Parametry: zobacz parametry wspólne.

Przykłady:

```
sumofsquares(/host/calculated.net.if.rate,1m) #suma kwadratów dla szybkości ruchu sieciowego (przychodzące
sumofsquares(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #suma kwadratów dla najnowszych wartoś
```

varpop(/host/key,(sec|#num)<:przesunięcie czasu>)

Wariancja populacji zebranych wartości w zdefiniowanym okresie oceny. Zobacz także: Variance.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
varpop(/host/system.cpu.util[,user],10m)>50 #wyzwalacz, jeśli użycie CPU mocno się waha (wariancja jest wi
```

varsamp(/host/key,(sec|#num)<:time shift>)

Wariancja próbki zebranych wartości w zdefiniowanym okresie oceny. Zobacz także: Wariancja.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Do działania tej funkcji wymagane są co najmniej dwie wartości danych.

Przykłady:

```
varsamp(/host/system.cpu.util[,user],10m)>50 #wyzwalacz, jeśli użycie CPU mocno się waha (wariancja próbki
```

Zobacz wszystkie obsługiwane funkcje.

1 Funkcje foreach

Przegląd

Funkcje foreach są używane w obliczeniach agregujących do zwracania jednej wartości agregowanej dla każdej pozycji wybranej przez użyty **filtr pozycji**. Zwracana jest tablica wartości.

> **Attention:**
> Funkcje foreach są obsługiwane tylko w pozycjach obliczanych jako część formuł obliczeń agregujących. Nie można ich wywoływać w zwykłych kluczach pozycji ani w wyzwalaczach poza tym kontekstem.

Na przykład funkcja `avg_foreach` zwróci tablicę wartości, gdzie każda wartość jest *średnią* wartością historyczną wybranej pozycji w określonym przedziale czasu.

Filtr pozycji jest częścią składni używanej przez funkcje foreach. W filtrze pozycji obsługiwane są symbole wieloznaczne, dzięki czemu wymagane pozycje można wybierać bardzo elastycznie.

Obsługiwane funkcje

Function	Description
avg_foreach	Zwraca średnią wartość dla każdej pozycji.
bucket_rate_foreach	Zwraca pary (górna granica kubełka, wartość szybkości) odpowiednie do użycia w funkcji histogram_quantile(), gdzie ”górna granica kubełka” jest wartością parametru klucza pozycji zdefiniowaną przez parametr <parameter number>.
count_foreach	Zwraca liczbę wartości dla każdej pozycji.
exists_foreach	Zwraca ’1’ dla każdej włączonej pozycji.
last_foreach	Zwraca ostatnią wartość dla każdej pozycji.
max_foreach	Zwraca maksymalną wartość dla każdej pozycji.
min_foreach	Zwraca minimalną wartość dla każdej pozycji.
sum_foreach	Zwraca sumę wartości dla każdej pozycji.

Składnia funkcji

Funkcje foreach obsługują dwa wspólne parametry: `filtr pozycji` (szczegóły poniżej) oraz `okres czasu`:

```
foreach_function(item filter,time period)
```

Na przykład:

```
avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m)
```

zwróci pięciominutową średnią dla każdej pozycji „mysql.qps” w grupie serwerów MySQL.

Należy pamiętać, że niektóre funkcje obsługują dodatkowe parametry.

Składnia filtra pozycji

Filtr pozycji:

`/host/key[parameters]?[conditions]`

składa się z czterech części, gdzie:

- host - nazwa hosta
- key - klucz pozycji (bez parametrów)
- parameters - parametry klucza pozycji
- conditions - warunki oparte na grupie hostów i/lub tagach pozycji (w postaci wyrażenia)

Spacje są dozwolone wyłącznie wewnątrz wyrażenia warunków.

Użycie symbolu wieloznacznego

- Symbol wieloznaczny może być użyty do zastąpienia nazwy hosta, klucza pozycji lub pojedynczego parametru klucza pozycji.
- Albo host, albo klucz pozycji musi być określony bez symbolu wieloznacznego. Dlatego `/host/*` i `/*/key` są poprawnymi filtrami, ale `/*/*` jest nieprawidłowy.
- Symbol wieloznaczny nie może być użyty dla *części* nazwy hosta, klucza pozycji ani parametru klucza pozycji.
- Symbol wieloznaczny nie dopasowuje więcej niż jednego parametru klucza pozycji. Dlatego symbol wieloznaczny musi być określony osobno dla każdego parametru (tj. `key[abc,*,*]`).

Wyrażenie warunków

Wyrażenie warunków obsługuje:

- operandy:
  - group - grupa hostów
  - tag - tag pozycji
  - `"<text>"` - stała tekstowa, ze znakiem ucieczki \ do poprzedzania " i \
- operatory porównania ciągów znaków z rozróżnianiem wielkości liter: =, <>
- operatory logiczne: `and`, `or`, `not`
- grupowanie za pomocą nawiasów: ( )

Ujęcie stałych tekstowych w cudzysłów jest obowiązkowe. Obsługiwane jest tylko pełne porównanie ciągów znaków z rozróżnianiem wielkości liter.

Warning:
Podczas określania tagów w filtrze (tj. `tag="tagname:value"`), dwukropek ":" jest używany jako separator. Wszystko po nim jest traktowane jako wartość tagu. Dlatego obecnie nie jest obsługiwane określanie nazwy tagu zawierającej ":".

Przykłady

Można użyć złożonego filtra odwołującego się do klucza pozycji, grupy hostów i tagów, co ilustrują poniższe przykłady:

Przykład składni	Opis
`/host/key[abc,*]`	Dopasowuje podobne pozycje na tym hoście.
`/*/key`	Dopasowuje tę samą pozycję dowolnego hosta.
`/*/key?[group="ABC" and tag="tagname:value"]`	Dopasowuje tę samą pozycję dowolnego hosta z grupy ABC mającego tagi 'tagname:value'.
`/*/key[a,*,c]?[(group="ABC" and tag="Tag1") or (group="DEF" and (tag="Tag2" or tag="Tag3:value"))]`	Dopasowuje podobne pozycje dowolnego hosta z grupy ABC lub DEF z odpowiednimi tagami.

Wszystkie pozycje, do których istnieją odwołania, muszą istnieć i zbierać dane. W obliczeniach uwzględniane są tylko włączone pozycje na włączonych hostach. Pozycje w stanie nieobsługiwanym nie są uwzględniane.

Attention:
Jeśli klucz pozycji, do której istnieje odwołanie, zostanie zmieniony, filtr musi zostać zaktualizowany ręcznie.

Określenie nadrzędnej grupy hostów obejmuje grupę nadrzędną oraz wszystkie zagnieżdżone grupy hostów wraz z ich pozycjami.

Okres czasu

Drugi parametr pozwala określić okres czasu dla agregacji. Okres czasu może być wyrażony wyłącznie jako czas; liczba wartości (poprzedzona prefiksem **#**) nie jest obsługiwana.

Dla wygody w tym parametrze można używać obsługiwanych symboli jednostek, na przykład „5m” (pięć minut) zamiast „300s” (300 sekund) lub „1d” (jeden dzień) zamiast „86400” (86400 sekund).

W przypadku funkcji *last_foreach* okres czasu jest parametrem opcjonalnym (obsługiwanym od Zabbix 7.0), który można pominąć:

```
last_foreach(/*/key?[group="host group"])
```

Okres czasu nie jest obsługiwany przez funkcję *exists_foreach*.

Dodatkowe parametry

bucket_rate_foreach

Trzeci opcjonalny parametr jest obsługiwany przez funkcję *bucket_rate_foreach*:

```
bucket_rate_foreach(item filter,time period,<parameter number>)
```

gdzie <parameter number> to pozycja wartości „bucket” w kluczu pozycji. Na przykład, jeśli wartość „bucket” w myItem[aaa,0.2] to „0.2”, to jej pozycja wynosi 2.

Domyślna wartość <parameter number> to „1”.

count_foreach

Trzeci i czwarty opcjonalny parametr są obsługiwane przez funkcję *count_foreach*:

```
count_foreach(item filter,time period,<operator>,<pattern>)
```

Gdzie:

- **operator** to operator warunkowy dla wartości pozycji (musi być ujęty w podwójny cudzysłów). Obsługiwane `operators`:
eq - równe
ne - nierówne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like - pasuje, jeśli zawiera wzorzec (z uwzględnieniem wielkości liter)
bitand - bitowe AND
regexp - dopasowanie z uwzględnieniem wielkości liter do wyrażenia regularnego podanego w `pattern`
iregexp - dopasowanie bez uwzględniania wielkości liter do wyrażenia regularnego podanego w `pattern`
- **pattern** to wymagany wzorzec (argumenty tekstowe muszą być ujęte w podwójny cudzysłów); obsługiwany, jeśli *operator* jest określony w trzecim parametrze.

Uwagi:

- Opcjonalne parametry *operator* lub *pattern* nie mogą być pozostawione puste po przecinku, mogą być tylko całkowicie pominięte.
- Gdy *bitand* jest użyty jako trzeci parametr, czwarty parametr `pattern` może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. count_foreach() oblicza „bitowe AND” z wartości i *mask* oraz porównuje wynik z *number_to_compare_with*. Jeśli wynik „bitowego AND” jest równy *number_to_compare_with*, wartość jest zliczana.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Gdy *regexp* lub *iregexp* jest użyty jako trzeci parametr, czwarty parametr `pattern` może być zwykłym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnych wyrażeń regularnych. Na potrzeby dopasowania regexp wartości zmiennoprzecinkowe będą zawsze przedstawiane z 4 cyframi dziesiętnymi po '.'. Należy również pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę dziesiętną.

Więcej szczegółów i przykładów użycia funkcji foreach można znaleźć w sekcji obliczenia agregowane.

Zachowanie zależne od dostępności

Poniższa tabela ilustruje, jak każda funkcja zachowuje się w przypadkach ograniczonej dostępności hosta/pozycji oraz danych historycznych.

Function	Disabled host	Unavailable host with data	Unavailable host without data	Disabled item	Unsupported item	Data retrieval error (SQL)
avg_foreach	ignoruj	zwróć średnią	ignoruj	ignoruj	ignoruj	ignoruj
bucket_rate_foreach	ignoruj	zwróć bucket rate	ignoruj	ignoruj	ignoruj	ignoruj
count_foreach	ignoruj	zwróć liczbę	0	ignoruj	ignoruj	ignoruj
exists_foreach	ignoruj	1	1	ignoruj	1	n/d
last_foreach	ignoruj	zwróć ostatnią wartość	ignoruj	ignoruj	ignoruj	ignoruj
max_foreach	ignoruj	zwróć maksimum	ignoruj	ignoruj	ignoruj	ignoruj

Function	Disabled host	Unavailable host with data	Unavailable host without data	Disabled item	Unsupported item	Data retrieval error (SQL)
min_foreach	ignoruj	zwróć minimum	ignoruj	ignoruj	ignoruj	ignoruj
sum_foreach	ignoruj	zwróć sumę	ignoruj	ignoruj	ignoruj	ignoruj

Jeśli pozycja jest *ignorowana*, nic nie jest dodawane do agregacji.

2 Funkcje bitowe

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
bitand	Wartość ”bitowego AND” wartości pozycji i maski.
bitlshift	Bitowe przesunięcie w lewo wartości pozycji.
bitnot	Wartość ”bitowego NOT” wartości pozycji.
bitor	Wartość ”bitowego OR” wartości pozycji i maski.
bitrshift	Bitowe przesunięcie w prawo wartości pozycji.
bitxor	Wartość ”bitowego wyłącznego OR” wartości pozycji i maski.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Wyrażenia są akceptowane jako parametry
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >

bitand(value,mask)

Wartość operacji „bitwise AND” wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** — wartość do sprawdzenia;
- **mask** (wymagane) — 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu wykonuje się przez porównanie do 4, a nie 100.

Przykłady:

```
bitand(last(/host/key),12)=8 or bitand(last(/host/key),12)=4 #ustawiony 3. lub 4. bit, ale nie oba jednocz
bitand(last(/host/key),20)=16 #3. bit nie jest ustawiony, a 5. bit jest ustawiony
```

bitlshift(value,bits to shift)

Przesunięcie bitowe w lewo wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **bits to shift** (wymagane) - liczba bitów do przesunięcia.

Chociaż porównanie jest wykonywane w sposób bitowy, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie do 4, a nie 100.

bitnot(value)

Wartość operacji „bitwise NOT” dla wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametr:

- **value** — wartość do sprawdzenia.

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z 4, a nie 100.

bitor(value,mask)

Wartość operacji „bitwise OR” dla wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **mask** (wymagane) - 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu wykonuje się przez porównanie z 4, a nie 100.

bitrshift(value,bits to shift)

Przesunięcie bitowe w prawo wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **bits to shift** (wymagane) - liczba bitów do przesunięcia.

Chociaż porównanie jest wykonywane w sposób bitowy, wszystkie wartości muszą być podane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z 4, a nie 100.

bitxor(value,mask)

Wartość operacji „bitowe wykluczające LUB” dla wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **mask** (wymagany) - 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z wartością 4, a nie 100.

Zobacz wszystkie obsługiwane funkcje.

3 Funkcje daty i czasu

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Attention:
Funkcji daty i czasu nie można używać samodzielnie w wyrażeniu; musi ono zawierać co najmniej jedną funkcję z innej grupy, odwołującą się do pozycji hosta (z wyjątkiem funkcji nodata()). Szczegółowe informacje o tym, jak funkcje daty i czasu działają w wyrażeniach, można znaleźć w sekcji Czas obliczeń.

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
date	Bieżąca data w formacie YYYYMMDD.
dayofmonth	Dzień miesiąca w zakresie od 1 do 31.
dayofweek	Dzień tygodnia w zakresie od 1 do 7.
now	Liczba sekund od Epoki (00:00:00 UTC, 1 stycznia 1970).
time	Bieżący czas w formacie HHMMSS.

Szczegóły funkcji

date

Bieżąca data w formacie YYYYMMDD.

Przykład:

`date()<20220101`

dayofmonth

Dzień miesiąca w zakresie od 1 do 31.

Przykład:

`dayofmonth()=1`

dayofweek

Dzień tygodnia w zakresie od 1 do 7 (pon. - 1, niedz. - 7).

Przykład (tylko dni robocze):

`dayofweek()<6`

Przykład (tylko weekend):

`dayofweek()>5`

now

Liczba sekund od Epoki (00:00:00 UTC, 1 stycznia 1970).

Przykład:

`now()<1640998800`

time

Bieżący czas w formacie HHMMSS.

Przykład (tylko pora nocna, 00:00-06:00):

`time()<060000`

Zobacz wszystkie obsługiwane funkcje.

4 Funkcje historyczne

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
change	Wielkość różnicy między poprzednią a najnowszą wartością.
changecount	Liczba zmian między sąsiednimi wartościami w zdefiniowanym okresie oceny.
count	Liczba wartości w zdefiniowanym okresie oceny.
countunique	Liczba unikalnych wartości w zdefiniowanym okresie oceny.
find	Znajdź dopasowanie wartości w zdefiniowanym okresie oceny.
first	Pierwsza (najstarsza) wartość w zdefiniowanym okresie oceny.
firstclock	Znacznik czasu pierwszej (najstarszej) wartości w zdefiniowanym okresie oceny.
fuzzytime	Sprawdź, o ile czas pasywnego agenta różni się od czasu serwera/proxy Zabbixa.
last	Najnowsza wartość.
lastclock	Znacznik czasu N-tej najnowszej wartości w zdefiniowanym okresie oceny.
logeventid	Sprawdź, czy identyfikator zdarzenia ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
logseverity	Poziom ważności dziennika ostatniego wpisu dziennika.
logsource	Sprawdź, czy źródło dziennika ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
logtimestamp	Znacznik czasu komunikatu dziennika dla N-tej najnowszej wartości pozycji dziennika.
monodec	Sprawdź, czy nastąpił monotoniczny spadek wartości.
monoinc	Sprawdź, czy nastąpił monotoniczny wzrost wartości.
nodata	Sprawdź, czy nie odebrano danych.
percentile	P-ty percentyl okresu, gdzie P (procent) jest określony przez trzeci parametr.
rate	Średnie tempo wzrostu na sekundę monotonicznie rosnącego licznika w zdefiniowanym przedziale czasu.

Wspólne parametry

- `/host/key` jest wspólnym obowiązkowym pierwszym parametrem dla funkcji odwołujących się do historii pozycji hosta
- `(sec|#num)<:time shift>` jest wspólnym drugim parametrem dla funkcji odwołujących się do historii pozycji hosta, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać sufiksów czasu), lub
  - **#num** - maksymalny zakres oceny w najnowszych zebranych wartościach (jeśli poprzedzony znakiem hash)
  - **time shift** (opcjonalnie) umożliwia przesunięcie punktu oceny wstecz w czasie. Zobacz więcej szczegółów dotyczących określania time shift.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone za pomocą < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

change(/host/key)

Wielkość różnicy między poprzednią a najnowszą wartością.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.
 Dla ciągów znaków zwraca: 0 - wartości są równe; 1 - wartości są różne.

Parametry: zobacz common parameters.

Komentarze:

- Różnica liczbowa zostanie obliczona, jak widać na poniższych przykładowych wartościach wejściowych ('previous' i 'latest' value = różnica):
'1' i '5' = +4
'3' i '1' = −2
'0' i '-2.5' = −2.5.
- Zobacz także: abs do porównania.

Przykłady:

```
change(/host/system.uptime)<0 #zmiana uptime systemu była ujemna od ostatniej wartości (co wskazuje na res
change(/host/system.cpu.load[all,avg1])>2 #obciążenie CPU (dla jednej minuty) wzrosło o więcej niż 2 od os
change(/host/vfs.fs.size[/,free])<-1G #wolna przestrzeń dyskowa spadła o więcej niż 1 GB między sprawdzeni
```

changecount(/host/key,(sec|#num)<:time shift>,<mode>)

Liczba zmian między sąsiednimi wartościami w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz common parameters
- **mode** (musi być ujęty w podwójne cudzysłowy) - możliwe wartości: *all* - zliczaj wszystkie zmiany (domyślnie); *dec* - zliczaj spadki; *inc* - zliczaj wzrosty

Dla nieliczbowych typów wartości parametr *mode* jest ignorowany.

Przykłady:

```
changecount(/host/icmpping,10m)>5 #status ping zmienił się więcej niż 5 razy w ciągu 10 minut
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5 #stan operacyjny interfejsu eth
changecount(/host/proc.num[httpd],15m)>10 #liczba procesów httpd zmieniła się więcej niż 10 razy w ciągu 1
changecount(/host/key,#10,"inc") #liczba wzrostów wartości (w odniesieniu do sąsiedniej wartości) spośród
changecount(/host/key,24h,"dec") #liczba spadków wartości (w odniesieniu do sąsiedniej wartości) z ostatni
```

count(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Liczba wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **operator** (musi być ujęty w podwójny cudzysłów) Obsługiwane `operatory`:
eq - równe (domyślnie dla integer, float)
ne - nierówne
gt - większe niż
ge - większe lub równe
lt - mniejsze niż
le - mniejsze lub równe
like (domyślnie dla string, text, log) - pasuje, jeśli zawiera wzorzec (z uwzględnieniem wielkości liter)
bitand - bitowe AND
regexp - dopasowanie z uwzględnieniem wielkości liter do wyrażenia regularnego podanego w `pattern`
iregexp - dopasowanie bez uwzględniania wielkości liter do wyrażenia regularnego podanego w `pattern`
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójny cudzysłów)

Komentarze:

- Pozycje typu float są dopasowywane z precyzją 2.22e-16.
- *like* nie jest obsługiwany jako operator dla wartości integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości float.
- Dla wartości string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* i *iregexp*.
- Gdy jako operator używany jest *bitand*, czwarty parametr `pattern` można określić jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. count() oblicza „bitowe AND" z wartości i *mask* oraz porównuje wynik z *number_to_compare_with*. Jeśli wynik „bitowego AND" jest równy *number_to_compare_with*, wartość jest zliczana.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Gdy jako operator używany jest *regexp* lub *iregexp*, czwarty parametr `pattern` może być zwykłym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnych wyrażeń regularnych. Na potrzeby dopasowania regexp wartości float będą zawsze przedstawiane z 4 cyframi dziesiętnymi po '.'. Należy również pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Przykłady:

```
count(/host/icmpping,30m,,"0")>5 #ping has failed more than 5 times in 30 minutes
count(/host/key,10m,"like","error") #the number of values for the last 10 minutes until now that contain '
count(/host/key,10m,,12) #the number of values for the last 10 minutes until now that equal '12'
count(/host/key,10m,"gt",12) #the number of values for the last 10 minutes until now that are over '12'
count(/host/key,#10,"gt",12) #the number of values within the last 10 values until now that are over '12'
count(/host/key,10m:now-1d,"gt",12) #the number of values between 24 hours and 10 minutes and 24 hours ago
count(/host/key,10m,"bitand","6/7") #the number of values for the last 10 minutes until now having '110' (
count(/host/key,10m:now-1d) #the number of values between 24 hours and 10 minutes and 24 hours ago from no
```

countunique(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Liczba unikalnych wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **operator** (musi być ujęty w podwójny cudzysłów). Obsługiwane operators:
eq - równe (domyślnie dla integer, float)
ne - nierówne
gt - większe niż
ge - większe lub równe
lt - mniejsze niż
le - mniejsze lub równe
like (domyślnie dla string, text, log) - pasuje, jeśli zawiera wzorzec (z rozróżnianiem wielkości liter)
bitand - bitowe AND
regexp - dopasowanie z rozróżnianiem wielkości liter do wyrażenia regularnego podanego w `pattern`
iregexp - dopasowanie bez rozróżniania wielkości liter do wyrażenia regularnego podanego w `pattern`
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójny cudzysłów)

Uwagi:

- Pozycje typu float są dopasowywane z precyzją 2.22e-16.
- *like* nie jest obsługiwany jako operator dla wartości integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości float.
- Dla wartości string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* oraz *iregexp*.
- Przy użyciu *bitand* jako operatora, czwarty parametr `pattern` można określić jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. countunique() oblicza „bitowe AND" z wartości i *mask*, a następnie porównuje wynik z *number_to_compare_with*. Jeśli wynik „bitowego AND" jest równy *number_to_compare_with*, wartość jest zliczana.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Przy użyciu *regexp* lub *iregexp* jako operatora, czwarty parametr `pattern` może być zwykłym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnych wyrażeń regularnych. Na potrzeby dopasowania regexp wartości float będą zawsze przedstawiane z 4 cyframi dziesiętnymi po '.'. Należy również pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę dziesiętną.

Przykłady:

```
countunique(/host/key,10m) #liczba unikalnych wartości z ostatnich 10 minut do teraz
countunique(/host/key,10m,"like","error") #liczba unikalnych wartości z ostatnich 10 minut do teraz, które
countunique(/host/key,10m,,12) #liczba unikalnych wartości z ostatnich 10 minut do teraz równych '12'
countunique(/host/key,10m,"gt",12) #liczba unikalnych wartości z ostatnich 10 minut do teraz większych niż
countunique(/host/key,#10,"gt",12) #liczba unikalnych wartości wśród ostatnich 10 wartości do teraz większ
countunique(/host/key,10m:now-1d,"gt",12) #liczba unikalnych wartości między 24 godzinami i 10 minutami te
countunique(/host/key,10m,"bitand","6/7") #liczba unikalnych wartości z ostatnich 10 minut do teraz mający
countunique(/host/key,10m:now-1d) #liczba unikalnych wartości między 24 godzinami i 10 minutami temu a 24
```

find(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Znajduje dopasowanie wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.
 Zwraca: 1 - znaleziono; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **sec** lub **#num** (opcjonalne) - jeśli nie podano, domyślnie używana jest najnowsza wartość
- **operator** (musi być ujęty w podwójny cudzysłów) Obsługiwane `operatory`:
eq - równe (domyślnie dla integer, float)
ne - nierówne
gt - większe niż
ge - większe lub równe
lt - mniejsze niż
le - mniejsze lub równe
like (domyślnie dla string, text, log) - dopasowuje, jeśli zawiera ciąg znaków podany w `pattern` (z uwzględnieniem wielkości liter)
bitand - bitowe AND
regexp - dopasowanie z uwzględnieniem wielkości liter do wyrażenia regularnego podanego w `pattern`
iregexp - dopasowanie bez uwzględniania wielkości liter do wyrażenia regularnego podanego w `pattern`
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójny cudzysłów); wyrażenie regularne Perl Compatible Regular Expression (PCRE), jeśli `operator` to *regexp*, *iregexp*

Uwagi:

- Jeśli przetwarzana jest więcej niż jedna wartość, zwracane jest '1', jeśli istnieje co najmniej jedna pasująca wartość.
- *like* nie jest obsługiwany jako operator dla wartości integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości float.
- Dla wartości string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* i *iregexp*.
- Gdy jako operator użyto *regexp* lub *iregexp*, czwarty parametr `pattern` może być zwykłym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego.

Przykłady:

```
find(/host/key,10m,"like","error") #znajdź wartość zawierającą 'error' w ciągu ostatnich 10 minut do teraz
find(/host/agent.version,,"like","beta")=1 #agent Zabbix ma wersję beta, musi zostać zaktualizowany
find(/host/log[/var/log/nginx/access.log],,"regexp"," 500 ")=1 #znaleziono wewnętrzny błąd serwera WWW
```

first(/host/key,sec<:time shift>)

Pierwsza (najstarsza) wartość w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry

Zobacz także last().

Przykład:

```
first(/host/key,1h) #retrieve the oldest value within the last hour until now
```

firstclock(/host/key,sec<:time shift>)

Znacznik czasu najstarszej wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry.

Funkcja kończy się błędem, jeśli w podanym okresie nie zebrano żadnych danych.

Zobacz także lastclock().

Przykłady:

```
firstclock(/host/key,1h) #pobierz znacznik czasu najstarszej wartości z ostatniej godziny
firstclock(/host/key,1h:now-24h) #pobierz znacznik czasu najstarszej wartości z ostatniej godziny sprzed d
```

fuzzytime(/host/key,sec)

Sprawdza, jak bardzo czas pasywnego agenta różni się od czasu serwera/proxy Zabbix.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Zwraca: 1 - różnica między wartością pasywnej pozycji (jako znacznik czasu) a znacznikiem czasu serwera/proxy Zabbix (czasem pobrania wartości) jest mniejsza lub równa *sec* sekund; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry.

Komentarze:

- Zwykle używane z pozycją 'system.localtime' do sprawdzenia, czy czas lokalny jest zsynchronizowany z czasem lokalnym serwera Zabbix. *Uwaga*: dla agenta Zabbix pozycja 'system.localtime' musi być skonfigurowana jako sprawdzenie pasywne; w Zabbix agent 2 może być skonfigurowana jako sprawdzenie aktywne.
- Może być również używane z kluczem `vfs.file.time[/path/file,modify]` do sprawdzenia, czy plik nie był aktualizowany przez długi czas.
- Ta funkcja nie jest zalecana do użycia w złożonych wyrażeniach wyzwalaczy (z wieloma pozycjami), ponieważ może powodować nieoczekiwane wyniki (różnica czasu będzie mierzona względem najnowszej metryki), np. w `fuzzytime(/Host/system.l or last(/Host/trap)<>0`.

Przykłady:

```
fuzzytime(/host/system.localtime,5m)=0 #czas lokalny klienta różni się od czasu serwera/proxy Zabbix o wię
fuzzytime(/host/system.localtime,5m)=0 and nodata(/host/system.localtime,10m)=0 #czas lokalny klienta różn
```

last(/host/key,<#num<:time shift»)

Najnowsza wartość.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **#num** (opcjonalnie) — N-ta najnowsza wartość

Uwagi:

- Należy zwrócić uwagę, że okres czasu oznaczony hashem (#N) działa tutaj inaczej niż w przypadku wielu innych funkcji. Na przykład: `last(/host/key)` jest zawsze równe `last(/host/key,#1)`; `last(/host/key,#3)` — trzecia najnowsza wartość (*nie* trzy najnowsze wartości).
- Zabbix nie gwarantuje dokładnej kolejności wartości, jeśli w historii istnieje więcej niż dwie wartości w ciągu jednej sekundy.
- Zobacz także first().

Przykłady:

```
last(/host/key) #retrieve the last value
last(/host/key,#2) #retrieve the previous value
last(/host/key,#1)<>last(/host/key,#2) #last two values differ
```

lastclock(/host/key,<#num<:time shift»)

Znacznik czasu N-tej najnowszej wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry;
- **#num** (opcjonalnie) — N-ta najnowsza wartość.

Funkcja kończy się błędem, jeśli w podanym okresie nie zebrano żadnych danych lub nie zebrano N-tej wartości.

Zobacz także firstclock().

Przykłady:

```
lastclock(/host/key) #pobierz znacznik czasu najnowszej wartości
lastclock(/host/key,#2) #pobierz znacznik czasu 2. najnowszej wartości
now()-lastclock(/host/heartbeat.item)>300 #minęło więcej niż 5 minut od aktualizacji pozycji heartbeat
lastclock(/host/system.cpu.load[all,avg1])-lastclock(/host/system.cpu.load[all,avg1],#2)>300 #interwał akt
((now()-lastclock(/host/system.cpu.load[all,avg1])<120 and last(/host/system.cpu.load[all,avg1])>5)) #wart
```

logeventid(/host/key,<#num<:time shift»,<pattern>)

Sprawdza, czy identyfikator zdarzenia ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - nie pasuje; 1 - pasuje.

Parametry:

- Zobacz common parameters
- **#num** (opcjonalnie) - N-ta najnowsza wartość
- **pattern** (opcjonalnie) - wyrażenie regularne opisujące wymagany wzorzec, w stylu Perl Compatible Regular Expression (PCRE) (argumenty tekstowe muszą być ujęte w podwójne cudzysłowy)

Przykłady:

```
logeventid(/host/eventlog[Security],,"^4625$")=1 #wpis dziennika z identyfikatorem pasującym do "4625" (ni
logeventid(/host/eventlog[System],,"^(6008|41)$")=1 #wpis dziennika z identyfikatorem pasującym do "6008"
```

logseverity(/host/key,<#num<:time shift»)

Poziom ważności ostatniego wpisu logu.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - domyślny poziom ważności; N - poziom ważności (liczba całkowita, przydatne dla dzienników zdarzeń Windows: 1 - Informacja, 2 - Ostrzeżenie, 4 - Błąd, 7 - Audyt niepowodzeń, 8 - Audyt powodzeń, 9 - Krytyczny, 10 - Szczegółowy).

Parametry:

- Zobacz wspólne parametry
- **#num** (opcjonalnie) - N-ta najnowsza wartość

Zabbix pobiera poziom ważności logu z pola **Information** dziennika zdarzeń Windows.

Przykłady:

```
logseverity(/host/log[/var/log/syslog],10m)>3 #a log entry with severity above "3" found
logseverity(/host/eventlog[System],10m)=4 #a log entry with severity equaling "Error" found
```

logsource(/host/key,<#num<:time shift»,<pattern>)

Sprawdza, czy źródło logu ostatniego wpisu logu pasuje do wyrażenia regularnego.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - nie pasuje; 1 - pasuje.

Parametry:

- Zobacz common parameters
- **#num** (opcjonalnie) - N-ta najnowsza wartość
- **pattern** (opcjonalnie) - wyrażenie regularne opisujące wymagany wzorzec, w stylu Perl Compatible Regular Expression (PCRE) (argumenty tekstowe muszą być ujęte w podwójne cudzysłowy)

Zwykle używane dla dzienników zdarzeń systemu Windows.

Przykłady:

```
logsource(/host/eventlog[Application],,"MSSQLSERVER")=1 #znaleziono wpis logu ze źródłem pasującym do "MSS
logsource(/host/eventlog[System],,"Service Control Manager")=1 #znaleziono wpis logu ze źródłem pasującym
logsource(/host/eventlog[System],,"Service Control Manager")=1 and logeventid(/host/eventlog[System],,"^70
```

logtimestamp(/host/key,<#num<:time shift»)

Znacznik czasu komunikatu dziennika dla N-tej najnowszej wartości pozycji dziennika.
 Obsługiwane typy wartości: *Log*.

Parametry:

- Zobacz wspólne parametry;
- **#num** (opcjonalnie) — N-ta najnowsza wartość.

Uwagi:

- Obliczanie przesunięcia czasu opiera się na zegarze wartości pozycji, a nie na znaczniku czasu komunikatu dziennika;
- Funkcja kończy się błędem, jeśli:
  - odebrano pozycję typu innego niż log;
  - w podanym okresie nie zebrano żadnych danych;
  - nie zebrano N-tej wartości;
  - komunikat dziennika nie zawiera wartości znacznika czasu.

Przykłady:

```
logtimestamp(/host/key) #retrieve the timestamp of the latest log message
logtimestamp(/host/key,#2) #retrieve the timestamp of the 2nd latest log message
logtimestamp(/host/key,#3:now-1d) #retrieve the timestamp of the 3rd latest log message a day ago
```

monodec(/host/key,(sec|#num)<:time shift>,<mode>)

Sprawdza, czy nastąpił monotoniczny spadek wartości.
 Obsługiwane typy wartości: *Integer*.
 Zwraca: 1 - jeśli wszystkie elementy w okresie czasu stale maleją; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry

- **mode** (musi być ujęty w podwójny cudzysłów) - *weak* (każda wartość jest mniejsza lub taka sama jak poprzednia; domyślnie) lub *strict* (każda wartość zmalała)

Przykłady:

```
monodec(/host/system.swap.size[all,free],60s) + monodec(/host2/system.swap.size[all,free],60s) + monodec(/
monodec(/host/proc.num[nginx],10m,"strict")=1 #liczba procesów nginx monotonicznie malała przez 10 minut
```

monoinc(/host/key,(sec|#num)<:time shift>,<mode>)

Sprawdza, czy wystąpił monotoniczny wzrost wartości.
 Obsługiwane typy wartości: *Integer*.
 Zwraca: 1 - jeśli wszystkie elementy w okresie czasu stale rosną; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **mode** (musi być ujęty w podwójny cudzysłów) - *weak* (każda wartość jest większa lub taka sama jak poprzednia; domyślnie) lub *strict* (każda wartość wzrosła)

Przykłady:

```
monoinc(/host/system.localtime,#3,"strict")=0 #lokalny czas systemowy nie wzrastał w sposób ciągły
monoinc(/host/vfs.dir.count[/mnt/data/logs],24h,"weak")=0 #wyzwalacz, jeśli liczba plików pozostawała bez
```

nodata(/host/key,sec,<mode>)

Sprawdza brak otrzymanych danych.
 Obsługiwane typy wartości: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Zwraca: 1 - jeśli w zdefiniowanym przedziale czasu nie otrzymano danych; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **sec** - okres nie powinien być krótszy niż 30 sekund, ponieważ proces synchronizacji historii oblicza tę funkcję tylko co 30 sekund; `nodata(/host/key,0)` jest niedozwolone
- **mode** - jeśli ustawiono na *strict* (w podwójnych cudzysłowach), ta funkcja będzie niewrażliwa na dostępność proxy (szczegóły w komentarzach)

Komentarze:

- wyzwalacze 'nodata' monitorowane przez proxy są domyślnie wrażliwe na dostępność proxy - jeśli proxy stanie się niedostępne, wyzwalacze 'nodata' nie uruchomią się natychmiast po przywróceniu połączenia, lecz pominą dane za opóźniony okres. Należy pamiętać, że dla pasywnych proxy tłumienie jest aktywowane, jeśli połączenie zostanie przywrócone po więcej niż 15 sekundach i nie mniej niż 2 sekundy później. Dla aktywnych proxy tłumienie jest aktywowane, jeśli połączenie zostanie przywrócone po więcej niż 15 sekundach. Aby wyłączyć wrażliwość na dostępność proxy, użyj trzeciego parametru, np.: `nodata(/host/key,5m,"strict")`; w takim przypadku funkcja uruchomi się, gdy tylko upłynie okres oceny (pięć minut) bez danych.
- Ta funkcja wyświetli błąd, jeśli w okresie określonym przez 1. parametr:
- nie ma danych i serwer Zabbix został uruchomiony ponownie
- nie ma danych i zakończono konserwację
- nie ma danych i pozycja została dodana lub ponownie włączona.
- Błędy są wyświetlane w kolumnie *Info* w konfiguracji wyzwalacza.
- Ta funkcja może nie działać prawidłowo, jeśli występują różnice czasu między serwerem Zabbix, proxy i agentem. Zobacz także: Wymaganie synchronizacji czasu.

Przykład:

```
nodata(/host/agent.ping,5m)=1 #wyzwalacz, jeśli przez 5 minut nie otrzymano danych od agenta Zabbix
```

percentile(/host/key,(sec|#num)<:time shift>,percentage)

P-ty percentyl okresu, gdzie P (percentage) jest określony przez trzeci parametr.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry
- **percentage** - liczba zmiennoprzecinkowa z zakresu od 0 do 100 (włącznie), z maksymalnie 4 cyframi po przecinku

Przykłady:

```
percentile(/host/net.if.in[eth0,bytes],1h,95)>1000000 #95. percentyl wejściowego ruchu sieciowego (bajty/s
percentile(/host/system.cpu.util,5m,95)>80 #95. percentyl procentowego wykorzystania CPU w czasie użytkown
percentile(/host/icmppingsec[192.168.0.2],15m,95)>0.15 #większość pomiarów opóźnienia jest poniżej 150 ms,
percentile(/host/net.if.in[eth0,bytes],1h,50) #oblicz 50. percentyl (wartość mediany) przychodzącego ruchu
(percentile(/host/net.if.in[eth0,bytes],1h,50)+percentile(/host/net.if.in[eth0,bytes],1h,51))/2 #oblicz do
```

rate(/host/key,sec<:przesunięcie czasu>)

Średnie tempo wzrostu na sekundę monotonicznie rosnącego licznika w zdefiniowanym przedziale czasu.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz common parameters

Funkcjonalnie odpowiada funkcji 'rate' w PromQL.

Przykłady:

```
rate(/host/key,30s) #jeśli monotoniczny wzrost w ciągu 30 sekund wynosi 20, ta funkcja zwraca 0.67.
rate(/host/net.if.in[eth0,bytes],5m)>500000 #tempo ruchu przychodzącego interfejsu na eth0 przekroczyło 50
rate(/host/app.requests.count,1m)>100 #licznik liczby żądań wzrósł do >100 żądań na sekundę w ostatniej mi
```

Zobacz wszystkie obsługiwane funkcje.

5 Funkcje trendów

Funkcje trendów, w przeciwieństwie do funkcji historii, używają danych trend do obliczeń.

Trendy przechowują godzinowe wartości agregowane. Funkcje trendów używają tych godzinowych średnich, dlatego są przydatne do długoterminowej analizy.

Wyniki funkcji trendów są buforowane, więc wielokrotne wywołania tej samej funkcji z tymi samymi parametrami pobierają informacje z bazy danych tylko raz. Pamięć podręczna funkcji trendów jest kontrolowana przez parametr serwera TrendFunctionCacheSize.

Wyzwalacze, które odwołują się **wyłącznie** do funkcji trendów, są oceniane raz na najmniejszy okres czasu w wyrażeniu. Na przykład wyzwalacz taki jak

```
trendavg(/host/key,1d:now/d) > 1 or trendavg(/host/key2,1w:now/w) > 2
```

będzie oceniany raz dziennie. Jeśli wyzwalacz zawiera zarówno funkcje trendów, jak i historii (lub daty i czasu oraz/lub nodata()), jest obliczany zgodnie ze zwykłymi zasadami.

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- wyrażeniach wyzwalaczy
- pozycjach obliczanych
- makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
baselinedev	Zwraca liczbę odchyleń (według algorytmu stddevpop) między ostatnim okresem danych a tymi samymi okresami danych w poprzednich sezonach.
baselinewma	Oblicza linię bazową przez uśrednianie danych z tego samego przedziału czasu w wielu równych okresach czasu („sezonach") przy użyciu algorytmu ważonej średniej ruchomej.
trendavg	Średnia wartości trendów w zdefiniowanym okresie czasu.
trendcount	Liczba pomyślnie pobranych wartości historii użytych do obliczenia wartości trendu w zdefiniowanym okresie czasu.
trendmax	Maksimum wartości trendów w zdefiniowanym okresie czasu.
trendmin	Minimum wartości trendów w zdefiniowanym okresie czasu.
trendstl	Zwraca współczynnik anomalii w okresie wykrywania - wartość dziesiętną między 0 a 1, równą `((liczba wartości anomalii)/(łączna liczba wartości))`.
trendsum	Suma wartości trendów w zdefiniowanym okresie czasu.

Wspólne parametry

- `/host/key` to wspólny obowiązkowy pierwszy parametr
- `time period:time shift` to wspólny drugi parametr, gdzie:
  - **time period** - okres czasu (minimum „1h"), zdefiniowany jako <N><time unit>, gdzie `N` - liczba jednostek czasu, `time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok).
  - **time shift** - przesunięcie okresu czasu (zobacz przykłady funkcji)

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczane za pomocą < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `time period:time shift` nigdy nie mogą być ujmowane w cudzysłów

baselinedev(/host/key,data period:time shift,season unit,num seasons)

Zwraca liczbę odchyleń (według algorytmu stddevpop) między ostatnim okresem danych a tymi samymi okresami danych w poprzednich sezonach.

Parametry:

- Zobacz wspólne parametry
- **data period** - okres zbierania danych w ramach sezonu, zdefiniowany jako <N><time unit>, gdzie:
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok), musi być równa sezonowi lub od niego mniejsza
- **season unit** - jednostka kalendarzowa definiująca jeden sezon (h, d, w, M, y); nie może być mniejsza niż data period
- **num seasons** - liczba sezonów do oceny

Przykłady:

```
baselinedev(/host/key,1d:now/d,"M",6) #obliczanie liczby odchyleń standardowych (populacyjnych) między pop
baselinedev(/host/key,1h:now/h,"d",10)  #obliczanie liczby populacyjnych odchyleń standardowych między pop
```

baselinewma(/host/key,data period:time shift,season unit,num seasons)

Oblicza linię bazową przez uśrednianie danych z tego samego przedziału czasowego w wielu równych okresach czasu („sezonach”) przy użyciu algorytmu ważonej średniej ruchomej.

Parametry:

- Zobacz wspólne parametry
- **data period** - okres zbierania danych w ramach sezonu, zdefiniowany jako <N><jednostka czasu>, gdzie:
`N` - liczba jednostek czasu
`jednostka czasu` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok), musi być równa sezonowi lub od niego mniejsza
Time shift - przesunięcie czasowe, określa przesunięcie okresu czasu, definiuje koniec przedziału zbierania danych w sezonach (zobacz przykłady)
- **season unit** - jednostka kalendarzowa definiująca jeden sezon (h, d, w, M, y); nie może być mniejsza niż data period
- **num seasons** - liczba sezonów do oceny

Przykłady:

```
baselinewma(/host/key,1h:now/h,"d",3) #obliczanie linii bazowej dla tej samej godziny dnia w ostatnich 3 p
baselinewma(/host/key,2h:now/h,"d",3) #obliczanie linii bazowej dla tego samego dwugodzinnego przedziału c
baselinewma(/host/key,1d:now/d,"M",4) #obliczanie linii bazowej dla tego samego dnia miesiąca co „wczoraj”
```

trendavg(/host/key,time period:time shift)

Średnia wartości trendów w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendavg(/host/key,1h:now/h) #średnia dla poprzedniej godziny (np. 12:00-13:00)
trendavg(/host/key,1h:now/h-1h) #średnia sprzed dwóch godzin (11:00-12:00)
trendavg(/host/key,1h:now/h-2h) #średnia sprzed trzech godzin (10:00-11:00)
trendavg(/host/key,1M:now/M-1y) #średnia dla poprzedniego miesiąca rok temu
```

trendcount(/host/key,time period:time shift)

Liczba pomyślnie pobranych wartości historii użytych do obliczenia wartości trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendcount(/host/key,1h:now/h) #liczba wartości dla poprzedniej godziny (np. 12:00-13:00)
trendcount(/host/key,1h:now/h-1h) #liczba wartości sprzed dwóch godzin (11:00-12:00)
trendcount(/host/key,1h:now/h-2h) #liczba wartości sprzed trzech godzin (10:00-11:00)
trendcount(/host/key,1M:now/M-1y) #liczba wartości dla poprzedniego miesiąca rok temu
```

trendmax(/host/key,time period:time shift)

Maksimum w wartościach trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendmax(/host/key,1h:now/h) #maksimum dla poprzedniej godziny (np. 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → oblicza różnicę między wartościami maksymaln
trendmax(/host/key,1h:now/h-1h) #maksimum sprzed dwóch godzin (11:00-12:00)
trendmax(/host/key,1h:now/h-2h) #maksimum sprzed trzech godzin (10:00-11:00)
trendmax(/host/key,1M:now/M-1y) #maksimum dla poprzedniego miesiąca rok temu
```

trendmin(/host/key,time period:time shift)

Minimum w wartościach trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendmin(/host/key,1h:now/h) #minimum dla poprzedniej godziny (np. 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → oblicza różnicę między wartościami maksymaln
trendmin(/host/key,1h:now/h-1h) #minimum sprzed dwóch godzin (11:00-12:00)
trendmin(/host/key,1h:now/h-2h) #minimum sprzed trzech godzin (10:00-11:00)
trendmin(/host/key,1M:now/M-1y) #minimum dla poprzedniego miesiąca rok temu
```

trendstl(/host/key,eval period:time shift,detection period,season,<deviations>,<devalg>,<s window>)

Zwraca współczynnik anomalii w okresie wykrywania — wartość dziesiętną z zakresu od 0 do 1, obliczaną jako `((liczba wartości anomalii)/(całkowita liczba wartości))`.

Parametry:

- Zobacz wspólne parametry
- **eval period** - okres czasu, który ma zostać poddany dekompozycji (minimum „1h”), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok)
- **detection period** - okres czasu przed końcem eval period, dla którego obliczane są anomalie (minimum „1h”, nie może być dłuższy niż eval period), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień)
- **season** - najkrótszy okres czasu, w którym oczekiwany jest powtarzający się wzorzec („sezon”) (minimum „2h”, nie może być dłuższy niż eval period, liczba wpisów w eval period musi być większa niż dwukrotność wynikowej częstotliwości (season/h)), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień)
- **deviations** - liczba odchyleń (obliczana przez devalg), przy której wartość jest uznawana za anomalię (może być dziesiętna), (musi być większa lub równa 1, domyślnie 3)
- **devalg** (musi być ujęty w podwójny cudzysłów) - algorytm odchylenia; może to być *stddevpop*, *stddevsamp* lub *mad* (domyślnie)
- **s window** - rozpiętość (w opóźnieniach) okna loess dla ekstrakcji sezonowej (domyślnie 10 * liczba wpisów w eval period + 1)

Przykłady:

```
trendstl(/host/key,100h:now/h,10h,2h) #analizuje ostatnie 100 godzin danych trendu, znajduje współczynnik
trendstl(/host/key,100h:now/h-10h,100h,2h,2.1,"mad") #analizuje okres 100 godzin danych trendu do 10 godzi
trendstl(/host/key,100d:now/d-1d,10d,1d,4,,10) #analizuje 100 dni danych trendu do jednego dnia wstecz, zn
trendstl(/host/key,1M:now/M-1y,1d,2h,,"stddevsamp") #analizuje poprzedni miesiąc sprzed roku, znajduje wsp
```

trendsum(/host/key,time period:time shift)

Suma wartości trendów w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendsum(/host/key,1h:now/h) #suma dla poprzedniej godziny (np. 12:00-13:00)
trendsum(/host/key,1h:now/h-1h) #suma sprzed dwóch godzin (11:00-12:00)
trendsum(/host/key,1h:now/h-2h) #suma sprzed trzech godzin (10:00-11:00)
trendsum(/host/key,1M:now/M-1y) #suma dla poprzedniego miesiąca sprzed roku
```

Zobacz wszystkie obsługiwane funkcje.

6 Funkcje matematyczne

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje matematyczne są obsługiwane dla typów wartości float i integer, o ile nie zaznaczono inaczej.

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
abs	Wartość bezwzględna wartości.
acos	Arcus cosinus wartości jako kąt wyrażony w radianach.
asin	Arcus sinus wartości jako kąt wyrażony w radianach.
atan	Arcus tangens wartości jako kąt wyrażony w radianach.
atan2	Arcus tangens współrzędnych rzędnej (wartości) i odciętej określonych jako kąt wyrażony w radianach.
avg	Średnia wartość odwołanych wartości pozycji.
cbrt	Pierwiastek sześcienny z wartości.
ceil	Zaokrągla wartość w górę do najbliższej większej lub równej liczby całkowitej.
cos	Cosinus wartości, gdzie wartość jest kątem wyrażonym w radianach.
cosh	Cosinus hiperboliczny wartości.
cot	Cotangens wartości, gdzie wartość jest kątem wyrażonym w radianach.
degrees	Konwertuje wartość z radianów na stopnie.
e	Liczba Eulera (2.718281828459045).
exp	Liczba Eulera podniesiona do potęgi wartości.
expm1	Liczba Eulera podniesiona do potęgi wartości minus 1.
floor	Zaokrągla wartość w dół do najbliższej mniejszej lub równej liczby całkowitej.
log	Logarytm naturalny.
log10	Logarytm dziesiętny.
max	Najwyższa wartość odwołanych wartości pozycji.
min	Najniższa wartość odwołanych wartości pozycji.
mod	Reszta z dzielenia.
pi	Stała Pi (3.14159265358979).
power	Potęga wartości.
radians	Konwertuje wartość ze stopni na radiany.
rand	Zwraca losową wartość całkowitą.
round	Zaokrągla wartość do określonej liczby miejsc po przecinku.
signum	Zwraca '-1', jeśli wartość jest ujemna, '0', jeśli wartość wynosi zero, '1', jeśli wartość jest dodatnia.
sin	Sinus wartości, gdzie wartość jest kątem wyrażonym w radianach.
sinh	Sinus hiperboliczny wartości, gdzie wartość jest kątem wyrażonym w radianach.
sqrt	Pierwiastek kwadratowy z wartości.
sum	Suma odwołanych wartości pozycji.
tan	Tangens wartości.
truncate	Obcina wartość do określonej liczby miejsc po przecinku.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielone przecinkiem
- Wyrażenia są akceptowane jako parametry
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >

abs(value)

Wartość bezwzględna (od 0) wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość bezwzględna zarówno „3”, jak i „-3” będzie wynosić „3”.

Przykład:

```
abs(last(/host/key))>10
```

acos(value)

Arcus cosinus wartości jako kąta, wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Wartość musi mieścić się w zakresie od -1 do 1. Na przykład arcus cosinus wartości „0.5” będzie wynosił „2.0943951”.

Przykład:

```
acos(last(/host/key))
```

asin(value)

Arcus sinus wartości jako kąt wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Wartość musi mieścić się w zakresie od -1 do 1. Na przykład arcus sinus wartości „0.5” będzie wynosić „-0.523598776”.

Przykład:

```
asin(last(/host/key))
```

atan(value)

Arcus tangens wartości jako kąt wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład arcus tangens wartości „1” będzie wynosił „0.785398163”.

Przykład:

```
atan(last(/host/key))
```

atan2(value,abscissa)

Arcus tangens współrzędnych rzędnej (value) i odciętej, określony jako kąt wyrażony w radianach.

Parametry:

- **value** - wartość do sprawdzenia;
- **abscissa** - wartość odciętej.

Na przykład arcus tangens współrzędnych rzędnej i odciętej dla wartości „1” będzie wynosić „2.21429744”.

Przykład:

```
atan2(last(/host/key),2)
```

avg(<value1>,<value2>,...)

Średnia wartość wartości pozycja, do których istnieje odwołanie.

Parametr:

- **valueX** — wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
avg(avg(/host/key,1h),avg(/host2/key2,1h))
```

cbrt(value)

Pierwiastek sześcienny z wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład pierwiastek sześcienny z „64” wyniesie „4”, a z „63” — „3.97905721”.

Przykład:

```
cbrt(last(/host/key))
```

ceil(value)

Zaokrągla wartość w górę do najbliższej większej lub równej liczby całkowitej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład „2.4” zostanie zaokrąglone w górę do „3”. Zobacz także floor().

Przykład:

```
ceil(last(/host/key))
```

cos(value)

Cosinus wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cosinus wartości „1” będzie wynosił „0.54030230586”.

Przykład:

```
cos(last(/host/key))
```

cosh(value)

Cosinus hiperboliczny wartości. Zwraca wartość jako liczbę rzeczywistą, a nie w notacji naukowej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cosinus hiperboliczny wartości „1” wyniesie „1.54308063482”.

Przykład:

```
cosh(last(/host/key))
```

cot(value)

Cotangens wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cotangens wartości „1” będzie wynosił „0.54030230586”.

Przykład:

```
cot(last(/host/key))
```

degrees(value)

Konwertuje wartość z radianów na stopnie.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość „1” po konwersji na stopnie będzie wynosić „57.2957795”.

Przykład:

```
degrees(last(/host/key))
```

e

Liczba Eulera (2.718281828459045).

Przykład:

```
e()
```

exp(value)

Liczba Eulera podniesiona do potęgi wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład liczba Eulera podniesiona do potęgi wartości „2” będzie równa „7.38905609893065”.

Przykład:

```
exp(last(/host/key))
```

expm1(value)

Liczba Eulera podniesiona do potęgi wartości minus 1.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład liczba Eulera podniesiona do potęgi wartości „2” minus 1 będzie równa „6.38905609893065”.

Przykład:

```
expm1(last(/host/key))
```

floor(value)

Zaokrągla wartość w dół do najbliższej mniejszej lub równej liczby całkowitej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład „2.6” zostanie zaokrąglone w dół do „2”. Zobacz także ceil().

Przykład:

```
floor(last(/host/key))
```

log(value)

Logarytm naturalny.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład logarytm naturalny wartości „2” wyniesie „0.69314718055994529”.

Przykład:

```
log(last(/host/key))
```

log10(value)

Logarytm dziesiętny.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład logarytm dziesiętny wartości „5” wyniesie „0.69897000433”.

Przykład:

```
log10(last(/host/key))
```

max(<value1>,<value2>,...)

Najwyższa wartość spośród wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
max(avg(/host/key,1h),avg(/host2/key2,1h))
```

min(<value1>,<value2>,...)

Najniższa wartość spośród wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
min(avg(/host/key,1h),avg(/host2/key2,1h))
```

mod(value,denominator)

Reszta z dzielenia.

Parametry:

- **value** - wartość do sprawdzenia;
- **denominator** - mianownik dzielenia.

Na przykład reszta z dzielenia wartości ’5’ przez mianownik dzielenia ’2’ będzie wynosić ’1’.

Przykład:

```
mod(last(/host/key),2)
```

pi

Stała Pi (3.14159265358979).

Przykład:

```
pi()
```

power(value,power value)

Potęga wartości.

Parametry:

- **value** - wartość do sprawdzenia;
- **power value** - wykładnik potęgi do użycia.

Na przykład trzecia potęga wartości „2” będzie równa „8”.

Przykład:

```
power(last(/host/key),3)
```

radians(value)

Konwertuje wartość ze stopni na radiany.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość „1” po konwersji na radiany będzie wynosić „0.0174532925”.

Przykład:

```
radians(last(/host/key))
```

rand

Zwraca losową wartość całkowitą. Jest to liczba pseudolosowa generowana z użyciem czasu jako ziarna (wystarczająca do celów matematycznych, ale nie do kryptografii).

Przykład:

```
rand()
```

round(value,decimal places)

Zaokrągla wartość do określonej liczby miejsc dziesiętnych.

Parametry:

- **value** - wartość do sprawdzenia;
- **decimal places** - określa liczbę miejsc dziesiętnych do zaokrąglenia (możliwe jest również 0).

Na przykład wartość „2.5482” zaokrąglona do 2 miejsc dziesiętnych będzie wynosić „2.55”.

Przykład:

```
round(last(/host/key),2)
```

signum(value)

Zwraca ’-1’, jeśli wartość jest ujemna, ’0’, jeśli wartość jest równa zero, oraz ’1’, jeśli wartość jest dodatnia.

Parametr:

- **value** - wartość do sprawdzenia.

Przykład:

```
signum(last(/host/key))
```

sin(value)

Sinus wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład sinus wartości „1” wyniesie „0.8414709848”.

Przykład:

```
sin(last(/host/key))
```

sinh(value)

Sinus hiperboliczny wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład sinus hiperboliczny wartości „1” wyniesie „1.17520119364”.

Przykład:

```
sinh(last(/host/key))
```

sqrt(value)

Pierwiastek kwadratowy z wartości.
 Ta funkcja zakończy się niepowodzeniem dla wartości ujemnej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład pierwiastek kwadratowy z wartości „3.5” będzie wynosił „1.87082869339”.

Przykład:

```
sqrt(last(/host/key))
```

sum(<value1>,<value2>,...)

Suma wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która pracuje z historią pozycji.

Przykład:

```
sum(avg(/host/key,1h),avg(/host2/key2,1h))
```

tan(value)

Tangens wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład tangens wartości „1” wyniesie „1.55740772465”.

Przykład:

```
tan(last(/host/key))
```

truncate(value,decimal places)

Obetnij wartość do określonej liczby miejsc dziesiętnych.

Parametry:

- **value** - wartość do sprawdzenia;
- **decimal places** - określ liczbę miejsc dziesiętnych do obcięcia (możliwe jest również 0).

Na przykład wartość „2.5482” obcięta do 2 miejsc dziesiętnych będzie wynosić „2.54”.

Przykład:

```
truncate(last(/host/key),2)
```

Zobacz wszystkie obsługiwane funkcje.

7 Funkcje operatorów

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Obliczanych pozycjach
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij nazwę funkcji, aby zobaczyć pełne szczegóły.

Function	Description
between	Sprawdza, czy wartość należy do podanego zakresu.
in	Sprawdza, czy wartość jest równa co najmniej jednej z wymienionych wartości.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielone przecinkiem
- Wyrażenia są akceptowane jako parametry

between(value,min,max)

Sprawdza, czy wartość należy do podanego zakresu.
 Obsługiwane typy wartości: *Integer*, *Float*.
 Zwraca: 1 - w zakresie; 0 - w przeciwnym razie.

Parametry:

- **value** - wartość do sprawdzenia;
- **min** - wartość minimalna;
- **max** - wartość maksymalna.

Przykład:

```
between(last(/host/key),1,10)=1 #wyzwalacz, jeśli wartość jest między 1 a 10
```

in(value,value1,value2,...valueN)

Sprawdza, czy wartość jest równa co najmniej jednej z podanych wartości.
 Obsługiwane typy wartości: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Zwraca: 1 - jeśli równa; 0 - w przeciwnym razie.

Parametry:

- **value** - wartość do sprawdzenia;

- **valueX** - podane wartości (wartości tekstowe muszą być ujęte w podwójny cudzysłów).

Wartość jest porównywana z podanymi wartościami jako liczby, jeśli wszystkie te wartości mogą zostać przekonwertowane na wartości liczbowe; w przeciwnym razie jest porównywana jako ciągi znaków.

Przykład:

```
in(last(/host/key),5,10)=1 #wyzwalacz, jeśli ostatnia wartość jest równa 5 lub 10
in("text",last(/host/key),last(/host/key,#2))=1 #wyzwalacz, jeśli "text" jest równe jednej z 2 ostatnich w
```

Zobacz wszystkie obsługiwane funkcje.

8 Funkcje predykcyjne

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
forecast	Przyszła wartość, max, min, delta lub avg pozycji.
timeleft	Czas w sekundach potrzebny, aby pozycja osiągnęła określony próg.

Wspólne parametry

- `/host/key` jest wspólnym obowiązkowym pierwszym parametrem dla funkcji odwołujących się do historii pozycji hosta
- `(sec|#num)<:time shift>` jest wspólnym drugim parametrem dla funkcji odwołujących się do historii pozycji hosta, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać przyrostków czasu), lub
  - **#num** - maksymalny zakres oceny w najnowszych zebranych wartościach (jeśli poprzedzony znakiem hash)
  - **time shift** (opcjonalnie) umożliwia przesunięcie punktu oceny wstecz w czasie. Zobacz więcej szczegółów dotyczących określania time shift.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

forecast(/host/key,(sec|#num)<:time shift>,time,<fit>,<mode>)

Przyszła wartość, maksimum, minimum, delta lub średnia pozycji.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry;
- **time** — horyzont prognozy w sekundach (mogą być używane sufiksy czasu); obsługiwane są wartości ujemne;
- **fit** (opcjonalny; musi być ujęty w podwójny cudzysłów) — funkcja używana do dopasowania danych historycznych. Obsługiwane dopasowania:
linear — funkcja liniowa (domyślnie)
polynomialN — wielomian stopnia N (1 <= N <= 6)
exponential — funkcja wykładnicza
logarithmic — funkcja logarytmiczna
power — funkcja potęgowa
Zwróć uwagę, że *polynomial1* jest równoważne *linear*;
- **mode** (opcjonalny; musi być ujęty w podwójny cudzysłów) — wymagane dane wyjściowe. Obsługiwane tryby:
value — wartość (domyślnie)
max — maksimum
min — minimum
delta — *max-min*
avg — średnia
Zwróć uwagę, że *value* szacuje wartość pozycji w momencie `now + time`; *max*, *min*, *delta* i *avg* analizują oszacowanie wartości pozycji w przedziale między `now` a `now + time`.

Komentarze:

- Jeśli zwracana wartość jest większa niż 1.7976931348623158E+308 lub mniejsza niż -1.7976931348623158E+308, zwracana wartość jest odpowiednio ograniczana do 1.7976931348623158E+308 lub -1.7976931348623158E+308;
- Staje się nieobsługiwana tylko w przypadku nieprawidłowego użycia w wyrażeniu (niewłaściwy typ pozycji, nieprawidłowe parametry), w przeciwnym razie w przypadku błędów zwraca -1;
- Zobacz także dodatkowe informacje o funkcjach predykcyjnych wyzwalaczy.

Przykłady:

```
forecast(/host/key,#10,1h) #prognoza wartości pozycji za godzinę na podstawie ostatnich 10 wartości
forecast(/host/key,1h,30m) #prognoza wartości pozycji za 30 minut na podstawie danych z ostatniej godziny
forecast(/host/key,1h:now-1d,12h) #prognoza wartości pozycji za 12 godzin na podstawie danych z jednej god
forecast(/host/key,1h,10m,"exponential") #prognoza wartości pozycji za 10 minut na podstawie danych z osta
forecast(/host/key,1h,2h,"polynomial3","max") #prognoza maksymalnej wartości, jaką pozycja może osiągnąć w
forecast(/host/key,#2,-20m) #oszacowanie wartości pozycji sprzed 20 minut na podstawie dwóch ostatnich war
```

timeleft(/host/key,(sec|#num)<:przesunięcie czasu>,threshold,<fit>)

Czas w sekundach potrzebny, aby pozycja osiągnęła określony próg.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry;
- **threshold** - wartość do osiągnięcia (można używać sufiksów jednostek);
- **fit** (opcjonalny; musi być ujęty w podwójny cudzysłów) - zobacz forecast().

Uwagi:

- Jeśli zwracana wartość jest większa niż 1.7976931348623158E+308, zostanie przycięta do 1.7976931348623158E+308;
- Zwraca 1.7976931348623158E+308, jeśli próg nie może zostać osiągnięty;
- Staje się nieobsługiwana tylko w przypadku nieprawidłowego użycia w wyrażeniu (niewłaściwy typ pozycji, nieprawidłowe parametry), w przeciwnym razie zwraca -1 w przypadku błędów;
- Zobacz także dodatkowe informacje o funkcjach predykcyjnych wyzwalaczy.

Przykłady:

```
timeleft(/host/key,#10,0) #czas do momentu, aż wartość pozycji osiągnie zero na podstawie ostatnich 10 war
timeleft(/host/key,1h,100) #czas do momentu, aż wartość pozycji osiągnie 100 na podstawie danych z ostatni
timeleft(/host/key,1h:now-1d,100) #czas do momentu, aż wartość pozycji osiągnie 100 na podstawie jednej go
timeleft(/host/key,1h,200,"polynomial2") #czas do momentu, aż wartość pozycji osiągnie 200 na podstawie da
```

Zobacz wszystkie obsługiwane funkcje.

9 Funkcje tekstowe

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij nazwę funkcji, aby zobaczyć pełne szczegóły.

Function	Description
ascii	Kod ASCII najbardziej lewego znaku wartości.
bitlength	Długość wartości w bitach.
bytelength	Długość wartości w bajtach.
char	Zwraca znak, interpretując wartość jako kod ASCII.
concat	Łańcuch wynikający z połączenia wartości odwołanych pozycji lub wartości stałych.
insert	Wstawia określone znaki lub spacje do łańcucha znaków, zaczynając od określonej pozycji w łańcuchu.
jsonpath	Zwraca wynik JSONPath.
left	Zwraca najbardziej lewe znaki wartości.
length	Długość wartości w znakach.
ltrim	Usuwa określone znaki z początku łańcucha.
mid	Zwraca podciąg o N znakach, zaczynający się od pozycji znaku określonej przez ’start’.
repeat	Powtarza łańcuch.
replace	Wyszukuje wzorzec w wartości i zastępuje go wartością zastępczą.
right	Zwraca najbardziej prawe znaki wartości.
rtrim	Usuwa określone znaki z końca łańcucha.
trim	Usuwa określone znaki z początku i końca łańcucha.
xmlxpath	Zwraca wynik XML XPath.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Wyrażenia są akceptowane jako parametry
- Parametry tekstowe muszą być ujęte w podwójny cudzysłów; w przeciwnym razie mogą zostać błędnie zinterpretowane
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczane przez < >

ascii(value)

Kod ASCII skrajnie lewego znaku wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość taka jak 'Abc' zwróci '65' (kod ASCII dla 'A').

Przykład:

```
ascii(last(/host/key))
```

bitlength(value)

Długość wartości w bitach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Integer*.

Parametr:

- **value** - wartość do sprawdzenia

Przykład:

```
bitlength(last(/host/key))
```

bytelength(value)

Długość wartości w bajtach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Integer*.

Parametr:

- **value** - wartość do sprawdzenia

Przykład:

```
bytelength(last(/host/key))
```

char(value)

Zwraca znak poprzez interpretację wartości jako kodu ASCII.
 Obsługiwane typy wartości: *Integer*.

Parametr:

- **value** — wartość do sprawdzenia

Wartość musi mieścić się w zakresie 0–255. Na przykład wartość taka jak „65" (interpretowana jako kod ASCII) zwróci „A".

Przykład:

```
char(last(/host/key))
```

concat(<value1>,<value2>,...)

Ciąg znaków wynikający z połączenia wartości referencyjnych pozycji lub wartości stałych.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Float*, *Integer*.

Parametr:

- **valueX** - wartość zwracana przez jedną z funkcji historii lub wartość stała (ciąg znaków, liczba całkowita lub zmienno-przecinkowa). Musi zawierać co najmniej dwa parametry.

Na przykład wartość taka jak 'Zab' połączona z 'bix' (stały ciąg znaków) zwróci 'Zabbix'.

Przykłady:

```
concat(last(/host/key),"bix")
concat("1 min: ",last(/host/system.cpu.load[all,avg1]),", 15 min: ",last(/host/system.cpu.load[all,avg15]))
```

insert(value,start,length,replacement)

Wstawia określone znaki lub spacje do ciągu znaków, zaczynając od wskazanej pozycji w ciągu.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **start** - pozycja początkowa;
- **length** - pozycje do zastąpienia;
- **replacement** - ciąg zastępujący.

Na przykład wartość taka jak 'Zabbbix' zostanie zastąpiona przez 'Zabbix', jeśli 'bb' (pozycja początkowa 3, liczba pozycji do zastąpienia 2) zostanie zastąpione przez 'b'.

Przykład:

```
insert(last(/host/key),3,2,"b")
```

jsonpath(value,path,<default>)

Zwraca wynik JSONPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **path** - ścieżka (musi być ujęta w cudzysłów);
- **default** - opcjonalna wartość zapasowa, jeśli zapytanie JSONPath nie zwróci danych. Należy pamiętać, że w przypadku innych błędów zwracane jest niepowodzenie (np. „unsupported construct").

Przykład:

```
jsonpath(last(/host/proc.get[zabbix_agentd,,,summary]),"$..size")
```

left(value,count)

Zwraca skrajnie lewe znaki wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **count** - liczba znaków do zwrócenia.

Na przykład można zwrócić „Zab" z „Zabbix", określając 3 skrajnie lewe znaki do zwrócenia. Zobacz także right().

Przykład:

```
left(last(/host/key),3) #zwraca trzy skrajnie lewe znaki
```

length(value)

Długość wartości w znakach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametr:

- **value** - wartość do sprawdzenia.

Przykłady:

```
length(last(/host/key)) #długość najnowszej wartości
length(last(/host/key,#3)) #długość trzeciej najnowszej wartości
length(last(/host/key,#1:now-1d)) #długość najnowszej wartości sprzed jednego dnia
```

ltrim(value,<chars>)

Usuń określone znaki z początku ciągu.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie usuwane są białe znaki z początku ciągu (jeśli nie określono opcjonalnych znaków). Zobacz także: rtrim(), trim().

Przykłady:

```
ltrim(last(/host/key)) #usuń białe znaki z początku ciągu
ltrim(last(/host/key),"Z") #usuń dowolne 'Z' z początku ciągu
ltrim(last(/host/key)," Z") #usuń dowolną spację i 'Z' z początku ciągu
```

mid(value,start,length)

Zwraca podciąg N znaków, zaczynając od pozycji znaku określonej przez parametr „start".
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **start** - pozycja początkowa podciągu;
- **length** - liczba pozycji do zwrócenia w podciągu.

Na przykład możliwe jest zwrócenie „abbi” z wartości takiej jak „Zabbix”, jeśli pozycja początkowa wynosi 2, a liczba pozycji do zwrócenia wynosi 4.

Przykład:

```
mid(last(/host/key),2,4)="abbi"
```

repeat(value,count)

Powtórz ciąg znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **count** - liczba powtórzeń.

Przykład:

```
repeat(last(/host/key),2) #powtórz wartość dwa razy
```

replace(value,pattern,replacement)

Znajduje wzorzec w wartości i zastępuje go ciągiem replacement. Wszystkie wystąpienia wzorca zostaną zastąpione.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **pattern** - wzorzec do znalezienia;
- **replacement** - ciąg, którym zostanie zastąpiony wzorzec.

Przykład:

```
replace(last(/host/key),"ibb","abb") #zastępuje wszystkie 'ibb' przez 'abb'
```

right(value,count)

Zwraca skrajnie prawe znaki wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **count** - liczba znaków do zwrócenia.

Na przykład można zwrócić „bix” z „Zabbix”, podając 3 skrajnie prawe znaki do zwrócenia. Zobacz także left().

Przykład:

```
right(last(/host/key),3) #zwraca trzy skrajnie prawe znaki
```

rtrim(value,<chars>)

Usuwa określone znaki z końca ciągu znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie usuwane są białe znaki z prawej strony (jeśli nie określono opcjonalnych znaków). Zobacz także: ltrim(), trim().

Przykłady:

```
rtrim(last(/host/key)) #usuń białe znaki z końca ciągu znaków
rtrim(last(/host/key),"x") #usuń wszystkie znaki 'x' z końca ciągu znaków
rtrim(last(/host/key),"x ") #usuń wszystkie znaki 'x' i spacje z końca ciągu znaków
```

trim(value,<chars>)

Usuń określone znaki z początku i końca ciągu znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;

- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie z obu stron usuwane są białe znaki (jeśli nie określono opcjonalnych znaków). Zobacz także: ltrim(), rtrim().

Przykłady:

```
trim(last(/host/key)) #usuń białe znaki z początku i końca ciągu znaków
trim(last(/host/key),"_") #usuń '_' z początku i końca ciągu znaków
```

xmlxpath(value,path,<default>)

Zwraca wynik XML XPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **path** - ścieżka (musi być ujęta w cudzysłów);
- **default** - opcjonalna wartość zastępcza, jeśli zapytanie XML XPath zwróci pusty zbiór węzłów. Nie zostanie zwrócona, jeśli pusty wynik nie jest zbiorem węzłów (tj. pustym ciągiem). W przypadku innych błędów zwracane jest niepowodzenie (np. „invalid expression”).

Przykład:

```
xmlxpath(last(/host/xml_result),"/response/error/status")
```

Zobacz wszystkie obsługiwane funkcje.

6 Makra

Możliwe jest użycie gotowych obsługiwanych makr oraz makr użytkownika obsługiwanych w zależności od lokalizacji.

1 Makra obsługiwane przez lokalizację

Przegląd

Tabela makr obsługiwanych w danej lokalizacji zawiera informacje referencyjne o wszystkich makrach (wbudowanych, wyrażeń oraz definiowanych przez użytkownika), które są obsługiwane w określonej lokalizacji.

Jeśli obsługiwane są wszystkie makra z powiązanej grupy, podawana jest tylko nazwa grupy. Kliknij nazwę każdego makra lub nazwę grupy, aby zobaczyć pełne szczegóły dotyczące makr.

Lista szczegółów makr zawiera szczegółowy wykaz wszystkich makr **wbudowanych**, pogrupowanych według obszaru zastosowania. Należy pamiętać, że aby dostosować wartości makr (na przykład skrócić je lub wyodrębnić określone podciągi), można użyć funkcji makr.

Makra obsługiwane w zależności od lokalizacji

Lokalizacja	Obsługiwane makra	Komentarze
Action		
Warunek okresu czasu, Domyślny czas trwania kroku operacji, Czas trwania kroku	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane
Powiadomienia i polecenia		

Lokalizacja	Obsługiwane makra	Komentarze
Oparte na wyzwalaczu	Makra Action Makra Data/czas {ESC.HISTORY} Makra Zdarzenie, makra Zdarzenie przyczyny/objawu Makra aktualizacji zdarzenia: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS} Makro wyrażenia: {?EXPRESSION} {FUNCTION.VALUE} Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra host docelowy (tylko polecenia) Makra inwentarza hosta Makra pozycji (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) Makra proxy Makra wyzwalacza: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.ID}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} Makra definiowane przez użytkownika	Zobacz też: Używanie makr w wiadomościach
Aktualizacja problemu	Makra Action Makra Data/czas {ESC.HISTORY} Makra Zdarzenie, Zdarzenie przyczyny/objawu, Zdarzenie odzyskania Makra aktualizacji zdarzenia: wszystkie (z wyjątkiem {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}) Makro wyrażenia: {?EXPRESSION} Makra funkcji Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra host docelowy (tylko polecenia) Makra inwentarza hosta Makra pozycji (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) Makra proxy Makra wyzwalacza: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.ID}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} {USER.FULLNAME} Makra definiowane przez użytkownika	
Odzyskanie problemu	Makra Zdarzenie odzyskania {FUNCTION.RECOVERY.VALUE} {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}	

Lokalizacja	Obsługiwane makra	Komentarze
Odkrywanie	Makra Action Makra Data/czas Makra odkrywania Makra zdarzenia: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} Makra host docelowy (tylko polecenia) Makra proxy	
Autorejestracja	Makra Action Makra Data/czas Makra zdarzenia: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} Makra hosta: {HOST.HOST}, {HOST.IP}, {HOST.PORT}, {HOST.METADATA} Makra host docelowy (tylko polecenia) Makra proxy	
Oparte na usługach	Makra Action Makra Data/czas {ESC.HISTORY} Makra zdarzenia: wszystkie z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA} Makra usługi Makra definiowane przez użytkownika	
Odzyskanie usługi	Makra Zdarzenie (z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA}), makra Zdarzenie odzyskania	
Aktualizacja usługi	Makra Action Makra Data/czas {ESC.HISTORY} Makra Zdarzenie (z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA}) Makra aktualizacji zdarzenia: {EVENT.UPDATE.DATE}, {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}, {EVENT.UPDATE.STATUS}, {EVENT.UPDATE.TIME}, {EVENT.UPDATE.TIMESTAMP} Makra usługi Makra definiowane przez użytkownika	
Powiadomienia wewnętrzne	Makra Action Makra Data/czas {ESC.HISTORY} Makra Zdarzenie (z wyjątkiem {EVENT.NSEVERITY}, {EVENT.SEVERITY}), makra Zdarzenie odzyskania Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji: {ITEM.DESCRIPTION}, {ITEM.DESCRIPTION.ORIG}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, {ITEM.NAME}, {ITEM.NAME.ORIG}, {ITEM.VALUETYPE} Makra proxy	
Oparte na pozycji	Makra pozycji: {ITEM.STATE}, {ITEM.STATE.ERROR}	
Oparte na regule LLD	Makra powiadomienia odkrywania niskiego poziomu	
Oparte na wyzwalaczu	Makra wyzwalacza: {TRIGGER.DESCRIPTION}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.ID}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATE}, {TRIGGER.STATE.ERROR}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME} Makra definiowane przez użytkownika	

Lokalizacja	Obsługiwane makra	Komentarze
Parametry skryptu alertu Connector	Makra Alert Makra definiowane przez użytkownika (tylko globalne)	
Pole URL, Nazwa użytkownika, Hasło, token Bearer, proxy HTTP, certyfikat SSL, plik klucza SSL, hasło klucza SSL	Makra definiowane przez użytkownika (tylko globalne)	
Pole Timeout	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Nazwa wykresu	Makro wyrażenia: {?EXPRESSION}	W tym makrze w tej lokalizacji obsługiwana jest tylko jedna funkcja **avg**, **last**, **max** lub **min** z sekundami jako parametrem (można używać sufiksów czasu). Makra {HOST.HOST<1-9>} można użyć jako hosta wewnątrz makra (zobacz też makra indeksowane). Przykład użycia: {?avg(/{HOST.HOST}/item.key,1h)} Nieobsługiwane użycie: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Host, prototyp hosta		
Adres IP/DNS interfejsu	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} Makra definiowane przez użytkownika [1]	[1] Dla adresu IP pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane
Port interfejsu	Makra definiowane przez użytkownika	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Społeczność SNMP Nazwa kontekstu SNMPv3, nazwa zabezpieczeń, hasło uwierzytelniania, hasło prywatności	Makra definiowane przez użytkownika	

Lokalizacja	Obsługiwane makra	Komentarze
Nazwa użytkownika IPMI, hasło		
Pozycja, prototyp pozycji, reguła LLD		
Nazwa	Makra definiowane przez użytkownika	Nieobsługiwane w nazwach reguł LLD.
Opis	Makra definiowane przez użytkownika	
Parametry klucza	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	Obsługiwane makra {HOST.*} zostaną rozwiązane do interfejsu wybranego dla pozycji. W przypadku użycia w pozycjach bez interfejsów zostaną rozwiązane do interfejsu Zabbix agent, SNMP, JMX lub IPMI hosta w tej kolejności priorytetu albo do wartości 'UNKNOWN', jeśli host nie ma żadnego interfejsu.
Interwał aktualizacji, Interwały niestandardowe, Timeout (dostępne dla obsługiwanych typów pozycji), Przechowuj do (dla historii/trendów) pola	Makra definiowane przez użytkownika	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Usuń utracone zasoby, Wyłącz utracone zasoby pola	Makra definiowane przez użytkownika	Te pola są obsługiwane tylko dla reguł LLD. Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Parametry kroku preprocessingu (w tym skrypty niestandardowe), *niestandardowe parametry obsługi błędów* (pola *Ustaw wartość na* i *Ustaw błąd na*)	Makra definiowane przez użytkownika	
Wyrażenia regularne filtra	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	Te pola są obsługiwane tylko dla reguł LLD.
Wyrażenia regularne filtra nadpisania	Makra definiowane przez użytkownika	Te pola są obsługiwane tylko dla reguł LLD.

Lokalizacja	Obsługiwane makra	Komentarze
Operacje nadpisania: *Interwał aktualizacji*, *Okres przechowywania historii*, *Okres przechowywania trendów* (dla prototypów pozycji)	Makra definiowane przez użytkownika	Te pola są obsługiwane tylko dla reguł LLD. Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Typy pozycji		
Browser pozycja *nazwy i wartości parametrów*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Makra definiowane przez użytkownika	
Script pozycja *nazwy i wartości parametrów*		
Calculated/aggregate pozycja (*stałe wyrażenia*, *parametry funkcji*, *parametry klucza pozycji*, (tylko pozycja agregująca) *warunki filtra* (nazwa grupy hostów i nazwa tagu))	Makra definiowane przez użytkownika	
Database monitor pozycja *pola Nazwa użytkownika, Hasło, zapytanie SQL*	Makra definiowane przez użytkownika	
HTTP agent pozycja *pola URL, pola zapytania, treść żądania, nagłówki (nazwy i wartości), plik certyfikatu SSL, plik klucza SSL*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Makra definiowane przez użytkownika	Adresy URL zawierające sekretne makro nie będą działać, ponieważ makro w nich zostanie rozwiązane jako "******"
HTTP agent pozycja *pole HTTP Proxy*	Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Makra definiowane przez użytkownika	
HTTP agent pozycja *pola Wymagane kody statusu, Nazwa użytkownika/hasło uwierzytelniania HTTP, hasło klucza SSL*	Makra definiowane przez użytkownika	
HTTP agent pozycja *pole Dozwolone hosty*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	
JMX agent pozycja *pole Endpoint*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.PORT} Makra definiowane przez użytkownika	
Trapper pozycja *pole Dozwolone hosty*		
SNMP agent pozycja *pole SNMP OID*	Makra definiowane przez użytkownika	
SSH and Telnet pozycja *skrypt*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	

Lokalizacja	Obsługiwane makra	Komentarze
SSH agent pozycja *pola Nazwa użytkownika, plik klucza publicznego, plik klucza prywatnego, Hasło*	Makra definiowane przez użytkownika	
Telnet agent pozycja *pola Nazwa użytkownika, Hasło*	Makra definiowane przez użytkownika	
Mapa		
URL, nazwa URL	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} Makra inwentarza hosta Makra mapy {TRIGGER.ID}	
Etykieta elementu	Makro wyrażenia: {?EXPRESSION} [1] Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} Makra inwentarza hosta Makra mapy Makra wyzwalacza: {TRIGGER.EVENTS.*}, {TRIGGER.ID}, {TRIGGER.PROBLEM.EVENTS.*}, {TRIGGERS.ACK}, {TRIGGERS.PROBLEM.ACK}, {TRIGGERS.PROBLEM.UNACK}, {TRIGGERS.UNACK}	[1] W tej lokalizacji w tym makrze obsługiwana jest tylko jedna funkcja **avg**, **last**, **max** lub **min** z sekundami jako parametrem (można używać sufiksów czasu). Makra {HOST.HOST<1-9>} można użyć jako hosta wewnątrz makra. Przykład użycia: {?avg(/{HOST.HOST}/item.key,1h)} Nieobsługiwane użycie: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Etykieta łącza	Makro wyrażenia: {?EXPRESSION} [1]	
Etykieta kształtu	Makro wyrażenia: {?EXPRESSION} [1]	
Pole tekstowe w kształtach mapy	{MAP.NAME}	
Odkrywanie sieci		
Pole Interwał aktualizacji	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
pola Społeczność SNMP, SNMP OID pola Nazwa kontekstu SNMPv3, nazwa zabezpieczeń, hasło uwierzytelniania, hasło prywatności	Makra definiowane przez użytkownika (tylko globalne)	
Proxy		

Lokalizacja	Obsługiwane makra	Komentarze
Adres dla aktywnych agentów > *Port* (jeśli proxy należy do grupy) W trybie pasywnego proxy: pola interfejsu *Adres* i *Port* Pola *Timeouty dla typów pozycji* w trybie nadpisywania	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Grupa proxy		
Okres awaryjnego przełączenia, *Minimalna liczba proxy* pola	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Skrypty		
Polecenia, tekst potwierdzenia (Typ: skrypt, ręczna **akcja hosta**)	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID} [1], {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta [2] {MANUALINPUT} Makra nazwa użytkownika Makra definiowane przez użytkownika	[1] Tylko tekst potwierdzenia [2] Obsługiwane dla Zabbix server i Zabbix proxy
Polecenia (Typ: skrypt, ręczna **akcja zdarzenia**)	Makra Data/czas Makra Zdarzenie, makra Zdarzenie przyczyny/objawu, makra Zdarzenie odzyskania (jeśli nastąpiło odzyskanie) Makra aktualizacji zdarzenia: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS} {FUNCTION.VALUE} Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) {MANUALINPUT} Makra proxy Makra wyzwalacza: {TRIGGER.DESCRIPTION}, {TRIGGER.EVENTS.*}, {TRIGGER.EXPRESSION}, {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY}, {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}, {TRIGGER.HOSTGROUP.NAME}, {TRIGGER.ID}, {TRIGGER.NAME}, {TRIGGER.NAME.ORIG}, {TRIGGER.NSEVERITY}, {TRIGGER.SEVERITY}, {TRIGGER.STATUS}, {TRIGGER.TEMPLATE.NAME}, {TRIGGER.URL}, {TRIGGER.URL.NAME}, {TRIGGER.VALUE} Makra nazwa użytkownika Makra definiowane przez użytkownika	
Tekst potwierdzenia (Typ: skrypt, ręczna **akcja zdarzenia**)	Makra zdarzenia: {EVENT.ID}, {EVENT.NAME}, {EVENT.NSEVERITY}, {EVENT.SEVERITY}, {EVENT.STATUS}, {EVENT.VALUE} Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta {MANUALINPUT} Makra nazwa użytkownika Makra definiowane przez użytkownika	

Lokalizacja	Obsługiwane makra	Komentarze
URL, tekst potwierdzenia (Typ: URL, ręczna **akcja hosta/zdarzenia**)		
Nazwa i wartość tagu	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra wyzwalacza: {TRIGGER.ID} (tylko wartość tagu wyzwalacza) Makra definiowane przez użytkownika	Makra nazwy i wartości tagu są rozwijane tylko podczas generowania zdarzenia.
Trigger		
Nazwa	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra pozycyjne/odwołania: $1 . . . $9 Makra definiowane przez użytkownika	
Nazwa zdarzenia	Makra Data/czas: {TIME}, {TIMESTAMP} Makro wyrażenia: {?EXPRESSION} Makra funkcji Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra wyzwalacza: {TRIGGER.EXPRESSION.EXPLAIN}	
Dane operacyjne	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra definiowane przez użytkownika	
Wyrażenie	{TRIGGER.VALUE} Makra definiowane przez użytkownika [1]	[1] Tylko w stałych i parametrach funkcji; sekretne makra nie są obsługiwane
Pola Tag do dopasowania, Nazwa wpisu menu, URL wpisu menu	Makra definiowane przez użytkownika	Adresy URL zawierające sekretne makro nie będą działać, ponieważ makro w nich zostanie rozwiązane jako "******".
Opis	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra definiowane przez użytkownika {EVENT.ID}	
URL	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} {TRIGGER.ID}	
Polecenia parametrów użytkownika	Makra pozycyjne/odwołania: $1 . . . $9	
Scenariusz web		

Lokalizacja	Obsługiwane makra	Komentarze
Pola Nazwa, Agent, proxy HTTP	Makra definiowane przez użytkownika	
Pole Interwał aktualizacji	Makra definiowane przez użytkownika	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Pola Zmienne, Nagłówki, certyfikat SSL, plik klucza SSL	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	
Pola Użytkownik, Hasło, hasło klucza SSL	Makra definiowane przez użytkownika	
Krok scenariusza web *pola Nazwa, Zmienne (tylko wartości)*	Makra definiowane przez użytkownika	
Krok scenariusza web *pola Timeout, Wymagane kody statusu*	Makra definiowane przez użytkownika	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Krok scenariusza web *pola URL, Nagłówki (nazwy i wartości), Wymagany ciąg*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	
Krok scenariusza web *pole Post*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}	
Typ mediów web-hook		
Wartości parametrów	Makra Alert Wszystkie makra obsługiwane w powiadomieniach o problemach opartych na wyzwalaczu (patrz wyżej)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
skrypt JavaScript, nazwa parametru skryptu *JavaScript*	Makra definiowane przez użytkownika (tylko globalne)	
Nazwa wpisu menu, URL wpisu menu	{EVENT.TAGS.<tag name>}	
Widżety		
Gauge widżet *pole Opis*	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji (z wyjątkiem {ITEM.STATE.ERROR}) Makra definiowane przez użytkownika (tylko globalne)	
Honeycomb widżet *pole Etykieta główna/pomocnicza* **Item value** widżet *pole Opis*		
Top hosts widżet *kolumna Dane tekstowe*	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra definiowane przez użytkownika (tylko globalne)	

Lokalizacja	Obsługiwane makra	Komentarze
URL (Dynamiczny) widżet *pole URL*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra definiowane przez użytkownika	
Inne lokalizacje		
Pole Czas pracy	Makra definiowane przez użytkownika (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub połączenie makr z tekstem nie jest obsługiwane.
Pole Timeouty dla typów pozycji (ustawienie globalne)		
Media użytkownika *pole Gdy aktywne*		
Typ mediów Email *pola Nazwa użytkownika, Hasło*	Makra definiowane przez użytkownika (tylko globalne)	Zalecane są sekretne makra.
Raporty zaplanowane *pola Temat, Wiadomość*	Makra Data/czas: {TIME}, {TIMESTAMP}	

Szczegóły makr

Ta lista zawiera szczegóły dotyczące każdego wbudowanego makra, pogrupowane według obszaru zastosowania:

- Makra akcji
- Makra alertów
- Makra daty/czasu
- Makra wykrywania
- Makra zdarzeń
  - Makra aktualizacji zdarzeń
  - Makra zdarzeń przyczyny/objawu
  - Makra zdarzeń odzyskiwania
- Makra eskalacji
- Makra funkcji
- Makra hosta
  - Makra hosta docelowego
- Makra grup hostów
- Makra inwentarza hosta
- Makra pozycji
- Makra powiadomień wykrywania niskiego poziomu
- Makra map
- Makra proxy
- Makra skryptów
- Makra usług
- Makra wyzwalaczy
- Makra nazw użytkowników

Makra akcji

{ACTION.ID}

Numeryczny identyfikator ID wyzwolonej akcji.

{ACTION.NAME}

Nazwa uruchomionej akcji.

Makra alertów

{ALERT.MESSAGE}

Wartość *Domyślna wiadomość* z konfiguracji akcji.

{ALERT.SENDTO}

Wartość *Send to* z konfiguracji mediów użytkownika.

{ALERT.SUBJECT}

Wartość *Default subject* z konfiguracji akcji.

Makra daty i czasu

{DATE}

Bieżąca data w formacie `yyyy.mm.dd`.

{TIME}

Bieżący czas w formacie `hh:mm:ss`.

{TIMESTAMP}

Bieżący czas w formacie znacznika czasu UNIX.

Makra wykrywania

{DISCOVERY.DEVICE.IPADDRESS}

Adres IP wykrytego urządzenia.
 Dostępne zawsze, nie zależy od dodania hosta.

{DISCOVERY.DEVICE.DNS}

Nazwa DNS wykrytego urządzenia.
 Zawsze dostępna, nie zależy od dodania hosta.

{DISCOVERY.DEVICE.STATUS}

Status wykrytego urządzenia (UP/DOWN).

{DISCOVERY.DEVICE.UPTIME}

Czas od ostatniej zmiany statusu wykrywania dla określonego urządzenia.
 Na przykład: 1h 29m 01s.
 Dla urządzeń ze statusem DOWN jest to okres ich niedostępności.

{DISCOVERY.RULE.NAME}

Nazwa reguły wykrywania, która wykryła obecność lub brak urządzenia/usługi.

{DISCOVERY.SERVICE.NAME}

Nazwa wykrytej usługi.
 Na przykład: HTTP.

{DISCOVERY.SERVICE.PORT}

Port wykrytej usługi.
 Na przykład: 80.

{DISCOVERY.SERVICE.STATUS}

Status wykrytej usługi (UP/DOWN).

{DISCOVERY.SERVICE.UPTIME}

Czas, jaki upłynął od ostatniej zmiany statusu wykrywania dla określonej usługi.
 Na przykład: 1h 29m 01s.
 Dla usług ze statusem DOWN jest to okres ich niedostępności.

Makra eskalacji

{ESC.HISTORY}

Dziennik wcześniej wysłanych powiadomień, ich krok eskalacji i status (*wysłane*, *w trakcie* lub *nieudane*).

Makra zdarzeń

{EVENT.ACK.STATUS}

Status potwierdzenia zdarzenia (Tak/Nie).

{EVENT.AGE}

Wiek zdarzenia, które wyzwoliło akcję, z dokładnością do jednej sekundy.
 Przydatne w eskalowanych wiadomościach. Na przykład: 9m 13s.

{EVENT.DATE}

Data zdarzenia, które wyzwoliło akcję, w formacie `rrrr.mm.dd`.
 Na przykład: 2025.04.14.

{EVENT.DURATION}

Czas trwania zdarzenia (różnica czasu między zdarzeniami problemu i odzyskania), z dokładnością do jednej sekundy.
 Przydatne w wiadomościach odzyskiwania problemu.

{EVENT.ID}

Numeryczny identyfikator zdarzenia, które wywołało akcję.

{EVENT.NAME}

Nazwa zdarzenia, które wyzwoliło akcję.

{EVENT.NSEVERITY}

Wartość liczbowa ważności zdarzenia.
 Możliwe wartości: *0* - Niesklasyfikowane, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średni, *4* - Wysoki, *5* - Katastrofa.

{EVENT.OBJECT}

Wartość liczbowa obiektu zdarzenia.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykryty host, *2* - Wykryta usługa, *3* - Autorejestracja, *4* - Pozycja, *5* - Reguła wykrywania niskiego poziomu.

{EVENT.OPDATA}

Dane operacyjne bazowego wyzwalacza problemu.

{EVENT.SEVERITY}

Nazwa ważności zdarzenia.

{EVENT.SOURCE}

Wartość liczbowa źródła zdarzenia.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykrywanie, *2* - Autorejestracja, *3* - Wewnętrzne, *4* - Usługa.

{EVENT.STATUS}

Słowna wartość zdarzenia, które wyzwoliło akcję.

{EVENT.TAGS}

Lista tagów zdarzenia oddzielonych przecinkami.
Rozwijane do pustego ciągu znaków, jeśli nie istnieją żadne tagi.

{EVENT.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia.
Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.

{EVENT.TAGS.<tag name>}

Wartość tagu zdarzenia, do której odwołuje się nazwa tagu.
 Nazwa tagu zawierająca znaki niealfanumeryczne (w tym nieangielskie wielobajtowe znaki UTF) musi być ujęta w podwójny cudzysłów. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłów muszą być poprzedzone znakiem ucieczki, czyli ukośnikiem odwrotnym.

{EVENT.TIME}

Czas zdarzenia, które wyzwoliło akcję, w formacie `hh:mm:ss`.
 Na przykład: 12:57:53.

{EVENT.TIMESTAMP}

Znacznik czasu UNIX zdarzenia, które wyzwoliło akcję.

{EVENT.VALUE}

Wartość liczbowa zdarzenia, które wyzwoliło akcję.
 Możliwe wartości: *1* - problem, *0* - odzyskiwanie.

Makra zdarzeń przyczyny/objawu

{EVENT.CAUSE.ACK.STATUS}

Status potwierdzenia zdarzenia przyczyny (Tak/Nie).
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.AGE}

Wiek zdarzenia będącego przyczyną, z dokładnością do jednej sekundy.
 Przydatne w eskalowanych wiadomościach.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.DATE}

Data zdarzenia będącego przyczyną w formacie `rrrr.mm.dd`.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.DURATION}

Czas trwania zdarzenia przyczyny (różnica czasu między zdarzeniami problemu i odzyskania), z dokładnością do jednej sekundy.
 Przydatne w wiadomościach odzyskiwania problemu.
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.ID}

Numeryczny identyfikator zdarzenia będącego przyczyną.
 Używany w kontekście zdarzenia objawowego.

{EVENT.CAUSE.NAME}

Nazwa zdarzenia przyczyny.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.NSEVERITY}

Wartość liczbowa ważności zdarzenia będącego przyczyną.
 Możliwe wartości: *0* - Niesklasyfikowane, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średnie, *4* - Wysokie, *5* - Katastrofa.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.OBJECT}

Wartość liczbowa obiektu zdarzenia będącego przyczyną.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykryty host, *2* - Wykryta usługa, *3* - Autorejestracja, *4* - Pozycja, *5* - Reguła wykrywania niskiego poziomu.
 Używane w kontekście zdarzenia będącego symptomem.

{EVENT.CAUSE.OPDATA}

Dane operacyjne bazowego wyzwalacza problemu.
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.SEVERITY}

Nazwa poziomu ważności zdarzenia będącego przyczyną.
 Możliwe wartości: *Niesklasyfikowane*, *Informacja*, *Ostrzeżenie*, *Średni*, *Wysoki*, *Katastrofa*.
 Używane w kontekście zdarzenia będącego symptomem.

{EVENT.CAUSE.SOURCE}

Wartość liczbowa źródła zdarzenia przyczyny.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Odkrywanie, *2* - Autorejestracja, *3* - Wewnętrzne.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.STATUS}

Słowna wartość zdarzenia będącego przyczyną.
 Używana w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TAGS}

Lista tagów zdarzenia przyczyny, rozdzielonych przecinkami.
Rozwijane do pustego ciągu znaków, jeśli tagi nie istnieją.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia będącego przyczyną.
Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.
 Używana w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TAGS.<tag name>}

Wartość tagu zdarzenia-przyczyny, do której odwołuje się nazwa tagu.
 Nazwa tagu zawierająca znaki niealfanumeryczne (w tym nieangielskie wielobajtowe znaki UTF) musi być ujęta w podwójny cudzysłów. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłów muszą być poprzedzone ukośnikiem odwrotnym.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.TIME}

Czas zdarzenia będącego przyczyną w formacie `hh:mm:ss`.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TIMESTAMP}

Znacznik czasu UNIX zdarzenia będącego przyczyną.
 Używany w kontekście zdarzenia objawowego.

{EVENT.CAUSE.UPDATE.HISTORY}

Dziennik aktualizacji problemu będącego przyczyną (potwierdzenia itp.).
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.VALUE}

Wartość liczbowa zdarzenia będącego przyczyną.
 Możliwe wartości: *1* - Problem, *0* - Odzyskiwanie.
 Używane w kontekście zdarzenia objawowego.

{EVENT.SYMPTOMS}

Lista zdarzeń symptomów. Zawiera następujące szczegóły: nazwę hosta, nazwę zdarzenia, ważność, wiek, tagi usługi i wartości.
 To makro jest używane w kontekście zdarzenia przyczyny i zwraca informacje o zdarzeniach symptomów.

Makra zdarzenia odzyskania

{EVENT.RECOVERY.DATE}

Data zdarzenia odzyskania w formacie `yyyy.mm.dd`.

{EVENT.RECOVERY.ID}

Numeryczny identyfikator zdarzenia odzyskania.

{EVENT.RECOVERY.NAME}

Nazwa zdarzenia odzyskiwania.

{EVENT.RECOVERY.STATUS}

Słowna wartość zdarzenia odzyskiwania.

{EVENT.RECOVERY.TAGS}

Lista tagów zdarzenia odzyskania, rozdzielona przecinkami. Rozwijana do pustego ciągu znaków, jeśli nie istnieją żadne tagi.

{EVENT.RECOVERY.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia odzyskiwania. Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.

{EVENT.RECOVERY.TIME}

Czas zdarzenia odzyskania w formacie `hh:mm:ss`.

{EVENT.RECOVERY.TIMESTAMP}

Znacznik czasu UNIX zdarzenia odzyskania.

{EVENT.RECOVERY.VALUE}

Wartość liczbowa zdarzenia odzyskiwania.

Makra aktualizacji zdarzenia

{EVENT.UPDATE.ACTION}

Czytelna dla człowieka nazwa działania(ń) wykonanych podczas aktualizacji problemu.
 Przyjmuje następujące wartości: *potwierdzono*, *niepotwierdzono*, *skomentowano*, *zmieniono ważność z (oryginalna ważność) na (zaktualizowana ważność)* oraz *zamknięto* (w zależności od tego, ile działań wykonano w ramach jednej aktualizacji).

{EVENT.UPDATE.ACTIONJSON}

Tablica JSON zawierająca szczegóły działania(ń) wykonanych podczas aktualizacji problemu.
 Możliwe wartości właściwości JSON:
- true (dla właściwości `acknowledge`, `unacknowledge`, `close`, `unsuppress`, `cause` i `symptom`);
- <ciąg znaków wiadomości> (dla właściwości `message`);
- znacznik czasu (dla właściwości `suppress_until` i `timestamp`) lub 0 (dla właściwości `suppress_until`, jeśli wyciszenie jest bezterminowe);
- 0, 1, 2, 3, 4, 5 (dla właściwości ważności `old` i `new`).

Na przykład: `{"acknowledge":true,"message":"Monthly maintenance.","severity":{"old":2,"new":1},"suppress_unti`

{EVENT.UPDATE.DATE}

Data zdarzenia aktualizacji (potwierdzenia itp.) w formacie `yyyy.mm.dd`.
`{ACK.DATE}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{EVENT.UPDATE.HISTORY}

Dziennik aktualizacji problemu (potwierdzenia itp.).
 `{EVENT.ACK.HISTORY}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{EVENT.UPDATE.MESSAGE}

Wiadomość aktualizacji problemu.
 `{ACK.MESSAGE}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{EVENT.UPDATE.NSEVERITY}

Wartość liczbowa nowego poziomu ważności zdarzenia ustawionego podczas operacji aktualizacji problemu.

{EVENT.UPDATE.SEVERITY}

Nazwa nowego poziomu ważności zdarzenia ustawionego podczas operacji aktualizacji problemu.

{EVENT.UPDATE.STATUS}

Wartość numeryczna statusu aktualizacji problemu.
 Możliwe wartości: *0* - webhook został wywołany z powodu zdarzenia problemu/odzyskania, *1* - operacja aktualizacji.

{EVENT.UPDATE.TIME}

Czas aktualizacji zdarzenia (potwierdzenie itp.) w formacie `hh:mm:ss`.
`{ACK.TIME}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{EVENT.UPDATE.TIMESTAMP}

Znacznik czasu UNIX aktualizacji zdarzenia update (potwierdzenia itp.).

Makra funkcji

{FUNCTION.VALUE}

Wartość N-tej funkcji opartej na pozycji w wyrażeniu wyzwalacza w momencie wystąpienia zdarzenia.
 Zliczane są tylko funkcje z `/host/key` jako pierwszym parametrem.

 To makro może być używane z indeksem numerycznym, np. {FUNCTION.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. funkcję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{FUNCTION.RECOVERY.VALUE}

Wartość N-tej funkcji opartej na pozycji w wyrażeniu odzyskiwania w momencie wystąpienia zdarzenia.*
 Liczone są tylko funkcje z `/host/key` jako pierwszym parametrem.

 To makro może być używane z indeksem numerycznym, np. {FUNCTION.RECOVERY.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. funkcję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra hosta

{HOST.CONN}

Adres IP lub nazwa DNS hosta, w zależności od ustawień hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach web makro zostanie rozwinięte do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwinięte do 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.CONN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.DESCRIPTION}

Opis hosta.

 To makro może być używane z indeksem numerycznym jako {HOST.DESCRIPTION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.DNS}

Nazwa DNS hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach web makro zostanie rozwinięte do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwinięte do 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.DNS<1-9>}, aby wskazywać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.HOST}

Techniczna nazwa hosta.
 Funkcje makr nie są obsługiwane dla tego makra, jeśli jest ono używane jako symbol zastępczy w pierwszym parametrze funkcji historii, na przykład `last(/{HOST.HOST}/{ITEM.KEY})`.
 `{HOSTNAME<1-9>}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 To makro może być używane z indeksem numerycznym jako {HOST.HOST<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indeksowane makra.

{HOST.ID}

ID hosta.

 Ten makro może być używane z indeksem numerycznym jako {HOST.ID<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.IP}

Adres IP hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach webowych makro zostanie rozwiązane do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwiązane do wartości 'UNKNOWN'.
 `{IPADDRESS<1-9>}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 To makro może być używane z indeksem numerycznym jako {HOST.IP<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{HOST.METADATA}

Metadane hosta.

 Używane tylko do aktywnej autorejestracji agenta.

{HOST.NAME}

Widoczna nazwa hosta.

 To makro może być używane z indeksem numerycznym jako {HOST.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.PORT}

Port hosta (agent).
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach WWW makro zostanie rozwinięte do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwinięte do 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.PORT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra hosta docelowego

{HOST.TARGET.CONN}

Adres IP lub nazwa DNS hosta docelowego.

{HOST.TARGET.DNS}

Nazwa DNS docelowego hosta.

{HOST.TARGET.HOST}

Techniczna nazwa docelowego hosta.

{HOST.TARGET.IP}

Adres IP docelowego hosta.

{HOST.TARGET.NAME}

Widoczna nazwa docelowego hosta.

Makra grup hostów

{HOSTGROUP.ID}

ID grupy hostów.

Makra inwentarza hosta

{INVENTORY.ALIAS}

Pole *Alias* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.ALIAS<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.ASSET.TAG}

Pole *Asset tag* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.ASSET.TAG<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.CHASSIS}

Pole *Chassis* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.CHASSIS<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.CONTACT}

Pole *Contact* w inwentarzu hosta.
 `{PROFILE.CONTACT<1-9>}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.CONTACT<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.CONTRACT.NUMBER}

Pole *Numer umowy* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.CONTRACT.NUMBER<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.DEPLOYMENT.STATUS}

Pole *Status wdrożenia* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.DEPLOYMENT.STATUS<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HARDWARE}

Pole *Hardware* w inwentarzu hosta.
 {PROFILE.HARDWARE<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.HARDWARE<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyzwalaczu. Zobacz indexed macros.

{INVENTORY.HARDWARE.FULL}

Pole *Sprzęt (pełne szczegóły)* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.HARDWARE.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HOST.NETMASK}

Pole *Maska podsieci hosta* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HOST.NETMASK<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HOST.NETWORKS}

Pole *Sieci hosta* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HOST.NETWORKS<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HOST.ROUTER}

Pole *Host router* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HOST.ROUTER<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.ARCH}

Pole *Architektura sprzętowa* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.ARCH<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.DATE.DECOMM}

Pole *Data wycofania sprzętu z eksploatacji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.DATE.DECOMM<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.DATE.EXPIRY}

Pole *Data wygaśnięcia konserwacji sprzętu* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.DATE.EXPIRY<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.DATE.INSTALL}

Pole *Data instalacji sprzętu* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.DATE.INSTALL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.DATE.PURCHASE}

Pole *Data zakupu sprzętu* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.DATE.PURCHASE<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.INSTALLER.NAME}

Pole *Nazwa instalatora* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.INSTALLER.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.LOCATION}

Pole *Location* w inwentarzu hosta.
 {PROFILE.LOCATION<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.LOCATION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.LOCATION.LAT}

Pole *Szerokość geograficzna lokalizacji* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.LOCATION.LAT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.LOCATION.LON}

Pole *Długość geograficzna lokalizacji* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.LOCATION.LON<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.MACADDRESS.A}

Pole *MAC address A* w inwentarzu hosta.
 {PROFILE.MACADDRESS<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.MACADDRESS.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.MACADDRESS.B}

Pole *MAC address B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.MACADDRESS.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.MODEL}

Pole *Model* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.MODEL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.NAME}

Pole *Name* w inwentarzu hosta.
 {PROFILE.NAME<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.NAME<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyzwalaczu. Zobacz indexed macros.

{INVENTORY.NOTES}

Pole *Notes* w inwentarzu hosta.
 {PROFILE.NOTES<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.NOTES<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.OOB.IP}

Pole *adresu IP OOB* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OOB.IP<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OOB.NETMASK}

Pole *maski podsieci OOB* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OOB.NETMASK<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OOB.ROUTER}

Pole *routera OOB* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OOB.ROUTER<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OS}

Pole *OS* w inwentarzu hosta.
 {PROFILE.OS<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.OS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OS.FULL}

Pole *OS (Full details)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OS.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OS.SHORT}

Pole *OS (Short)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OS.SHORT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.CELL}

Pole *Primary POC cell* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.CELL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.EMAIL}

Pole *E-mail głównego punktu kontaktowego* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.EMAIL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.NAME}

Pole *Nazwa głównego punktu kontaktowego* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.NOTES}

Pole *Uwagi głównego punktu kontaktowego* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.PHONE.A}

Pole *Primary POC phone A* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.PHONE.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.PHONE.B}

Pole *Primary POC phone B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.PHONE.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.PRIMARY.SCREEN}

Pole *Nazwa ekranu głównego punktu kontaktowego* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.SCREEN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.CELL}

Pole *Secondary POC cell* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.CELL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.EMAIL}

Pole *E-mail dodatkowego punktu kontaktowego (POC)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.EMAIL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.NAME}

Pole *Nazwa dodatkowego punktu kontaktowego (POC)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.NOTES}

Pole *Uwagi dodatkowego punktu kontaktowego* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.PHONE.A}

Pole *Secondary POC phone A* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.PHONE.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.PHONE.B}

Pole *Secondary POC phone B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.PHONE.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.SCREEN}

Pole *Nazwa ekranu dodatkowego POC* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.SCREEN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SERIALNO.A}

Pole *Serial number A* w inwentarzu hosta.
 {PROFILE.SERIALNO<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ten makro może być używany z indeksem numerycznym, np. {INVENTORY.SERIALNO.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SERIALNO.B}

Pole *Numer seryjny B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SERIALNO.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ADDRESS.A}

Pole *Adres lokalizacji A* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.ADDRESS.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ADDRESS.B}

Pole *Adres lokalizacji B* w inwentarzu hosta.

 Ten makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.ADDRESS.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ADDRESS.C}

Pole *Adres lokalizacji C* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.ADDRESS.C<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.CITY}

Pole *Miasto lokalizacji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.CITY<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.COUNTRY}

Pole *Kraj lokalizacji* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.SITE.COUNTRY<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.SITE.NOTES}

Pole *Uwagi dotyczące lokalizacji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.RACK}

Pole *Lokalizacja szafy w lokalizacji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.RACK<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.STATE}

Pole *Województwo/stan* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.STATE<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ZIP}

Pole *Kod pocztowy lokalizacji* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.SITE.ZIP<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.SOFTWARE}

Pole *Software* w inwentarzu hosta.
 {PROFILE.SOFTWARE<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.A}

Pole *Aplikacja A* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.B}

Pole *Aplikacja B* oprogramowania w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.C}

Pole *Aplikacja C* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.C<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.D}

Pole *Aplikacja D* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.D<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.E}

Pole *Aplikacja oprogramowania E* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.E<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.FULL}

Pole *Oprogramowanie (pełne szczegóły)* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.SOFTWARE.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.TAG}

Pole *Tag* w inwentarzu hosta.
 {PROFILE.TAG<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.TAG<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyzwalaczu. Zobacz indexed macros.

{INVENTORY.TYPE}

Pole *Type* w inwentarzu hosta.
 {PROFILE.DEVICETYPE<1-9>} jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 Ta makro może być używane z indeksem numerycznym, np. {INVENTORY.TYPE<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.TYPE.FULL}

Pole *Typ (pełne szczegóły)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.TYPE.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.URL.A}

Pole *URL A* w inwentarzu hosta.

 Ta makroinstrukcja może być używana z indeksem numerycznym, np. {INVENTORY.URL.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makroinstrukcje indeksowane.

{INVENTORY.URL.B}

Pole *URL B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.URL.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.URL.C}

Pole *URL C* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.URL.C<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.VENDOR}

Pole *Vendor* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.VENDOR<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra pozycji

{ITEM.DESCRIPTION}

Opis N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.DESCRIPTION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.DESCRIPTION.ORIG}

Oryginalny opis (z nierozwiniętymi makrami) N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.DESCRIPTION.ORIG<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.ID}

Numeryczny identyfikator N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.ID<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.KEY}

Klucz N-tej pozycji w wyrażeniu wyzwalacza, które spowodowało powiadomienie.
 Funkcje makr nie są obsługiwane dla tego makra, jeśli jest używane jako symbol zastępczy w pierwszym parametrze funkcji historii, na przykład `last(/{HOST.HOST}/{ITEM.KEY}).<{TRIGGER.KEY}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

 To makro może być używane z indeksem numerycznym jako {ITEM.KEY<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.KEY.ORIG}

Oryginalny klucz (z nierozwiniętymi makrami) N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.KEY.ORIG<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE}

Najnowsza wartość N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.
 W frontendzie zostanie rozwinięta do *UNKNOWN*, jeśli najnowsza wartość historii została zebrana dawniej niż wynosi czas *Max history display period* (ustawiany w sekcji menu Administration → General).
 W przypadku użycia w nazwie problemu makro nie zostanie rozwinięte do najnowszej wartości pozycji podczas przeglądania zdarzeń problemów; zamiast tego zachowa wartość pozycji z momentu wystąpienia problemu.
W przypadku użycia w powiadomieniach, w niektórych sytuacjach makro może nie zostać rozwinięte do najnowszej wartości pozycji w momencie wyzwolenia wyzwalacza. Na przykład, jeśli pozycja szybko otrzyma dwie wartości, „A” i „B”, a wyzwalacz uruchomi się dla „A”, powiadomienia mogą pokazywać „B” jako najnowszą wartość z powodu niewielkiego opóźnienia przetwarzania — najnowsza wartość pozycji zmieniła się między momentem wyzwolenia wyzwalacza a utworzeniem powiadomienia. Aby tego uniknąć, można użyć makra {ITEM.VALUE}, które rozwija się do wartości z momentu wyzwolenia wyzwalacza, zapewniając użycie poprawnej wartości w powiadomieniu.
 Jest aliasem dla `last(/{HOST.HOST}/{ITEM.KEY})`.
 Rozwinięta wartość dla pozycji tekstowych/logów jest obcinana do 20 znaków przez frontend w następujących miejscach:
- Dane operacyjne;
- Opis wyzwalacza;
- URL-e wyzwalacza;
- Etykiety URL-i wyzwalacza;
- Opis widżetu wartości pozycji.
Aby uzyskać pełną wartość, można użyć funkcji makr, ponieważ żadne wartości nie są obcinane przez serwer. Na przykład: `{{ITEM.LASTVALUE}.regsub("(.*)", \1)}`.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.AGE}

Czas, jaki upłynął między pobraniem ostatniej wartości pozycji a obliczeniem makra.
 Wyświetlany w formacie czytelnym dla użytkownika (np. 1m 45s).

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.DATE}

Data, kiedy została zebrana najnowsza wartość pozycji.
 Wyświetlana w formacie `YYYYMMDD`.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście Monitoring -> Problems), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.TIME}

Czas, kiedy została zebrana najnowsza wartość pozycji.
 Wyświetlany w formacie `HHMMSS`.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście Monitoring -> Problems), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.TIMESTAMP}

Znacznik czasu UNIX określający moment pobrania najnowszej wartości pozycji.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.AGE}

Wiek zdarzenia logu.
Z dokładnością do jednej sekundy.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.DATE}

Data, kiedy wpis dziennika został zapisany do logu.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.EVENTID}

ID zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.EVENTID<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.NSEVERITY}

Numeryczny poziom ważności zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.NSEVERITY<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.SEVERITY}

Słowny poziom ważności zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.SEVERITY<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.SOURCE}

Źródło zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń systemu Windows.

 Ten makro może być używany z indeksem numerycznym, np. {ITEM.LOG.SOURCE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.TIME}

Czas, kiedy wpis dziennika został zapisany do dziennika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.TIMESTAMP}

Znacznik czasu UNIX określający moment zapisania wpisu dziennika w logu.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.NAME}

Nazwa pozycji ze wszystkimi makrami rozwiniętymi.

 To makro może być używane z indeksem numerycznym, np. {ITEM.NAME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.NAME.ORIG}

Oryginalna nazwa (z nierozwiniętymi makrami) pozycji.

 To makro może być używane z indeksem numerycznym, np. {ITEM.NAME.ORIG<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.STATE}

Najnowszy stan N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.
 Możliwe wartości: *Not supported*, *Normal*.

 To makro może być używane z indeksem numerycznym, np. {ITEM.STATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego pozycja stała się nieobsługiwana.
 Jeśli pozycja przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwana, pole błędu może być puste.

{ITEM.VALUE}

Rozwiązuje się do jednego z poniższych:

- Historycznej (z chwili zdarzenia) wartości N-tej pozycji w wyrażeniu wyzwalacza, jeśli jest używana w kontekście zmiany stanu wyzwalacza, na przykład podczas wyświetlania zdarzeń lub wysyłania powiadomień;
- Najnowszej wartości N-tej pozycji w wyrażeniu wyzwalacza, jeśli jest używana bez kontekstu zmiany stanu wyzwalacza, na przykład podczas wyświetlania listy wyzwalaczy w oknie wyboru typu pop-up. W tym przypadku działa tak samo jak {ITEM.LASTVALUE}.

W pierwszym przypadku rozwiąże się do *UNKNOWN*, jeśli wartość historyczna została już usunięta lub nigdy nie została zapisana.
W drugim przypadku, i tylko we frontendzie, rozwiąże się do *UNKNOWN*, jeśli najnowsza wartość historyczna została zebrana wcześniej niż *Max history display period* temu (ustawiane w sekcji menu Administration→General).
Rozwiązana wartość dla pozycji tekstowych/logów jest przycinana przez frontend do 20 znaków w następujących miejscach:
- Dane operacyjne;
- Opis wyzwalacza;
- URL-e wyzwalacza;
- Etykiety URL-i wyzwalacza;
- Opis widżetu wartości

pozycji.
Aby uzyskać pełną wartość, można użyć funkcji makr, ponieważ serwer nie przycina żadnych wartości. Na przykład: {{ITEM.VALUE}.regsub("(.*)", \1)}

To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.AGE}

Czas, jaki upłynął między zebraniem wartości pozycji a obliczeniem makra.
 Wyświetlany w formacie czytelnym dla człowieka (np. 1m 45s).

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.DATE}

Data, kiedy wartość pozycji została zebrana.
 Wyświetlana w formacie YYYYMMDD.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście Monitoring -> Problems), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.TIME}

Czas, kiedy wartość pozycji została zebrana.
 Wyświetlany w formacie HHMMSS.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście Monitoring -> Problems), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.TIMESTAMP}

Znacznik czasu UNIX określający moment zebrania wartości pozycji.

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUETYPE}

Typ wartości N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.
 Możliwe wartości: *0* - Liczba zmiennoprzecinkowa, *1* - Znak, *2* - Log, *3* - Liczba całkowita bez znaku, *4* - Tekst.

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUETYPE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra powiadomień wykrywania niskiego poziomu

{LLDRULE.DESCRIPTION}

Opis reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.DESCRIPTION.ORIG}

Oryginalny opis (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.ID}

Numeryczny identyfikator reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.KEY}

Klucz reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.KEY.ORIG}

Oryginalny klucz (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.NAME}

Nazwa reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.NAME.ORIG}

Oryginalna nazwa (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.STATE}

Najnowszy stan reguły wykrywania niskiego poziomu.
 Możliwe wartości: *Nieobsługiwane*, *Normalny*.

 To makro może być używane z indeksem numerycznym, np. {LLDRULE.STATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{LLDRULE.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego reguła wykrywania niskiego poziomu stała się nieobsługiwana.
 Jeśli reguła wykrywania niskiego poziomu przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwana, pole błędu może być puste.

Makra mapy

{MAP.ID}

ID mapy sieciowej.

{MAP.NAME}

Nazwa mapy sieciowej.

Makra proxy

{PROXY.DESCRIPTION}

Opis proxy.
 Rozwijane jako jedno z poniższych:

- Proxy N-tej pozycji w wyrażeniu wyzwalacza (w powiadomieniach opartych na wyzwalaczach). Można tutaj używać makr indeksowanych;
- Proxy, które wykonało wykrywanie (w powiadomieniach o wykrywaniu). Użyj tutaj {PROXY.DESCRIPTION}, bez indeksowania;
- Proxy, do którego zarejestrował się aktywny agent (w powiadomieniach o autorejestracji). Użyj tutaj {PROXY.DESCRIPTION}, bez indeksowania.

To makro może być używane z indeksem numerycznym, np. {PROXY.DESCRIPTION<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{PROXY.NAME}

Nazwa proxy.
 Rozwiązuje się do jednej z następujących wartości:

- Proxy N-tej pozycji w wyrażeniu wyzwalacza (w powiadomieniach opartych na wyzwalaczach). Można tutaj używać makr indeksowanych;
- Proxy, który wykonał wykrywanie (w powiadomieniach o wykrywaniu). Użyj tutaj {PROXY.NAME}, bez indeksowania;
- Proxy, do którego zarejestrował się aktywny agent (w powiadomieniach o autorejestracji). Użyj tutaj {PROXY.NAME}, bez indeksowania.

To makro może być używane z indeksem numerycznym, np. {PROXY.NAME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra skryptu

{MANUALINPUT}

Wartość ręcznego wejścia określona przez użytkownika w czasie wykonywania skryptu.

Makra usługi

{SERVICE.DESCRIPTION}

Opis usługi z rozwiązanymi makrami.

{SERVICE.ID}

Numeryczny identyfikator usługi, która wywołała akcję.

{SERVICE.NAME}

Nazwa usługi z rozwiniętymi makrami.

{SERVICE.ROOTCAUSE}

Lista zdarzeń problemowych wyzwalaczy, które spowodowały awarię usługi, posortowana według ważności i nazwy hosta.
 Zawiera następujące szczegóły: nazwa hosta, nazwa zdarzenia, ważność, wiek, tagi usługi i wartości.

{SERVICE.TAGS}

Lista tagów zdarzenia usługi rozdzielonych przecinkami.
 Tagi zdarzenia usługi można zdefiniować w sekcji tagów konfiguracji usługi. Rozwijane do pustego ciągu znaków, jeśli nie istnieją żadne tagi.

{SERVICE.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia usługi.
 Tagi zdarzenia usługi można zdefiniować w sekcji tagów konfiguracji usługi. Rozwijane do pustej tablicy, jeśli nie istnieją żadne tagi.

{SERVICE.TAGS.<tag name>}

Wartość tagu zdarzenia usługi wskazana przez nazwę tagu.
 Tagi zdarzeń usługi można zdefiniować w sekcji tagów konfiguracji usługi.
 Nazwa tagu zawierająca znaki niealfanumeryczne (w tym nieangielskie wielobajtowe znaki UTF) musi być ujęta w podwójny cudzysłów. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłów muszą być poprzedzone ukośnikiem odwrotnym.

Makra wyzwalaczy

{TRIGGER.DESCRIPTION}

Opis wyzwalacza.
 Wszystkie makra obsługiwane w opisie wyzwalacza zostaną rozwinięte, jeśli `{TRIGGER.DESCRIPTION}` zostanie użyte w tekście powiadomienia.
 `{TRIGGER.COMMENT}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{TRIGGER.EVENTS.ACK}

Liczba potwierdzonych zdarzeń dla elementu mapy na mapach lub dla wyzwalacza, który wygenerował bieżące zdarzenie w powiadomieniach.

{TRIGGER.EVENTS.PROBLEM.ACK}

Liczba potwierdzonych zdarzeń problemów dla wszystkich wyzwalaczy, niezależnie od ich stanu.

{TRIGGER.EVENTS.PROBLEM.UNACK}

Liczba niepotwierdzonych zdarzeń problemowych dla wszystkich wyzwalaczy, niezależnie od ich stanu.

{TRIGGER.EVENTS.UNACK}

Liczba niepotwierdzonych zdarzeń dla elementu mapy na mapach lub dla wyzwalacza, który wygenerował bieżące zdarzenie w powiadomieniach.

{TRIGGER.EXPRESSION}

Wyrażenie wyzwalacza.

{TRIGGER.EXPRESSION.EXPLAIN}

Częściowo obliczone wyrażenie wyzwalacza.
 Funkcje oparte na pozycjach są obliczane i zastępowane wynikami w momencie generowania zdarzenia, natomiast wszystkie pozostałe funkcje są wyświetlane w postaci zapisanej w wyrażeniu. Może być używane do debugowania wyrażeń wyzwalaczy.

{TRIGGER.EXPRESSION.RECOVERY}

Wyrażenie odzyskiwania wyzwalacza, jeśli *OK event generation* w konfiguracji wyzwalacza jest ustawione na „Recovery expression”; w przeciwnym razie zwracany jest pusty ciąg.

{TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}

Częściowo obliczone wyrażenie odzyskiwania wyzwalacza.
 Funkcje oparte na pozycjach są obliczane i zastępowane wynikami w momencie generowania zdarzenia, natomiast wszystkie pozostałe funkcje są wyświetlane w postaci zapisanej w wyrażeniu. Może być używane do debugowania wyrażeń odzyskiwania wyzwalacza.

{TRIGGER.HOSTGROUP.NAME}

Posortowana (według zapytania SQL) lista grup hostów, rozdzielonych przecinkiem i spacją, w których zdefiniowany jest wyzwalacz.

{TRIGGER.ID}

Numeryczny identyfikator wyzwalacza, który uruchomił tę akcję.

{TRIGGER.NAME}

Nazwa wyzwalacza z rozwiniętymi makrami.
 Zwróć uwagę, że `{EVENT.NAME}` może być używane w akcjach do wyświetlania nazwy wywołanego zdarzenia/problemu z rozwiniętymi makrami.

{TRIGGER.NAME.ORIG}

Oryginalna nazwa wyzwalacza (z nierozwiniętymi makrami).

{TRIGGER.NSEVERITY}

Numeryczny poziom ważności wyzwalacza.
 Możliwe wartości: *0* - Niesklasyfikowany, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średni, *4* - Wysoki, *5* - Katastrofa.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.ACK}

Liczba potwierdzonych zdarzeń problemów dla wyzwalaczy w stanie problemu.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.UNACK}

Liczba niepotwierdzonych zdarzeń problemów dla wyzwalaczy w stanie problemu.

{TRIGGER.SEVERITY}

Nazwa ważności wyzwalacza.
 Może być zdefiniowana w *Administration > General > Trigger displaying options*.

{TRIGGER.STATE}

Najnowszy stan wyzwalacza.
 Możliwe wartości: *Unknown*, *Normal*.

{TRIGGER.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego wyzwalacz stał się nieobsługiwany.
 Jeśli wyzwalacz przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwany, pole błędu może być puste.

{TRIGGER.STATUS}

Wartość wyzwalacza w momencie wykonania kroku operacji.
 Możliwe wartości: *Problem*, *OK*.
 `{STATUS}` jest przestarzałe i może nie zostać rozwiązane w obsługiwanych lokalizacjach.

{TRIGGER.TEMPLATE.NAME}

Posortowana (według zapytania SQL) lista szablonów, oddzielona przecinkami i spacjami, w których zdefiniowano wyzwalacz, lub *UNKNOWN*, jeśli wyzwalacz jest zdefiniowany w hoście.

{TRIGGER.URL}

URL wyzwalacza.

{TRIGGER.URL.NAME}

Etykieta dla URL wyzwalacza.

{TRIGGER.VALUE}

Aktualna numeryczna wartość wyzwalacza.
 Możliwe wartości: *0* - wyzwalacz ma status OK, *1* - wyzwalacz ma status Problem.

{TRIGGERS.ACK}

Liczba potwierdzonych wyzwalaczy dla elementu mapy, bez uwzględniania stanu wyzwalacza.
 Wyzwalacz jest uznawany za potwierdzony, jeśli wszystkie jego zdarzenia problemowe są potwierdzone.

{TRIGGERS.PROBLEM.ACK}

Liczba potwierdzonych wyzwalaczy problemów dla elementu mapy.
 Wyzwalacz jest uznawany za potwierdzony, jeśli wszystkie jego zdarzenia problemowe są potwierdzone.

{TRIGGERS.PROBLEM.UNACK}

Liczba niepotwierdzonych wyzwalaczy problemów dla elementu mapy.
 Wyzwalacz jest uznawany za niepotwierdzony, jeśli co najmniej jedno z jego zdarzeń problemu nie zostało potwierdzone.

{TRIGGERS.UNACK}

Liczba niepotwierdzonych wyzwalaczy dla elementu mapy, niezależnie od stanu wyzwalacza.
 Wyzwalacz jest uznawany za niepotwierdzony, jeśli co najmniej jedno z jego zdarzeń problemowych nie zostało potwierdzone.

Makra nazwy użytkownika

{USER.FULLNAME}

Imię, nazwisko i nazwa użytkownika użytkownika, który dodał potwierdzenie zdarzenia lub uruchomił skrypt.

{USER.NAME}

Nazwa użytkownika, który uruchomił skrypt.

{USER.SURNAME}

Nazwisko użytkownika, który uruchomił skrypt.

{USER.USERNAME}

Nazwa użytkownika, który uruchomił skrypt.
 `{USER.ALIAS}` jest teraz przestarzałe i może nie być rozpoznawane w obsługiwanych lokalizacjach.

Makra indeksowane

Składnia makr indeksowanych **{MACRO<1-9>}** może odwoływać się wyłącznie do n-tej pozycji, funkcji lub hosta w polu *Expression* wyzwalacza:

- **{HOST.IP1}, {HOST.IP2}, {HOST.IP3}** są rozwijane do adresów IP pierwszego, drugiego i trzeciego hosta w wyrażeniu wyzwalacza (jeśli występują).
- **{ITEM.VALUE1}, {ITEM.VALUE2}, {ITEM.VALUE3}** są rozwijane do wartości pierwszej, drugiej i trzeciej pozycji w wyrażeniu wyzwalacza w momencie zdarzenia (jeśli występują).
- **{FUNCTION.VALUE1}, {FUNCTION.VALUE2}, {FUNCTION.VALUE3}** są rozwijane do wartości pierwszej, drugiej i trzeciej funkcji opartej na pozycji w momencie zdarzenia (jeśli występują).

W kontekście wyzwalaczy makra indeksowane zawsze odnoszą się do pola *Expression* konfiguracji wyzwalacza, a nie do *Recovery expression*. Na przykład w zdarzeniu odzyskania **{ITEM.VALUE2}** zostanie rozwinięte do wartości drugiej pozycji z wyrażenia problemu w momencie odzyskania.

Makro **{HOST.HOST<1-9>}** jest również obsługiwane w makrze wyrażenia `{?func(/host/key,param)}` w **nazwach wykresów**. Na przykład `{?func(/{HOST.HOST2}/key,param)}` w nazwie wykresu zostanie rozwinięte do hosta drugiej pozycji na wykresie.

Warning:
Makra indeksowane nie zostaną rozwinięte w żadnym innym kontekście poza przypadkami wymienionymi tutaj. W innych kontekstach używaj makr bez indeksu (**{HOST.HOST}, {HOST.IP}** itd.).

2 Makra użytkownika obsługiwane według lokalizacji

Przegląd

Ta sekcja zawiera listę miejsc, w których obsługiwane są makra definiowane przez użytkownika.

Note:
Dla *Akcji*, *Odkrywania sieci*, *Proxy* oraz wszystkich lokalizacji wymienionych w sekcji *Inne lokalizacje* na tej stronie obsługiwane są tylko makra użytkownika na poziomie globalnym. W wymienionych lokalizacjach makra na poziomie hosta i na poziomie szablonu nie będą rozwijane.

Note:
Aby dostosować wartości makr (na przykład skrócić je lub wyodrębnić określone podciągi), można użyć funkcji makr.

Akcje

W akcjach makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Powiadomienia i polecenia oparte na wyzwalaczach		tak
Wewnętrzne powiadomienia oparte na wyzwalaczach		tak
Powiadomienia o aktualizacji problemu		tak
Powiadomienia i polecenia oparte na usługach		tak
Powiadomienia o aktualizacji usługi		tak
Warunek okresu czasu		nie
Operacje		
	Domyślny czas trwania kroku operacji	nie
	Czas trwania kroku	nie

Hosty/prototypy hostów

W konfiguracji hosta i prototypu hosta makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
IP/DNS interfejsu		tylko DNS
Port interfejsu		nie
SNMP v1, v2		
	Społeczność SNMP	tak
SNMP v3		
	Nazwa kontekstu	tak
	Nazwa bezpieczeństwa	tak
	Hasło uwierzytelniania	tak

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
	Hasło prywatności	tak
IPMI		
	Nazwa użytkownika	tak
	Hasło	tak
Tagi[2]		
	Nazwy tagów	tak
	Wartości tagów	tak

Pozycje / prototypy pozycji

W konfiguracji pozycji lub prototypu pozycji makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Nazwa pozycji	tak
Parametry klucza pozycji	tak
Interwał aktualizacji	nie
Niestandardowe interwały	nie
Limit czasu (dostępny dla obsługiwanych typów pozycji)	nie
Przechowuj do (dla historii i trendów)	nie
Opis	tak
Pozycja obliczana/agregująca	
Formuła (stałe wyrażenia i parametry funkcji; parametry klucza pozycji; (*tylko pozycja agregująca*) warunki filtra (nazwa grupy hostów i nazwa tagu))	tak
Monitor bazy danych	
Nazwa użytkownika	tak
Hasło	tak
Zapytanie SQL	tak
HTTP agent	

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
URL[3]	tak
Pola zapytania	tak
Treść żądania	tak
Nagłówki (nazwy i wartości)	tak
Wymagane kody statusu	tak
HTTP proxy	tak
Nazwa użytkownika do uwierzytelniania HTTP	tak
Hasło do uwierzytelniania HTTP	tak
Plik certyfikatu SSL	tak
Plik klucza SSL	tak
Hasło klucza SSL	tak
Dozwolone hosty	tak
JMX agent	
Punkt końcowy JMX	tak
Pozycja skryptowa	
Nazwy i wartości parametrów	tak
Pozycja przeglądarki	
Nazwy i wartości parametrów	tak
SNMP agent	
OID SNMP	tak
SSH agent	
Nazwa użytkownika	tak
Plik klucza publicznego	tak
Plik klucza prywatnego	tak
Hasło	tak
Skrypt	tak
TELNET agent	
Nazwa użytkownika	tak
Hasło	tak
Skrypt	tak
Zabbix trapper	
Dozwolone hosty	tak
Tagi[2]	
Nazwy tagów	tak
Wartości tagów	tak
Kroki przetwarzania wstępnego	
Parametry (w tym niestandardowe skrypty)	tak
Parametry niestandardowej obsługi błędów (pola *Ustaw wartość na* i *Ustaw błąd na*)	tak

Odkrywanie niskiego poziomu

W regule odkrywania niskiego poziomu makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/
Parametry klucza	tak
Interwał aktualizacji	nie

Lokalizacja		Wiele makr/
Interwał niestandardowy		nie
Limit czasu (dostępny dla obsługiwanych typów pozycji)		nie
Usuń utracone zasoby		nie
Wyłącz utracone zasoby		nie
Opis		tak
SNMP agent		
	OID SNMP	tak
SSH agent		
	Nazwa użytkownika	tak
	Plik klucza publicznego	tak
	Plik klucza prywatnego	tak
	Hasło	tak
	Skrypt	tak
TELNET agent		
	Nazwa użytkownika	tak
	Hasło	tak
	Skrypt	tak
Zabbix trapper		
	Dozwolone hosty	tak
Monitor bazy danych		
	Nazwa użytkownika	tak
	Hasło	tak
	Zapytanie SQL	tak
JMX agent		
	Punkt końcowy JMX	tak
HTTP agent		
	URL[3]	tak
	Pola zapytania	tak
	Treść żądania	tak
	Nagłówki (nazwy i wartości)	tak
	Wymagane kody statusu	tak
	Nazwa użytkownika do uwierzytelniania HTTP	tak
	Hasło do uwierzytelniania HTTP	tak
Filtry		
	Wyrażenie regularne	tak
Nadpisania		
	Filtry: wyrażenie regularne	tak
	Operacje: interwał aktualizacji (dla prototypów pozycji)	nie
	Operacje: okres przechowywania historii (dla prototypów pozycji)	nie
	Operacje: okres przechowywania trendów (dla prototypów pozycji)	nie

Odkrywanie sieci

W regule odkrywania sieci makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Interwał aktualizacji		nie
SNMP v1, v2		
	Społeczność SNMP	tak
	OID SNMP	tak
SNMP v3		
	Nazwa kontekstu	tak
	Nazwa bezpieczeństwa	tak
	Hasło uwierzytelniania	tak
	Hasło prywatności	tak
	OID SNMP	tak

Proxy

W konfiguracji proxy makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Adres dla aktywnych agentów > *Port* (jeśli proxy należy do grupy)	nie
Adres interfejsu i port (dla pasywnego proxy)	nie
Timeouty dla typów pozycji	nie

Grupy proxy

W konfiguracji grupy proxy makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Okres przełączenia awaryjnego	nie
Minimalna liczba proxy	nie

Szablony

W konfiguracji szablonu makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Tagi[2]		
	Nazwy tagów	tak
	Wartości tagów	tak

Wyzwalacze

W konfiguracji wyzwalacza makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Nazwa	tak
Dane operacyjne	tak
Wyrażenie (tylko w stałych i parametrach funkcji; tajne makra nie są obsługiwane)	tak
Tag do dopasowania	tak
Nazwa wpisu menu	tak
URL wpisu menu[3]	tak

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Opis	tak
Tagi[2]	
Nazwy tagów	tak
Wartości tagów	tak

Scenariusz web

W konfiguracji scenariusza web makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Nazwa		tak
Interwał aktualizacji		nie
Agent		tak
HTTP proxy		tak
Zmienne (tylko wartości)		tak
Nagłówki (nazwy i wartości)		tak
Kroki		
	Nazwa	tak
	URL[3]	tak
	Zmienne (tylko wartości)	tak
	Nagłówki (nazwy i wartości)	tak
	Limit czasu	nie
	Wymagany ciąg znaków	tak
	Wymagane kody statusu	nie
Uwierzytelnianie		
	Użytkownik	tak
	Hasło	tak
	Certyfikat SSL	tak
	Plik klucza SSL	tak
	Hasło klucza SSL	tak
Tagi[2]		
	Nazwy tagów	tak
	Wartości tagów	tak

Inne lokalizacje

Oprócz lokalizacji wymienionych tutaj, makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Globalne skrypty (URL, skrypt, SSH, Tel-net, IPMI), w tym tekst potwierdzenia	yes
Webhooki	
Skrypt JavaScript	no
Nazwa parametru skryptu JavaScript	no
Wartość parametru skryptu JavaScript	yes
Pulpity	
Kolumna typu danych *Tekst* w widżecie pulpitu *Top hosts*	yes
Parametr *Opis* w widżecie pulpitu *Wartość pozycja* i *Gauge*	yes

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Parametr *Tekst* etykiety głównej/drugorzędnej w widżecie pulpitu *Honeycomb*	yes
Parametr *URL*[3] w widżecie pulpitu *URL*	yes
Użytkownicy → *Użytkownicy* → *Media*	
Gdy aktywne	no
Administracja → *Ogólne* → *GUI*	
Czas pracy	no
Administracja → *Ogólne* → *Limity czasu*	
Limity czasu dla typów pozycji	no
Administracja → *Ogólne* → *Łączniki*	
URL	yes
Nazwa użytkownika	yes
Hasło	yes
Token Bearer	yes
Limit czasu	no
Proxy HTTP	yes
Plik certyfikatu SSL	yes
Plik klucza SSL	yes
Hasło klucza SSL	yes
Alarmy → *Typy mediów* → *Szablony wiadomości*	
Temat	yes
Wiadomość	yes
Alarmy → *Typy mediów* → *Skrypt*	
Parametry skryptu	yes

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Alarmy → *Typy mediów* → *Typ medium*	
Pola *Nazwa użytkownika* i *Hasło* dla typu medium *Email* (gdy *Uwierzytelnianie* jest ustawione na "Nazwa użytkownika i hasło"; zalecane są sekretne makra)	yes

Pełną listę wszystkich makr obsługiwanych w Zabbiksie można znaleźć w sekcji obsługiwane makra.

Przypisy

[1] Jeśli wiele makr w polu lub makra zmieszane z tekstem nie są obsługiwane w danej lokalizacji, pojedyncze makro musi wypełniać całe pole.

[2] Makra używane w nazwach i wartościach tagów są rozwijane tylko podczas procesu generowania zdarzenia.

[3] Adresy URL zawierające tajne makro nie będą działać, ponieważ makro w nich zostanie rozwinięte jako "******".

7 Symbole jednostek

Przegląd

Praca z dużymi wartościami, takimi jak 86400, 104857600 lub 1000000, może być trudna i prowadzić do błędów. Aby uprościć konfigurację i poprawić czytelność, Zabbix obsługuje symbole jednostek (sufiksy), które działają jako mnożniki wartości.

Wyrażenia wyzwalaczy bez sufiksów:

```
last(/host/system.uptime)<86400
avg(/host/system.cpu.load,600s)<10
last(/host/vm.memory.size[available])<20971520
```

Wyrażenia wyzwalaczy z sufiksami:

```
last(/host/system.uptime)<1d
avg(/host/system.cpu.load,10m)<10
last(/host/vm.memory.size[available])<20M
```

Sufiksy mogą również uprościć konfigurację innych obiektów, takich jak pozycje, widgety itp., a także pomóc w wyświetlaniu wartości pozycji w formacie czytelnym dla człowieka.

Note:
Aby sprawdzić, czy pole konfiguracji danego obiektu obsługuje sufiksy, zawsze należy zajrzeć na odpowiednią stronę dotyczącą konfigurowanego obiektu.

Przyrostki czasu

Frontend Zabbixa obsługuje następujące przyrostki czasu w konfiguracji elementów:
 s - sekundy *(w użyciu działa tak samo jak wartość surowa)*;
 m - minuty;
 h - godziny;
 d - dni;
 w - tygodnie;
 M - miesiące *(tylko funkcje trendów)*;
 y - lata *(tylko funkcje trendów)*.

Przyrostki czasu są obsługiwane tylko dla liczb całkowitych. Na przykład 1h jest obsługiwane, ale 1,5h lub 1.5h nie są obsługiwane; zamiast tego użyj 90m.

Przyrostki rozmiaru pamięci

Zabbix obsługuje następujące przyrostki rozmiaru pamięci:
 K - kilobajty
 M - megabajty
 G - gigabajty
 T - terabajty

Przyrostki wartości pozycji

Przyrostki mogą być również używane do wyświetlania numerycznych wartości pozycji w czytelnym dla człowieka formacie.

Aby to włączyć, użyj jednego z następujących przyrostków w polu *Units* podczas konfigurowania pozycji:
 B - bajty;
 Bps - bajty na sekundę;
 s - sekundy, wyświetlane z użyciem maksymalnie trzech największych niezerowych jednostek czasu;
 uptime - czas, który upłynął, w formacie hh:mm:ss lub N dni, hh:mm:ss;
 unixtime - znacznik czasu Unix, sformatowany jako yyyy.mm.dd hh:mm:ss.

Do interpretacji i wyświetlania tych przyrostków mają zastosowanie następujące dodatkowe reguły:

- Dla B i Bps Zabbix używa konwersji opartej na systemie dwójkowym (1K = 1024B), zgodnie ze standardem JEDEC.
- Dla innych jednostek (takich jak Hz, W itp.) Zabbix używa konwersji dziesiętnej (1K = 1000).
- Dla s (sekund):
  - Format obejmuje yyy mmm ddd hhh mmm sss ms; wyświetlane są tylko maksymalnie trzy największe niezerowe jednostki czasu (np. 1M 10d 4h).
  - Jeśli jednostka ma wartość zero i znajduje się między dwiema niezerowymi jednostkami, jest pomijana (np. 10d 56m zamiast 10d 0h 56m).

Gdy używane są *Units*, do wartości pozycji automatycznie stosowane są następujące przyrostki mnożnika:
 K, **M**, **G**, **T** - kilo, mega, giga, tera;
 P, **E**, **Z**, **Y** - peta, eksa, zetta, yotta *(są one stosowane tylko w frontend)*.

Aby zapobiec konwersji jednostek, użyj prefiksu `!` (np. !B lub !s).

Poniższe przykłady pokazują, jak odebrane wartości pozycji są konwertowane na podstawie określonych jednostek:

```
1 B → 1 B
1024 B → 1 KB
1536 B → 1.5 KB
881764 B → 881.76 KB
881764 !B → 881764 B

0.0000155 s → 0.016ms
3470400 s → 1M 10d 4h
2606400 s → 1M 4h
2592000 s → 1M
2592001 s → 1M
2592001 !s → 2592001 s

17764 uptime → 04:56:04
86400 uptime → 1 day, 00:00:00
881764 uptime → 10 days, 04:56:04
32417764 uptime → 375 days, 04:56:04
32417764 !uptime → 32417764 uptime

881764 unixtime → 1970-01-11 04:56:04 AM

17764 Hz → 17.76 KHz
86400 Hz → 86.4 KHz
881764 Hz → 881.76 KHz
32417764 Hz → 32.42 MHz

0 ! → 0
0 !! → 0 !
```

Note:
Przed Zabbix 4.0 istniała zakodowana na stałe lista wykluczeń jednostek, obejmująca `ms`, `rpm`, `RPM`, `%`. Lista ta została wycofana, dlatego poprawnym sposobem zapobiegania konwersji takich jednostek jest `!ms`, `!rpm`, `!RPM`, `!%`.

8 Składnia okresu czasu

Przegląd

Aby ustawić przedział czasu, należy użyć następującego formatu:

`d-d,hh:mm-hh:mm`

gdzie symbole oznaczają:

Symbol	Opis
d	Dzień tygodnia: 1 - poniedziałek, 2 - wtorek, ... , 7 - niedziela
hh	Godziny: 00-24
mm	Minuty: 00-59

Możesz określić więcej niż jeden przedział czasu, używając separatora średnika (;):

`d-d,hh:mm-hh:mm;d-d,hh:mm-hh:mm...`

Pozostawienie przedziału czasu pustego jest równoznaczne z 1-7,00:00-24:00, co jest wartością domyślną.

Attention:
Górna granica przedziału czasu nie jest uwzględniana. Dlatego jeśli określisz 09:00-18:00, ostatnią sekundą uwzględnioną w tym przedziale czasu jest 17:59:59.

Przykłady

Godziny pracy. Od poniedziałku do piątku od 9:00 do 18:00:

`1-5,09:00-18:00`

Godziny pracy plus weekend. Od poniedziałku do piątku od 9:00 do 18:00 oraz w sobotę i niedzielę od 10:00 do 16:00:

`1-5,09:00-18:00;6-7,10:00-16:00`

9 Wykonywanie poleceń

Zabbix używa wspólnej funkcjonalności dla kontroli zewnętrznych, parametrów użytkownika, pozycji system.run, niestandardowych skryptów alertów, zdalnych poleceń i globalnych skryptów.

Kroki wykonywania

Note:
Domyślnie wszystkie skrypty w Zabbix są wykonywane przy użyciu powłoki *sh* i nie ma możliwości zmiany domyślnej powłoki. Aby użyć innej powłoki, można zastosować obejście: utworzyć plik skryptu i wywołać ten skrypt podczas wykonywania polecenia.

Polecenie/skrypt jest wykonywane w podobny sposób zarówno na platformach Unix, jak i Windows:

1. Zabbix (proces nadrzędny) tworzy potok do komunikacji
2. Zabbix ustawia potok jako wyjście dla tworzonego procesu potomnego
3. Zabbix tworzy proces potomny (uruchamia polecenie/skrypt)
4. Dla procesu potomnego tworzona jest nowa grupa procesów (w Unix) lub zadanie (w Windows)
5. Zabbix odczytuje z potoku do momentu wystąpienia limitu czasu lub gdy nikt nie zapisuje do drugiego końca (WSZYSTKIE uchwyty/deskriptory plików zostały zamknięte). Należy pamiętać, że proces potomny może utworzyć więcej procesów i zakończyć działanie, zanim one zakończą działanie lub zamkną uchwyt/deskrpytor pliku.
6. Jeśli limit czasu nie został osiągnięty, Zabbix czeka, aż początkowy proces potomny zakończy działanie lub wystąpi limit czasu
7. Jeśli początkowy proces potomny zakończył działanie i limit czasu nie został osiągnięty, Zabbix sprawdza kod wyjścia początkowego procesu potomnego i porównuje go z 0 (wartość różna od zera jest uznawana za błąd wykonania, tylko dla niestandardowych skryptów alertów, zdalnych poleceń i skryptów użytkownika wykonywanych na serwer i proxy Zabbix)
8. Na tym etapie zakłada się, że wszystko zostało wykonane i całe drzewo procesów (tj. grupa procesów lub zadanie) jest kończone

Attention:
Zabbix zakłada, że polecenie/skrypt zakończyło przetwarzanie, gdy początkowy proces potomny zakończył działanie ORAZ żaden inny proces nie utrzymuje już otwartego uchwytu wyjściowego/deskriptora pliku. Gdy przetwarzanie zostanie zakończone, WSZYSTKIE utworzone procesy są kończone.

Wszystkie podwójne cudzysłowy i ukośniki odwrotne w poleceniu są poprzedzane ukośnikami odwrotnymi, a polecenie jest ujmowane w podwójne cudzysłowy.

Sprawdzanie kodu wyjścia

Kody wyjścia są sprawdzane przy następujących warunkach:

- Tylko dla niestandardowych skryptów alertów, zdalnych poleceń i skryptów użytkownika wykonywanych na serwer Zabbix i proxy Zabbix.
- Każdy kod wyjścia różny od 0 jest uznawany za błąd wykonania.
- Zawartość standardowego strumienia błędów i standardowego wyjścia dla nieudanych wykonań jest zbierana i dostępna we frontendzie (gdzie wynik wykonania jest wyświetlany).
- Dodatkowy wpis w logu może zostać utworzony dla zdalnych poleceń wykonywanych na Zabbix agent/proxy po włączeniu parametru LogRemoteCommands w agent/proxy.

Możliwe komunikaty we frontendzie i wpisy w logu dla nieudanych poleceń/skryptów:

- Zawartość standardowego strumienia błędów i standardowego wyjścia dla nieudanych wykonań (jeśli występuje).
- ”Proces zakończył działanie z kodem: N.” (dla pustego wyjścia i kodu wyjścia różnego od 0).
- ”Proces został zakończony sygnałem: N.” (dla procesu zakończonego sygnałem, tylko w systemie Linux).
- ”Proces został nieoczekiwanie zakończony.” (dla procesu zakończonego z nieznanych przyczyn).

Zobacz także

- Kontrole zewnętrzne
- Parametry użytkownika
- pozycje system.run
- Niestandardowe skrypty alertów
- Polecenia zdalne
- Skrypty globalne

10 Zgodność wersji

Obsługiwane agenty

Aby zapewnić zgodność z Zabbix 7.4, agent Zabbix nie może być starszy niż wersja 1.4 i nie może być nowszy niż 7.4.

Może być konieczne sprawdzenie konfiguracji starszych agentów, ponieważ niektóre parametry uległy zmianie, na przykład parametry związane z rejestrowaniem w wersjach wcześniejszych niż 3.0.

Aby w pełni wykorzystać najnowszą funkcjonalność, metryki, lepszą wydajność i mniejsze zużycie pamięci, używaj najnowszego obsługiwanego agenta.

Uwagi dotyczące Windows XP

- W 32-bitowym Windows XP nie używaj agentów Zabbix nowszych niż 6.0.x;
- W Windows XP/Server 2003 nie używaj szablonów agentów nowszych niż Zabbix 4.0.x. Nowsze szablony używają angielskich liczników wydajności, które są obsługiwane dopiero od Windows Vista/Server 2008.

Obsługiwane agenty 2

Starsze agenty 2 Zabbix od wersji 4.4 wzwyż są zgodne z Zabbix 7.4; agent 2 Zabbix nie może być nowszy niż 7.4.

Należy pamiętać, że podczas korzystania z agentów 2 Zabbix w wersjach 4.4 i 5.0 używany jest domyślny interwał 10 minut do odświeżania nieobsługiwanych pozycji.

Aby w pełni wykorzystać najnowszą funkcjonalność, metryki, lepszą wydajność i mniejsze zużycie pamięci, używaj najnowszego obsługiwanego agenta 2.

Obsługiwane proxy Zabbix

Aby zapewnić pełną zgodność z Zabbix 7.4, proxy muszą odpowiadać głównej wersji serwera. Tylko proxy Zabbix 7.4.x są w pełni zgodne z serwerem Zabbix 7.4.x.

Przestarzałe proxy są obsługiwane częściowo: nadal mogą zbierać dane i wykonywać skrypty, ale nie mogą otrzymywać aktualizacji konfiguracji, takich jak nowe pozycje.

W odniesieniu do serwera Zabbix, proxy mogą być:

- *Aktualne* (proxy i serwer mają tę samą wersję główną);
- *Nieaktualne* (wersja proxy jest starsza niż wersja serwera, ale jest częściowo obsługiwana);
- *Niewspierane* (wersja proxy jest starsza niż wersja poprzedniej wersji LTS serwera *lub* wersja proxy jest nowsza niż główna wersja serwera).

Przykłady:

Wersja serwera	*Aktualna* wersja proxy	*Nieaktualna* wersja proxy	*Niewspierana* wersja proxy
6.4	6.4	6.0, 6.2	Starsza niż 6.0; nowsza niż 6.4
7.0	7.0	6.0, 6.2, 6.4	Starsza niż 6.0; nowsza niż 7.0
7.2	7.2	7.0	Starsza niż 7.0; nowsza niż 7.2
7.4	7.4	7.0	Starsza niż 7.0; nowsza niż 7.4

Funkcjonalność obsługiwana przez proxy:

Wersja proxy	Aktualizacja danych	Aktualizacja konfiguracji	Zadania
Bieżąca	Tak	Tak	Tak
Nieaktualna	Tak	Nie	Polecenia zdalne (np. skrypty powłoki); Natychmiastowe sprawdzenia wartości pozycji (tj. *Wykonaj teraz*); Uwaga: testy przetwarzania wstępnego z rzeczywistą wartością nie są obsługiwane.
Nieobsługiwana	Nie	Nie	Nie

Ostrzeżenia dotyczące używania niekompatybilnych wersji demonów Zabbix są zapisywane w logu.

Obsługiwane pliki XML

Pliki XML nie starsze niż wersja 1.8 są obsługiwane podczas importu w Zabbix 7.4.

Attention:
W formacie eksportu XML zależności wyzwalaczy są przechowywane wyłącznie według nazwy. Jeśli istnieje kilka wyzwalaczy o tej samej nazwie (na przykład o różnych poziomach ważności i wyrażeniach), między którymi zdefiniowano zależność, nie można ich zaimportować. Takie zależności należy ręcznie usunąć z pliku XML i dodać ponownie po imporcie.

11 Dynamiczna biblioteka łącza Zabbix sender dla Windows

Omówienie

W środowisku Windows aplikacje mogą wysyłać dane do serwera/proxy Zabbix, korzystając z dynamicznej biblioteki łącza Zabbix sender (zabbix_sender.dll), zamiast uruchamiać zewnętrzny proces (zabbix_sender.exe).

Pliki zabbix_sender.h i zabbix_sender.lib są wymagane do kompilowania aplikacji użytkownika z użyciem zabbix_sender.dll.

Pobieranie

Istnieją dwa sposoby uzyskania zabbix_sender.dll.

1. Pobierz pliki zabbix_sender.h, zabbix_sender.lib i zabbix_sender.dll jako archiwum ZIP.

Podczas wybierania opcji pobierania upewnij się, że w sekcji *Encryption* wybrano opcję "No encryption", a w sekcji *Packaging* opcję "Archive". Następnie pobierz Zabbix agent (nie Zabbix agent 2).

Pliki zabbix_sender.h, zabbix_sender.lib i zabbix_sender.dll będą znajdować się w pobranym archiwum ZIP w katalogu `bin\dev`. Rozpakuj pliki w wybranej lokalizacji.

2. Zbuduj zabbix_sender.dll z kodu źródłowego (zobacz instrukcje).

Dynamiczna biblioteka łącza wraz z plikami deweloperskimi będzie znajdować się w katalogu `bin\winXX\dev`. Aby z niej korzystać, dołącz plik nagłówkowy zabbix_sender.h i połącz aplikację z biblioteką zabbix_sender.lib.

Zobacz także

- przykład prostego narzędzia Zabbix sender zaimplementowanego z użyciem dynamicznej biblioteki łącza Zabbix sender, ilustrujący sposób użycia biblioteki;
- plik zabbix_sender.h zawierający funkcje interfejsu dynamicznej biblioteki łącza Zabbix sender. Plik ten zawiera dokumentację wyjaśniającą przeznaczenie każdej funkcji interfejsu, jej argumenty oraz wartość zwracaną.

12 Aktualizacja monitorowania usług

Overview In Zabbix 6.0, service monitoring functionality has been reworked significantly (see What's new in Zabbix 6.0.0 for the list of changes).

This page describes how services and SLAs defined in earlier Zabbix versions are changed during an upgrade to Zabbix 6.0 or newer.

Usługi W starszych wersjach Zabbix usługi miały dwa typy zależności: miękkie i twarde. Po aktualizacji wszystkie zależności staną się równorzędne.

Jeśli usługa „Usługa podrzędna" była wcześniej powiązana z „Usługa nadrzędna 1" za pomocą zależności twardej, a dodatkowo z „Usługa nadrzędna 2" za pomocą zależności miękkiej, po aktualizacji „Usługa podrzędna" będzie miała dwie usługi nadrzędne: „Usługa nadrzędna 1" i „Usługa nadrzędna 2".

Mapowanie problemów na usługi oparte na wyzwalaczach zostało zastąpione mapowaniem opartym na tagach. W Zabbix 6.0 i nowszych formularz konfiguracji usługi ma nowy parametr *Tagi problemów*, który umożliwia określenie jednej lub wielu par nazwy i wartości tagu do dopasowywania problemów. Wyzwalacze, które zostały powiązane z usługą, otrzymają nowy tag `ServiceLink : <trigger ID>:<trigger name>` (wartość tagu zostanie obcięta do 32 znaków). Powiązane usługi otrzymają tag problemu `ServiceLink` o tej samej wartości.

Zasady obliczania statusu

„Status calculation algorithm" zostanie zaktualizowany zgodnie z następującymi zasadami:

- Nie obliczaj → Ustaw status na OK
- Problem, jeśli co najmniej jedno dziecko ma problem → Najbardziej krytyczny spośród usług podrzędnych
- Problem, jeśli wszystkie dzieci mają problemy → Najbardziej krytyczny, jeśli wszystkie usługi podrzędne mają problemy

SLA Wcześniej cele SLA musiały być definiowane osobno dla każdej usługi. Od wersji Zabbix 6.0 SLA stało się odrębną encją, która zawiera informacje o harmonogramie usługi, oczekiwanym celu poziomu usługi (SLO) oraz okresach przestoju wyłączonych z obliczeń. Po skonfigurowaniu SLA można przypisać do wielu usług za pomocą tagów usługi.

Podczas aktualizacji:

- Identyczne SLA zdefiniowane dla każdej usługi zostaną zgrupowane i dla każdej grupy zostanie utworzone jedno SLA.
- Każda usługa, której to dotyczy, otrzyma specjalny tag `SLA:<ID>`, a ten sam tag zostanie określony w parametrze *Service tags* odpowiedniego SLA.
- Czas utworzenia usługi, nowa metryka w raportach SLA, zostanie ustawiony na 01/01/2000 00:00 dla istniejących usług.

13 Inne problemy

Logowanie i systemd

Zalecamy utworzenie użytkownika *zabbix* jako użytkownika systemowego, czyli bez możliwości logowania. Niektórzy użytkownicy ignorują to zalecenie i używają tego samego konta do logowania się (np. przez SSH) na host, na którym działa Zabbix. Może to spowodować awarię demona Zabbix podczas wylogowywania. W takim przypadku w logu serwera Zabbix zobaczysz coś podobnego do poniższego:

```
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
```

a w logu agenta Zabbix:

```
zabbix_agentd [27796]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

Dzieje się tak z powodu domyślnego ustawienia systemd `RemoveIPC=yes` skonfigurowanego w `/etc/systemd/logind.conf`. Po wylogowaniu się z systemu semafory utworzone wcześniej przez Zabbix są usuwane, co powoduje awarię.

Cytat z dokumentacji 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".
```

Istnieją 2 rozwiązania tego problemu:

1. (zalecane) Przestań używać konta *zabbix* do czegokolwiek innego niż procesy Zabbix, utwórz osobne konto do innych zadań.
2. (niezalecane) Ustaw `RemoveIPC=no` w `/etc/systemd/logind.conf` i uruchom ponownie system. Pamiętaj, że `RemoveIPC` jest parametrem obowiązującym w całym systemie, więc jego zmiana wpłynie na cały system.

Używanie frontend Zabbix za proxy

Jeśli frontend Zabbix działa za serwerem proxy, ścieżka cookie w pliku konfiguracyjnym proxy musi zostać przepisana tak, aby odpowiadała ścieżce reverse proxy. Zobacz przykłady poniżej. Jeśli ścieżka cookie nie zostanie przepisana, użytkownicy mogą napotkać problemy z autoryzacją podczas próby logowania do frontend Zabbix.

Przykładowa konfiguracja dla nginx

```
# ..
location / {
# ..
proxy_cookie_path /zabbix /;
proxy_pass http://192.168.0.94/zabbix/;
# ..
```

Przykładowa konfiguracja dla Apache

```
# ..
ProxyPass "/" http://host/zabbix/
ProxyPassReverse "/" http://host/zabbix/
ProxyPassReverseCookiePath /zabbix /
ProxyPassReverseCookieDomain host zabbix.example.com
# ..
```

14 Porównanie agent vs agent 2

Ta sekcja opisuje różnice między Zabbix agent i Zabbix agent 2.

Parameter	Zabbix agent	Zabbix agent 2
Język programowania	C	Go z niektórymi częściami w C
Demonizacja	tak	tylko przez systemd (tak w Windows)
Obsługiwane rozszerzenia	Niestandardowe moduły ładowalne w C.	Niestandardowe wtyczki w Go.
Wymagania		
Obsługiwane platformy	Linux, IBM AIX, FreeBSD, NetBSD, OpenBSD, HP-UX, Mac OS X, Solaris: 9, 10, 11, Windows: wszystkie wersje desktopowe i serwerowe od XP	Linux, Windows: wszystkie wersje desktopowe i serwerowe, na których można zainstalować obsługiwaną wersję Go.
Obsługiwane biblioteki kryptograficzne	GnuTLS 3.1.18 i nowsze OpenSSL 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x LibreSSL - testowano z wersjami 2.7.4, 2.8.2 (obowiązują pewne ograniczenia, zobacz stronę Encryption, aby uzyskać szczegóły).	Linux: OpenSSL 1.0.1 i nowszy. MS Windows: OpenSSL 1.1.1 lub nowszy. Biblioteka OpenSSL musi mieć włączoną obsługę PSK. LibreSSL nie jest obsługiwany.
Procesy monitorowania		
Procesy	Oddzielny proces aktywnego sprawdzania dla każdego wpisu serwer/proxy.	Jeden proces z automatycznie tworzonymi wątkami. Maksymalna liczba wątków jest określana przez zmienną środowiskową GOMAXPROCS.

Parameter	Zabbix agent	Zabbix agent 2
Metryki	**UNIX**: zobacz listę obsługiwanych pozycji. **Windows**: zobacz listę dodatkowych, specyficznych dla Windows pozycji.	**UNIX**: Wszystkie metryki obsługiwane przez Zabbix agent. Ponadto agent 2 zapewnia natywne rozwiązanie monitorowania Zabbix dla: Docker, Memcached, MySQL, PostgreSQL, Redis, systemd i innych celów monitorowania - zobacz pełną listę pozycji specyficznych dla agent 2. **Windows**: Wszystkie metryki obsługiwane przez Zabbix agent, a także sprawdzenia net.tcp.service* dla HTTPS, LDAP. Ponadto agent 2 zapewnia natywne rozwiązanie monitorowania Zabbix dla: PostgreSQL, Redis.
Współbieżność	Aktywne sprawdzenia dla pojedynczego serwera są wykonywane sekwencyjnie.	Sprawdzenia z różnych wtyczek lub wiele sprawdzeń w ramach jednej wtyczki mogą być wykonywane współbieżnie.
Trapy firm trzecich	nie	tak
Dodatkowe funkcje		
Trwała pamięć masowa	nie	tak
Trwałe pliki dla metryk log*[]	tak (tylko w Unix)	nie
Przesyłanie danych logów	Może być wykonywane podczas zbierania logów, aby zwolnić bufor.	Zbieranie logów jest zatrzymywane, gdy bufor jest pełny, dlatego parametr BufferSize musi być co najmniej równy MaxLinesPerSecond x 2.
Zmienia użytkownika w czasie działania	tak (tylko systemy podobne do Unix)	nie (kontrolowane przez systemd)
Konfigurowalne przez użytkownika zestawy szyfrów	tak	nie

Zobacz także:

- *Opis procesów Zabbix*: Zabbix agent, Zabbix agent 2
- *Parametry konfiguracji*: Zabbix agent UNIX / Windows, Zabbix agent 2 UNIX / Windows

15 Przykłady ucieczki znaków

Przegląd

Ta strona zawiera przykłady poprawnego stosowania znaków ucieczki podczas używania wyrażeń regularnych w różnych kontekstach.

> **Note:**
> Podczas korzystania z konstruktora wyrażeń wyzwalacza poprawne znaki ucieczki w wyrażeniach regularnych są dodawane automatycznie.

Przykłady

Makro użytkownika z kontekstem

Wyrażenie regularne: `\.+\"[a-z]+<br>` Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}`

Uwaga:

- ukośniki odwrotne nie są poprzedzane znakiem ucieczki;
- cudzysłowy są poprzedzane znakiem ucieczki.

Funkcja makra wewnątrz parametru klucza pozycji

Wyrażenie regularne: `.+:(\d+)$`
 Klucz pozycji: `net.tcp.service[tcp,,"{{$ENDPOINT}.regsub(\".+:(\d+)$\",\1)}"]`

Uwaga:

- wyrażenie regularne wewnątrz funkcji makra `regsub` jest ujęte w podwójny cudzysłów (ponieważ zawiera zamykający nawias);
- cudzysłowy wokół wyrażenia regularnego są poprzedzane znakiem ucieczki (ponieważ cały trzeci parametr pozycji jest ujęty w podwójny cudzysłów);
- trzeci parametr klucza pozycji jest ujęty w podwójny cudzysłów, ponieważ zawiera przecinek.

Funkcja makra LLD

Wyrażenie regularne: `\.+\"([a-z]+)`
 Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`

Uwaga:

- ukośniki odwrotne nie są poprzedzane znakiem ucieczki;
- cudzysłowy są poprzedzane znakiem ucieczki.

Funkcja makra LLD wewnątrz kontekstu makra użytkownika

Wyrażenie regularne: `\.+\"([a-z]+)`
 Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`
 Makro użytkownika z kontekstem: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"[a-z]+\", \1)}"}`

Uwaga:

- sposób poprzedzania znakiem ucieczki ukośników odwrotnych dla LLD nie zmienia się;
- przy wstawianiu makra LLD do kontekstu makra użytkownika należy umieścić je w ciągu znaków:

1. wokół wyrażenia makra dodawane są cudzysłowy;
2. cudzysłowy są poprzedzane znakiem ucieczki; łącznie wprowadzane są 3 nowe ukośniki odwrotne.

Parametr tekstowy funkcji (dowolnej)

Jako przykład użyto `concat`.

Zawartość ciągu: `\.+\"[a-z]+`
 Wyrażenie: `concat("abc", "\\.\\\"[a-z]+")`

Uwaga:

- parametry tekstowe wymagają poprzedzania znakiem ucieczki zarówno ukośników odwrotnych, jak i cudzysłowów.

Funkcja makra LLD wewnątrz parametru tekstowego funkcji

Wyrażenie regularne: `\.+\"([a-z]+)`
 Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`
 Wyrażenie: `concat("abc, "{{#MACRO}.iregsub(\"\\.+\\\\\"([a-z]+)\", \\1)}")`

Uwaga:

- parametry tekstowe wymagają poprzedzania znakiem ucieczki zarówno ukośników odwrotnych, jak i cudzysłowów;
- dodawana jest kolejna warstwa poprzedzania znakiem ucieczki, ponieważ makro zostanie rozwiązane dopiero po usunięciu cudzysłowów z ciągu;

Makro użytkownika z kontekstem wewnątrz parametru tekstowego funkcji

Wyrażenie regularne: `\.+\"[a-z]+`
 Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}`
 Wyrażenie: `concat("abc, "{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}")`

Uwaga:

- podobnie jak w poprzednim przykładzie potrzebna jest dodatkowa warstwa poprzedzania znakiem ucieczki;
- ukośniki odwrotne i cudzysłowy są poprzedzane znakiem ucieczki tylko dla najwyższego poziomu poprzedzania znakiem ucieczki (ponieważ jest to parametr tekstowy).

Funkcja makra LLD wewnątrz kontekstu makra użytkownika wewnątrz funkcji

Wyrażenie regularne: `\.+\"([a-z])+`
 Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`
 Makro użytkownika z kontekstem: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"([a-z]+)\", \1)}"}`
 Wyrażenie: `concat("abc, "{$MACRO:\"{{#MACRO}.iregsub(\\\"\.+\\\\\\\"([a-z]+)\\\", \\1)}\"}")`

Zwróć uwagę na trzy warstwy poprzedzania znakiem ucieczki:

1. Dla funkcji makra LLD, bez poprzedzania znakiem ucieczki ukośników odwrotnych;
2. Dla makra użytkownika z kontekstem, bez poprzedzania znakiem ucieczki ukośników odwrotnych;
3. Dla parametru tekstowego funkcji, z poprzedzaniem znakiem ucieczki ukośników odwrotnych.

Makro użytkownika z kontekstem bezpośrednio wewnątrz ciągu

Wyrażenie regularne: `\.+\"[a-z]+<br>` Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Wewnątrz ciągu w pewnym wyrażeniu, na przykład: `func(arg1, arg2, arg3)="{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}"`

Uwaga:

- ciągi również wymagają poprzedzania znakiem ucieczki ukośników odwrotnych;
- ciągi również wymagają poprzedzania znakiem ucieczki cudzysłowów;
- ponownie jest to przypadek z 2 poziomami poprzedzania znakiem ucieczki:

1. Poprzedzanie znakiem ucieczki dla kontekstu makra użytkownika bez poprzedzania znakiem ucieczki ukośników odwrotnych;
2. Poprzedzanie znakiem ucieczki wynikające z tego, że jest to ciąg, z poprzedzaniem znakiem ucieczki ukośników odwrotnych.

23 Szybkie przewodniki referencyjne

Omówienie

Ta sekcja dokumentacji zawiera krótkie instrukcje dotyczące konfiguracji Zabbixa pod kątem niektórych często wymaganych celów monitorowania.

Została zaprojektowana z myślą o nowych użytkownikach Zabbixa i może służyć jako przewodnik po innych sekcjach dokumentacji, które zawierają informacje potrzebne do rozwiązania danego zadania.

Dostępne są następujące krótkie przewodniki:

- Monitorowanie systemu Linux za pomocą agent Zabbix
- Monitorowanie systemu Windows za pomocą agent Zabbix
- Monitorowanie Apache przez HTTP
- Monitorowanie MySQL za pomocą agent Zabbix 2
- Monitorowanie VMware za pomocą Zabbix
- Monitorowanie ruchu sieciowego za pomocą Zabbix
- Monitorowanie ruchu sieciowego za pomocą Zabbix z użyciem sprawdzeń aktywnych
- Monitorowanie witryn internetowych za pomocą pozycji przeglądarki
- Monitorowanie certyfikatów witryn internetowych za pomocą agent Zabbix 2 (pasywne)
- Monitorowanie przełącznika sieciowego lub routera za pomocą Zabbix
- Monitorowanie dziennika zdarzeń systemu Windows z użyciem sprawdzeń aktywnych

1. Monitor Linux with Zabbix agent

Wprowadzenie Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania maszyn z systemem Linux za pomocą Zabbix.
Kroki opisane w tym samouczku można zastosować do dowolnego systemu operacyjnego opartego na Linuksie.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania Twojej maszyny z systemem Linux. Jeśli szukasz opcji zaawansowanego dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed przystąpieniem do tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

Instalacja Zabbix agent Zabbix agent to proces odpowiedzialny za zbieranie danych.

Sprawdź wersję swojego serwera Zabbix:

```
zabbix_server -V
```

Zainstaluj Zabbix agent w tej samej wersji (zalecane) na komputerze z systemem Linux, który chcesz monitorować.
W zależności od potrzeb monitorowania może to być ten sam komputer, na którym zainstalowano serwer Zabbix, albo zupełnie inny komputer.

Wybierz najbardziej odpowiednią metodę instalacji:

- Uruchom jako kontener Docker - zobacz listę dostępnych obrazów w repozytorium Zabbix Docker.
- Zainstaluj z pakietów Zabbix packages (dostępne dla Alma Linux, CentOS, Debian, Oracle Linux, Raspberry Pi OS, RHEL, Rocky Linux, SUSE Linux Enterprise Server, Ubuntu).
- Skompiluj ze źródeł.

Skonfiguruj Zabbix do monitorowania Zabbix agent może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie).

> **Note:**
> Sprawdzenie pasywne to proste żądanie danych. Zabbix serwer lub proxy prosi o określone dane (na przykład obciążenie CPU), a Zabbix agent odsyła wynik do serwera. Sprawdzenia aktywne wymagają bardziej złożonego przetwarzania. Agent musi najpierw pobrać z serwera(ów) listę pozycji do niezależnego przetwarzania, a następnie zbiorczo odesłać dane. Więcej informacji znajduje się w Pasywne i aktywne sprawdzenia agenta.

Szablony monitorowania dostarczane przez Zabbix zwykle oferują dwie alternatywy — szablon dla Zabbix agent oraz szablon dla Zabbix agent (active). W pierwszym przypadku agent będzie zbierać metryki w trybie pasywnym. Takie szablony zapewniają identyczne wyniki monitorowania, ale z użyciem różnych protokołów komunikacyjnych.

Dalsza konfiguracja Zabbix zależy od tego, czy wybierzesz szablon dla aktywnych czy pasywnych sprawdzeń Zabbix agent.

Kontrole pasywne Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której agent jest zainstalowany.

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalnie port) swojego serwera Zabbix do parametru server. Na przykład:

```
Server=192.0.2.1:10051
```

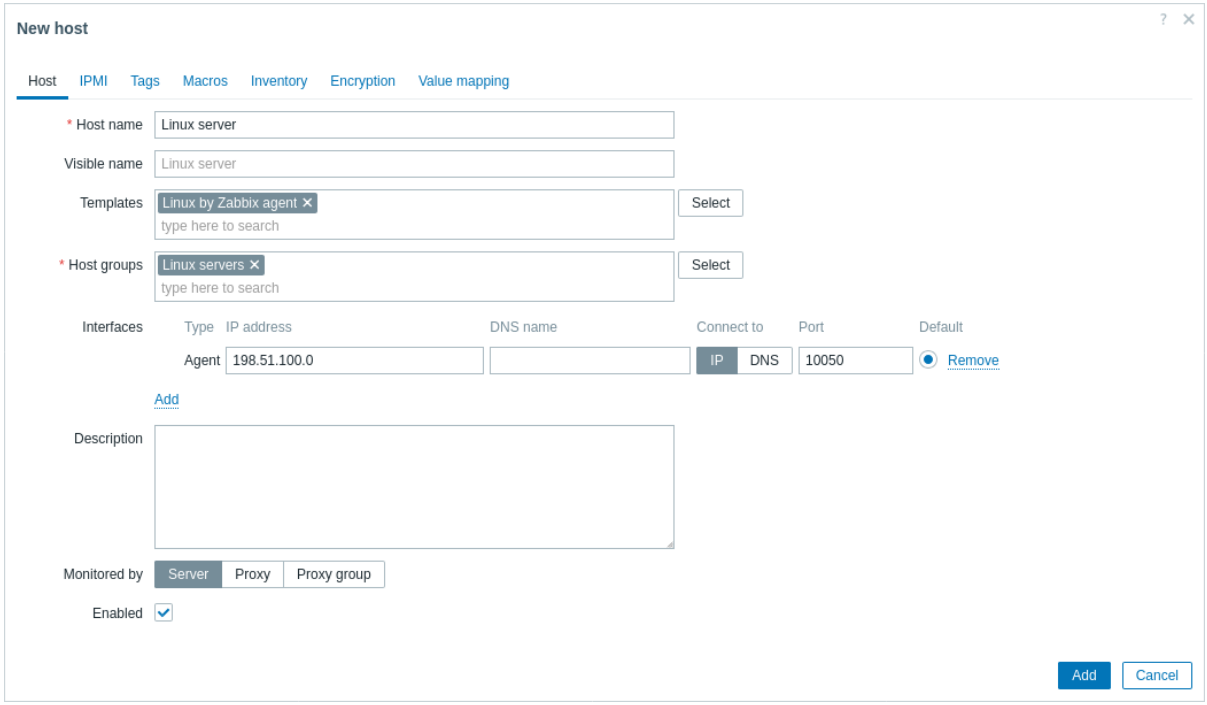
Agent Zabbix będzie używać tego adresu do akceptowania połączeń przychodzących tylko od określonych serwerów Zabbix lub proxy w celu pobierania danych.

3. Uruchom ponownie agenta Zabbix.

```
systemctl restart zabbix-agent
```

Zabbix frontend

1. Zaloguj się do Zabbix frontend.
2. Utwórz host w interfejsie webowym Zabbix.
   - W polu *Host name* wprowadź nazwę hosta (np. „Linux server”).
   - W polu *Templates* wpisz lub wybierz szablon „Linux by Zabbix agent”, który zostanie podłączony do hosta.
   - W polu *Host groups* wpisz lub wybierz grupę hostów (np. „Linux servers”).
   - W parametrze *Interfaces* dodaj interfejs *Agent* i określ adres IP lub nazwę DNS maszyny z systemem Linux, na której zainstalowano agent.



3. Kliknij *Add*, aby dodać hosta. Ten host będzie reprezentować monitorowaną maszynę z systemem Linux.

Kontrole aktywne Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której jest zainstalowany agent.

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalny port) swojego serwera Zabbix do parametru ServerActive. Na przykład:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent użyje tego adresu, aby połączyć się z portem trapper serwera Zabbix (domyślnie: 10051) i pobrać dane konfiguracyjne dla aktywnych kontroli.

3. Zdefiniuj parametr Hostname, który musi odpowiadać nazwie hosta zdefiniowanej w frontend Zabbix. W naszym przykładzie jest to:

```
Hostname=Linux server
```

Wartość Hostname musi się zgadzać, ponieważ w przypadku aktywnych kontroli Zabbix agent używa jej do pobrania poprawnej konfiguracji hosta z serwera. Dokładniej, agent inicjuje połączenie z serwerem i identyfikuje się za pomocą wartości Hostname. Następnie serwer udostępnia konfigurację monitorowania dla tego hosta. Jeśli te wartości będą różne, agent nie otrzyma odpowiedniej konfiguracji, co spowoduje brak metryk lub problemy z monitorowaniem.

4. Uruchom ponownie Zabbix agent.

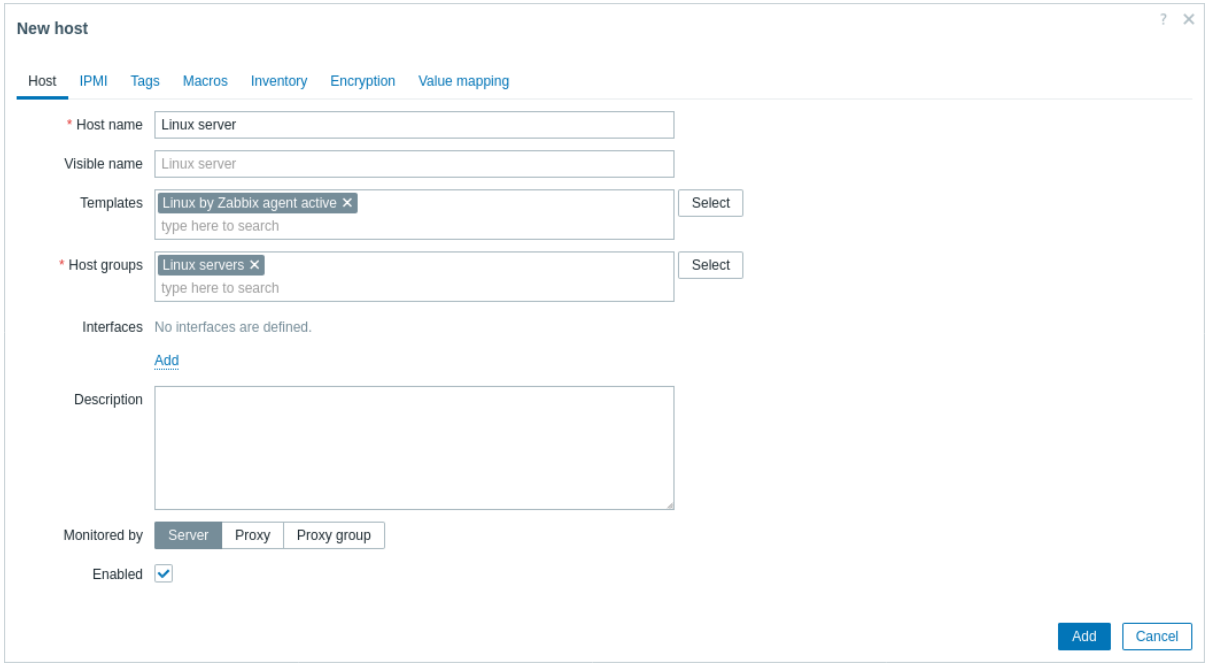
```
systemctl restart zabbix-agent
```

Zabbix frontend

1. Zaloguj się do Zabbix frontend.

2. Utwórz hosta w interfejsie webowym Zabbix.

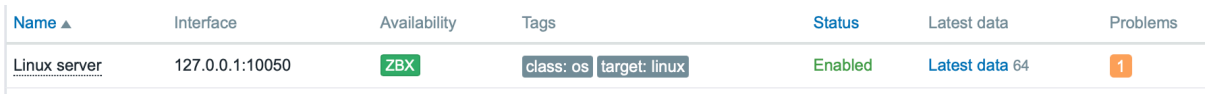
- W polu *Host name* wprowadź nazwę hosta (np. „Linux server”), która odpowiada wartości parametru *Hostname* zdefiniowanej wcześniej w pliku konfiguracyjnym agenta.
- W polu *Templates* wpisz lub wybierz szablon „Linux by Zabbix agent active”, który zostanie podłączony do hosta.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. „Linux servers”).
- Pozostaw *Interfaces* niezdefiniowane; interfejs nie jest wymagany dla aktywnych testów, ponieważ agent inicjuje połączenie z serwerem, zamiast nasłuchiwać połączenia z serwera.



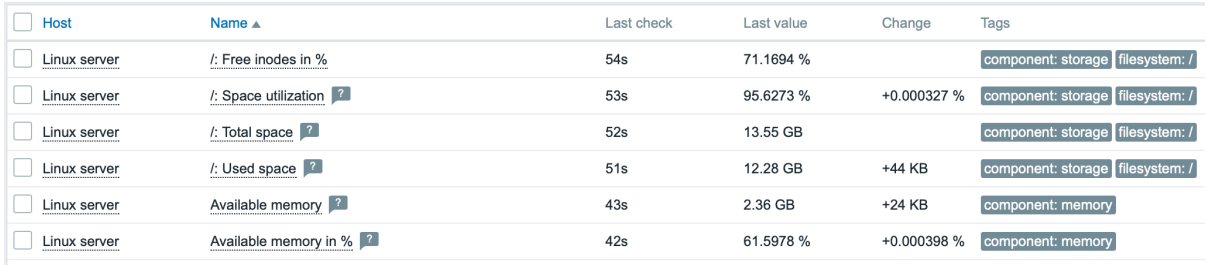
3. Kliknij *Add*, aby dodać hosta. Ten host będzie reprezentować monitorowaną maszynę z systemem Linux.

Wyświetlanie zebranych metryk Gratulacje! Na tym etapie Zabbix już monitoruje Twoją maszynę z systemem Linux.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring->Hosts* menu section i kliknij *Latest data* obok hosta.

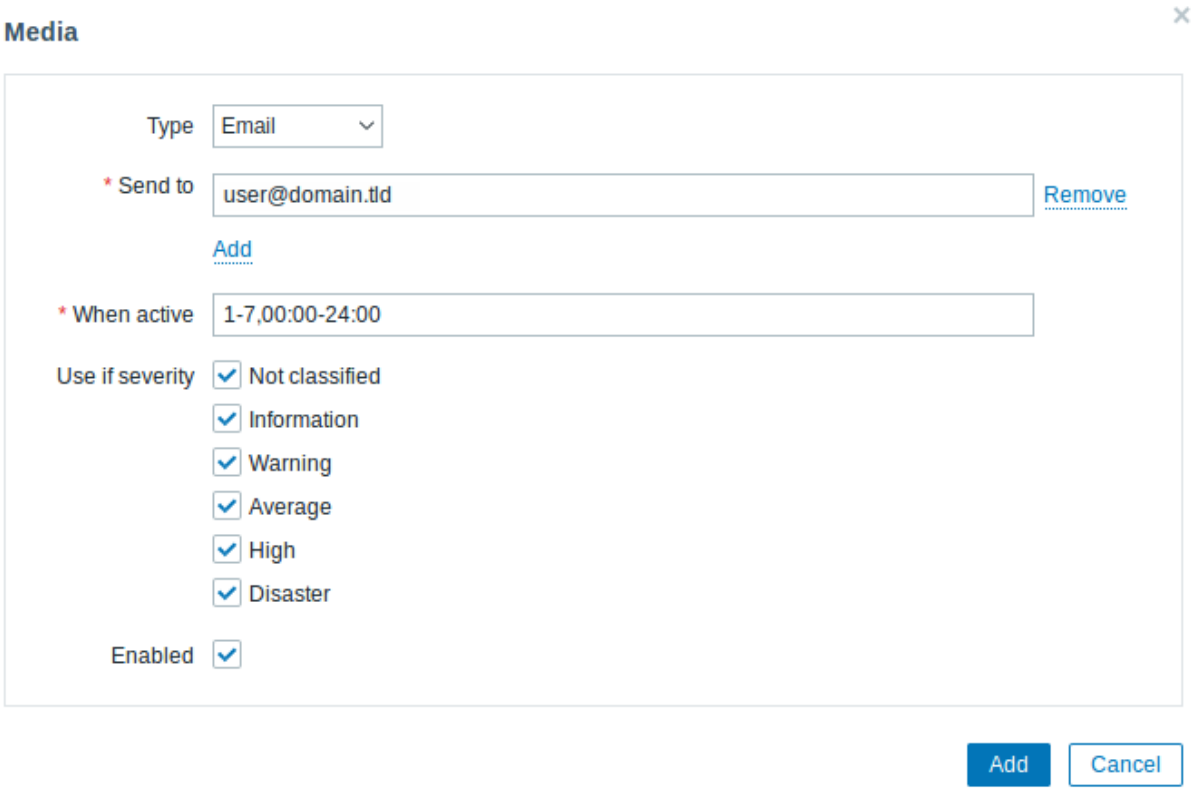


Ta czynność otworzy listę wszystkich najnowszych metryk zebranych z hosta serwera Linux.



Skonfiguruj alerty o problemach Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *User settings -> Profile*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami w przewodniku Receiving problem notification.

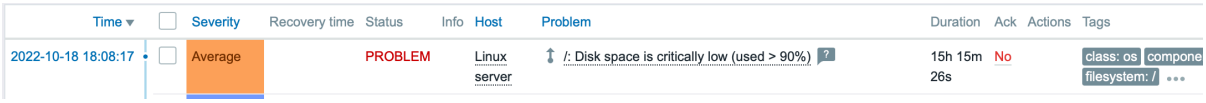
Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację W systemie Linux możesz zasymulować wysokie obciążenie CPU i w rezultacie otrzymać alert o problemie, uruchamiając:

```
cat /dev/urandom | md5sum
```

Może być konieczne uruchomienie kilku procesów md5sum, aby obciążenie CPU przekroczyło próg.

Gdy Zabbix wykryje problem, pojawi się on w sekcji Monitoring->Problems.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

Zobacz także:

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk (monitorowanie niestandardowe bez szablonów).
- Pozycje agenta Zabbix, Pozycje agenta Zabbix dla Windows - pełna lista metryk, które można monitorować za pomocą agenta Zabbix w systemie Windows.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Instalacja z pakietów - jak instalować komponenty Zabbix przy użyciu oficjalnych pakietów RPM i DEB dla różnych dystrybucji systemu Linux, zapewniając dostęp do najnowszych funkcji i poprawek błędów.

2 Monitorowanie systemu Windows za pomocą agent Zabbix

Wprowadzenie Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania maszyn z systemem Windows za pomocą Zabbix.

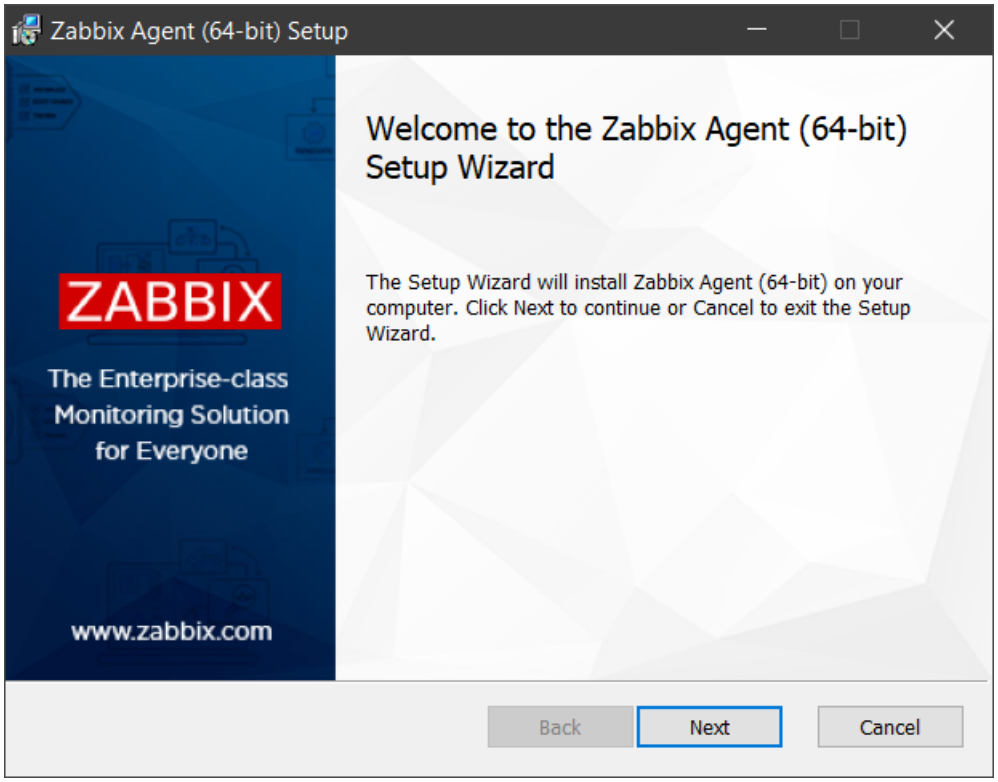
Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania maszyny z systemem Windows. Jeśli szukasz opcji zaawansowanego dostosowania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Configuration w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika instalacji musisz pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla swojego systemu operacyjnego.

Instalacja Zabbix agent Zabbix agent to proces odpowiedzialny za zbieranie danych. Należy go zainstalować na komputerze z systemem Windows, który chcesz monitorować. Postępuj zgodnie z instrukcjami instalacji Zabbix agent dla Windows.



Konfiguracja Zabbix do monitorowania Zabbix agent może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie).

Note:
Sprawdzenie pasywne jest prostym żądaniem danych. Serwer Zabbix lub proxy prosi o określone dane (na przykład obciążenie CPU), a Zabbix agent odsyła wynik do serwera. Sprawdzenia aktywne wymagają bardziej złożonego przetwarzania. Agent musi najpierw pobrać z serwera(ów) listę pozycji do niezależnego przetwarzania, a następnie zbiorczo odesłać dane. Więcej informacji można znaleźć w Pasywne i aktywne sprawdzenia agenta.

Szablony monitorowania dostarczane przez Zabbix zwykle oferują dwie alternatywy — szablon dla Zabbix agent oraz szablon dla Zabbix agent (active). W pierwszym przypadku agent będzie zbierać metryki w trybie pasywnym. Takie szablony zapewniają identyczne wyniki monitorowania, ale z użyciem różnych protokołów komunikacyjnych.

Dalsza konfiguracja Zabbix zależy od tego, czy wybierzesz szablon dla aktywnych czy pasywnych sprawdzeń Zabbix agent.

Pasywne kontrole Zabbix frontend

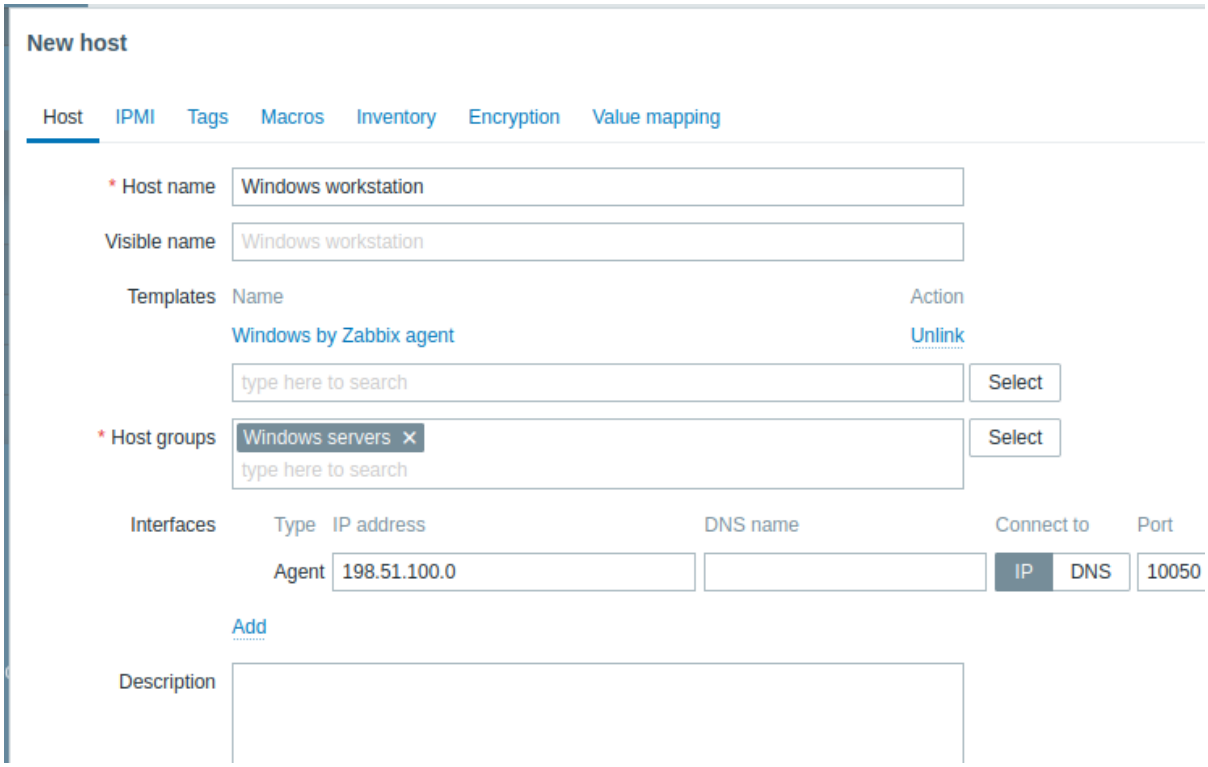
1. Zaloguj się do Zabbix frontend.

2. Utwórz hosta w interfejsie webowym Zabbix.

Ten host będzie reprezentował Twoją maszynę z systemem Windows.

3. W parametrze *Interfaces* dodaj interfejs *Agent* i określ adres IP lub nazwę DNS maszyny z systemem Windows, na której zainstalowano agent.

4. W parametrze *Templates* wpisz lub wybierz *Windows by Zabbix agent*.



Zabbix agent

W przypadku sprawdzeń pasywnych agent Zabbix musi znać adres IP lub nazwę DNS serwera Zabbix. Jeśli podczas instalacji agent podałeś poprawne informacje, plik konfiguracyjny jest już zaktualizowany. W przeciwnym razie musisz określić je ręcznie. Przejdź do folderu `C:\Program files\Zabbix Agent`, otwórz plik *zabbix_agentd.conf* i dodaj IP/DNS swojego serwera Zabbix do parametru *Server*.

Przykład:

```
Server=192.0.2.22
```

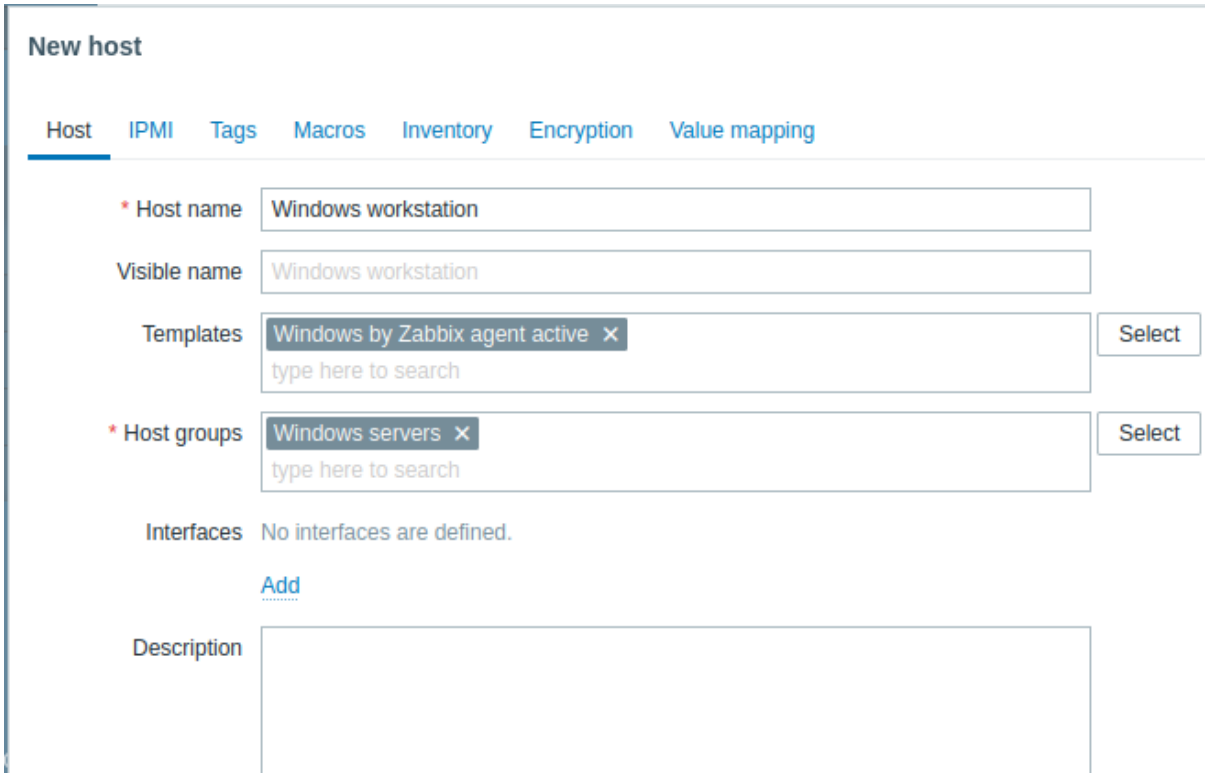
Aktywne kontrole Frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix.

Ten host będzie reprezentował Twój komputer z systemem Windows.

3. W parametrze *Templates* wpisz lub wybierz *Windows by Zabbix agent active*.



Zabbix agent

W folderze `C:\Program files\Zabbix Agent` otwórz plik *zabbix_agentd.conf* i dodaj:

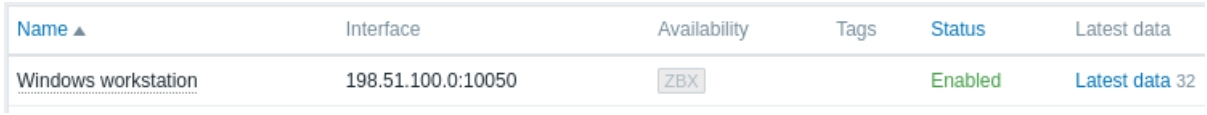
- Nazwę hosta utworzonego w interfejsie webowym Zabbix do parametru *Hostname*.
- Adres IP lub nazwę DNS maszyny serwera Zabbix do parametru *ServerActive* (może być już wstępnie uzupełniony, jeśli został podany podczas konfiguracji Zabbix agent).

Przykład:

```
ServerActive= 192.0.2.22
Hostname=Windows workstation
```

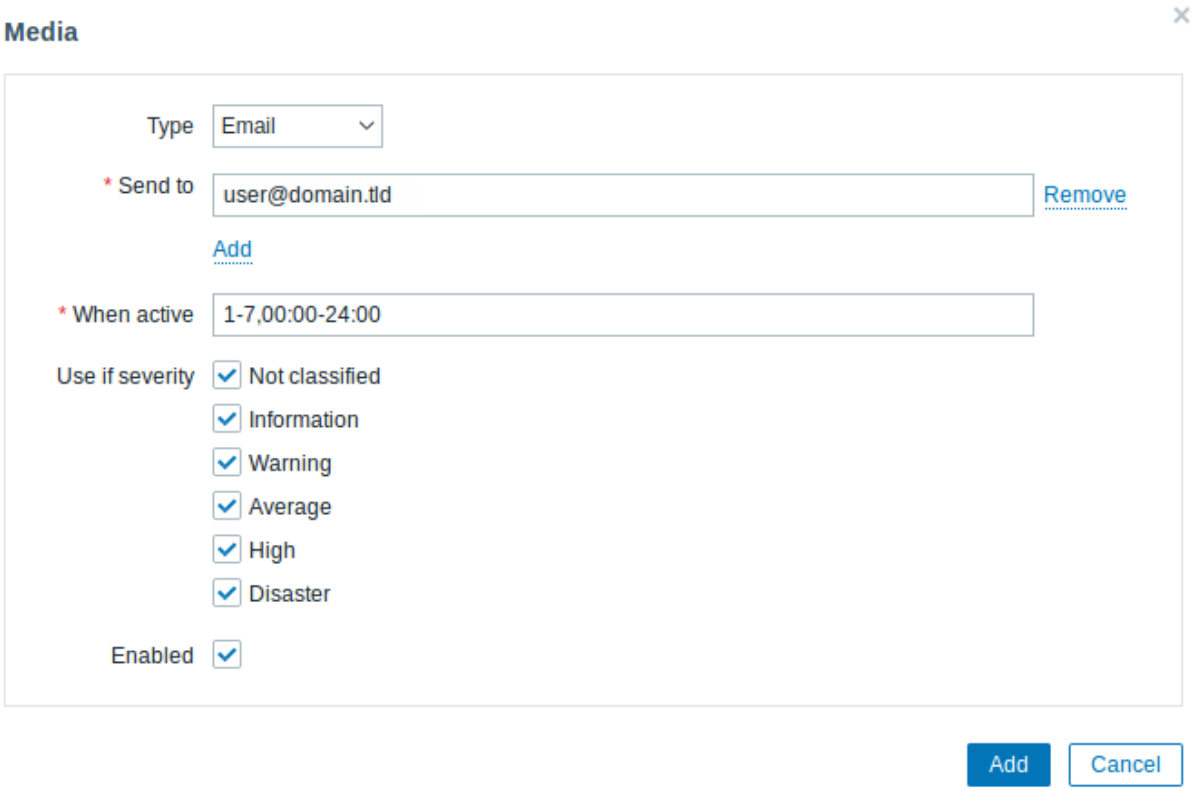
Wyświetlanie zebranych metryk Gratulacje! Na tym etapie Zabbix już monitoruje Twoją maszynę z systemem Windows.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring->Hosts* menu section i kliknij *Latest data* obok hosta.



Skonfiguruj alerty o problemach Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *User settings -> Profile*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami w przewodniku Receiving problem notification.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Note:
W systemie Windows możesz użyć narzędzia CpuStres, aby zasymulować duże obciążenie CPU i w rezultacie otrzymać alert o problemie.

Zobacz także:

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk (monitorowanie niestandardowe bez szablonów).
- Pozycje agenta Zabbix, Pozycje agenta Zabbix dla Windows - pełna lista metryk, które można monitorować za pomocą agenta Zabbix w systemie Windows.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).

3. Monitor Apache via HTTP

Wprowadzenie Ta strona pokazuje szybki i prosty sposób na rozpoczęcie monitorowania serwera WWW Apache bez instalowania żadnego dodatkowego oprogramowania.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania instalacji Apache. Jeśli szukasz opcji zaawansowanego dostosowania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Configuration podręcznika Zabbix.

Wymagania wstępne

Przed przystąpieniem do korzystania z tego przewodnika instalacji należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

Przygotowanie Apache 1. Sprawdź, której wersji Apache używasz:

W systemie opartym na RHEL uruchom:

```
httpd -v
```

W Debianie/Ubuntu uruchom:

```
apache2 -v
```

2. Upewnij się, że moduł Status jest włączony w Twojej instancji Apache.

W systemie opartym na RHEL uruchom:

```
httpd -M | grep status
status_module (shared)
```

W Debianie/Ubuntu uruchom:

```
apache2ctl -M | grep status
status_module (shared)
```

Jeśli nie widzisz status_module na liście, włącz moduł, uruchamiając:

W systemie opartym na RHEL uruchom:

```
LoadModule status_module /usr/lib/apache2/modules/mod_status.so
```

W Debianie/Ubuntu uruchom:

```
sudo /usr/sbin/a2enmod status
```

3. Edytuj plik konfiguracyjny Apache, aby zezwolić na dostęp do raportów statusu z adresu IP serwera Zabbix.

W systemie opartym na RHEL: `/etc/httpd/conf.modules.d/status.conf`:

```
sudo vi /etc/httpd/conf.modules.d/status.conf
```

W Debianie/Ubuntu: `/etc/apache2/mods-enabled/status.conf`:

```
sudo vi /etc/apache2/mods-enabled/status.conf
```

Dodaj następujące linie do pliku (**zastąp 198.51.100.255** adresem IP swojego serwera Zabbix):

- Dla Apache 2.2:
  <Location /server-status> SetHandler server-status

  Order Deny,Allow Deny from all Allow from 198.51.100.255 </Location>
- Dla Apache 2.4:
  <Location ”/server-status”> SetHandler server-status Require ip 198.51.100.255 </Location>

4. Uruchom ponownie Apache

W systemie opartym na RHEL uruchom:

```
sudo systemctl restart httpd
```

W Debianie/Ubuntu uruchom:

```
sudo systemctl restart apache2
```

5. Aby sprawdzić, czy wszystko jest skonfigurowane poprawnie, uruchom (**zastąp 198.51.100.255** adresem IP swojego serwera Zabbix):

```
curl 198.51.100.255/server-status
```

Odpowiedź powinna zawierać statystyki serwera WWW Apache.

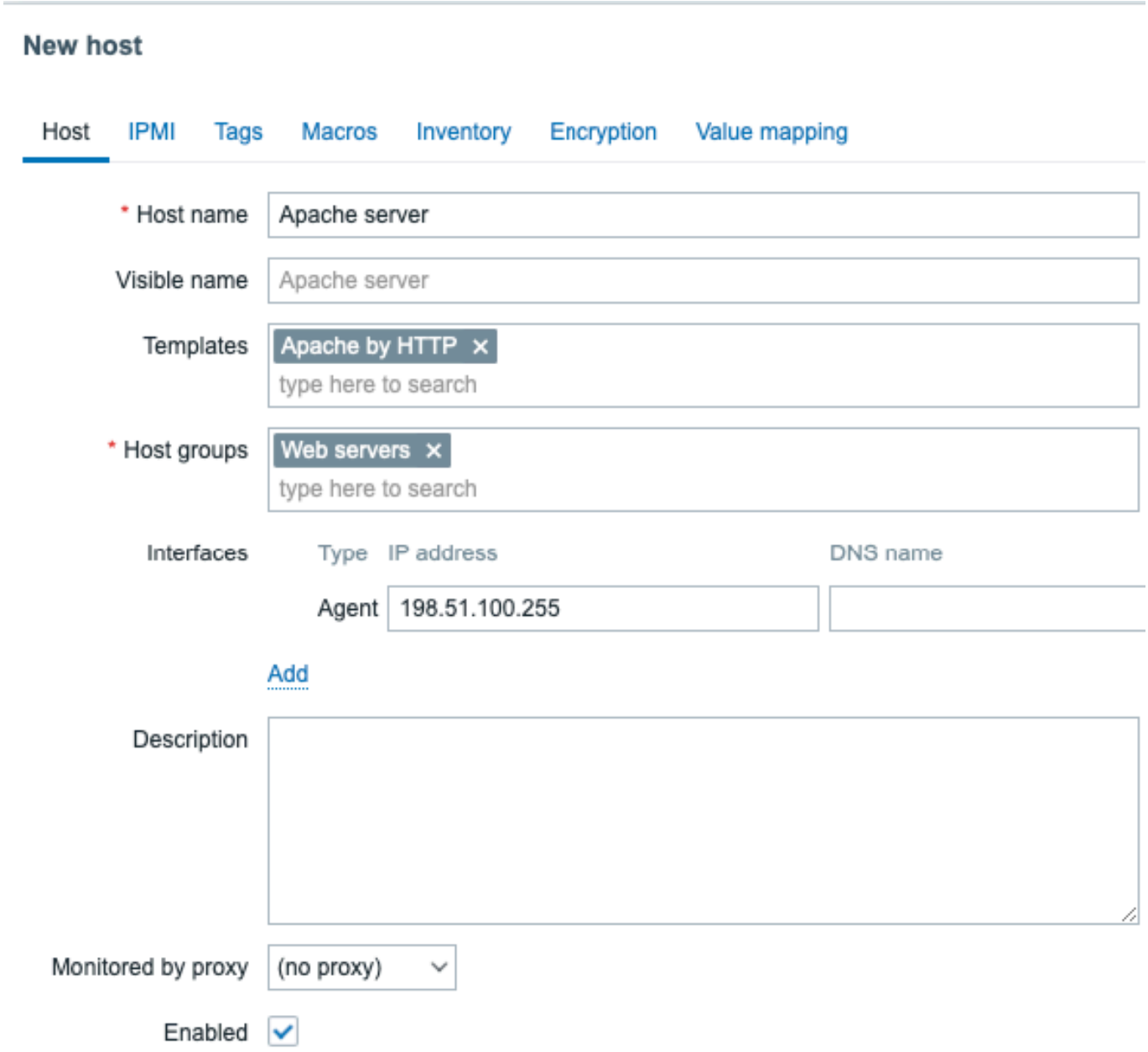
Skonfiguruj Zabbix do monitorowania 1. Zaloguj się do frontend Zabbix.

2. Utwórz hosta w interfejsie webowym Zabbix.

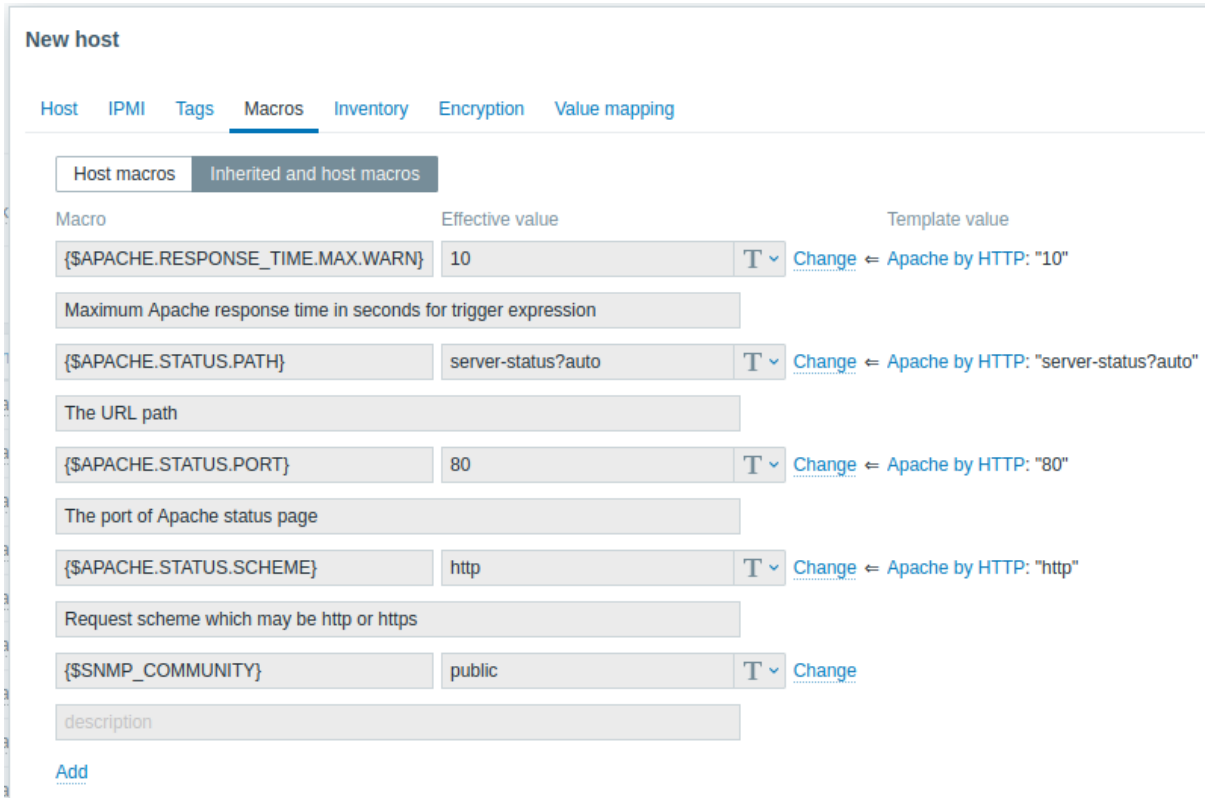
Ten host będzie reprezentował Twój serwer Apache.

3. W parametrze *Interfaces* dodaj interfejs *Agent* i podaj adres IP swojej instancji Apache. **Nie musisz instalować Zabbix agent na tej maszynie**, interfejs będzie używany wyłącznie do rozwiązywania makra {HOST.CONN}. To makro jest używane w pozycjach szablonu do lokalizowania instancji Apache.

4. W parametrze *Templates* wpisz lub wybierz *Apache by HTTP*.



5. Przejdź do zakładki **Macros** i wybierz tryb *Inherited and host macros*. Sprawdź, czy wartości makr {$APACHE.STATUS.PORT} i {$APACHE.STATUS.SCHEME} odpowiadają ustawieniom Twojej instalacji. Domyślnie port to 80, a schemat to http. Zmień wartości makr, jeśli używasz innego portu i/lub schematu.



Wyświetlanie zebranych metryk Gratulacje! Na tym etapie Zabbix monitoruje już Twój serwer WWW Apache.

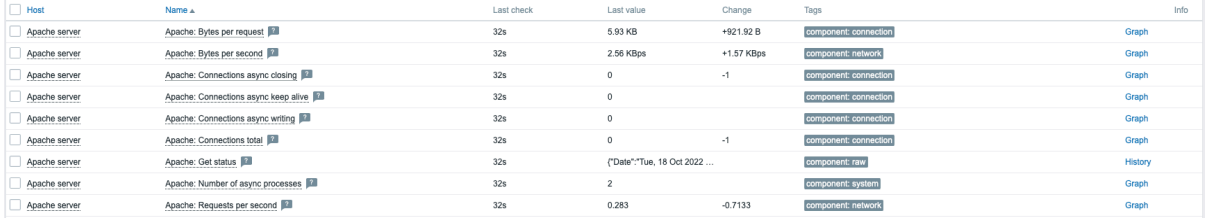
Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring->Hosts* menu section i kliknij *Dashboards* obok hosta.



Ta czynność przeniesie Cię do pulpitu hosta z najważniejszymi metrykami zebranymi ze strony Apache /server-status.

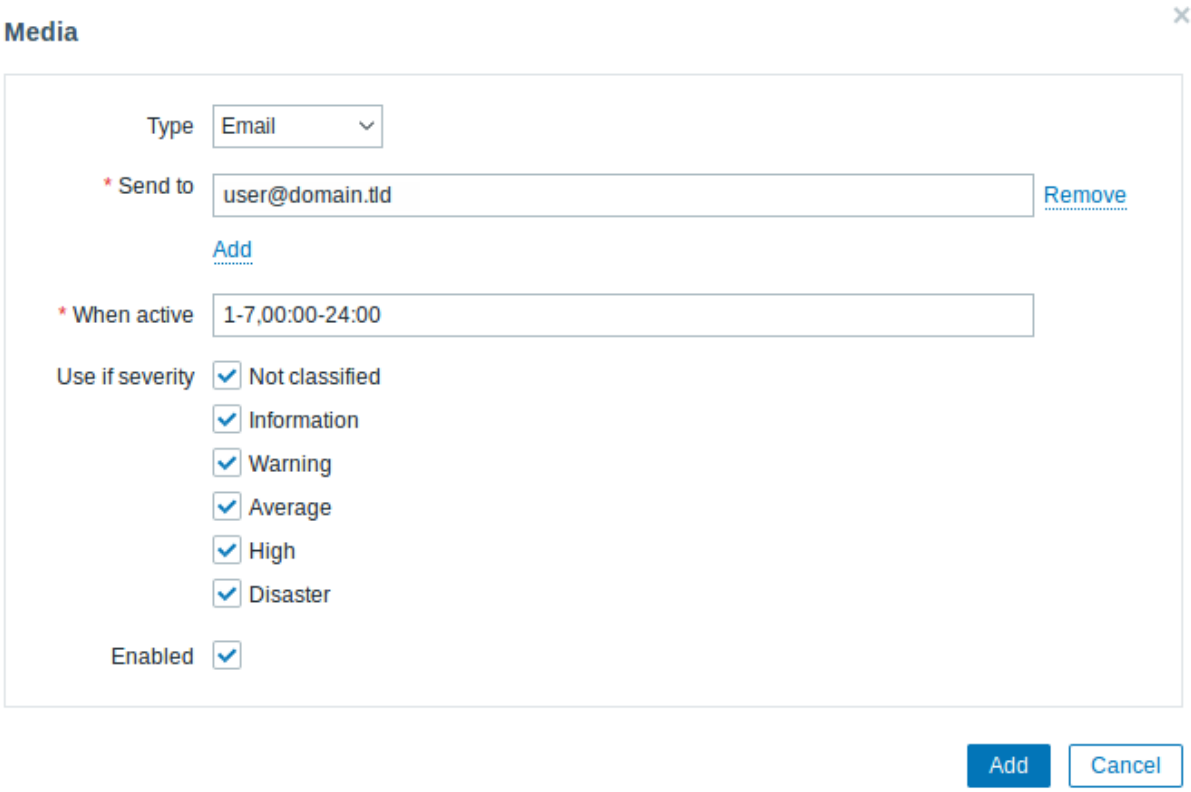


Alternatywnie, w sekcji *Monitoring->Hosts* możesz kliknąć *Latest data*, aby wyświetlić wszystkie ostatnio zebrane metryki na liście.



Skonfiguruj alerty o problemach Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika -> Profil*, przejdź do karty *Media* i dodaj swój adres e-mail.

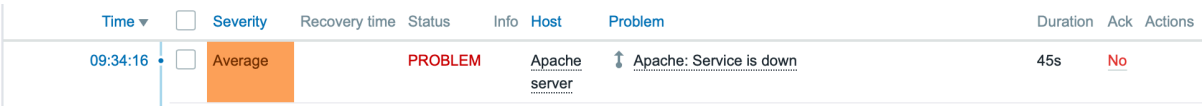


2. Postępuj zgodnie z przewodnikiem dotyczącym odbierania powiadomień o problemach.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację Aby zasymulować rzeczywisty problem i otrzymać testowy alert o problemie:

1. Otwórz konfigurację hosta *Apache server* w Zabbix.
2. Przejdź do zakładki Makra i wybierz *Inherited and host macros*.
3. Naciśnij *Change* obok makra {$APACHE.STATUS.PORT} i ustaw inny port.
4. Naciśnij Update, aby zapisać konfigurację hosta.
5. Za kilka minut Zabbix wykryje problem *Apache service is down*, ponieważ nie będzie już mógł połączyć się z instancją. Pojawi się on w sekcji Monitoring->Problems.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem, aby rozwiązać problem i kontynuować monitorowanie Apache.

Zobacz także:

- Wzmacnianie zabezpieczeń serwera WWW - zalecane ustawienia zwiększające bezpieczeństwo serwera WWW.
- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Pozycje HTTP - jak monitorować niestandardowe metryki przy użyciu HTTP agent.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).

4. Monitor MySQL with Zabbix agent 2

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania serwera MySQL.

Aby monitorować serwer MySQL, dostępnych jest kilka podejść: Zabbix agent, Zabbix agent 2 lub standard Open Database Connectivity (ODBC). Głównym celem tego przewodnika jest monitorowanie serwera MySQL za pomocą Zabbix agent 2, co jest podejściem **zalecanym** ze względu na jego bezproblemową konfigurację w różnych środowiskach. Jednak ta strona zawiera również instrukcje dotyczące innych podejść, więc możesz wybrać to, które najlepiej odpowiada Twoim wymaganiom.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania serwera MySQL. Jeśli szukasz opcji zaawansowanego dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed przystąpieniem do tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix 2 zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki opisane w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Utwórz użytkownika MySQL

Aby monitorować serwer MySQL, Zabbix wymaga dostępu do niego oraz do jego procesów. Twoja instalacja MySQL ma już użytkownika z wymaganym poziomem dostępu (użytkownika „zabbix", który został utworzony podczas instalacji Zabbix), jednak ten użytkownik ma więcej uprawnień, niż jest to konieczne do prostego monitorowania (uprawnienia do DROP baz danych, DELETE wpisów z tabel itp.). Dlatego należy utworzyć użytkownika MySQL przeznaczonego *wyłącznie* do monitorowania serwera MySQL.

1. Połącz się z klientem MySQL, utwórz użytkownika „zbx_monitor" (zastąp <*password*> dla użytkownika „zbx_monitor" wybranym przez siebie hasłem) i przyznaj użytkownikowi niezbędne uprawnienia za pomocą GRANT:

```
mysql -u root -p
# Enter password:

mysql> CREATE USER 'zbx_monitor'@'%' IDENTIFIED BY '<password>';
mysql> GRANT REPLICATION CLIENT,PROCESS,SHOW DATABASES,SHOW VIEW ON *.* TO 'zbx_monitor'@'%';
mysql> quit;
```

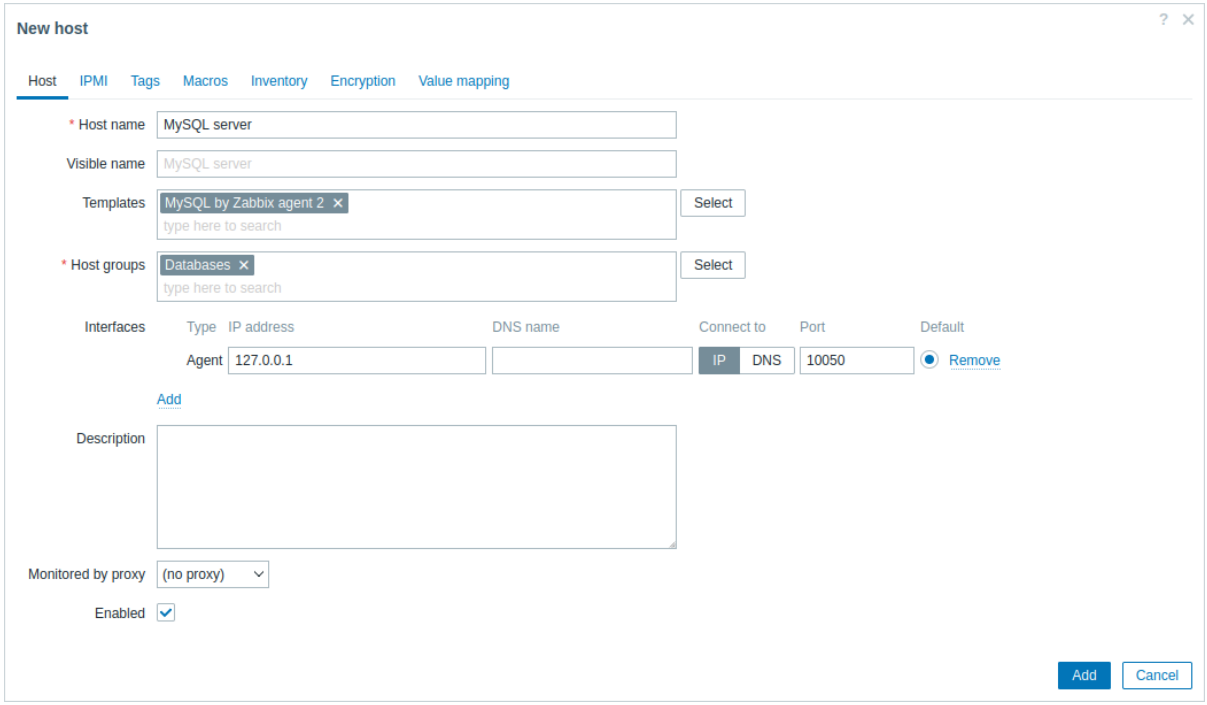
Po utworzeniu użytkownika możesz przejść do następnego kroku.

Skonfiguruj frontend Zabbix

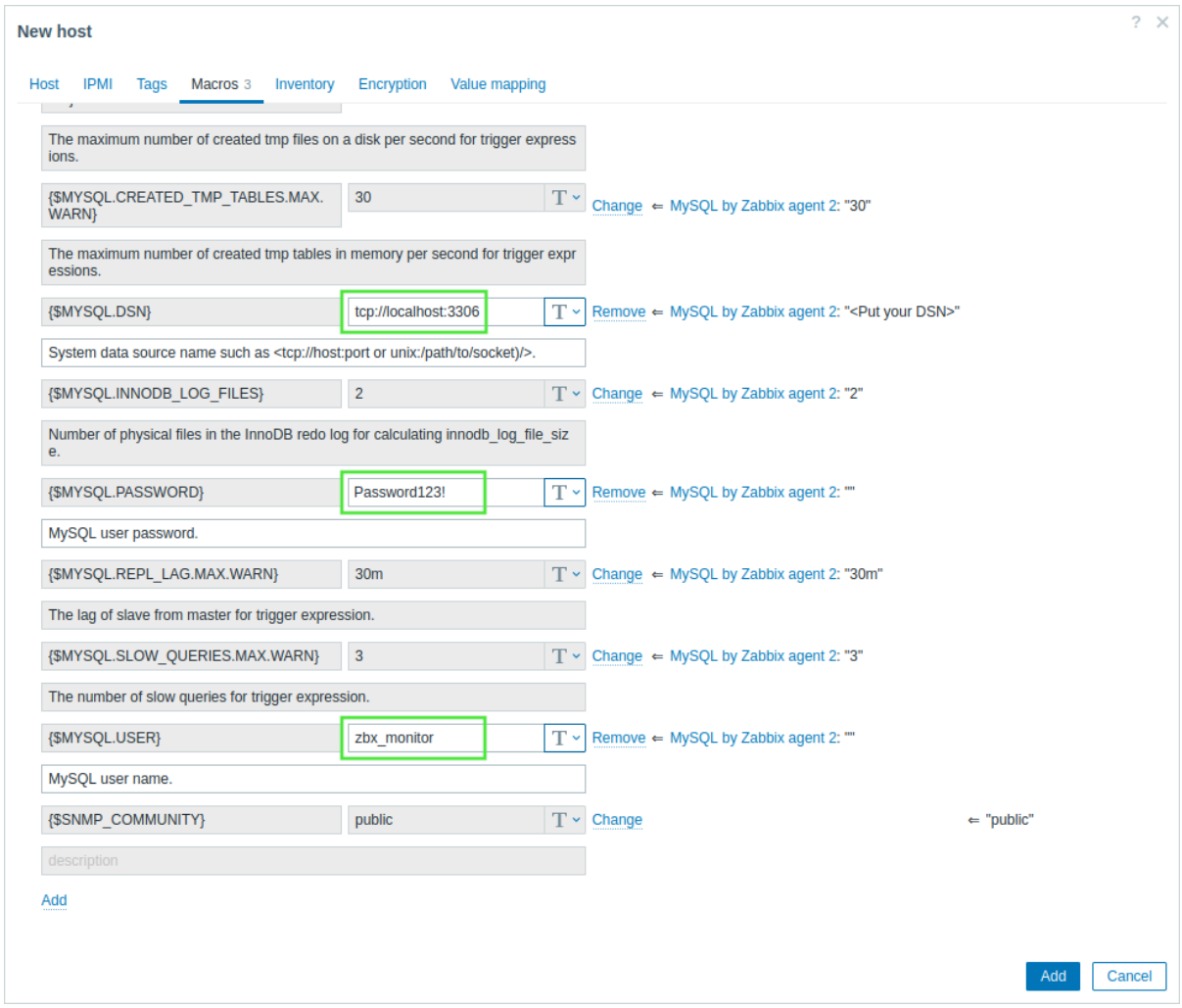
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie web Zabbix:

- W polu *Host name* wpisz nazwę hosta (np. "MySQL server").
- W polu *Templates* wpisz lub wybierz szablon "MySQL by Zabbix agent 2", który zostanie powiązany z hostem.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. "Databases").
- W polu *Interfaces* dodaj interfejs typu "Agent" i podaj adres IP serwera MySQL. W tym przewodniku użyto "127.0.0.1" (localhost) do monitorowania serwera MySQL, który jest zainstalowany na tym samym komputerze co serwer Zabbix i agent Zabbix agent 2.



- Na karcie *Macros* przejdź do *Inherited and host macros*, znajdź następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - {$MYSQL.DSN} - ustaw źródło danych serwera MySQL (ciąg połączenia nazwanej sesji](/manual/appendix/config/zabbix_agent2_pl z pliku konfiguracyjnego wtyczki MySQL dla Zabbix agent 2). W tym przewodniku użyto domyślnego źródła danych "tcp://localhost:3306" do monitorowania serwera MySQL, który jest zainstalowany na tym samym komputerze co serwer Zabbix i agent Zabbix agent 2.
  - {$MYSQL.PASSWORD} - ustaw hasło wcześniej utworzonego użytkownika MySQL "zbx_monitor".
  - {$MYSQL.USER} - ustaw nazwę wcześniej utworzonego użytkownika MySQL "zbx_monitor".

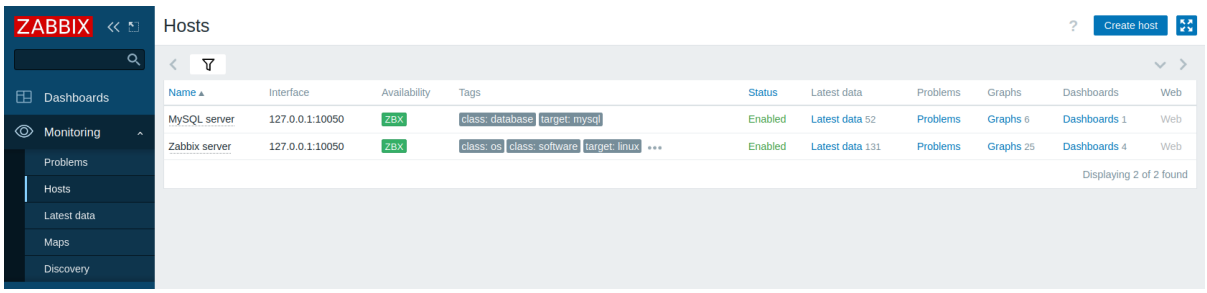


3. Kliknij *Add*, aby dodać host. Ten host będzie reprezentował serwer MySQL.

Wyświetlanie zebranych metryk

Gratulacje! Na tym etapie Zabbix monitoruje już Twój serwer MySQL.

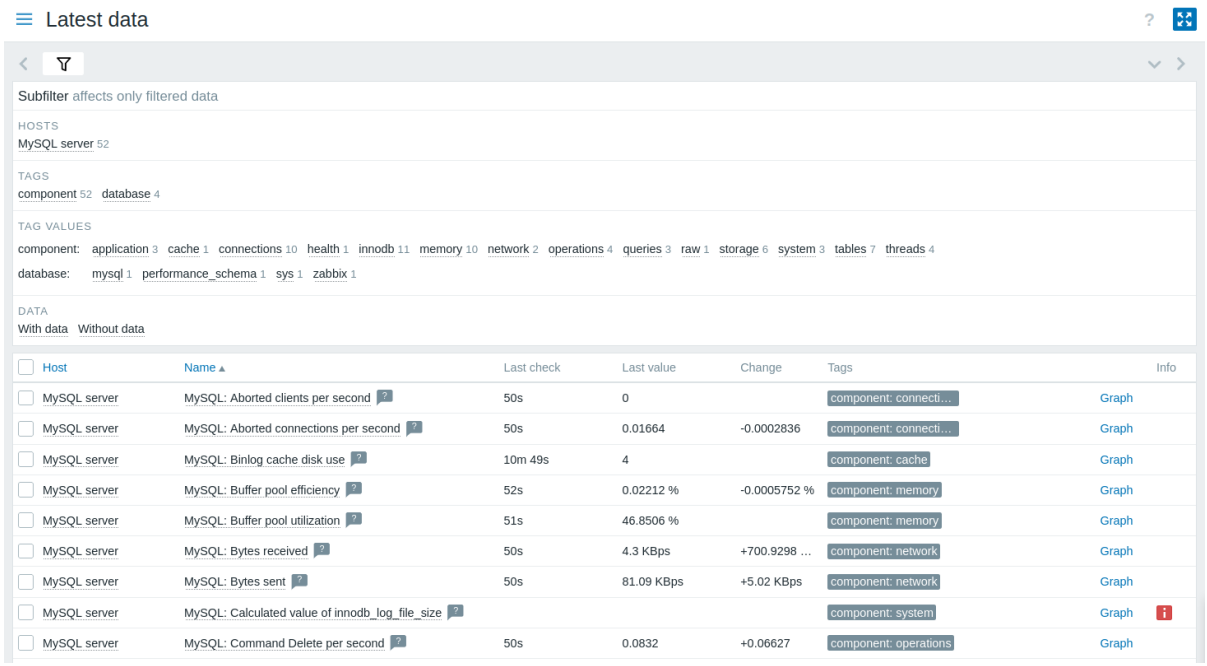
Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring → Hosts* i kliknij *Dashboards* obok hosta.



Ta czynność przeniesie Cię do pulpitu hosta (skonfigurowanego na poziomie szablonu) z najważniejszymi metrykami zebranymi z serwera MySQL.



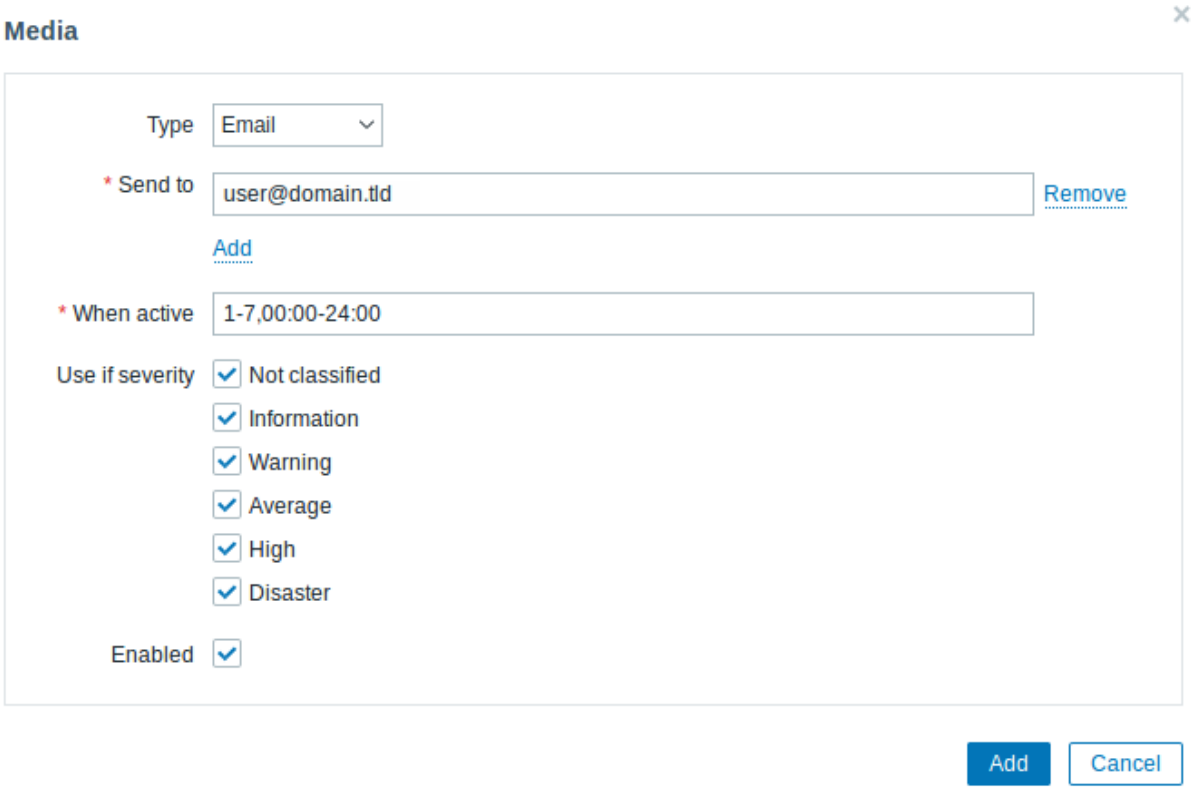
Alternatywnie, w sekcji menu *Monitoring → Hosts* możesz kliknąć *Latest data*, aby wyświetlić wszystkie najnowsze zebrane metryki w formie listy. Zwróć uwagę, że pozycja *MySQL: Calculated value of innodb_log_file_size* zgodnie z oczekiwaniami nie będzie miała danych, ponieważ wartość zostanie obliczona na podstawie danych z ostatniej godziny.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika → Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami w sekcji Odbieranie powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta w Zabbix frontend.

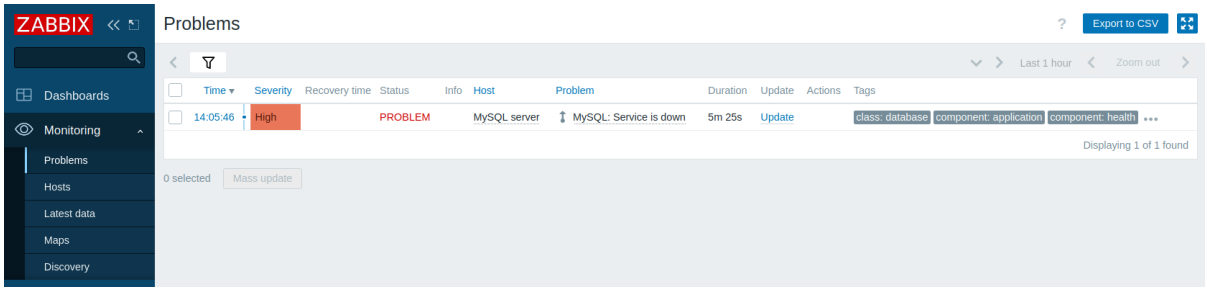
1. Otwórz konfigurację swojego hosta serwera MySQL w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok, na przykład, wcześniej skonfigurowanej wartości makra {$MYSQL.USER} i ustaw inną nazwę użytkownika MySQL.

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Za kilka chwil Zabbix wykryje problem „MySQL: Service is down", ponieważ nie będzie mógł połączyć się z serwerem MySQL. Problem pojawi się w *Monitoring → Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem na poprzednią, aby rozwiązać problem i kontynuować monitorowanie serwera MySQL.

Inne podejścia do monitorowania MySQL

Zamiast monitorować serwer MySQL za pomocą Zabbix agent 2, można również użyć Zabbix agent lub standardu Open Database Connectivity (ODBC). Chociaż zalecane jest używanie Zabbix agent 2, mogą istnieć środowiska, które nie obsługują Zabbix agent 2 lub wymagają niestandardowego podejścia.

Kluczowa różnica między Zabbix agent a ODBC polega na metodzie zbierania danych — Zabbix agent jest instalowany bezpośrednio na serwerze MySQL i zbiera dane przy użyciu wbudowanej funkcjonalności, natomiast ODBC opiera się na sterowniku ODBC, aby nawiązać połączenie z serwerem MySQL i pobierać dane za pomocą zapytań SQL.

Chociaż wiele kroków konfiguracji jest podobnych do monitorowania serwera MySQL za pomocą Zabbix agent 2, istnieją pewne istotne różnice — należy skonfigurować Zabbix agent lub ODBC, aby możliwe było monitorowanie serwera MySQL. Poniższe instrukcje przeprowadzą Cię przez te **różnice**.

Monitorowanie MySQL za pomocą agent Zabbix

Aby monitorować serwer MySQL za pomocą agent Zabbix, należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

Po pomyślnym zainstalowaniu wymaganych komponentów Zabbix należy utworzyć użytkownika MySQL zgodnie z opisem w sekcji *Tworzenie użytkownika MySQL*.

Po utworzeniu użytkownika MySQL należy skonfigurować agent Zabbix tak, aby mógł nawiązać połączenie z serwerem MySQL i go monitorować. Obejmuje to skonfigurowanie wielu parametrów użytkownika do wykonywania niestandardowych kontroli agent, a także dostarczenie agent Zabbix niezbędnych poświadczeń do łączenia się z serwerem MySQL jako wcześniej utworzony użytkownik „zbx_monitor".

Konfiguracja Zabbix agent

1. Przejdź do katalogu dodatkowych konfiguracji Zabbix agent.

```
cd /usr/local/etc/zabbix/zabbix_agentd.d
```

> **Attention:**
> Katalog dodatkowych konfiguracji Zabbix agent powinien znajdować się w tym samym katalogu co plik konfiguracji Zabbix agent (*zabbix_agentd.conf*). W zależności od systemu operacyjnego i instalacji Zabbix, katalog ten może znajdować się w innym miejscu niż podano w tym przewodniku. Aby sprawdzić domyślne lokalizacje, zobacz parametr `Include` w pliku konfiguracji Zabbix agent.

Zamiast definiować w pliku konfiguracji Zabbix agent wszystkie niezbędne parametry użytkownika do monitorowania serwera MySQL, parametry te zostaną zdefiniowane w osobnym pliku w katalogu dodatkowych konfiguracji.

2. Utwórz plik *template_db_mysql.conf* w katalogu dodatkowych konfiguracji Zabbix agent.

```
vi template_db_mysql.conf
```

3. Skopiuj zawartość pliku *template_db_mysql.conf* (znajdującego się w repozytorium Zabbix) do utworzonego pliku *template_db_mysql.conf* i zapisz go.

4. Uruchom ponownie Zabbix agent, aby zaktualizować jego konfigurację.

```
systemctl restart zabbix-agent
```

Po skonfigurowaniu parametrów użytkownika Zabbix agent możesz przejść do konfiguracji poświadczeń, które umożliwią Zabbix agentowi dostęp do serwera MySQL.

5. Przejdź do katalogu domowego agenta Zabbix (jeśli nie istnieje w systemie, należy go utworzyć; domyślnie: */var/lib/zabbix*).

```
cd /var/lib/zabbix
```

6. Utwórz plik *.my.cnf* w katalogu domowym agenta Zabbix.

```
vi .my.cnf
```

7. Skopiuj poniższą zawartość do pliku *.my.cnf* (zastąp *<password>* hasłem użytkownika „zbx_monitor”).

```
[client]
user='zbx_monitor'
password='<password>'
```

Skonfiguruj frontend Zabbix i przetestuj swoją konfigurację

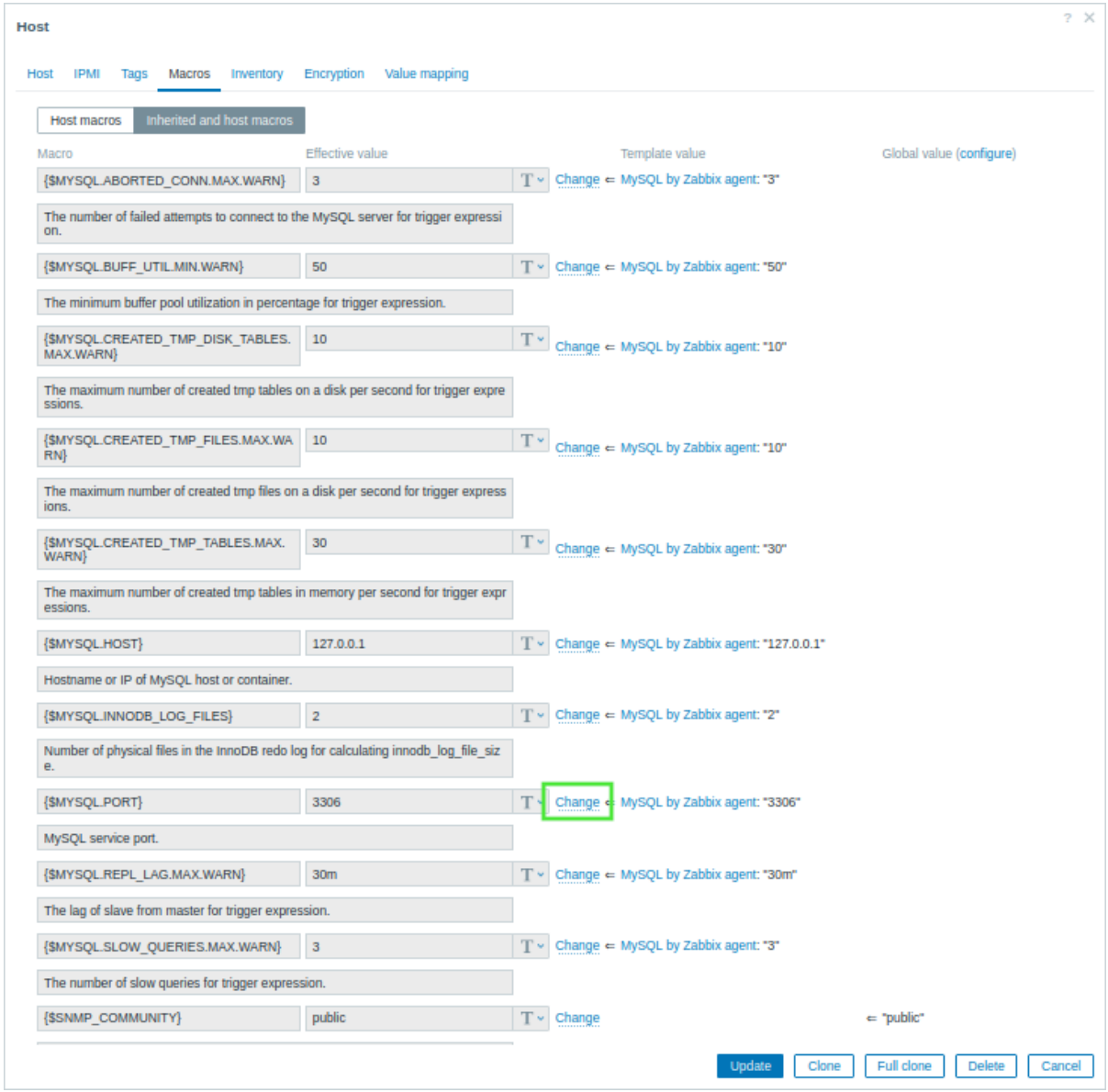
Aby skonfigurować frontend Zabbix, postępuj zgodnie z instrukcjami w sekcji *Skonfiguruj frontend Zabbix*, wprowadzając następujące zmiany:

- W polu *Templates* wpisz lub wybierz szablon „MySQL by Zabbix agent”, który zostanie podlinkowany do hosta.
- Konfiguracja *Macros* nie jest wymagana.

Po skonfigurowaniu frontendu Zabbix możesz wyświetlić zebrane metryki i skonfigurować alerty problemów.

Aby przetestować swoją konfigurację, postępuj zgodnie z instrukcjami w sekcji *Przetestuj swoją konfigurację*, wprowadzając następujące zmiany:

- W sekcji *Inherited and host macros* konfiguracji hosta serwera MySQL kliknij *Change* obok wartości makra {$MYSQL.PORT} i ustaw inny port (np. „6033”).



Monitorowanie MySQL za pomocą ODBC

Aby monitorować serwer MySQL za pomocą ODBC, należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix.

Po pomyślnym zainstalowaniu wymaganych komponentów Zabbix należy utworzyć użytkownika MySQL zgodnie z opisem w sekcji *Tworzenie użytkownika MySQL*.

Po utworzeniu użytkownika MySQL należy skonfigurować ODBC. Obejmuje to instalację jednej z najczęściej używanych implementacji API ODBC typu open source — unixODBC — oraz sterownika unixODBC, a także edycję pliku konfiguracyjnego sterownika ODBC.

Konfiguracja ODBC

1. Zainstaluj unixODBC. Zalecanym sposobem instalacji unixODBC jest użycie domyślnych repozytoriów pakietów systemu operacyjnego Linux.

```
apt install unixodbc
```

2. Zainstaluj sterownik bazy danych MariaDB unixODBC. Mimo że używasz bazy danych MySQL, sterownik MariaDB unixODBC jest używany ze względu na kwestie kompatybilności.

```
apt install odbc-mariadb
```

3. Sprawdź lokalizację plików konfiguracyjnych ODBC *odbcinst.ini* i *odbc.ini*.

```
odbcinst -j
```

Wynik wykonania tego polecenia powinien być podobny do poniższego.

```
unixODBC 2.3.9
DRIVERS............: /etc/odbcinst.ini
```

```
SYSTEM DATA SOURCES: /etc/odbc.ini
FILE DATA SOURCES..: /etc/ODBCDataSources
...
```

4. Aby skonfigurować sterownik ODBC do monitorowania bazy danych MySQL, potrzebujesz nazwy sterownika, która znajduje się w pliku *odbcinst.ini*. W poniższym przykładzie pliku *odbcinst.ini* nazwa sterownika to "MariaDB Unicode".

```
[MariaDB Unicode]
Driver=libmaodbc.so
Description=MariaDB Connector/ODBC(Unicode)
Threading=0
UsageCount=1
```

5. Skopiuj poniższą zawartość do pliku *odbc.ini* (zastąp *<password>* hasłem użytkownika "zbx_monitor"). Ten przewodnik używa adresu "127.0.0.1" (localhost) jako adresu serwera MySQL do monitorowania serwera MySQL zainstalowanego na tej samej maszynie co sterownik ODBC. Zwróć uwagę na nazwę źródła danych (DSN) "test", która będzie wymagana podczas konfiguracji frontend Zabbix.

```
[test]
Driver=MariaDB Unicode
Server=127.0.0.1
User=zbx_monitor
Password=<password>
Port=3306
Database=zabbix
```

Skonfiguruj frontend Zabbix i przetestuj swoją konfigurację

Aby skonfigurować frontend Zabbix, postępuj zgodnie z instrukcjami w sekcji *Skonfiguruj frontend Zabbix*, wprowadzając następujące zmiany:

- W polu *Templates* wpisz lub wybierz szablon „MySQL by ODBC", który zostanie podłączony do hosta.
- Konfigurowanie *Interfaces* nie jest wymagane.
- Wartość makra {$MYSQL.DSN} w sekcji *Inherited and host macros* konfiguracji hosta serwera MySQL powinna być ustawiona na nazwę DSN z pliku *odbc.ini*.

Po skonfigurowaniu frontend Zabbix możesz wyświetlić zebrane metryki, skonfigurować alerty problemów oraz przetestować swoją konfigurację.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Monitorowanie ODBC - jak skonfigurować ODBC w innych dystrybucjach Linuksa oraz jak rozpocząć monitorowanie dodatkowych metryk związanych z bazą danych za pomocą ODBC.
- Szablon *MySQL by Zabbix agent* - dodatkowe informacje o szablonie *MySQL by Zabbix agent*.
- Szablon *MySQL by Zabbix agent 2* - dodatkowe informacje o szablonie *MySQL by Zabbix agent 2*.
- Szablon *MySQL by ODBC* - dodatkowe informacje o szablonie *MySQL by ODBC*.

5 Monitoruj VMware za pomocą Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane by rozpocząć podstawowe monitorowanie VMware.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania VMware. Jeśli szukasz opcji zaawansowanego dostosowania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Monitorowanie maszyn wirtualnych lub sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Zakłada się, że VMware jest już skonfigurowane. Ten przewodnik nie obejmuje konfiguracji VMware.

Konfiguracja serwera Zabbix

Aby monitorować VMware, należy włączyć procesy Zabbix *vmware collector*. Więcej informacji o sposobie działania monitorowania VMware można znaleźć w sekcji Monitorowanie maszyn wirtualnych.

1. Otwórz plik konfiguracyjny serwera Zabbix.

```
vi /etc/zabbix/zabbix_server.conf
```

2. Odszukaj parametr `StartVMwareCollectors` w pliku konfiguracyjnym serwera Zabbix i ustaw go na 2 lub więcej (wartość domyślna to 0).

```
##### Option: StartVMwareCollectors
###       Number of pre-forked vmware collector instances.
###
### Mandatory: no
### Range: 0-250
### Default:
### StartVMwareCollectors=0

StartVMwareCollectors=2
```

3. Uruchom ponownie serwer Zabbix.

```
systemctl restart zabbix-server
```

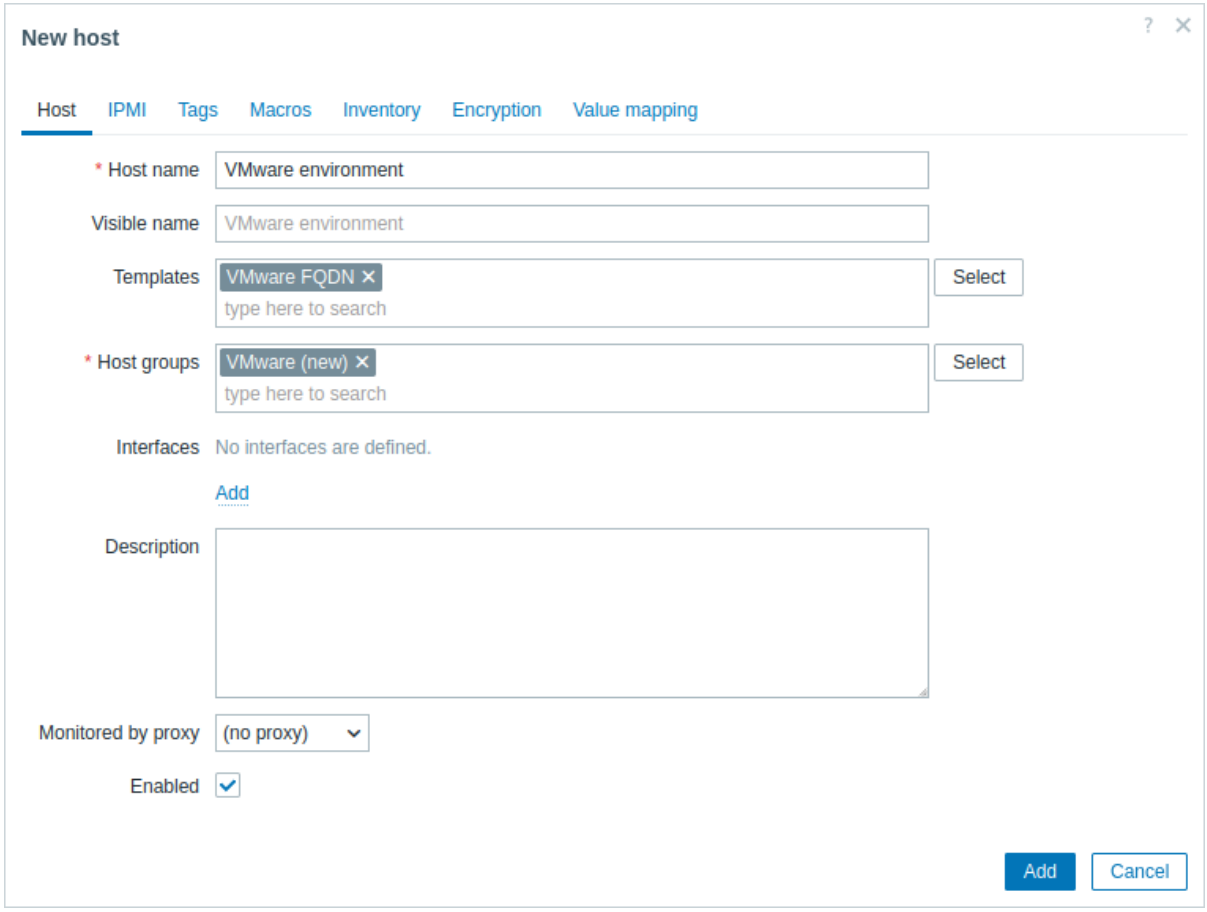
Po uruchomieniu procesów *vmware collector* przejdź do następnego kroku.

Konfiguracja frontendu Zabbix

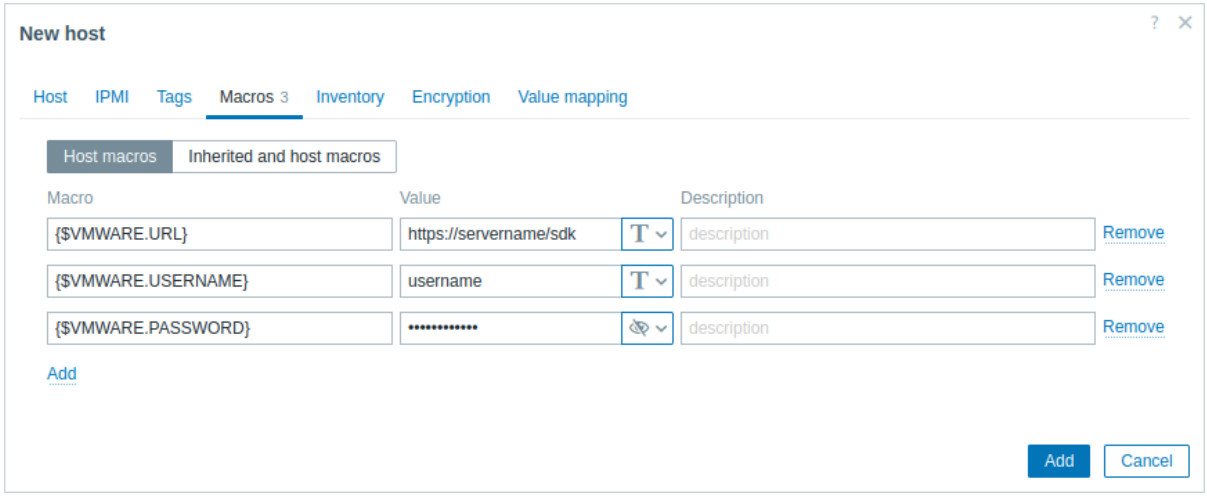
1. Zaloguj się do frontendu Zabbix.

2. Utwórz host w interfejsie webowym Zabbix:

- W polu *Host name* wprowadź nazwę hosta (na przykład „VMware environment”).
- W polu *Templates* wpisz lub wybierz szablon „VMware FQDN” (lub „VMware”). Więcej informacji o tych szablonach znajdziesz w sekcji Monitorowanie maszyn wirtualnych.
- W polu *Host groups* wpisz lub wybierz grupę hostów (na przykład nową grupę hostów „VMware”).



- Na karcie *Macros* ustaw następujące makra hosta:
  - {$VMWARE.URL} — URL SDK usługi VMware (vCenter lub hiperwizora ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} — nazwa użytkownika usługi VMware
  - {$VMWARE.PASSWORD} — hasło użytkownika {$VMWARE.USERNAME} usługi VMware



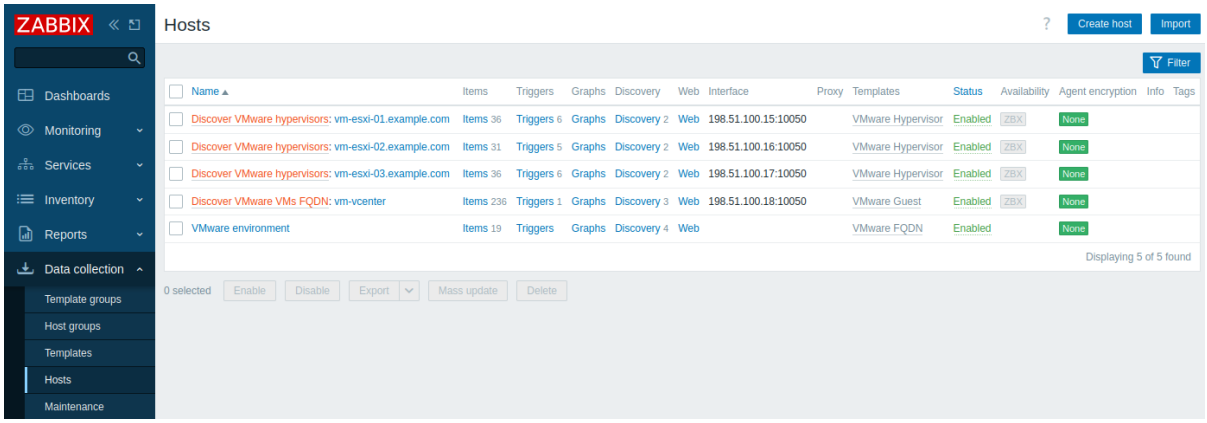
3. Kliknij przycisk *Add*, aby utworzyć host. Ten host będzie reprezentował Twoje środowisko VMware.

Wyświetlanie zebranych metryk

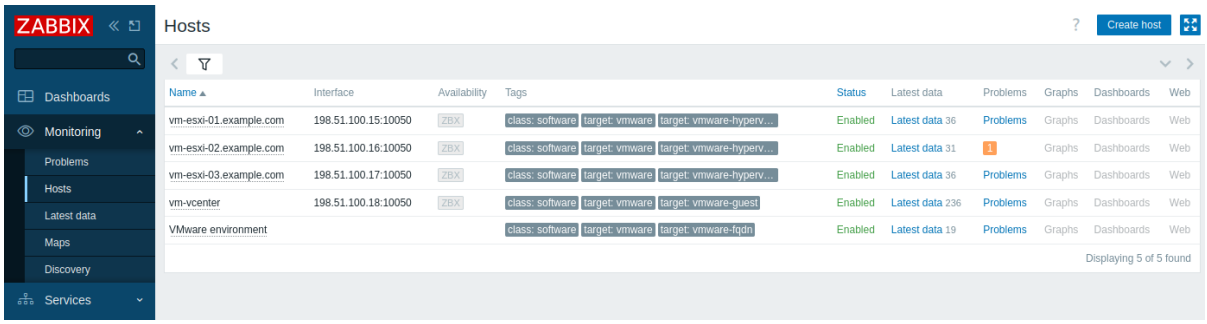
Gratulacje! Na tym etapie Zabbix już monitoruje Twoje środowisko VMware.

W zależności od konfiguracji środowiska VMware, Zabbix może wykryć, a następnie utworzyć hosty dla wykrytych encji. Pamiętaj, że wykrywanie i tworzenie hostów może również zostać wykonane ręcznie, jeśli to konieczne.

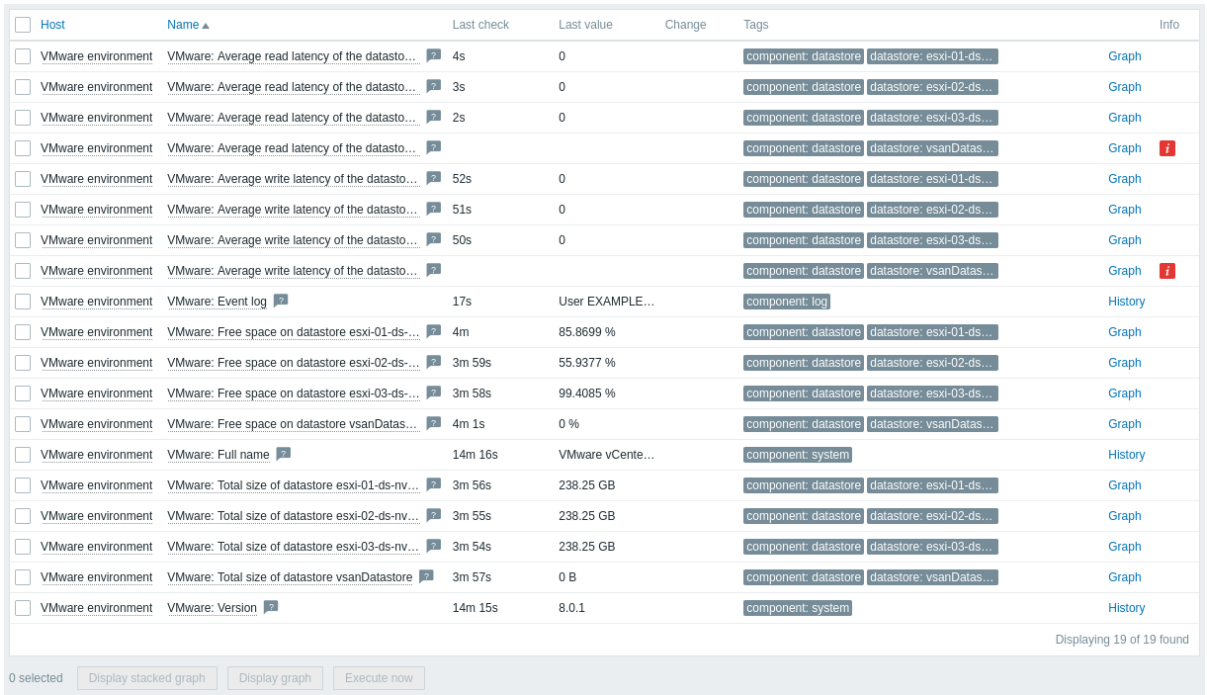
Aby wyświetlić utworzone hosty, przejdź do sekcji menu *Data collection → Hosts*.



Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring → Hosts* i kliknij *Latest data* obok utworzonego hosta „VMware environment” lub jednego z hostów utworzonych dla wykrytych encji.



Ta czynność otworzy listę wszystkich najnowszych metryk zebranych z wybranego hosta.

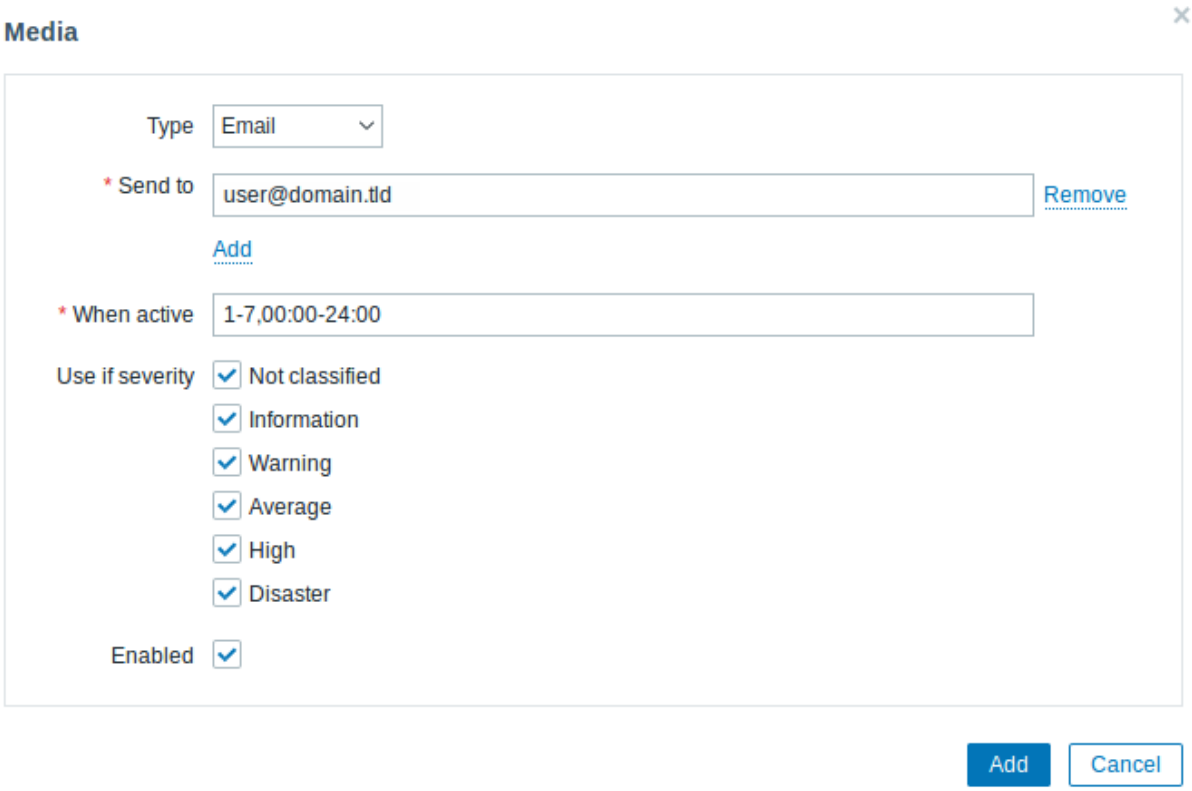


Pamiętaj, że niektóre pozycje nie mają danych i mają stan *Not supported*. Dzieje się tak, ponieważ Zabbix nie może znaleźć prawidłowych liczników wydajności w określonym magazynie danych, ponieważ nie są one włączone w monitorowanym środowisku VMware.

Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemie z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika → Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami dla Odbierania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Monitorowanie maszyn wirtualnych - dodatkowe informacje o monitorowaniu VMware (proces zbierania danych, opcje konfiguracji serwera, wskazówki dotyczące rozwiązywania problemów itp.).
- Klucze pozycji monitorowania VMware - pełna lista metryk VMware, które można monitorować za pomocą Zabbixa.
- Szablon *VMware* - dodatkowe informacje o szablonie *VMware*.
- Szablon *VMware FQDN* - dodatkowe informacje o szablonie *VMware FQDN*.

6 Monitoruj ruch sieciowy za pomocą Zabbix

Wprowadzenie Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania ruchu sieciowego za pomocą Zabbix.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania ruchu sieciowego. Jeśli szukasz opcji zaawansowanego dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Konfiguracja podręcznika Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować Zabbix serwer, Zabbix frontend oraz Zabbix agent zgodnie z instrukcjami dla używanego systemu operacyjnego. Należy pamiętać, że Zabbix agent powinien zostać zainstalowany na maszynie, która wymaga monitorowania ruchu sieciowego. Może to być ten sam host, na którym zainstalowany jest Zabbix serwer, lub inny host.

Ten przewodnik zawiera instrukcje dotyczące konfiguracji monitorowania ruchu sieciowego interfejsu *eth0* na oddzielnej maszynie o nazwie *Remote host*.

Konfiguracja Zabbix do monitorowania Agent Zabbix może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie). Więcej informacji można znaleźć w sekcji Pasywne i aktywne kontrole agenta. W tym przewodniku zostanie opisane monitorowanie za pomocą kontroli pasywnych.

Konfigurowanie agenta Zabbix

1. Otwórz plik konfiguracyjny agenta na maszynie, na której agent jest zainstalowany (domyślnie ścieżka to */usr/local/etc/zabbix_agentd.conf*):

```
sudo vi /usr/local/etc/zabbix_agentd.conf
```

2. Dodaj adres IP lub nazwę DNS swojego serwera Zabbix do parametru *Server*. Na przykład:

```
Server=192.0.2.22
```

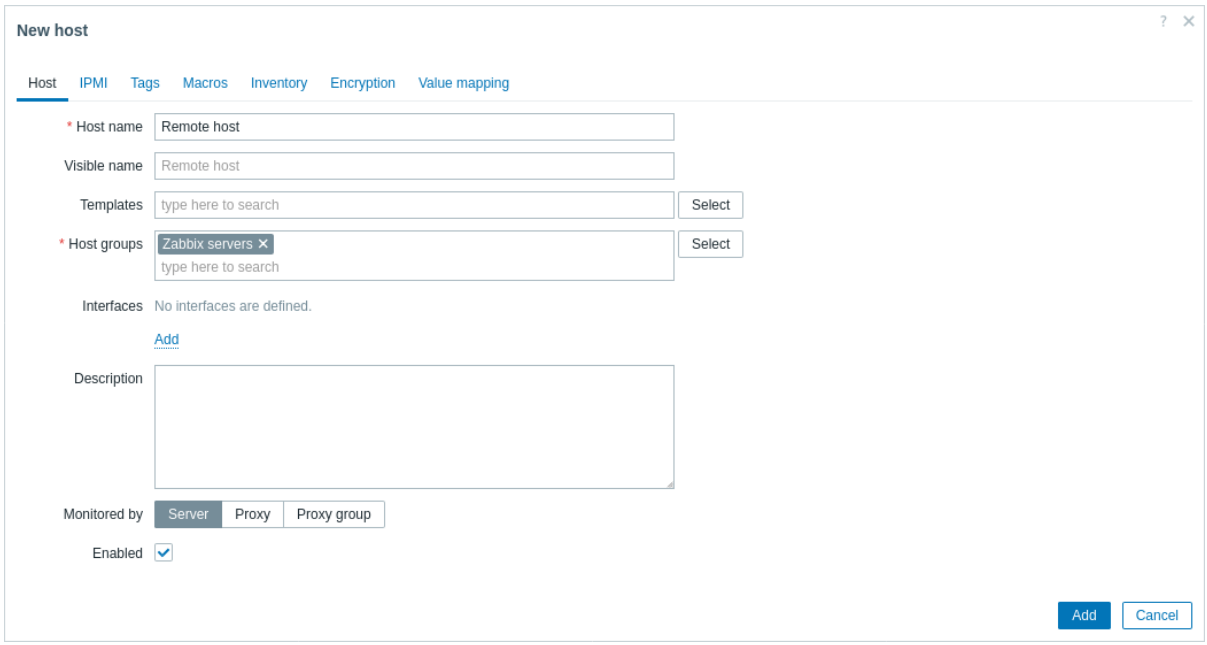
3. Uruchom ponownie agenta Zabbix:

```
systemctl restart zabbix-agent
```

Frontend Zabbix

1. Zaloguj się do frontend Zabbix.

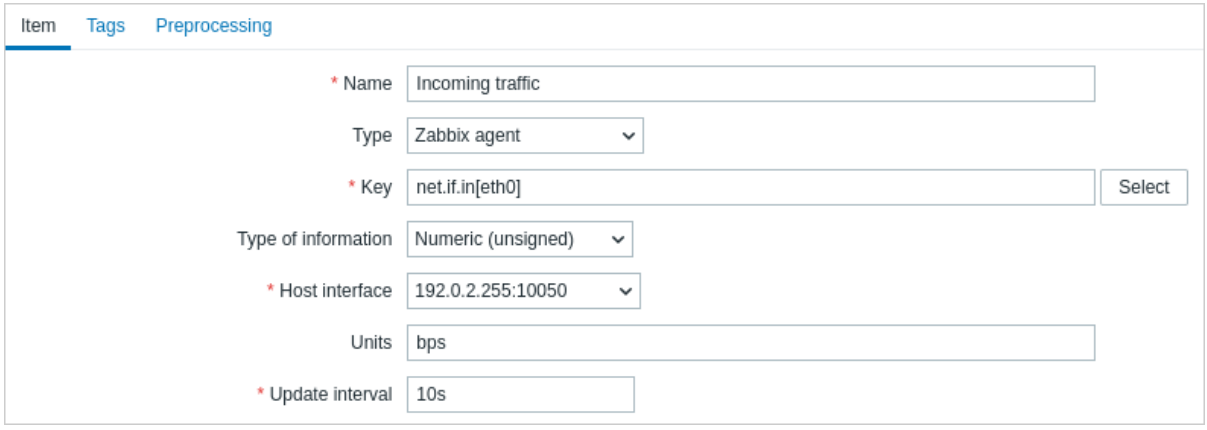
2. Utwórz hosta w interfejsie webowym Zabbix, podając adres IP lub nazwę DNS komputera, na którym zainstalowano agent.



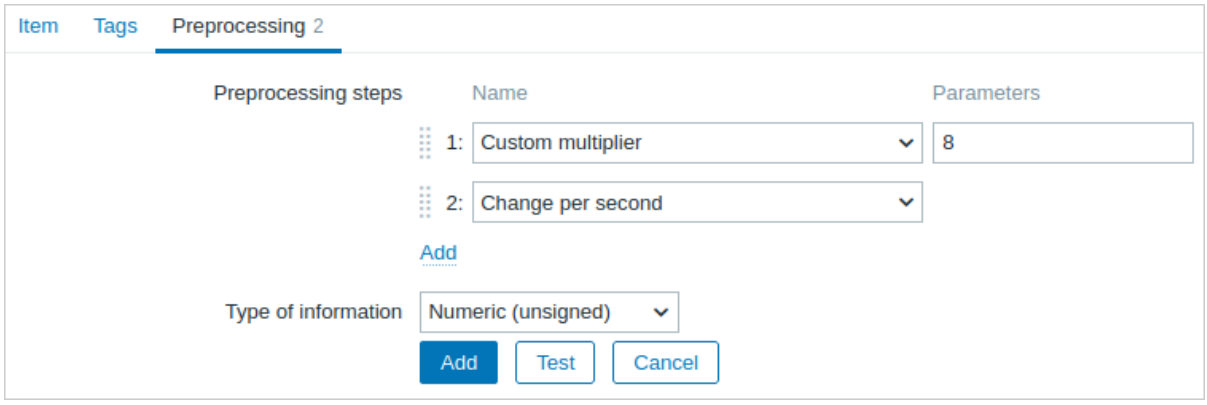
Tworzenie pozycji Postępuj zgodnie z instrukcjami dotyczącymi tworzenia pozycji, aby dodać pozycje do monitorowania ruchu, a mianowicie:

- Ruch przychodzący
- Ruch wychodzący
- Całkowity ruch

Prosta konfiguracja pozycji monitorującej ruch przychodzący będzie wyglądać następująco:

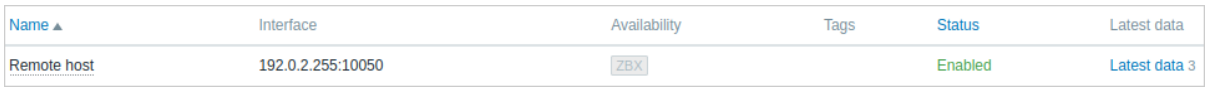


Aby zebrane dane nadawały się do praktycznego wykorzystania, podczas tworzenia pozycji możesz skonfigurować kilka kroków przetwarzania wstępnego. W tym przypadku mogą to być mnożenie przez 8 (aby przeliczyć bajty na bity) oraz prezentacja jako zmiana na sekundę.

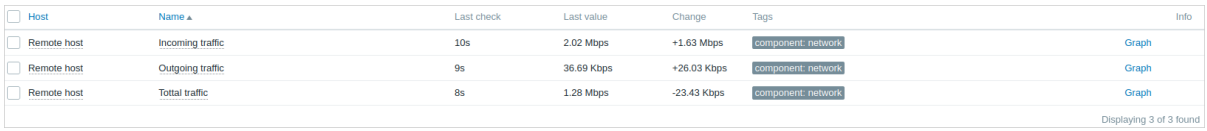


Wyświetlanie zebranych danych Gratulacje! Na tym etapie Zabbix już monitoruje ruch w Twojej sieci.

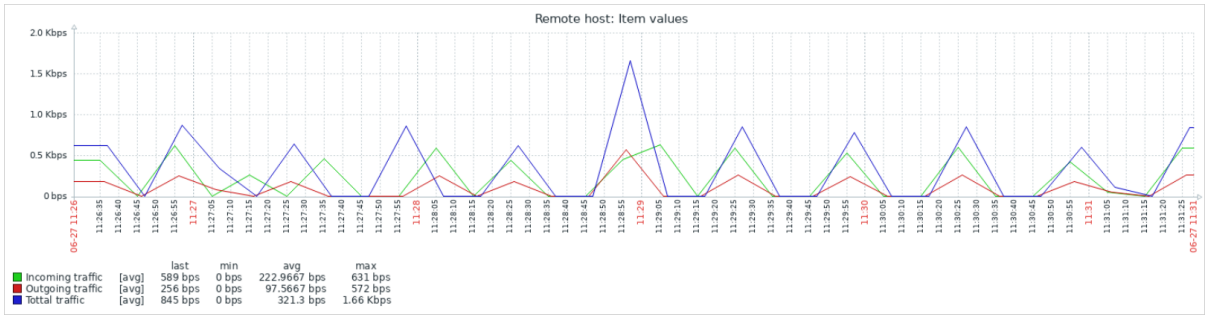
Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring → Hosts* i kliknij *Latest data* w wierszu hosta.



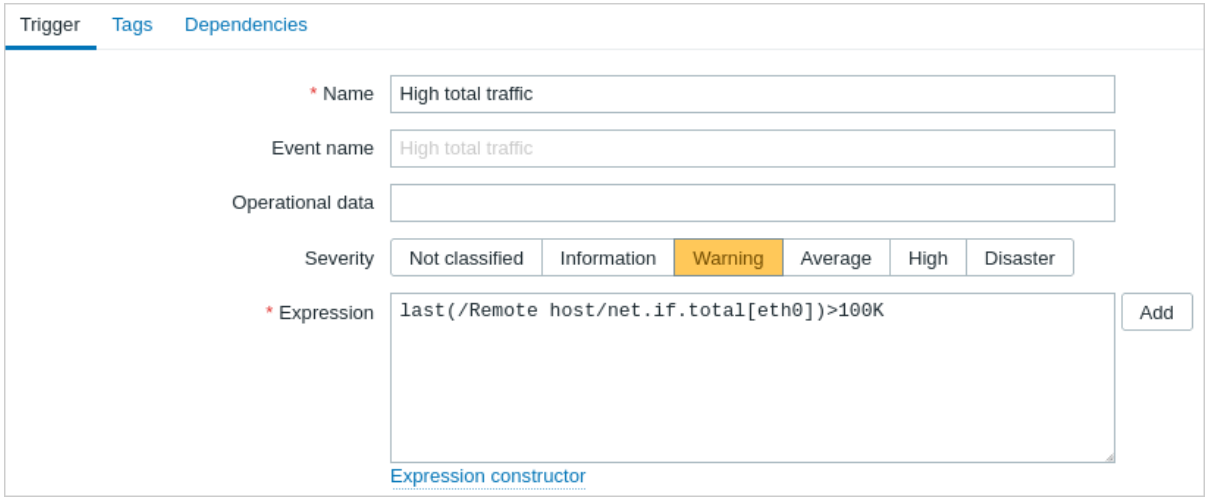
Zobaczysz dane o ruchu.



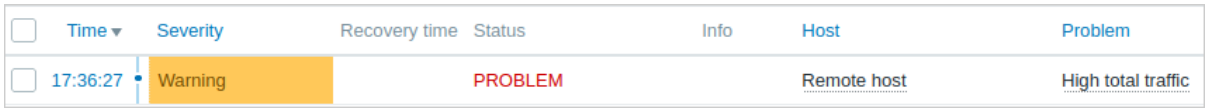
Wyświetlanie wykresów Zebrane dane mogą być wyświetlane jako wykresy. Aby je wyświetlić, w sekcji *Najnowsze dane* kliknij *Wykres* w wierszu danej pozycji lub wybierz wymagane pozycje i kliknij *Wyświetl wykres* poniżej.



Konfiguracja wyzwalaczy Możesz skonfigurować wyzwalacze, aby wykrywać nieprawidłowy ruch sieciowy. Zapoznaj się z instrukcjami dotyczącymi konfigurowania wyzwalacza i dodaj wyzwalacz sygnalizujący, że całkowity ruch jest zbyt wysoki, na przykład:



Teraz spraw, aby ruch przekroczył próg ustawiony w wyrażeniu wyzwalacza, a następnie przejdź do *Monitoring* → *Problems*, aby sprawdzić, czy problem jest tam wyświetlany.



Skonfiguruj alerty o problemach Istnieje kilka sposobów otrzymywania powiadomień o problemie. E-mail jest najpopularniejszą opcją, dlatego postępuj zgodnie z instrukcjami dotyczącymi konfiguracji powiadomienia o problemie za pośrednictwem e-maila. Możesz również wybrać inne typy mediów, które będą używane do dostarczania powiadomień.

Zobacz także:

- Eskalacje problemów — jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Potwierdzanie problemów — jak wskazać, że problem jest znany, dodawać komentarze dotyczące jego rozwiązywania, tłumić lub zamykać problem.
- Monitorowanie systemu Linux za pomocą agent Zabbix — jak rozpocząć podstawowe monitorowanie najważniejszych pozycji przez podłączenie wstępnie skonfigurowanego szablon.

7 Monitoruj ruch sieciowy za pomocą aktywnych kontroli

Wprowadzenie Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania ruchu sieciowego za pomocą Zabbix z użyciem aktywnych kontroli.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania ruchu sieciowego przy użyciu aktywnych kontroli. Jeśli szukasz opcji zaawansowanego dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Configuration podręcznika Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego. Należy pamiętać, że agent Zabbix powinien zostać zainstalowany na maszynie, która wymaga monitorowania ruchu sieciowego. Może to być ten sam host, na którym zainstalowany jest serwer Zabbix, lub inny host.

Ten przewodnik zawiera instrukcje dotyczące konfiguracji monitorowania ruchu sieciowego interfejsu *eth0* na oddzielnej maszynie o nazwie *Remote host*.

Konfiguracja Zabbix do monitorowania Agent Zabbix może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie). Więcej informacji można znaleźć w sekcji Pasywne i aktywne kontrole agenta. W tym przewodniku zostanie opisane monitorowanie za pomocą **aktywnych kontroli**.

Konfigurowanie Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której agent jest zainstalowany.

Jeśli używasz Zabbix agent:

```
sudo vi /etc/zabbix/zabbix_agentd.conf
```

Jeśli używasz Zabbix agent 2:

```
sudo vi /etc/zabbix/zabbix_agent2.conf
```

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalny port) swojego serwera Zabbix do parametru ServerActive. Na przykład:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent użyje tego adresu do połączenia się z portem trapper serwera Zabbix (domyślnie: 10051) i zażądania danych konfiguracyjnych dla aktywnych kontroli.

3. Zdefiniuj parametr Hostname, który musi odpowiadać nazwie hosta zdefiniowanej w Zabbix frontend. W naszym przykładzie jest to:

```
Hostname=Remote host
```

Wartość Hostname musi się zgadzać, ponieważ w przypadku aktywnych kontroli Zabbix agent używa jej do pobrania z serwera prawidłowej konfiguracji hosta. Dokładniej mówiąc, agent inicjuje połączenie z serwerem i identyfikuje się za pomocą wartości Hostname. Następnie serwer udostępnia konfigurację monitorowania dla tego hosta. Jeśli te wartości będą różne, agent nie otrzyma odpowiedniej konfiguracji, co spowoduje brakujące metryki lub problemy z monitorowaniem.

4. Uruchom ponownie Zabbix agent.

Jeśli używasz Zabbix agent:

```
systemctl restart zabbix-agent
```

Jeśli używasz Zabbix agent 2:

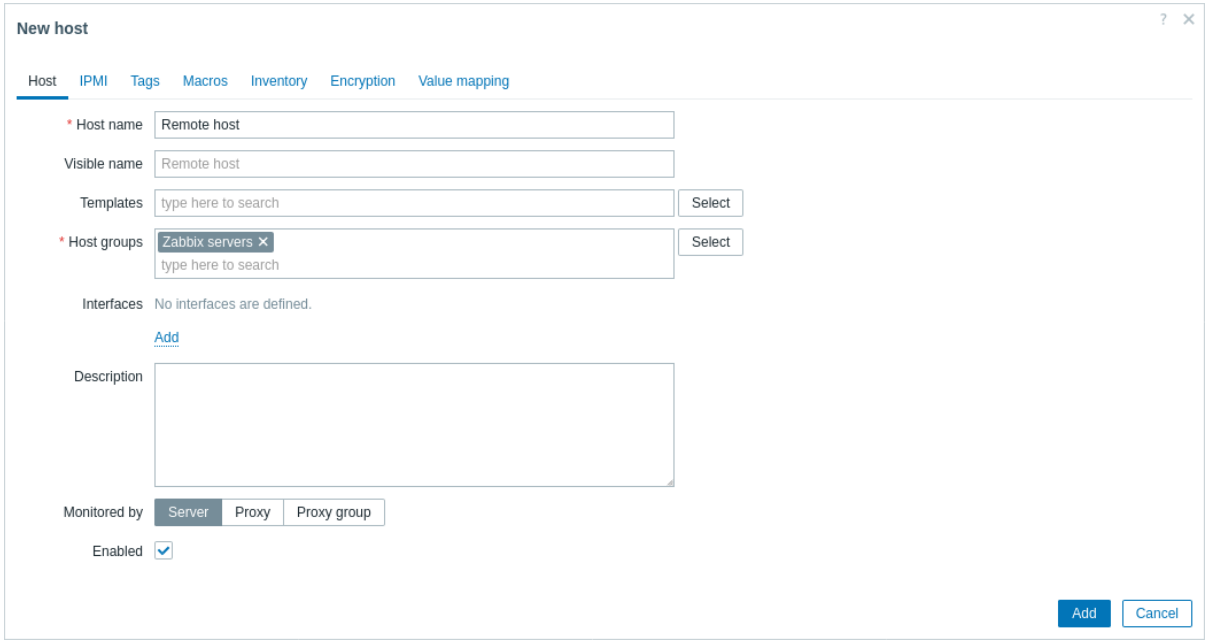
```
systemctl restart zabbix-agent2
```

Zabbix frontend

1. Zaloguj się do Zabbix frontend.

2. Utwórz host w interfejsie webowym Zabbix.

- W polu *Host name* wprowadź nazwę hosta (np. „Remote host”), która odpowiada wartości parametru Hostname zdefiniowanej wcześniej w pliku konfiguracyjnym agent.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. „Zabbix servers”).
- Pozostaw *Interfaces* niezdefiniowane; interfejs nie jest wymagany dla aktywnych testów, ponieważ agent inicjuje połączenie z serwerem, zamiast nasłuchiwać połączenia od serwera.

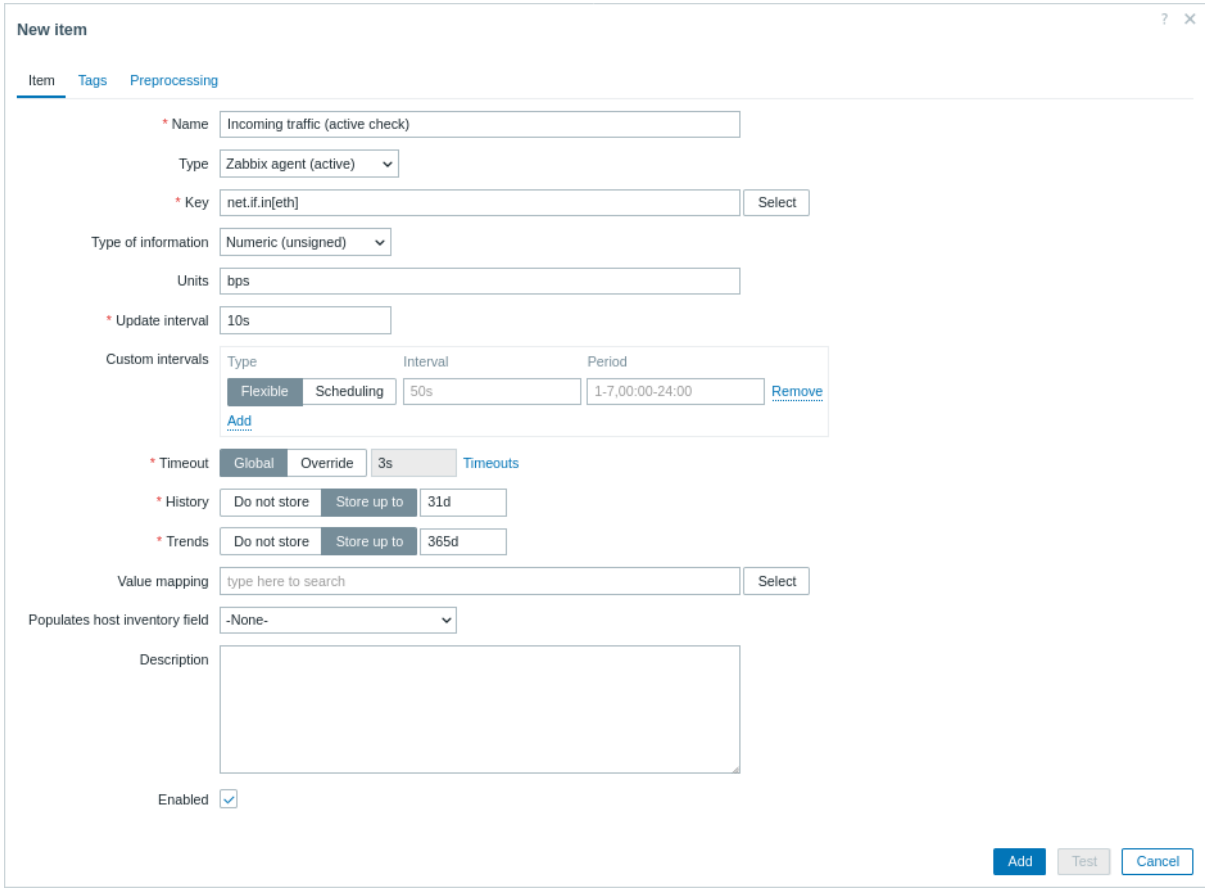


3. Kliknij *Add*, aby dodać hosta. Ten host będzie reprezentował monitorowaną maszynę z systemem Linux.

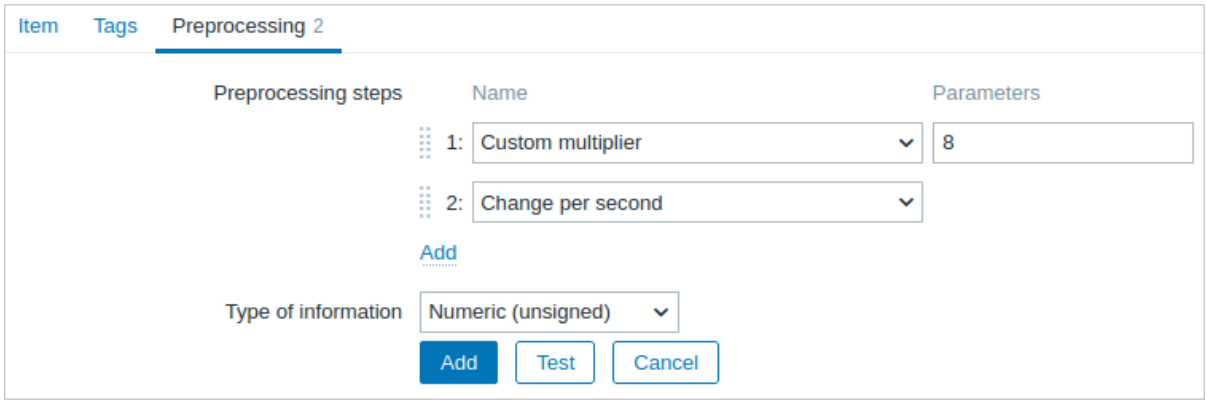
Tworzenie pozycji Postępuj zgodnie z instrukcjami dotyczącymi tworzenia pozycji, aby dodać pozycje do monitorowania ruchu, a mianowicie:

- Ruch przychodzący
- Ruch wychodzący
- Całkowity ruch

Prosta konfiguracja pozycji monitorowania ruchu przychodzącego z aktywnym sprawdzaniem wyglądałaby następująco:

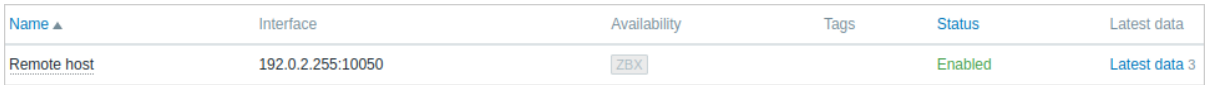


Aby zebrane dane nadawały się do praktycznego wykorzystania, podczas tworzenia pozycji można ustawić kilka kroków przetwarzania wstępnego. W podanym przypadku może to być mnożenie przez 8 (aby przeliczyć bajty na bity) oraz prezentacja jako zmiana na sekundę.

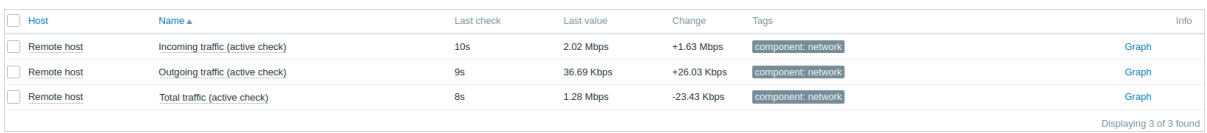


Wyświetlanie zebranych danych Gratulacje! Na tym etapie Zabbix monitoruje już ruch sieciowy.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring → Hosts* i kliknij *Latest data* w wierszu hosta.



Zobaczysz dane dotyczące ruchu.



Zobacz także:

- Wyświetlanie wykresów - jak wyświetlać zebrane dane jako wykresy.
- Konfiguracja wyzwalaczy - jak skonfigurować wyzwalacze do wykrywania nietypowego ruchu sieciowego.
- Konfiguracja alertów o problemach - jak skonfigurować powiadomienia o sytuacjach problemowych.

- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Potwierdzanie problemu - jak wskazać, że problem jest znany, dodawać komentarze dotyczące jego rozwiązania, wstrzymywać lub zamykać problem.
- Monitorowanie Linuxa za pomocą agent Zabbix - jak rozpocząć podstawowe monitorowanie najważniejszych pozycji poprzez powiązanie wstępnie skonfigurowanego szablonu.
- Instalacja z pakietów - jak instalować komponenty Zabbix przy użyciu oficjalnych pakietów RPM i DEB dla różnych dystrybucji Linuxa, zapewniając dostęp do najnowszych funkcji i poprawek błędów.

8 Monitorowanie stron internetowych za pomocą pozycji przeglądarki

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania stron internetowych za pomocą pozycji przeglądarkowych.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania stron internetowych za pomocą pozycji przeglądarkowych. Jeśli szukasz opcji zaawansowanego dostosowania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz stronę Pozycje przeglądarkowe lub sekcję Konfiguracja podręcznika Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Konfiguracja WebDriver

Pozycje przeglądarki wymagają frameworka automatyzacji (Selenium Server lub zwykłego WebDriver, na przykład ChromeDriver) jako punktu końcowego testów web, który steruje przeglądarką i wchodzi z nią w interakcję, wykonując polecenia testowe, takie jak klikanie przycisków lub wprowadzanie tekstu. W tym przewodniku jako przykład zostanie użyty Selenium Server z Chrome w kontenerze Docker.

Zakłada się, że Docker jest już skonfigurowany. Ten przewodnik nie obejmuje konfiguracji Dockera. Instrukcje instalacji można znaleźć tutaj: Install Docker Engine on Ubuntu.

1. Uruchom Selenium Server z Chrome w kontenerze Docker z następującymi opcjami:

- **docker run --name browser** — uruchamia nowy kontener Docker o nazwie „browser”;
- **-p 4444:4444** — mapuje port 4444 na twoim hoście na port 4444 w kontenerze (jest to port używany przez Selenium Server do przyjmowania poleceń);
- **-p 7900:7900** — mapuje port 7900 na twoim hoście na port 7900 w kontenerze (jest to port używany przez serwer Virtual Network Computing (VNC), umożliwiający zdalne wyświetlanie GUI przeglądarki; wymaga klienta VNC);
- **--shm-size=”2g”** — przydziela kontenerowi 2 GB pamięci współdzielonej (jest to ważne dla prawidłowego działania Chrome, ponieważ może on wymagać znacznej ilości pamięci współdzielonej, aby uniknąć awarii);
- **-d** — uruchamia kontener w trybie odłączonym, co oznacza, że będzie działał w tle;
- **selenium/standalone-chrome:latest** — określa obraz Docker do użycia; w tym przypadku jest to najnowsza wersja Selenium Server with Chrome.

```
docker run --name browser \
-p 4444:4444 \
-p 7900:7900 \
--shm-size="2g" \
-d selenium/standalone-chrome:latest
```

2. Upewnij się, że kontener Docker `browser` działa i jest dostępny.

- Pobierz adres IP kontenera (w tym przykładzie 192.0.2.1):

```
ip addr

### 1: lo: <LOOPBACK,UP,LOWER_UP>
```

```
###     ...
### 3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> ...
###     inet 192.0.2.1/16 brd 192.0.255.255 scope global docker0
###     ...
```

- Przetestuj połączenie z kontenerem za pomocą Ncat:

```
nc -zv 192.0.2.1 4444

### Connection to 192.0.2.1 4444 port [tcp/*] succeeded!
```

- Pobierz zawartość strony internetowej z Selenium Server za pomocą curl:

```
curl -L 192.0.2.1:4444

### <!DOCTYPE html>
### <html lang="en">
###
### <head>
###   <meta charset="utf-8"/>
###   <link href="favicon.svg" rel="icon" type="image/svg">
###   <meta content="width=device-width, initial-scale=1" name="viewport"/>
###   <link href="logo192.png" rel="apple-touch-icon"/>
###   <link href="manifest.json" rel="manifest"/>
###   <title>Selenium Grid</title>
### </head>
###
### <body>
### ...
```

Note:
W celu rozwiązywania problemów zapoznaj się z Docker documentation.

Skonfiguruj serwer Zabbix

Pozycje przeglądarki są wykonywane i przetwarzane przez procesy Zabbix *browser poller*, które należy włączyć, dostosowując parametr konfiguracji serwera StartBrowserPollers. Dodatkowo parametr WebDriverURL powinien wskazywać wcześniej skonfigurowany endpoint testów webowych.

Domyślnie parametr StartBrowserPollers jest ustawiony na 1, dlatego wystarczy określić endpoint testów webowych.

1. Otwórz plik konfiguracyjny serwera Zabbix.

```
vi /etc/zabbix/zabbix_server.conf
```

2. Odszukaj i ustaw parametr WebDriverURL w pliku konfiguracyjnym serwera Zabbix:

```
##### Option: WebDriverURL
###WebDriver interface HTTP[S] URL. For example http://localhost:4444 used with Selenium WebDriver standal
###
### Mandatory: no
### Default:
### WebDriverURL=

WebDriverURL=192.0.2.1:4444
```

3. Uruchom ponownie serwer Zabbix.

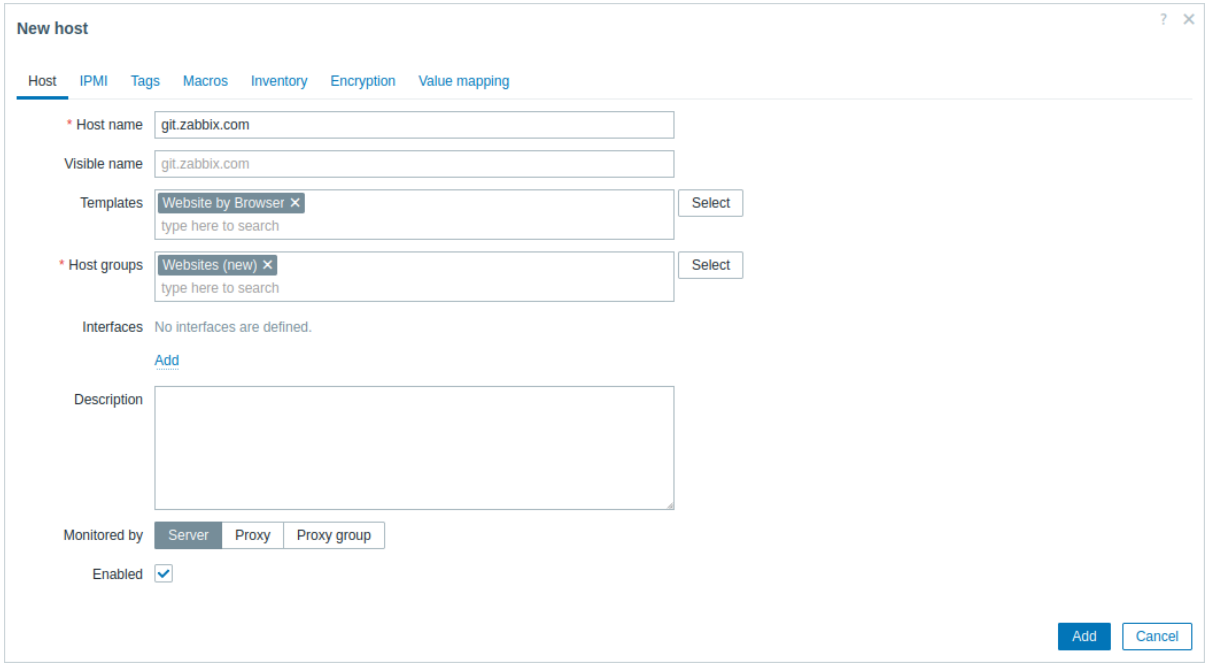
```
systemctl restart zabbix-server
```

Skonfiguruj frontend Zabbixa

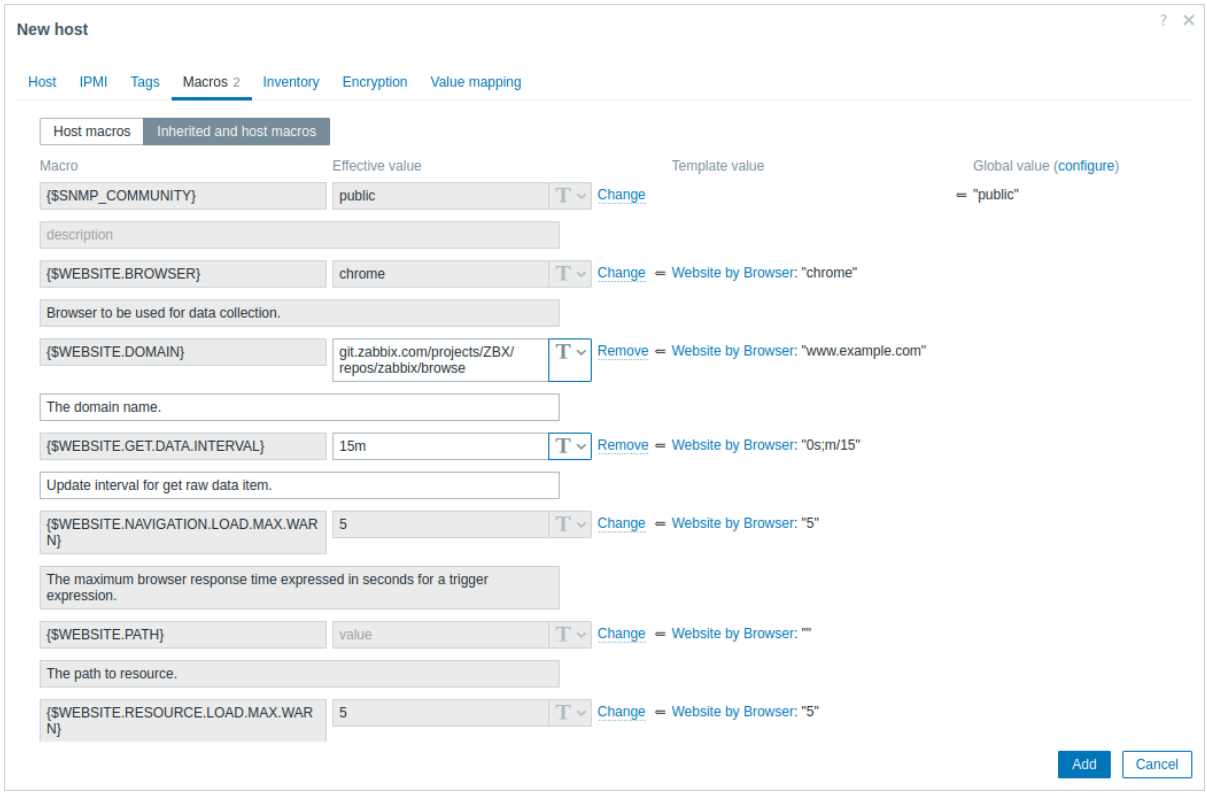
1. Zaloguj się do frontend Zabbixa.

2. Utwórz host w interfejsie webowym Zabbixa:

- W polu *Host name* wpisz nazwę hosta (na przykład "git.zabbix.com").
- W polu *Templates* wpisz lub wybierz szablon "Website by Browser". Więcej informacji o tym szablonie znajdziesz tutaj: Website by Browser.
- W polu *Host groups* wpisz lub wybierz grupę hostów (na przykład nową grupę hostów "Websites").



- Na karcie *Macros* przełącz się na *Inherited and host macros*, znajdź następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - {$WEBSITE.DOMAIN} - nazwa domeny (na przykład git.zabbix.com/projects/ZBX/repos/zabbix/browse)
  - {$WEBSITE.GET.DATA.INTERVAL} - interwał aktualizacji danych pozycji (na przykład 15m)

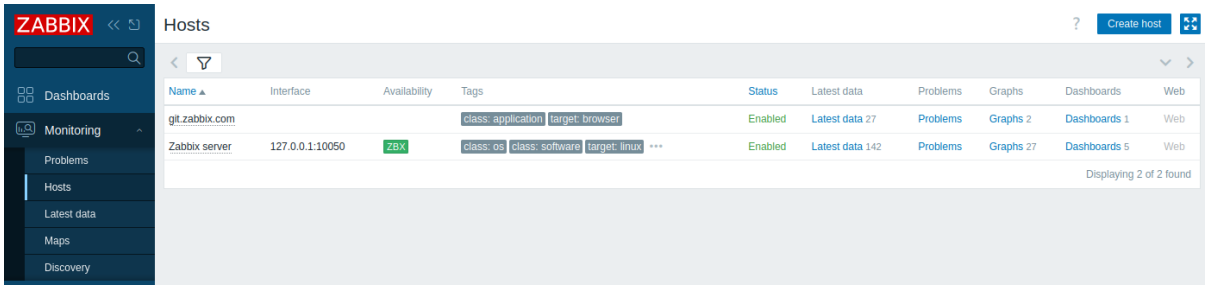


3. Kliknij przycisk *Add*, aby utworzyć host. Ten host będzie reprezentować witrynę, którą chcesz monitorować.

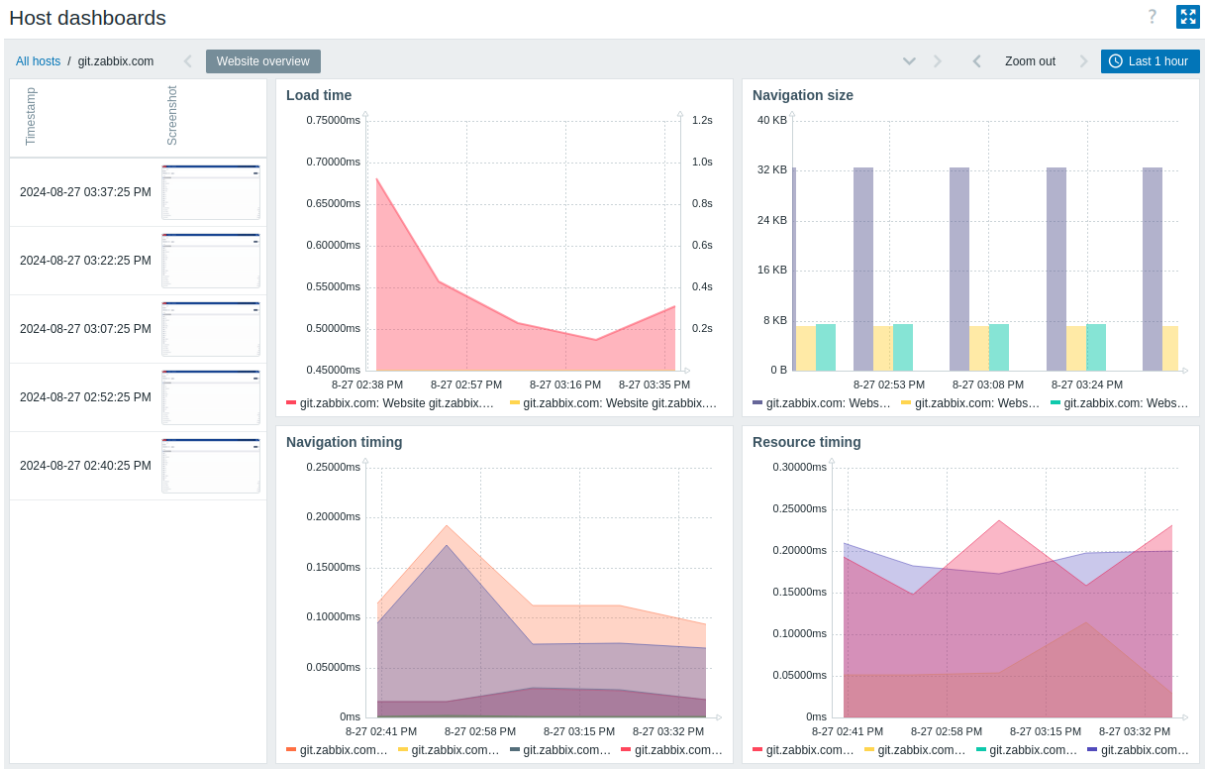
Wyświetlanie zebranych metryk

Gratulacje! Na tym etapie Zabbix monitoruje już wskazaną przez Ciebie witrynę internetową.

Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring → Hosts* i kliknij *Dashboards* obok hosta.



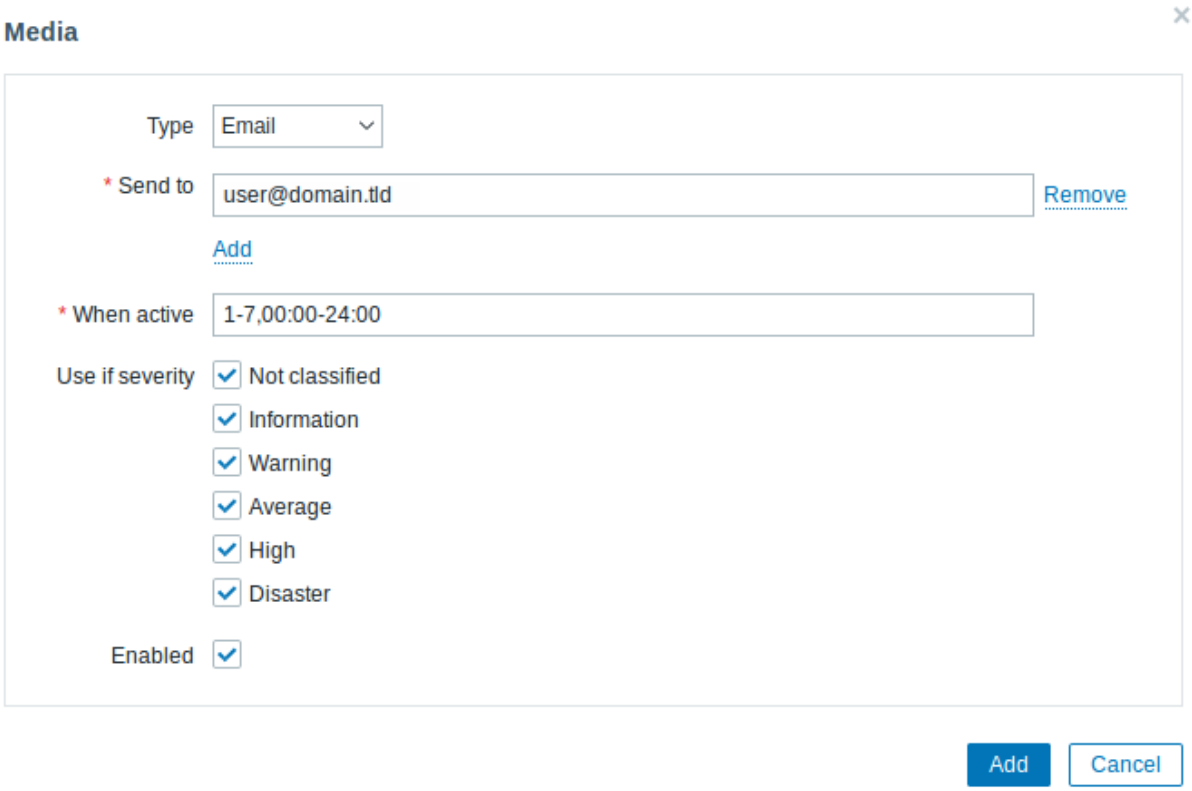
Ta czynność przeniesie Cię do pulpitu hosta (skonfigurowanego na poziomie szablonu) z najważniejszymi metrykami zebranymi z witryny internetowej.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemie z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika → Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami dla Odbierania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta w Zabbix frontend.

1. Otwórz konfigurację hosta swojej witryny internetowej w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok, na przykład, wcześniej skonfigurowanej wartości makra {$WEBSITE.DOMAIN} i ustaw nieprawidłową nazwę domeny (na przykład /git.zabbix.com/projects/ZBX/repos/zabbix/browse).

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Po chwili Zabbix wykryje problem „Failed to get JSON of the requested website", ponieważ nie będzie mógł połączyć się z określoną witryną internetową. Problem pojawi się w *Monitoring → Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem na poprzednią, aby rozwiązać problem i kontynuować monitorowanie witryny internetowej.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Pozycje przeglądarki - jak skonfigurować pozycje przeglądarki.
- Szablon Website by Browser - dodatkowe informacje o szablonie *Website by Browser*.

9 Monitorowanie certyfikatów stron internetowych za pomocą agent Zabbix 2 (pasywnie)

Wprowadzenie

Ten przewodnik zawiera kompleksowy przegląd sposobu konfiguracji i monitorowania certyfikatów SSL/TLS przy użyciu klucza `web.certificate.get` w Zabbix agent 2. Został opracowany, aby uprościć monitorowanie certyfikatów dla pojedynczych lub wielu witryn internetowych, zapewniając administratorom możliwość szybkiego identyfikowania potencjalnych problemów, takich jak wygasłe lub nieprawidłowe certyfikaty.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania certyfikatów stron internetowych. Jeśli szukasz opcji zaawansowanego dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Configuration w podręczniku Zabbix.

Wymagania wstępne

Przed przystąpieniem do tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix 2 zgodnie z instrukcjami dla używanego systemu operacyjnego. W tym samouczku założono, że zarówno serwer Zabbix, jak i agent są zainstalowane na tej samej maszynie; dlatego w konfiguracji użyto `127.0.0.1`.

Skonfiguruj Zabbix agent 2

1. Otwórz plik konfiguracyjny Zabbix agent (domyślna ścieżka: `/etc/zabbix/zabbix_agent2.conf`):

```
sudo vi /etc/zabbix/zabbix_agent2.conf
```

2. Ustaw parametr Server na 127.0.0.1, ponieważ agent i serwer działają na tej samej maszynie:

```
Server=127.0.0.1
```

3. Zapisz plik i uruchom ponownie usługę Zabbix agent 2:

```
sudo systemctl restart zabbix-agent2
```

4. Po zakończeniu instalacji i konfiguracji Zabbix agent 2 przetestuj jego dostępność za pomocą:

```
zabbix_get -s 127.0.0.1 -k web.certificate.get[<website_DNS_name>]
```

Zabbix agent 2 domyślnie zawiera wtyczkę WebCertificate, więc nie jest wymagana osobna instalacja ani konfiguracja.

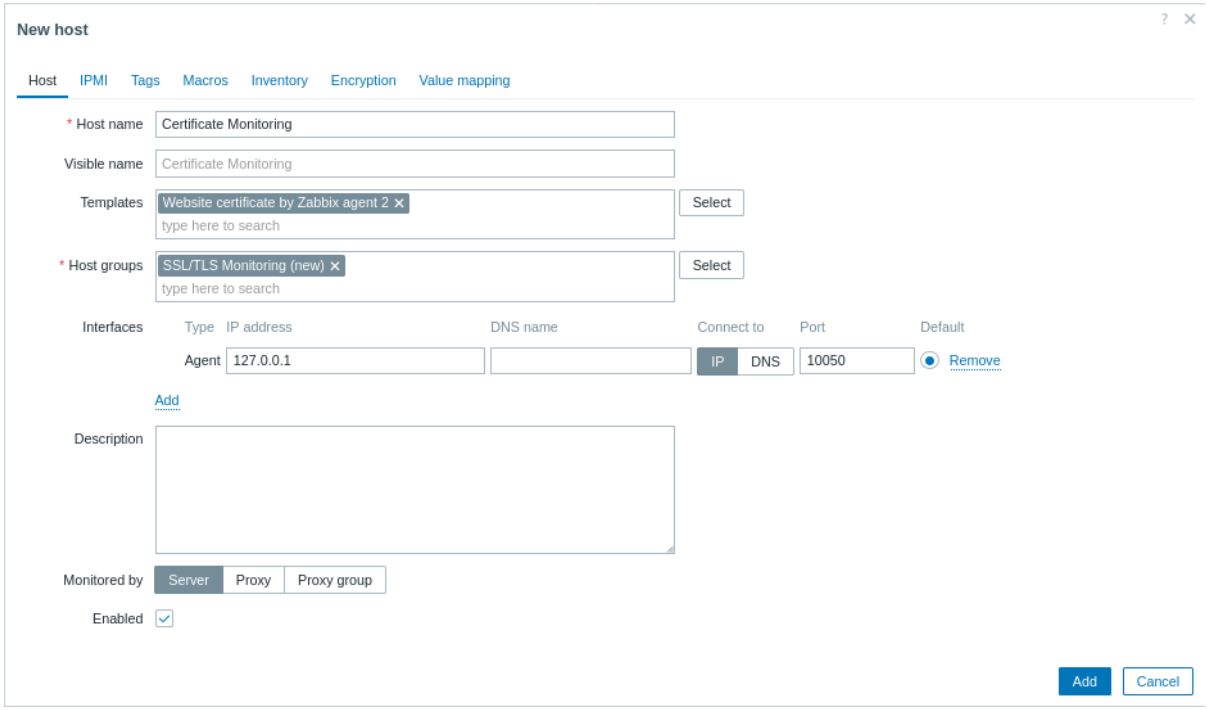
Konfiguracja frontendu Zabbix

1. Zaloguj się do frontendu Zabbix.

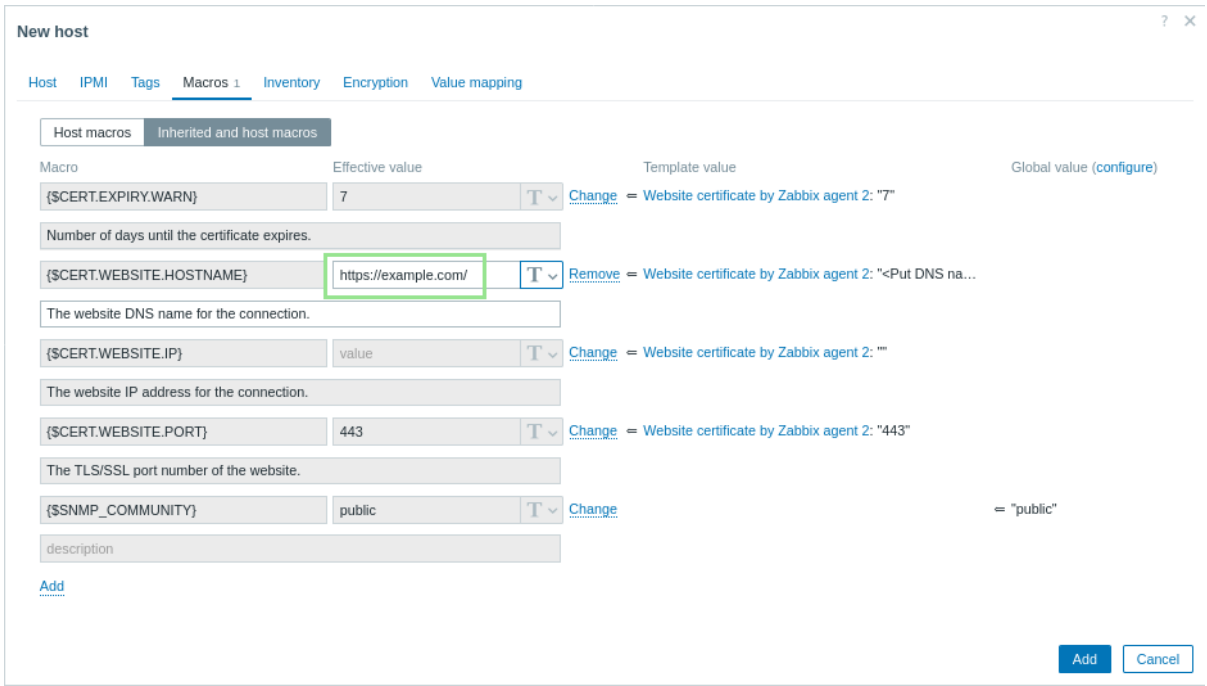
2. Przejdź do *Monitoring* > *Hosts*.

3. Kliknij istniejący host, na którym chcesz monitorować certyfikaty stron internetowych, lub utwórz host, jeśli to konieczne:

- W polu *Host name* wprowadź nazwę hosta (np. „Certificate Monitoring”).
- W polu *Templates* wpisz lub wybierz szablon „Website certificate by Zabbix agent 2”, który zostanie podłączony do hosta.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. „SSL/TLS Monitoring”).
- W polu *Interfaces* dodaj interfejs typu „Agent” i określ adres IP. W tym przykładzie użyto adresu „127.0.0.1”.



- Na karcie *Macros* przełącz się na *Inherited and host macros*, odszukaj następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - `{$CERT.WEBSITE.HOSTNAME}` — jako wartość wprowadź żądaną nazwę DNS strony internetowej.



4. Kliknij *Add*, aby dodać host.

Aby monitorować wiele stron internetowych, możesz podać rozdzieloną przecinkami listę ich nazw hostów w makrze `{$CERT.WEBSITE.HOSTN`. Opcjonalnie możesz również podać rozdzieloną przecinkami listę portów w makrze `{$CERT.WEBSITE.PORT}`, gdzie każdy port odpowiada odpowiedniej nazwie hosta zgodnie z kolejnością. Na przykład, aby monitorować example.com i example.org na porcie 8443:

- `{$CERT.WEBSITE.HOSTNAME}`: example.com,example.org
- `{$CERT.WEBSITE.PORT}`: 443,8443

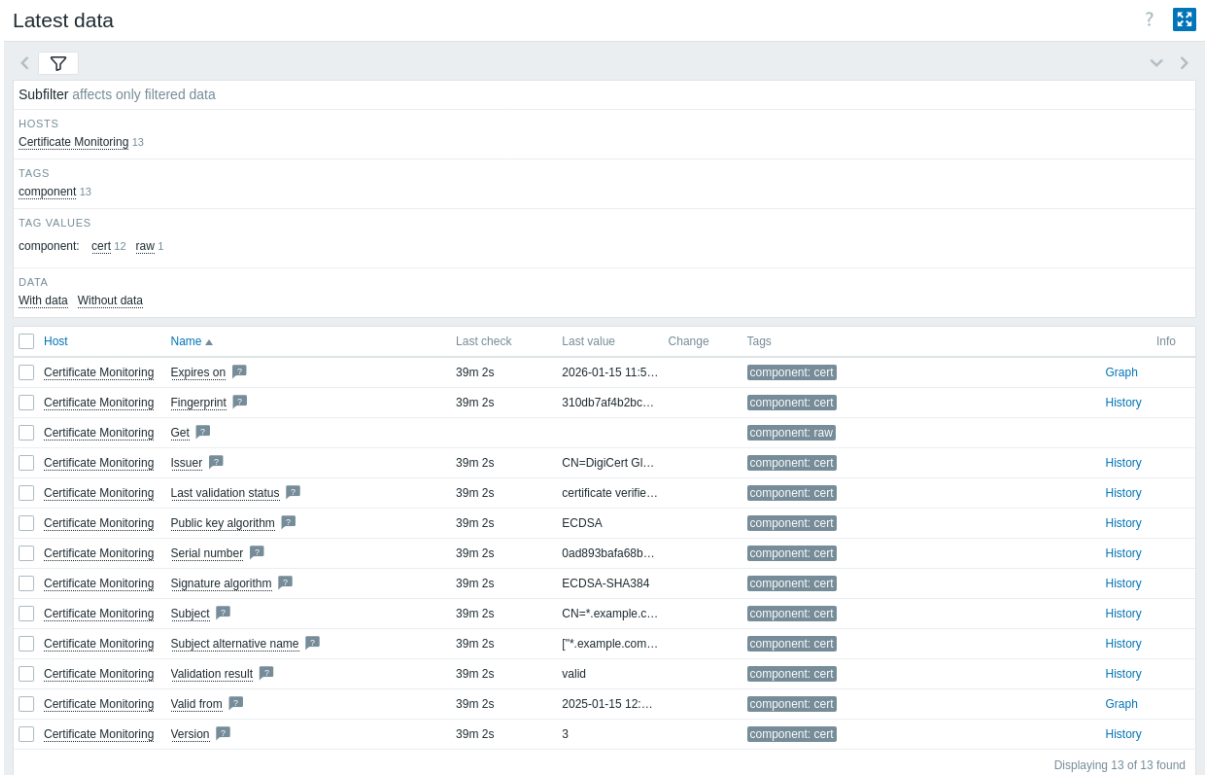
Dla każdej strony internetowej określonej w makrze `{$CERT.WEBSITE.HOSTNAME}` Zabbix utworzy odpowiedni zestaw pozycji i wyzwalaczy. Umożliwia to indywidualne monitorowanie i alertowanie dla certyfikatu SSL każdej strony internetowej.

Host	Name ▲	Last check	Last value
Certificate Monitoring	Get data	11m 16s	{"error":{"message…
Certificate Monitoring	[example.com]: Expires on	11m 6s	2026-01-15 11:59:…
Certificate Monitoring	[example.com]: Fingerprint	11m 6s	310db7af4b2bc90…
Certificate Monitoring	[example.com]: Get		
Certificate Monitoring	[example.com]: Issuer	11m 6s	CN=DigiCert Glob…
Certificate Monitoring	[example.com]: Last validation status	11m 6s	certificate verified …
Certificate Monitoring	[example.com]: Public key algorithm	11m 6s	ECDSA
Certificate Monitoring	[example.com]: Serial number	11m 6s	0ad893bafa68b0b…
Certificate Monitoring	[example.com]: Signature algorithm	11m 6s	ECDSA-SHA384
Certificate Monitoring	[example.com]: Subject	11m 6s	CN=*.example.co…
Certificate Monitoring	[example.com]: Subject alternative name	11m 6s	["*.example.com","…
Certificate Monitoring	[example.com]: Validation result	11m 6s	valid
Certificate Monitoring	[example.com]: Valid from	11m 6s	2025-01-15 12:00:…
Certificate Monitoring	[example.com]: Version	11m 6s	3
Certificate Monitoring	[example.org]: Expires on	10m 52s	2026-01-15 11:59:…
Certificate Monitoring	[example.org]: Fingerprint	10m 52s	3b451cfce915637…
Certificate Monitoring	[example.org]: Get		
Certificate Monitoring	[example.org]: Issuer	10m 52s	CN=DigiCert Glob…
Certificate Monitoring	[example.org]: Last validation status	10m 52s	certificate verified …
Certificate Monitoring	[example.org]: Public key algorithm	10m 52s	ECDSA
Certificate Monitoring	[example.org]: Serial number	10m 52s	0722a749b558476…
Certificate Monitoring	[example.org]: Signature algorithm	10m 52s	ECDSA-SHA384
Certificate Monitoring	[example.org]: Subject	10m 52s	CN=*.example.org…
Certificate Monitoring	[example.org]: Subject alternative name	10m 52s	["*.example.org","e…
Certificate Monitoring	[example.org]: Validation result	10m 52s	valid
Certificate Monitoring	[example.org]: Valid from	10m 52s	2025-01-15 12:00:…

Wyświetlanie zebranych metryk

Gratulacje! Na tym etapie Zabbix monitoruje już wybrany certyfikat WWW.

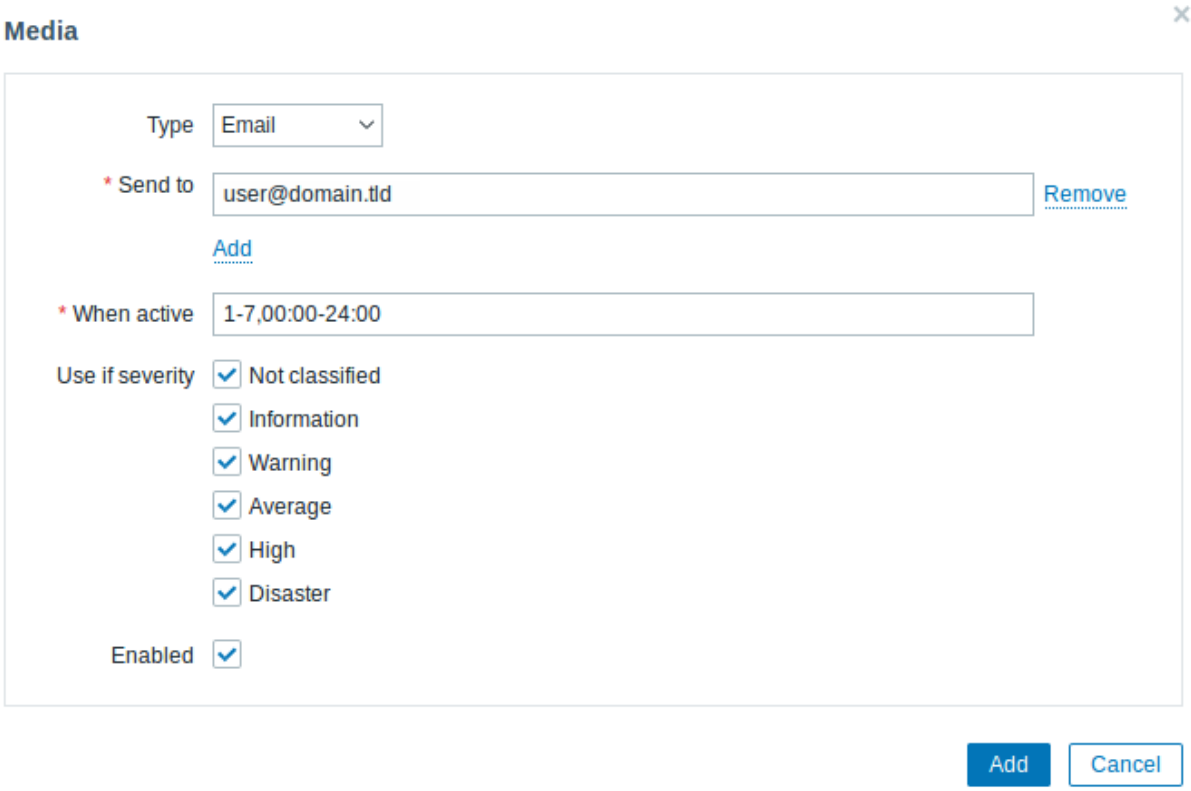
Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosts* i kliknij *Latest data* obok hosta, aby wyświetlić na liście wszystkie najnowsze zebrane metryki, takie jak data wygaśnięcia, wystawca i podmiot.



Konfiguracja alertów o problemach

Zabbix może powiadamiać o problemach z infrastrukturą. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przejdź do zakładki *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z instrukcjami w przewodniku Odbieranie powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta w Zabbix frontend.

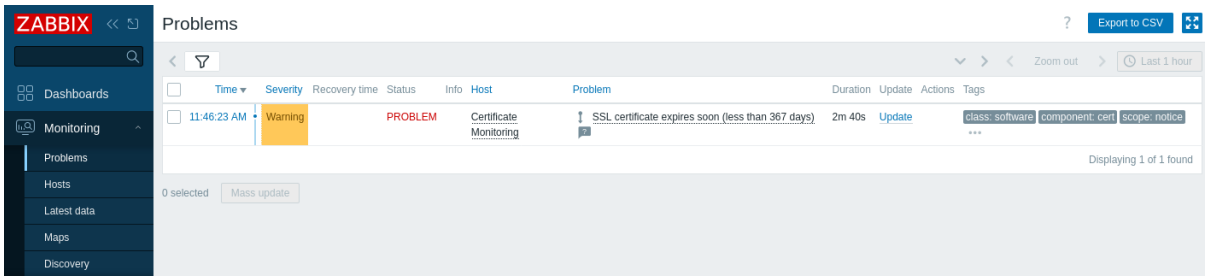
1. Otwórz konfigurację swojego hosta „Certificate Monitoring” w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok wcześniej skonfigurowanej wartości makra `{$CERT.EXPIRY.WARN}` i ustaw bardzo dużą liczbę dni (więcej niż 365 dni powinno wystarczyć), aby otrzymać ostrzeżenie przed wygaśnięciem certyfikatu.

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Za chwilę Zabbix wykryje problem „SSL certificate expires soon” wraz z liczbą dni pozostałych do wygaśnięcia. Problem pojawi się w *Monitoring > Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem na poprzednią, aby rozwiązać problem i kontynuować monitorowanie wartości certyfikatu.

Zobacz także

- Zabbix agent 2 - zawiera listę kluczy pozycji.
- Szablon *Website certificate by Zabbix agent 2* - dodatkowe informacje o szablonie *Website certificate by Zabbix agent 2*.
- Szablon *Website certificate by Zabbix agent 2 active* - dodatkowe informacje o szablonie *Website certificate by Zabbix agent 2 active*.
- Monitorowanie witryn internetowych za pomocą pozycji Browser - jak rozpocząć podstawowe monitorowanie witryn internetowych za pomocą pozycji Browser.

10 Monitorowanie przełącznika sieciowego lub routera za pomocą Zabbix

Wprowadzenie Ten przewodnik przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania przełącznika sieciowego lub routera za pomocą Zabbix. Jako przykład użyto routera Cisco, ale procedura ma zastosowanie do dowolnego urządzenia sieciowego z obsługą SNMP.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix oraz administratorów sieci, którzy chcą szybko włączyć podstawowe monitorowanie urządzeń sieciowych. Jeśli potrzebujesz zaawansowanej personalizacji lub opcji rozbudowanej konfiguracji, zapoznaj się ze stroną SNMP agent lub sekcją Configuration podręcznika Zabbix.

Wymagania wstępne

Przed kontynuowaniem upewnij się, że masz:

- Zainstalowany serwer Zabbix i frontend Zabbix: zainstaluj zgodnie z instrukcjami dla swojego systemu operacyjnego (zobacz Installation from packages oraz Web interface installation).
- Zainstalowany agent Zabbix, jeśli monitorujesz lokalne metryki sieciowe.
- Urządzenie z obsługą SNMP: przełącznik sieciowy lub router (na przykład router Cisco) z włączonym SNMP.
- Zainstalowane pliki MIB: instalacja plików MIB umożliwia Zabbix tłumaczenie numerycznych OID na czytelne dla człowieka nazwy i opisy. Bez odpowiedniej obsługi MIB możesz widzieć tylko wartości numeryczne, co utrudnia konfigurację pozycji i rozwiązywanie problemów.

Aby zainstalować pliki MIB w Ubuntu:

1. Zainstaluj pakiet pobierający MIB:

```
sudo apt-get update
sudo apt-get install snmp-mibs-downloader
```

Jeśli musisz dodać MIB specyficzne dla producenta (np. Cisco, Juniper), umieść je w odpowiednim katalogu MIB:

- W systemach opartych na Linuksie typowe lokalizacje to /usr/share/snmp/mibs/ lub /usr/local/share/snmp/mibs/.
- W instalacjach Zabbix pliki MIB mogą być przechowywane w /var/lib/zabbix/mibs/.

Upewnij się, że zmienna środowiskowa MIBDIRS lub plik snmp.conf zawiera poprawną ścieżkę.

Aby sprawdzić, czy system rozpoznaje nowe MIB, użyj:

```
snmptranslate -IR -On <MIB-NAME>::<object>
```

Szczegółowe instrukcje znajdziesz w dokumentacji biblioteki SNMP:

- Cisco MIBs
- Juniper MIBs

2. Edytuj `/etc/snmp/snmp.conf` i zakomentuj linię zaczynającą się od `mibs :`, aby umożliwić systemowi wczytanie wszystkich dostępnych MIB.

3. Zweryfikuj działanie, uruchamiając `snmpwalk` (na przykład `snmpwalk -v 2c -c <your_community_string> <device_IP>`) i sprawdź, czy OID są wyświetlane z opisowymi nazwami.

W zależności od środowiska niektóre kroki w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku z systemem Ubuntu oraz monitorowanym urządzeniu sieciowym Cisco Catalyst 3750V2-24FS.

Zakłada się, że urządzenie sieciowe jest już fizycznie zainstalowane i podłączone.

Skonfiguruj urządzenie sieciowe (przykład routera Cisco) Aby monitorowanie przez SNMP było możliwe, musisz skonfigurować urządzenie sieciowe tak, aby zezwalało na zapytania SNMP. Poniższy przykład dotyczy SNMPv2 i nie uwzględnia istniejących ustawień. Uwaga: zastosowanie tych poleceń może nadpisać bieżące konfiguracje SNMP.

W przypadku routera Cisco konfiguracja zazwyczaj obejmuje kroki opisane poniżej.

Przykład SNMPv2 1. Włącz SNMP i ustaw ciąg community.

Zaloguj się do konsoli routera Cisco i przejdź do trybu konfiguracji:

```
configure terminal
```

Następnie włącz SNMP, podając ciąg community tylko do odczytu. Na przykład:

```
snmp-server community <your_community_string> RO
```

Zastąp `<your_community_string>` swoim bezpiecznym ciągiem community. Uwaga: opcja RO (Read-Only) umożliwia SNMP pobieranie danych z urządzenia, ale uniemożliwia wprowadzanie jakichkolwiek zmian konfiguracyjnych.

Ze względów bezpieczeństwa zaleca się ograniczenie dostępu SNMP wyłącznie do niezbędnych urządzeń. Aby uzyskać dalsze wskazówki dotyczące konfigurowania list kontroli dostępu (ACL), zapoznaj się z oficjalną dokumentacją Cisco.

2. Zapisz konfigurację.

Zapisz zmiany, aby ustawienia SNMP zostały zachowane po ponownym uruchomieniu:

```
write memory
```

Przykład SNMPv3 SNMPv3 zapewnia zwiększone bezpieczeństwo dzięki uwierzytelnianiu i szyfrowaniu. Jego konfiguracja jest bezpieczniejsza niż w przypadku SNMPv2 i powinna zostać zweryfikowana z dokumentacją właściwą dla danego urządzenia.

1. Utwórz grupę SNMP.

Skonfiguruj grupę SNMPv3 z włączoną prywatnością (szyfrowaniem):

```
configure terminal
snmp-server group <your_group> v3 priv
```

2. Utwórz użytkownika SNMP.

Dodaj użytkownika SNMPv3 z uwierzytelnianiem i prywatnością. Zastąp symbole zastępcze żądanymi wartościami:

```
snmp-server user <your_user> <your_group> v3 auth md5 <auth_password> priv aes 128 <priv_password>
```

3. Zapisz konfigurację:

```
write memory
```

Aby uzyskać więcej informacji lub instrukcje dotyczące konkretnego modelu, możesz skorzystać z zewnętrznych samouczków konfiguracji Cisco SNMP. Ten przewodnik przedstawia jednak podstawowe kroki umożliwiające włączenie monitorowania SNMP.

Konfiguracja frontendu Zabbix

Utwórz host w frontendzie Zabbix 1. Zaloguj się do frontendu Zabbix.

2. Dodaj nowy host.

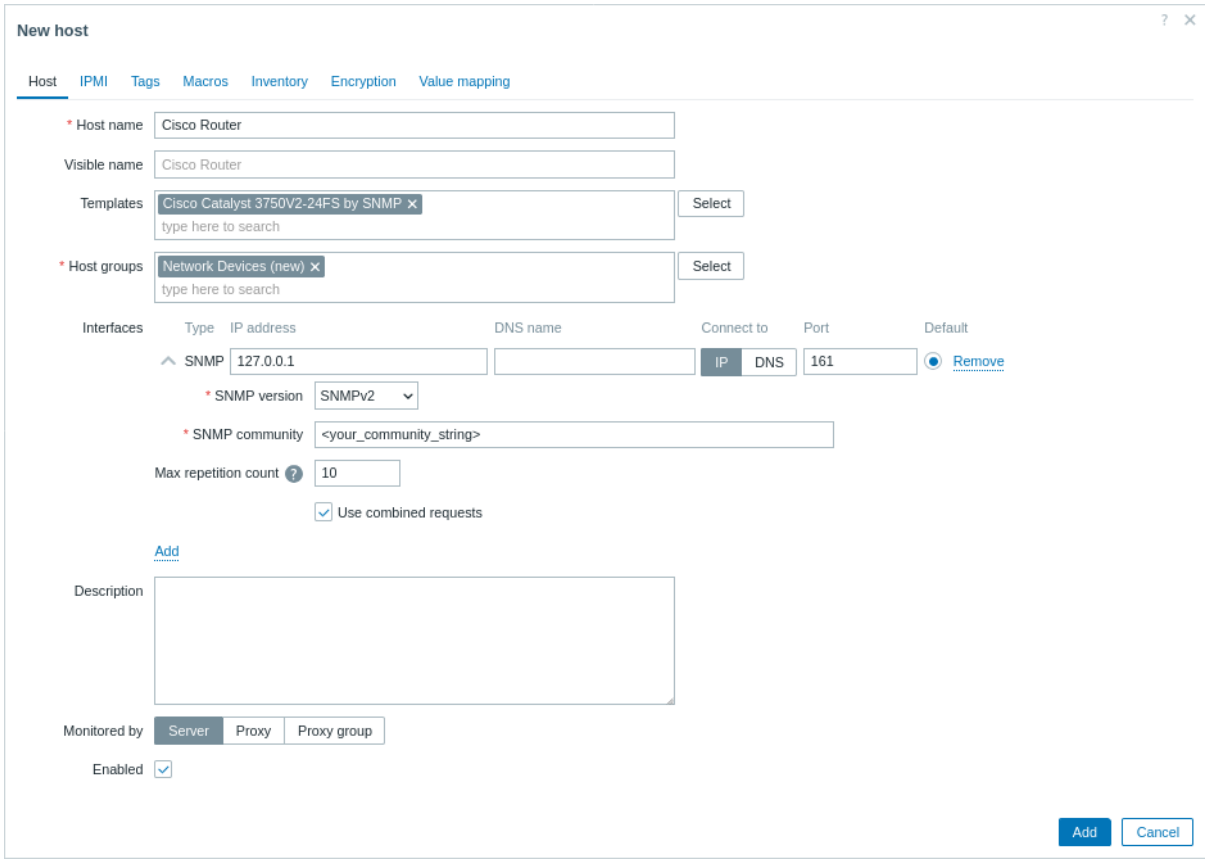
Przejdź do *Data collection* > *Hosts* i kliknij *Create host*.

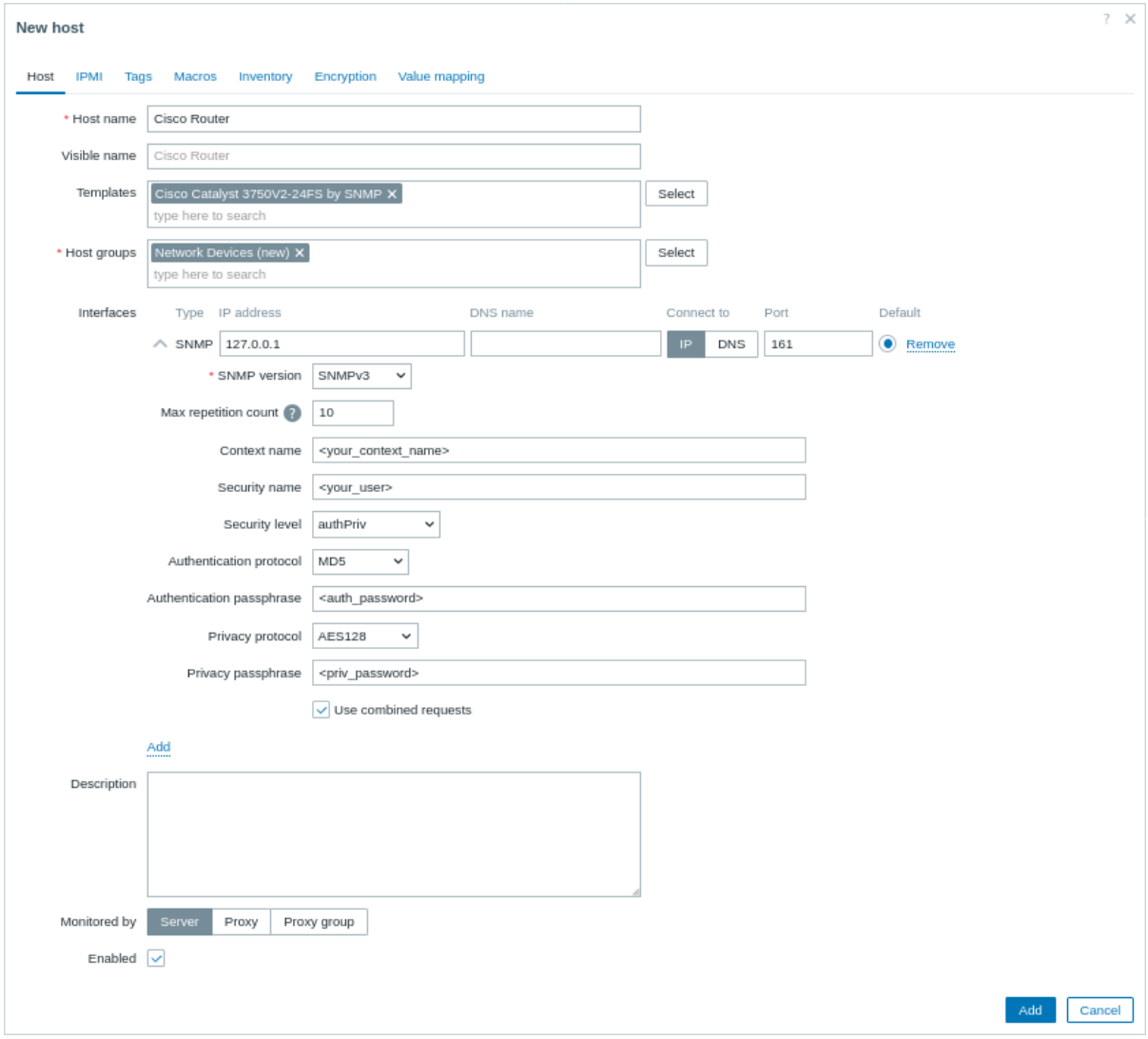
- *Host name*: wprowadź nazwę swojego urządzenia (np. „Cisco Router”).
- Host groups: wybierz istniejącą grupę lub utwórz nową grupę, taką jak „Network Devices”.
- Interfaces:
  - Kliknij *Add* w sekcji Interfaces.
  - Wybierz *SNMP* jako typ interfejsu.
  - Wprowadź adres IP lub nazwę DNS swojego routera Cisco.
  - Ustaw domyślny port SNMP (zwykle 161).
  - Użyj menu rozwijanego, aby wybrać odpowiednią wersję SNMP (np. SNMPv2).
  - Dla SNMPv1/v2 wprowadź community string w polu *SNMP community*. W przypadku SNMPv3 zostaniesz poproszony o podanie dodatkowych poświadczeń (*Context name*, *Security name* i *Security level* itp.).

3. Podłącz szablony

W polu *Templates* wybierz szablon SNMP, który najlepiej pasuje do Twojego urządzenia. Zabbix udostępnia szereg gotowych szablonów SNMP dla wielu rodzin urządzeń. Na przykład, jeśli monitorujesz urządzenie Cisco, wybierz szablon odpowiadający systemowi operacyjnemu lub modelowi Twojego urządzenia (taki jak Cisco IOS SNMP lub Cisco Catalyst 3750<model urządzenia> SNMP).

4. Kliknij *Add*, aby zapisać host.

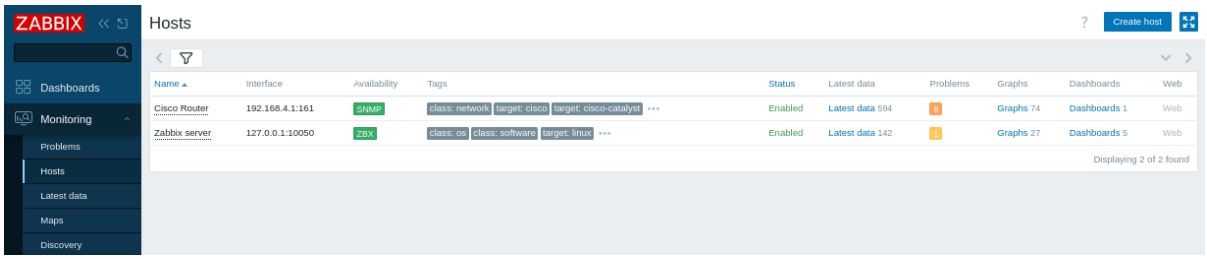




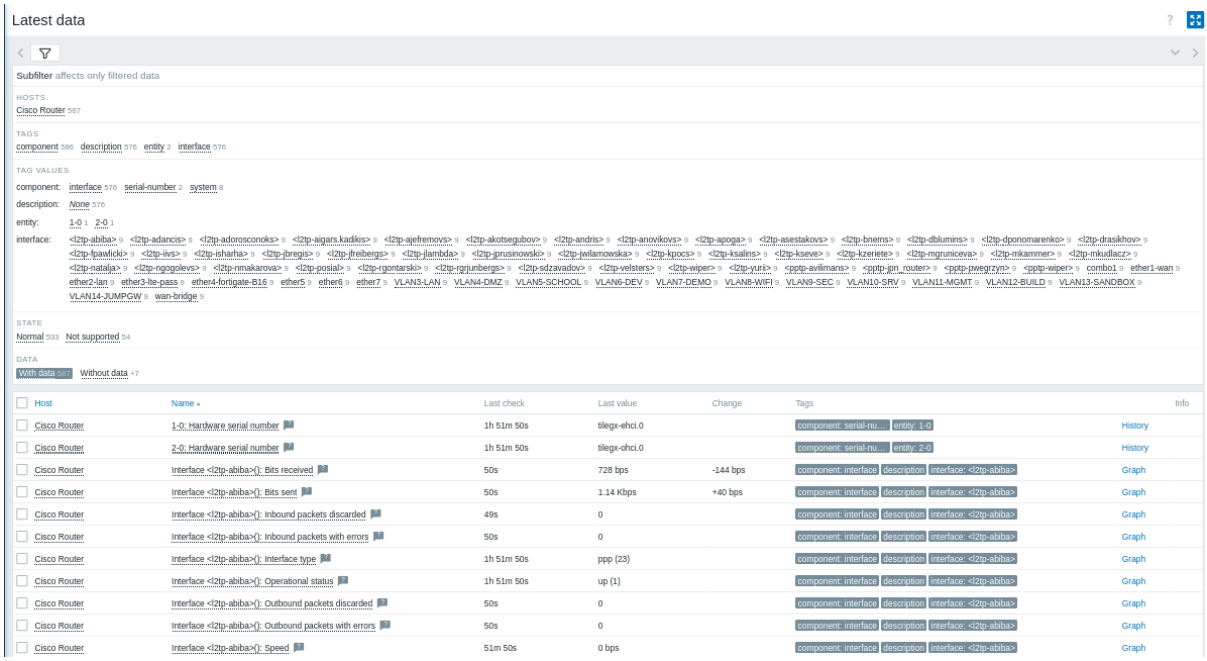
Wyświetlanie zebranych metryk Gratulacje! Zabbix jest teraz skonfigurowany do monitorowania Twojego urządzenia sieciowego.

Najnowsze dane:

- Przejdź do Monitoring > Latest data w frontendzie Zabbix.



- Wybierz host ”Cisco Router” (lub wykryte hosty), aby wyświetlić metryki, takie jak czas działania sprzętu i sieci, utrata pakietów ICMP, ping, czas odpowiedzi itp.



- Wykresy i ekrany:

Aby zwizualizować dane wydajności, kliknij *Graphs* obok pozycji SNMP, aby zobaczyć szczegółowe metryki.

W następnym kroku możesz:

- Dodać niestandardowe pozycje SNMP, aby monitorować dodatkowe metryki.
- Skonfigurować alerty problemów, aby otrzymywać powiadomienia o potencjalnych problemach.

Tworzenie pozycji SNMP Gdy host jest już skonfigurowany, możesz utworzyć pozycje do monitorowania określonych metryk. Uwaga: ten krok jest opcjonalny, jeśli używasz szablonu, ponieważ szablony zawierają już domyślne zestawy pozycji.

1. Zidentyfikuj OID SNMP:

Użyj polecenia `snmpwalk`, aby wyświetlić listę dostępnych OID-ów na swoim urządzeniu. Na przykład:

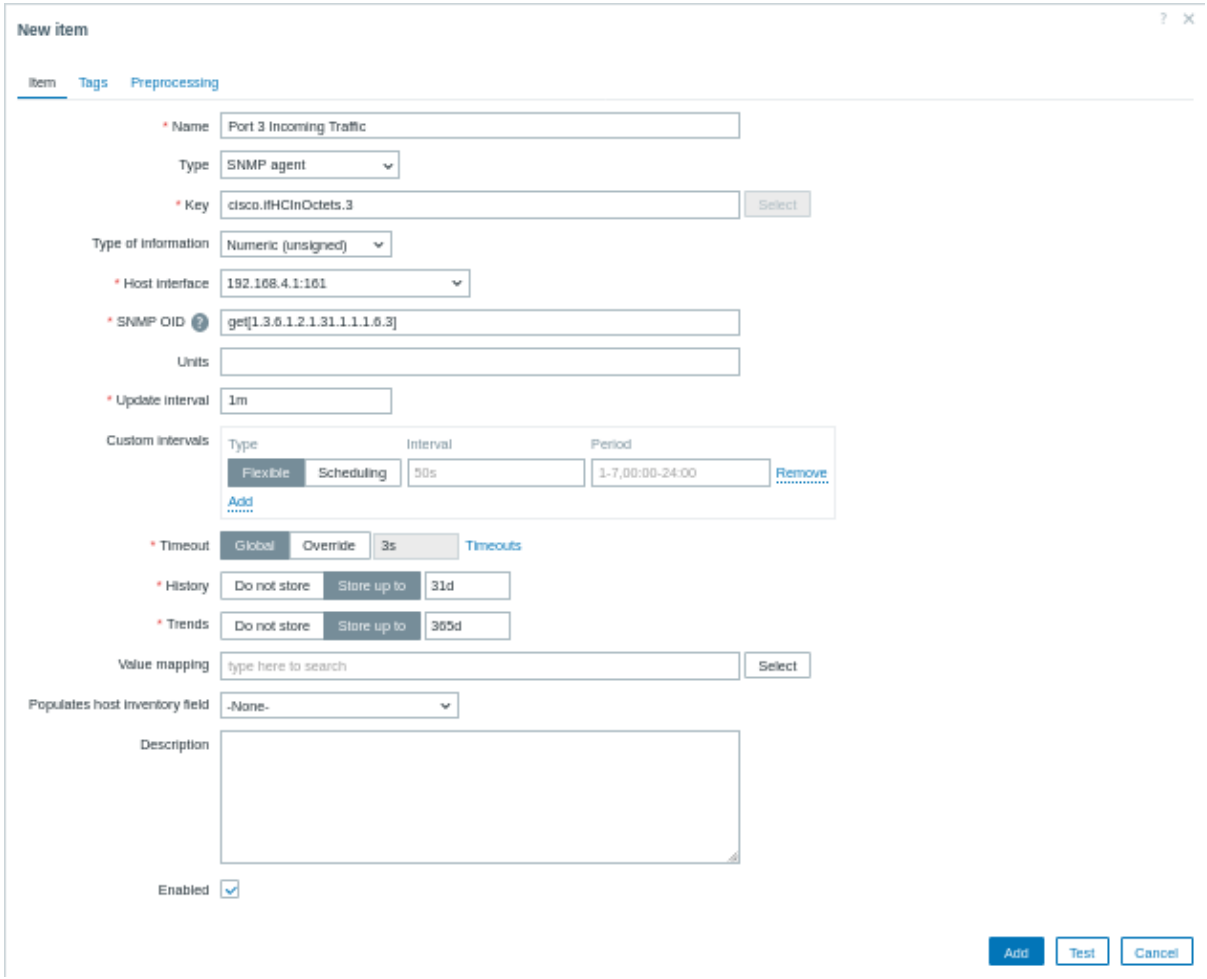
```
snmpwalk -v 2c -c <your_community_string> <device_IP> .
```

Znajdź OID dla metryki, którą chcesz monitorować (na przykład IF-MIB::ifHCInOctets.3 dla ruchu przychodzącego na porcie 3). Aby uzyskać numeryczny OID, możesz użyć:

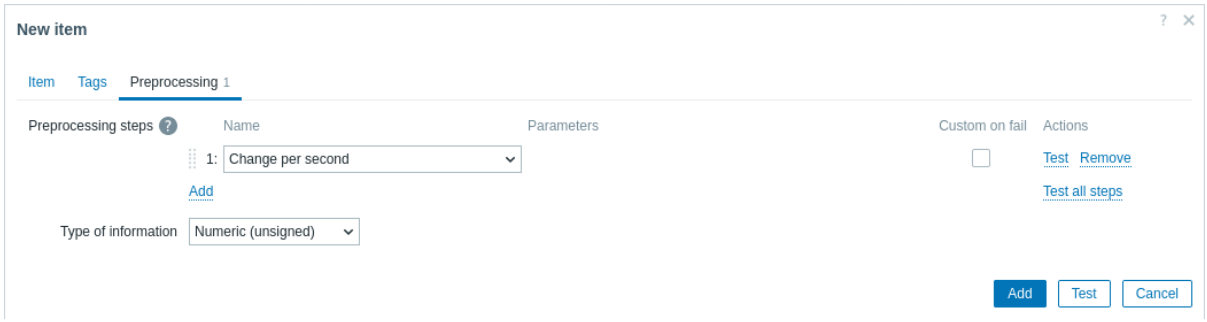
```
snmpget -v 2c -c <your_community_string> -On <device_IP> IF-MIB::ifHCInOctets.3
```

2. Utwórz pozycję SNMP:

- Przejdź do *Data collection* > *Hosts*, kliknij kartę *Items* dla swojego hosta SNMP, a następnie kliknij *Create item*.
- *Name*: wprowadź opisową nazwę (np. „Ruch przychodzący na porcie 3”).
- *Type*: wybierz *SNMP agent*.
- *Key*: podaj znaczący klucz (np. `cisco.ifHCInOctets.3`).
- *Host interface*: upewnij się, że wybrany jest interfejs SNMP.
- *SNMP OID*: wprowadź OID, używając jednego z obsługiwanych formatów, na przykład:
  - `get[1.3.6.1.2.1.31.1.1.1.6.3]` dla pojedynczej wartości;
  - `walk[1.3.6.1.2.1.31.1.1.1.6.3]` do asynchronicznego pobrania poddrzewa wartości.



- *Preprocessing* (w razie potrzeby): jeśli pozycja zwraca licznik skumulowany (taki jak ruch na interfejsie), przejdź do karty *Preprocessing* i dodaj krok przetwarzania wstępnego, taki jak „Zmiana na sekundę”, aby obliczyć szybkość.



Aby pobrać wiele wartości w jednej transakcji SNMP, możesz podać kilka OID-ów, używając składni `walk[OID1,OID2,...]`.

Tłumaczenie OID-ów między nazwami numerycznymi a MIB Podczas pracy z SNMP może być konieczna konwersja między numerycznymi OID-ami a odpowiadającymi im nazwami MIB. Takie tłumaczenie ułatwia identyfikację metryk i rozwiązywanie problemów.

- Tłumaczenie nazwy MIB na numeryczny OID: użyj polecenia `snmptranslate` z opcją `-On`. Na przykład, aby przetłumaczyć nazwę MIB `IF-MIB::ifHCInOctets.3` na jej numeryczny OID, uruchom:

```
snmptranslate -On IF-MIB::ifHCInOctets.3
```

To polecenie może zwrócić:

```
.1.3.6.1.2.1.31.1.1.1.6.3
```

- Tłumaczenie numerycznego OID-a na jego nazwę MIB: użyj polecenia `snmptranslate` z opcją `-IR` (lub `-m ALL`), aby odwrócić tłumaczenie. Na przykład, aby przetłumaczyć numeryczny OID `.1.3.6.1.2.1.31.1.1.1.6.3` z powrotem na jego nazwę MIB, uruchom:

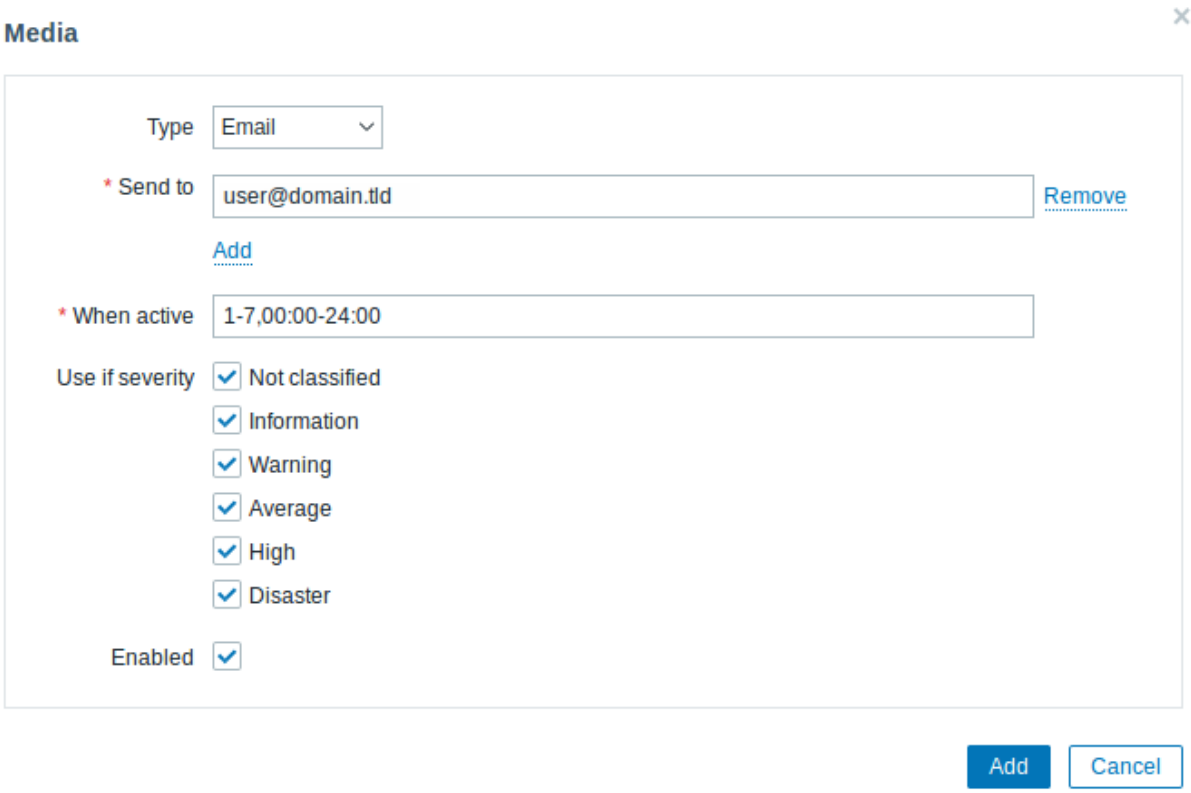
```
snmptranslate -IR -On .1.3.6.1.2.1.31.1.1.1.6.3
```

To polecenie może zwrócić:

```
IF-MIB::ifHCInOctets.3
```

Skonfiguruj alerty o problemach Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.

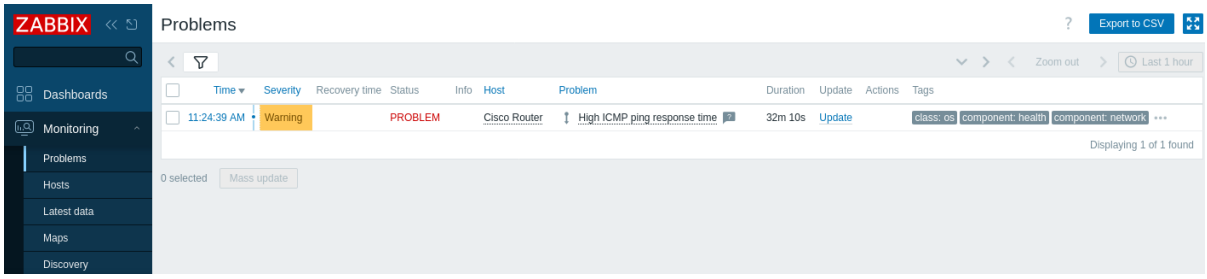


2. Postępuj zgodnie z instrukcjami w przewodniku Odbieranie powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mailem.

Przetestuj swoją konfigurację Aby upewnić się, że Zabbix prawidłowo wykrywa problemy z wydajnością sieci, zasymuluj rzeczywisty problem, zwiększając próg czasu odpowiedzi ICMP ping.

1. Otwórz konfigurację swojego hosta „Cisco Router” w Zabbix.
2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.
3. Znajdź makro `{$ICMP_RESPONSE_TIME_WARN}` (lub podobne makro progu czasu odpowiedzi).
4. Ustaw bardzo niską wartość (np. 0.001), aby wyzwolić alert, gdy odpowiedź ping przekroczy tę wartość.
5. Kliknij *Update*, aby zastosować zmiany.
6. Poczekaj chwilę, aż Zabbix wykryje zasymulowany problem.
7. Przejdź do *Monitoring > Problems*, aby sprawdzić, czy pojawił się alert (np. „High ICMP ping response time”).



Jeśli alerty są skonfigurowane, powinieneś również otrzymać powiadomienie o problemie.

8. Przywróć pierwotną wartość makra i kliknij *Update*, aby zapisać zmiany.
9. Potwierdź, że problem został rozwiązany i zniknął z sekcji *Problems*.

Rozwiązywanie problemów z monitorowaniem SNMP Jeśli zauważysz, że ikona SNMP w frontendzie Zabbix jest CZERWONA lub nie są zbierane żadne dane, spróbuj wykonać następujące kroki:

1. Sprawdź łączność SNMP.

Dla SNMPv2 uruchom następujące polecenie z serwera Zabbix:

```
snmpwalk -v 2c -c <community_string> <device_IP> .
```

To polecenie weryfikuje, czy urządzenie odpowiada na zapytania SNMP.

Dla SNMPv3 podaj odpowiednie poświadczenia SNMPv3:

```
snmpwalk -v3 -u <your_user> -l authPriv -a MD5 -A <auth_password> -x AES -X <priv_password> <device_IP> .
```

To weryfikuje, czy poświadczenia SNMPv3 są poprawne oraz czy urządzenie odpowiada w sposób bezpieczny.

2. Upewnij się, że pliki MIB są zainstalowane i włączone zgodnie z opisem w sekcji wymagania wstępne. Aby to potwierdzić, następujące polecenie nie powinno zwracać błędu podczas odpytywania urządzenia sieciowego:

```
snmpwalk -v 2c -c <your_community_string> <device_IP> ifInOctets
```

Powinno to zwrócić przetłumaczone OID bez błędów.

3. Potwierdź, że wersja SNMP i poświadczenia skonfigurowane w Zabbix odpowiadają tym ustawionym na urządzeniu. Na przykład sprawdź ustawienia SNMP w konfiguracji hosta Zabbix i porównaj je z konfiguracją urządzenia. Na urządzeniu Cisco możesz sprawdzić ustawienia SNMP, uruchamiając:

```
show running-config | include snmp
```

To zapewnia, że community string (dla SNMPv2) lub dane użytkownika SNMPv3 są poprawne.

4. Sprawdź, czy SNMP jest poprawnie włączone na urządzeniu sieciowym. Na routerze Cisco zaloguj się do konsoli i uruchom:

```
show running-config | include snmp
```

To polecenie wyświetla aktywną konfigurację SNMP i pomaga potwierdzić, że SNMP jest poprawnie skonfigurowane.

5. Upewnij się, że żadne zapory sieciowe ani problemy sieciowe nie blokują ruchu SNMP (zwykle na porcie 161) między serwerem Zabbix a urządzeniem. Możesz przetestować łączność za pomocą:

nc -zv <device_IP> 161

`nc -zv` sprawdza, czy port 161 jest otwarty i nasłuchuje na urządzeniu.

Dodatkowo, jeśli używasz UFW w Ubuntu, sprawdź stan zapory:

```
sudo ufw status
```

Lub, w przypadku iptables:

```
sudo iptables -L -n
```

6. Przejrzyj pliki dziennika serwera Zabbix pod kątem błędów związanych z SNMP, aby ułatwić zlokalizowanie problemu:

```
tail -f /tmp/zabbix_server.log
```

`tail -f` pozwala monitorować aktualizacje dziennika w czasie rzeczywistym.

Zobacz także:

- Tworzenie pozycji - dowiedz się, jak dodawać dodatkowe metryki.
- Agent SNMP - dodatkowe informacje o monitorowaniu SNMP w Zabbix.
- Standaryzowane szablony dla urządzeń sieciowych - informacje o dostępnych szablonach SNMP.
- Odkrywanie OID-ów SNMP - dodatkowe informacje o wykrywaniu SNMP na przełączniku.
- Konfigurowanie reguły wykrywania sieci - dodatkowe informacje o tym, jak skonfigurować regułę wykrywania sieci używaną przez Zabbix do wykrywania hostów i usług.

11 Monitorowanie dziennika zdarzeń Windows przy użyciu aktywnych kontroli

Wprowadzenie

Ten przewodnik wyjaśnia, jak monitorować dzienniki zdarzeń systemu Windows w Zabbix przy użyciu aktywnych kontroli. Korzystając z kluczy pozycji specyficznych dla systemu Windows w Zabbix, można zbierać i analizować krytyczne zdarzenia (takie jak nieudane próby logowania, błędy systemowe itp.) w czasie rzeczywistym.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix oraz administratorów sieci, którzy chcą monitorować dzienniki zdarzeń systemu Windows. Informacje o zaawansowanych opcjach konfiguracji można znaleźć w dokumentacji kluczy pozycji specyficznych dla systemu Windows.

Wymagania wstępne

Przed kontynuowaniem należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego. Należy również pobrać i zainstalować agent Zabbix na komputerze z systemem Windows, który ma być monitorowany.

Skonfiguruj Zabbix agent do monitorowania dziennika zdarzeń systemu Windows

1. Otwórz plik `zabbix_agentd.conf` (domyślna ścieżka `C:\Program Files\Zabbix Agent\zabbix_agentd.conf`) na swoim hoście Windows i upewnij się, że parametr ServerActive jest ustawiony na adres IP twojego serwera Zabbix, a parametr Hostname odpowiada nazwie hosta, która zostanie zdefiniowana w frontend Zabbix. Umożliwia to agentowi pobieranie aktywnych kontroli dla swojego hosta od wskazanego serwera Zabbix. Na przykład:

```
ServerActive=192.0.2.1
Hostname=MyWindowsHost
```

2. Uruchom ponownie usługę Zabbix agent, aby zastosować zmiany:

```
net stop "Zabbix Agent" && net start "Zabbix Agent"
```

3. Sprawdź, czy host Windows działa:

- Upewnij się, że usługa Zabbix agent działa na hoście Windows.
- Sprawdź, czy host Windows może połączyć się z serwerem Zabbix na porcie 10051. Aby przetestować łączność z hosta Windows, otwórz PowerShell i uruchom następujące polecenie:

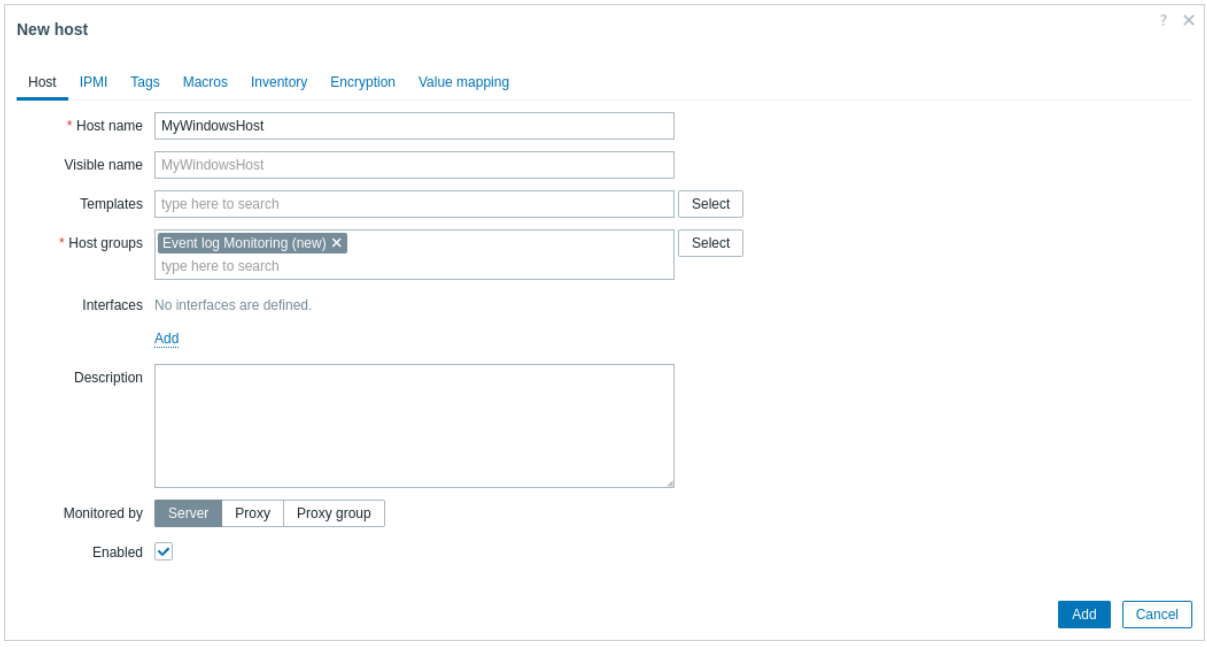
```
Test-NetConnection -ComputerName <Zabbix-server-IP> -Port 10051
```

Konfiguracja frontend Zabbix

1. Przejdź do *Data collection* > *Hosts* i utwórz hosta:

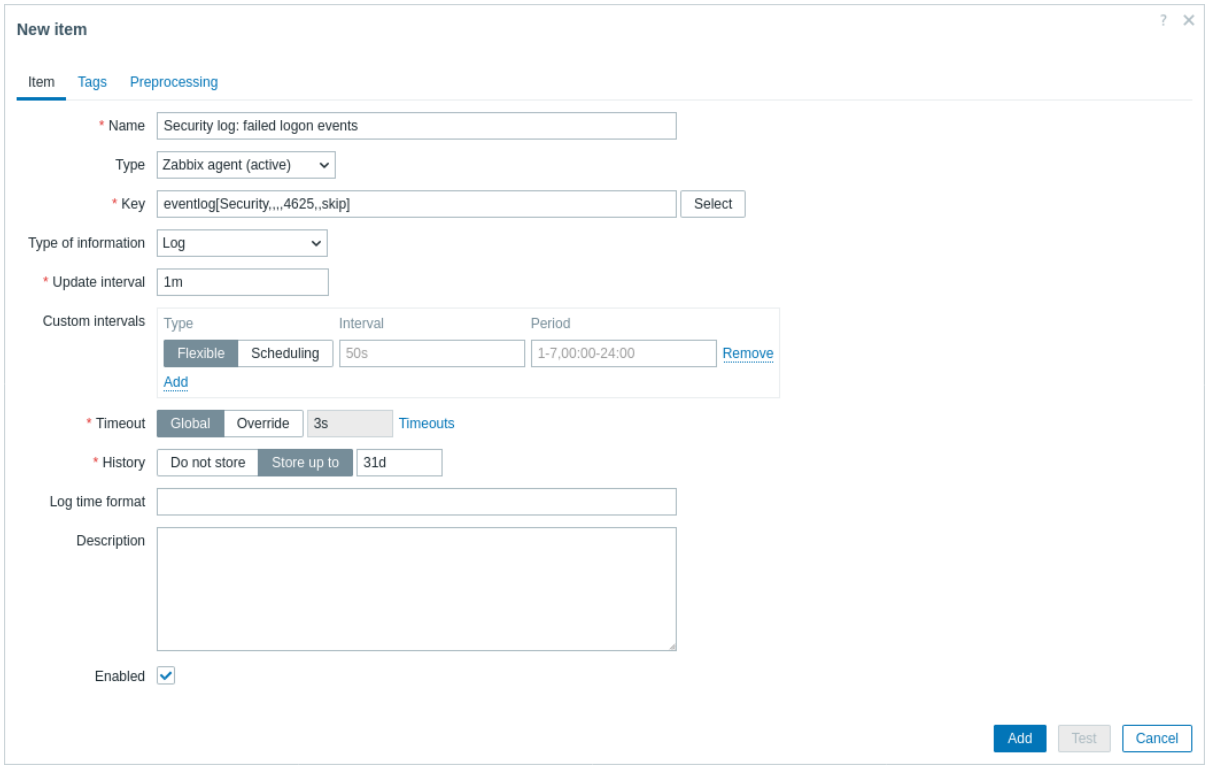
- W polu *Host name* wprowadź nazwę hosta (np. „MyWindowsHost”).
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. „Event log Monitoring”).
- Naciśnij *Add*, aby zapisać skonfigurowanego hosta.

Note:
W polu *Templates* możesz dodać szablon „Windows by Zabbix agent active”, aby ułatwić rozwiązywanie problemów przez obserwowanie, czy inne aktywne pozycje na tym samym hoście są aktualizowane.



2. Utwórz nową pozycję z następującymi parametrami:

- W polu *Name* wprowadź opisową nazwę pozycji (np. „Security log: failed logon events”).
- Z listy rozwijanej *Type* wybierz „Zabbix agent (active)” (wymagane do monitorowania dziennika zdarzeń).
- W polu *Key* użyj klucza pozycji eventlog. Na przykład, aby monitorować nieudane próby logowania (Event ID: 4625) w dzienniku Security i ignorować wpisy starsze niż ostatnie sprawdzenie pozycji (przy użyciu parametru `skip`), wprowadź następujący klucz pozycji: `eventlog[Security,,,,4625,,skip]`
- Z listy rozwijanej *Type of information* wybierz „Log”.



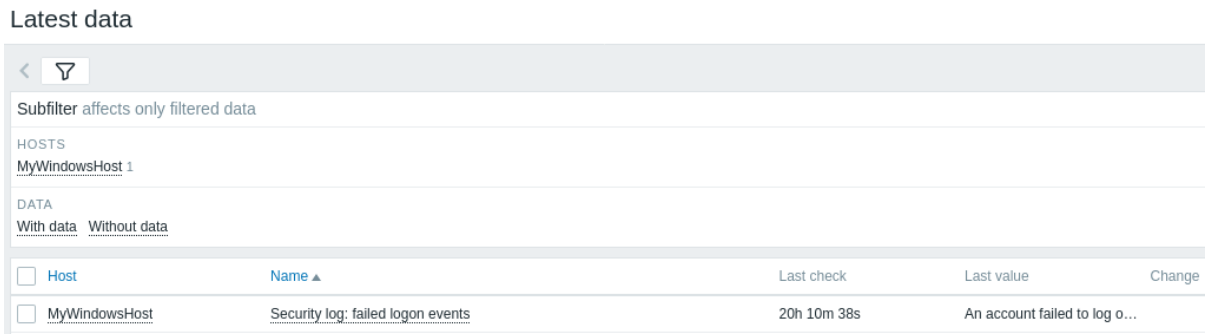
3. Kliknij *Add*, aby zapisać pozycję.

Testowanie i przegląd zebranych metryk

Gratulacje! Zabbix jest teraz skonfigurowany do zbierania dzienników zdarzeń systemu Windows. Aby sprawdzić, czy dzienniki zdarzeń są zbierane, możesz przetestować pozycję „Security log: failed logon events”, wylogowując się ze swojego konta Windows i próbując zalogować się przy użyciu nieprawidłowych poświadczeń.

Następnie wyświetl zebrane dzienniki w Zabbix frontend:

1. Przejdź do *Monitoring* > *Latest data* w Zabbix frontend.



2. Odfiltruj dane według hosta „MyWindowsHost” w polu *Name*.

3. Kliknij *History*, aby wyświetlić zarejestrowane wartości dziennika.

MyWindowsHost: Security log: failed logon events

Timestamp	Local time	Source	Severity	Event ID	Value
2025-04-07 11:13:11 PM	2025-04-07 11:12:19 PM	Microsoft-Windows-Security-Auditing	Failure Audit	4625	An account failed to log on. Subject: Security ID: NT AUTHORITY\SYSTEM Account Name: DESKTOP-NNEM26U$ Account Domain: WORKGROUP Logon ID: 0x3E7 Logon Type: 2 Account For Which Logon Failed: Security ID: NULL SID Account Name: Admin Account Domain: DESKTOP-NNEM26U Failure Information: Failure Reason: Unknown user name or bad password. Status: 0xC000006D Sub Status: 0xC000006A

4. Jeśli wartości dziennika nie są widoczne, przejdź do sekcji Rozwiązywanie problemów w tym przewodniku.

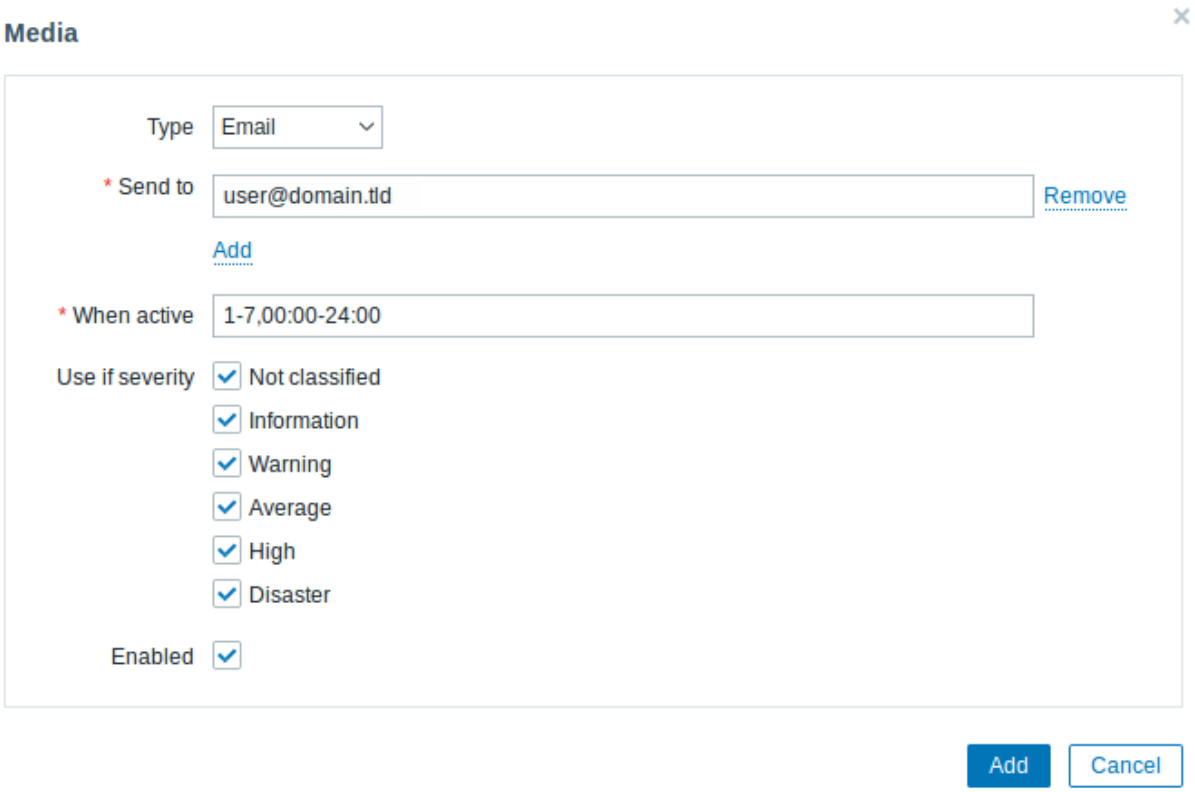
Konfiguracja alertów o problemach

Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do Data collection > Hosts, aby zdefiniować wyzwalacz, który uruchomi się, gdy pozycja dziennika zdarzeń zapisze interesujący Cię wzorzec. Na przykład, aby wykrywać nieudane próby logowania w dzienniku Security, użyj funkcji find():

find(/MyWindowsHost/eventlog[Security,,,4625,skip],10m,"like","Logon failed")

2. Przejdź do *User settings > Profile*, przejdź do karty *Media* i dodaj swój adres e-mail.



3. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Rozwiązywanie problemów

Jeśli napotkasz problemy ze zbieraniem lub wyświetlaniem dzienników zdarzeń systemu Windows, skorzystaj z poniższych wskazówek, aby zidentyfikować i rozwiązać typowe problemy:

1. Na serwerze Zabbix (Linux) wyświetl reguły iptables za pomocą następującego polecenia:

```
sudo iptables -L -n
```

i sprawdź, czy istnieje reguła ACCEPT dla portu TCP 10051.

2. Upewnij się, że klucz `eventlog[...]` używa dokładnej nazwy dziennika (z uwzględnieniem wielkości liter), identyfikatora zdarzenia, trybu (np. skip) oraz innych parametrów dokładnie tak, jak pokazano w kluczach pozycji specyficznych dla systemu Windows.

Zobacz także:

- Tworzenie pozycji - dowiedz się, jak dodawać dodatkowe metryki.
- agent Zabbix na Microsoft Windows - szczegółowe instrukcje instalacji.
- Monitorowanie Windows za pomocą agent Zabbix - kompleksowy przewodnik po konfiguracji podstawowego monitorowania komputerów z systemem Windows przy użyciu agent Zabbix.
- Klucze pozycji specyficzne dla Windows - szczegółowe informacje o kluczach pozycji specyficznych dla Windows obsługiwanych przez agent Zabbix, w tym dotyczących monitorowania dziennika zdarzeń.
- Monitorowanie plików dziennika - instrukcje dotyczące konfiguracji Zabbix do scentralizowanego monitorowania i analizy plików dziennika, mające zastosowanie do dzienników zdarzeń Windows.

Zabbix Cloud

Zabbix Cloud

Poznaj dostępne opcje konfiguracji i mechanizmy kontroli dostępu dla organizacji oraz węzłów w Zabbix Cloud.

Należy pamiętać, że węzły Zabbix Cloud obecnie działają w oparciu o **Zabbix 7.0**.
 Szczegółowe informacje na temat Zabbix 7.0 można znaleźć w oficjalnej dokumentacji.

Konfiguracja i zarządzanie Zabbix Cloud

Rozpocznij korzystanie z Zabbix Cloud Instrukcje krok po kroku, które pomogą szybko uruchomić instancję Zabbix Cloud, w tym przeprowadzić wstępną konfigurację i podłączyć monitorowane urządzenia.

Konfiguracja węzła Dowiedz się, jak skonfigurować węzeł Zabbix Cloud, koncentrując się na dostępie, obsłudze danych i ustawieniach zasobów.

Dodawanie użytkowników Dowiedz się, jak dodawać użytkowników i przypisywać im role w organizacji Zabbix Cloud oraz jej węzłach.

Dowiedz się więcej o Zabbix Cloud

Zabbix Cloud a on-premises Porównaj różnice w zarządzaniu, skalowalności i utrzymaniu między tymi dwoma wdrożeniami i dowiedz się, które z nich może najpłynniej wspierać Twój przepływ pracy.

Poznaj Zabbix Cloud Zapoznaj się z przeglądem Zabbix Cloud, w tym z jego kluczowymi funkcjami, korzyściami oraz tym, co obejmuje subskrypcja. Odpowiedzi na często zadawane pytania dotyczące konfiguracji i użytkowania znajdziesz w FAQ.

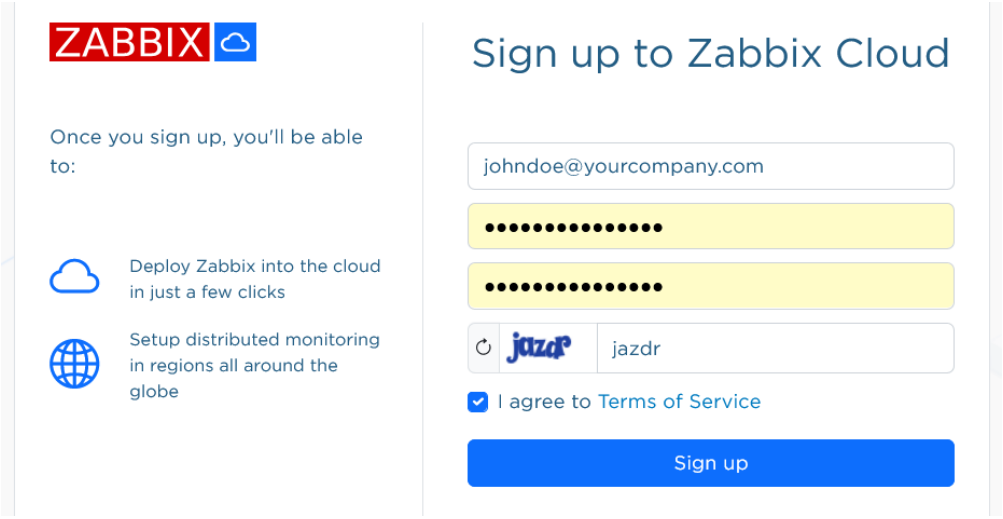
Wdróż Zabbix w chmurze

Aby uruchomić Zabbix w chmurze, wykonaj następujące kroki:

- Zaloguj się lub zarejestruj na cloud.zabbix.com
- Utwórz węzeł chmurowy
- Rozpocznij korzystanie z Zabbix w chmurze

Logowanie lub rejestracja Możesz zalogować się na cloud.zabbix.com przy użyciu istniejącego konta Google, GitHub lub Microsoft.

Alternatywnie przejdź do strony rejestracji, aby utworzyć konto.



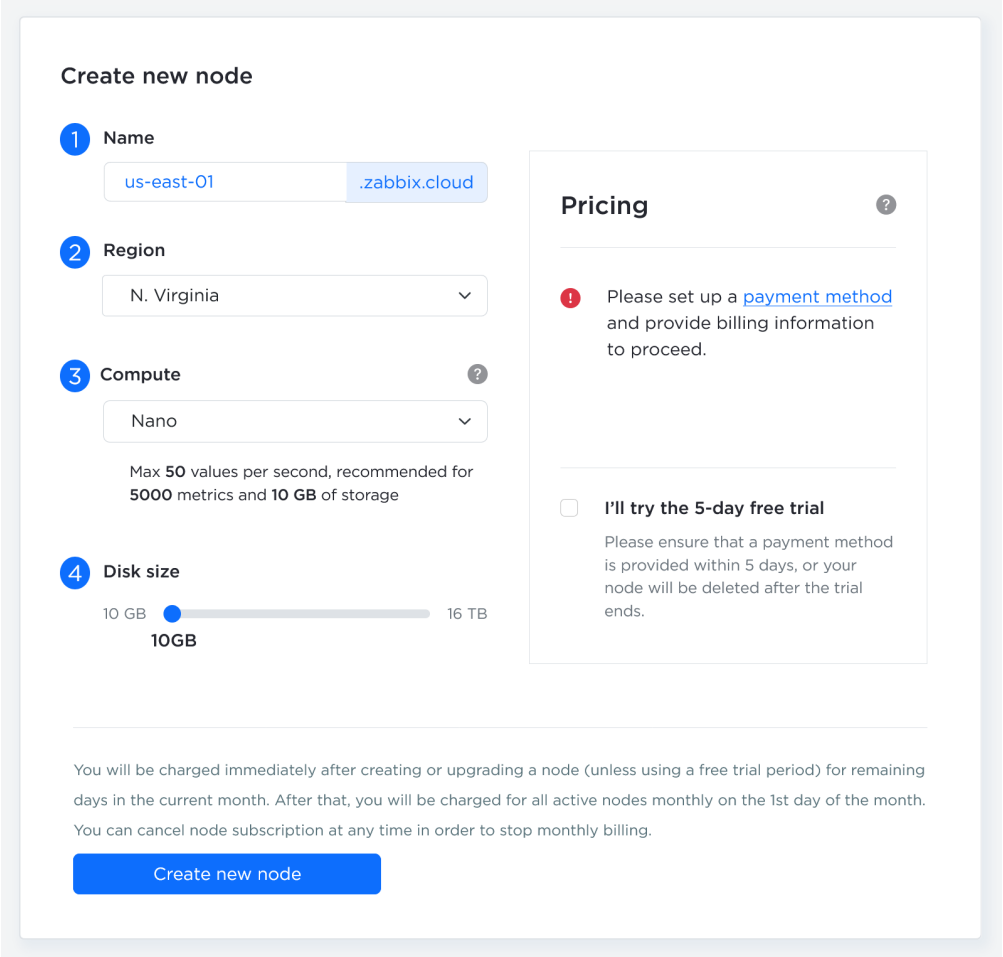
Po rejestracji otrzymasz w wiadomości e-mail potwierdzenie utworzenia nowego konta.

Przejdź do cloud.zabbix.com, aby się zalogować. Podczas logowania zostaniesz poproszony o jednorazowe hasło. Hasło to zostanie wysłane do Ciebie w wiadomości e-mail.

Tworzenie węzła chmurowego Węzeł chmurowy to instancja Zabbix działająca w chmurze. Obejmuje ona serwer Zabbix, bazę danych oraz frontend webowy.

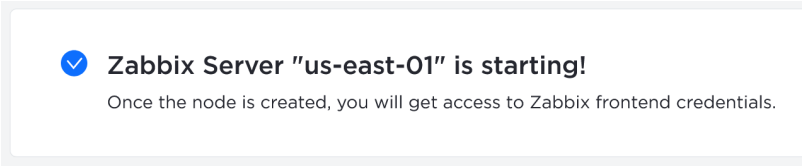
Po zalogowaniu zobaczysz, że dla Twojego konta domyślnie istnieje już organizacja *Private*. Domyślną organizację możesz później zmienić.

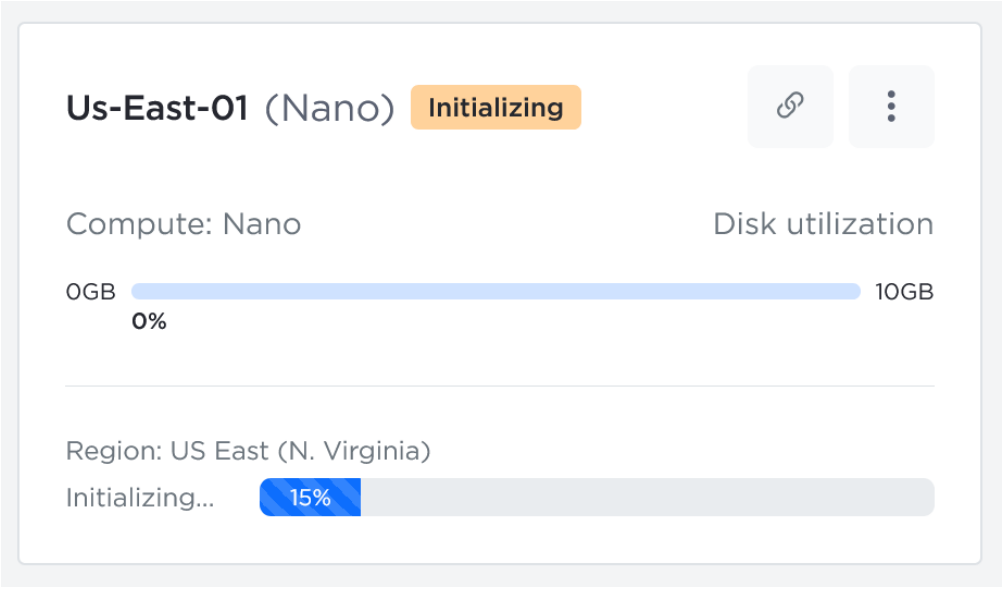
Kliknij *Create New Node*. Otworzy się formularz konfiguracji węzła.

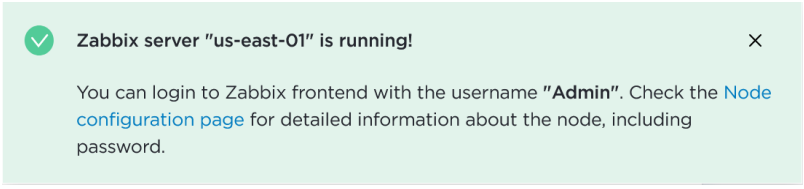


Parametr	Opis
Name	Wybierz unikalną nazwę węzła. Nazwa węzła tworzy unikalny adres URL serwera/frontendu chmurowego w formacie `<your-name>.zabbix.cloud`.
Region	Wybierz najbliższą lokalizację centrum danych, aby zminimalizować opóźnienia sieciowe.
Compute	Wybierz moc obliczeniową dla węzła. Aby określić, jaki poziom zasobów jest potrzebny, zobacz szczegóły cenowe. W razie potrzeby zawsze możesz później zwiększyć zasoby. Przed utworzeniem węzła innego niż próbny wymagane będzie skonfigurowanie metody płatności. Dodaj informacje rozliczeniowe w menu *Billing*. Dostępny jest 5-dniowy **bezpłatny okres próbny** umożliwiający wypróbowanie węzła poziomu *Nano*. W okresie próbnym nie zostaniesz obciążony opłatami, jednak po jego zakończeniu węzeł zostanie usunięty, jeśli nie zostanie przekształcony w węzeł płatny.
Disk size	Suwak domyślnie ustawiony jest na zalecane minimum zależne od poziomu. Zwiększ tę wartość, jeśli przechowujesz duże ilości historii lub trendów. Pamiętaj, że większe dyski wiążą się z wyższymi miesięcznymi kosztami oraz że nie możesz samodzielnie zmniejszyć rozmiaru dysku — aby go zmniejszyć, musisz skontaktować się z pomocą techniczną.

Po wypełnieniu formularza kliknij *Create New Node*. Podczas inicjalizacji węzła wyświetlany jest odpowiedni komunikat, a sam węzeł jest już widoczny w sekcji Organizations.





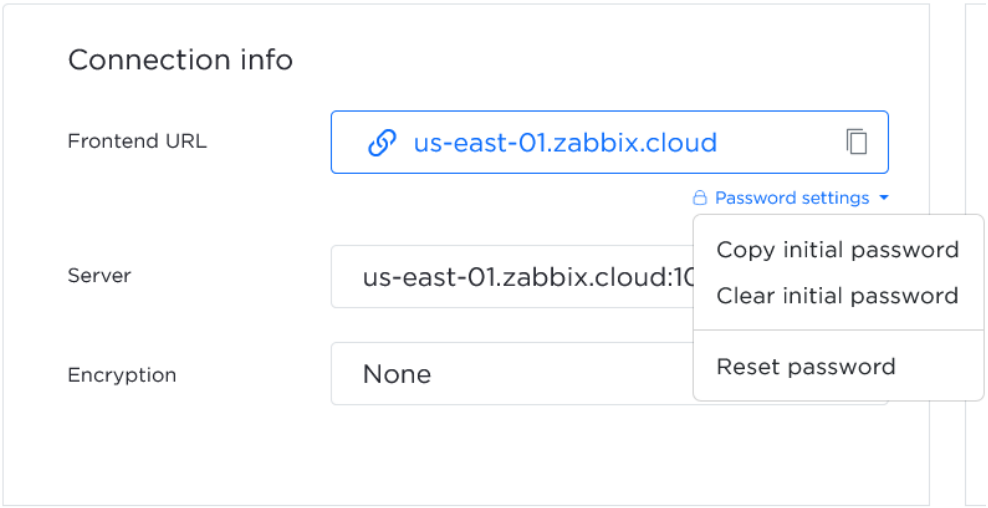


Rozpocznij korzystanie z Zabbix

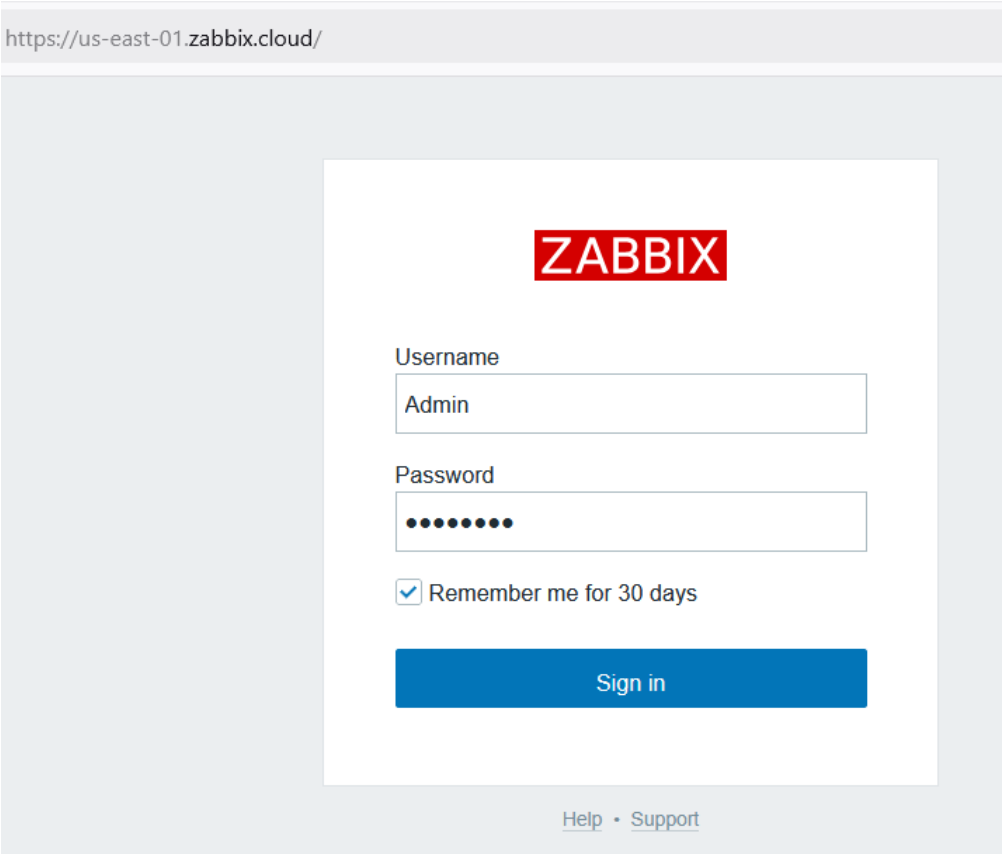
Gdy status węzła zmieni się z *Initializing* na *Running*, możesz zalogować się do frontend Zabbix.

Aby uzyskać dostęp do początkowego hasła frontend, kliknij utworzony węzeł.

Zostaniesz przeniesiony do ustawień węzła (przegląd, filtry dostępu, szyfrowanie, kopie zapasowe itp.). W sekcji **Connection info** kliknij *Password settings* i wybierz *Copy initial password* z listy rozwijanej.



Kliknij adres URL frontend, aby otworzyć frontend Zabbix (`us-east-01.zabbix.cloud`).



Zaloguj się, używając następujących danych logowania:

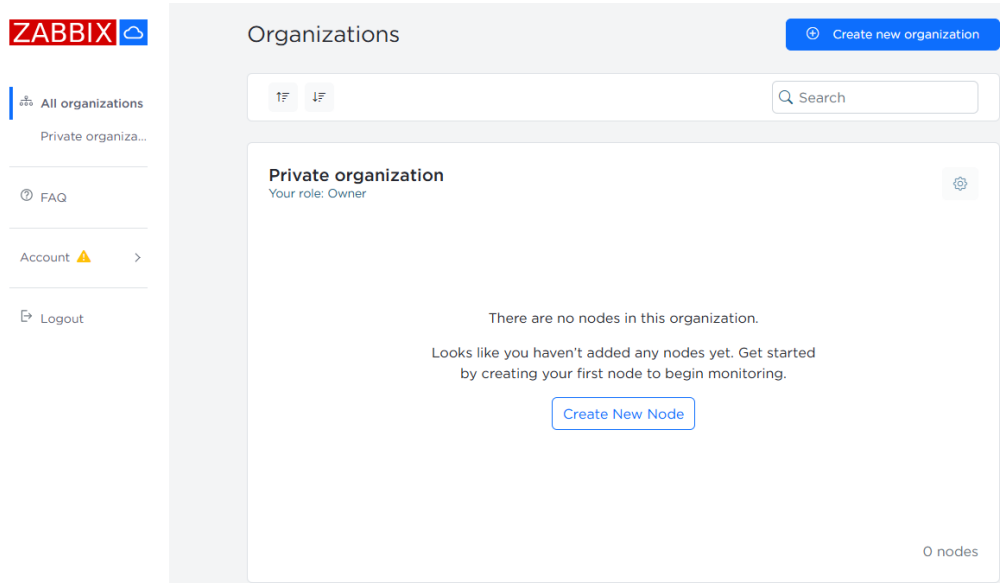
- *Username* - `Admin`
- *Password* - podaj początkowe hasło (np. HzTG9t7Y)

Aby rozpocząć monitorowanie, musisz zdefiniować filtry dostępu — adres IP lub adresy, którym wolno łączyć się z serwerem chmurowym Zabbix. Należy zdefiniować co najmniej jeden.

Zarządzanie organizacją Organizacja jest kontenerem dla jednego lub większej liczby węzłów. Organizacje pomagają również porządkować zasoby i delegować obowiązki, takie jak:

- Zarządzanie węzłami
- Rozliczenia

Po zalogowaniu do Zabbix Cloud wyświetlana jest strona *Wszystkie organizacje*. Domyślnie dla Twojego konta tworzona jest prywatna organizacja.



Jako właściciel tej organizacji możesz zarządzać użytkownikami, administratorami, węzłami oraz nazwą organizacji. Kliknij przycisk ustawień , aby uzyskać dostęp do opcji zarządzania.

Konfiguracja węzła

Przegląd

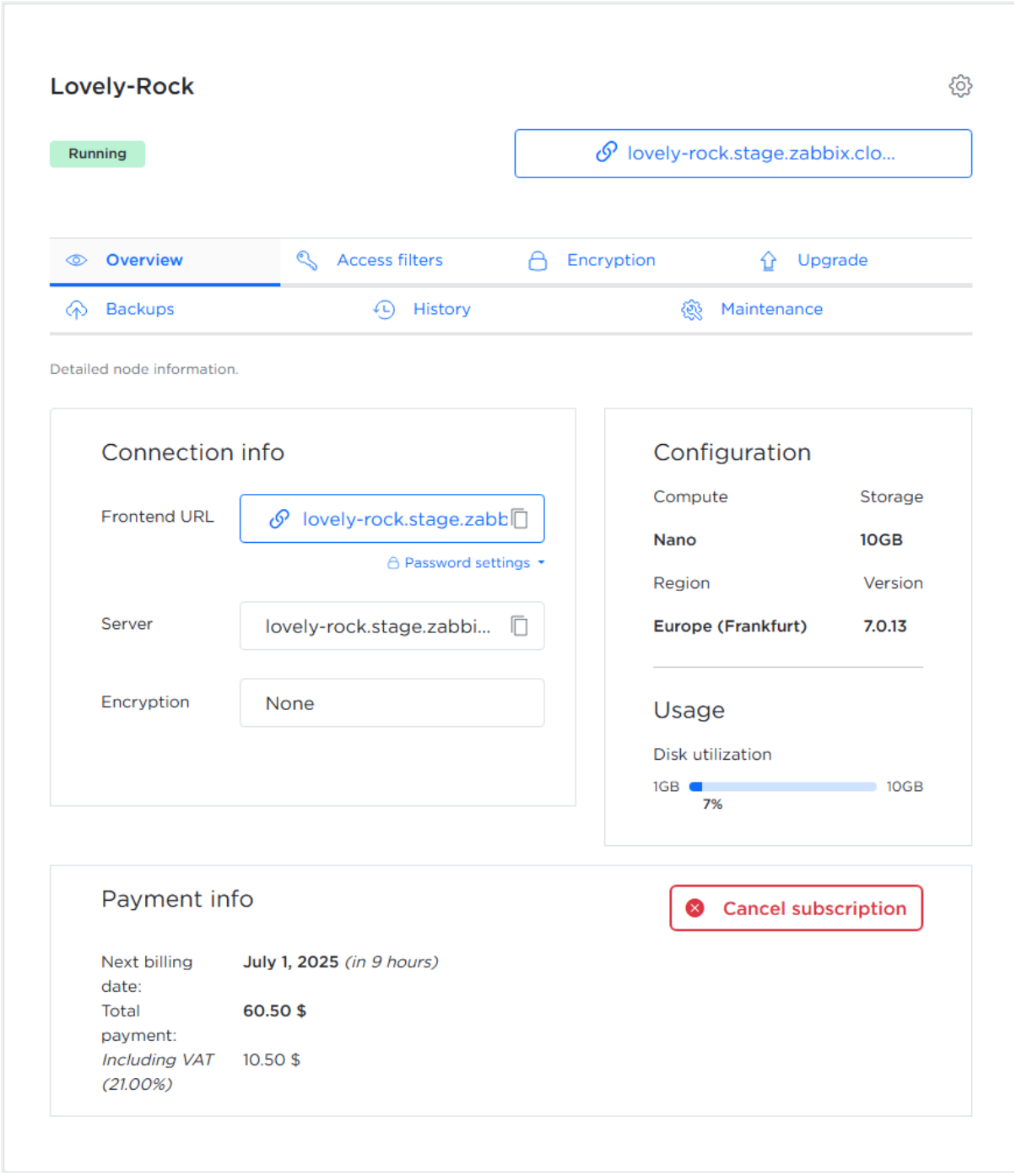
Węzeł chmurowy jest odpowiednikiem instancji Zabbix działającej w chmurze. Obejmuje serwer Zabbix, bazę danych oraz frontend WWW. Węzeł chmurowy ma dedykowane zasoby obliczeniowe (CPU, pamięć) oraz przestrzeń dyskową.

Użytkownicy mogą tworzyć jeden lub więcej węzłów chmurowych. Węzły chmurowe mogą być grupowane w **organizacje** (strukturalne, regionalne itp.), tworząc rozproszone środowisko monitoringu.

Węzły chmurowe i organizacje są tworzone za pomocą interfejsu użytkownika Zabbix Cloud. Aby uzyskać dostęp do strony konfiguracji węzła, kliknij węzeł w *All organizations*, w menu organizacji lub użyj odnośnika *Node configuration page* wyświetlanego po uruchomieniu węzła.

W widoku konfiguracji węzła każda karta umożliwia zarządzanie określonym aspektem węzła:

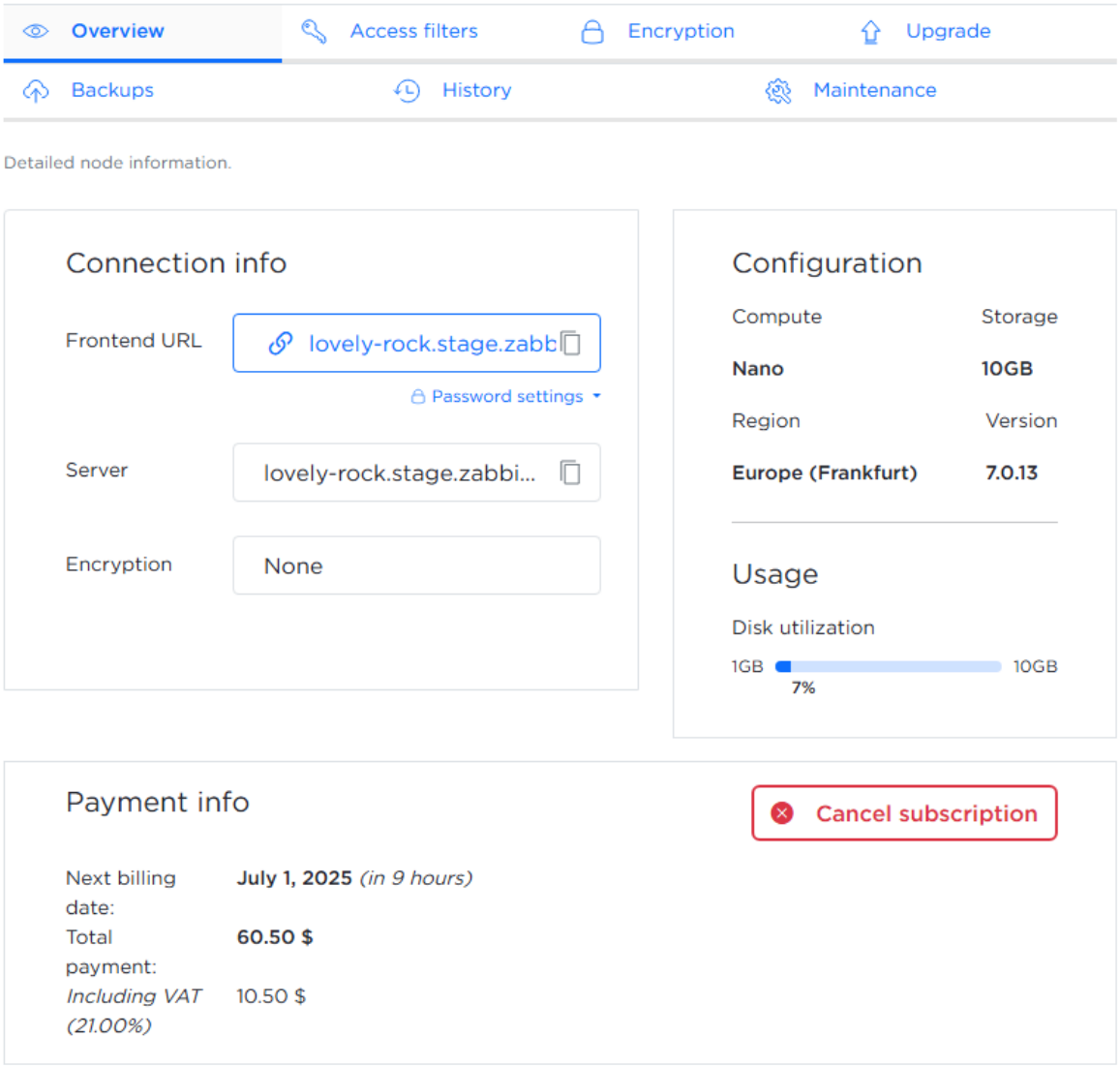
- Przegląd — wyświetlanie szczegółów węzła, otwieranie jego frontend oraz zarządzanie użytkownikami węzła.
- Filtry dostępu — kontrola dostępu według adresu IP lub podsieci.
- Szyfrowanie — konfiguracja TLS przez przesyłanie certyfikatów, kluczy i opcjonalnych list CRL.
- Kopie zapasowe — tworzenie, przywracanie i usuwanie migawek węzła; wyświetlanie terminu następnej zaplanowanej kopii zapasowej.
- Historia — ustawianie okresów przechowywania dla dzienników audytu, historii i trendów, z predefiniowanymi jednostkami oraz limitem aktualizacji 3 razy dziennie.
- Konserwacja — definiowanie cotygodniowych okien konserwacyjnych według strefy czasowej, dnia i godziny.
- Aktualizacja — skalowanie zasobów obliczeniowych/pamięci masowej lub zarządzanie statusem wersji próbnej/płatności.



Note:
Niektóre karty mogą być niedostępne w zależności od typu użytkownika, który wyświetla stronę, oraz statusu węzła (wersja próbna/poza okresem próbnym).

Karta Przegląd

Karta *Przegląd* zawiera szczegółowe informacje o węźle.



Jest podzielona na trzy sekcje:

Informacje o połączeniu:

- Klikalny adres URL frontend z opcją kopiowania do schowka
- Ustawienia hasła
- Bieżąca metoda szyfrowania

Konfiguracja i użycie:

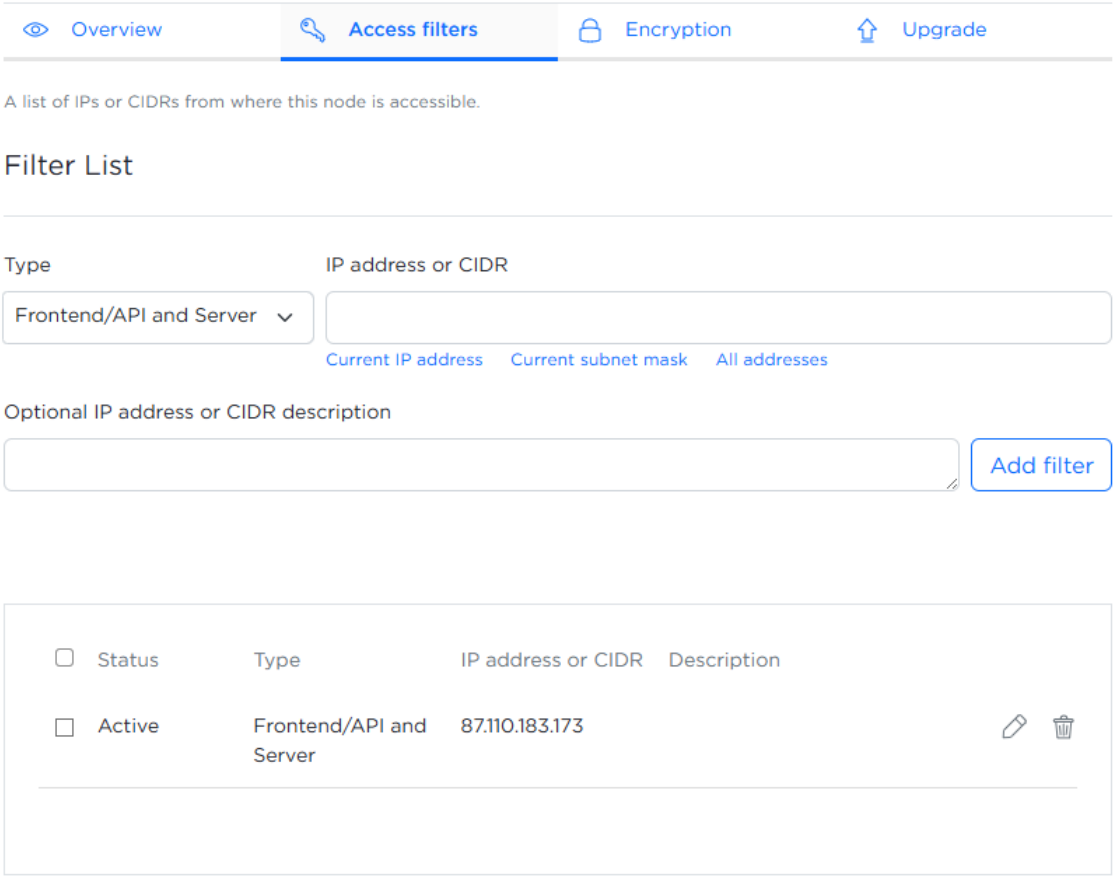
- Poziom mocy obliczeniowej, maksymalna przestrzeń dyskowa, region
- Wersja komponentów Zabbix
- Wykorzystanie dysku

Informacje o węźle/Informacje o płatności:

- Data wygaśnięcia (dla węzłów testowych i przedpłaconych)
- Po zakończeniu inicjalizacji węzła wyświetlany jest przycisk zależny od kontekstu, w zależności od typu węzła: - *Przejdź na plan płatny* - dostępne dla węzłów w bezpłatnym okresie próbnym - *Anuluj subskrypcję* - dostępne dla płatnych węzłów z aktywną subskrypcją - *Ponownie zasubskrybuj plan płatny* - wyświetlane, jeśli subskrypcja została anulowana - Dla węzłów przedpłaconych nie jest wyświetlany żaden przycisk akcji, ponieważ mają długi okres ważności i nie można ich anulować

Karta filtrów dostępu

Karta *Filtry dostępu* umożliwia ograniczenie dostępu do węzła Zabbix na podstawie adresu IP lub podsieci.



Aby skonfigurować filtr dostępu, uzupełnij następujące pola:

- *Typ* — użyj listy rozwijanej, aby wybrać, który komponent Zabbix ma zostać ograniczony: tylko frontend/API, tylko serwer lub oba.
- *Adres IP lub CIDR* — wprowadź adres IP lub CIDR ręcznie albo uzupełnij pole automatycznie, wybierając odpowiednią opcję.
- *Opcjonalny opis adresu IP lub CIDR* — dodaj opcjonalny opis dla każdego adresu IP lub podsieci.

Po uzupełnieniu pól kliknij *Dodaj filtr*.
Dodane filtry pojawią się w poniższej tabeli, zawierającej status, typ, adres IP lub podsieć oraz opis.

Warning:
Podczas inicjalizacji węzła filtr dostępu dla użytego adresu IP jest tworzony automatycznie. Jeśli później połączysz się z innego adresu IP lub z innej sieci, musisz dodać ten adres tutaj, w przeciwnym razie dostęp zostanie zablokowany.

Możesz:

- — edytować filtr.
- — usunąć filtr lub wiele filtrów za pomocą pól wyboru.

Karta Szyfrowanie

Karta *Szyfrowanie* umożliwia skonfigurowanie TLS encryption między komponentami Zabbix przez przesłanie wymaganych plików certyfikatów.

Aby włączyć bezpieczną komunikację z użyciem certyfikatów, wybierz *CERTIFICATE* i prześlij z komputera następujące pliki:

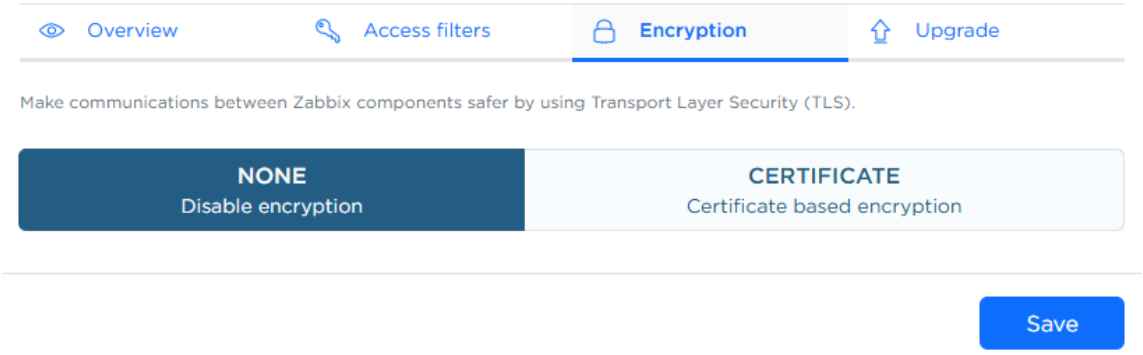
- *Certyfikaty głównego urzędu certyfikacji (Root CA)* — używane do weryfikacji autentyczności certyfikatów drugiej strony. Kliknij *Choose file* i wybierz plik zawierający zaufane certyfikaty głównego urzędu certyfikacji (zwykle plik `.crt` lub `.pem`, zakodowany w formacie PEM).

- *Łańcuch certyfikatów* — reprezentuje łańcuch zaufania dla własnego certyfikatu.
  Kliknij *Choose file* i wybierz plik zakodowany w formacie PEM (np. `.crt` lub `.pem`), który zawiera Twój certyfikat oraz wszystkie pośrednie certyfikaty CA.

- *Klucz prywatny* — klucz prywatny odpowiadający Twojemu certyfikatowi.
  Kliknij *Choose file* i wybierz plik `.key` zakodowany w formacie PEM.

- *Unieważnione certyfikaty* (opcjonalnie) — jeśli utrzymujesz listę unieważnionych certyfikatów (CRL), kliknij *Choose file* i wybierz plik CRL, aby wskazać certyfikaty, które nie powinny być już uznawane za zaufane.

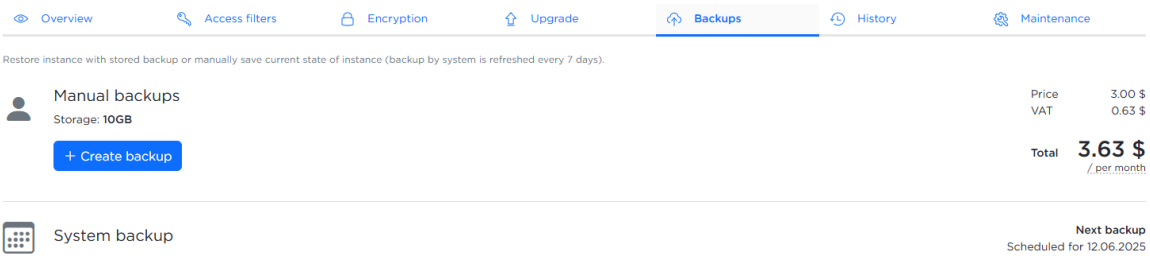
Po wybraniu plików kliknij *Save*. Pliki zostaną przesłane, zweryfikowane i zastosowane do skonfigurowania szyfrowania TLS między komponentami Zabbix.



Karta Kopie zapasowe

Karta *Kopie zapasowe* umożliwia przeglądanie, tworzenie, przywracanie i usuwanie kopii zapasowych węzła Zabbix.

Możesz przywrócić instancję z dowolnej zapisanej kopii zapasowej lub w dowolnym momencie utworzyć nową ręczną kopię zapasową.
Karta pokazuje również datę następnej zaplanowanej kopii zapasowej.



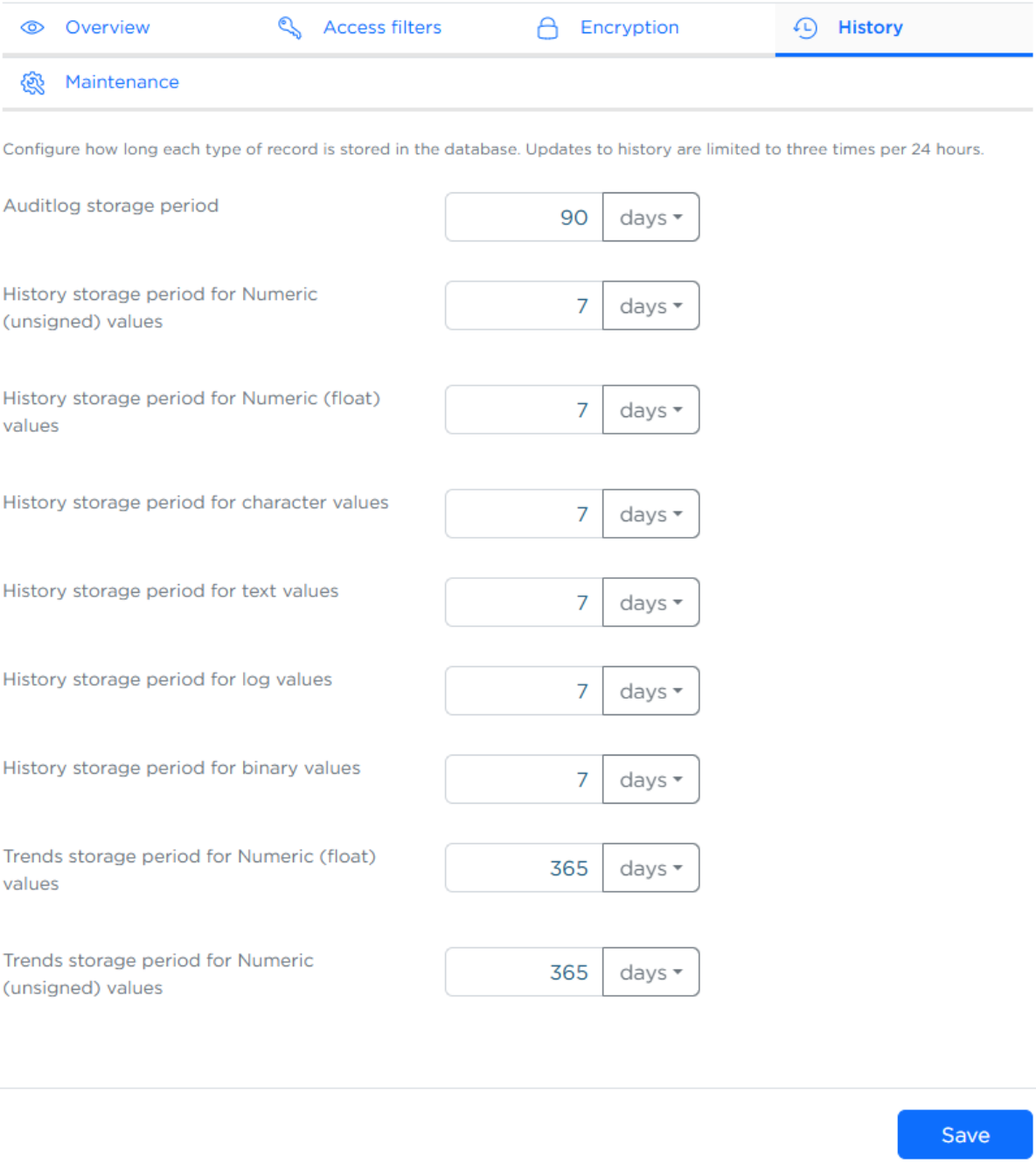
Aby utworzyć kopię zapasową:

1. Kliknij przycisk *Create backup*.
2. W wyskakującym oknie wprowadź opcjonalny opis.
3. Kliknij *Create*, aby potwierdzić.

Warning:
Dodatkowe kopie zapasowe są rozliczane miesięcznie na podstawie wykorzystania przestrzeni dyskowej.

Karta Historia

Karta *Historia* umożliwia skonfigurowanie, jak długo przechowywane są różne typy danych historycznych.



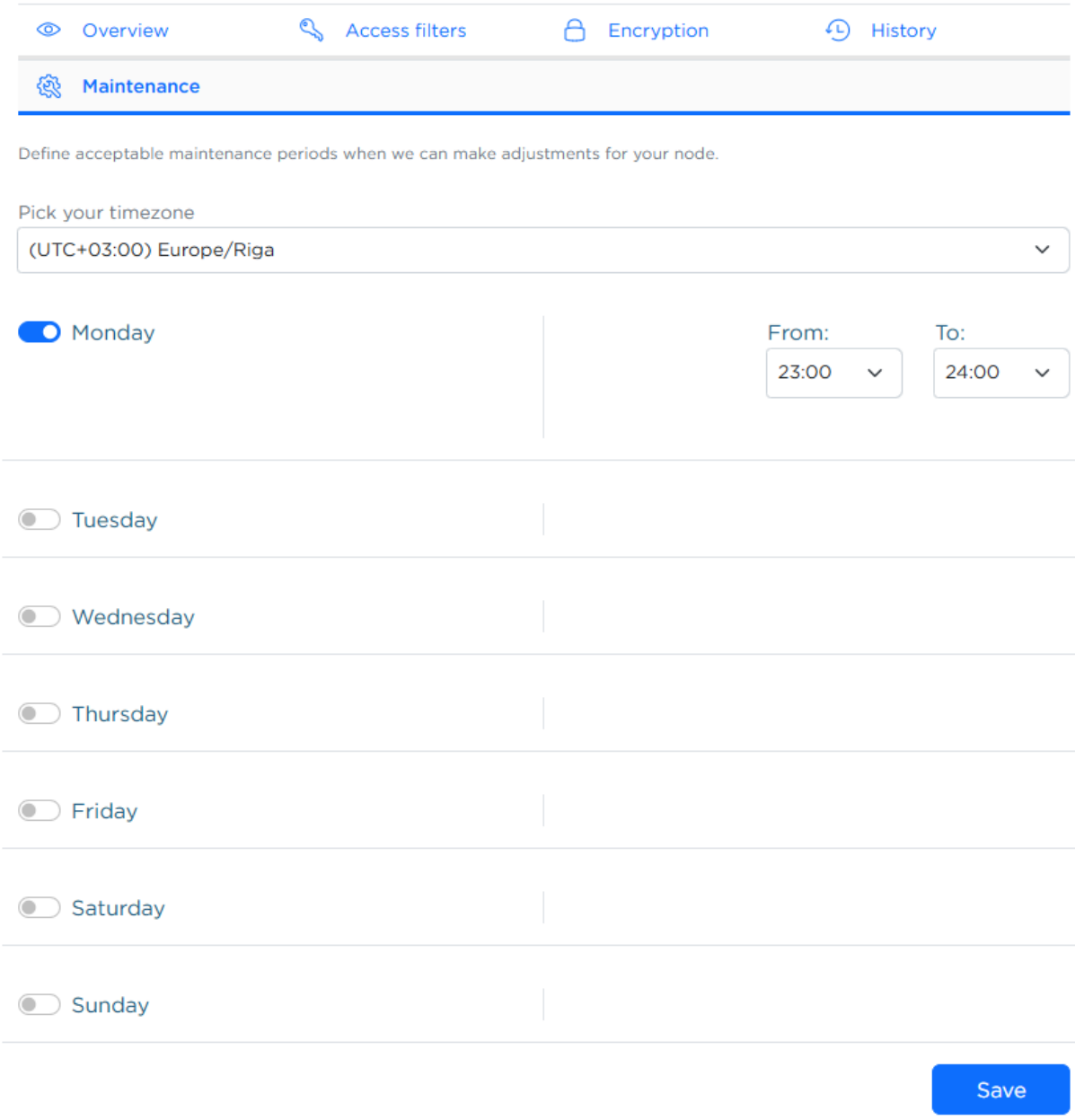
Możesz ustawić okres przechowywania dla każdego typu danych (np. dane liczbowe, tekst, logi) oraz wybrać jednostkę (dni, miesiące, lata). Zmiany można wprowadzać maksymalnie trzy razy w ciągu 24 godzin.

Po zaktualizowaniu wartości kliknij *Zapisz*, aby zastosować zmiany.

Warning:
Dłuższe okresy przechowywania mogą zwiększyć wykorzystanie przestrzeni dyskowej i mogą wpływać na wydajność. Zaleca się konfigurowanie okresów przechowywania na podstawie rzeczywistych danych i potrzeb analitycznych.
 Zabbix Cloud używa partycjonowania bazy danych do zarządzania przechowywaniem danych zamiast tradycyjnego mechanizmu housekeeping w Zabbix. Umożliwia to szybsze działanie i bardziej przewidywalną obsługę retencji danych.

Karta Konserwacja

Karta *Konserwacja* umożliwia zdefiniowanie akceptowalnych okien konserwacyjnych, w których na węźle mogą być wykonywane aktualizacje lub dostosowania.

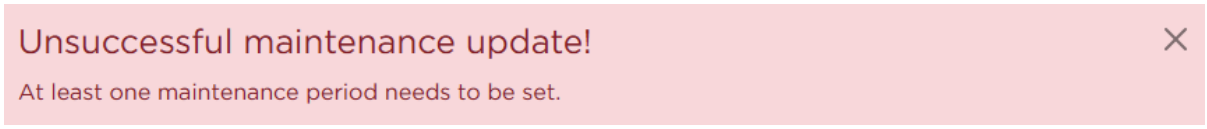


Należy:

- Wybrać strefę czasową z listy rozwijanej.
- Wybrać co najmniej jeden dzień i jedno godzinne okno konserwacyjne.

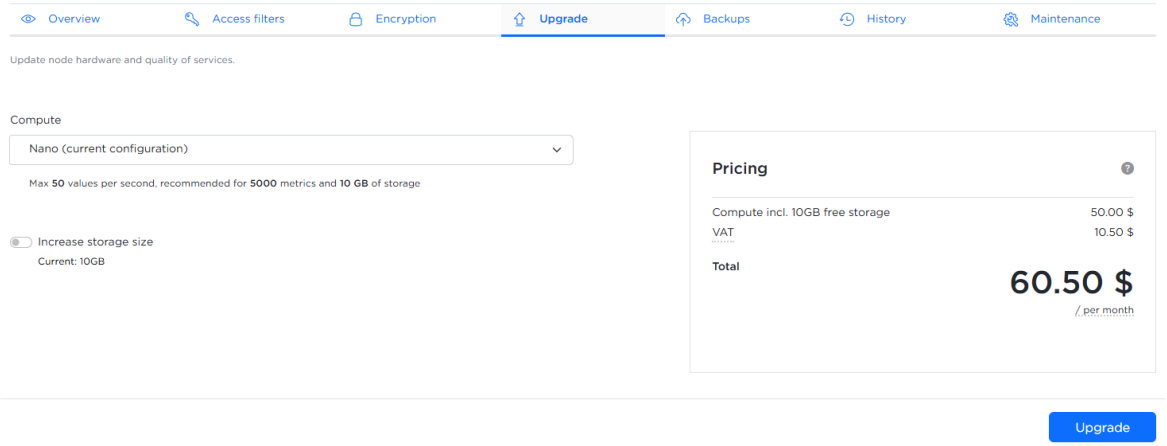
Kliknij *Zapisz*, aby zastosować zmiany.

Jeśli nie zostanie wybrany żaden dzień ani okno czasowe, zostanie wyświetlony błąd.



Karta Upgrade

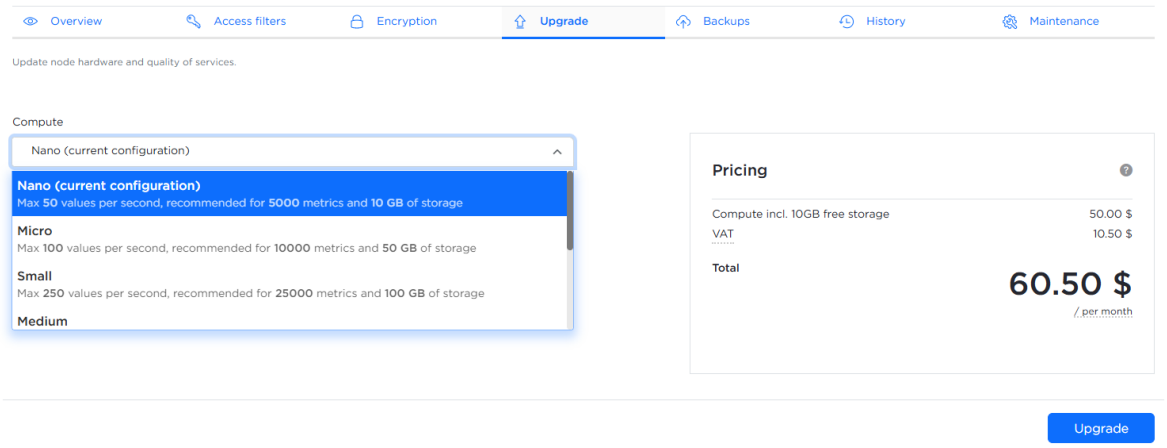
Karta *Upgrade* umożliwia skalowanie zasobów węzła lub zarządzanie jego subskrypcją.



Możesz zwiększyć poziom mocy obliczeniowej lub rozmiar dysku — pamiętaj, że rozmiar dysku można zwiększyć tylko raz na 24 godziny i nie można go zmniejszyć za pomocą interfejsu użytkownika (w przypadku próśb o zmniejszenie pojemności skontaktuj się ze wsparciem Zabbix).
Węzły w okresie próbnym można przekształcić w płatne subskrypcje.

Aby zwiększyć poziom mocy obliczeniowej:

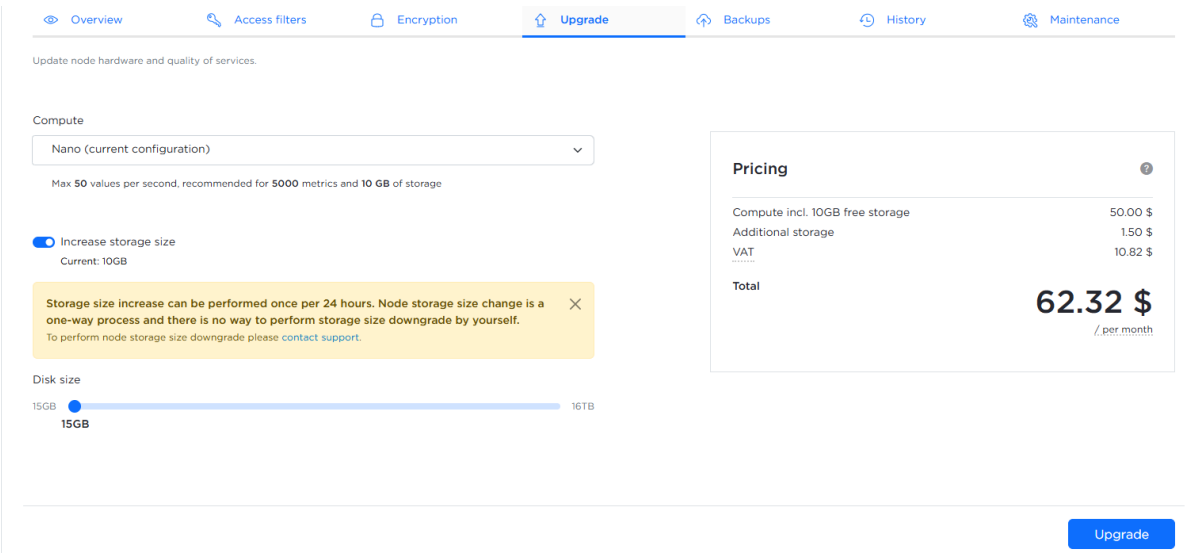
1. Kliknij listę rozwijaną *Compute*.



2. Wybierz żądaną konfigurację.
3. Kliknij przycisk *Upgrade*.

Aby zwiększyć rozmiar dysku:

1. Kliknij przycisk opcji *Increase storage size*.
2. Ustaw suwak *Disk size* na żądaną wartość.



3. Kliknij przycisk *Upgrade*.

Note:
Zaktualizowana całkowita cena miesięczna jest wyświetlana w tabeli *Pricing* przed potwierdzeniem zmian.

Dodawanie użytkowników

Wprowadzenie

Poniższe przykłady opisują, jak dodawać użytkowników do organizacji i węzłów.

Note:
Kopiowanie, czyszczenie i resetowanie hasła frontend może być wykonywane wyłącznie przez właściciela.

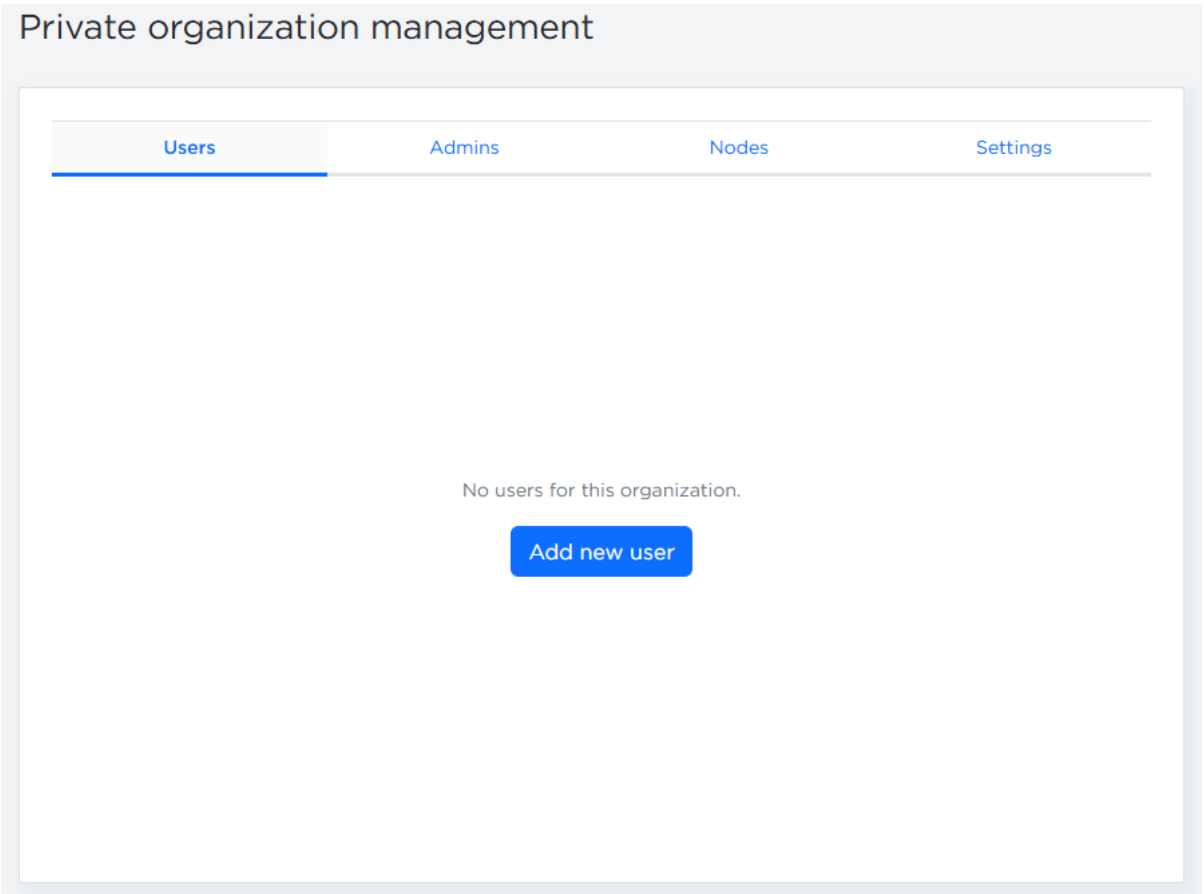
Dodawanie użytkowników do organizacji

Podobnie jak użytkownicy w Zabbix, możesz tworzyć użytkowników w Zabbix Cloud i przypisywać im role, aby kontrolować, co mogą robić w organizacji i jej węzłach.

Aby dodać użytkowników do swojej organizacji:

1. Otwórz stronę *All organizations*.

2. Kliknij przycisk ustawień wybranej organizacji, aby przejść do jej menu zarządzania.



3. Na karcie *Users* lub *Admins* kliknij *Add new user*.

4. Wprowadź adres e-mail użytkownika i wybierz rolę: *Administrator* lub *User*.

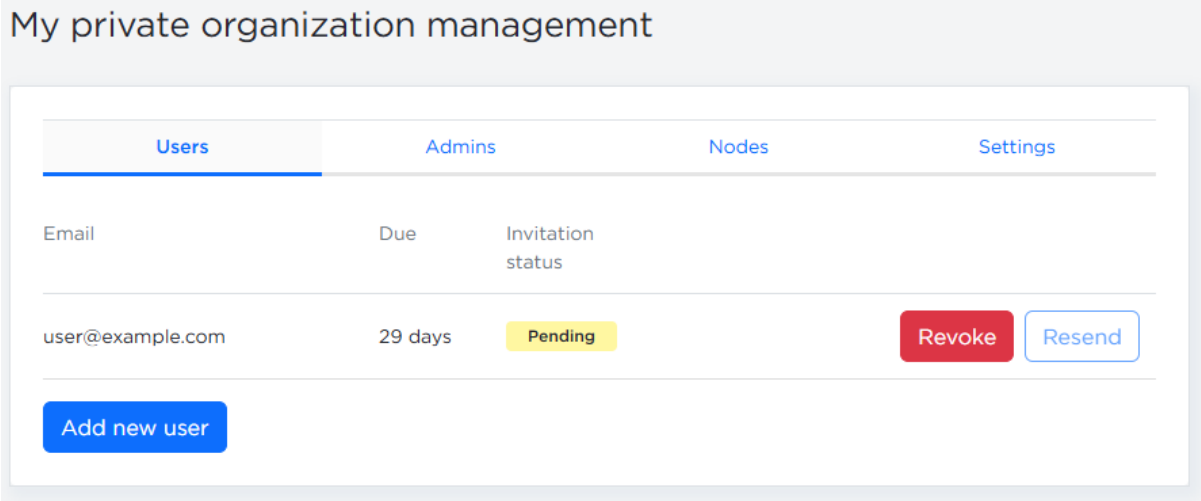
Pamiętaj, że wybranie roli *User* dodaje użytkownika do organizacji, ale nie nadaje dostępu do żadnych węzłów.

Każda rola zapewnia inny poziom dostępu:

- **User** — może uzyskiwać dostęp do przypisanych mu węzłów i wykonywać zadania konfiguracyjne, które nie generują opłat, na przykład:
  - Zarządzać filtrami dostępu
  - Konfigurować szyfrowanie
  - Ustawiać okresy przechowywania historii/trendów
  - Planować okresy konserwacji
- **Administrator** — ma wszystkie uprawnienia roli User i może również wykonywać działania, które mogą generować opłaty, na przykład:
  - Tworzyć i przywracać kopie zapasowe
  - Zwiększać lub zmniejszać zasoby węzła (warstwę obliczeniową i rozmiar dysku)
  - Anulować subskrypcje
- **Owner** — ma pełną kontrolę nad organizacją. Oprócz wszystkich uprawnień roli Administrator, Owner może:
  - Tworzyć i usuwać węzły
  - Dodawać lub usuwać użytkowników i administratorów
  - Zmieniać nazwę organizacji lub ją usuwać
  - Zarządzać danymi rozliczeniowymi

> **Attention:**
> Metoda płatności Owner jest obciążana za wszystkie usługi. Jeśli wymagane jest potwierdzenie bankowe, Owner musi autoryzować transakcję, nawet jeśli została zainicjowana przez Administratora.

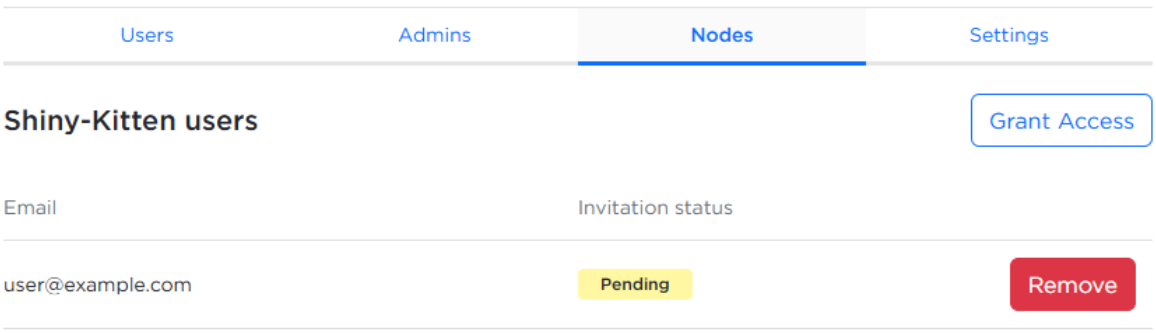
Zaproszony użytkownik zostanie dodany ze statusem *Pending*.
Możesz cofnąć zaproszenie lub wysłać je ponownie (ponowne wysłanie jest dostępne raz na 10 minut — przycisk będzie wyłączony po osiągnięciu tego limitu).



Dodawanie użytkowników do węzłów

Aby dodać użytkownika do węzła:

1. Przejdź do zakładki *Nodes*.
2. Kliknij *Grant Access*.
3. Wybierz z listy rozwijanej użytkownika dodanego do organizacji.
4. Kliknij *Grant access*.



Alternatywnie możesz nadać dostęp z okna *Node configuration*:

1. Kliknij przycisk ustawień  .
2. Wybierz *Add user*.
3. Wybierz użytkownika i kliknij *Grant access*.

Zaproszeni użytkownicy muszą zaakceptować zaproszenie, tworząc konto Zabbix Cloud przy użyciu tego samego adresu e-mail, na który wysłano zaproszenie. Jeśli użytkownik ma już konto Zabbix Cloud przypisane do tego adresu e-mail, musi się zalogować i potwierdzić zaproszenie. Zaproszenia są powiązane z konkretnym adresem e-mail i nie mogą zostać zaakceptowane z innego konta.

Kluczowe różnice w Zabbix Cloud

Wprowadzenie

Zabbix Cloud to platforma oparta na chmurze, która umożliwia użytkownikom uruchamianie samodzielnych instancji Zabbix w chmurze (węzły chmurowe).

W porównaniu ze standardowym wdrożeniem Zabbix lokalnie, w infrastrukturze własnej:

- Nie jest wymagana instalacja ani konserwacja.
- Węzły chmurowe są w pełni zarządzane przez Zabbix.
- Węzły chmurowe można uruchomić w ciągu kilku minut.

Usługa jest oferowana w modelu płatności za rzeczywiste użycie (dostępny jest **bezpłatny okres próbny**).

Tabela porównawcza Chociaż większość funkcjonalności pokrywa się między lokalnymi instalacjami Zabbix a węzłami chmurowymi, istnieją również różnice. Poniżej przedstawiono zbiorczą tabelę porównawczą. W kolejnych sekcjach szczegółowo opisano wybrane pozycje.

Funkcja / Aspekt	Zabbix Cloud	Zabbix lokalnie
Cykl wydań	Zgodny z wydaniami LTS releases	Wybierasz własną wersję Zabbix (stabilną, beta, niestandardowe buildy)
Łatwość użycia	Platforma gotowa do użycia, w pełni zarządzana przez Zabbix	Wymaga instalacji, konfiguracji i utrzymania
Łatanie podatności	Automatyczne aktualizacje systemu operacyjnego/Zabbix	Ręczne aktualizacje wykonywane przez użytkownika
Certyfikat HTTPS	Ważny certyfikat dostępny od razu po uruchomieniu (bez Let's Encrypt)	Domyślnie samopodpisany (można skonfigurować Let's Encrypt lub inne CA)
Wydajność DB (INSERT/UPDATE/SELECT)	Automatyczne partycjonowanie dla maksymalnej szybkości; brak długotrwałych operacji DELETE	Ręczne partycjonowanie i czyszczenie (możliwe długie transakcje DELETE)
Dostrajanie wydajności	Wstępnie dostrojone przez zespół Zabbix Cloud (brak changelogów)	Pełna widoczność i kontrola nad parametrami dostrajania
Kontrola zapory sieciowej	Zarządzana przez zespół Zabbix Cloud; porty GUI/trapper nie są publicznie wystawione	Pełna kontrola nad regułami zapory i ekspozycją portów
Izolacja i ochrona DB	Wysoki poziom izolacji, brak bezpośredniego dostępu do DB/SSH	Zależy od Twojej sieci i zabezpieczeń hosta
Wysoka dostępność (HA)	Działa na platformie chmurowej o wysokiej dostępności, proxy Zabbix mogą być używane dla większej odporności zbierania danych	HA zarządzane przez użytkownika
Pułapki SNMP	Tylko przez dedykowany proxy (brak odbiornika pułapek z HA/balansowaniem obciążenia)	Bezpośrednio lub przez proxy, możliwe HA/balansowanie obciążenia
Odpytywanie SNMP	Trudne bez proxy (każde urządzenie wymaga NAT/niestandardowego portu)	Natywne odpytywanie SNMP; proxy opcjonalny
Ustawienia retencji	Historia/Trendy/Audyt tylko przez UI (bez API ani pliku konfiguracyjnego)	Konfigurowalne w pliku konfiguracyjnym serwera Zabbix lub przez API

Funkcja / Aspekt	Zabbix Cloud	Zabbix lokalnie
Niestandardowe skrypty	`AlertScriptsPath`, `ExternalScripts`, frontend i moduły społeczności nie są obsługiwane	W pełni obsługiwane (ścieżka skryptów, moduły, integracje)
Monitorowanie ODBC	Tylko PostgreSQL (sterownik `{postgresql}`); wtyczka MariaDB obecna jako placeholder	ODBC dla PostgreSQL, MySQL, Oracle itd. - w pełni konfigurowalne
Ograniczanie wywołań ODBC	Niemożliwe (`StartODBCPollers=1` tylko; ciężkie synchroniczne zapytania blokują inną pracę)	W pełni dostrajana liczba pollerów i harmonogramowanie
Przesyłanie certyfikatu SAML	Nieobsługiwane	Obsługiwane przez UI lub API
Raporty cykliczne	Należy samodzielnie utworzyć nośnik Email (nośnik skryptowy nie jest obsługiwany)	Obsługuje zarówno nośnik skryptowy, jak i Email od razu po uruchomieniu
Interfejs hosta dla aktywnych testów	Adres IP przydzielany przez platformę (tworzony automatycznie)	Interfejsy zarządzane przez Ciebie; adresy IP pod Twoją kontrolą

Różnice funkcjonalne Pułapki SNMP

Pułapki SNMP są obsługiwane wyłącznie przez dedykowany proxy Zabbix. Jeśli wymagane jest monitorowanie pułapek SNMP, nie można używać automatycznego równoważenia obciążenia ani wysokiej dostępności dla proxy, ponieważ pułapki SNMP muszą być wysyłane na pojedynczy adres IP.

Odpytywanie SNMP

Bez proxy odpytywanie SNMP wymaga, aby każde urządzenie było udostępnione przez NAT i niestandardowe porty.

Ustawienia retencji

Okresy retencji historii, trendów i dziennika audytu można konfigurować w chmurze wyłącznie za pośrednictwem interfejsu webowego. Tych ustawień nie można konfigurować za pomocą `zabbix_server.conf` ani przez API.

Ręczne nadpisywanie historii dla poszczególnych pozycji nie jest obsługiwane (partycjonowanie jest kontrolowane globalnie). Adresy URL węzłów chmurowych nie akceptują tych samych parametrów zapytania co instalacje lokalne.

Własne skrypty

Następujące typy własnych skryptów nie są obsługiwane przez Zabbix Cloud:

- Skrypty alertów (`AlertScriptsPath`)
- Skrypty zewnętrzne (`ExternalScripts`)
- Skrypty frontend na serwerze Zabbix

Nie można instalować modułów frontend opracowanych przez społeczność.

Monitorowanie ODBC

Zabbix Cloud obsługuje monitorowanie ODBC tylko dla PostgreSQL. Użyj oficjalnego szablonu ODBC Zabbix i zdefiniuj connection string w szablonie, używając:

```
Driver={postgresql}
```

Wtyczka MariaDB jest również zainstalowana, ale obecnie nie działa. Jeśli jest używana, zdefiniuj:

```
Driver={mysql}
```

Attention:
Nie jest znana żadna działająca konfiguracja dla Oracle MySQL 8.0 w Zabbix Cloud. Chociaż proste zapytania (takie jak `SELECT 1`) mogą się powieść, bardziej złożone zapytania zwracają `SQL_ERROR`.

 Nie ma możliwości ograniczenia wywołań ODBC (`StartODBCPollers=1` only). Intensywne synchroniczne raportowanie może wpływać na wydajność, a jednocześnie może być uruchomione tylko pojedyncze `SELECT`.

Dostęp do infrastruktury

Zabbix Cloud nie zapewnia dostępu SSH do podstawowych węzłów ani nie umożliwia bezpośrednich połączeń z bazą danych (na przykład do portu 3306). Wszystkie działania związane z konfiguracją, monitorowaniem i rozwiązywaniem problemów muszą być wykonywane za pośrednictwem interfejsu użytkownika chmury lub API, co zapewnia izolację i bezpieczeństwo warstw systemu operacyjnego oraz bazy danych.

Interfejs aktywnych kontroli

Aktywne kontrole w chmurze automatycznie tworzą interfejs hosta z adresem IP niezwiązanym z lokalną siecią. Domyślnie ten adres IP może być zewnętrzny. Aby zachować spójność, po utworzeniu możesz ręcznie usunąć lub dostosować interfejs hosta.

Przesyłanie certyfikatu SAML

Uwierzytelnianie SAML nie jest obsługiwane, ponieważ niewielu dostawców akceptuje niezaszyfrowane lub niepodpisane komunikaty SAML.

Zaplanowane raporty

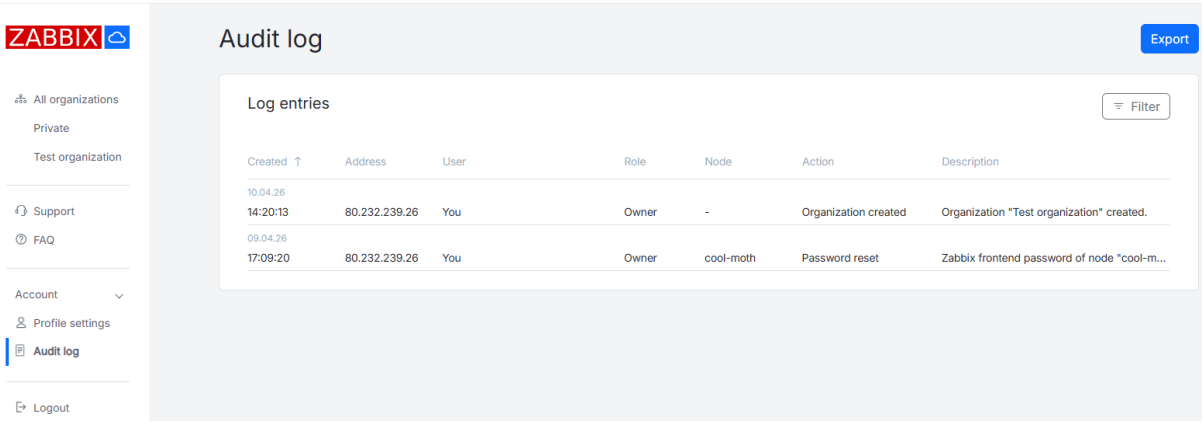
Domyślny typ mediów *Cloud Email* używa transportu skryptowego i nie może być używany do zaplanowanych raportów. Aby wysyłać raporty pocztą e-mail:

1. Utwórz nowy typ mediów Email.
2. Przypisz medium e-mail do swojego użytkownika w *Ustawienia użytkownika* > *Media*.
3. Wybierz ten typ mediów podczas konfigurowania zaplanowanych raportów.

Dziennik audytu

Przegląd W Zabbix Cloud dziennik audytu rejestruje aktywność użytkowników i systemu.

Po zalogowaniu się do konta Zabbix Cloud przejdź do *Account* > *Audit log*, aby wyświetlić wpisy dziennika audytu. Dostępne dla użytkownika wpisy dziennika audytu zależą od roli organizacyjnej użytkownika.



Wyświetlanie rekordów dziennika audytu Rekordy dziennika audytu są wyświetlane w tabeli.

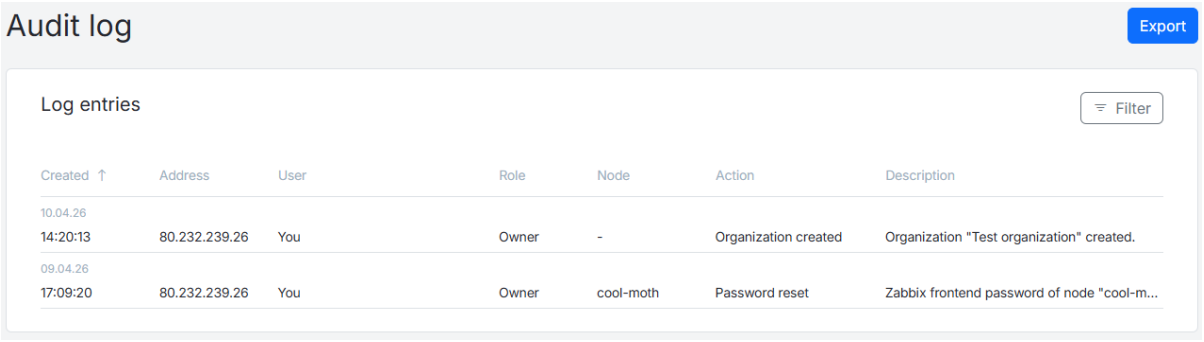


Tabela zawiera następujące informacje:

Column	Description
Created	Data i godzina utworzenia wpisu dziennika. Wyświetlany czas odpowiada strefie czasowej użytkownika wybranej w rozwijanym polu *User timezone* w *Account* > *Profile settings*.
User	Nazwa użytkownika organizacji, który wykonał akcję.
Role	Rola użytkownika, który wykonał akcję.
Node	Nazwa węzła powiązanego z akcją.
Action	Nazwa wykonanej akcji.
Description	Opis wykonanej akcji.

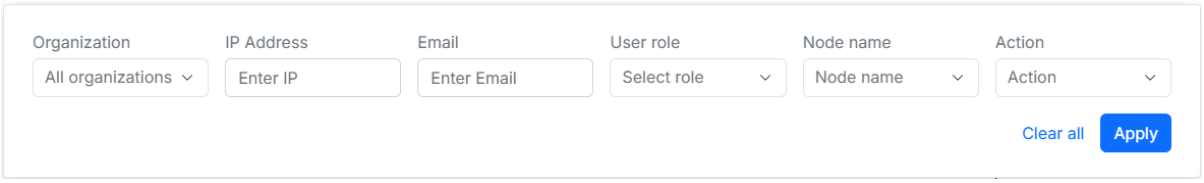
Nazwy kolumn można kliknąć, aby sortować rekordy rosnąco lub malejąco.

Przefiltrowane rekordy dziennika audytu można eksportować w formacie `.csv` za pomocą przycisku *Export*.

Korzystanie z filtrów

Użyj przycisku *Filter*, aby otworzyć filtr.

Aby wyświetlić przefiltrowane wyniki, kliknij *Apply* po wybraniu jednego lub kilku parametrów filtrowania. Aby wyczyścić filtr, kliknij *Clear all*.



Możliwe jest filtrowanie rekordów według jednego lub kilku parametrów. Dostępne są następujące parametry filtrowania:

Parameter	Description
Organization	Wybierz wszystkie organizacje lub konkretną.
IP Address	Określ adres IP.
Email	Określ adres e-mail przypisany do użytkownika.
User role	Wybierz wszystkie role lub konkretną: Owner, Admin albo User.
Node name	Wybierz wszystkie węzły lub konkretny.
Action	Wybierz konkretny typ akcji.

Dziennik audytu śledzi następujące akcje:

- *Node created*
- *Node deleted*
- *Node upgraded*
- *Node version updated*
- *Password reset*
- *Maintenance updated*
- *History setting updated*
- *Node expiration date extended*
- *Backup created*
- *Backup deleted*
- *Backup restored*
- *Backup updated*
- *Access filter added*
- *Access filter updated*
- *Access filter deleted*
- *Encryption enabled*
- *Encryption disabled*
- *Subscription cancelled*
- *Subscription renewed*
- *Subscription payment failed*
- *Payment requires confirmation*
- *Organization created*
- *Organization deleted*
- *Admin invited*
- *User invited*
- *Admin joined*
- *User joined*
- *Admin removed from organization*
- *User removed from organization*
- *Node added to organization*
- *Node removed from organization*
- *Node access granted*
- *Node access revoked*
- *User settings updated*
- *Payment created*
- *Payment completed*

- *Payment failed*
- *VAT charge applied*
- *VAT validation failed*
- *Billing info updated*
- *Payment method updated*
- *Payment method removed*
- *Prepaid node created*
- *Prepaid node updated*

Centrum dewelopera

Centrum deweloperskie

Rozpocznij tworzenie niestandardowych rozszerzeń dla Zabbix, korzystając ze wskazówek dotyczących wtyczek, modułów i widgetów.

Twórz na Zabbix

Wtyczki Przegląd sposobu tworzenia i zarządzania wtyczkami Zabbix w celu rozszerzenia funkcjonalności lub integracji z systemami zewnętrznymi.

Biblioteka Python dla Zabbix Zasoby do korzystania z oficjalnej biblioteki `zabbix_utils` do interakcji z API Zabbix i tworzenia integracji.

Moduły frontend Przewodniki i materiały referencyjne dotyczące tworzenia niestandardowych modułów, które rozszerzają lub modyfikują frontend Zabbix, aby dostosować go do konkretnych przypadków użycia.

Widgety Omówienie struktury i logiki widgetów wraz z instrukcjami tworzenia niestandardowych elementów pulpitu dostosowanych do Twoich potrzeb.

Moduły

Co to jest moduł frontendowy PHP?

- Moduł to jednostka posiadająca unikalny identyfikator, nazwę, opis, autora i inne pola zdefiniowane w pliku manifestu, wraz z PHP, JavaScript i innymi plikami znajdującymi się w jednym katalogu w katalogu *modules* instalacji frontendu Zabbix (na przykład *zabbix/ui/modules*).
- Moduł powinien spełniać proste zasady aby zapewnić poprawne działanie.
- Moduł musi zostać zainstalowany (rozpakowany) i włączony w interfejsie frontendu przez administratora.

Do czego można wykorzystać moduł

- Dodawanie nowych funkcjonalności poprzez niestandardowe sekcje frontendu;
- Tworzenie niestandardowych typów widgetów dashboardu (zobacz moduły widgetów);
- Zastąpienie lub rozszerzenie istniejącej funkcjonalności.

Do czego nie można wykorzystać modułu

- Rejestracja nowej metody API lub modyfikacja istniejącej.

Jak działają moduły

- Włączony moduł jest uruchamiany przy każdym żądaniu HTTP, przed wykonaniem kodu akcji.
- Moduł zarejestruje nowe akcje lub zdefiniuje na nowo istniejące.
- Moduł doda nowe sekcje front-endu i usunie lub zdefiniuje na nowo istniejące.
- Moduł będzie podłączał się do zdarzeń front-endu, takich jak onBeforeAction i onTerminate, jeśli będzie to konieczne.
- Żądana akcja jest ostatecznie wykonywana przez uruchomienie kodu akcji - domyślnego lub zdefiniowanego w module.

Gdzie dalej Niezależnie od tego, czy wolisz uczyć się poprzez działanie, czy najpierw przeczytać wytyczne, te strony zawierają informacje i kroki wymagane do zbudowania własnych modułów:

- Samouczki krok po kroku dotyczące pisania pierwszego modułu
- Struktura pliku modułu
- Szczegóły modułu widżetu
- Przykłady modułów do ponownego wykorzystania

Struktura plików modułu

Cały kod związany z modułem jest przechowywany w pojedynczym katalogu wewnątrz katalogu **modules** instalacji frontendu Zabbix (na przykład *zabbix/ui/modules*).

```
example_module_directory/ (wymagane)
   manifest.json (wymagane) Metadane i definicja akcji.
   Module.php Inicjalizacja modułu i obsługa zdarzeń.
   actions/   pliki kontrolera akcji.
       SomethingView.php
       SomethingCreate.php
       SomethingDelete.php
       data_export/
           ExportAsXml.php
           ExportAsExcel.php
   views/   pliki widoków.
       example.something.view.php
       example.something.delete.php
   assets/   wszelkie dodatkowe pliki do wykorzystania w widokach. Należy określić w pliku manifest.json.
       js/   skrypty JavaScript używane w widokach.
           example.something.view.js.php
       css/   pliki CSS używane w widokach.
           example.something.css
        image.png   obraz do użycia w widokach.
        example.something.file   dowolny plik do użycia w widokach.
```

Drzewo plików modułów

Tworzenie modułu Przykładowy proces tworzenia modułu składa się z następujących kroków (jeśli jest dostępny, kliknij nazwę pliku lub folderu, aby wyświetlić dodatkowe szczegóły dotyczące kroku):

1. Utwórz nowy katalog dla modułu w **zabbix/ui/modules/**.
2. Dodaj plik manifest.json z metadanymi modułu.
3. Utwórz folder *views* i zdefiniuj widok(y) modułu.
4. Utwórz folder *actions* i zdefiniuj akcję(y) modułu.
5. Utwórz plik Module.php (lub Widget.php dla widżetów pulpitu) i zdefiniuj reguły inicjalizacji i obsługi reguł.
6. Utwórz folder *assets* dla plików JavaScript (umieść w *assets/js*), stylów CSS (umieść w *assets/css*) lub innych dodatkowych plików.
7. Upewnij się, że określiłeś wymagane widoki, akcje i pliki zasobów w pliku manifest.json.
8. Zarejestruj moduł w interfejsie użytkownika Zabbix i zacznij go używać.

> **Note:**
> Możesz zarejestrować i włączyć moduł zaraz po utworzeniu pliku manifest.json. Po włączeniu modułu możesz natychmiast wyświetlić podgląd wszystkich zmian wprowadzonych do plików modułu, odświeżając interfejs użytkownika Zabbix.

manifest.json

Każdy moduł wymaga pliku manifest.json. Plik musi znajdować się w głównym katalogu (na przykład *zabbix/ui/modules/module_name/manifes*

Jako minimum plik manifest.json powinien określać następujące pola:

```
{
    "manifest_version": 2.0,
    "id": "my_ip_address",
    "name": "My IP Address",
    "namespace": "MyIPAddress",
    "version": "1.0"
}
```

Parametry obsługiwane w **manifest.json** (kliknij nazwę parametry, aby zobaczyć szczegółowy opis):

Parametr	Opis	Wymagane
manifest_version	Wersja modułu manifestu.	Tak
id	Unikalny identyfikator modułu.	
name	Nazwa modułu, która będzie wyświetlana w sekcji Administracja.	
namespace	Przestrzeń nazw PHP dla klas modułów.	
version	Wersja modułu.	
type	Typ modułu. Dla widgetu musi być ustawiony jako *widget*	Tak dla widgetów, w przeciwnym razie nie
widget	Konfiguracja widgetu. Używany tylko dla widgetów.	
actions	Akcje umożliwiające rejestrację w module.	
assets	Style CSS i pliki JavaScript do uwzględnienia.	No
author	Autor modułu.	
config	Domyślne wartości niestandardowych opcji modułów .	
description	Opis modułu.	
url	Link do opisu modułu.	

manifest_version

Wersja manifestu modułu. Obecnie obsługiwana wersja to **2.0**.

Typ: Double

Przykład:

```
"manifest_version": 2.0
```

ID

Identyfikator modułu. Musi być unikalny. Aby uniknąć przyszłych konfliktów w nazewnictwie, zaleca się używanie przedrostków dla modułów (nazwa autora, firmy lub jakakolwiek inna). Na przykład, jeśli moduł jest przykładem lekcji, a nazwa modułu to ”Mój moduł”, wówczas identyfikatorem będzie ”przykład_mój_moduł”.

Typ: String

Przykład:

```
"id": "przykład_mój_moduł"
```

name

Nazwa modułu, która będzie wyświetlana w sekcji Administracja.

Typ: String

Przykład:

```
"name": "My module"
```

namespace

Przestrzeń nazw PHP dla klas modułów.

Typ: String

Przykład:

```
"namespace": "ClockWidget"
```

version

Wersja modułu. Wersja zostanie wyświetlona w sekcji Administracja.

Typ: String

Przykład:

```
"version": "1.0"
```

type

Typ modułu. Wymagane w przypadku widgetów i musi być równe ”widget”.

Typ: String

Domyślne: ”module”

Przykład:

```
"type": "widget"
```

działania

Akcje do zarejestrowania w module. Wymagane jest zdefiniowanie klucza obiektu *class* dla każdej akcji, pozostałe klucze akcji są opcjonalne.

Typ: Obiekt

Obsługiwane klucze obiektu, jeśli *typ* to *module*:

- **write.your.action.name** (obiekt) - nazwa akcji, należy zapisać małymi literami [a-z], oddzielając słowa kropkami. Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji.
  - **layout** (string) - układ akcji. Obsługiwane wartości: *layout.json*, *layout.htmlpage* (domyślnie), *null*.
  - **view** (string) - widok akcji.

Przykład:

```
"actions": {
    "moduł.example.list": {
        "class": "ExampleList",
        "view": "example.list",
        "layout": "layout.htmlpage"
        }
    }
```

Obsługiwane klucze obiektu, jeśli *type* to *widget*:

- **widget.{id}.view** (object) - nazwa pliku i klasy dla widoku widgetu. Zamień **{id}** na wartość id widżetu (na przykład *widget.example_clock.view*). Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji dla trybu widoku widgetu w celu rozszerzenia domyślnej klasy CControllerDashboardWidgetView. Plik źródłowy klasy musi znajdować się w katalogu *actions*.
  - **view** (string) - widok widżetu. Musi znajdować się w katalogu *views*. Jeśli plik widoku to *widget.view.php*, co jest wartością domyślną, można pozostawić pustą wartość. Jeśli używasz innej nazwy, podaj ją tutaj.
- **widget.{id}.edit** (object) - nazwa pliku widoku konfiguracji widgetu. Zamień **{id}** na wartość id widżetu (na przykład *widget.example_clock.edit*). Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji dla trybu widoku konfiguracji widgetu. Plik źródłowy klasy musi znajdować się w katalogu *actions*.
  - **view** (string) - widok konfiguracji widgetu. Musi znajdować się w katalogu *views*. Jeśli plik widoku to *widget.edit.php*, co jest wartością domyślną, można pozostawić pustą wartość. Jeśli używasz innej nazwy, podaj ją tutaj.

Przykład:

```
"actions": {
    "widget.tophosts.view": {
        "class": "WidgetView"
    },
    "widget.tophosts.column.edit": {
        "class": "ColumnEdit",
        "view": "column.edit",
        "layout": "layout.json"
    }
}
```

assets

Style CSS i pliki JavaScript do uwzględnienia.

Typ: Object

Obsługiwane klucze obiektów:

- **css** (tablica) - pliki CSS do dołączenia. Pliki muszą znajdować się w katalogu *assets/css*.
- **js** (tablica) - pliki JavaScript do dołączenia. Pliki muszą znajdować się w katalogu *assets/js*.

Przykład:

```
"assets": {
    "css": ["widget.css"],
    "js": ["class.widget.js"]
}
```

author

Autor modułu. Autor zostanie wyświetlony w sekcji Administracja.

Typ: String

Przykład:

```
"author": "John Smith"
```

konfiguracja

Domyślne wartości opcji modułu. Obiekt może zawierać dowolne klucze niestandardowe. Jeśli podane, wartości te zostaną zapisane do bazy danych podczas rejestracji modułu. Nowe zmienne dodane później zostaną zapisane przy pierwszym wywołaniu. Po zapisaniu, wartości zmiennych można zmieniać wyłącznie bezpośrednio w bazie danych.

Typ: Obiekt

Przykład:

```
„config”: {
    "username": "Administrator",
    "password": "",
    "auth_url": "https://example.com/auth"
}
```

description

Opis modułu.

Typ: String

Przykład:

```
"description": "To jest widget zegara."
```

widget

Konfiguracja widgetu. Używana, jeśli *type* jest ustawiony na *widget*.

Typ: Object

Obsługiwane klucze obiektu:

- **name** (string) - używana na liście widgetów oraz jako domyślny nagłówek. Jeśli jest pusta, zostanie użyty parametr „name” z modułu.
- **size** (object) - domyślne wymiary widgetu. Obsługuje klucze:
  - *width* (integer) - domyślna szerokość widgetu.
  - *height* (integer) - domyślna wysokość widgetu.
- **form_class** (string) - klasa z formularzem pól widgetu. Musi znajdować się w katalogu *includes*. Jeśli klasą jest *WidgetForm.php*, oczekiwana domyślnie, ten parametr może zostać pominięty. W przypadku użycia innej nazwy należy ją tutaj określić.
- **js_class** (string) - nazwa klasy JavaScript dla trybu widoku widgetu, rozszerzającej domyślną klasę CWidget. Klasa zostanie załadowana wraz z pulpitem. Plik źródłowy klasy musi znajdować się w katalogu assets/js. Zobacz także: assets.

- **use_time_selector** (boolean) - określa, czy widget wymaga selektora czasu pulpitu. Obsługiwane wartości: *true*, *false* (domyślnie).
- **refresh_rate** (integer) - częstotliwość odświeżania widgetu w sekundach (domyślnie: 60).

Przykład:

```
"widget": {
    "name": "",
    "size": {
        "width": 12,
        "height": 5
    },
    "form_class": "WidgetForm",
    "js_class": "CWidget",
    "use_time_selector": false,
    "refresh_rate": 60
}
```

url

Link do opisu modułu. W przypadku widżetów ten link zostanie otwarty po kliknięciu ikony pomocy ? w oknie Dodaj widżet lub Edytuj widżet. Jeśli nie określono **url**, kliknięcie ikony pomocy spowoduje otwarcie strony ogólnej Widżety pulpitu nawigacyjnego.

Typ: Łańcuch znaków

Przykład:

```
"url": "http://example.com"
```

Akcje

Za 'logikę biznesową' modułu odpowiadają akcje. Akcja zwykle składa się z controller i action view.

Moduł może:

- Wywołać akcje, które są już zdefiniowane w interfejsie frontowym Zabbix.
- Zastąpić akcje domyślne akcjami niestandardowymi.
- Zdefiniować zupełnie nowe akcje.

Aby zastąpić domyślne zachowanie akcji jakimś niestandardowym zachowaniem, zdefiniuj akcję o tej samej nazwie w konfiguracji modułu. Kiedy akcja zostanie wywołana, zamiast domyślnej akcji Zabbix'a zostanie wykonana akcja modułu.

Pliki akcji powinny być przechowywane w folderze *akcje*. Akcje muszą być określone w pliku manifest.json.

Controller

Przepływ pracy kontrolera akcji:

1. Sprawdź, czy wszystkie parametry przekazane w żądaniu HTTP są poprawne:
   - Wywołaj metodę *checkInput()* kontrolera;
   - Użyj reguł walidacji zdefiniowanych w CNewValidator.php;
   - Wywołaj metodę *validateInput()*.
2. Sprawdź uprawnienia użytkownika.
3. Przygotuj dane zgodnie z przekazanymi parametrami: jeśli *checkInput()* zwróci true, Zabbix wywoła metodę *doAction()* kontrolera.
4. Przygotuj tablicę **$data** dla widoku. Użyj *CControllerResponseData* oraz metody *setResponse()*, aby zapisać odpowiedź w tablicy **$data**.

Przykład:

```
/**
 * Validate input parameters.
 *
 * @return bool
 */
protected function checkInput(): bool {
    $ret = $this->validateInput([
```

```
            'status' => 'in '.implode(',', [HOST_STATUS_MONITORED, HOST_STATUS_NOT_MONITORED])
        ]);

        if (!$ret) {
            $this->setResponse(new CControllerResponseFatal());
        }

        return $ret;
    }

    /**
     * Check user permissions.
     *
     * @return bool
     */
    protected function checkPermissions() {
        return $this->getUserType() >= USER_TYPE_ZABBIX_ADMIN;
    }

    /**
     * Execute action and generate response object.
     */
    protected function doAction(): void {
        $data = [
            'hosts_count' => API::Host()->get([
                'countOutput' => true,
                'filter' => [
                    'status' => $this->getInput('status')
                ]
            ])
        ];

        $this->setResponse(new CControllerResponseData($data));
    }
```

Note:
Pełną listę dostępnych klas kontrolerów można zobaczyć w kodzie źródłowym Zabbixa.

Widoki

Plik widoku otrzymuje dane z kontrolera, a następnie przygotowuje ich wygląd HTML.

Note:
Definiowanie widoku(-ów) dla modułu frontend jest opcjonalne, chyba że moduł jest widżetem.
 Widżety pulpitu nawigacyjnego potrzebują co najmniej dwóch widoków: jednego dla trybu edycji i jednego dla trybu podglądu (powinny być przechowywane w katalogu *views*).

W widoku można używać predefiniowanych klas HTML Zabbixa (z */zabbix/ui/include/classes/html*) oraz dodawać nowe klasy HTML i CSS. Nowe klasy powinny być przechowywane w folderze *assets* modułu.

Przykład:

```
...
(new CColHeader(_('Name')))
```

Spowoduje to dodanie nowej kolumny o nazwie *Name* oraz stylizację górnego wiersza tabeli tak jak na innych stronach Zabbixa.

Widok akcji

To jest plik referencyjny do definiowania widoku akcji.

```
<?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();
```

Aktywa

Folder *assets* może zawierać dowolne pliki i podfoldery, które nie należą do innych katalogów. Możesz go używać do:

- Styli JavaScript (muszą znajdować się w *assets/js*);
- Styli CSS (muszą znajdować się w *assets/css*);
- Obrazów;
- Czcionek;
- Wszystkiego innego, co chcesz uwzględnić.

assets/js

Katalog *assets/js* jest zastrzeżony i powinien zawierać wyłącznie pliki JavaScript. Aby widget mógł używać tych plików, określ je w pliku *manifest.json*.

Na przykład:

```
"assets": {
    "js": ["class.widget.js"]
}
```

assets/css

Katalog *assets/css* jest zastrzeżony i powinien zawierać wyłącznie pliki stylu CSS. Aby widget mógł używać tych plików, określ je w pliku *manifest.json*.

Na przykład:

```
"assets": {
    "css": ["mywidget.css"]
}
```

Style CSS

Pliki CSS mogą zawierać niestandardowy atrybut `theme`, aby zdefiniować różne style dla określonych motywów frontend.

Dostępne motywy i ich wartości atrybutów:

- **Blue** - [theme='blue-theme']
- **Dark** - [theme='dark-theme']
- **High-contrast light** - [theme='hc-light']
- **High-contrast dark** - [theme='hc-dark']

Przykład:

```
.widget {
    background-color: red;
}

[theme='dark-theme'] .widget {
    background-color: green;
}
```

Zarejestruj nowy moduł

Ta sekcja wyjaśnia jak dodać nowy moduł do frontendu Zabbix.

Wymagania wstępne Przed kontynuowaniem upewnij się, że:

- Moduł znajduje się w katalogu *moduły* (modules) instalacji front-endu Zabbix (na przykład *zabbix/ui/modules*).
- Moduł ma co najmniej podstawową wersję pliku manifest.json.
- Masz dostęp do sekcji menu Administracja w Zabbix (wymaga roli użytkownika Super administrator).

> **Note:**
> The frontend will not install or even recognize incompatible modules.

Dodawanie modułu Otwórz stronę *Administracja* → *Ogólne* → *Moduły* i naciśnij *Skanuj katalog*.



Znajdź swój moduł na liście i aktywuj go.

Aby aktywować moduł, naciśnij hiperłącze *Wyłączony* - stan modułu zmieni się na *Włączony*.

Po naciśnięciu na nazwę modułu można wyświetlić dodatkowe informacje o module, takie jak autor, wersja czy krótki opis (jeśli został zdefiniowany w manifeście).

Podgląd widżetu Moduły widżetów, po dodaniu, stają się natychmiast widoczne na liście widżetów pulpitu.

Możesz otworzyć pulpit, przełączyć go w tryb edycji i dodać widżet do pulpitu w standardowy sposób.

Gdy wprowadzisz zmiany w widżecie, odśwież pulpit, aby zobaczyć, jak widżet wygląda z najnowszymi aktualizacjami.

Widżety

Widżety to moduły front-endu Zabbix używane w pulpitach nawigacyjnych. O ile nie zaznaczono inaczej, wszystkie wytyczne dotyczące modułów mają zastosowanie również do widżetów.

Jednak widżet różni się znacząco od modułu. Aby zbudować widżet:

- określ typ "widget" w pliku manifest.json ("type": "widget");
- uwzględnij co najmniej dwa widoki: jeden dla trybu prezentacji widżetu i jeden dla trybu konfiguracji widżetu (example.widget.view.php i example.widget.edit.php);
- i kontroler dla prezentacji widżetu (WidgetView.php);
- użyj i rozszerz domyślne klasy widżetów.

Konfiguracja

Konfiguracja widżetu jest częścią modułu widżetu, która pozwala użytkownikowi definiować ustawienia widżetu dla prezentacja. Ta strona opisuje klasy, które mogą być użyte do stworzenia formularza konfiguracji widżetu z niestandardowymi polami.

Widżet

Główna klasa widżetu, rozszerzająca bazową klasę wszystkich widżetów pulpitu - *CWidget*. Wymagana do nadpisania domyślnego zachowania widżetu.

Klasa Widget powinna znajdować się w katalogu głównym widżetu (na przykład, zabbix/ui/modules/my_custom_widget).

Przykład Widget.php

```
<?php

namespace Modules\MyCustomWidget;

use Zabbix\Core\CWidget;

class Widget extends CWidget {

    public const MY_CONSTANT = 0;
```

```
    public function getTranslationStrings(): array {
        return [
            'class.widget.js' => [
                'No data' => _('No data')
            ]
        ];
    }
}
```

WidgetForm

Klasa *WidgetForm* rozszerza domyślną klasę *CWidgetForm* i zawiera zestaw pól *CWidgetField*, które są wymagane do zdefiniowania struktury przechowywania konfiguracji widgetu w bazie danych oraz obsługi walidacji danych wejściowych.

Klasa *WidgetForm* powinna znajdować się w katalogu *includes*. Jeśli klasa ma inną nazwę, należy ją określić w parametrze *widget/form_class* w pliku *manifest.json*.

Przykład pliku includes/WidgetForm.php

```
<?php

namespace Modules\MyCustomWidget\Includes;

use Modules\MyCustomWidget\Widget;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldMultiSelectItem,
    CWidgetFieldTextBox,
    CWidgetFieldColor
};


class WidgetForm extends CWidgetForm {

    public const DEFAULT_COLOR_PALETTE = [
        'FF465C', 'B0AF07', '0EC9AC', '524BBC', 'ED1248', 'D1E754', '2AB5FF', '385CC7', 'EC1594', 'BAE37D',
        '6AC8FF', 'EE2B29', '3CA20D', '6F4BBC', '00A1FF', 'F3601B', '1CAE59', '45CFDB', '894BBC', '6D6D6D'
    ];

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            )
            ->addField(
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            );
    }
}
```

CWidgetFormView

Klasa *CWidgetFormView* jest wymagana do określenia logiki prezentacji pól zdefiniowanych w klasie *WidgetForm*, określając ich wygląd i zachowanie podczas renderowania w widoku konfiguracji.

Klasa *CWidgetFormView* obsługuje następujące metody:

- *addField()* - przyjmuje jako parametr instancję klasy *CWidgetFieldView*; każda klasa *CWidgetField* ma odpowiadającą jej klasę *CWidgetFieldView* do użycia w widoku konfiguracji widgetu.
- *addFieldset()* - przyjmuje instancję klasy *CWidgetFieldsGroupView*, która łączy pola w zwijany kontener.
- *addFieldsGroup()* - przyjmuje instancję *CWidgetFormFieldsetCollapsibleView*, która wizualnie (za pomocą obramowania) łączy pola w grupę.
- *includeJsFile()* - umożliwia dodanie pliku JavaScript do widoku konfiguracji widgetu.
- *addJavaScript()* - umożliwia dodanie osadzonego kodu JavaScript, który zostanie wykonany natychmiast po załadowaniu widoku konfiguracji widgetu.

Klasa *CWidgetFormView* powinna znajdować się w katalogu *views*.

Przykład views/widget.edit.php

```
<?php

/**
 * Widok formularza mojego niestandardowego widgetu.
 *
 * @var CView $this
 * @var array $data
 */

use Modules\MyCustomWidget\Includes\WidgetForm;

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))->setPopupParameter('numeric', tru
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Konfiguracja zaawansowana')))
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('my_custom_widget_form.init('.json_encode([
        'color_palette' => WidgetForm::DEFAULT_COLOR_PALETTE
    ]).');')
    ->show();
```

JavaScript

Klasa JavaScript może być użyta do dodania dynamicznego zachowania i interaktywności do widoku konfiguracji widżetu. Na przykład, można zainicjować selektor kolorów, zdefiniowany w klasie *CWidgetFormView*.

Klasa JavaScript powinna zostać załadowana wraz z formularzem, dlatego należy odwołać się do niej w klasie *CWidgetFormView* za pomocą metod *includeJsFile()* i *addJavaScript()*.

W poniższym przykładzie instancja klasy singleton jest natychmiast tworzona i przechowywana pod nazwą *window.my_custom_widget_form*. W ten sposób otwarcie formularza po raz drugi spowoduje ponowne utworzenie instancji.

Klasa JavaScript powinna znajdować się w katalogu *views*.

views/widget.edit.js.php example

```
<?php

use Modules\MyCustomWidget\Widget;

?>

window.my_custom_widget_form = new class {

    init({color_palette}) {
        colorPalette.setThemeColors(color_palette);
```

```
        for (const colorpicker of jQuery('.<?= ZBX_STYLE_COLOR_PICKER ?> input')) {
            jQuery(colorpicker).colorpicker();
        }

        const overlay = overlays_stack.getById('widget_properties');

        for (const event of ['overlay.reload', 'overlay.close']) {
            overlay.$dialogue[0].addEventListener(event, () => { jQuery.colorpicker('hide'); });
        }
    }
};
```

CWidgetField

Klasa *CWidgetField* jest klasą bazową, po której dziedziczą wszystkie klasy pól formularza (*CWidgetFieldCheckBox*, *CWidgetFieldTextArea*, *CWidgetFieldRadioButtonList* itd.). Klasy rozszerzające *CWidgetField* odpowiadają za odbieranie, zapisywanie i walidację wartości konfiguracji widżetu.

Dostępne są następujące klasy *CWidgetField*.

CWidgetField class	Database field type	Description
CWidgetFieldCheckBox	int32	Pojedyncze pole wyboru.
CWidgetFieldCheckBoxList	array of int32	Wiele pól wyboru w ramach jednego pola konfiguracji.
CWidgetFieldColor	string	Pole wyboru koloru.
CWidgetFieldDatePicker	string	Pole wyboru daty.
CWidgetFieldIntegerBox	int32	Pole do wprowadzania liczby całkowitej. Może służyć do konfigurowania wartości minimalnej i maksymalnej.
CWidgetFieldLatLng	string	Pole tekstowe umożliwiające wprowadzenie oddzielonych przecinkami wartości: szerokości geograficznej, długości geograficznej i poziomu powiększenia mapy.
CWidgetFieldMultiSelect	-	Klasa bazowa dla pól wielokrotnego wyboru, rozszerzana przez wszystkie klasy *CWidgetFieldMultiSelect**.
CWidgetFieldMultiSelectAction	ID	Pole wielokrotnego wyboru do wybierania działań z listy działań zdefiniowanych w *Alerts → Actions*.
CWidgetFieldMultiSelectGraph	ID	Pole wielokrotnego wyboru do wybierania niestandardowych wykresów.
CWidgetFieldMultiSelectGraphPrototype	ID	Pole wielokrotnego wyboru do wybierania prototypów niestandardowych wykresów.
CWidgetFieldMultiSelectGroup	ID	Pole wielokrotnego wyboru do wybierania grup hostów.
CWidgetFieldMultiSelectHost	ID	Pole wielokrotnego wyboru do wybierania hostów.
CWidgetFieldMultiSelectHostInventory	array of int32	Pole wielokrotnego wyboru do wybierania pól inwentarza hosta.
CWidgetFieldMultiSelectItem	ID	Pole wielokrotnego wyboru do wybierania pozycji.
CWidgetFieldMultiSelectItemPrototype	ID	Pole wielokrotnego wyboru do wybierania prototypów pozycji.
CWidgetFieldMultiSelectMap	ID	Pole wielokrotnego wyboru do wybierania map.
CWidgetFieldMultiSelectMediaType	ID	Pole wielokrotnego wyboru do wybierania typów mediów.
CWidgetFieldMultiSelectOverrideHost	ID	Pole wielokrotnego wyboru do wybierania źródła danych dla hostów z dashboardu lub innego widżetu.
CWidgetFieldMultiSelectService	ID	Pole wielokrotnego wyboru do wybierania usług.
CWidgetFieldMultiSelectSla	ID	Pole wielokrotnego wyboru do wybierania SLA.
CWidgetFieldMultiSelectUser	ID	Pole wielokrotnego wyboru do wybierania użytkowników.
CWidgetFieldNavTree	array of (multiple mixed)	Przechowuje strukturę drzewa nawigacji map do użycia w widżecie *Map navigation tree*.
CWidgetFieldNumericBox	string	Pole do wprowadzania liczby zmiennoprzecinkowej.
CWidgetFieldPatternSelect	-	Klasa bazowa dla pól wyboru wzorców, rozszerzana przez wszystkie klasy *CWidgetPatternSelect**.
CWidgetFieldPatternSelectHost	string	Pole wielokrotnego wyboru do wybierania hostów. Obsługuje wzorce nazw hostów.
CWidgetFieldPatternSelectItem	string	Pole wielokrotnego wyboru do wybierania pozycji. Obsługuje wzorce nazw pozycji.
CWidgetFieldRadioButtonList	int32	Grupa przycisków radiowych składająca się z jednego lub większej liczby przycisków radiowych.
CWidgetFieldRangeControl	int32	Suwak do wybierania wartości typu całkowitego.

CWidgetField class	Database field type	Description
CWidgetFieldReference	string	Przechowuje unikalny identyfikator tego widżetu na dashboardzie; używany do komunikacji między widżetami.
CWidgetFieldSelect	int32	Rozwijane pole wyboru.
CWidgetFieldSeverities	array of int32	*CWidgetFieldCheckBoxList* z ustawionymi domyślnie poziomami ważności wyzwalaczy.
CWidgetFieldSparkline	array of (multiple mixed)	Umożliwia konfigurację wykresu sparkline.
CWidgetFieldTags	array of (string, int32, string)	Umożliwia konfigurację jednego lub większej liczby wierszy filtra tagów.
CWidgetFieldTextArea	string	Pole tekstowe do wprowadzania wielowierszowego tekstu.
CWidgetFieldTextBox	string	Pole tekstowe do wprowadzania jednoliniowego tekstu.
CWidgetFieldThresholds	array of (string, string)	Umożliwia konfigurację par kolorów i liczb.
CWidgetFieldTimePeriod	array of string	Pole wyboru okresu czasu.
CWidgetFieldTimeZone	string	Rozwijana lista stref czasowych.
CWidgetFieldUrl	string	Pole tekstowe umożliwiające wprowadzanie adresów URL.

Note:
Maksymalna długość pól `int32` wynosi 10 znaków, pól `ID` 20 znaków, a pól `string` 2048 znaków. W przypadku typów złożonych (np. `array of (string, int32, string)`) limit dotyczy każdego elementu osobno. Ograniczenia długości specyficzne dla danego pola można ustawić za pomocą metod takich jak `setMaxLength()`.

Prezentacja

Ta strona opisuje komponenty, których można użyć do utworzenia widoku prezentacji widżetu. Widok prezentacji widżetu to część widżetu, która odbiera dane zgodnie z jego konfiguracją i wyświetla je na pulpicie nawigacyjnym w kontenerze.

Widok prezentacji składa się z trzech części:

- Widget action
- Widok widżetu
- JavaScript

Akcja widgetu

Klasa akcji widgetu (*WidgetView*) zawiera metody operacji na widgetach w trybie widoku prezentacji. Większość akcji widgetów używa domyślnej klasy kontrolera *CControllerDashboardWidgetView* i/lub ją rozszerza.

Klasa akcji widgetu powinna znajdować się w katalogu *actions* i być określona w parametrze *actions* (*actions/widget.{id}.view/class*) w pliku *manifest.json*.

Przykład actions/WidgetView.php (zaimplementowany w natywnym widżecie Zabbix *Informacje o systemie*)

```
class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getDefaultName()),
            'system_info' => CSystemInfoHelper::getData(),
            'info_type' => $this->fields_values['info_type'],
            'user_type' => CWebUser::getType(),
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

Widok widgetu

Klasa widoku widgetu (*CWidgetView*) odpowiada za budowanie widoku prezentacji widgetu.

Klasa widoku widgetu powinna znajdować się w katalogu *views*. Jeśli plik zawierający klasę widoku widgetu ma inną nazwę niż domyślna (*widget.view.php*), należy ją określić w parametrze *actions* pliku *manifest.json* (*actions/widget.{id}.view/view*).

Przykład views/widget.view.php

```php
<?php

/**
 * Мój niestandardowy widok widgetu.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['name'])
    )
    ->show();
```

JavaScript

Klasa JavaScript odpowiada za określanie zachowania widżetu, takiego jak aktualizowanie danych widżetu, zmiana rozmiaru widżetu, wyświetlanie elementów widżetu itp.

Wszystkie operacje JavaScript używają i/lub rozszerzają bazową klasę JavaScript wszystkich widżetów pulpitu - *CWidget*. Klasa *CWidget* zawiera zestaw metod z domyślną implementacją zachowania widżetu. W zależności od złożoności widżetu metody te mogą być używane bez zmian lub rozszerzane.

Klasa *CWidget* zawiera następujące metody:

- Metody definiujące cykl życia widżetu: *onInitialize()*, *onStart()*, *onActivate()*, *onDeactivate()*, *onDestroy()*, *onEdit()*.
- Metody obsługujące aktualizację i wyświetlanie danych widżetu: *promiseUpdate()*, *getUpdateRequestData()*, *processUpdateResponse(response)*, *processUpdateErrorResponse(error)*, *setContents(response)*.
- Metody modyfikujące wygląd widżetu: *onResize()*, *hasPadding()*.

Klasa JavaScript powinna znajdować się w katalogu *assets/js* i być określona w parametrze *assets* (*assets/js*) w pliku *manifest.json*.

Metody cyklu życia

Metody cyklu życia widżetu są wywoływane przez pulpit nawigacyjny oraz na różnych etapach cyklu życia widżetu podczas jego istnienia w pulpicie nawigacyjnym.

Metoda ***onInitialize()*** definiuje początkowy stan i/lub wartości widżetu, bez wykonywania jakichkolwiek manipulacji HTML lub danymi. Metoda ta jest wywoływana, gdy tworzony jest widżet (tworzona jest instancja obiektu widżetu), zazwyczaj poprzez dodanie widżetu do strony pulpitu nawigacyjnego lub załadowanie strony pulpitu nawigacyjnego.

Przykład:

```
onInitialize() {
    this._time_offset = 0;
    this._interval_id = null;
    this._clock_type = CWidgetClock.TYPE_ANALOG;
    this._time_zone = null;
    this._show_seconds = true;
    this._time_format = 0;
    this._tzone_format = 0;
    this._show = [];
    this._has_contents = false;
    this._is_enabled = true;
}
```

Metoda ***onStart()*** definiuje strukturę HTML widgetu, bez wykonywania jakiejkolwiek manipulacji danymi. Ta metoda jest wywoływana przed pierwszą aktywacją strony pulpitu, czyli zanim pulpit i jego widgety zostaną w pełni wyświetlone użytkownikowi.

Przykład:

```
onStart() {
    this._events.resize = () => {
        const padding = 25;
        const header_height = this._view_mode === ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER
            ? 0
            : this._header.offsetHeight;
```

```
        this._target.style.setProperty(
            '--content-height',
            `${this._cell_height * this._pos.height - padding * 2 - header_height}px`
        );
    }
}
```

Metoda ***onActivate()*** uaktywnia widget i czyni go interaktywnym poprzez włączenie niestandardowych detektorów zdarzeń (służących do reagowania na działania użytkownika) oraz zainicjowanie cyklu aktualizacji widgetu (aby utrzymywać jego zawartość na bieżąco). Ta metoda jest wywoływana po aktywacji strony pulpitu, czyli gdy zostanie ona w pełni wyświetlona w interfejsie użytkownika.

Należy pamiętać, że przed wywołaniem metody *onActivate()* widget znajduje się w stanie nieaktywnym (`WIDGET_STATE_INACTIVE`). Po pomyślnym wywołaniu widget przechodzi do stanu aktywnego (`WIDGET_STATE_ACTIVE`). W stanie aktywnym widget reaguje na działania, nasłuchuje zdarzeń, okresowo aktualizuje swoją zawartość i może współdziałać z innymi widgetami.

Przykład:

```
onActivate() {
    this._startClock();

    this._resize_observer = new ResizeObserver(this._events.resize);
    this._resize_observer.observe(this._target);
}
```

Metoda ***onDeactivate()*** zatrzymuje wszelką aktywność i interaktywność widżetu przez dezaktywację niestandardowych nasłuchiwaczy zdarzeń oraz zatrzymanie cyklu aktualizacji widżetu. Ta metoda jest wywoływana, gdy strona pulpitu jest dezaktywowana, to znaczy po przełączeniu na inną stronę lub usunięciu, albo gdy widżet zostanie usunięty ze strony pulpitu.

Należy pamiętać, że przed wywołaniem metody *onDeactivate()* widżet znajduje się w stanie aktywnym (`WIDGET_STATE_ACTIVE`). Po pomyślnym wywołaniu widżet przechodzi do stanu nieaktywnego (`WIDGET_STATE_INACTIVE`).

Przykład:

```
onDeactivate() {
    this._stopClock();
    this._resize_observer.disconnect();
}
```

Metoda ***onDestroy()*** wykonuje zadania porządkowe przed usunięciem widgetu z dashboardu, co może obejmować zamknięcie połączenia z bazą danych ustanowionego podczas inicjalizacji widgetu, usunięcie danych tymczasowych w celu zwolnienia pamięci systemowej i uniknięcia wycieków zasobów, wyrejestrowanie detektorów zdarzeń związanych ze zdarzeniami zmiany rozmiaru lub kliknięciami przycisków, aby zapobiec niepotrzebnej obsłudze zdarzeń i wyciekom pamięci itd. Ta metoda jest wywoływana, gdy widget lub strona dashboardu, która go zawiera, zostaje usunięta.

Należy pamiętać, że przed wywołaniem metody *onDestroy()* widget w stanie aktywnym (`WIDGET_STATE_ACTIVE`) jest zawsze dezaktywowany przez wywołanie metody *onDeactivate()*.

Przykład:

```
onDestroy() {
    if (this._filter_widget) {
        this._filter_widget.off(CWidgetMap.WIDGET_NAVTREE_EVENT_MARK, this._events.mark);
        this._filter_widget.off(CWidgetMap.WIDGET_NAVTREE_EVENT_SELECT, this._events.select);
    }
}
```

Metoda ***onEdit()*** definiuje wygląd i zachowanie widżetu, gdy dashboard przechodzi w tryb edycji. Ta metoda jest wywoływana, gdy dashboard przechodzi w tryb edycji, zwykle wtedy, gdy użytkownik korzysta z przycisku *Edit* widżetu lub przycisku *Edit dashboard* na dashboardzie.

Przykład:

```
onEdit() {
    this._deactivateGraph();
}
```

Metoda procesu aktualizacji

Metoda procesu aktualizacji widżetu są odpowiedzialne za pobieranie zaktualizowanych danych z serwera Zabbix lub dowolnego innego źródła danych i wyświetlanie ich w widżecie.

Metoda ***promiseUpdate()*** inicjuje proces aktualizacji danych przez pobieranie danych, zazwyczaj za pomocą żądań web lub wywołań API. Ta metoda jest wywoływana, gdy strona pulpitu jest wyświetlana, a następnie okresowo, aż strona pulpitu zostanie przełączona na inną stronę pulpitu.

Poniżej przedstawiono przykład domyślnej implementacji metody *promiseUpdate()* używanej przez większość natywnych widgetów Zabbix. W domyślnej implementacji metoda *promiseUpdate()* stosuje ogólny wzorzec pobierania danych z serwera. Tworzy nowy obiekt `Curl` z odpowiednim adresem URL i parametrami żądania, wysyła żądanie `POST` za pomocą metody *fetch()* z obiektem danych zbudowanym przez metodę *getUpdateRequestData()*, a następnie odpowiednio przetwarza odpowiedź (lub odpowiedź błędu) za pomocą *processUpdateResponse(response)* lub *processUpdateErrorResponse(error)*. Ta implementacja jest odpowiednia dla większości widgetów, ponieważ zazwyczaj pobierają one dane w formacie JSON i obsługują je w spójny sposób.

```
promiseUpdate() {
    const curl = new Curl('zabbix.php');

    curl.setArgument('action', `widget.${this._type}.view`);

    return fetch(curl.getUrl(), {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify(this.getUpdateRequestData()),
        signal: this._update_abort_controller.signal
    })
        .then((response) => response.json())
        .then((response) => {
            if ('error' in response) {
                this.processUpdateErrorResponse(response.error);

                return;
            }

            this.processUpdateResponse(response);
        });
    }
```

Metoda ***getUpdateRequestData()*** przygotowuje dane żądania do serwera na potrzeby aktualizacji widżetu, zbierając różne właściwości i odpowiadające im wartości (identyfikatory widżetu, ustawienia filtrów, zakresy czasu itp.) ze stanu i konfiguracji widżetu, a następnie tworząc obiekt danych reprezentujący informacje wymagane do wysłania do serwera w żądaniu aktualizacji. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
getUpdateRequestData() {
    return {
        templateid: this._dashboard.templateid ?? undefined,
        dashboardid: this._dashboard.dashboardid ?? undefined,
        widgetid: this._widgetid ?? undefined,
        name: this._name !== '' ? this._name : undefined,
        fields: Object.keys(this._fields).length > 0 ? this._fields : undefined,
        view_mode: this._view_mode,
        edit_mode: this._is_edit_mode ? 1 : 0,
        dynamic_hostid: this._dashboard.templateid !== null || this.supportsDynamicHosts()
            ? (this._dynamic_hostid ?? undefined)
            : undefined,
        ...this._contents_size
    };
}
```

Metoda ***processUpdateResponse(response)*** obsługuje odpowiedź otrzymaną z serwera po żądaniu aktualizacji i, jeśli proces aktualizacji zakończył się powodzeniem i bez błędów, czyści dane widżetu oraz wyświetla nową zawartość za pomocą metody *setContents()*. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
processUpdateResponse(response) {
    this._setHeaderName(response.name);

    this._updateMessages(response.messages);
    this._updateInfo(response.info);
    this._updateDebug(response.debug);

    this.setContents(response);
}
```

Metoda ***processUpdateErrorResponse(error)*** obsługuje odpowiedź otrzymaną z serwera po żądaniu aktualizacji, jeśli odpowiedź zawiera błąd, i wyświetla komunikat/-y o błędzie. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
processUpdateErrorResponse(error) {
    this._updateMessages(error.messages, error.title);
}
```

Metoda ***setContents(response)*** wyświetla zawartość widżetu, jeśli proces aktualizacji widżetu zakończył się pomyślnie i bez błędów, co może obejmować manipulowanie elementami DOM, aktualizowanie komponentów interfejsu użytkownika, stosowanie stylów lub formatowania itp. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *processUpdateResponse(response)*, czyli podczas obsługi odpowiedzi otrzymanej z serwera po żądaniu aktualizacji.

Domyślna implementacja:

```
setContents(response) {
    this._body.innerHTML = response.body ?? '';
}
```

Metoda modyfikacji prezentacji

Metoda modyfikacji prezentacji widżetu jest odpowiedzialna za modyfikację wyglądu widżetu.

Metoda ***onResize()*** odpowiada za dostosowanie elementów wizualnych widgetu do nowego rozmiaru widgetu, co może obejmować zmianę rozmieszczenia elementów, dostosowanie wymiarów elementów, obcinanie tekstu, implementację lazy loading w celu poprawy responsywności podczas zmiany rozmiaru itp. Ta metoda jest wywoływana, gdy widget zmienia rozmiar, na przykład gdy użytkownik ręcznie zmienia rozmiar widgetu lub gdy zmieniany jest rozmiar okna przeglądarki.

Przykład:

```
onResize() {
    if (this.getState() === WIDGET_STATE_ACTIVE) {
        this._startUpdating();
    }
}
```

Metoda ***hasPadding()*** odpowiada za zastosowanie pionowego dopełnienia 8 px u dołu widżetu, gdy jest on skonfigurowany tak, aby wyświetlać nagłówek. Ta metoda jest wywoływana po aktywowaniu strony pulpitu, czyli gdy staje się ona stroną wyświetlaną w interfejsie użytkownika.

Implementacja domyślna:

```
hasPadding() {
    return this.getViewMode() !== ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER;
}
```

W przypadku niektórych widżetów konieczne jest wykorzystanie całej dostępnej przestrzeni widżetu do skonfigurowania na przykład niestandardowego koloru tła. Poniżej przedstawiono przykład implementacji metody *hasPadding()* używanej w natywnym widżecie Zabbix *Wartość pozycji*.

```
hasPadding() {
    return false;
}
```

Poradniki

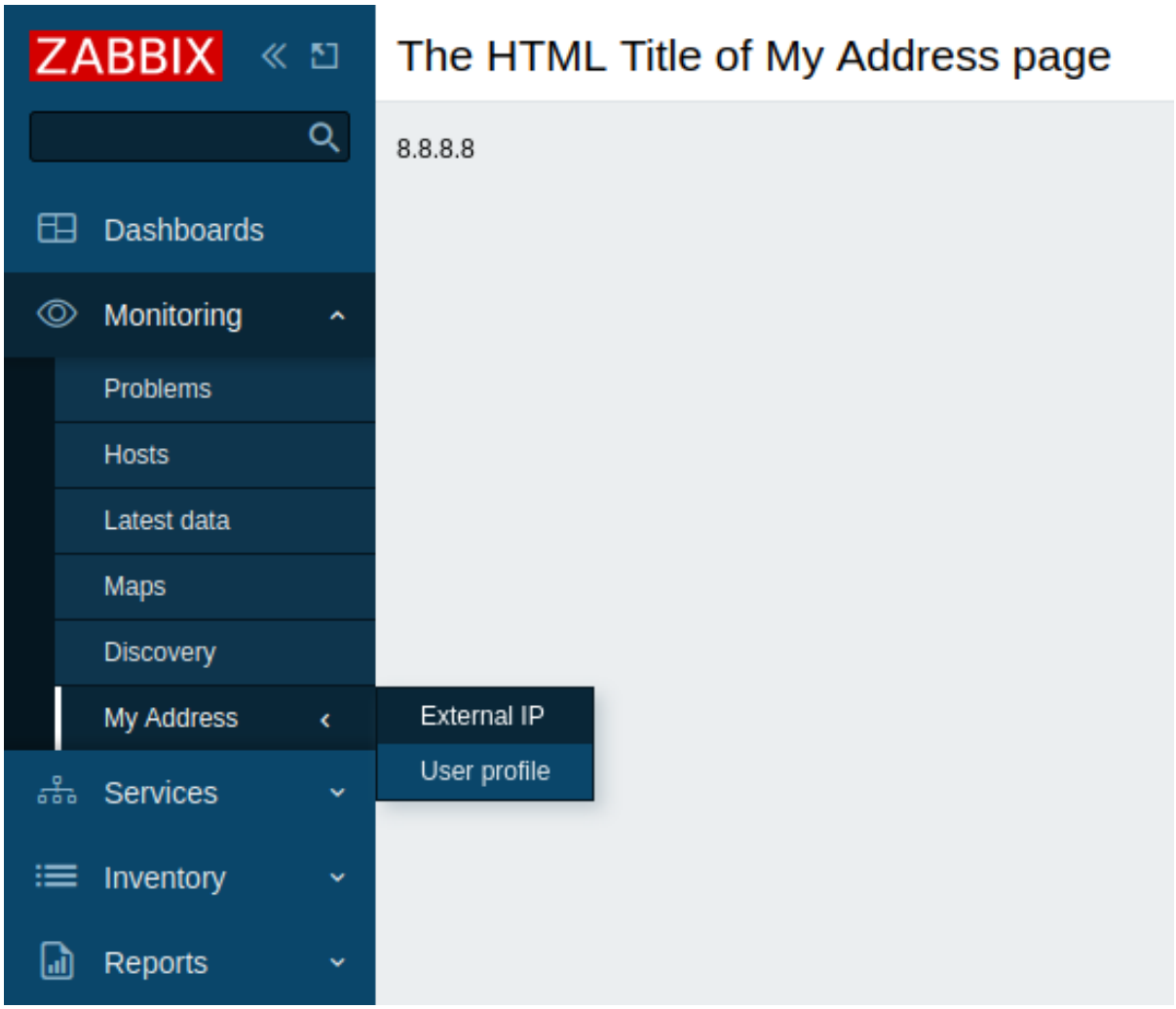
Ta sekcja zawiera praktyczne instrukcje krok po kroku, które pokazują, jak stworzyć własny moduł oraz widżet w Zabbix.

Utwórz moduł (samouczek)

To jest samouczek krok po kroku, który pokazuje, jak utworzyć prosty moduł front-endu Zabbix. Możesz pobrać wszystkie pliki tego modułu jako archiwum ZIP: MyAddress.zip.

Co zbudujesz

Podczas tego samouczka najpierw zbudujesz moduł frontend, który doda nową sekcję menu *My Address*, a następnie przekształcisz go w bardziej zaawansowany moduł frontend, który wykonuje żądanie HTTP do *https://api.seeip.org* i wyświetla odpowiedź — adres IP twojego komputera — na nowej stronie w nowo utworzonej sekcji menu *My Address*. Oto jak będzie wyglądał gotowy moduł:



Część I - Podstawowy moduł

Dodaj pusty moduł do frontendu Zabbix

1. Utwórz katalog *MyAddress* w katalogu *modules* instalacji frontendu Zabbix (na przykład *zabbix/ui/modules*).
2. Utwórz plik *manifest.json* z podstawowymi metadanymi modułu (zobacz opis obsługiwanych parametrów).

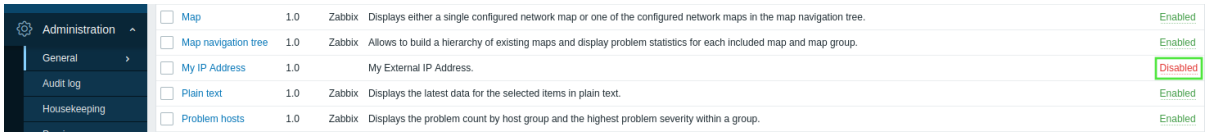
ui/modules/MyAddress/manifest.json

```
{
"manifest_version": 2.0,
"id": "my-address",
"name": "My IP Address",
"version": "1.0",
"namespace": "MyAddress",
"description": "My External IP Address."
}
```

3. W interfejsie użytkownika Zabbix przejdź do sekcji *Administracja* → *Ogólne* → *Moduły* i kliknij przycisk *Skanuj katalog*.



4. Znajdź nowy moduł *My IP Address* na liście i kliknij hiperłącze „Wyłączony”, aby zmienić status modułu z „Wyłączony” na „Włączony” (jeśli nie widzisz swojego modułu na liście, sprawdź uprawnienia).



Moduł jest teraz zarejestrowany w frontendzie. Jednak nie jest jeszcze widoczny, ponieważ nadal musisz zdefiniować funkcjonalność modułu. Po dodaniu zawartości do katalogu modułu, natychmiast zobaczysz zmiany w frontendzie Zabbix po odświeżeniu strony.

Utwórz sekcję menu

1. Utwórz plik *Module.php* w katalogu *MyAddress*.

Ten plik implementuje nową klasę *Module*, która rozszerza domyślną klasę *CModule*. Klasa *Module* wstawi nową sekcję menu *My Address* do menu głównego.

Metoda *setAction()* określa akcję, która ma zostać wykonana po kliknięciu sekcji menu. Na początek możesz użyć predefiniowanej akcji *userprofile.edit*, która otworzy stronę *Profil użytkownika*. W części III tego samouczka dowiesz się, jak utworzyć własną akcję.

ui/modules/MyAddress/Module.php

```
<?php

namespace Modules\MyAddress;

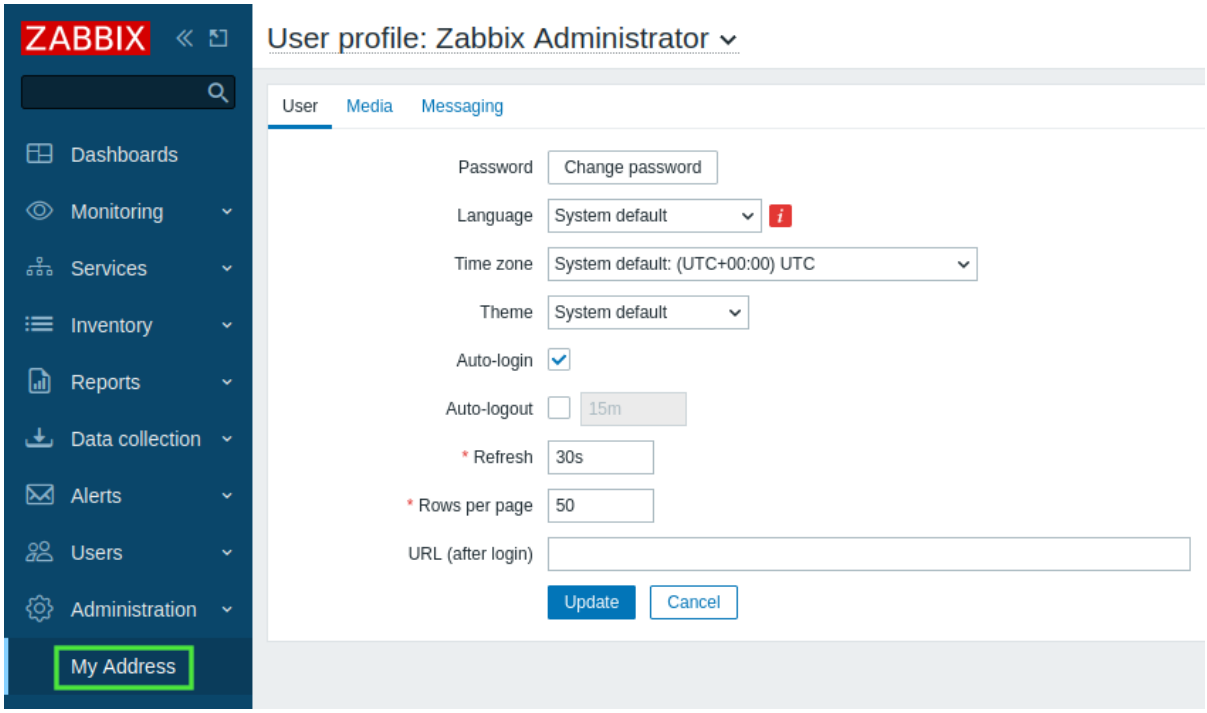
use Zabbix\Core\CModule,
    APP,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->add((new CMenuItem(_('My Address')))
            ->setAction('userprofile.edit'));
    }
}
```

Note:
Możesz zastąpić `'userprofile.edit'` innymi akcjami, na przykład `'charts.view'` (otwiera wykresy niestandardowe), `'problems.view'` (otwiera *Monitorowanie* → *Problemy*) lub `'report.status'` (otwiera raport *Informacje o systemie*).

3. Odśwież frontend Zabbix. Na dole menu głównego Zabbix znajduje się teraz nowa sekcja *My Address*. Kliknij *My Address*, aby otworzyć stronę *Profil użytkownika*.



Część II - Zmiana lokalizacji sekcji menu

W tej części przeniesiesz sekcję menu *My address* do sekcji *Monitorowanie*, a następnie dodasz do niej zagnieżdżone menu. W rezultacie użytkownicy będą mogli uzyskać dostęp do dwóch stron podmenu z sekcji menu *Monitorowanie* → *My address*.

1. Otwórz i edytuj plik *Module.php*.

ui/modules/MyAddress/Module.php

```
<?php

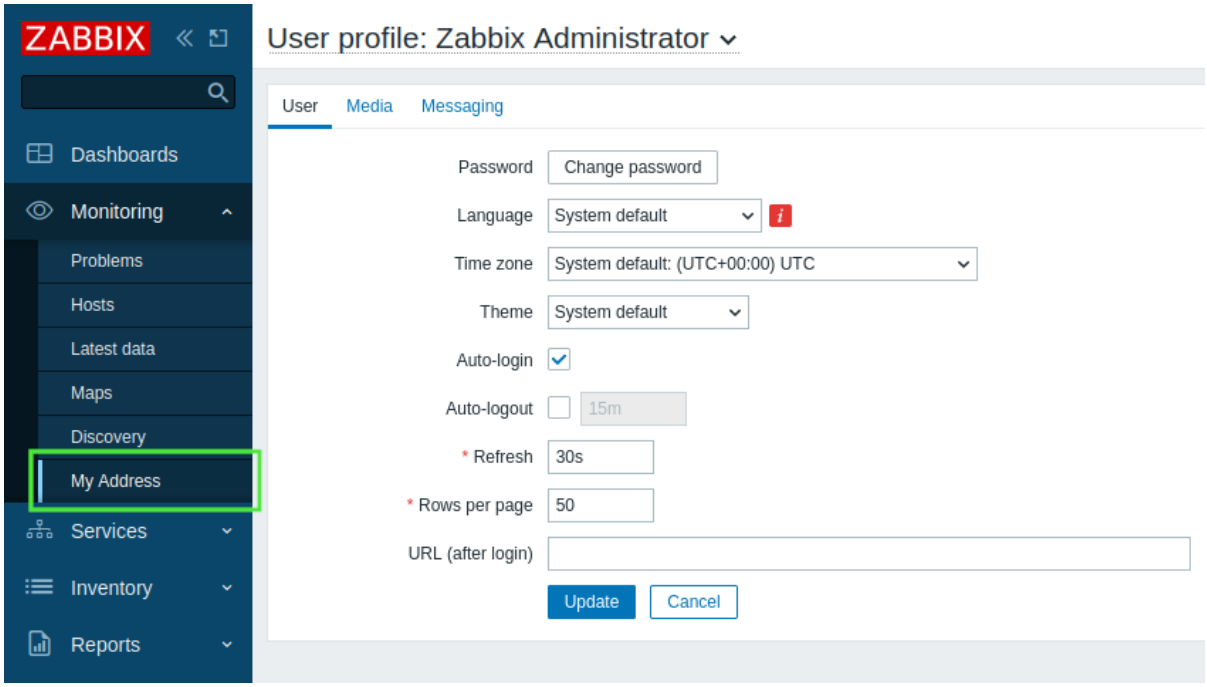
namespace Modules\MyAddress;

use Zabbix\Core\CModule,
APP,
CMenuItem;

class Module extends CModule {

public function init(): void {
APP::Component()->get('menu.main')
->findOrAdd(_('Monitoring'))
->getSubmenu()
->insertAfter(_('Discovery'),
(new CMenuItem(_('My Address')))->setAction('userprofile.edit')
);
}
}
```

2. Odśwież interfejs użytkownika Zabbix. Rozwiń sekcję menu *Monitoring* i zauważ, że sekcja *My address* znajduje się teraz poniżej sekcji *Odkrywanie*.



3. Aby dodać zagnieżdżone strony do sekcji menu *Mój adres*, otwórz i edytuj ponownie plik *Module.php*.

Ten krok tworzy dwie podsekcje:

- *Zewnętrzny adres IP*, który wykonuje nową akcję 'my.address', która zostanie zdefiniowana w kolejnych krokach;
- *Profil użytkownika*, który wykonuje wstępnie zdefiniowaną akcję 'userprofile.edit', aby otworzyć stronę *Profil użytkownika*.

Należy pamiętać, że w przypadku zagnieżdżonego menu należy użyć klasy *CMenu* oprócz klas użytych w poprzednich krokach.

ui/modules/MyAddress/Module.php

```
<?php

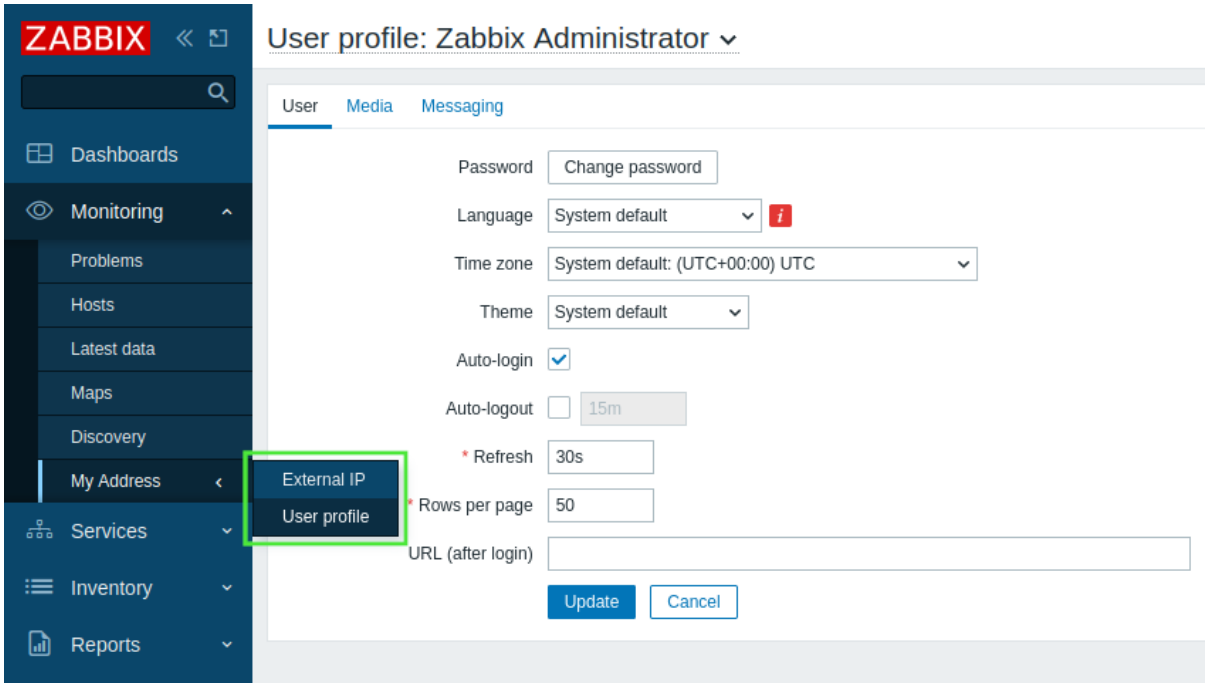
namespace Modules\MyAddress;

użyj Zabbix\Core\CModule,
APP,
CMenu,
CMenuItem;

class Module extends CModule {

public function init(): void {
APP::Component()->get('menu.main')
->findOrAdd(_('Monitoring'))
->getSubmenu()
->insertAfter(_('Discovery'),
(new CMenuItem(_('Mój adres')))->setSubMenu(
new CMenu([
(new CMenuItem(_('Zewnętrzny adres IP')))->setAction('my.address'),
(new CMenuItem(_('Profil użytkownika')))->setAction('userprofile.edit')
])
)
);
}
}
```

4. Odśwież frontend Zabbix. Zauważ, że sekcja menu *Mój adres* zawiera teraz menu trzeciego poziomu z dwiema stronami - *Zewnętrzny adres IP* i *Profil użytkownika*.



Część III - Akcja modułu

Akcja jest implementowana w dwóch plikach - *actions/MyAddress.php* i *views/my.address.php*. Plik ***actions/MyAddress.php*** zajmuje się implementacją logiki biznesowej, podczas gdy plik ***views/my.address.php*** odpowiada za widok.

1. Utwórz katalog *actions* w katalogu *MyAddress*.
2. Utwórz plik *MyAddress.php* w katalogu *actions*.

Logika akcji zostanie zdefiniowana w klasie *MyAddress*. Ta klasa akcji zaimplementuje cztery funkcje: *init()*, *checkInput()*, *checkPermissions()* i *doAction()*. Frontend Zabbix wywołuje funkcję *doAction()*, gdy żądana jest akcja. Ta funkcja odpowiada za logikę biznesową modułu.

> **Attention:**
> Dane muszą być zorganizowane jako tablica asocjacyjna. Tablica może być wielowymiarowa i może zawierać dowolne dane oczekiwane przez widok.

ui/modules/MyAddress/actions/MyAddress.php

```
<?php

namespace Modules\MyAddress\Actions;

use CController,
CControllerResponseData;

class MyAddress extends CController {

public function init(): void {
$this->disableCsrfValidation();
}

protected function checkInput(): bool {
return true;
}

protected function checkPermissions(): bool {
return true;
}

protected function doAction(): void {
$data = ['my-ip' => file_get_contents("https://api.seeip.org")];
$response = new CControllerResponseData($data);
```

```
$this->setResponse($response);
}
}
```

3. Utwórz nowy katalog *views* w katalogu *MyAddress*.
4. Utwórz plik *my.address.php* w katalogu *views* i zdefiniuj widok modułu.

Należy zauważyć, że zmienna `$data` jest dostępna w widoku bez jej konkretnego definiowania. Framework automatycznie przekazuje tablicę asocjacyjną do widoku.

ui/modules/MyAddress/views/my.address.php

```
<?php

(new CHtmlPage())
->setTitle(_('Tytuł HTML strony My Address'))
->addItem(new CDiv($data['my-ip']))
->show();
```

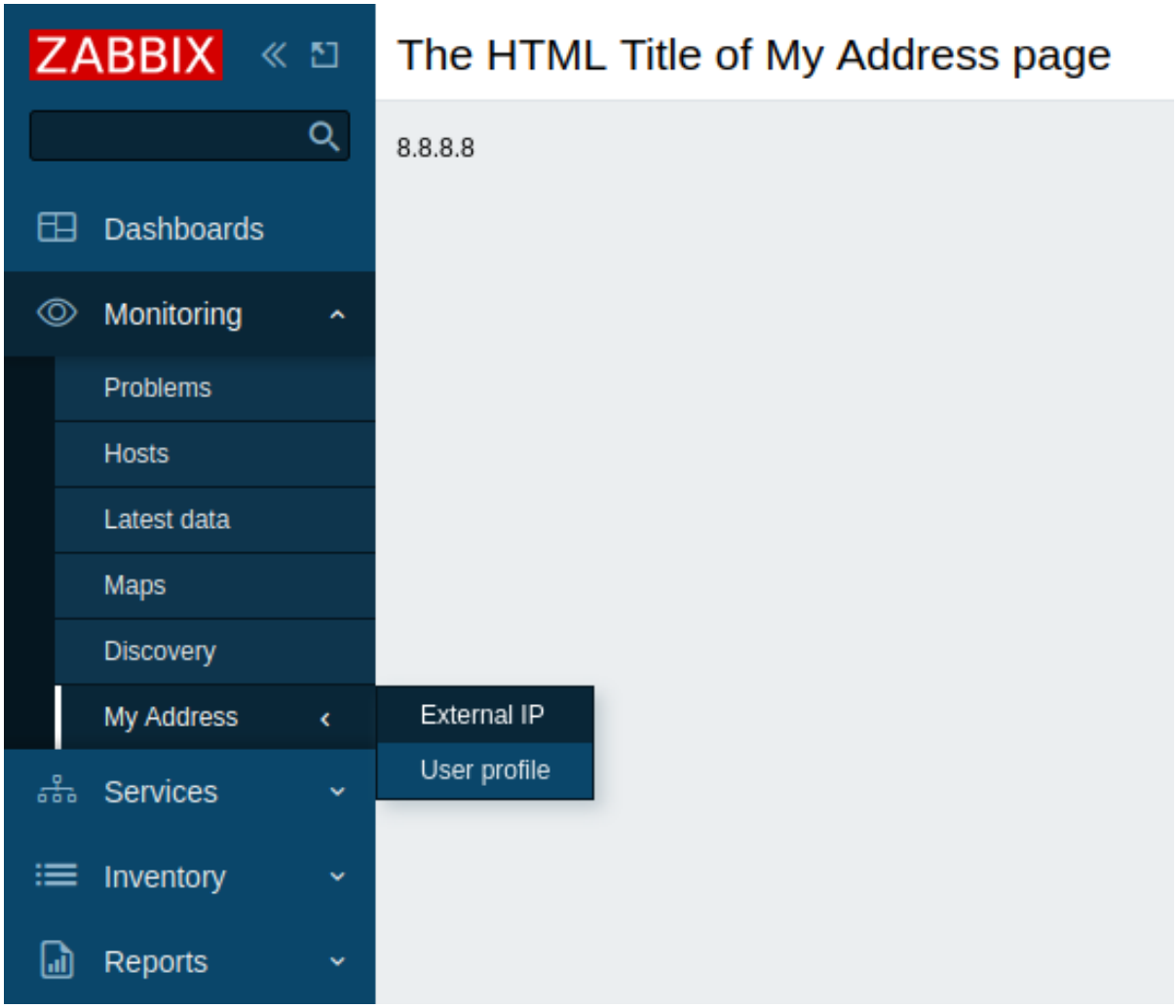
5. Akcja modułu musi zostać zarejestrowana w pliku *manifest.json*. Otwórz *manifest.json* i dodaj nowy obiekt `actions`, który zawiera:

- klucz akcji z nazwą akcji zapisaną małymi literami (a-z) i słowami oddzielonymi kropkami (na przykład `my.address`);
- nazwę klasy akcji (`MyAddress`) jako wartość klucza `class` obiektu `my.address`;
- nazwę widoku akcji (`my.address`) jako wartość klucza `view` obiektu `my.address`.

ui/modules/MyAddress/manifest.json

```
{
"manifest_version": 2.0,
"id": "my-address",
"name": "My IP Address",
"version": "1.0",
"namespace": "MyAddress",
"description": "My External IP Address.",
"actions": {
"my.address": {
"class": "MyAddress",
"view": "my.address"
}
}
}
```

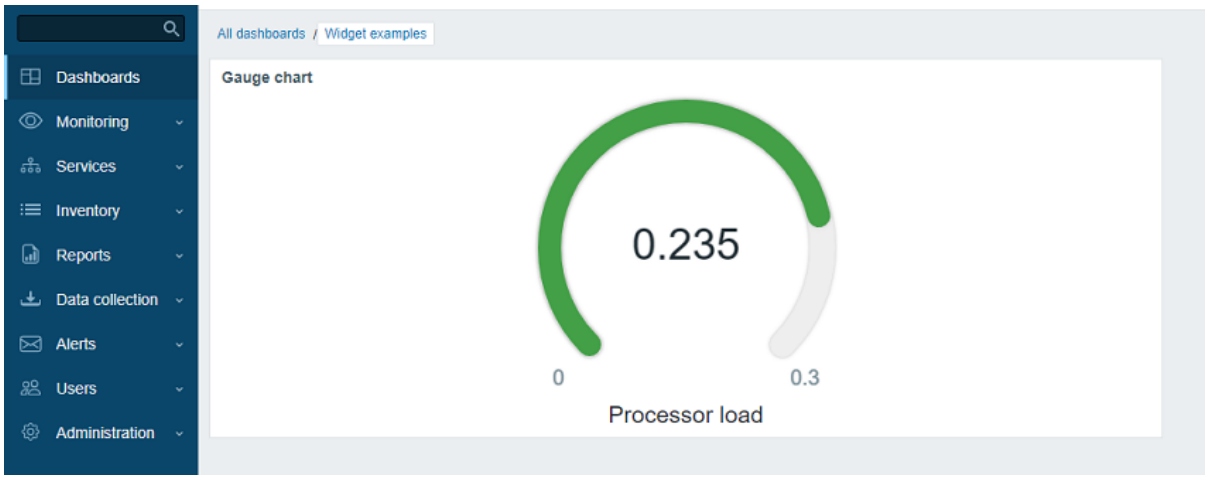
6. Odśwież interfejs użytkownika Zabbix. Kliknij *Mój adres → Zewnętrzny adres IP*, aby zobaczyć adres IP swojego komputera.



Utwórz widżet (samouczek)

To jest samouczek pokazujący krok po kroku, jak utworzyć prosty widget pulpitu nawigacyjnego. Możesz pobrać wszystkie pliki tego widgetu w postaci archiwum ZIP: lesson_gauge_chart.zip.

Co zbudujesz Podczas tego samouczka najpierw utworzysz podstawowy widget „Hello, world!”, a następnie przekształcisz go w bardziej zaawansowany widget, który wyświetla wartość pozycji jako wykres wskaźnikowy. Tak będzie wyglądać gotowy widget:



Część I - ”Witaj, świecie!” W tej sekcji dowiesz się, jak utworzyć minimalną wymaganą liczbę elementów widgetu i dodać nowy widget do frontendu Zabbix.

Dodawanie pustego widżetu do frontend Zabbix

1. Utwórz katalog *lesson_gauge_chart* w katalogu *modules* instalacji frontend Zabbix (na przykład *zabbix/ui/modules*).

Note:
Wszystkie niestandardowe widżety są traktowane jako moduły zewnętrzne i muszą zostać dodane do katalogu *modules* instalacji frontend Zabbix (na przykład *zabbix/ui/modules*). Katalog *zabbix/ui/widgets* jest zarezerwowany dla wbudowanych widżetów Zabbix i jest aktualizowany razem z frontendem Zabbix.

2. Utwórz plik *manifest.json* z podstawowymi metadanymi widżetu (zobacz opis obsługiwanych parametrów).

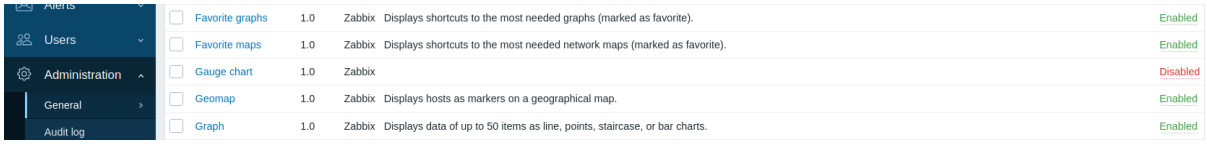
ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.1",
    "author": "Zabbix"
}
```

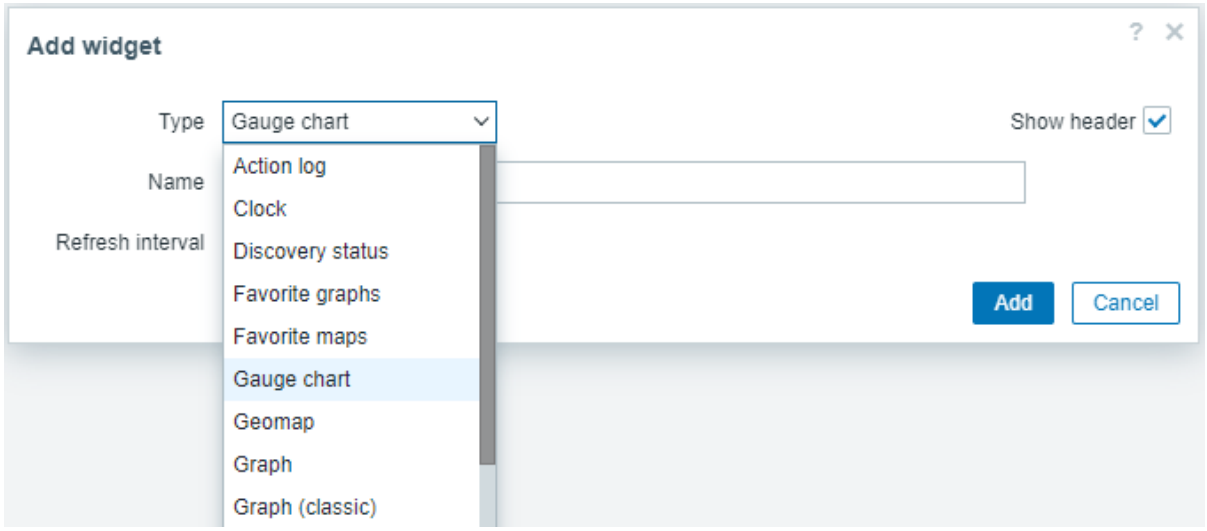
3. W frontend Zabbix przejdź do sekcji *Administration* → *General* → *Modules* i kliknij przycisk *Scan directory*.



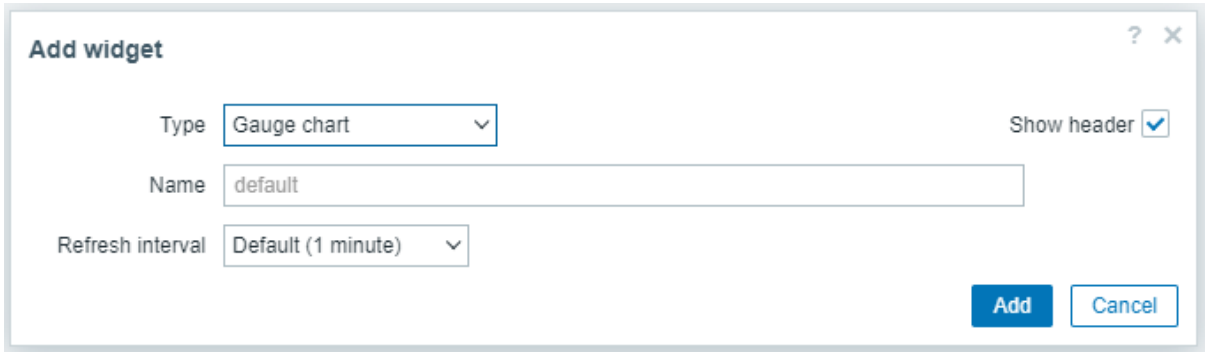
4. Znajdź nowy moduł *Gauge chart* na liście i kliknij hiperłącze "Disabled", aby zmienić status modułu z "Disabled" na "Enabled" (jeśli moduł nie jest widoczny na liście, zobacz sekcję troubleshooting).



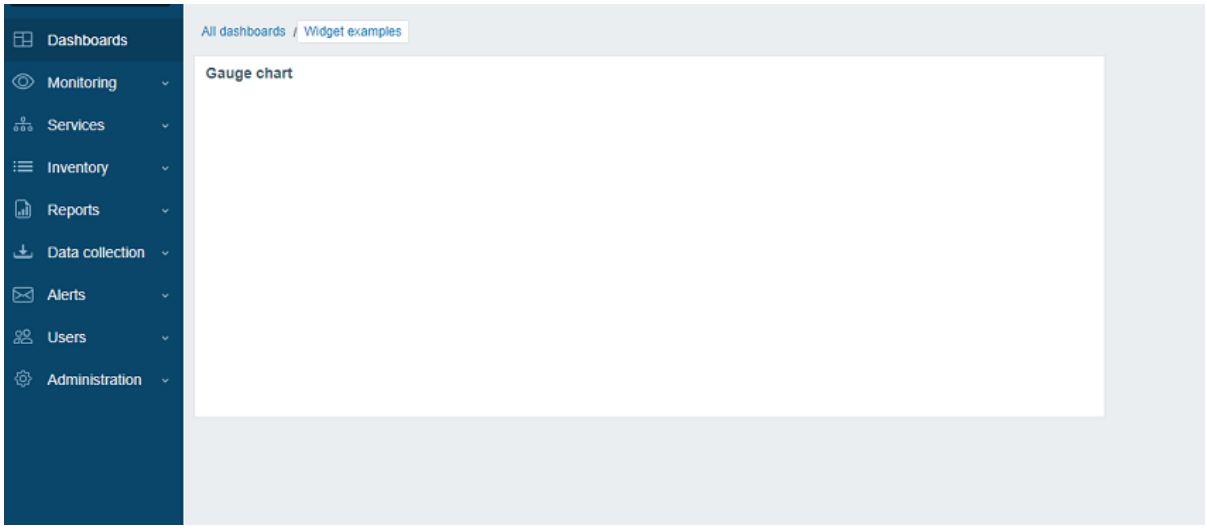
5. Otwórz dashboard, przełącz go w tryb edycji i dodaj nowy widżet. W polu *Type* wybierz "Gauge chart".



6. W tym momencie konfiguracja widżetu *Gauge chart* zawiera tylko ogólne pola widżetu *Name* i *Refresh interval*. Kliknij *Add*, aby dodać widżet do dashboardu.



7. Na dashboardzie powinien pojawić się pusty widżet. Kliknij *Save changes* w prawym górnym rogu, aby zapisać dashboard.



Dodawanie widoku widgetu

> **Note:**
> Plik **widoku** widgetu powinien znajdować się w katalogu *views* (w tym samouczku: *ui/modules/lesson_gauge_chart/views/*). Jeśli plik ma domyślną nazwę *widget.view.php*, nie trzeba go rejestrować w pliku *manifest.json*. Jeśli plik ma inną nazwę, należy ją określić w sekcji *actions/widget.lesson_gauge_chart.view* pliku manifest.json.

1. Utwórz katalog *views* w katalogu *lesson_gauge_chart*.
2. Utwórz plik *widget.view.php* w katalogu *views*.

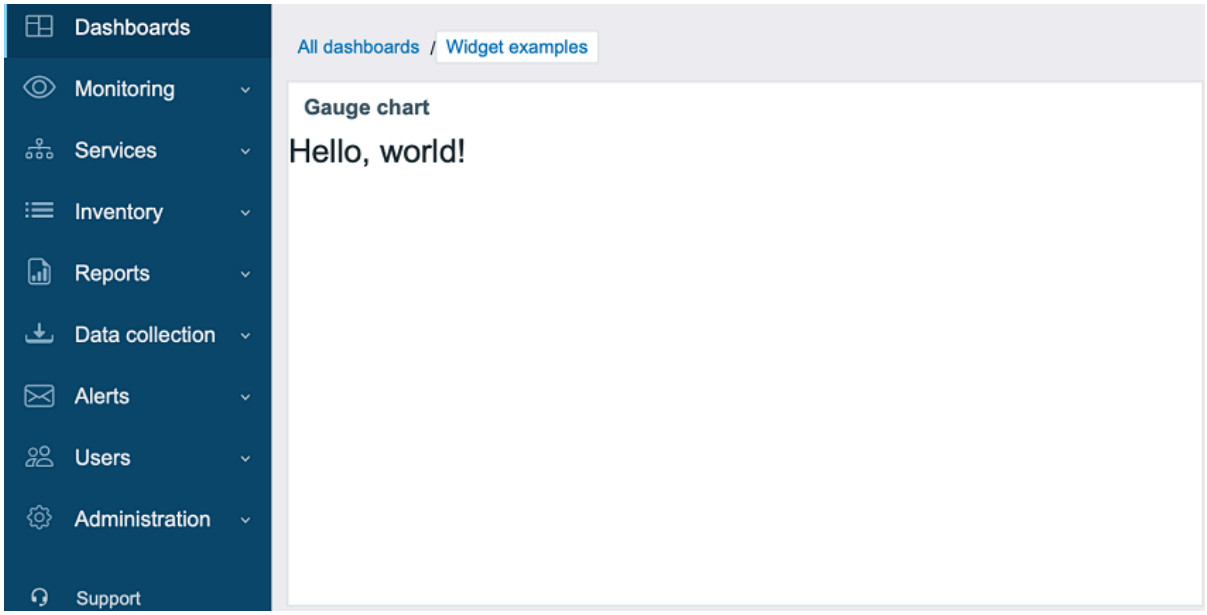
ui/modules/lesson_gauge_chart/views/widget.view.php

```
<?php

/**
 * Widok widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, 'Hello, world!')
    )
    ->show();
```

3. Odśwież pulpit. Widget *Gauge chart* wyświetla teraz „Hello, world!”.



Część II - Wykres wskaźnikowy Dodawanie ustawień do widoku konfiguracji i używanie ich w widoku widgetu

W tej sekcji dowiesz się, jak dodać pole konfiguracji widgetu i wyświetlić wprowadzaną wartość w widoku widgetu jako tekst.

Konfiguracja widgetu składa się z formularza (*Zabbix\Widgets\CWidgetForm*) oraz widoku formularza widgetu (*widget.edit.php*). Aby dodać pola (*Zabbix\Widgets\CWidgetField*), należy utworzyć klasę *WidgetForm*, która będzie rozszerzać *Zabbix\Widgets\CWidgetForm*.

Formularz zawiera zestaw pól (*Zabbix\Widgets\CWidgetField*) różnych typów, które są używane do walidacji wartości wprowadzanych przez użytkownika. Pole formularza (*Zabbix\Widgets\CWidgetField*) dla każdego typu elementu wejściowego konwertuje wartość do jednolitego formatu, aby można ją było przechowywać w bazie danych.

> **Note:**
> Plik **form** widgetu powinien znajdować się w katalogu *includes* (w tym samouczku: *ui/modules/lesson_gauge_chart/includes/*). Jeśli plik ma domyślną nazwę *WidgetForm.php*, nie trzeba go rejestrować w pliku *manifest.json*. Jeśli plik ma inną nazwę, należy ją określić w sekcji *widget/form_class* pliku manifest.json.

1. Utwórz nowy katalog *includes* w katalogu *lesson_gauge_chart*.
2. Utwórz plik *WidgetForm.php* w katalogu *includes*.

ui/modules/lesson_gauge_chart/includes/WidgetForm.php

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

class WidgetForm extends CWidgetForm {
}
```

3. Dodaj pole *Description* do formularza konfiguracji widgetu. Jest to zwykłe pole tekstowe, w którym użytkownik może wprowadzić dowolny zestaw znaków. Możesz do tego użyć klasy *CWidgetFieldTextBox*.

ui/modules/lesson_gauge_chart/includes/WidgetForm.php

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

use Zabbix\Widgets\Fields\CWidgetFieldTextBox;

class WidgetForm extends CWidgetForm {

    public function addFields(): self {
```

```
        return $this
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

4. W katalogu *views* utwórz plik widoku konfiguracji widgetu *widget.edit.php* i dodaj widok dla nowego pola *Description*. Dla klasy pola *CWidgetFieldTextBox* widokiem jest *CWidgetFieldTextBoxView.*

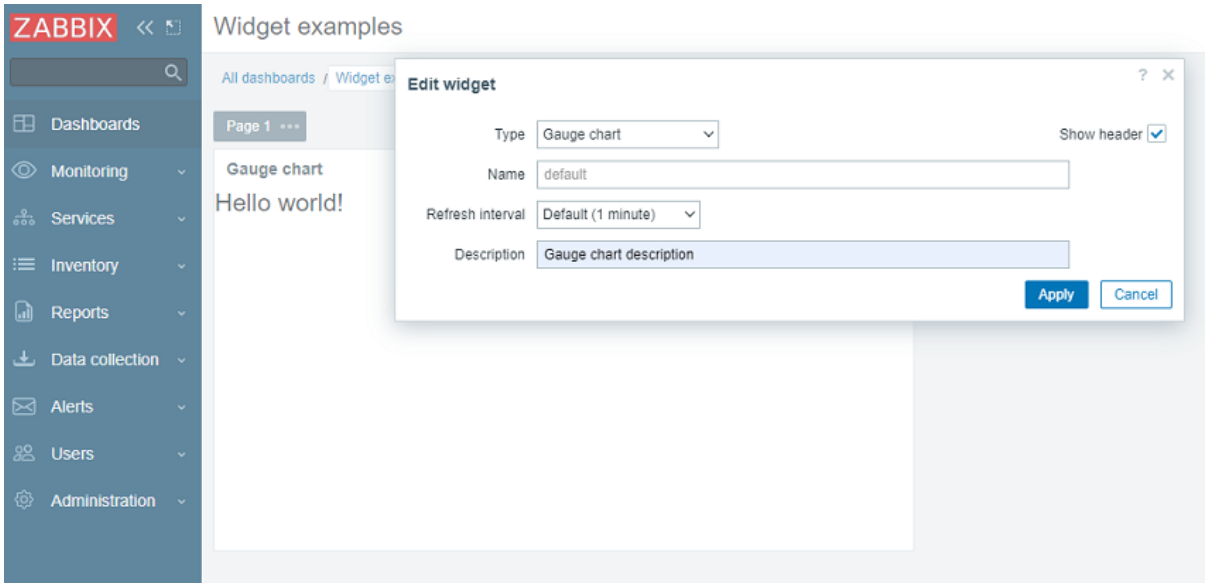
ui/modules/lesson_gauge_chart/views/widget.edit.php

```
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

5. Przejdź do pulpitu i kliknij ikonę koła zębatego w widżecie, aby otworzyć formularz konfiguracji widgetu.
6. Formularz konfiguracji widgetu zawiera teraz nowe pole tekstowe *Description*. Wprowadź dowolną wartość, na przykład *Gauge chart description*.



7. Kliknij *Apply* w formularzu konfiguracji widgetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać pulpit. Zwróć uwagę, że nowy opis nie jest nigdzie widoczny, a widget nadal wyświetla „Hello, world!”.

Aby nowy opis pojawił się w widżecie, wartość pola *Description* musi zostać pobrana z bazy danych i przekazana do widoku widgetu. W tym celu należy utworzyć klasę akcji.

8. Utwórz nowy katalog *actions* w katalogu *lesson_gauge_chart*.
9. Utwórz plik *WidgetView.php* w katalogu *actions*. Klasa akcji *WidgetView* będzie rozszerzać klasę *CControllerDashboardWidgetView*.

Wartości pól konfiguracji widgetu są przechowywane we właściwości **$fields_values** klasy akcji.

ui/modules/lesson_gauge_chart/actions/WidgetView.php

```
<?php
```

```
namespace Modules\LessonGaugeChart\Actions;

use CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'description' => $this->fields_values['description'],
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

10. Otwórz plik *manifest.json* i zarejestruj *WidgetView* jako klasę akcji w sekcji *actions/widget.lesson_gauge_chart.view*.

ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    }
}
```

11. Teraz możesz użyć wartości pola opisu, zawartej w *$data['description']*, w widoku widgetu. Otwórz plik *views/widget.view.php* i zastąp statyczny tekst „Hello, world!" wartością *$data['description']*.

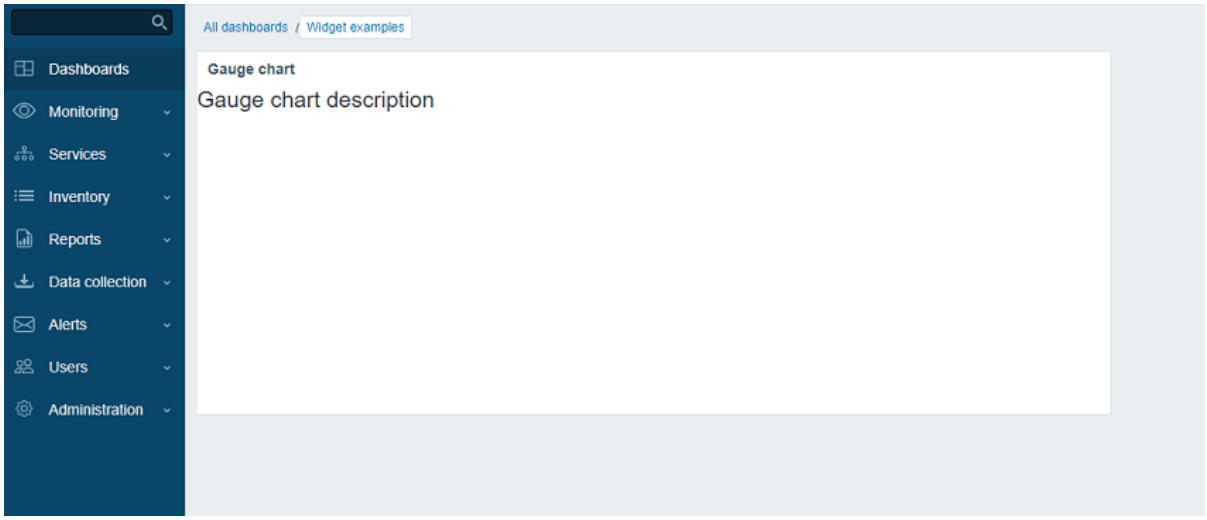
ui/modules/lesson_gauge_chart/views/widget.view.php

```
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['description'])
    )
    ->show();
```

12. Odśwież stronę pulpitu. Powinien teraz być widoczny tekst opisu widgetu zamiast „Hello, world!".



Pobieranie wartości pozycji przez API

Widget powinien wyświetlać ostatnią wartość pozycji wybranej przez użytkownika. W tym celu należy dodać możliwość wyboru pozycji w konfiguracji widgetu.

W tej sekcji dowiesz się, jak dodać pole wyboru pozycji do formularza widgetu oraz jak dodać wizualną część tego pola do widoku konfiguracji. Następnie kontroler widgetu będzie mógł pobrać dane pozycji i jej wartość za pomocą żądania API. Po odebraniu wartość będzie mogła zostać wyświetlona w widoku widgetu.

1. Otwórz plik *includes/WidgetForm.php* i dodaj pole *CWidgetFieldMultiSelectItem*. Pozwoli to na wybór pozycji w formularzu konfiguracji.

ui/modules/lesson_gauge_chart/includes/WidgetForm.php

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldMultiSelectItem,
    CWidgetFieldTextBox
};

/**
 * Formularz widgetu wykresu wskaźnikowego.
 */
class WidgetForm extends CWidgetForm {

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

2. Otwórz plik *views/widget.edit.php* i dodaj wizualny komponent pola do widoku konfiguracji.

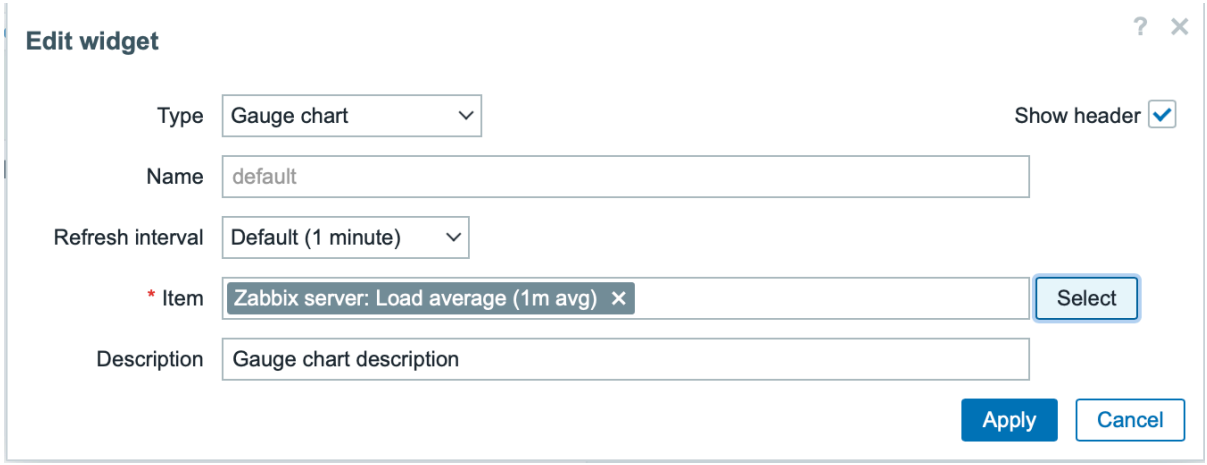
ui/modules/lesson_gauge_chart/views/widget.edit.php

```php
<?php

/**
 * Widok formularza widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldMultiSelectItemView($data['fields']['itemid'])
    )
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

3. Wróć do pulpitu i kliknij ikonę koła zębatego w widżecie, aby otworzyć formularz konfiguracji widgetu.
4. Formularz konfiguracji widgetu zawiera teraz nowe pole wejściowe *Item*. Wybierz host "Zabbix server" oraz pozycję "Load average (1m avg)".



5. Kliknij *Apply* w formularzu konfiguracji widgetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać pulpit.
6. Otwórz i zmodyfikuj plik *actions/WidgetView.php*.

Od tej chwili ID pozycji będzie dostępne w kontrolerze widgetu w *$this->fields_values['itemid']*. Metoda kontrolera *doAction()* pobiera dane pozycji (nazwę, typ wartości, jednostki) za pomocą metody API *item.get* oraz ostatnią wartość pozycji za pomocą metody API *history.get*.

ui/modules/lesson_gauge_chart/actions/WidgetView.php

```php
<?php

namespace Modules\LessonGaugeChart\Actions;

use API,
    CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $db_items = API::Item()->get([
            'output' => ['itemid', 'value_type', 'name', 'units'],
            'itemids' => $this->fields_values['itemid'],
            'webitems' => true,
            'filter' => [
                'value_type' => [ITEM_VALUE_TYPE_UINT64, ITEM_VALUE_TYPE_FLOAT]
            ]
```

```
        ]);

        $value = null;

        if ($db_items) {
            $item = $db_items[0];

            $history = API::History()->get([
                'output' => API_OUTPUT_EXTEND,
                'itemids' => $item['itemid'],
                'history' => $item['value_type'],
                'sortfield' => 'clock',
                'sortorder' => ZBX_SORT_DOWN,
                'limit' => 1
            ]);

            if ($history) {
                $value = convertUnitsRaw([
                    'value' => $history[0]['value'],
                    'units' => $item['units']
                ]);
            }
        }

        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'value' => $value,
            'description' => $this->fields_values['description'],
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

7. Otwórz plik *views/widget.view.php* i dodaj wartość pozycji do widoku widgetu.

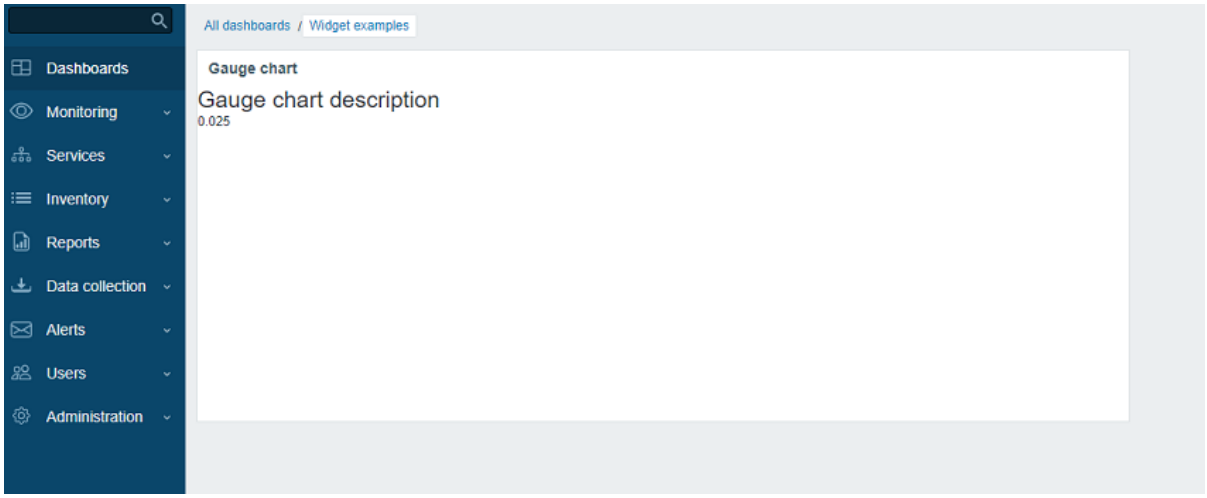
ui/modules/lesson_gauge_chart/views/widget.view.php

```
<?php

/**
 * Widok widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem([
        new CTag('h1', true, $data['description']),
        new CDiv($data['value'] !== null ? $data['value']['value'] : _('No data'))
    ])
    ->show();
```

8. Odśwież stronę pulpitu. Widget wyświetli najnowszą wartość pozycji.



Dodaj zaawansowane ustawienia konfiguracji do widoku konfiguracji

W tej sekcji dowiesz się, jak dodać rozwijaną/zwijaną sekcję *Advanced configuration* z opcjonalnymi parametrami, takimi jak kolor, wartości minimalna i maksymalna, jednostki oraz pole *Description* utworzone wcześniej.

1. Utwórz plik *Widget.php* w głównym katalogu widżetu *lesson_gauge_chart*, aby utworzyć nową klasę *Widget*.

Klasa *Widget* rozszerzy klasę bazową *CWidget*, aby dodać/nadpisać domyślne ustawienia widżetu (w tym przypadku - tłumaczenia). Poniższy kod JavaScript wyświetla ciąg ”No data” w przypadku braku danych. Ciąg ”No data” jest obecny w plikach tłumaczeń interfejsu Zabbix.

Jeśli istnieją jakiekolwiek stałe widżetu, zaleca się również określić je w klasie *Widget*.

ui/modules/lesson_gauge_chart/Widget.php

```
<?php

namespace Modules\LessonGaugeChart;

use Zabbix\Core\CWidget;

class Widget extends CWidget {

    public const UNIT_AUTO = 0;
    public const UNIT_STATIC = 1;

    public function getTranslationStrings(): array {
        return [
            'class.widget.js' => [
                'No data' => _('No data')
            ]
        ];
    }
}
```

2. Otwórz *includes/WidgetForm.php* i dodaj nowe pola *Color* (selektor koloru), *Min* (pole numeryczne), *Max* (pole numeryczne) oraz *Units* (lista wyboru), a także zdefiniuj domyślną paletę kolorów dla selektora koloru, aby można było użyć jej w kolejnych krokach.

ui/modules/lesson_gauge_chart/includes/WidgetForm.php

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Modules\LessonGaugeChart\Widget;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};
```

```
use Zabbix\Widgets\Fields\{
    CWidgetFieldColor,
    CWidgetFieldMultiSelectItem,
    CWidgetFieldNumericBox,
    CWidgetFieldSelect,
    CWidgetFieldTextBox
};

/**
 * Formularz widżetu wykresu wskaźnikowego.
 */
class WidgetForm extends CWidgetForm {

    public const DEFAULT_COLOR_PALETTE = [
        'FF465C', 'B0AF07', '0EC9AC', '524BBC', 'ED1248', 'D1E754', '2AB5FF', '385CC7', 'EC1594', 'BAE37D',
        '6AC8FF', 'EE2B29', '3CA20D', '6F4BBC', '00A1FF', 'F3601B', '1CAE59', '45CFDB', '894BBC', '6D6D6D'
    ];

    public function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            )
            ->addField(
                (new CWidgetFieldNumericBox('value_min', _('Min')))
                    ->setDefault(0)
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
            )
            ->addField(
                (new CWidgetFieldNumericBox('value_max', _('Max')))
                    ->setDefault(100)
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
            )
            ->addField(
                (new CWidgetFieldSelect('value_units', _('Units'), [
                    Widget::UNIT_AUTO => _x('Auto', 'history source selection method'),
                    Widget::UNIT_STATIC => _x('Static', 'history source selection method')
                ]))->setDefault(Widget::UNIT_AUTO)
            )
            ->addField(
                (new CWidgetFieldTextBox('value_static_units'))
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            );
    }
}
```

3. Otwórz *views/widget.edit.php* i dodaj komponenty wizualne pól do widoku konfiguracji.

ui/modules/lesson_gauge_chart/views/widget.edit.php

```
<?php

/**
 * Widok formularza widżetu wykresu wskaźnikowego.
 *
 * @var CView $this
```

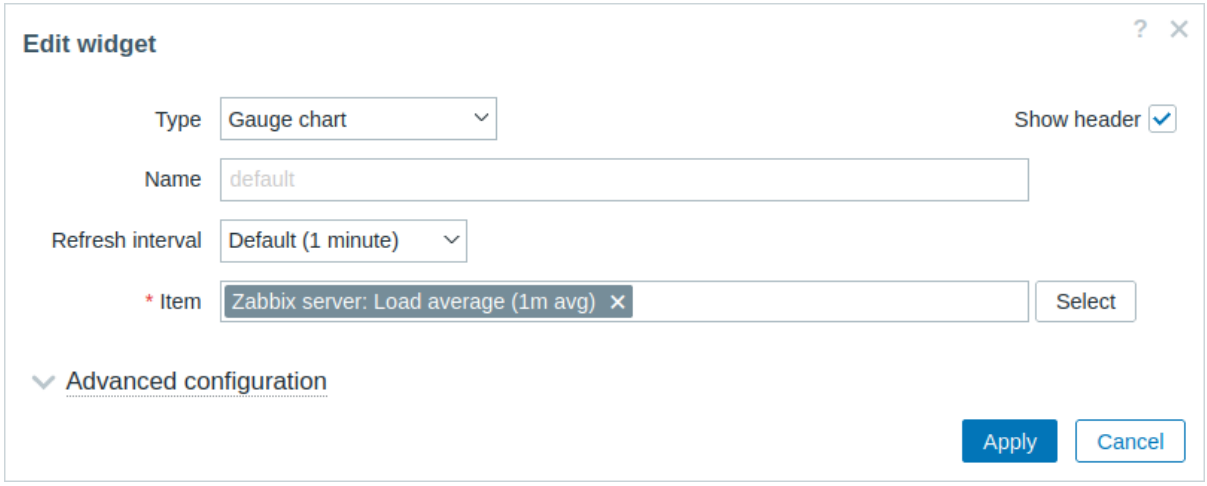
```
 * @var array $data
 */

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

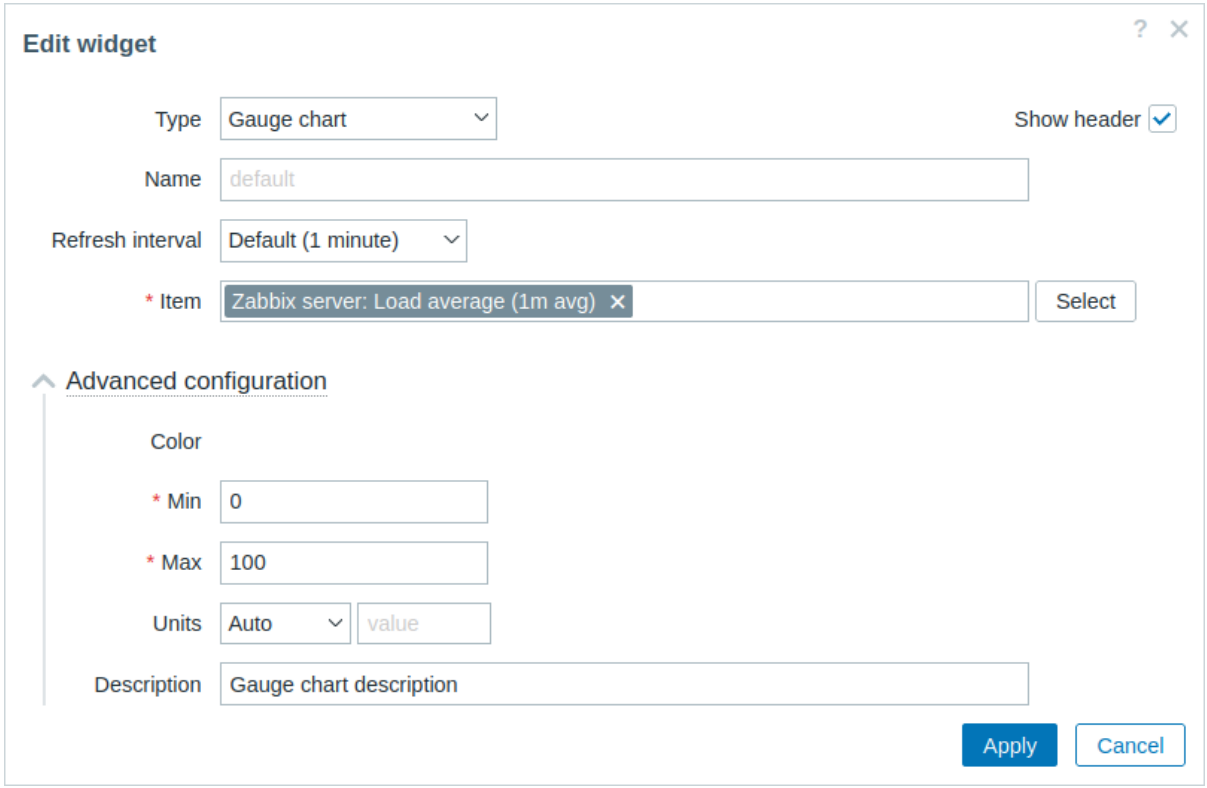
(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_min'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_max'])
            )
            ->addItem([
                $lefty_units->getLabel(),
                (new CFormField([
                    $lefty_units->getView()->addClass(ZBX_STYLE_FORM_INPUT_MARGIN),
                    $lefty_static_units->getView()
                ]))
            ])
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
    )
    ->show();
```

Note:
Metoda *addField()* klasy *CWidgetFormView* przyjmuje ciąg klasy CSS jako drugi parametr.

4. Wróć do dashboardu, przełącz się w tryb edycji i kliknij ikonę koła zębatego w widżecie, aby otworzyć formularz konfiguracji widżetu. Formularz konfiguracji widżetu zawiera teraz nową rozwijaną/zwijaną sekcję *Advanced configuration*.



5. Rozwiń sekcję *Advanced configuration*, aby zobaczyć dodatkowe pola konfiguracji widżetu. Zwróć uwagę, że pole *Color* nie ma jeszcze selektora koloru. Dzieje się tak, ponieważ selektor koloru musi zostać zainicjalizowany za pomocą JavaScript, który zostanie dodany w następnej sekcji - *Dodaj JavaScript do widżetu*.



Dodawanie JavaScript do widgetu

W tej sekcji dowiesz się, jak dodać wykres wskaźnikowy (gauge chart) — utworzony przy użyciu JavaScript — który pokazuje, czy najnowsza wartość jest prawidłowa, czy zbyt wysoka/zbyt niska.

1. Utwórz plik *widget.edit.js.php* w katalogu *views*.

JavaScript będzie odpowiadał za inicjalizację selektora kolorów w widoku konfiguracji.

ui/modules/lesson_gauge_chart/views/widget.edit.js.php

```
<?php

use Modules\LessonGaugeChart\Widget;

?>

window.widget_lesson_gauge_chart_form = new class {

    init({color_palette}) {
        this._unit_select = document.getElementById('value_units');
        this._unit_value = document.getElementById('value_static_units');

        this._unit_select.addEventListener('change', () => this.updateForm());

        colorPalette.setThemeColors(color_palette);

        for (const colorpicker of jQuery('.<?= ZBX_STYLE_COLOR_PICKER ?> input')) {
            jQuery(colorpicker).colorpicker();
        }

        const overlay = overlays_stack.getById('widget_properties');

        for (const event of ['overlay.reload', 'overlay.close']) {
            overlay.$dialogue[0].addEventListener(event, () => { jQuery.colorpicker('hide'); });
        }

        this.updateForm();
    }
```

```
    updateForm() {
        this._unit_value.disabled = this._unit_select.value == <?= Widget::UNIT_AUTO ?>;
    }
};
```

2. Otwórz *views/widget.edit.php* i dodaj plik *widget.edit.js.php* z kodem JavaScript do widoku konfiguracji. Aby to zrobić, użyj metody *includeJsFile()*. Aby dodać osadzony kod JavaScript, użyj metody *addJavaScript()*.

ui/modules/lesson_gauge_chart/views/widget.edit.php

```
<?php

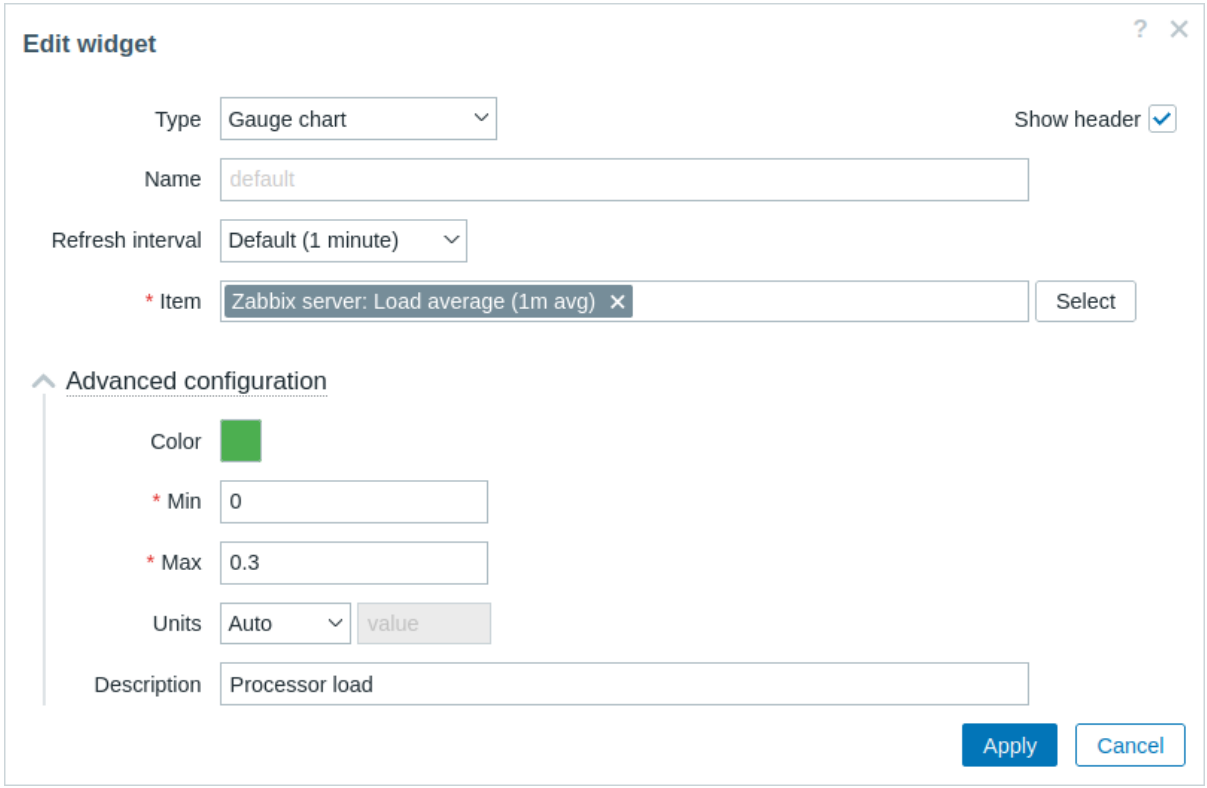
/**
 * Widok formularza widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

use Modules\LessonGaugeChart\Includes\WidgetForm;

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_min'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_max'])
            )
            ->addItem([
                $lefty_units->getLabel(),
                (new CFormField([
                    $lefty_units->getView()->addClass(ZBX_STYLE_FORM_INPUT_MARGIN),
                    $lefty_static_units->getView()
                ]))
            ])
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('widget_lesson_gauge_chart_form.init('.json_encode([
        'color_palette' => WidgetForm::DEFAULT_COLOR_PALETTE
    ], JSON_THROW_ON_ERROR).');')
    ->show();
```

3. Wróć do pulpitu, kliknij ikonę koła zębatego w widżecie, aby otworzyć formularz konfiguracji widgetu. Następnie rozwiń sekcję *Advanced configuration*, aby zobaczyć zainicjalizowany selektor kolorów. Wypełnij pola wartościami i wybierz kolor dla wykresu wskaźnikowego.



4. Kliknij *Apply* w formularzu konfiguracji widgetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać pulpit.
5. Otwórz *actions/WidgetView.php* i zaktualizuj kontroler.

Właściwość **$this->fields_values** zawiera teraz wartości wszystkich pól z sekcji *Advanced configuration*. Uzupełnij kontroler, aby umożliwić przekazywanie konfiguracji i wartości wybranej pozycji do widoku widgetu.

ui/modules/lesson_gauge_chart/actions/WidgetView.php

```
<?php

namespace Modules\LessonGaugeChart\Actions;

use API,
    CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $db_items = API::Item()->get([
            'output' => ['itemid', 'value_type', 'name', 'units'],
            'itemids' => $this->fields_values['itemid'],
            'webitems' => true,
            'filter' => [
                'value_type' => [ITEM_VALUE_TYPE_UINT64, ITEM_VALUE_TYPE_FLOAT]
            ]
        ]);

        $history_value = null;

        if ($db_items) {
            $item = $db_items[0];

            $history = API::History()->get([
                'output' => API_OUTPUT_EXTEND,
                'itemids' => $item['itemid'],
                'history' => $item['value_type'],
                'sortfield' => 'clock',
                'sortorder' => ZBX_SORT_DOWN,
```

```
                'limit' => 1
            ]);

            if ($history) {
                $history_value = convertUnitsRaw([
                    'value' => $history[0]['value'],
                    'units' => $item['units']
                ]);
            }
        }

        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'history' => $history_value,
            'fields_values' => $this->fields_values,
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

6. Otwórz i zmodyfikuj *views/widget.view.php*.

Musisz utworzyć kontener dla wykresu wskaźnikowego, który narysujesz w kolejnych krokach, oraz kontener dla opisu.

Aby przekazać wartości do JavaScript jako obiekt JSON, użyj metody *setVar()*.

ui/modules/lesson_gauge_chart/views/widget.view.php

```
<?php

/**
 * Widok widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem([
        (new CDiv())->addClass('chart'),
        $data['fields_values']['description']
            ? (new CDiv($data['fields_values']['description']))->addClass('description')
            : null
    ])
    ->setVar('history', $data['history'])
    ->setVar('fields_values', $data['fields_values'])
    ->show();
```

7. Utwórz nowy katalog *assets* w katalogu *lesson_gauge_chart*. Ten katalog będzie używany do przechowywania JavaScript, CSS oraz potencjalnie innych zasobów, takich jak czcionki lub obrazy.
8. Dla JavaScript widoku widgetu utwórz katalog *js* w katalogu *assets*.
9. Utwórz plik *class.widget.js* w katalogu *assets/js*.

Ta klasa JavaScript widgetu będzie rozszerzać bazową klasę JavaScript wszystkich widgetów pulpitu — *CWidget*.

Pulpit opiera się na poprawnej implementacji widgetu i przekazuje do niego wszystkie istotne informacje poprzez wywoływanie odpowiednich metod JavaScript. Pulpit oczekuje również, że widget będzie generował zdarzenia, gdy wystąpi jakaś interakcja. Dlatego klasa *CWidget* zawiera zestaw metod z domyślną implementacją zachowania widgetu, które można dostosować przez rozszerzenie tej klasy.

W tym przypadku konieczne jest pewne dostosowanie, dlatego zostanie zaimplementowana własna logika dla następujących aspektów działania widgetu:

- inicjalizacja widgetu, która odpowiada za zdefiniowanie początkowego stanu widgetu (zobacz metodę *onInitialize()*);

- wyświetlanie zawartości widgetu (to znaczy rysowanie wykresu wskaźnikowego), jeśli proces aktualizacji widgetu zakończył się powodzeniem i bez błędów (zobacz metodę *processUpdateResponse(response)* oraz powiązane metody *_resizeChart()* i *_updatedChart()*)
- zmiana rozmiaru widgetu (zobacz metodę *onResize()* oraz powiązaną metodę *_resizeChart()*)

W przypadku innych aspektów widgetu wykresu wskaźnikowego zostanie użyta domyślna implementacja zachowania widgetu. Aby dowiedzieć się więcej o metodach JavaScript klasy *CWidget*, zobacz: JavaScript.

Ponieważ ten kod JavaScript jest wymagany dla widoku widgetu, powinien być ładowany wraz ze stroną pulpitu. Aby włączyć ładowanie JavaScript, musisz zaktualizować parametry *assets/js* oraz *js_class* w pliku ***manifest.json***, jak pokazano w kroku 10.

ui/modules/lesson_gauge_chart/assets/js/class.widget.js

```
class WidgetLessonGaugeChart extends CWidget {

    static UNIT_AUTO = 0;
    static UNIT_STATIC = 1;

    onInitialize() {
        super.onInitialize();

        this._refresh_frame = null;
        this._chart_container = null;
        this._canvas = null;
        this._chart_color = null;
        this._min = null;
        this._max = null;
        this._value = null;
        this._last_value = null;
        this._units = '';
    }

    processUpdateResponse(response) {
        if (response.history === null) {
            this._value = null;
            this._units = '';
        }
        else {
            this._value = Number(response.history.value);
            this._units = response.fields_values.value_units == WidgetLessonGaugeChart.UNIT_AUTO
                ? response.history.units
                : response.fields_values.value_static_units;
        }

        this._chart_color = response.fields_values.chart_color;
        this._min = Number(response.fields_values.value_min);
        this._max = Number(response.fields_values.value_max);

        super.processUpdateResponse(response);
    }

    setContents(response) {
        if (this._canvas === null) {
            super.setContents(response);

            this._chart_container = this._body.querySelector('.chart');
            this._chart_container.style.height =
                `${this._getContentsSize().height - this._body.querySelector('.description').clientHeight}
            this._canvas = document.createElement('canvas');

            this._chart_container.appendChild(this._canvas);

            this._resizeChart();
        }
```

```
        this._updatedChart();
    }

    onResize() {
        super.onResize();

        if (this._state === WIDGET_STATE_ACTIVE) {
            this._resizeChart();
        }
    }

    _resizeChart() {
        const ctx = this._canvas.getContext('2d');
        const dpr = window.devicePixelRatio;

        this._canvas.style.display = 'none';
        const size = Math.min(this._chart_container.offsetWidth, this._chart_container.offsetHeight);
        this._canvas.style.display = '';

        this._canvas.width = size * dpr;
        this._canvas.height = size * dpr;

        ctx.scale(dpr, dpr);

        this._canvas.style.width = `${size}px`;
        this._canvas.style.height = `${size}px`;

        this._refresh_frame = null;

        this._updatedChart();
    }

    _updatedChart() {
        if (this._last_value === null) {
            this._last_value = this._min;
        }

        const start_time = Date.now();
        const end_time = start_time + 400;

        const animate = () => {
            const time = Date.now();

            if (time <= end_time) {
                const progress = (time - start_time) / (end_time - start_time);
                const smooth_progress = 0.5 + Math.sin(Math.PI * (progress - 0.5)) / 2;
                let value = this._value !== null ? this._value : this._min;
                value = (this._last_value + (value - this._last_value) * smooth_progress - this._min) / (t

                const ctx = this._canvas.getContext('2d');
                const size = this._canvas.width;
                const char_weight = size / 12;
                const char_shadow = 3;
                const char_x = size / 2;
                const char_y = size / 2;
                const char_radius = (size - char_weight) / 2 - char_shadow;

                const font_ratio = 32 / 100;

                ctx.clearRect(0, 0, size, size);

                ctx.beginPath();
```

```
                ctx.shadowBlur = char_shadow;
                ctx.shadowColor = '#bbb';
                ctx.strokeStyle = '#eee';
                ctx.lineWidth = char_weight;
                ctx.lineCap = 'round';
                ctx.arc(char_x, char_y, char_radius, Math.PI * 0.749, Math.PI * 2.251, false);
                ctx.stroke();

                ctx.beginPath();
                ctx.strokeStyle = `#${this._chart_color}`;
                ctx.lineWidth = char_weight - 2;
                ctx.lineCap = 'round';
                ctx.arc(char_x, char_y, char_radius, Math.PI * 0.75,
                    Math.PI * (0.75 + (1.5 * Math.min(1, Math.max(0, value)))), false
                    );
                ctx.stroke();

                ctx.shadowBlur = 2;
                ctx.fillStyle = '#1f2c33';
                ctx.font = `${(char_radius * font_ratio)|0}px Arial`;
                ctx.textAlign = 'center';
                ctx.textBaseline = 'middle';
                ctx.fillText(`${this._value !== null ? this._value : t('No data')}${this._units}`,
                    char_x, char_y, size - char_shadow * 4 - char_weight * 2
                );

                ctx.fillStyle = '#768d99';
                ctx.font = `${(char_radius * font_ratio * .5)|0}px Arial`;
                ctx.textBaseline = 'top';

                ctx.textAlign = 'left';
                ctx.fillText(`${this._min}${this._min != '' ? this._units : ''}`,
                    char_weight * .75, size - char_weight * 1.25, size / 2 - char_weight
                );

                ctx.textAlign = 'right';
                ctx.fillText(`${this._max}${this._max != '' ? this._units : ''}`,
                    size - char_weight * .75, size - char_weight * 1.25, size / 2 - char_weight
                );

                requestAnimationFrame(animate);
            }
            else {
                this._last_value = this._value;
            }
        };

        requestAnimationFrame(animate);
    }
}
```

10. Otwórz *manifest.json* i dodaj:

- nazwę pliku (*class.widget.js*) do tablicy w sekcji *assets/js*;
- nazwę klasy (*WidgetLessonGaugeChart*) do parametru *js_class* w sekcji *widget*.

Klasa *WidgetLessonGaugeChart* będzie teraz automatycznie ładowana wraz z pulpitem.

ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
```

```
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    },
    "widget": {
        "js_class": "WidgetLessonGaugeChart"
    },
    "assets": {
        "js": ["class.widget.js"]
    }
}
```

Dodawanie stylów CSS do widgetu

W tej sekcji dowiesz się, jak dodać niestandardowe style CSS, aby widget wyglądał bardziej atrakcyjnie.

1. Dla stylów widgetu utwórz nowy katalog *css* w katalogu *assets*.
2. Utwórz plik *widget.css* w katalogu *assets/css*. Aby nadać styl elementom widgetu, użyj selektora *div.dashboard-widget-{widget id}*. Aby skonfigurować CSS dla całego widgetu, użyj selektora *form.dashboard-widget-{widget id}*

ui/modules/lesson_gauge_chart/assets/css/widget.css

```
div.dashboard-widget-lesson_gauge_chart {
    display: grid;
    grid-template-rows: 1fr;
    padding: 0;
}

div.dashboard-widget-lesson_gauge_chart .chart {
    display: grid;
    align-items: center;
    justify-items: center;
}

div.dashboard-widget-lesson_gauge_chart .chart canvas {
    background: white;
}

div.dashboard-widget-lesson_gauge_chart .description {
    padding-bottom: 8px;
    font-size: 1.750em;
    line-height: 1.2;
    text-align: center;
}

.dashboard-grid-widget-hidden-header div.dashboard-widget-lesson_gauge_chart .chart {
    margin-top: 8px;
}
```

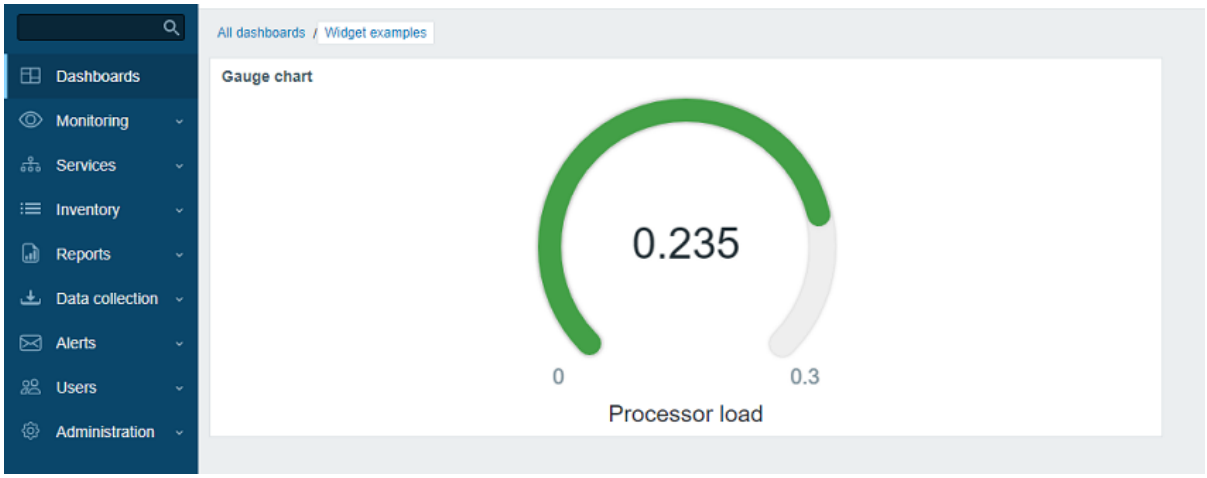
3. Otwórz plik *manifest.json* i dodaj nazwę pliku CSS (*widget.css*) do tablicy w sekcji *assets/css*. Umożliwi to wczytanie stylów CSS zdefiniowanych w pliku *widget.css* wraz ze stroną pulpitu.

ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
```

```
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    },
    "widget": {
        "js_class": "WidgetLessonGaugeChart"
    },
    "assets": {
        "css": ["widget.css"],
        "js": ["class.widget.js"]
    }
}
```

4. Odśwież stronę pulpitu, aby zobaczyć gotową wersję widgetu.



Przykłady

Ta sekcja zawiera pliki przykładowych modułów i widgetów, których możesz użyć jako podstawy do tworzenia własnych modułów.

Aby użyć modułu:

1. Pobierz archiwum ZIP.
2. Rozpakuj zawartość do oddzielnego katalogu w katalogu *modules* Twojej instalacji frontowej Zabbix (na przykład *zabbix/ui/modules*).
3. Zarejestruj moduł w interfejsie frontu Zabbix.

Przykład modułu Podczas tworzenia grupy hostów, przyznaj uprawnienia do odczytu skonfigurowanym grupom użytkowników - hg_auto_perm.zip

Przykłady widżetów

- Minimalny widżet - widget_min.zip
- Widżet ”Hello, world” wykorzystujący wyłącznie CSS - hello_world_css.zip
- Widżet ”Hello, world” wykorzystujący wyłącznie JavaScript - hello_world_js.zip
- Widżet ”Hello, world” wykorzystujący PHP - hello_world_php.zip

> **Note:**
> Jako przykłady możesz również użyć natywnych widżetów Zabbixa.

Biblioteka Python dla Zabbix

zabbix_utils to oficjalna biblioteka Pythona dla Zabbix.

Za pomocą zabbix_utils możesz:

- Używać API Zabbix do zarządzania obiektami Zabbix (tworzyć hosty, aktualizować pozycje, pobierać zdarzenia itp.).
- Zbierać dane z agent Zabbix (podobnie jak za pomocą Zabbix get).
- Wysyłać dane do serwer Zabbix lub proxy (podobnie jak za pomocą Zabbix sender).

Biblioteka obsługuje zarówno synchroniczne, jak i asynchroniczne wykonywanie skryptów, co pozwala jej obsługiwać zarówno proste przepływy pracy, jak i zakrojone na szeroką skalę operacje równoległe, takie jak masowe zbieranie danych, eksport zbiorczy, przetwarzanie alertów oraz efektywne skalowanie systemu.

Biblioteka jest zgodna z obsługiwanymi wersjami serwer Zabbix oraz nowoczesnymi wydaniami Pythona.

Aby rozpocząć, zainstaluj bibliotekę, a następnie skorzystaj z przewodnika szybki start.

Instalacja

zabbix_utils można zainstalować przy użyciu dowolnej z metod opisanych poniżej.

Wymagania:

- Zabbix 6.0 lub nowszy (testowano na 6.0, 7.0, 7.2, 7.4)
- Python 3.8 lub nowszy (testowano na 3.8–3.13)
- asynchroniczny framework HTTP aiohttp (wymagany tylko w trybie asynchronicznym)

Python Package Index (PyPI)

Jest to najczęściej stosowana metoda w większości środowisk:

```
pip install zabbix_utils
pip install zabbix_utils[async] # Tylko dla obsługi async
```

Repozytorium Zabbix

Użyj tej metody, jeśli wolisz zarządzać zależnościami za pomocą menedżera pakietów swojego systemu.

Pobierz pakiety Zabbix dla swojej dystrybucji Linuksa, a następnie uruchom poniższe polecenia.

W systemie RHEL i pochodnych:

```
dnf install python3-zabbix-utils
dnf install epel-release # Tylko dla obsługi async
dnf install python3-aiohttp # Tylko dla obsługi async
```

W systemach Debian/Ubuntu i pochodnych:

```
apt install python3-zabbix-utils
apt install python3-aiohttp # Tylko dla obsługi async
```

Źródło (GitHub)

Użyj tej metody, jeśli preferujesz najnowszą wersję rozwojową:

```
git clone https://github.com/zabbix/python-zabbix-utils
cd python-zabbix-utils/
pip install -r requirements.txt # Tylko dla obsługi asynchronicznej
python3 setup.py install
```

Aby zainstalować zależności wymagane dla trybu asynchronicznego, możesz także użyć jednej z metod opisanych powyżej.

Przewodnik szybkiego startu

Po zainstalowaniu zabbix_utils możesz używać go w swoim skrypcie.

Ten przewodnik szybkiego startu pokazuje, jak:

- Pobierać nazwę hosta z Zabbix agent.
- Tworzyć host i pozycja typu trapper za pomocą Zabbix API.
- Wysyłać wartość do pozycji.

Przewodnik przedstawia skrypt krok po kroku, wyjaśniając każdą część w miarę jej wprowadzania. Na końcu strony znajduje się kompletny skrypt.

W przewodniku założono również, że Twój serwer Zabbix, agent i API działają lokalnie.

Pobieranie nazwy hosta z Zabbix agent

Zacznij od pobrania nazwy hosta systemu, na którym działa Zabbix agent. Ta nazwa hosta będzie potrzebna do utworzenia hosta przy użyciu Zabbix API.

1. Zaimportuj klasę `Getter` z zabbix_utils. Ta klasa działa podobnie jak Zabbix get i umożliwia żądanie danych z Zabbix agent.

2. Utwórz instancję `Getter`, która łączy się z lokalnym Zabbix agent pod adresem `127.0.0.1` na porcie 10050.

3. Wywołaj metodę `get()` na instancji `Getter`, aby zażądać nazwy hosta z Zabbix agent.

- Metoda `get()` przyjmuje klucz pozycji jako parametr i wysyła żądanie do Zabbix agent o tę pozycję.
- Metoda `get()` zwraca obiekt, a nazwa hosta jest przechowywana w atrybucie `value` tego obiektu.
- `system.hostname` to wbudowany klucz Zabbix agent, który zwraca nazwę hosta.

```
from zabbix_utils import Getter

agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value
```

Połącz się i zaloguj do Zabbix API

Następnie połącz się z Zabbix API. Pozwoli to Twojemu skryptowi zarządzać obiektami Zabbix, takimi jak hosty i pozycje.

1. Zaimportuj klasę `ZabbixAPI` z zabbix_utils.

2. Utwórz instancję `ZabbixAPI` i podaj adres URL swojego interfejsu webowego Zabbix.

3. Wywołaj metodę `login()` na instancji `ZabbixAPI` i podaj swoją nazwę użytkownika oraz hasło. Konto musi mieć uprawnienie do uzyskiwania dostępu do Zabbix API.

4. Wywołaj metodę `logout()`, aby zamknąć sesję po zakończeniu operacji API.

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### API operations go here

api.logout()
```

Utwórz host w Zabbix

Teraz, gdy masz nazwę hosta z Zabbix agent i jesteś połączony z Zabbix API, możesz utworzyć nowy host w Zabbix.

1. Wywołaj metodę API `host.create()` na instancji `ZabbixAPI` i podaj szczegóły hosta:

- `host` — ustaw na zmienną `hostname` zawierającą nazwę hosta pobraną z Zabbix agent.
- `interfaces` — zawiera szczegóły połączenia dla Zabbix agent uruchomionego na hoście.
- `groups` — zawiera co najmniej jedną grupę hostów, do której host powinien należeć.

2. Ponieważ Zabbix API zwraca ID nowo utworzonego hosta, zapisz to ID w zmiennej `host_id` do późniejszego użycia.

3. Wyświetl komunikat potwierdzający, że host został utworzony.

```
api.host.create(
    host=hostname,
    interfaces=[{
        'type': 1,
        'main': 1,
        'useip': 1,
        'ip': '127.0.0.1',
        'dns': '',
        'port': '10050',
    }],
    groups=[{'groupid': '2'}]
)
```

```
host_id = host['hostids'][0]

print(f"Host '{hostname}' created with ID {host_id}")
```

Utwórz nową pozycję w Zabbix

Po utworzeniu hosta możesz dodać do niego pozycję.

1. Zdefiniuj unikalny klucz dla pozycji, która ma zostać utworzona.

2. Wywołaj metodę `item.create()` na instancji `ZabbixAPI` i podaj szczegóły pozycji:

   - `hostid` - ustaw na zmienną `host_id` zawierającą ID hosta, który został właśnie utworzony.
   - `name` - nazwa pozycji.
   - `key_` - ustaw na zmienną `item_key`, którą właśnie zdefiniowano.
   - `type` - ustaw na 2 (pozycja trapper); jest to wymagane do odbierania wartości wysyłanych ze skryptu w kolejnych krokach.
   - `value_type` - ustaw na 3 (numeric unsigned), czyli typ danych przechowywanych przez tę pozycję.

3. Ponieważ Zabbix API zwraca ID nowo utworzonej pozycji, zapisz to ID w zmiennej `item_id` do późniejszego użycia.

4. Wyświetl komunikat potwierdzający utworzenie pozycji.

```
item_key = 'app.myservice.heartbeat'

item = api.item.create(
    hostid=host_id,
    name='App heartbeat',
    key_=item_key,
    type=2,
    value_type=3,
)

item_id = item['itemids'][0]

print(f"Item '{item_key}' created with ID {item_id}")
```

Wyślij wartość do hosta

Teraz, gdy utworzyłeś host i pozycję, możesz wysłać do niej dane.

1. Zaimportuj klasę `time` i odczekaj kilka sekund przed wysłaniem danych. Dzięki temu Zabbix w pełni przetworzy nowy host i pozycję.

2. Zaimportuj klasę `Sender` z zabbix_utils. Ta klasa działa podobnie jak Zabbix sender i umożliwia skryptowi wysyłanie danych do Zabbix.

3. Utwórz instancję `Sender`, która łączy się z lokalnym serwerem Zabbix pod adresem `127.0.0.1` na porcie `10051`.

4. Wywołaj metodę `send_value()` na instancji `Sender`, aby wysłać wartość do pozycji hosta, i podaj następujące szczegóły:

   - `hostname` — ustaw na zmienną `hostname` zawierającą nazwę hosta pobraną z Zabbix agent.
   - `item_key` — ustaw na zmienną `item_key`, którą właśnie wcześniej zdefiniowałeś w skrypcie.
   - `1` — wartość do wysłania.

4. Wyświetl komunikat potwierdzający, że wartość została wysłana.

```
import time
time.sleep(10)

from zabbix_utils import Sender

sender = Sender(server='127.0.0.1', port=10051)

response = sender.send_value(hostname, item_key, 1)

print(f"Sender response: {response}")
```

Po pomyślnym wykonaniu tego kodu powinien zostać wyświetlony komunikat o powodzeniu informujący, że Zabbix odebrał wartość:

```
Sender response: {"processed": 1, "failed": 0, "total": 1, "time": "0.000151", "chunk": 1}
```

Teraz możesz również sprawdzić wartość w interfejsie webowym Zabbix (*Monitoring* > *Latest data*).

Kompletny skrypt

Poniżej znajduje się kompletny skrypt, który łączy wszystkie kroki: pobranie nazwy hosta, utworzenie hosta i pozycji przy użyciu Zabbix API oraz wysłanie danych do Zabbix.

```
import time
from zabbix_utils import Getter, ZabbixAPI, Sender

### Pobierz nazwę hosta z Zabbix agent
agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value

### Połącz z Zabbix API i zaloguj się
api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### Utwórz host w Zabbix
host = api.host.create(
    host=hostname,
    interfaces=[{
        'type': 1,
        'main': 1,
        'useip': 1,
        'ip': '127.0.0.1',
        'dns': '',
        'port': '10050',
    }],
    groups=[{'groupid': '2'}]
)

host_id = host['hostids'][0]

print(f"Host '{hostname}' created with ID {host_id}")

### Utwórz pozycję w Zabbix
item_key = 'app.myservice.heartbeat'

item = api.item.create(
    hostid=host_id,
    name='App heartbeat',
    key_=item_key,
    type=2,
    value_type=3
)

item_id = item['itemids'][0]

print(f"Item '{item_key}' created with ID {item_id}")

### Wyloguj się z API
api.logout()

### Poczekaj, aż Zabbix przetworzy nowy host
time.sleep(10)

### Wyślij wartość do hosta
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value(hostname, item_key, 1)

print(f"Sender response: {response}")
```

Korzystanie z API Zabbix

zabbix_utils umożliwia korzystanie z API Zabbix do zarządzania obiektami Zabbix, w tym tworzenia hostów, aktualizowania pozycji, pobierania zdarzeń i nie tylko.

Żądania API mogą być wykonywane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła żądanie i czeka na odpowiedź przed kontynuowaniem działania, co jest odpowiednie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła żądania bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej efektywne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać zabbix_utils do pracy z API Zabbix, zaimportuj klasę `ZabbixAPI` w swoim skrypcie:

```
from zabbix_utils import ZabbixAPI
```

> **Note:**
> Jeśli już używasz biblioteki Python społeczności, zazwyczaj możesz zastąpić ten import przez `ZabbixAPI` z zabbix_utils.

Logowanie

Przed wykonywaniem żądań API należy utworzyć instancję `ZabbixAPI` i zalogować się do Zabbix API.

Wybierz metodę, która najlepiej odpowiada sposobowi zarządzania poświadczeniami. Możesz użyć nazwy użytkownika i hasła lub tokenów API.

Podczas inicjalizacji

Podczas tworzenia instancji `ZabbixAPI` możesz od razu podać adres URL interfejsu webowego Zabbix oraz swoje dane uwierzytelniające:

```
### Nazwa użytkownika i hasło:
api = ZabbixAPI(url="127.0.0.1/zabbix", user="Admin", password="zabbix")

### Token API:
api = ZabbixAPI(url="127.0.0.1/zabbix", token="your_api_token")
```

Jeśli wolisz pogrupować parametry połączenia, możesz przekazać je jako słownik Pythona i rozpakować go za pomocą składni argumentów nazwanych Pythona:

```
ZABBIX_AUTH = {
    "url": "127.0.0.1/zabbix",
    "user": "Admin",
    "password": "zabbix"
}

api = ZabbixAPI(**ZABBIX_AUTH)
```

Używanie `login()`

Możesz najpierw utworzyć instancję `ZabbixAPI`, podając tylko URL interfejsu webowego Zabbix, a następnie użyć metody `login()` ze swoimi poświadczeniami:

```
### Nazwa użytkownika i hasło:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### Token API:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(token="your_api_token")
```

Używanie zmiennych środowiskowych

Możesz najpierw zapisać swoje dane uwierzytelniające jako zmienne środowiskowe, na przykład przy użyciu terminala:

```
### Nazwa użytkownika i hasło:
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"

### Token:
export ZABBIX_TOKEN="your_api_token"
```

Następnie, jeśli uruchomisz swój skrypt z tej samej sesji terminala, instancja `ZabbixAPI` w skrypcie wymaga jedynie adresu URL interfejsu webowego Zabbix:

```
api = ZabbixAPI(url="127.0.0.1/zabbix")
```

Możesz również zapisać adres URL w zmiennej środowiskowej:

```
### Nazwa użytkownika i hasło:
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"
export ZABBIX_URL="https://127.0.0.1/zabbix"

### Token:
export ZABBIX_TOKEN="your_api_token"
export ZABBIX_URL="https://127.0.0.1/zabbix"
```

Gdy adres URL i dane uwierzytelniające są już ustawione jako zmienne środowiskowe, instancja `ZabbixAPI` w skrypcie nie wymaga już żadnych argumentów:

```
api = ZabbixAPI()
```

Wylogowanie

Jeśli zalogowano się przy użyciu nazwy użytkownika i hasła, po zakończeniu operacji API należy wywołać metodę `logout()`:

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### API actions go here

api.logout()
```

> **Note:**
> Wywołanie metody `logout()` nie jest wymagane podczas używania tokenów API.

Żądania API

Po zalogowaniu możesz wykonywać dowolne żądania API, wywołując metody opisane w opisie metod API Zabbix.

Metody API są wywoływane przy użyciu następującego formatu:

```
api_instance.zabbix_object.method(parameters)
```

> **Note:**
> Niektóre nazwy metod lub obiektów API Zabbix używają słów, które są słowami zastrzeżonymi w Pythonie (np. `import`). Aby uniknąć błędów Pythona, podczas używania ich w Pythonie dodaj podkreślenie do nazwy metody lub obiektu (np. `api.configuration.import_`).

Na przykład, aby wyświetlić listę hostów za pomocą `host.get`:

```
### 1. Importuj ZabbixAPI z zabbix_utils:
from zabbix_utils import ZabbixAPI

### 2. Utwórz instancję ZabbixAPI i zaloguj się:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### 3. Pobierz hosty pasujące do filtra:
```

```
hosts = api.host.get(
    filter={
        "host": [
            "Zabbix server",
            "Linux server"
        ]
    }
)

### 4. Iteruj po zwróconych hostach i wyświetl szczegóły każdego hosta:
for host in hosts:
    print(host)

### 5. Wyloguj się z API, aby zamknąć sesję:
api.logout()
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi wysyłać żądania API bez czekania na zakończenie każdego z nich. Może to zwiększyć wydajność skryptu, gdy musi on wykonać wiele żądań API lub gdy niektóre żądania zajmują dużo czasu.

Podczas korzystania z trybu asynchronicznego występują istotne różnice w porównaniu z trybem synchronicznym:

- Zaimportuj moduł `asyncio` języka Python (należy najpierw zainstalować wymagane zależności).
- Zaimportuj `AsyncZabbixAPI` zamiast `ZabbixAPI`.
- Napisz kod wewnątrz funkcji `async`.
- Używaj `await` podczas wywoływania metod API, w tym `login()` i `logout()`.

Attention:
Podczas tworzenia instancji `AsyncZabbixAPI` nie można zalogować się podczas inicjalizacji.

Na przykład, aby wyświetlić listę hostów za pomocą `host.get` w trybie asynchronicznym:

```
### 1. Zaimportuj asyncio dla trybu asynchronicznego oraz AsyncZabbixAPI z zabbix_utils:
import asyncio
from zabbix_utils import AsyncZabbixAPI

### 2. Zdefiniuj główną funkcję async, w której będą wykonywane wszystkie operacje API:
async def main():

    # 3. Utwórz instancję AsyncZabbixAPI i zaloguj się (wymaga await):
    api = AsyncZabbixAPI(url="127.0.0.1")
    await api.login(user="User", password="zabbix")

    # 4. Pobierz hosty pasujące do filtra (wymaga await):
    hosts = await api.host.get(
        filter={
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    )

    # 5. Iteruj po zwróconych hostach i wypisz szczegóły każdego hosta:
    for host in hosts:
        print(host)

    # 6. Wyloguj się z API, aby zamknąć sesję (wymaga await):
    await api.logout()

### 7. Uruchom główną funkcję async main() przy użyciu pętli zdarzeń asyncio:
asyncio.run(main())
```

Przykłady

Poniższe przykłady pokazują typowe zadania API Zabbix z użyciem biblioteki zabbix_utils.

Dodatkowe przykłady są dostępne w katalogu `examples` repozytorium GitHub zabbix_utils.

Zbieranie danych z agenta Zabbix

Przegląd

zabbix_utils umożliwia zbieranie danych z Zabbix agent (podobnie jak Zabbix get).

Dane mogą być zbierane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła żądanie i czeka na dane przed kontynuowaniem działania, co jest odpowiednie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła żądania bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej wydajne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać `zabbix_utils` do zbierania wartości pozycji, zaimportuj klasę `Getter` w swoim skrypcie Python:

```
from zabbix_utils import Getter
```

Dane żądania

Aby zażądać wartości pozycji:

1. Utwórz instancję `Getter`, określając adres IP i port swojego agenta Zabbix.
2. Wywołaj metodę `get()` na instancji `Getter`, określając klucz pozycji, którą chcesz pobrać.

Na przykład, aby zażądać danych dla pozycji `system.uname`:

```
agent = Getter(host='192.0.2.1', port=10050)
response = agent.get('system.uname')
```

Używanie niestandardowego IP

Jeśli serwer, na którym uruchamiany jest skrypt, ma wiele adresów IP, możesz określić `source_ip`, którego `Getter` ma używać podczas łączenia się z agentem Zabbix:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    source_ip='10.10.7.1'
)
```

Używanie timeoutu

Możesz ustawić `timeout` odpowiedzi dla `Getter`, aby kontrolować, jak długo skrypt ma czekać na odpowiedź od agent przed rezygnacją:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    timeout=30
)
```

Używanie szyfrowania

`Getter` nie zawiera wbudowanej obsługi szyfrowania, ale można ją zapewnić, tworząc wrapper z użyciem bibliotek firm trzecich:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementation of TLS PSK wrapper for the socket
    # ...

agent = Getter(
    host='192.0.2.1',
```

```
    port=10050,
    socket_wrapper=psk_wrapper
)
```

Odpowiedź

Odpowiedź z Zabbix agent jest przetwarzana przez bibliotekę i zwracana jako obiekt `AgentResponse`:

```
print(response)
### {
###     "error": null,
###     "raw": "Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64",
###     "value": "Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64"
### }

print(response.value)
### Linux zabbix_server 5.15.0-3.60.5.1.el9uek.x86_64

print(response.error)
### None
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi zbierać wartości bez czekania na nadejście każdej z nich. Może to zwiększyć wydajność skryptu, gdy musi on zebrać wiele wartości lub gdy niektóre wartości są zbierane przez dłuższy czas.

Podczas korzystania z trybu asynchronicznego występują ważne różnice w porównaniu z trybem synchronicznym:

- Zaimportuj moduł Python `asyncio` (najpierw musisz zainstalować wymagane zależności).
- Zaimportuj `AsyncGetter` zamiast `Getter`.
- Napisz kod wewnątrz funkcji `async`.
- Użyj `await` podczas wywoływania metody `get()`.

Na przykład, aby zebrać pojedynczą wartość przy użyciu trybu asynchronicznego:

```
### 1. Import asyncio for asynchronous mode, and AsyncGetter from zabbix_utils:
import asyncio
from zabbix_utils import AsyncGetter

### 2. Define the main async function where all data requests will be executed:
async def main():
    agent = AsyncGetter(host='192.0.2.1', port=10050)

    # 3. Fetch the system.uname value from Zabbix agent (must await):
    response = await agent.get('system.uname')

    # 4. Print the value returned by Zabbix agent:
    print(response.value)

### 5. Run the async main() function using asyncio's event loop:
asyncio.run(main())
```

Wysyłanie danych do serwera lub proxy Zabbix

zabbix_utils umożliwia wysyłanie wartości pozycji do pozycji trapper na serwer lub proxy Zabbix (podobnie jak Zabbix sender).

Możesz wysłać pojedynczą wartość, wiele wartości, a nawet kierować dane do wielu klastrów Zabbix.

Dane mogą być wysyłane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła wartości i czeka na odpowiedź przed kontynuowaniem działania; jest to odpowiednie rozwiązanie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła wartości bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej efektywne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać zabbix_utils do wysyłania wartości pozycji, zaimportuj klasę Sender w swoim skrypcie:

```
from zabbix_utils import Sender
```

Aby wysyłać wiele wartości, możesz również zaimportować klasę ItemValue:

```
from zabbix_utils import Sender, ItemValue
```

Wyślij pojedynczą wartość

Aby wysłać wartość pozycji:

1. Utwórz instancję Sender, podając adres IP i port swojego serwera lub proxy Zabbix.
2. Wywołaj metodę send_value() na instancji Sender, używając następującego formatu:

```
sender_instance.send_value('host', 'item.key', 'value', optional_timestamp, optional_nanoseconds)
```

Na przykład, aby wysłać 1 do pozycji trapper service.status na hoście Linux server:

```
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value('Linux server', 'service.status', 1)
```

Używanie niestandardowego adresu IP

Jeśli serwer uruchamiający skrypt ma wiele adresów IP, możesz określić source_ip, którego Sender ma używać podczas wysyłania wartości do serwera Zabbix lub proxy:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    source_ip='10.10.7.1'
)
```

Używanie timeout

Możesz ustawić timeout odpowiedzi dla Sender, aby kontrolować, jak długo skrypt powinien czekać na odpowiedź z serwera Zabbix lub proxy, zanim zrezygnuje:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    timeout=30
)
```

Używanie pliku konfiguracyjnego agent

Możesz pozwolić, aby zabbix_utils odczytywał parametry Server lub ServerActive z lokalnego pliku konfiguracyjnego Zabbix agent lub agent 2. W takich przypadkach nie musisz określać parametrów połączenia podczas tworzenia instancji Sender:

```
sender = Sender(
    use_config=True,
    config_path='/etc/zabbix/zabbix_agent2.conf'
)
```

Attention:
Jeśli ServerActive zawiera jeden lub więcej klastrów Zabbix z wieloma instancjami serwer, Sender wysyła dane do pierwszego dostępnego serwer w każdym klastrze. Jeśli ServerActive nie jest ustawiony, używany jest adres z Server z domyślnym portem (10051).

Używanie szyfrowania

Sender nie zawiera wbudowanej obsługi szyfrowania, ale można ją zapewnić, tworząc opakowanie przy użyciu bibliotek zewnętrznych:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementacja opakowania TLS PSK dla gniazda
```

```
    # ...

sender = Sender(
    server='127.0.0.1',
    port=10051,
    socket_wrapper=psk_wrapper
)
```

Odpowiedź dla pojedynczej wartości

Odpowiedź zwrócona przez serwer Zabbix lub proxy jest przetwarzana przez bibliotekę i zwracana jako obiekt TrapperResponse:

```
print(response)
### {"processed": 1, "failed": 0, "total": 1, "time": "0.000123", "chunk": 1}

print(response.processed)
### 1

print(response.failed)
### 0

print(response.total)
### 1
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi Python wysyłać wartości bez oczekiwania na odpowiedź z serwera Zabbix lub proxy. Może to zwiększyć wydajność skryptu, gdy musi on wysłać wiele wartości albo gdy wysłanie niektórych wartości zajmuje dużo czasu.

Podczas korzystania z trybu asynchronicznego występuje kilka istotnych różnic w porównaniu z trybem synchronicznym:

- Zaimportuj moduł asyncio języka Python (najpierw należy zainstalować wymagane zależności).
- Zaimportuj AsyncSender zamiast Sender.
- Napisz kod wewnątrz funkcji async.
- Użyj await podczas wywoływania metody send_value().

Na przykład, aby wysłać pojedynczą wartość w trybie asynchronicznym:

```
### 1. Import asyncio for asynchronous mode, and AsyncSender from zabbix_utils:
import asyncio
from zabbix_utils import AsyncSender

### 2. Define the main async function where all data sending operations (must await) will be executed:
async def main():
    sender = AsyncSender(server='127.0.0.1', port=10051)
    response = await sender.send_value('Linux server', 'service.status', 1)

    # 3. Print the response returned by Zabbix server or proxy:
    print(response)

### 4. Run the async main() function using asyncio's event loop:
asyncio.run(main())
```

Wysyłanie wielu wartości

Aby wysłać wiele wartości:

1. Przygotuj tablicę obiektów ItemValue, z których każdy używa tego samego formatu co metoda send_value().
2. Utwórz instancję Sender, podając adres IP i port serwera lub proxy Zabbix.
3. Wywołaj metodę send() (zamiast send_value()) na instancji Sender, przekazując tablicę obiektów z wartościami do wysłania.

Na przykład, aby wysłać pięć wartości do różnych hostów:

```
items = [
    ItemValue('server-de', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-fr', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-uk', 'service.status', 'up', 1770887205, 100),
    ItemValue('server-nl', 'service.status', 'up', 1770887205, 100),
```

```
    ItemValue('server-pl', 'service.status', 'up', 1770887205, 100),
]

sender = Sender(server='127.0.0.1', port=10051)
response = sender.send(items)
```

Używanie niestandardowego rozmiaru fragmentu

Jeśli chcesz wysłać więcej wartości, niż pozycja trapper może przyjąć w pojedynczym żądaniu, możesz podzielić je na fragmenty.

Domyślnie rozmiar fragmentu wynosi 250 wartości. Możesz go zmienić, ustawiając parametr `chunk_size` podczas tworzenia instancji `Sender`.

Na przykład, aby wysłać pięć wartości w trzech fragmentach (2-2-1), ustaw parametr `chunk_size` na 2:

```
items = [
    ItemValue('server-de', 'service.status', 'up'),
    ItemValue('server-fr', 'service.status', 'up'),
    ItemValue('server-uk', 'service.status', 'up'),
    ItemValue('server-nl', 'service.status', 'up'),
    ItemValue('server-pl', 'service.status', 'up'),
]

sender = Sender(server='127.0.0.1', port=10051, chunk_size=2)
response = sender.send(items)
```

Wysyłanie wartości do wielu klastrów Zabbix

Aby wysyłać wartości do wielu klastrów Zabbix:

1. Przygotuj tablicę klastrów Zabbix. Jeśli klaster ma wiele węzłów, wartość zostanie wysłana do pierwszego **dostępnego** węzła każdego klastra.
2. Utwórz obiekt `Sender`, podając tablicę klastrów Zabbix.
3. Wywołaj metodę `send_value()` na instancji `Sender`, używając tego samego formatu jak w metodzie `send_value()`.

Na przykład, aby wysłać wartość do pierwszego dostępnego węzła w każdym klastrze:

```
zabbix_clusters = [
    ['zabbix.cluster1.node1', 'zabbix.cluster1.node2:10051'],
    ['zabbix.cluster2.node1:10051', 'zabbix.cluster2.node2', 'zabbix.cluster2.node3']
]

sender = Sender(clusters=zabbix_clusters)
response = sender.send_value('Linux server', 'service.status', 1)
```

Odpowiedź dla wielu wartości

Domyślnie `Sender` zwraca zagregowany wynik wysyłania wartości dla wszystkich hostów lub klastrów:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}
```

Jeśli potrzebujesz bardziej szczegółowych informacji, możesz sprawdzić wyniki dla każdego klastra i każdego fragmentu za pomocą atrybutu `response.details`:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}

if response.failed == 0:
    print(f"Wartość została wysłana pomyślnie w {response.time}")
else:
    print(response.details)
    # {
    #     127.0.0.1:10051: [
    #         {
    #             "processed": 1,
    #             "failed": 0,
    #             "total": 1,
    #             "time": "0.000051",
```

```
    #             "chunk": 1
    #         }
    #     ],
    #     zabbix.example.local:10051: [
    #         {
    #             "processed": 1,
    #             "failed": 0,
    #             "total": 1,
    #             "time": "0.000057",
    #             "chunk": 1
    #         }
    #     ]
    # }

    for node, chunks in response.details.items():
        for resp in chunks:
            print(f"Przetworzono {resp.processed} z {resp.total} na {node.address}:{node.port}")
            # Przetworzono 1 z 1 na 127.0.0.1:10051
            # Przetworzono 1 z 1 na zabbix.example.local:10051
```

Rejestrowanie debugowania

Aby rozwiązywać problemy podczas pracy z Zabbix API, zbierania danych z Zabbix agent lub wysyłania danych do Zabbix serwer lub proxy, można włączyć rejestrowanie debugowania.

Biblioteka zabbix_utils używa standardowego modułu `logging` języka Python.

Aby wyświetlać szczegółowe komunikaty debugowania z biblioteki:

1. Zaimportuj do skryptu moduł `logging` języka Python.
2. Skonfiguruj system rejestrowania tak, aby wyprowadzał komunikaty w czytelnym formacie, i ustaw poziom rejestrowania na DEBUG.

```
import logging
from zabbix_utils import Getter

logging.basicConfig(
    format=u'[%(asctime)s] %(levelname)s %(message)s',
    level=logging.DEBUG
)

agent = Getter(host='127.0.0.1', port=10050)
resp = agent.get('system.uname')

print(resp.value)
```

Po włączeniu rejestrowania debugowania wyświetlane są szczegółowe informacje o żądaniach i odpowiedziach, na przykład:

```
[2023-10-01 12:00:01,587] DEBUG Content of the packet: b'ZBXD\x01\x0c\x00\x00\x00\x00\x00\x00\x00system.un
[2023-10-01 12:00:01,722] DEBUG Zabbix response header: b'ZBXD\x01C\x00\x00\x00C\x00\x00\x00'
[2023-10-01 12:00:01,723] DEBUG Zabbix response body: Linux test_server 5.15.0-3.60.5.1.el9uek.x86_64
[2023-10-01 12:00:01,724] DEBUG Response from [127.0.0.1:10050]: Linux test_server 5.15.0-3.60.5.1.el9uek.
Linux test_server 5.15.0-3.60.5.1.el9uek.x86_64
```

Wtyczki

Omówienie Niestandardowe ładowalne wtyczki rozszerzają funkcjonalność agent 2 Zabbix. Są kompilowane oddzielnie, ale korzystają z pakietu współdzielonego z agent 2 Zabbix.

Każda wtyczka to pakiet *Go*, który definiuje strukturę i implementuje jeden lub kilka interfejsów wtyczki (*Exporter*, *Configurator*, *Runner*).

Przejdź do:

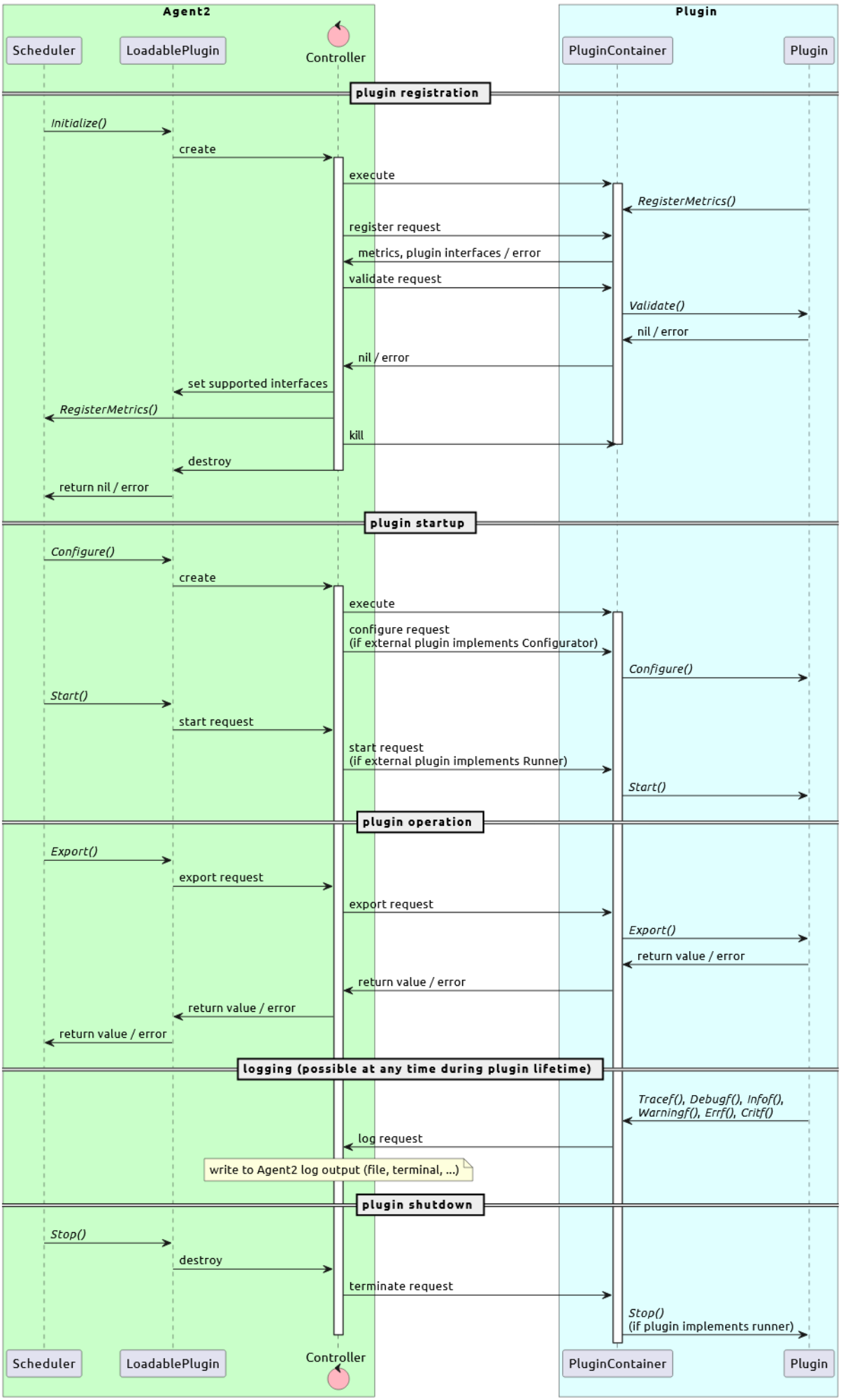
- Napisz swoją pierwszą wtyczkę
- Interfejsy wtyczek

Zobacz także:

- Przykładowa wtyczka dla agent 2 Zabbix

Diagram połączenia agent 2 Zabbix łączy się dwukierunkowo z wtyczkami za pomocą gniazd UNIX w systemie Linux oraz Named Pipes w systemie Windows.

Poniższy diagram połączenia ilustruje proces komunikacji między agent 2 Zabbix a ładowalną wtyczką oraz proces zbierania metryk.


Agent2
Plugin
Scheduler
LoadablePlugin
Controller
PluginContainer
Plugin
plugin registration
Initialize()
create
execute
RegisterMetrics()
register request
metrics, plugin interfaces / error
validate request
Validate()
nil / error
nil / error
set supported interfaces
RegisterMetrics()
kill
destroy
return nil / error
plugin startup
Configure()
create
execute
configure request
(if external plugin implements Configurator)
Configure()
Start()
start request
start request
(if external plugin implements Runner)
Start()
plugin operation
Export()
export request
export request
Export()
return value / error
return value / error
return value / error
return value / error
logging (possible at any time during plugin lifetime)
Tracef(), Debugf(), Infof(),
Warningf(), Errf(), Critf()
log request
write to Agent2 log output (file, terminal, ...)
plugin shutdown
Stop()
destroy
terminate request
Stop()
(if plugin implements runner)
Scheduler
LoadablePlugin
Controller
PluginContainer
Plugin

Przykłady

Jako punkt odniesienia można wykorzystać kilka pustych przykładów, a także istniejące ładowalne wtyczki dostarczane przez Zabbix:

- Przykładowa wtyczka dla Zabbix agent 2
- Wtyczka MSSQL
- Wtyczka MongoDB
- Wtyczka PostgreSQL

Zobacz pełną listę wtyczek dostarczanych przez Zabbix.

Interfejsy wtyczek

Ta sekcja opisuje dostępne interfejsy wtyczek.

plugin.Exporter *Exporter* to najprostszy interfejs, który wykonuje badanie, zwraca wartość (wartości), nic lub błąd. Akceptuje przygotowany klucz elementu, parametry i kontekst. Dostęp do wszystkich innych interfejsów wtyczek jest wyłączny i żadna metoda nie może zostać wywołana, jeśli wtyczka wykonuje już zadanie. Ponadto istnieje limit maksymalnie 100 równoczesnych wywołań *Export()* dla wtyczki, który można zmniejszyć zgodnie z wymaganiami dla każdej wtyczki.

plugin.Configurator Interfejs *Configurator* udostępnia parametry konfiguracji wtyczki z plików konfiguracyjnych agenta Zabbix 2.

plugin.Runner Interfejs *Runner* zapewnia możliwości do wykonywania inicjalizacji, gdy wtyczka jest uruchamiana (aktywowana) i deinicjalizacji, gdy wtyczka jest zatrzymywana (dezaktywowana). Na przykład wtyczka może uruchomić/zatrzymać jakąś w tle *goroutine*, implementując interfejs Runner.

Stwórz wtyczkę (samouczek)

To jest samouczek, który krok po kroku pokaże jak utworzyć prostą ładowalną wtyczkę dla agenta Zabbix 2.

Możesz też używać naszego przykładowego repozytorium jako szablonu lub przewodnika do tworzenia własnych wtyczek.

Co stworzysz Ten samouczek pokazuje, jak utworzyć nową wtyczkę **MyIP**. Wtyczka zaimplementuje pojedynczy klucz pozycji, **myip**, który zwraca zewnętrzny adres IP hosta, na którym uruchomiony jest agent Zabbix 2.

Krok 1: Konfiguracja

1. Plugin jest standardowym modułem Go. Zacznij od zainicjowania pliku `go.mod` w katalogu pluginu, aby śledzić zależności pluginu:

```
cd path/to/plugins/myip # Przełącz się do katalogu swojego pluginu
go mod init myip
```

2. Zainstaluj obowiązkową zależność Zabbix Go SDK (`golang.zabbix.com/sdk`):

```
go get golang.zabbix.com/sdk@$LATEST_COMMIT_HASH
```

Zastąp `$LATEST_COMMIT_HASH` najnowszym hashem commita `HEAD` z repozytorium `golang.zabbix.com/sdk` repository w odpowiedniej gałęzi wydania. Na przykład:

```
go get golang.zabbix.com/sdk@af85407
```

Należy pamiętać, że wersjonowanie `golang.zabbix.com/sdk` nie jest obecnie obsługiwane, ale może się to zmienić w przyszłości.

W razie potrzeby można zainstalować dodatkowe zależności za pomocą `go get`.

3. Utwórz pusty plik `main.go` dla kodu źródłowego pluginu:

```
touch main.go
```

Teraz wstępna konfiguracja jest zakończona, a plugin jest gotowy do dalszego rozwoju.

Krok 2: Struktura pluginu Moduł `golang.zabbix.com/sdk`, zainstalowany w poprzednim kroku, zapewnia niezbędne ramy do tworzenia pluginów i gwarantuje, że wszystkie pluginy mają spójną strukturę.

1. Skonfiguruj podstawowy przepływ wykonywania.

Zacznij od zdefiniowania głównego przepływu wykonywania pluginu. Dodaj poniższy kod do `main.go`:

```
package main

func main() {
err := run()
if err != nil {
panic(err)
}
}

func run() error {
return nil
}
```

To ustanawia podstawowy przepływ wykonywania dla pluginu. Funkcja run będzie później zawierać główną logikę pluginu.

2. Poznaj interfejsy pluginu.

Plugin Zabbix agent 2 powinien być reprezentowany przez strukturę, która implementuje interfejsy z pakietu `golang.zabbix.com/sdk/plug`

- *Accessor* - definiuje podstawowe metody, które wszystkie pluginy muszą implementować, takie jak ustawianie nazwy pluginu i obsługa limitów czasu dla kluczy pozycji.
- Jeden lub więcej z następujących funkcjonalnych interfejsów pluginu:
- *Exporter* - wykonuje odpytywanie i zwraca wartość (lub wartości), nic albo błąd; często używany razem z interfejsem *Collector*.
- *Collector* - zarządza okresowym zbieraniem danych.
- *Runner* - definiuje procedury uruchamiania i zamykania pluginu.
- *Watcher* - umożliwia implementację niezależnego odpytywania metryk z pominięciem wewnętrznego harmonogramu agenta; przydatny w monitorowaniu opartym na pułapkach lub zdarzeniach.
- *Configurator* - definiuje sposób, w jaki plugin odczytuje i stosuje swoje ustawienia konfiguracyjne.

Możesz zaimplementować te interfejsy samodzielnie albo użyć domyślnych, dostarczonych przez Zabbix Go SDK, modyfikując je w razie potrzeby. Ten samouczek korzysta z domyślnych implementacji.

3. Utwórz strukturę pluginu.

Teraz zaimportuj pakiet *plugin* i utwórz strukturę `myIP`, która osadza strukturę `plugin.Base`:

```
import "golang.zabbix.com/sdk/plugin"

type myIP struct {
plugin.Base
}
```

Struktura myIP obecnie spełnia interfejs Accessor. Metoda implementująca jeden z funkcjonalnych interfejsów pluginu, `Exporter`, zostanie dodana później w samouczku.

Krok 3: Zdefiniuj klucze pozycji Twój plugin potrzebuje kluczy pozycji, aby zbierać dane i przekazywać je do serwer lub proxy Zabbix.

1. Zaimportuj pakiet *errs* do obsługi błędów:

```
import "golang.zabbix.com/sdk/errs"
```

2. Zarejestruj klucze pozycji za pomocą funkcji `plugin.RegisterMetrics()` wewnątrz funkcji `run()`:

```
func run() error {
p := &myIP{}

// Zarejestruj klucz pozycji `myip`.
```

```
err := plugin.RegisterMetrics(
p,
"MyIP",                          // Nazwa pluginu
"myip",                          // Nazwa klucza pozycji
"Zwraca adres IP hosta.",        // Opis klucza pozycji
)
if err != nil {
return errs.Wrap(err, "nie udało się zarejestrować metryk")
}

return nil
}
```

Aby zarejestrować kilka kluczy pozycji, powtórz parametry *nazwa metryki* i *opis* dla każdej metryki. Na przykład:

```
plugin.RegisterMetrics(&impl, "Myip", "metric.one", "Opis metryki pierwszej.", "metric.two", "Opis metryki
```

Krok 4: Skonfiguruj handler Handler ułatwia komunikację między agentem a wtyczką.

1. Zaimportuj pakiet *container*:

```
import "golang.zabbix.com/sdk/plugin/container"
```

2. Wewnątrz funkcji `run()` dodaj kod, aby utworzyć i skonfigurować handler:

```
func run() error {
p := &myIP{}

// Zarejestruj klucz pozycji `myip`.
err := plugin.RegisterMetrics(
p,
"MyIP",                          // Nazwa wtyczki
"myip",                          // Nazwa klucza pozycji
"Zwraca adres IP hosta.",        // Opis klucza pozycji
)
if err != nil {
return errs.Wrap(err, "failed to register metrics")
}

// Utwórz nowy handler.
h, err := container.NewHandler("MyIP") // Nazwa wtyczki
if err != nil {
return errs.Wrap(err, "failed to create new handler")
}

// Skonfiguruj logowanie, aby przekazywać logi z wtyczki do agenta.
// Dostępne przez p.Logger.Infof, p.Logger.Debugf itd.
p.Logger = h

// Uruchom wykonanie wtyczki.
// Blokuje do momentu otrzymania żądania zakończenia od agenta.
err = h.Execute()
if err != nil {
return errs.Wrap(err, "failed to execute plugin handler")
}

return nil
}
```

Krok 5: Implementacja zbierania danych Zbieranie danych odbywa się za pośrednictwem interfejsu Exporter, który opisuje metodę `Export`:

```
func Export(
  key string,              // Klucz pozycji do zebrania.
  params []string,         // Argumenty przekazane do klucza pozycji (`myip[arg1, arg2]`).
```

```
  context ContextProvider // Metadane dotyczące zbierania danych dla klucza pozycji.
) (any, error)
```

1. Zaimportuj wymagane pakiety do obsługi żądań HTTP i odczytu odpowiedzi:

```
import (
"io"
"net/http"
)
```

2. Zaimplementuj metodę `Export` dla struktury `myIP`:

```
func (p *myIP) Export(
key string, params []string, context plugin.ContextProvider,
) (any, error) {
// Wtyczka może używać różnych metod zbierania danych w zależności od parametru `key`.
    // Ta implementacja jedynie sprawdza, czy podany `key` jest obsługiwany.
if key != "myip" {
return nil, errs.Errorf("unknown item key %q", key)
}

// Log zostanie przekazany do logu agent 2.
p.Infof(
"received request to handle %q key with %d parameters",
key,
len(params),
)

// Zbierz dane i zwróć wynik.

resp, err := http.Get("https://api.ipify.org")
if err != nil {
return nil, errs.Wrap(err, "failed to get IP address")
}

defer resp.Body.Close()

body, err := io.ReadAll(resp.Body)
if err != nil {
return nil, errs.Wrap(err, "failed to read response body")
}

return string(body), nil
}
```

Krok 6: Zbuduj i skonfiguruj plugin

1. Aby zbudować plugin, uruchom:

```
go mod tidy
go build
```

To powinno utworzyć plik wykonywalny `myip` w bieżącym katalogu.

2. Skonfiguruj agent Zabbix 2, aby używał pluginu:

```
echo "Plugins.MyIP.System.Path=$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE" > /etc/zabbix_agent2.d/plugins.d/myip.
```

Zastąp `$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE` ścieżką do `myip` utworzonego w kroku 5.

Nazwa pluginu w nazwie parametru konfiguracji (*MyIP* w tym samouczku) musi być zgodna z nazwą pluginu zdefiniowaną w funkcji *plugin.RegisterMetrics()*.

3. Aby przetestować plugin i jego pozycję `myip`, uruchom:

```
zabbix_agent2 -c /etc/zabbix_agent2.conf -t myip
```

Wynik powinien zawierać zewnętrzny adres IP twojego hosta i wyglądać podobnie do tego:

myip [s|192.0.2.1]

W ten sposób utworzyłeś prosty, ładowalny plugin dla agenta Zabbix 2. Gratulacje!

```
package main

import (
"io"
"net/http"

"golang.zabbix.com/sdk/errs"
"golang.zabbix.com/sdk/plugin"
"golang.zabbix.com/sdk/plugin/container"
)

var _ plugin.Exporter = (*myIP)(nil)

type myIP struct {
plugin.Base
}

func main() {
err := run()
if err != nil {
panic(err)
}
}

func run() error {
p := &myIP{}

// Zarejestruj klucz elementu `myip`.
err := plugin.RegisterMetrics(
p,
"MyIP", // Nazwa wtyczki
"myip", // Nazwa klucza pozycji
"Zwraca adres IP hosta.", // Opis klucza pozycji
)
if err != nil {
return errs.Wrap(err, "nie udało się zarejestrować metryk")
}

// Utwórz nowy program obsługi.
h, err := container.NewHandler("MyIP") // Nazwa wtyczki
if err != nil {
return errs.Wrap(err, "nie udało się utworzyć nowego programu obsługi")
}

// Skonfiguruj rejestrowanie, aby przekazywać logi z wtyczki do agenta.
// Dostępne za pośrednictwem p.Logger.Infof, p.Logger.Debugf itp.
p.Logger = h

// Rozpocznij wykonywanie wtyczki.
// Blokuje do momentu otrzymania żądania zakończenia od agenta.
err = h.Execute()
if err != nil {
return errs.Wrap(err, "nie udało się wykonać obsługi wtyczki")
}

return nil
}
```

```
func (p *myIP) Export(
key string, params []string, context plugin.ContextProvider,
) (any, error) {
// Wtyczka może używać innej logiki zbierania danych w oparciu o parametr `key`.
// Ta implementacja weryfikuje tylko, czy podany `key` jest obsługiwany.
if key != "myip" {
return nil, errs.Errorf("nieznany klucz elementu %q", key)
}

// Dziennik zostanie przesłany do dziennika agenta 2.
p.Infof(
"otrzymano żądanie obsługi klucza %q z %d parametrami",
key,
len(params),
)

// Zbierz dane i zwróć je.

resp, err := http.Get("https://api.ipify.org")
if err != nil {
return nil, errs.Wrap(err, "nie udało się pobrać adresu IP")
}

defer resp.Body.Close()

body, err := io.ReadAll(resp.Body)
if err != nil {
return nil, errs.Wrap(err, "nie udało się odczytać treści odpowiedzi")
}

return string(body), nil
}
```

Kompletny kod źródłowy

Zmiany w tworzeniu rozszerzeń

Ta strona zawiera listę wszystkich zmian, jeśli takie wystąpiły, dotyczących tworzenia niestandardowych rozszerzeń Zabbix.

Strony podręcznika Zabbix

Dokumentacja referencyjna narzędzi i usług Zabbix wiersza poleceń, zawierająca listę dostępnych poleceń, opcji i przykładów użycia.

zabbix_agent2

```
ZABBIX_AGENT2(8)            Podręcznik administratora systemu           ZABBIX_AGENT2(8)

NAZWA
       zabbix_agent2 - agent Zabbixa 2

SKŁADNIA
       zabbix_agent2 [-c config-file]
       zabbix_agent2 [-c config-file] [-v] -p
       zabbix_agent2 [-c config-file] [-v] -t item-key
       zabbix_agent2 [-c config-file] -R runtime-option
       zabbix_agent2 [-c config-file] -T
       zabbix_agent2 -h
```

```
zabbix_agent2 -V

OPIS
zabbix_agent2  to  aplikacja do monitorowania parametrów różnych usług.

OPCJE
-c, --config config-file
       Użyj alternatywnego pliku config-file zamiast domyślnego.

-R, --runtime-control runtime-option
       Wykonaj funkcje administracyjne zgodnie z runtime-option.

    Opcje sterowania w czasie działania
        userparameter_reload
            Załaduj ponownie parametry użytkownika z pliku konfiguracyjnego

        log_level_increase
            Zwiększ poziom logowania

        log_level_decrease
            Zmniejsz poziom logowania

        help
            Wyświetl dostępne opcje sterowania w czasie działania

        metrics
            Wyświetl dostępne metryki

        version
            Wyświetl wersję

-p, --print
       Wyświetl znane pozycje i zakończ działanie. Dla każdej pozycji
       używane są albo ogólne wartości domyślne, albo podawane są
       konkretne wartości domyślne do testowania. Te wartości domyślne
       są wyświetlane w nawiasach kwadratowych jako parametry klucza
       pozycji. Zwracane wartości są ujęte w nawiasy kwadratowe i
       poprzedzone typem zwracanej wartości, oddzielonym znakiem
       pionowej kreski. Dla parametrów użytkownika typ jest zawsze t,
       ponieważ agent nie może określić wszystkich możliwych wartości
       zwracanych. Pozycje oznaczone jako działające nie mają gwarancji
       działania z serwera Zabbixa lub zabbix_get podczas odpytywania
       działającego demona agenta, ponieważ uprawnienia lub środowisko
       mogą być inne. Typy zwracanych wartości to:

    d  Liczba z częścią dziesiętną.

    m  Nieobsługiwane. Może to być spowodowane odpytywaniem pozycji,
       która działa tylko w trybie aktywnym, na przykład pozycji
       monitorowania logów, albo pozycji wymagającej wielu zebranych
       wartości. Problemy z uprawnieniami lub nieprawidłowe parametry
       użytkownika również mogą skutkować stanem nieobsługiwane.

    s  Tekst. Maksymalna długość nie jest ograniczona.

    t  Tekst. To samo co s.

    u  Liczba całkowita bez znaku.

-t, --test item-key
       Przetestuj pojedynczą pozycję i zakończ działanie. Opis wyniku
       znajduje się w --print.
```

```
       -v, --verbose
              Włącz szczegółowe wyjście podczas testowania pozycji lub
              wyświetlania znanych pozycji.

       -T, --test-config
              Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.

       -h, --help
              Wyświetl tę pomoc i zakończ działanie.

       -V, --version
              Wyświetl informacje o wersji i zakończ działanie.

PLIKI
       /usr/local/etc/zabbix_agent2.conf
              Domyślna lokalizacja pliku konfiguracyjnego agenta Zabbixa 2
              (jeśli nie została zmieniona podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       zabbix_agentd(8),  zabbix_get(1),  zabbix_proxy(8),   zabbix_sender(1),
       zabbix_server(8), zabbix_js(1), zabbix_web_service(8)

Zabbix                            2022-01-06                  ZABBIX_AGENT2(8)
```

zabbix_agentd

```
ZABBIX_AGENTD(8)            Podręcznik administratora systemu           ZABBIX_AGENTD(8)

NAZWA
       zabbix_agentd - demon agenta Zabbix

SKŁADNIA
       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 [-c config-file] -T
       zabbix_agentd -h
       zabbix_agentd -V

OPIS
       zabbix_agentd jest demonem do monitorowania różnych parametrów serwera.

OPCJE
       -c, --config config-file
              Użyj alternatywnego config-file zamiast domyślnego.

       -f, --foreground
              Uruchom agent Zabbix na pierwszym planie.

       -R, --runtime-control runtime-option
              Wykonaj funkcje administracyjne zgodnie z runtime-option.

           Opcje sterowania w czasie działania
               userparameter_reload
                   Przeładuj parametry użytkownika z pliku konfiguracyjnego

               log_level_increase[=target]
                   Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony
```

```
            log_level_decrease[=target]
                Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony

        Cele sterowania poziomem logowania
            process-type
                Wszystkie procesy określonego typu (active checks, collector, listener)

            process-type,N
                Typ procesu i numer (np. listener,3)

            pid Identyfikator procesu, do 65535. Dla większych wartości określ target jako "process-typ

    -p, --print
           Wypisz znane pozycje i zakończ działanie. Dla każdej pozycji używane są albo ogólne wartości

        d   Liczba z częścią dziesiętną.

        m   Nieobsługiwane. Może to być spowodowane odpytywaniem pozycji, która działa tylko w trybie a

        s   Tekst. Maksymalna długość nie jest ograniczona.

        t   Tekst. Tak samo jak s.

        u   Liczba całkowita bez znaku.

    -t, --test item-key
           Przetestuj pojedynczą pozycję i zakończ działanie. Opis wyniku znajduje się w --print.

    -T, --test-config
           Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.

    -h, --help
           Wyświetl tę pomoc i zakończ działanie.

    -V, --version
           Wyświetl informacje o wersji i zakończ działanie.

PLIKI
    /usr/local/etc/zabbix_agentd.conf
           Domyślna lokalizacja pliku konfiguracyjnego agenta Zabbix (jeśli nie została zmodyfikowana p

ZOBACZ TAKŻE
    Dokumentacja

    zabbix_get(1),  zabbix_proxy(8),  zabbix_sender(1),   zabbix_server(8),
    zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                          2022-01-06                 ZABBIX_AGENTD(8)
```

zabbix_get

```
ZABBIX_GET(1)              Podręcznik poleceń ogólnych             ZABBIX_GET(1)

NAZWA
    zabbix_get - narzędzie Zabbix do pobierania danych

SKŁADNIA
    zabbix_get  -s  host-name-or-IP  [-p  port-number]  [-I IP-address] [-t
    timeout] -k item-key
    zabbix_get -s host-name-or-IP  [-p  port-number]  [-I  IP-address]  [-t
```

```
timeout]  --tls-connect  cert  --tls-ca-file  CA-file  [--tls-crl-file
CRL-file] [--tls-agent-cert-issuer cert-issuer]  [--tls-agent-cert-sub-
ject  cert-subject]  --tls-cert-file  cert-file --tls-key-file key-file
[--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k item-key
zabbix_get -s host-name-or-IP  [-p  port-number]  [-I  IP-address]  [-t
timeout]     --tls-connect    psk    --tls-psk-identity    PSK-identity
--tls-psk-file PSK-file  [--tls-cipher13  cipher-string]  [--tls-cipher
cipher-string] -k item-key
zabbix_get -h
zabbix_get -V
```

OPIS

zabbix_get to narzędzie wiersza poleceń służące do pobierania danych z agent.

OPCJE

-s, --host host-name-or-IP
Określa nazwę hosta lub adres IP host.

-p, --port port-number
Określa numer portu, na którym działa agent na hoście. Domyślnie jest to 10050.

-I, --source-address IP-address
Określa źródłowy adres IP.

-t, --timeout seconds
Określa limit czasu. Prawidłowy zakres: 1-600 sekund (domyślnie: 30)

-k, --key item-key
Określa klucz pozycji, dla której ma zostać pobrana wartość.

-P, --protocol value
Protokół używany do komunikacji z agent. Wartości:

auto	łączy przy użyciu protokołu JSON, a w razie niepowodzenia ponawia próbę z protokołem tekstowym (domyślnie)
json	łączy przy użyciu protokołu JSON
plaintext	łączy przy użyciu protokołu tekstowego, w którym wysyłany jest tylko klucz pozycji (wydania 6.4.x i starsze)

--tls-connect value
Sposób łączenia z agent. Wartości:

unencrypted	łączy bez szyfrowania (domyślnie)
psk	łączy przy użyciu TLS i klucza wstępnie współdzielonego
cert	łączy przy użyciu TLS i certyfikatu

--tls-ca-file CA-file
Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer.

--tls-crl-file CRL-file
Pełna ścieżka do pliku zawierającego unieważnione certyfikaty.

--tls-agent-cert-issuer cert-issuer
Dozwolony wystawca certyfikatu agent.

--tls-agent-cert-subject cert-subject
Dozwolony podmiot certyfikatu agent.

--tls-cert-file cert-file
Pełna ścieżka do pliku zawierającego certyfikat lub łańcuch certyfikatów.

--tls-key-file key-file
Pełna ścieżka do pliku zawierającego klucz prywatny.

--tls-psk-identity PSK-identity
Ciąg identyfikatora PSK.

--tls-psk-file PSK-file
Pełna ścieżka do pliku zawierającego klucz wstępnie współdzielony.

--tls-cipher13 cipher-string
Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów. Opcja ta nie jest dostępna, jeśli wersja OpenSSL jest niższa niż 1.1.1.

--tls-cipher cipher-string
Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfrów OpenSSL (tylko dla TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów.

-h, --help
Wyświetla tę pomoc i kończy działanie.

-V, --version
Wyświetla informacje o wersji i kończy działanie.

STATUS ZAKOŃCZENIA

Status zakończenia wynosi 0, jeśli zabbix_get pomyślnie pobierze wartość z agent. Jeśli pobieranie wartości nie powiedzie się, status zakończenia wynosi 1.

PRZYKŁADY

```
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=Signing CA,OU=IT operations,O=Example Corp,DC=example,DC=com" --tls-agent-cert-subject "CN=server1,OU=IT operations,O=Example Corp,DC=example,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
```

ZOBACZ TAKŻE

Dokumentacja

zabbix_agentd(8), zabbix_proxy(8), zabbix_sender(1), zabbix_server(8), zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

```
ZABBIX_JS(1)                Podręcznik poleceń ogólnych                ZABBIX_JS(1)

NAZWA
       zabbix_js - narzędzie Zabbix JS

SKŁADNIA
       zabbix_js -s plik-skryptu -p parametr-wejściowy [-l poziom-logowania] [-t limit-czasu]
       zabbix_js -s plik-skryptu -i plik-wejściowy [-l poziom-logowania] [-t limit-czasu]
       zabbix_js -h
       zabbix_js -V

OPIS
       zabbix_js  to  narzędzie wiersza poleceń, które można wykorzystać do testowania
       osadzonych skryptów.

OPCJE
       -s, --script plik-skryptu
              Określa nazwę pliku skryptu do wykonania. Jeśli jako nazwę pliku
              podano '-', skrypt zostanie odczytany ze stdin.

       -p, --param parametr-wejściowy
              Określa parametr wejściowy przekazywany do skryptu. Parametr jest
              przekazywany bez zmian jako ciąg znaków i jest dostępny w skrypcie
              za pośrednictwem zmiennej 'value'. Opcja wzajemnie wykluczająca się z -i.

       -i, --input plik-wejściowy
              Określa plik zawierający parametr wejściowy przekazywany do
              skryptu. Zawartość pliku jest przekazywana bez zmian jako ciąg znaków
              i jest dostępna w skrypcie za pośrednictwem zmiennej 'value'. Jeśli
              jako nazwę pliku podano '-', dane wejściowe zostaną odczytane ze stdin.
              Opcja wzajemnie wykluczająca się z -p.

       -w, --webdriver url
              Określa adres URL webdrivera.

       -l, --loglevel poziom-logowania
              Określa poziom logowania (zakres: 0-5).

       -t, --timeout limit-czasu
              Określa limit czasu w sekundach. Prawidłowy zakres: 1-600 sekund
              (domyślnie: 10)

       -h, --help
              Wyświetla tę pomoc i kończy działanie.

       -V, --version
              Wyświetla informacje o wersji i kończy działanie.

PRZYKŁADY
       zabbix_js -s plik-skryptu.js -p przykład

ZOBACZ TAKŻE
       Dokumentacja

       zabbix_agentd(8), zabbix_proxy(8), zabbix_sender(1), zabbix_server(8),
       zabbix_get(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                           2022-01-06                       ZABBIX_JS(1)
```

```
ZABBIX_PROXY(8)             Podręcznik administratora systemu            ZABBIX_PROXY(8)

NAZWA
       zabbix_proxy - demon proxy Zabbix

SKŁADNIA
       zabbix_proxy [-c plik-konfiguracyjny]
       zabbix_proxy [-c plik-konfiguracyjny] -R opcja-uruchomieniowa
       zabbix_proxy [-c plik-konfiguracyjny] -T
       zabbix_proxy -h
       zabbix_proxy -V

OPIS
       zabbix_proxy to demon, który zbiera dane monitorowania z urządzeń i
       wysyła je do serwer Zabbix.

OPCJE
       -c, --config plik-konfiguracyjny
              Użyj alternatywnego plik-konfiguracyjny zamiast domyślnego.

       -f, --foreground
              Uruchom proxy Zabbix na pierwszym planie.

       -R, --runtime-control opcja-uruchomieniowa
              Wykonaj funkcje administracyjne zgodnie z opcja-uruchomieniowa.

           Opcje sterowania w czasie działania
               config_cache_reload
                   Przeładuj pamięć podręczną konfiguracji. Ignorowane, jeśli
                   pamięć podręczna jest obecnie ładowana. Aktywny proxy
                   Zabbix połączy się z serwer Zabbix i zażąda danych
                   konfiguracyjnych. Zostanie użyty domyślny plik
                   konfiguracyjny (chyba że podano opcję -c) do odnalezienia
                   pliku PID, a sygnał zostanie wysłany do procesu
                   wskazanego w pliku PID.

               snmp_cache_reload
                   Przeładuj pamięć podręczną SNMP.

               housekeeper_execute
                   Uruchom housekeeper. Ignorowane, jeśli housekeeper jest
                   obecnie wykonywany.

               diaginfo[=section]
                   Zapisz wewnętrzne informacje diagnostyczne dla określonej
                   section. Section może mieć wartość historycache,
                   preprocessing, locks. Domyślnie zapisywane są informacje
                   diagnostyczne ze wszystkich sekcji.

               prof_enable[=target]
                   Włącz profilowanie, wpływa na wszystkie procesy, jeśli nie
                   określono target

               prof_disable[=target]
                   Wyłącz profilowanie, wpływa na wszystkie procesy, jeśli nie
                   określono target

               log_level_increase[=target]
                   Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli
```

```
                nie określono target.

            log_level_decrease[=target]
                Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli
                nie określono target.

        Cele sterowania poziomem logowania
            process-type
                Wszystkie procesy określonego typu (availability manager,
                browser poller, configuration syncer, data sender,
                discovery manager, discovery worker, history syncer,
                housekeeper, http poller, icmp pinger, internal poller,
                ipmi manager, ipmi poller, java poller, odbc poller,
                poller, agent poller, http agent poller, snmp poller,
                preprocessing manager, preprocessing worker,
                self-monitoring, snmp trapper, task manager, trapper,
                unreachable poller, vmware collector)

            process-type,N
                Typ procesu i numer (np. poller,3)

            pid Identyfikator procesu

        Cele sterowania profilowaniem
            process-type
                Wszystkie procesy określonego typu (availability manager,
                browser poller, configuration syncer, data sender,
                discovery manager, discovery worker, history syncer,
                housekeeper, http poller, icmp pinger, internal poller,
                ipmi manager, ipmi poller, java poller, odbc poller,
                poller, agent poller, http agent poller, snmp poller,
                preprocessing manager, preprocessing worker,
                self-monitoring, snmp trapper, task manager, trapper,
                unreachable poller, vmware collector)

            process-type,N
                Typ procesu i numer (np. history syncer,1)

            pid Identyfikator procesu

            scope
                (rwlock, mutex, processing) może być użyte z process-type
                (np. history syncer,1,processing)

    -T, --test-config
           Sprawdź poprawność pliku konfiguracyjnego i zakończ.

    -h, --help
           Wyświetl tę pomoc i zakończ.

    -V, --version
           Wyświetl informacje o wersji i zakończ.

PLIKI
    /usr/local/etc/zabbix_proxy.conf
           Domyślna lokalizacja pliku konfiguracyjnego proxy Zabbix (jeśli
           nie została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
    Dokumentacja

    zabbix_agentd(8), zabbix_get(1), zabbix_sender(1), zabbix_server(8),
    zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)
```

zabbix_sender

ZABBIX_SENDER(1) Podręcznik poleceń ogólnych ZABBIX_SENDER(1)

NAZWA

zabbix_sender - narzędzie Zabbix sender

SKŁADNIA

```
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] -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-iden-
tity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-ci-
pher 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-iden-
tity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-ci-
pher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender -h
zabbix_sender -V
```

OPIS

zabbix_sender to narzędzie wiersza poleceń służące do wysyłania danych monitorujących do serwera Zabbix lub proxy. Na serwerze Zabbix należy utworzyć pozycję typu Zabbix trapper z odpowiednim kluczem. Należy pamiętać, że wartości przychodzące będą akceptowane tylko od hostów określonych w polu Allowed hosts dla tej pozycji.

OPCJE

-c, --config config-file
: Użyj pliku config-file. Zabbix sender odczytuje szczegóły serwera z pliku konfiguracyjnego agentd. Domyślnie Zabbix sender nie odczytuje żadnego pliku konfiguracyjnego. Obsługiwane są tylko parametry Hostname, ServerActive, SourceIP, TLSConnect, TLSCAFile, TLSCRLFile, TLSServerCertIssuer, TLSServerCertSubject, TLSCertFile, TLSKeyFile, TLSPSKIdentity i TLSPSKFile. Nazwa hosta zdefiniowana przez parametr HostnameItem nie zostanie uwzględniona; w takim przypadku nazwę hosta należy podać w wierszu poleceń (zobacz opcję -s). Wszystkie adresy zdefiniowane w parametrze konfiguracyjnym ServerActive agenta są używane do wysyłania danych. Jeśli wysyłanie danych wsadowych do jednego adresu się nie powiedzie, kolejne partie nie są wysyłane do tego adresu.

-z, --zabbix-server server
: Nazwa hosta lub adres IP serwera Zabbix. Jeśli host jest monitorowany przez proxy, należy zamiast tego użyć nazwy hosta lub adresu IP proxy. W połączeniu z --config nadpisuje wpisy parametru ServerActive określone w pliku konfiguracyjnym agentd.

-p, --port port
: Określ numer portu trappera serwera Zabbix działającego na serwerze. Domyślnie: 10051. W połączeniu z --config nadpisuje wpisy portu parametru ServerActive określone w pliku konfiguracyjnym agentd.

-I, --source-address IP-address
: Określ źródłowy adres IP. W połączeniu z --config nadpisuje parametr SourceIP określony w pliku konfiguracyjnym agentd.

-t, --timeout seconds
: Określ limit czasu. Prawidłowy zakres: 1-300 sekund (domyślnie: 60)

-s, --host host
: Określ nazwę hosta, do którego należy pozycja (zarejestrowaną w frontend Zabbix). Adres IP hosta i nazwa DNS nie będą działać. W połączeniu z --config nadpisuje parametr Hostname określony w pliku konfiguracyjnym agentd.

-k, --key key
: Określ klucz pozycji, do której ma zostać wysłana wartość.

-o, --value value
       Określ wartość pozycji.

-i, --input-file input-file
       Wczytaj wartości z pliku wejściowego. Aby odczytać wartości ze
       standardowego wejścia, jako <input-file> podaj -. Każda linia
       pliku zawiera rozdzielone białymi znakami: <hostname> <key>
       <value>. Każda wartość musi być podana w osobnym wierszu. Każdy
       wiersz musi zawierać 3 wpisy rozdzielone białymi znakami:
       <hostname> <key> <value>, gdzie "hostname" to nazwa monitorowanego
       hosta zarejestrowana w frontend Zabbix, "key" to docelowy klucz
       pozycji, a "value" - wartość do wysłania. Aby użyć nazwy hosta z
       pliku konfiguracyjnego agenta lub z argumentu --host, podaj - jako
       <hostname>.

       Przykład wiersza pliku wejściowego:

        "Linux DB3" db.connections 43

       Typ wartości musi być poprawnie ustawiony w konfiguracji pozycji
       w frontend Zabbix. Zabbix sender wyśle maksymalnie 250 wartości w
       jednym połączeniu. Zawartość pliku wejściowego musi być zakodowana
       w UTF-8. Wszystkie wartości z pliku wejściowego są wysyłane
       sekwencyjnie, od góry do dołu. Wpisy muszą być sformatowane zgodnie
       z następującymi zasadami:

              • Obsługiwane są wpisy w cudzysłowie i bez cudzysłowu.

              • Znakiem cytowania jest podwójny cudzysłów.

              • Wpisy zawierające białe znaki muszą być ujęte w cudzysłów.

              • Wewnątrz wpisu w cudzysłowie znak podwójnego cudzysłowu i
                ukośnika odwrotnego musi być poprzedzony ukośnikiem
                odwrotnym.

              • Ucieczka znaków nie jest obsługiwana we wpisach bez
                cudzysłowu.

              • Sekwencje ucieczki znaku nowej linii (\n) są obsługiwane
                w ciągach w cudzysłowie.

              • Sekwencje ucieczki znaku nowej linii są usuwane z końca
                wpisu.

-T, --with-timestamps
       Tej opcji można używać tylko z opcją --input-file.

       Każdy wiersz pliku wejściowego musi zawierać 4 wpisy rozdzielone
       białymi znakami: <hostname> <key> <timestamp> <value>. Sygnatura
       czasowa powinna być podana w formacie Unix timestamp. Jeśli
       docelowa pozycja ma wyzwalacze, które się do niej odwołują, wszystkie
       sygnatury czasowe muszą być w kolejności rosnącej, w przeciwnym
       razie obliczanie zdarzeń nie będzie poprawne.

       Przykład wiersza pliku wejściowego:

        "Linux DB3" db.connections 1429533600 43

       Więcej informacji można znaleźć w opisie opcji --input-file.

       Jeśli wartość ze znacznikiem czasu zostanie wysłana dla hosta,

który znajduje się w typie konserwacji „no data", wartość ta zostanie odrzucona; jednak można wysłać wartość ze znacznikiem czasu dla wygasłego okresu konserwacji i zostanie ona zaakceptowana.

-N, --with-ns
Tej opcji można używać tylko z opcją --with-timestamps.

Każdy wiersz pliku wejściowego musi zawierać 5 wpisów rozdzielonych białymi znakami: <hostname> <key> <timestamp> <ns> <value>.

Przykład wiersza pliku wejściowego:

"Linux DB3" db.connections 1429533600 7402561 43

Więcej informacji można znaleźć w opisie opcji --input-file.

-r, --real-time
Wysyłaj wartości pojedynczo, zaraz po ich odebraniu. Można tego używać podczas odczytu ze standardowego wejścia.

--tls-connect value
Sposób połączenia z serwerem lub proxy. Wartości:

unencrypted
połączenie bez szyfrowania (domyślnie)

psk
połączenie z użyciem TLS i klucza współdzielonego

cert
połączenie z użyciem TLS i certyfikatu

--tls-ca-file CA-file
Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu CA do weryfikacji certyfikatu peer.

--tls-crl-file CRL-file
Pełna ścieżka do pliku zawierającego unieważnione certyfikaty.

--tls-server-cert-issuer cert-issuer
Dozwolony wystawca certyfikatu serwera.

--tls-server-cert-subject cert-subject
Dozwolony podmiot certyfikatu serwera.

--tls-cert-file cert-file
Pełna ścieżka do pliku zawierającego certyfikat lub łańcuch certyfikatów.

--tls-key-file key-file
Pełna ścieżka do pliku zawierającego klucz prywatny.

--tls-psk-identity PSK-identity
Ciąg identyfikatora PSK.

--tls-psk-file PSK-file
Pełna ścieżka do pliku zawierającego klucz współdzielony.

--tls-cipher13 cipher-string
Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3. Nadpisuje domyślne kryteria wyboru zestawu szyfrów. Opcja ta nie jest dostępna, jeśli wersja OpenSSL jest starsza niż 1.1.1.

--tls-cipher cipher-string

Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfrów OpenSSL (tylko dla TLS 1.2). Nadpisuje domyślne kryteria wyboru zestawu szyfrów.

-v, --verbose
Tryb szczegółowy, -vv dla większej ilości informacji.

-h, --help
Wyświetl tę pomoc i zakończ.

-V, --version
Wyświetl informacje o wersji i zakończ.

STATUS ZAKOŃCZENIA

Status zakończenia wynosi 0, jeśli wartości zostały wysłane i wszystkie zostały pomyślnie przetworzone przez serwer. Jeśli dane zostały wysłane, ale przetwarzanie co najmniej jednej z wartości nie powiodło się, status zakończenia wynosi 2. Jeśli wysyłanie danych nie powiodło się, status zakończenia wynosi 1.

PRZYKŁADY

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -k mysql.queries -o 342.45
Wyślij 342.45 jako wartość pozycji mysql.queries monitorowanego hosta. Użyj monitorowanego hosta i serwera Zabbix zdefiniowanych w pliku konfiguracyjnym agenta.

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -s "Monitored Host" -k mysql.queries -o 342.45
Wyślij 342.45 jako wartość pozycji mysql.queries hosta Monitored Host, używając serwera Zabbix zdefiniowanego w pliku konfiguracyjnym agenta.

zabbix_sender -z 192.168.1.113 -i data_values.txt
Wyślij wartości z pliku data_values.txt do serwera Zabbix o adresie IP 192.168.1.113. Nazwy hostów i klucze są zdefiniowane w pliku.

echo "- hw.serial.number 1287872261 SQ4321ASDF" | zabbix_sender -c /usr/local/etc/zabbix_agentd.conf -T -i -
Wyślij wartość ze znacznikiem czasu z wiersza poleceń do serwera Zabbix określonego w pliku konfiguracyjnym agenta. Myślnik w danych wejściowych oznacza, że nazwa hosta również powinna zostać pobrana z tego samego pliku konfiguracyjnego.

echo '"Zabbix server" trapper.item ""' | zabbix_sender -z 192.168.1.113 -p 10000 -i -
Wyślij pustą wartość pozycji do serwera Zabbix o adresie IP 192.168.1.113 na porcie 10000 z wiersza poleceń. Puste wartości muszą być oznaczone pustymi podwójnymi cudzysłowami.

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
Wyślij 342.45 jako wartość pozycji mysql.queries w hoście Monitored Host do serwera o adresie IP 192.168.1.113, używając TLS z certyfikatem.

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
Wyślij 342.45 jako wartość pozycji mysql.queries w hoście

```
              Monitored Host do serwera o adresie IP 192.168.1.113, używając
              TLS z kluczem współdzielonym (PSK).

ZOBACZ TAKŻE
       Dokumentacja

       zabbix_agentd(8),  zabbix_get(1),  zabbix_proxy(8),   zabbix_server(8),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                  ZABBIX_SENDER(1)
```

zabbix_server

```
ZABBIX_SERVER(8)            Podręcznik administratora systemu           ZABBIX_SERVER(8)

NAZWA
       zabbix_server - demon serwera Zabbix

SKŁADNIA
       zabbix_server [-c plik-konfiguracyjny]
       zabbix_server [-c plik-konfiguracyjny] -R opcja-czasu-uruchomienia
       zabbix_server [-c plik-konfiguracyjny] -T
       zabbix_server -h
       zabbix_server -V

OPIS
       zabbix_server jest głównym demonem oprogramowania Zabbix.

OPCJE
       -c, --config plik-konfiguracyjny
              Użyj alternatywnego pliku konfiguracyjnego zamiast domyślnego.

       -f, --foreground
              Uruchom serwer Zabbix na pierwszym planie.

       -R, --runtime-control opcja-czasu-uruchomienia
              Wykonaj funkcje administracyjne zgodnie z opcją-czasu-uruchomienia.

           Opcje sterowania w czasie uruchomienia
               config_cache_reload
                   Przeładuj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecni

               snmp_cache_reload
                   Przeładuj pamięć podręczną SNMP.

               housekeeper_execute
                   Uruchom housekeeper. Ignorowane, jeśli housekeeper jest obecnie wykonywany.

               trigger_housekeeper_execute
                   Uruchom trigger housekeeper (usuń problemy dla usuniętych wyzwalaczy). Ignorowane, jeśl

               secrets_reload
                   Przeładuj sekrety z Vault. Ignorowane, jeśli pamięć podręczna konfiguracji lub sekrety

               diaginfo[=section]
                   Zapisz wewnętrzne informacje diagnostyczne dla określonej sekcji. Sekcja może być histo

               prof_enable[=target]
                   Włącz profilowanie, wpływa na wszystkie procesy, jeśli target nie jest określony

               prof_disable[=target]
```

Wyłącz profilowanie, wpływa na wszystkie procesy, jeśli target nie jest określony

ha_status
Wyświetl status klastra wysokiej dostępności. Można wykonać tylko na aktywnym węźle.

ha_remove_node
ha_remove_node[=name|ID] Usuń węzeł wysokiej dostępności według jego nazwy lub ID wymie

ha_set_failover_delay[=delay]
Ustaw opóźnienie przełączenia awaryjnego wysokiej dostępności. Opóźnienie można określi

log_level_increase[=target]
Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony

log_level_decrease[=target]
Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony

Cele sterowania poziomem logowania
process-type
Wszystkie procesy określonego typu (alerter, alert manager, alert syncer, availability manager, browser poller, configuration syncer, configuration syncer worker, connector manager, connector worker, discovery manager, discovery worker, escalator, ha manager, history poller, history syncer, housekeeper, http poller, http agent poller, icmp pinger, internal poller, ipmi manager, ipmi poller, java poller, lld manager, lld worker, odbc poller, poller, agent poller, preprocessing manager, preprocessing worker, proxy poller, proxy group manager, report manager, report writer, self-monitoring, service manager, snmp poller, snmp trapper, task manager, timer, trapper, trigger housekeeper, unreachable poller, vmware collector)

process-type,N
Typ procesu i numer (np. poller,3)
pid Identyfikator procesu

Cele sterowania profilowaniem
process-type
Wszystkie procesy określonego typu (alerter, alert manager, alert syncer, availability manager, browser poller, configuration syncer, configuration syncer worker, connector manager, connector worker, discovery manager, discovery worker, escalator, ha manager, history poller, history syncer, housekeeper, http poller, http agent poller, icmp pinger, internal poller, ipmi manager, ipmi poller, java poller, lld manager, lld worker, odbc poller, poller, agent poller, preprocessing manager, preprocessing worker, proxy poller, proxy group manager, report manager, report writer, self-monitoring, service manager, snmp poller, snmp trapper, task manager, timer, trapper, trigger housekeeper, unreachable poller, vmware collector)

process-type,N
Typ procesu i numer (np. history syncer,1)

pid Identyfikator procesu

scope
(rwlock, mutex, processing) może być używane z process-type (np. history syncer,1,processing)

-T, --test-config

```
              Sprawdź poprawność pliku konfiguracyjnego i zakończ.

       -h, --help
              Wyświetl tę pomoc i zakończ.

       -V, --version
              Wyświetl informacje o wersji i zakończ.

PLIKI
       /usr/local/etc/zabbix_server.conf
              Domyślna lokalizacja pliku konfiguracyjnego serwera Zabbix (jeśli
              nie została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       zabbix_agentd(8),  zabbix_get(1),  zabbix_proxy(8),   zabbix_sender(1),
       zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Zabbix                            2022-01-06                  ZABBIX_SERVER(8)
```

zabbix_web_service

```
ZABBIX_WEB_SERVICE(8)       Podręcznik administratora systemu      ZABBIX_WEB_SERVICE(8)

NAZWA
       zabbix_web_service - usługa web Zabbixa

SKŁADNIA
       zabbix_web_service [-c plik-konfiguracyjny]
       zabbix_web_service [-c plik-konfiguracyjny] -T
       zabbix_web_service -h
       zabbix_web_service -V

OPIS
       zabbix_web_service to aplikacja służąca do udostępniania usług web dla
       komponentów Zabbixa.

OPCJE
       -c, --config plik-konfiguracyjny
              Użyj alternatywnego pliku konfiguracyjnego zamiast domyślnego.

       -T, --test-config
              Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.

       -h, --help
              Wyświetl tę pomoc i zakończ działanie.

       -V, --version
              Wyświetl informacje o wersji i zakończ działanie.

PLIKI
       /usr/local/etc/zabbix_web_service.conf
              Domyślna lokalizacja pliku konfiguracyjnego usługi web Zabbixa
              (jeśli nie została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       zabbix_agentd(8),  zabbix_get(1),  zabbix_proxy(8),   zabbix_sender(1),
       zabbix_server(8), zabbix_js(1), zabbix_agent2(8)
```

Informacja o prawach autorskich

Dokumentacja Zabbix NIE jest rozpowszechniana na licencji AGPL-3.0. Korzystanie z dokumentacji Zabbix podlega następującym warunkom:

Możesz utworzyć drukowaną kopię tej dokumentacji wyłącznie do własnego użytku osobistego. Konwersja do innych formatów jest dozwolona, o ile rzeczywista treść nie zostanie w żaden sposób zmieniona ani zredagowana. Nie wolno publikować ani rozpowszechniać tej dokumentacji w żadnej formie ani na żadnym nośniku, chyba że rozpowszechniasz dokumentację w sposób podobny do tego, w jaki rozpowszechnia ją Zabbix (to znaczy elektronicznie do pobrania na stronie internetowej Zabbix) albo na nośniku USB lub podobnym, pod warunkiem jednak, że dokumentacja jest rozpowszechniana razem z oprogramowaniem na tym samym nośniku. Jakiekolwiek inne wykorzystanie, takie jak jakiekolwiek rozpowszechnianie drukowanych kopii lub wykorzystanie tej dokumentacji, w całości lub w części, w innej publikacji, wymaga uprzedniej pisemnej zgody upoważnionego przedstawiciela Zabbix. Zabbix zastrzega sobie wszelkie prawa do tej dokumentacji, które nie zostały wyraźnie przyznane powyżej.