

Zabbix 最新バージョン 7.4の機能 と Zabbixユーザー会イベントの案内

2025/10/17

日本Zabbixユーザー会
Japanese Zabbix Community

自己紹介

- わたなべ はやと



BSmile

- 日本Zabbixユーザー会
 - コミュニティスタッフ
- 初登壇場所
 - 2014/4/13 第6回 ZABBIX-JP勉強会
 - ※ 非スタッフ時にLT参加



アジェンダ

- Zabbixとは
- Zabbix 7.4のインストール方法
- Zabbix 7.4の機能
- 開催される日本Zabbixユーザー会イベント

Zabbixとは

Zabbixとは

- Zabbixはエンタープライズクラスの分散監視ソリューション
 - オープンソースで配布されている
- データ収集
 - ネットワークの多数のパラメーター
 - サーバーや仮想マシン
 - アプリケーション
 - サービス
 - データベース
 - Webサイト
 - クラウド
- 通知
 - メール
 - さまざまなチャットツール (Slack , MS Teamsなど)

Zabbixとは

- データ収集方法
 - ポーリングとトラッピングの両方をサポート
- Webインターフェースからアクセス
 - すべてのZabbixのレポート
 - 統計情報
 - ホストなどの設定
- 監視規模
 - 少数のサーバーを有する小規模な組織
 - 大量のサーバーを有する大企業
- 使用料
 - 無料。ZabbixはAGPL-3.0ライセンスの下で作成および配布

Zabbix 7.4のインストール

RHEL系

- RDBMSインストール

```
# dnf install mysql-server  
# systemctl enable mysqld --now
```

- Zabbixリポジトリインストール

```
# rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/alma/9/noarch/zabbix-release-latest-7.4.el9.noarch.rpm  
# dnf clean all
```

- Zabbixインストール

```
# dnf install zabbix-server-mysql zabbix-web-mysql zabbix-apache-conf zabbix-sql-scripts  
zabbix-selinux-policy zabbix-agent zabbix-web-japanese
```

RHEL系

- RDBMS設定(MySQL)

```
# 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;

# zcat /usr/share/zabbix/sql-scripts/mysql/server.sql.gz | mysql --default-character-
set=utf8mb4 -uzabbix -p zabbix

# mysql -uroot -p
password
mysql> set global log_bin_trust_function_creators = 0;
mysql> quit;
```

RHEL系

- Zabbixサーバー設定ファイル編集

```
# vi /etc/zabbix/zabbix_server.conf
```

```
DBPassword=password
```

- Zabbix起動

```
# systemctl enable zabbix-server zabbix-agent httpd php-fpm --now
```

- 注意点: EPELリポジトリのZabbixは、公式サポートを受けられません

```
[epel]
```

```
...
```

```
excludepkgs=zabbix*
```

Webインタフェース設定

- `http(s)://IPアドレス/zabbix/`

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

ようこそ

Zabbix 7.4

デフォルトの言語 日本語 (ja_JP)

戻る

次のステップ

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

前提条件のチェック

	現在の値	要求文字列	
PHPバージョン	8.0.30	8.0.0	OK
PHPの"memory_limit"オプション	128M	128M	OK
PHPの"post_max_size"オプション	16M	16M	OK
PHPの"upload_max_filesize"オプション	2M	2M	OK
PHPの"max_execution_time"オプション	300	300	OK
PHPの"max_input_time"オプション	300	300	OK
PHPデータベースサポート	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK
PHPの"mbstring.func_overload"オプション	off	off	OK

戻る

次のステップ

Webインタフェース設定

- `http(s)://IPアドレス/zabbix/`

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

データベース接続設定

データベースを手動で作成してください。そしてデータベースに接続するための設定値を設定してください。終了したら"次のステップ"ボタンを押してください。

データベースタイプ

MySQL

データベースホスト

localhost

データベースポート

0

0 - デフォルトのポート番号を使用

データベース名

zabbix

資格情報保存先

プレーンテキスト

HashiCorp Vault

CyberArk Vault

ユーザー

zabbix

パスワード

データベース接続のTLS暗号化

ソケットファイル(UNIX系OS上)や共有メモリ(Windows上)を使用するため接続は暗号化されません。

戻る

次のステップ

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

設定

Zabbixサーバー名

osc2025

デフォルトのタイムゾーン

(UTC+09:00) Asia/Tokyo

デフォルトのテーマ

Blue

Webインターフェースからの接続を暗号化する。

☐

戻る

次のステップ

Webインタフェース設定

- `http(s)://IPアドレス/zabbix/`

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

設定パラメータの確認

設定値を確認してください。正しければ"次のステップ"ボタンを押してください。もしくは、"戻る"ボタンを押して設定値を修正してください。

データベースタイプ	MySQL
データベースサーバー	localhost
データベースポート	デフォルト
データベース名	zabbix
データベースユーザー	zabbix
データベースパスワード	*****
データベース接続のTLS暗号化	false
Zabbixサーバー名	osc2025
Webインターフェースからの接続を暗号化する。	false

戻る

次のステップ

ZABBIX

ようこそ

前提条件のチェック

データベース接続設定

設定

設定パラメータの確認

インストール

インストール

おめでとうございます！Webインターフェースのインストールが終了しました。

設定ファイル"etc/zabbix/web/zabbix.conf.php"を作成しました。

戻る

終了

Zabbix 7.4の機能

機能サマリーは [zabbix.com](https://www.zabbix.com)を参照

What's new in Zabbix 7.4

強化されたリソース検出ワークフロー、進化したデータ可視化オプション、
れたユーザーエクスペリエンスなど、多くの新機能を追加。

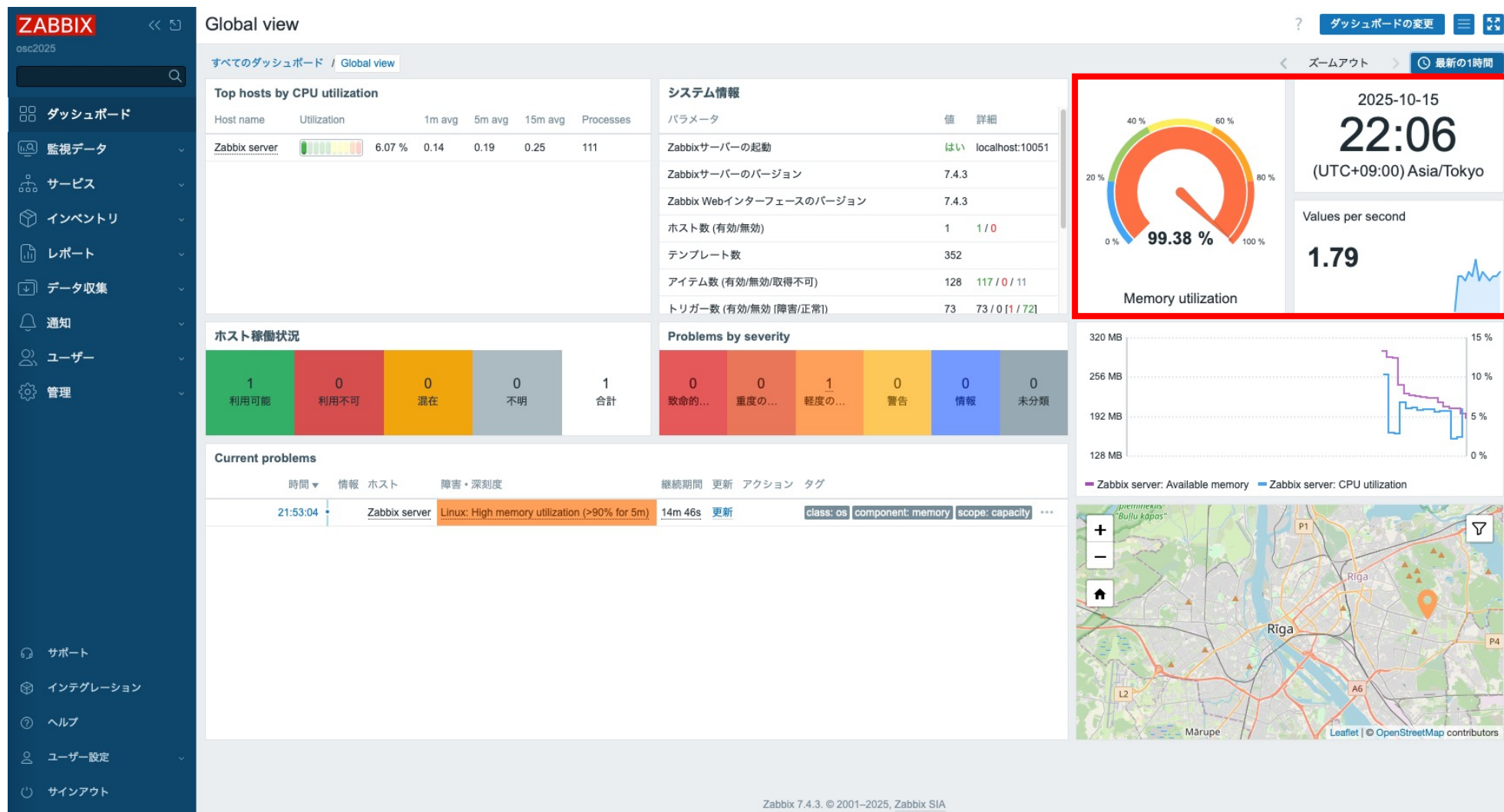
ダウンロード

Webセミナー参加

リリースノート

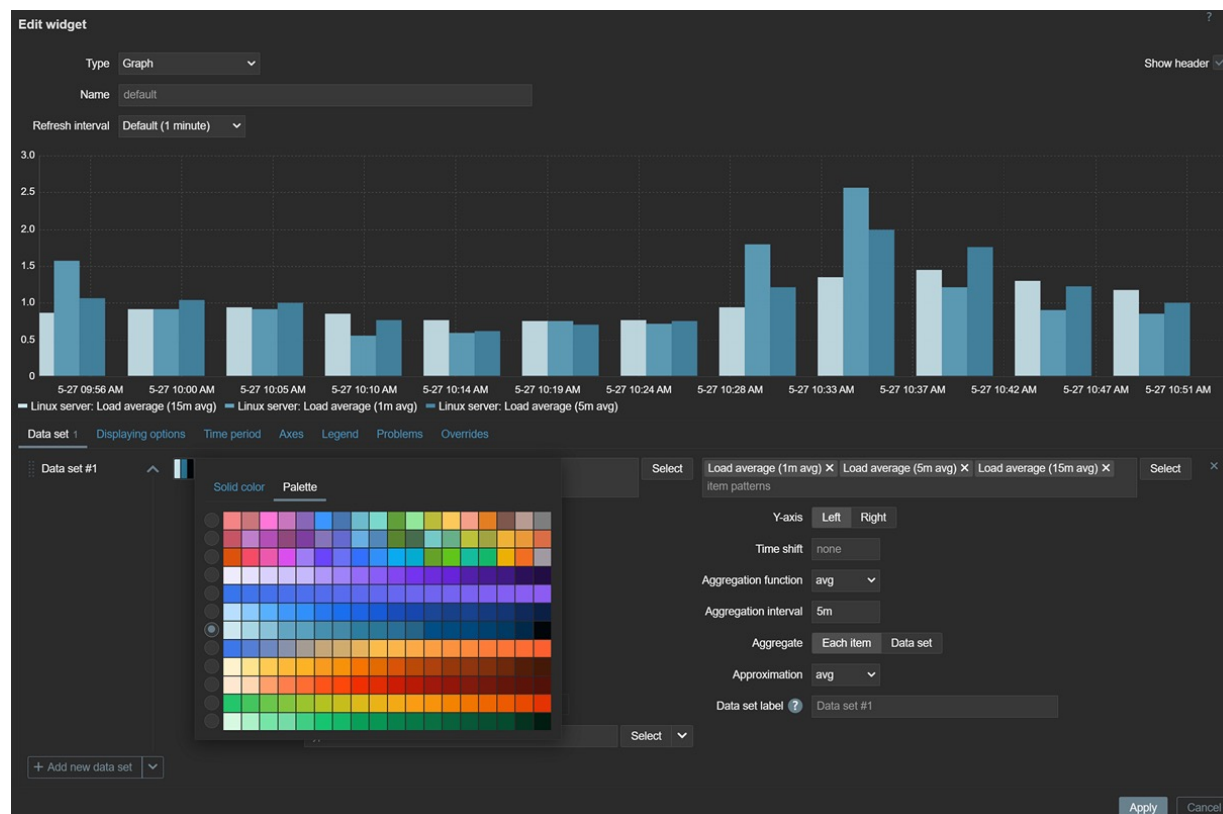
https://www.zabbix.com/jp/whats_new_7_4

デフォルトダッシュボード更新



ダッシュボードの「グラフ」の表示色

- これまでのグラフ色調
 - 単色
 - 16進数入力
- 7.4以降
 - 「パレット」タブ追加
 - 定義済みの一覧から選択



ホストウィザード機能

- ホストウィザード
 - Zabbix で監視対象 (デバイス、アプリケーション、サービスなど) の設定ウィザード。
- 従来の「ホストの作成」も機能は変わらず利用可

ホスト

?

ホストウィザード

ホストの作成

インポート

🔍 フィルター

☐	名前 ▲	アイテム	トリガー	グラフ	ディスカバリ	Web	インターフェース	プロキシ	テンプレート	ステータス	エージェントの状態	エージェント暗号化	情報	タグ
☐	*** Zabbix server	アイテム 128	トリガー 73	グラフ 10	ディスカバリ 6	Web	127.0.0.1:10050		Linux by Zabbix agent, Zabbix server health	有効	ZBX	なし		

1件のうち1件を表示しています

0 選択

有効

無効

エクスポート

▼

一括更新

削除

ホストウィザード機能

- テンプレートを選択
 - 互換性が必要
 - エージェント監視
 - エージェントレス監視

テンプレートを選擇:
Linux by Zabbix agent

ホストの作成または選擇

あと少しの手順

一部のテンプレート (269) はホスト ウィザードと互換性がありません。こちらを確認してください。 [更新方法](#)。カスタム テンプレートはサポートさ

テンプレートを選擇

テンプレートとは、監視対象に合わせて設計された、事前定義された構成 (収集するメトリック、アラートを生成する条件など) のセットです。

linux

テンプレートを検索するためにキーワードを入力してください。

データ収集 [?]

すべて エージェントを利用 エージェントレス

 エージェントモード [?]

すべて アクティブ パッシブ

テンプレート

Os (2)

Linux by Zabbix agent ☒

タグ
class: os target: linux

▼ 表示を増やす。

Linux by Zabbix agent active ☐

タグ
class: os target: linux

▼ 表示を増やす。

[キャンセル](#)

次

ホストウィザード機能

- ホストの作成と選択
 - ホスト名
 - ホストグループ

テンプレートを選択:
Linux by Zabbix agent

ホストの作成または選択:
Ubuntu (新規)

あと少しの手順

ホストの作成または選択

選択したテンプレート (Linux by Zabbix agent) は、監視対象を表す Zabbix 内のエンティティであるホストにリンクされている必要があります。

ホストは、管理とアクセス制御を容易にするためにホスト グループに編成されます。

* ホスト名

Ubuntu (新規) ×

選択

入力を開始するか、“選択” をクリックして既存のホストを選択するか、新しいホスト名を入力します。

* ホストグループ

Linux servers ×

選択

検索文字列を入力

入力を開始するか、選択 をクリックして既存のホスト グループを選択するか、新しいホスト グループ名を入力します。

キャンセル

戻る

次

ホストウィザード機能

- エージェント設定
 - サーバーIP
 - 暗号化設定
 - 監視対象のOS
 - インストール

テンプレートを選択:
Linux by Zabbix agent

ホストの作成または選択:
Ubuntu (新規)

Zabbixエージェントをインストール

ホストインターフェースを追加

ホストの設定

Zabbixエージェントをインストール

選択したテンプレート (Linux by Zabbix agent) では、監視対象にZabbixエージェントがインストールされ、実行されている必要があります。

Zabbixエージェントがすでにインストールされている場合は、OSの選択をスキップします。

1. Zabbix サーバー、プロキシ、またはクラスターのアドレスを確認する。

例:
192.0.2.0:10051, [2001:db8::]:10051, zbx1.local:10051;zbx2.local:10051

3. 監視対象のOSを選択してください

Linux☒

Windows☐

その他☐

4. 次のスクリプト[root下のbash]を実行して、監視対象にZabbixエージェントを設定します。

```
$(command -v curl || echo $(command -v wget) -O -)  
https://cdn.zabbix.com/scripts/7.4/install-zabbix.sh | bash -s -  
--server-host '192.168.11.188' --hostname 'Ubuntu' --psk-  
identity 'Ubuntu PSK' --psk  
15645aa538df5799ef6975c7a89e68e0cb8d84ebf37c5f5e6849255b4231519a
```

[キャンセル](#)

戻る次

ホストウィザード機能

- エージェントのインストール

3. 監視対象のOSを選択してください

Linux ☒	Windows ☐	その他 ☐
--	-------------------------------	---------------------------

4. 次のスクリプト[root下のbash]を実行して、監視対象にZabbixエージェントを設定します。

```
$(command -v curl || echo $(command -v wget) -O -)
https://cdn.zabbix.com/scripts/7.4/install-zabbix.sh | bash -s -- --
server-host '192.168.11.188' --hostname 'Ubuntu' --psk-identity
'Ubuntu PSK' --psk
15645aa538df5799ef6975c7a89e68e0cb8d84ebf37c5f5e6849255b4231519a
--2025-10-15 13:45:37--
https://cdn.zabbix.com/scripts/7.4/install-zabbix.sh
...
```

```
$(command -v curl || echo $(command -v wget) -O -)
https://cdn.zabbix.com/scripts/7.4/install-zabbix.sh | bash -s -- --
server-host '192.168.11.188' --hostname 'Ubuntu' --psk-identity
'Ubuntu PSK' --psk
15645aa538df5799ef6975c7a89e68e0cb8d84ebf37c5f5e6849255b4
231519a
--2025-10-15 13:45:37--
https://cdn.zabbix.com/scripts/7.4/install-zabbix.sh
...
```

おまけ

- インストールスクリプトは、個別に使用も可能

```
#!/bin/bash
function show_help
{
    echo "Usage:"
    echo "  $(basename ${BASH_SOURCE[0]}) (--install|--reinstall) [--agent|--agent2] [--version X.Y] [<configuration options>]"
    echo "  $(basename ${BASH_SOURCE[0]}) --configure [--agent|--agent2] [<configuration options>]"
    echo "  $(basename ${BASH_SOURCE[0]}) --uninstall [...]"
    echo ""
    echo "Options:"
    echo "  One of four modes can be selected --install, --reinstall, --configure, --uninstall"
    echo "    --install          : Install Zabbix agent or agent 2 and write values to its configuration file."
    echo "    --install mode includes --configure mode."
    echo "    --uninstall       : Purge Zabbix from the system."
    echo "    --configure       : Write values to agent or agent 2 configuration files."
    echo "    --reinstall       : Combines --uninstall and --install options."
    echo "  Default mode is --install"
    echo ""
    echo "    --agent | --agent2 : Select which Zabbix component to install."
    echo "    --version X.Y      : Select which Zabbix version to install. Default $default_zabbix_version"
    echo ""
    echo "  <configuration options>:"
    echo "    --server-host <str> : Value for Server and ServerActive fields."
    echo "    --server-host-stdin : Same as --server-host, but prompt for the key using stdin."
    echo "    --psk <key>         : Enable PSK encryption and setup the key."
    echo "    --psk-stdin         : Same as --psk, but prompt for the key using stdin."
    echo "    --psk-identity <str> : PSK identity to go with the PSK key."
    echo "    --psk-identity-stdin : Same as --psk-identity, but prompt for the key using stdin."
    echo "    --hostname <str>    : Set Hostname value."
    echo ""
    echo "    --repo-url <URL>    : Override default repo url. Default: $default_repo_url"
}
```

ZABBIX Zabbix Cloud Images and Appliances

Zabbix is an enterprise-class open source distributed monitoring solution designed to monitor and track performance and availability. It supports distributed and WEB monitoring, auto-discovery, and more.

Zabbix appliances are available on Google Cloud Platform along with previously released Zabbix appliances, Microsoft Azure, and AWS on all major cloud platforms.

These appliances are created and officially supported by Zabbix SIA.

Installation instructions are available in [Zabbix Cloud Images](#) page.

If you have any problems or suggestions, please report an issue on [Zabbix Bug Tracking System](#).

If you want to get professional support, installation or upgrade service, please see our [Zabbix technical support service](#) page.

Index of /scripts/7.4/

[../](#)
[install-zabbix.ps1](#)
[install-zabbix.sh](#)

03-Jul-2025 06:54
24-Sep-2025 13:49

6066
15592

<https://cdn.zabbix.com/scripts/7.4/>

ホストウィザード機能

- ホストインタフェース追加
 - Zabbixサーバー側の設定

テンプレートを選択:
Linux by Zabbix agent

ホストの作成または選択:
Ubuntu (新規)

Zabbixエージェン

ホストインターフェースを追加

選択したテンプレート (Linux by Zabbix agent) には、エージェントのインターフェースをホスト (Ubuntu) に追加する必要があります。

注意: 監視ターゲットで エージェント を構成して有効にする必要があります。

* エージェントのアドレス

192.168.11.191

* エージェントのポート

10050

監視対象にインストールされているZabbixエージェントのIP/DNSアドレスとポートを入力してください。

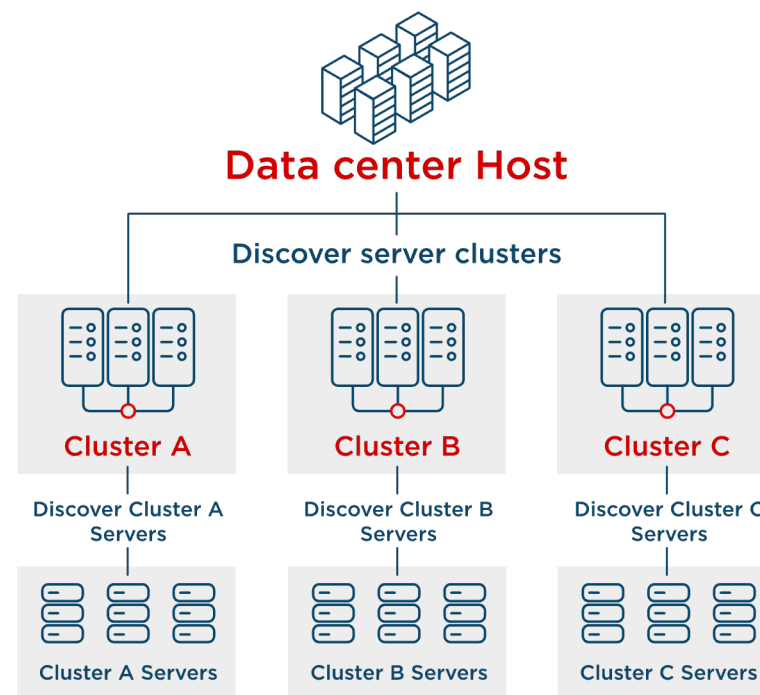
ホストウィザード機能

- ホストの監視が始まる
 - PSK暗号化はデフォルト
 - マクロ値の変更も可能

☐ ホスト	名前 ▲	最新のチェック時刻	最新の値	変化	タグ	
☐ Ubuntu	Available memory [?]	44s	957.46 MB		component: memory	グラフ
☐ Ubuntu	Available memory in % [?]	43s	93.502 %		component: memory	グラフ
☐ Ubuntu	Checksum of /etc/passwd	46s	212bf278d1a86a49...		component: security	ヒストリ
☐ Ubuntu	Context switches per second [?]	8s	2829.7651		component: cpu	グラフ
☐ Ubuntu	CPU guest nice time [?]	6s	0 %		component: cpu	グラフ
☐ Ubuntu	CPU guest time [?]	7s	0 %		component: cpu	グラフ
☐ Ubuntu	CPU idle time [?]	5s	98.1258 %	+0.4683 %	component: cpu	グラフ
☐ Ubuntu	CPU interrupt time [?]	4s	0 %		component: cpu	グラフ
☐ Ubuntu	CPU iowait time [?]	3s	0.5688 %	-0.1005 %	component: cpu	グラフ
☐ Ubuntu	CPU nice time [?]	2s	0 %		component: cpu	グラフ
☐ Ubuntu	CPU softirq time [?]	1s	0.01672 %	+0.01672 %	component: cpu	グラフ
☐ Ubuntu	CPU steal time [?]	50s	0 %		component: cpu	グラフ
☐ Ubuntu	CPU system time [?]	50s	1.227 %		component: cpu	グラフ
☐ Ubuntu	CPU user time [?]	50s	0.4462 %		component: cpu	グラフ
☐ Ubuntu	CPU utilization [?]	5s	1.8742 %	-0.4683 %	component: cpu	グラフ
☐ Ubuntu	Free swap space [?]	52s	7.83 GB		component: memory component: storage	グラフ
☐ Ubuntu	Free swap space in % [?]	51s	97.937 %		component: memory component: storage	グラフ

「ディスカバリのプロトタイプ」機能追加

- 従来の機能
 - データセンター
 - LLD 各種クラスタ: ホスト登録
- 7.4の機能
 - データセンター
 - LLD 各種クラスタ: ホスト登録
 - LLD(ディスカバリのプロトタイプ) → LLD
クラスタ配下サーバー:ホスト登録
 - 階層化は無制限
ディスカバリのプロトタイプ→ディスカバリのプロトタイプ ...



「ディスカバリのプロトタイプ」機能追加

- 設定例

- ディスカバリのプロトタイプで、取得した情報を使用し新しいディスカバリルールを作成

☐ ホスト	名前 ▲	アイテム	トリガー	グラフ	ホスト	ディスカバリルール	キー	監視間隔	タイプ	ステータス	情報
☐ discovery	Discover databases and tablespaces: db1	アイテムのプロトタイプ	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ	db.tablespace.discovery[d b1]		Nested	有効	
☐ discovery	Discover databases and tablespaces: db2	アイテムのプロトタイプ	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ	db.tablespace.discovery[d b2]		Nested	有効	
☐ discovery	Discover databases and tablespaces: db3	アイテムのプロトタイプ	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ	db.tablespace.discovery[d b3]		Nested	有効	
☐ discovery	Template [redacted] Discovery prototype: Discover databases and tablespaces	アイテムのプロトタイプ 1	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ 1	[redacted] b.discovery. rule	1h	スクリプト	有効	
☐ discovery	Template [redacted] Discovery prototype: [redacted] Discovery rule	アイテムのプロトタイプ	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ 1	[redacted] iscovery.rul e	1h	スクリプト	無効	

5件のうち5件を表示しています

「ディスカバリのプロトタイプ」機能追加

- 設定例
 - ディスカバリのプロトタイプ: ディスカバリルール

```
[  
  {  
    "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" }  
    ]  
  }, ...  
]
```

保存前処理	LLDマクロ 1	フィルター	オーバーライド
LLDマクロ	LLDマクロ	JSONPath	
	{#DB}	\$.database	

「ディスカバリのプロトタイプ」機能追加

- 設定例

- ディスカバリのプロトタイプ: ディスカバリのプロトタイプ

ディスカバリのプロトタイプ	保存前処理 1	LLDマクロ 1	フィルター	オーバーライド
* 名前	#DB			
タイプ	Nested			
* キー	db.tablespace.discovery[{#DB}]			

プロトタイプ	保存前処理 1	LLDマクロ 1	フィルター	オーバーライド
保存前処理の設定	名前	パラメータ		
1:	JSONPath	\$tablespaces		

タイプ	保存前処理 1	LLDマクロ 1	フィルター	オーバーライド
LLDマクロ	#TSNAME		JSONPath	
	\$name			

「ディスカバリのプロトタイプ」機能追加

- 設定例
 - ネストされたアイテムのプロトタイプ: アイテムのプロトタイプ

アイテムのプロトタイプ	タグ	保存前処理
* 名前	Size of tablespace {#TSNAME} for {#DB}	
タイプ	Zabbixエージェント ▼	
* キー	db.ts.size[{#DB}, {#TSNAME}]	

「ディスカバリのプロトタイプ」機能追加

- 結果

☐ ホスト	名前 ▲	アイテム	トリガー	グラフ	ホスト	ディスカバリルール
☐ [redacted] discovery	Discover databases and tablespaces: db1	アイテムのプロトタイプ1	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ
☐ [redacted] discovery	Discover databases and tablespaces: db2	アイテムのプロトタイプ1	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ
☐ [redacted] discovery	Discover databases and tablespaces: db3	アイテムのプロトタイプ1	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ
☐ [redacted] discovery	Template [redacted] Discovery prototype: Discover databases and tablespaces	アイテムのプロトタイプ1	トリガーのプロトタイプ	グラフのプロトタイプ	ホストのプロトタイプ	ディスカバリのプロトタイプ1

```
[
  {
    "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" }
    ]
  },
  ...
]
```

「ディスカバリのプロトタイプ」機能追加

- 結果: アイテム一覧

☐	名前 ▲	トリガー	キー	監視間隔	ヒストリ	トレンド	タイプ	ステータス	タグ	情報
☐	*** db1 : Size of tablespace ts1 for db1		db.ts.size[db1, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db2 : Size of tablespace ts1 for db2		db.ts.size[db2, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db3 : Size of tablespace ts1 for db3		db.ts.size[db3, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db1 : Size of tablespace ts2 for db1		db.ts.size[db1, ts2]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db2 : Size of tablespace ts2 for db2		db.ts.size[db2, ts2]	1m						
☐	*** db3 : Size of tablespace ts2 for db3		db.ts.size[db3, ts2]	1m						
☐	*** db1 : Size of tablespace ts3 for db1		db.ts.size[db1, ts3]	1m						
☐	*** db2 : Size of tablespace ts3 for db2		db.ts.size[db2, ts3]	1m						
☐	*** db3 : Size of tablespace ts3 for db3		db.ts.size[db3, ts3]	1m						

```
[
  {
    "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" }
    ]
  },
  ...
]
```

「ディスクバリのプロトタイプ」機能追加

- 7.4未満でも対応可能？
 - ディスカバリルールを個別で作成する必要がある
 - db1, db2, db3 をそれぞれ作成

☐	名前 ▲	トリガー	キー	監視間隔	ヒストリ	トレンド	タイプ	ステータス	タグ	情報
☐	* db1: Size of tablespace ts1 for db1		db.ts.size[db1, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db2: Size of tablespace ts1 for db2		db.ts.size[db2, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db3: Size of tablespace ts1 for db3		db.ts.size[db3, ts1]	1m	31d	365d	Zabbixエージェント	有効		
☐	* db1: Size of tablespace ts2 for db1		db.ts.size[db1, ts2]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db2: Size of tablespace ts2 for db2		db.ts.size[db2, ts2]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db3: Size of tablespace ts2 for db3		db.ts.size[db3, ts2]	1m	31d	365d	Zabbixエージェント	有効		
☐	* db1: Size of tablespace ts3 for db1		db.ts.size[db1, ts3]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db2: Size of tablespace ts3 for db2		db.ts.size[db2, ts3]	1m	31d	365d	Zabbixエージェント	有効		
☐	*** db3: Size of tablespace ts3 for db3		db.ts.size[db3, ts3]	1m	31d	365d	Zabbixエージェント	有効		

9件のうち9件を表示しています

Ping監視に機能を追加

- 概要
 - リトライオプション付きの新しいping監視
 - アイテムキーは「icmppingretry」
- icmppingとの違い
 - icmppingは必ずpacketsオプションの回数ping送信する
 - Icmppingretryは成功以降リトライしない
 - 送信されるパケット数を削減するのに役立つ

Ping監視に機能を追加

```
icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]  
icmppingretry[192.168.254.254,3,2.0,,1000]
```

名称	内容	デフォルト値
target	監視対象のIPアドレス または DNS名	ホスト インタフェース
retries	リトライ回数	1
backoff	指数バックオフ、タイムアウトを延ばす設定 浮動小数点指定が必要(1.0 ~ 5.0)	1.0
size	パケットサイズ (バイト単位)	fping依存
timeout	タイムアウト時間 (ミリ秒)	fping依存
options	リダイレクトの許可	未許可

開始

TO: 1.0秒

リトライ1 : タイムアウト1秒 + 指数バックオフ(2.0 x 1)
Timeout: 3.0秒

リトライ2 : タイムアウト3秒 + 指数バックオフ(2.0 x 2)
Timeout: 7.0秒

リトライ3 : タイムアウト7秒 + 指数バックオフ(2.0 x 4)
Timeout: 15.0秒

入力内容のチェック

- リアルタイム構文チェック
 - ホスト
 - アイテム
 - テンプレート
 - トリガー
- わかりやすい
エラーメッセージを表示

新しいホストの作成

ホスト IPMI タグ マクロ インベントリ 暗号化 値のマッピング

* ホスト名
Incorrect characters used for host name.

表示名

テンプレート
検索文字列を入力

* ホストグループ
検索文字列を入力

インターフェース	タイプ	IPアドレス	DNS名	接続方法	ポート	標準
エージェント		192.168.0.1 IPアドレス: Invalid IP address.		☒ IP ☐ DNS	10050	☒ 削除

[追加](#)

説明

監視元 ☒ サーバー ☐ プロキシ ☐ プロキシグループ

有効 ☒

入力内容のチェック

新規テンプレート

 **詳細 ▲** テンプレートを追加できません
パラメータ "/1/host"が正しくありません：ホスト名が正しくありません。

テンプレート タグ マクロ 値のマッピング

* テンプレート名 テンプレート

表示名 テンプレート

テンプレート 検索文字列を入力

* テンプレートグループ Templates ×
検索文字列を入力

説明

7.0

新規テンプレート

テンプレート タグ マクロ 値のマッピング

* テンプレート名 テンプレート
Incorrect characters used for template name.

表示名 テンプレート

テンプレート 検索文字列を入力

* テンプレートグループ Templates ×
検索文字列を入力

説明

切替時に表示

7.4

入力内容のチェック

- トリガーの構文エラーチェックも対応

* 条件式

`nodata(/Zabbix server/agent.ping,5n)=<1`

"<1"から始まる条件式が正しくありません。

* 条件式

`nodata(/Zabbix server/agent.ping,5y)<=1`

関数"nodata"内の2番目のパラメーターが正しくありません。

その他機能

- <https://www.zabbix.com/documentation/current/en/manual/introduction/whatsnew>

5 What's new in Zabbix 7.4.0

 Table of Contents



See [breaking changes](#) for this version.

Nested low-level discovery

It is now possible to create multi-level discovery of objects with the introduction of discovery prototypes within a low-level discovery rule. For example, you may want to discover all database instances on a database server, then discover tablespaces for each instance, then discover tables for each tablespace.

Discovery prototypes are nested discovery rules within a "parent" discovery rule. Discovery prototypes have their own item, trigger, graph, host, and discovery prototypes.

A nested discovery prototype may use the same JSON value as the parent rule, but then use a different "slice" of data from the JSON value.

The levels of nesting for discovery prototypes are unlimited.

Host prototypes on discovered hosts

Host prototypes are now supported on [discovered hosts](#), enabling Zabbix to automatically discover and monitor entities within other discovered entities (e.g., hypervisors, their virtual machines, and containers inside those virtual machines).

You can create host prototypes on discovered hosts by creating low-level discovery rules with host prototypes or by linking a template with host prototypes. Alternatively, you can link a template to the host prototype used for discovery, which will cause discovered hosts to inherit the host prototypes from the template.

If your current configuration includes host prototypes that use templates containing other host prototypes, please see [Upgrade notes](#).

5 What's new in Zabbix 7.4.0

Nested low-level discovery

Host prototypes on discovered hosts

OAuth 2.0 authentication

Host Wizard

Widgets

Item card

Item history

Real-time widget editing

Items

ICMP ping item with retry option

Functions

10/25 OSC 2025 Tokyo/Fall

- その他、7.4 の機能は現地でも直接相談ください。

10月
25

10/25 【展示/ステージ】 オープンソースカンファレンス2025 Tokyo/Fall

浅草でオープンソースの文化祭！出展者募集中です

主催：オープンソースカンファレンス実行委員会

オープンソースの「今」を伝える

Open Source Conference

コミュニティ・協賛企業・後援団体によるオープンソースの祭典！

ハッシュタグ： #osc25tk

<https://ospn.connpass.com/event/361403/>

日本Zabbixユーザー会イベント

今年からコミュニティイベントを復活してます



終了したイベント

全てのイベントを見る (4件)

日本Zabbixユーザー会 Japanese Zabbix Community

2025/08/21 (木) 18:30～

43/230人

日本Zabbixユーザー会 Zabbix創設者Alexei来日イベント

👤  _BSmile_ 他

📍 東京都港区東新橋2-16-3 カーザペルソーレ 8F

日本Zabbixユーザー会 Japanese Zabbix Community

2025/07/04 (金) 18:30～

111/263人

第8回 日本Zabbixユーザー会 勉強会

👤  _BSmile_ 他

📍 東京都中野区中野2-30-5 中野アーバンビル7F

日本Zabbixユーザー会 Japanese Zabbix Community

2025/04/04 (金) 19:00～

129人

第7回 日本Zabbixユーザー会(旧ZABBIX-JP) 勉強会

👤  _BSmile_ 他

📍 東京都中野区中野2-30-5 中野アーバンビル7F

次回開催のイベント



登壇時間約5分!

<https://qr.paps.jp/2J7NH>

Zabbix カンファレンスに
ユーザー会のための
特設ステージが!?



次回イベント



開催前 2025/11/06 (木) 17:30~

第9回 日本Zabbixユーザー会 in Zabbix Conference Japan 2025

👤  _BSmile_ 他

📍 〒105-7501 東京都港区海岸1-7-1 東京ポートシティ竹芝 オフィスタワー1階

次回開催のイベント

- Zabbixコミュニティが提供するSlackも活用してください👍



<https://qr.paps.jp/ztmfY>

 **わたなべ** 10:20
本日から開催している OSC Online の17:00 - 17:45で登壇させていただきます。
お耳に余裕がある方はぜひ聞いてください。

[Zabbix 最新バージョン 7.4の機能と Zabbixユーザー会イベントの案内](#)

- 担当：日本Zabbixユーザー会
- 講師：渡邊隼人：わたなべはやと（コミュニティスタッフ）

<https://ospn.connpass.com/event/361402/>

 **OSPN.jp**
[Zabbix 最新バージョン 7.4の機能と Zabbixユーザー会イベントの案内 - セミナープログラム - オープンソースカンファレンス2025 Online/Fall \(329 kB\)](#)



A photograph showing a large audience seated in a lecture hall, facing a stage with a green chalkboard and two large projection screens. The screens display presentation slides.

 **connpass**
[10/17-18 Open Source Conference 2025 Online/Fall \(2025/10/17 10:00~\)](#)
イベントご参加(視聴)の方へ：注意事項のご案内 本イベントのプログラムは録画されており、後日YouTubeでアーカイブ公開されます。connpassアカウントのメールアドレスが最新のアドレスになっているかどうかご確認ください。古いメールアドレスなどでご登録されていますと、参加用URLのお知らせが届けられません。参加用URLを受信していただくために、10月16日(木)16:00までに、当ページにて参加登録をお願いします。（それ以降の登録も可能です。） # Zoom参加にはZoomアカウントへのログインが必要 セキュリティの関係から、Zoom参加にはZoom... (76 kB)

END 🖐️