ZABBIX 5.0 LTS

Freedom and integrity of monitoring

Quick recap of Zabbix 4.2 and 4.4

Zabbix 4.2

April, 2019

High frequency monitoring with throttling

Data collection: HTTP agent, Prometheus

Preprocessing: validation and JavaScript!

Preprocessing by Proxies

Enhanced tag management

Zabbix 4.4

September, 2019

New Zabbix Agent: plugins, scheduler and more

Web hooks for alerting and notifications

Support of TimescaleDB

Built-in knowledge base for metrics and triggers

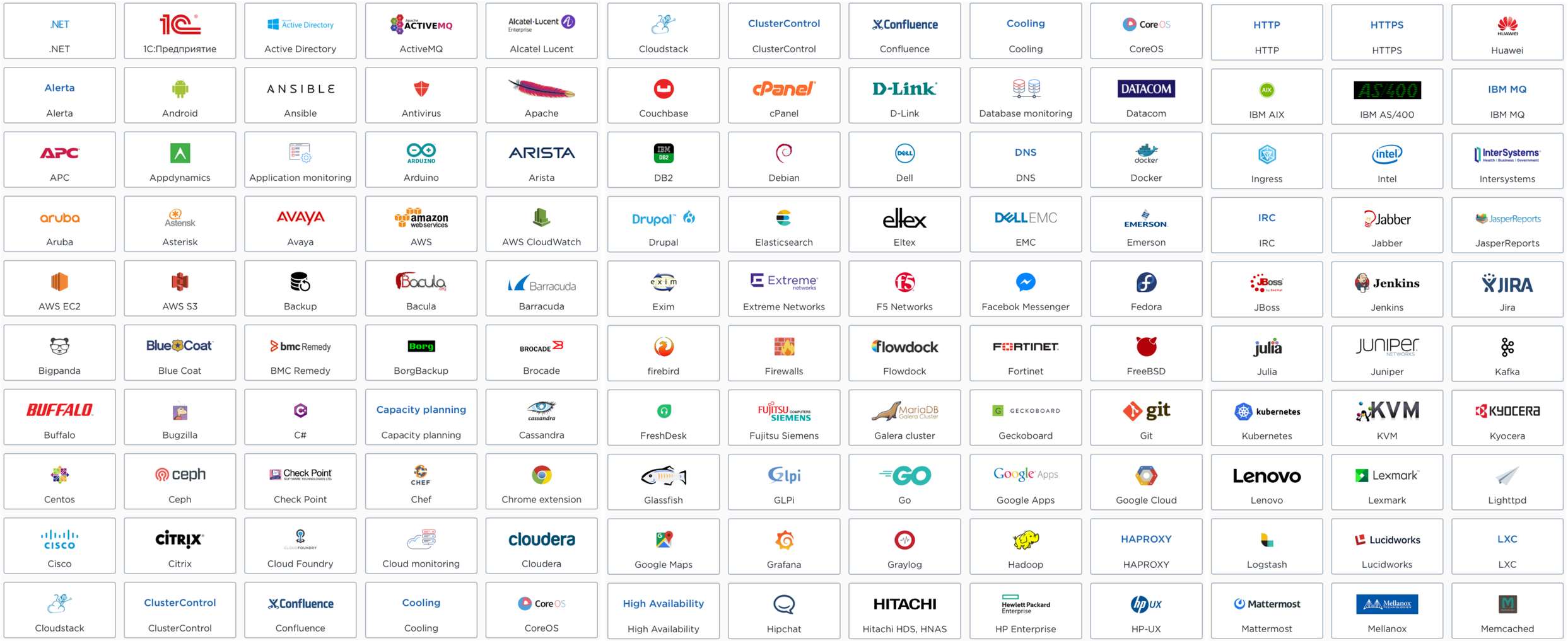
Standard for Zabbix Templates

5.0

LTS release

May 12, 2020

Available templates for monitoring & integrations



https://www.zabbix.com/integrations

Making a platform for high quality solutions

TEMPLATES & PLUGINS

Huawei
IBM MQ
Intersystems
JasperReports
Jira
Kafka
Kyocera
Lighttpd
LXC
Memcached

Zabbix Agent 2

Data

Notifications

Incidents

Events

WEBHOOK BASED INTEGRATIONS

.NET
Alerta
APC
Aruba
AWS EC2
Bigpanda
Buffalo
Centos
Cisco
Cloudstack

WEBHOOKS

Ticketing

Alerting

TEMPLATES & PLUGINS

Monitoring

Easy to contribute!

3 simple steps

Sign Zabbix Contributor Agreement (ZCA)
https://www.zabbix.com/developers

Make Zabbix Pull Request
https://git.zabbix.com

Zabbix Dev Team will review and accept if everything is fine

Official support of Zabbix Agent2 for Linux and Windows

Most advanced monitoring agent on the market!

New Zabbix Agent
(zabbix_agent2)

- Plugin infrastructure
- Support of long running scripts
- Parallel active checks
- Support of flexible intervals for all checks
- Support of persistent connections (DB connections)
- Accepting incoming traps and events (MQTT subscribe, listening TCP/UDP ports, etc)
- Monitoring of systemd services out of the box
- Drop-in replacement of the existing agent!

Persistent storage for Agent2

Zabbix Agent2

EnablePersistentBuffer=1
PersistentBufferFile=/var/spool/zabbix/agent.db
PersistentBufferPeriod=1d

No connection

Zabbix Server

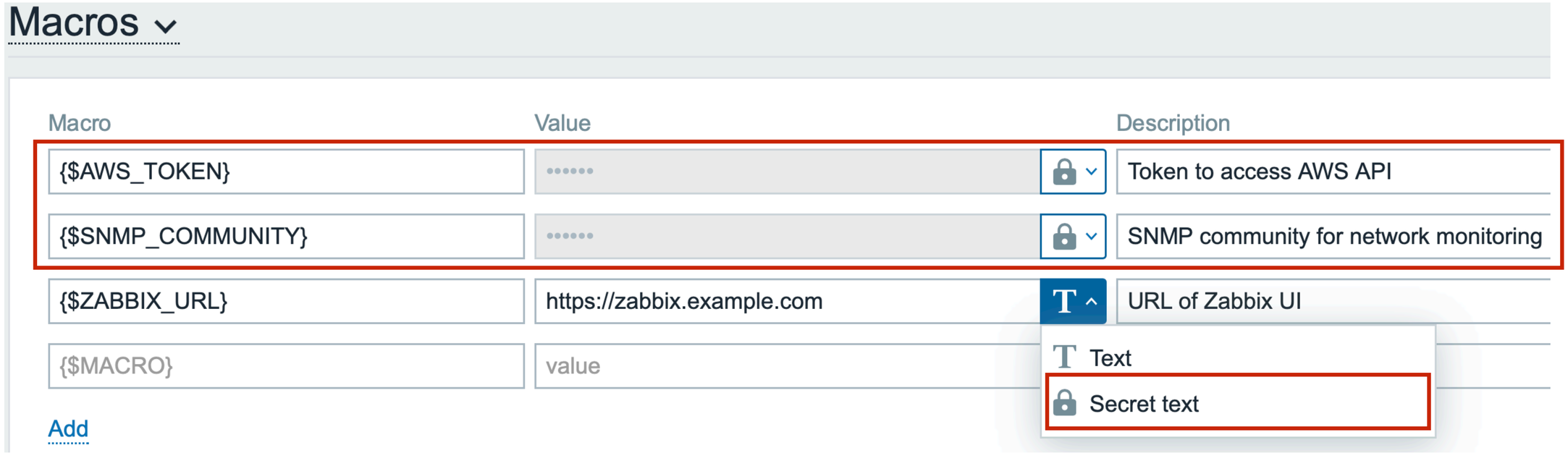
Use cases

Unstable communications
Monitoring of critical data
Bursts of data

Your data is safe!

Secure by design

Secret macros



Use cases

Hide any secrets: passwords, tokens, IDs

Data protection

Secret text cannot be retrieved in UI and alerts, masked with ******

No read access, can only be replaced with a new value

Security and encryption

Agent

Agent

Agent

Agent

Whitelists or blacklists for agent metrics

PSK or TLS

ZABBIX

Zabbix Server

HTTPS

HTTP Proxy

Webhooks over HTTP Proxy

HTTPS

Jira

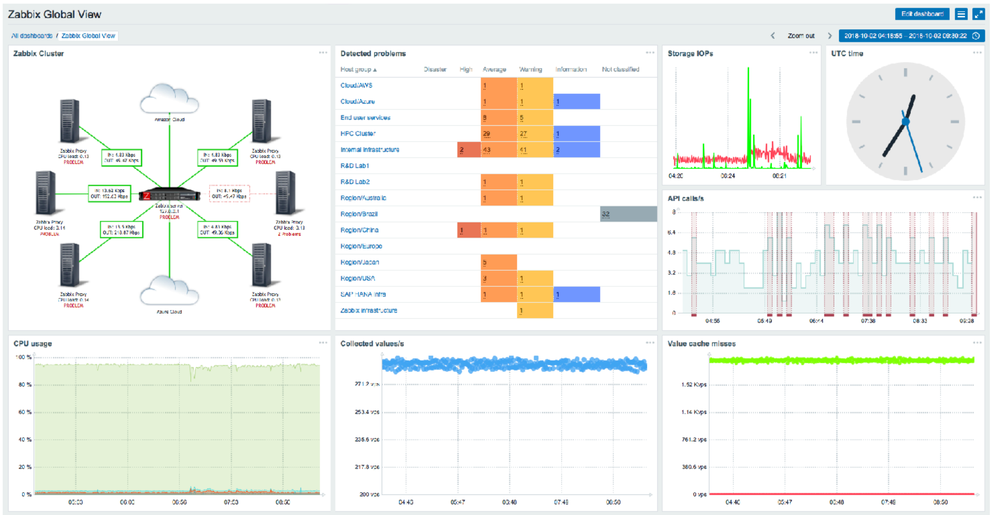
HTTPS

TLS

Database

Encrypted connections to MySQL and PostgreSQL

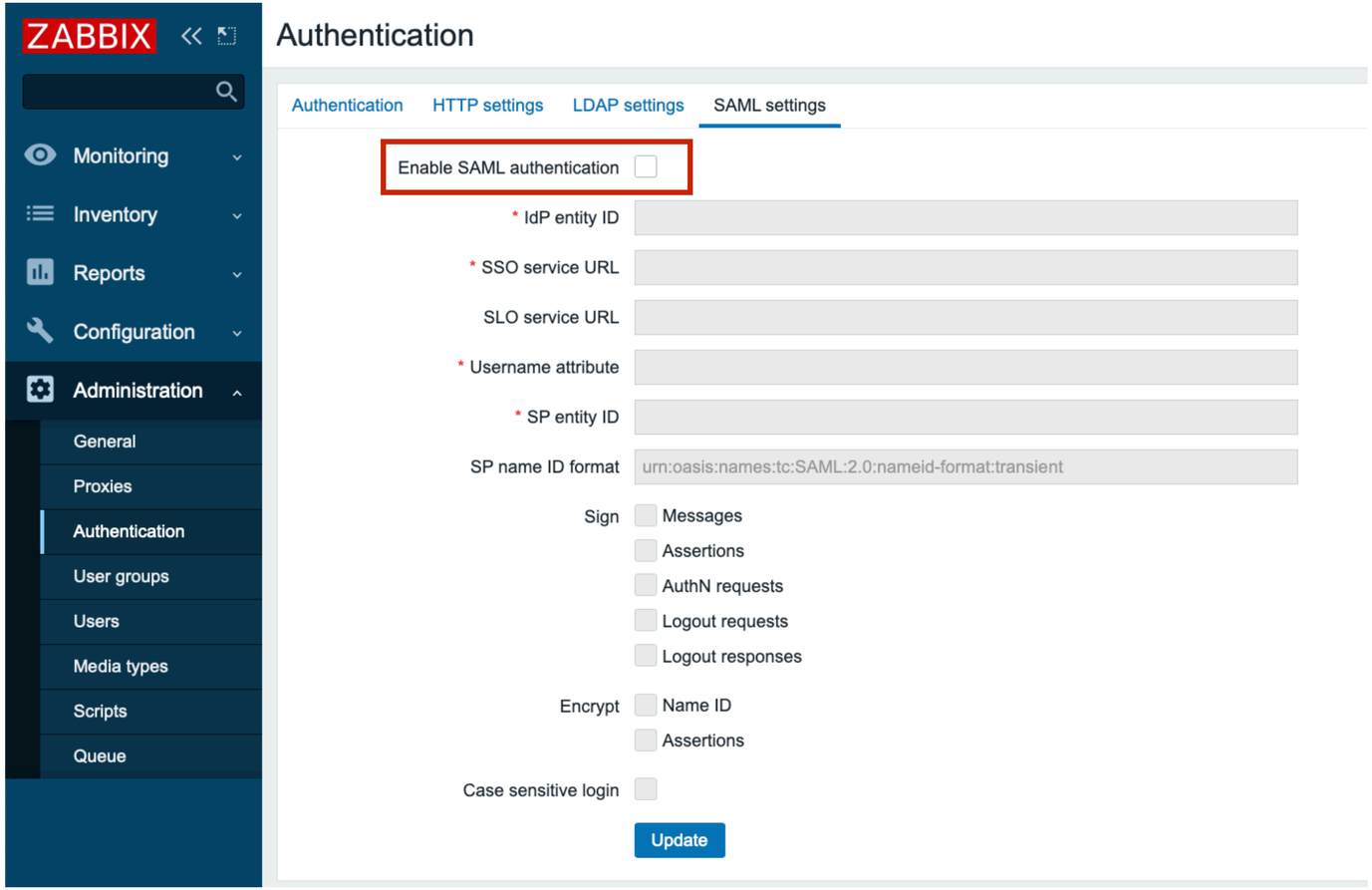
Bcrypt for password hashes



Masking secrets in UI and alerts

SAML authentication for single sign-on

Identity providers



Azure

AWS

Microsoft
Active Directory
Federation Services

OpenAM

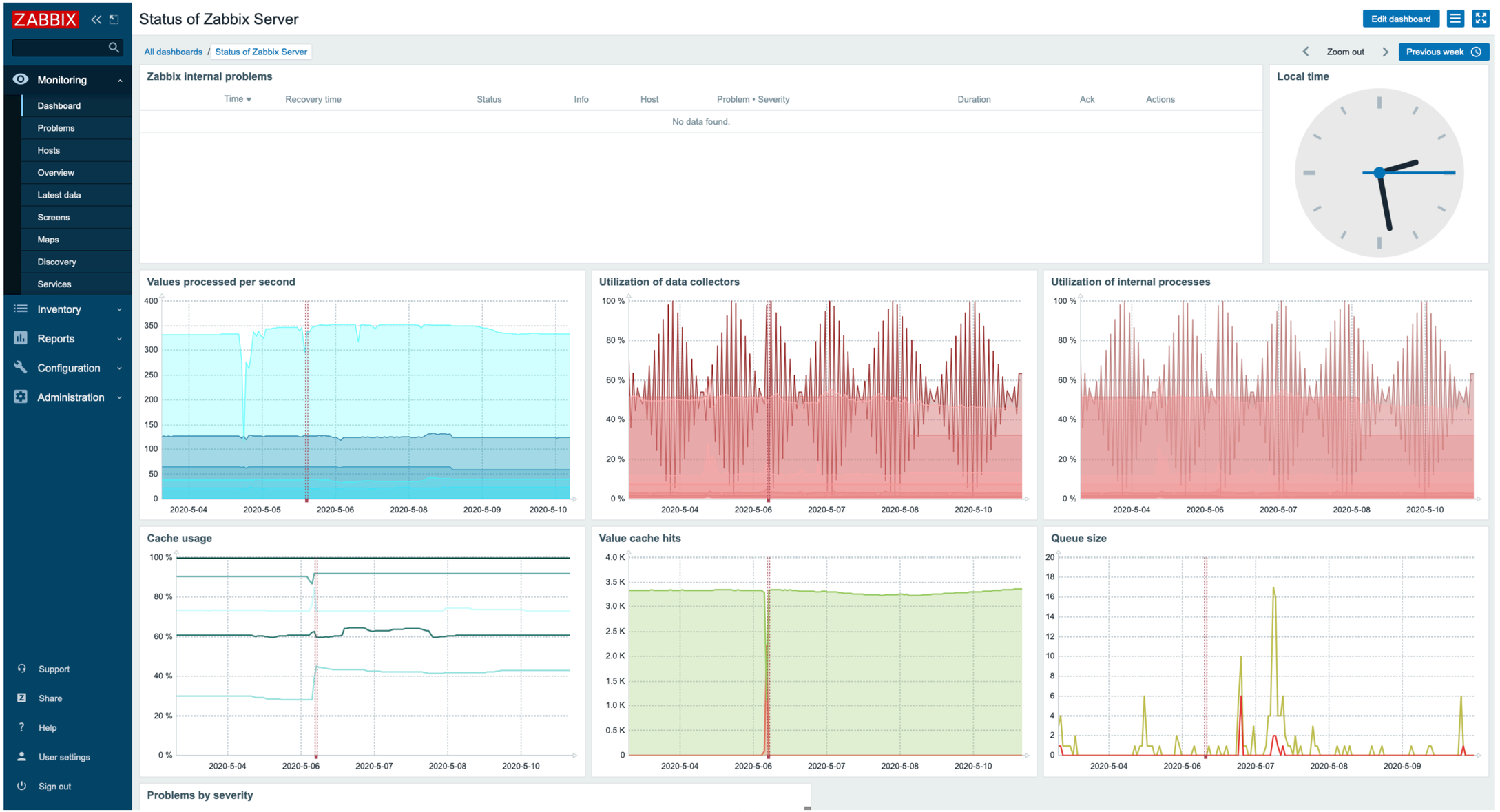
SECUREAUTH

okta

Auth0

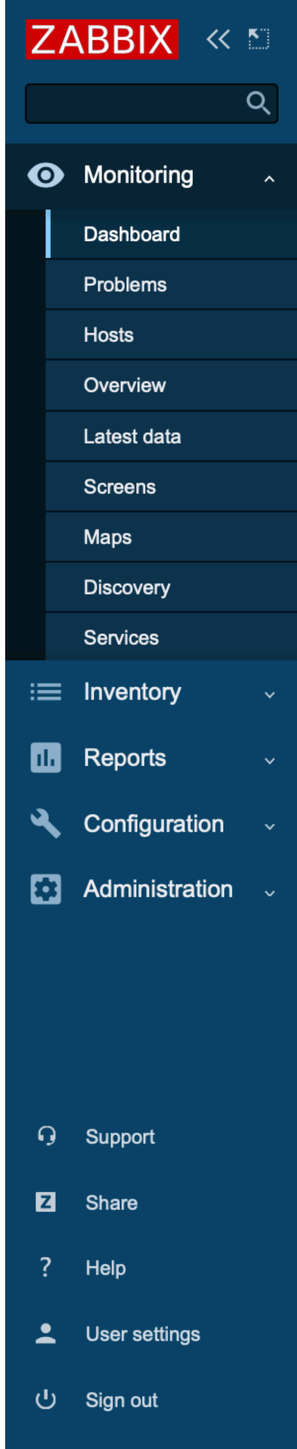
Usability improvements

Optimized for wide screens



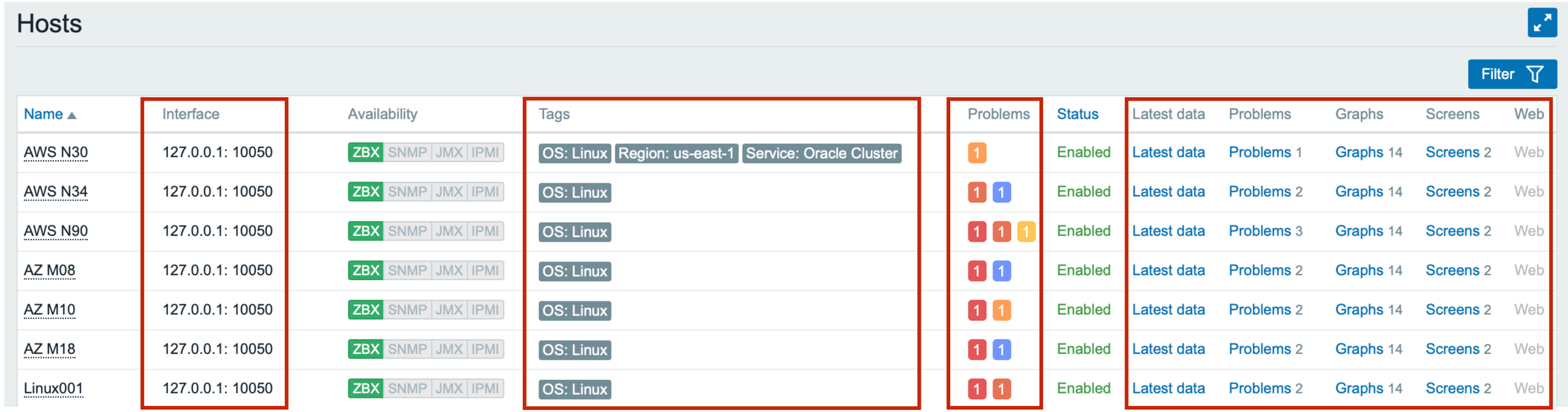
UI modules

- Create new menu entries
- Create new pages
- Extend existing functionality
- Use and share 3rd party extensions
- Permission control



List of monitored devices

Monitoring->Hosts



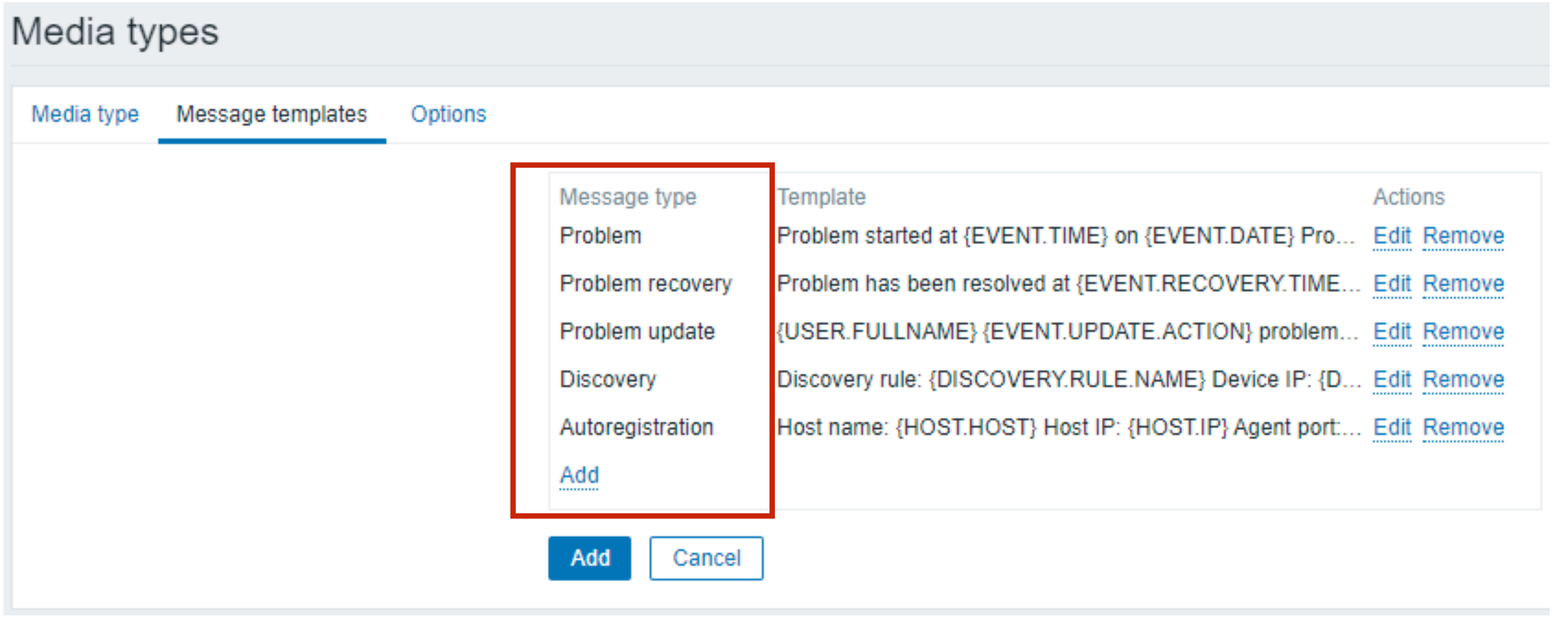
No more Monitoring->WEB and Monitoring->Graphs

Easy navigation to host related resources

Message templates for media types

Templates for different message types

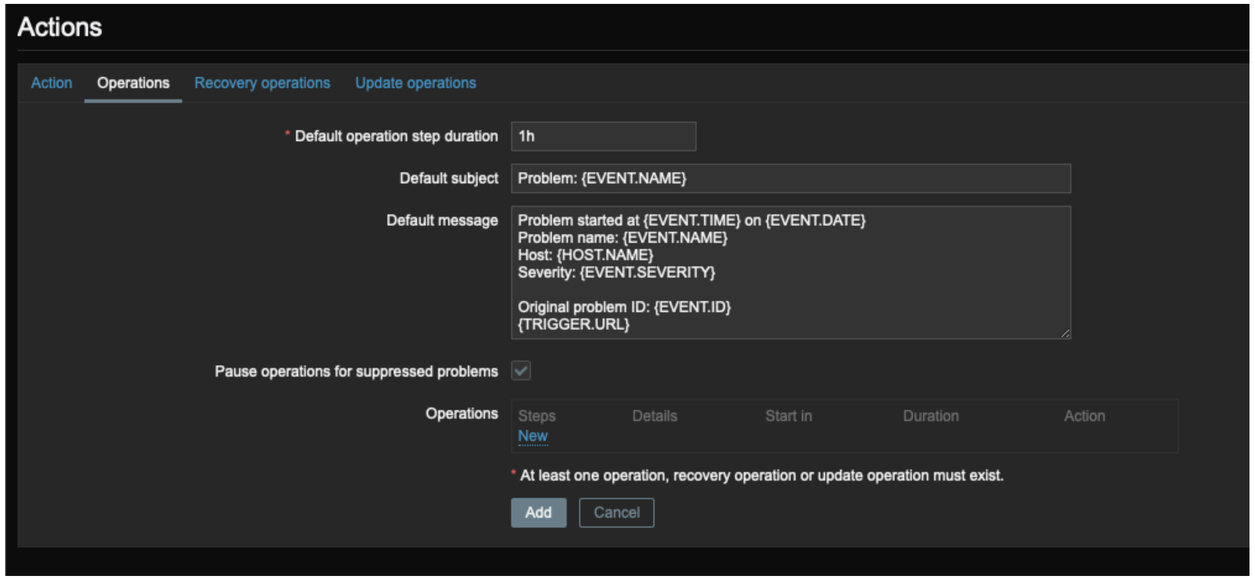
Email media type



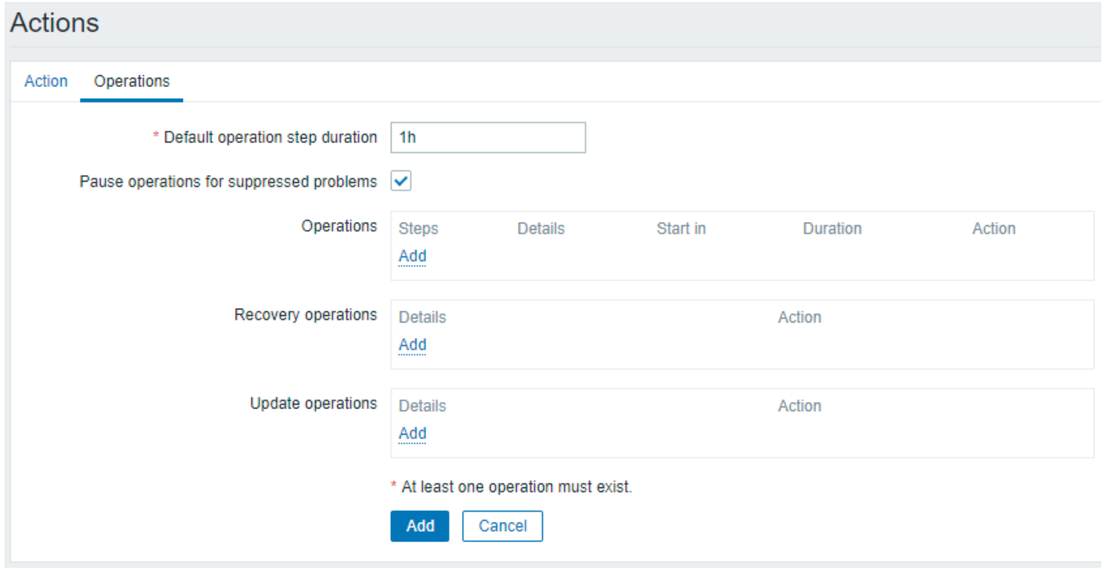
Message template

Configure message format in on place!

Before



After



CLI tool to test JS scripts

Why

Typical use cases

Test JavaScript code from command line:

- webhooks
- complex preprocessing scripts

How to use it?

```
shell> zabbix_js -help

Execute script using Zabbix embedded scripting engine.

General options:
  -s,--script script-file      Specify the filename of script to execute. Specify
- for
                               standard input.
  -i,--input input-file        Specify input parameter file name. Specify - for
                               standard input.
  -p,--param input-param       Specify input parameter
  -l,--loglevel log-level      Specify log level
  -t,--timeout timeout         Specify timeout in seconds
  -h --help                    Display this help message
  -V --version                 Display version number
```

Triggers support text operations

Text data

Typical use cases

Working with software versions

Log file monitoring

Comparing string values of different items

Comparing last and previous values

Supported operators: = <>

Comparing with text constant

```
{host:zabbix.version.last()}=“5.0.0”
{host:zabbix.version.last()}=“{$ZABBIX.VERSION}”
```

Comparing last value with previous one

```
{host:text.last()}<>{host:text.prev()}
```

OR

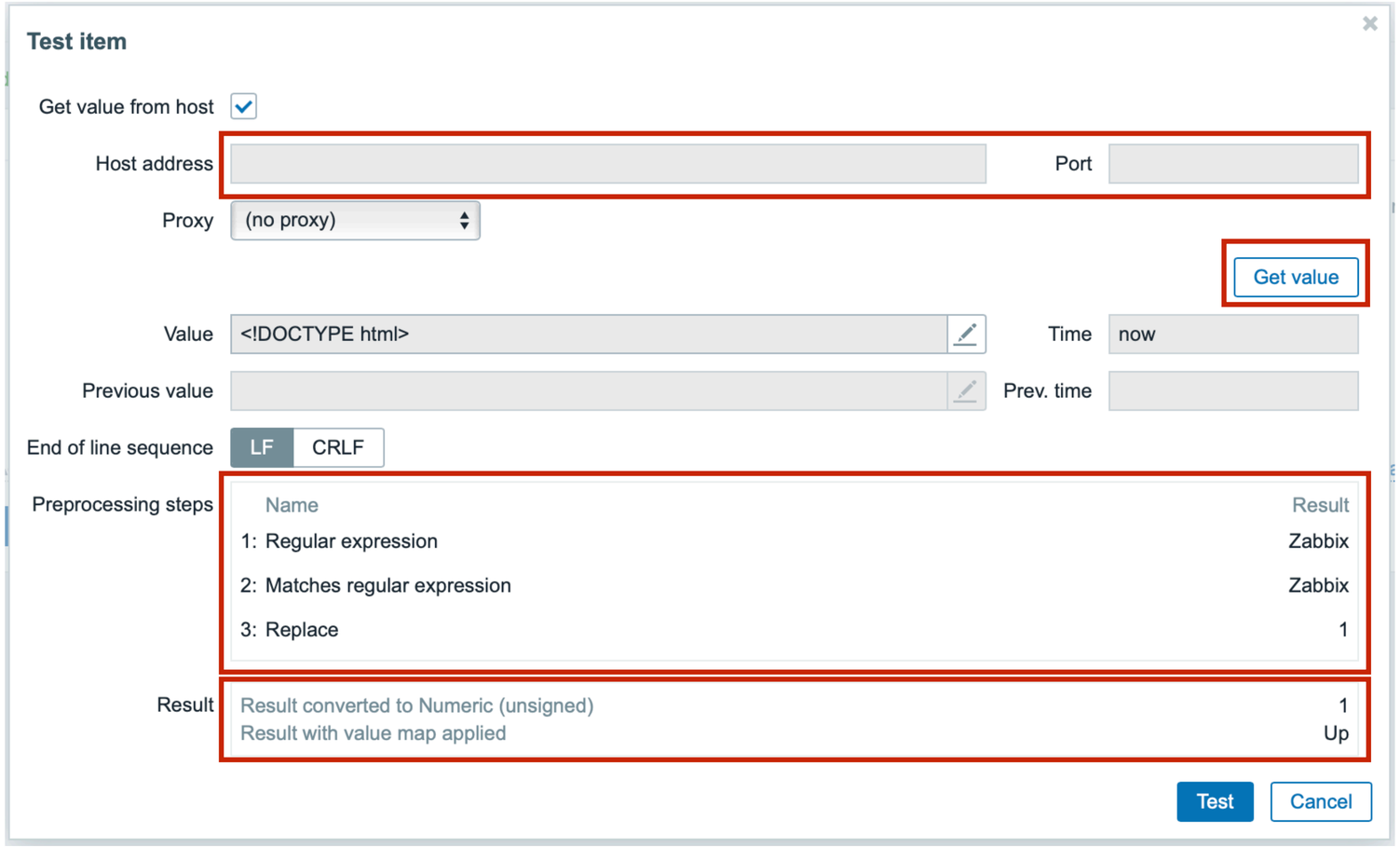
```
{host:text.last(#1)}<>{host:text.last(#2)}
```

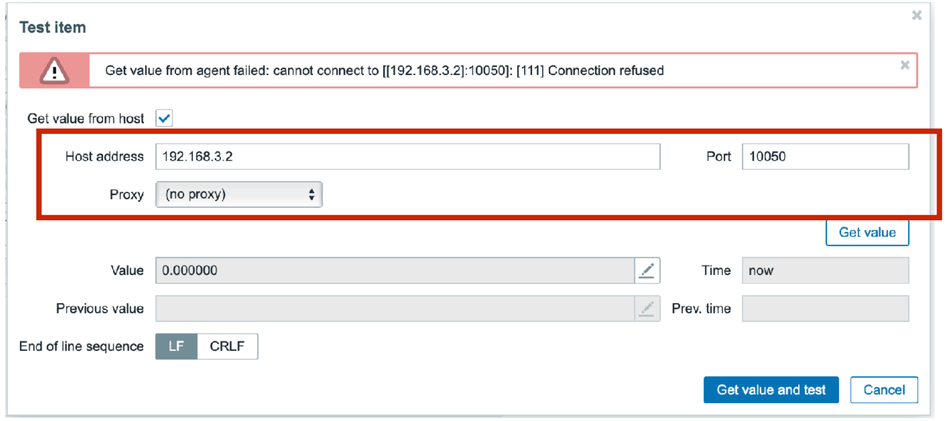
Comparing values of different items

```
{hostA:textA.last()}={hostB:textB.last()}
```

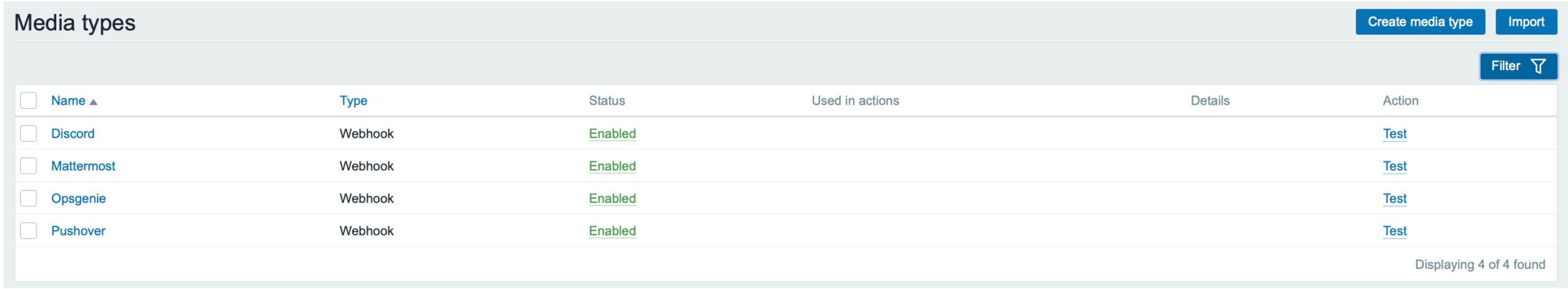
Test item from UI

For hosts and templates





Do not forget to test media types too!



Compression for efficiency

TimescaleDB = PostgreSQL + Extension

Lower storage cost

TIMESCALE

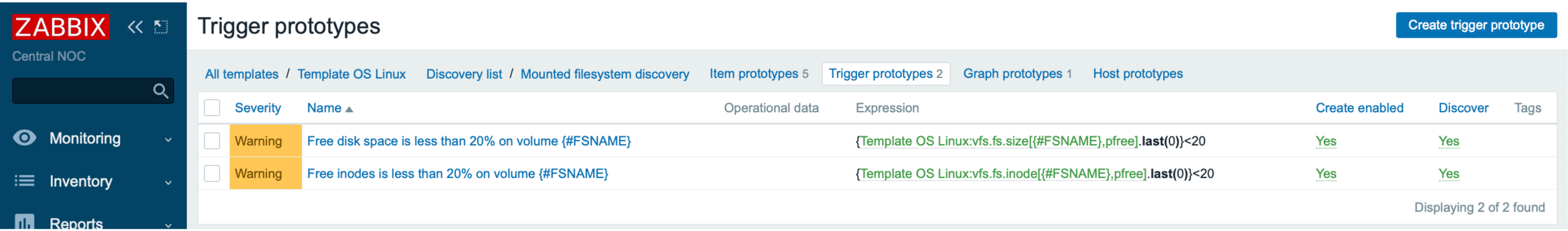
Numbers from a production setup

Zabbix 4.x: 355GB

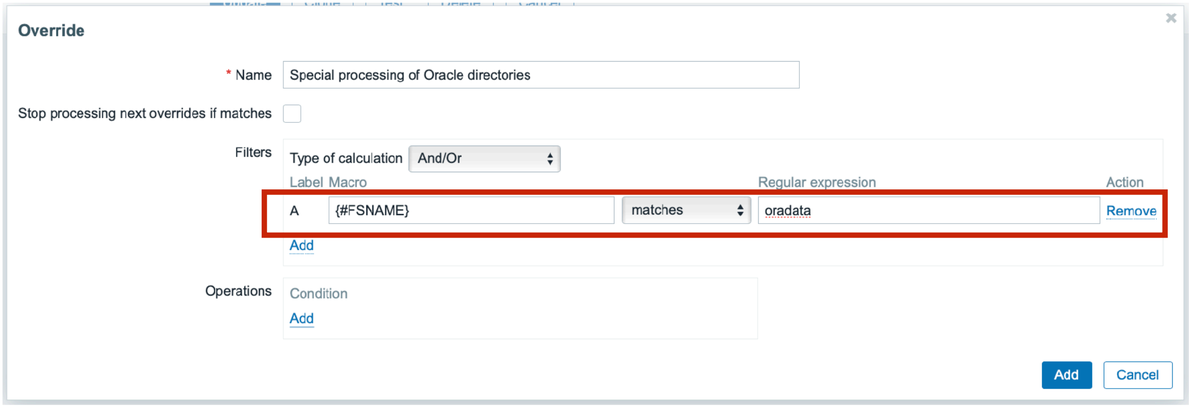
Zabbix 5.0: 43GB

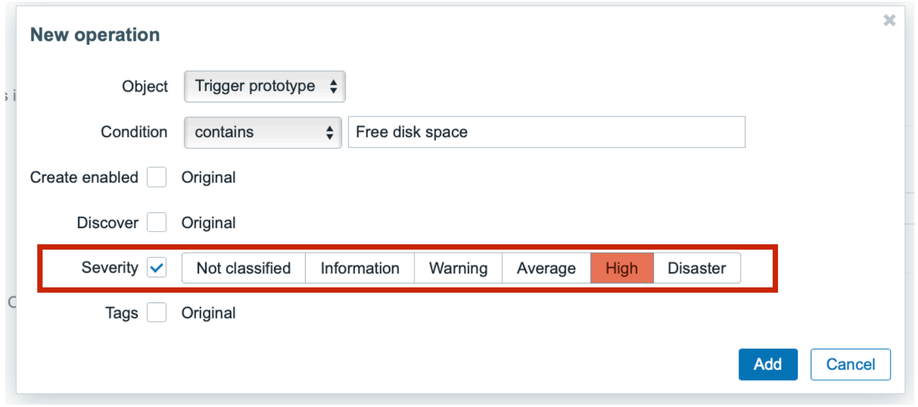
Overrides for LLD rules

Discovery of filesystems

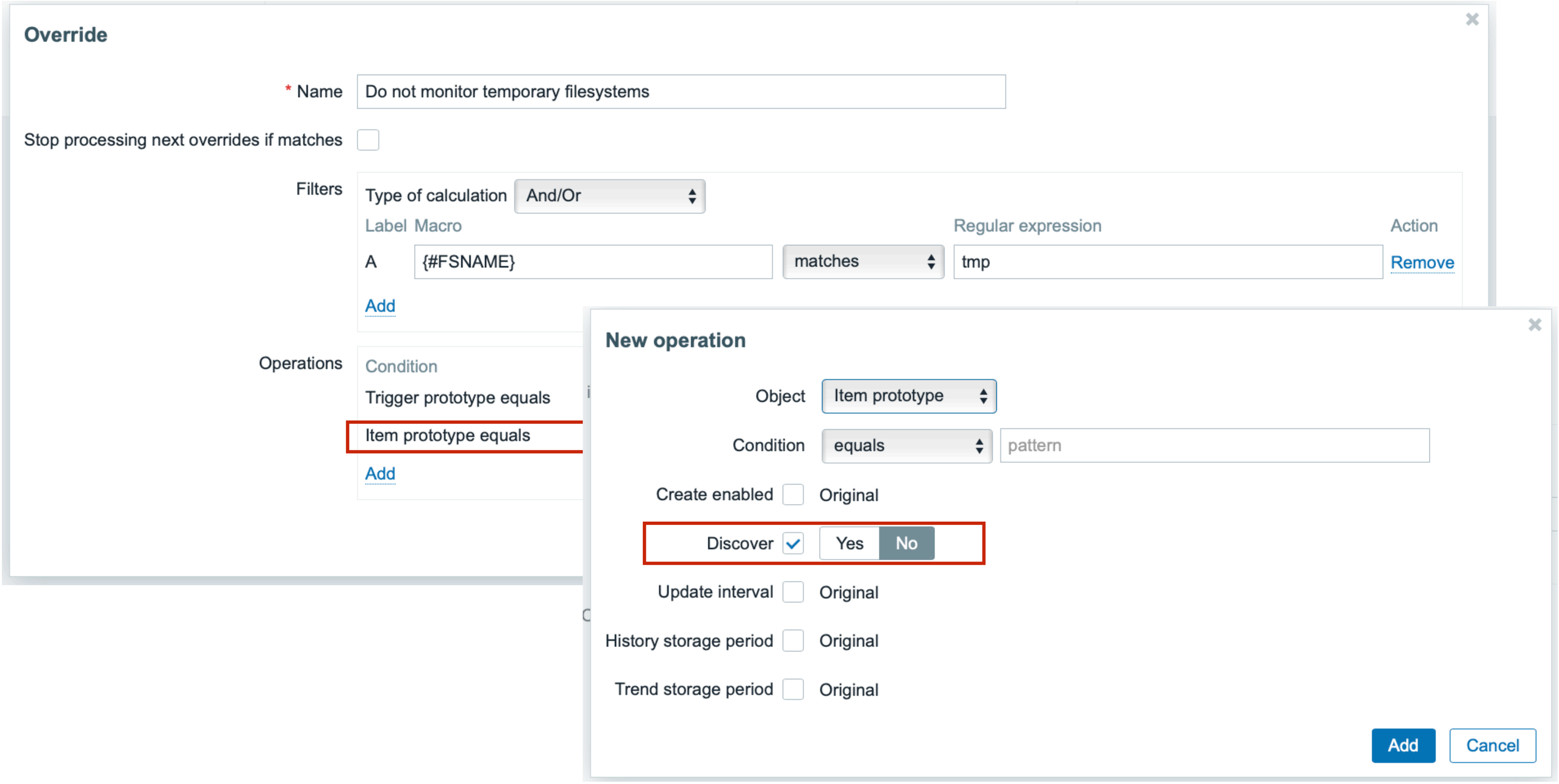


Special treating of Oracle related filesystems





Do not discover temporary filesystems



5.2

Standard release

Synthetic monitoring

Why?

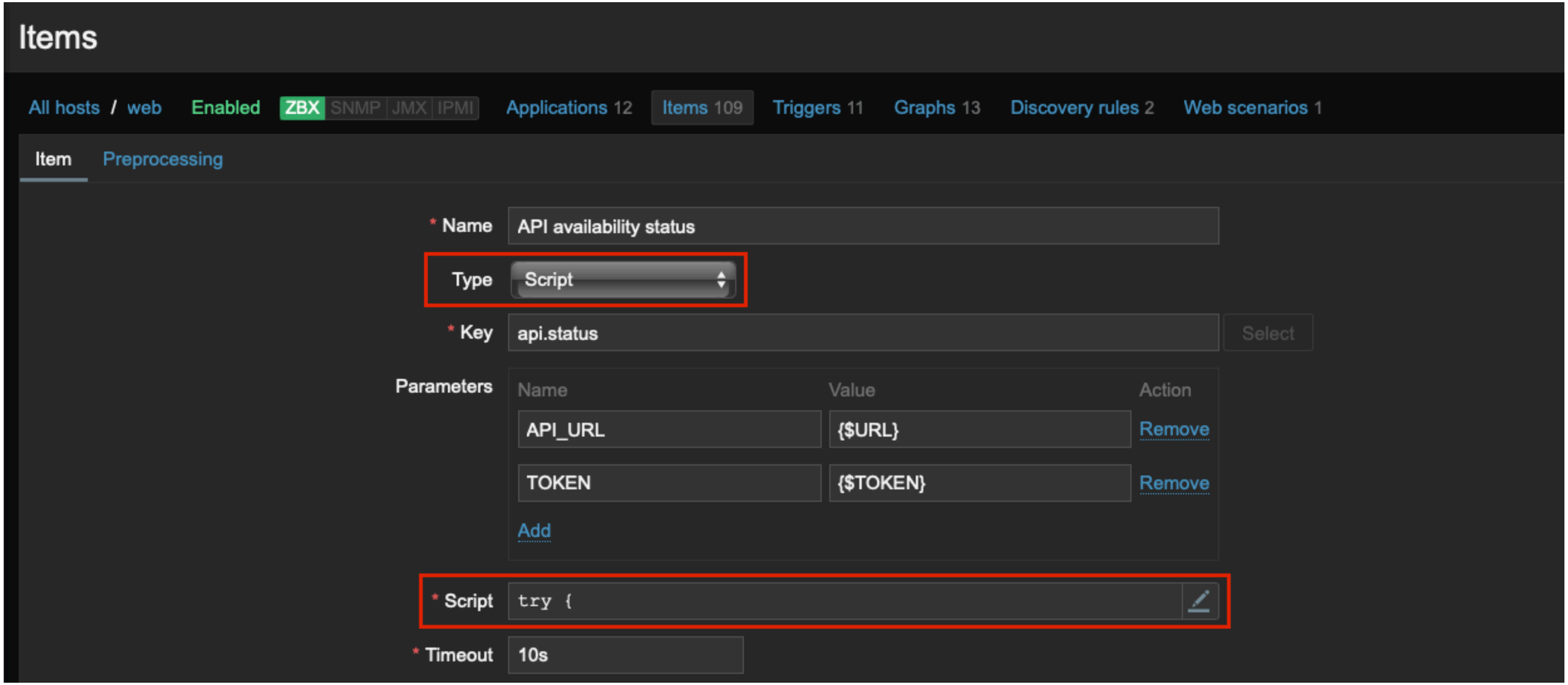
Typical use cases

Support of complex multi-step scripted monitoring scenarios

Advanced availability checks

Complex interaction with different HTTP APIs

New item type Script



Storage of secrets in external Vault

Why?

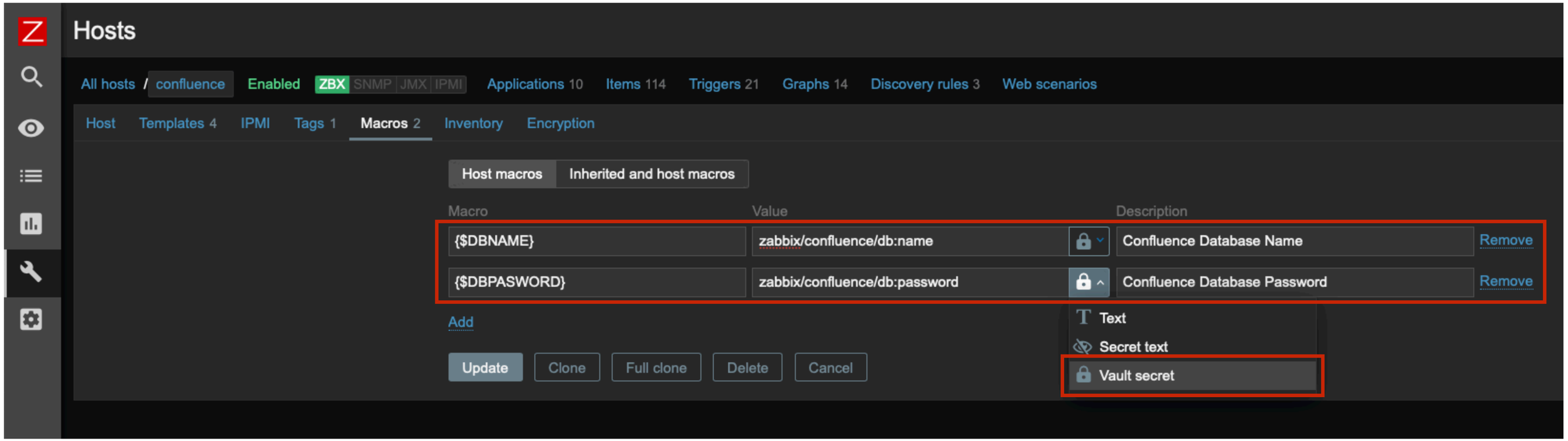
Benefits

All sensitive information is kept outside of Zabbix in a secure place

Therefore no secret data is stored in the database

Sensitive data: passwords, API tokens, user names, etc

New macro type Vault secret



Zabbix Insights

Zabbix Insights

Average number of transactions increased by 24% in September

Zabbix Insights

“Number of new user registrations is down 12% last week“

Why?

Benefits

Ability to analyze long term data efficiently using new trigger functions

Zabbix will provide you with information about anomalies

Granular control of user permissions

Why?

Benefits

Control what parts of UI are accessible

Control user operations:
- problem acknowledgements
- creation and editing of maps and dashboards
- management of maintenance times
- etc

Control access to API and its methods

IoT monitoring

Support of modbus and MQTT protocols

5.2

ETA: October, 2020

Do no miss Zabbix Summit 2020, it is FREE!

https://www.zabbix.com/events

Thank you!
Questions?

Some of the used icons made by Freepik from www.flaticon.com