



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

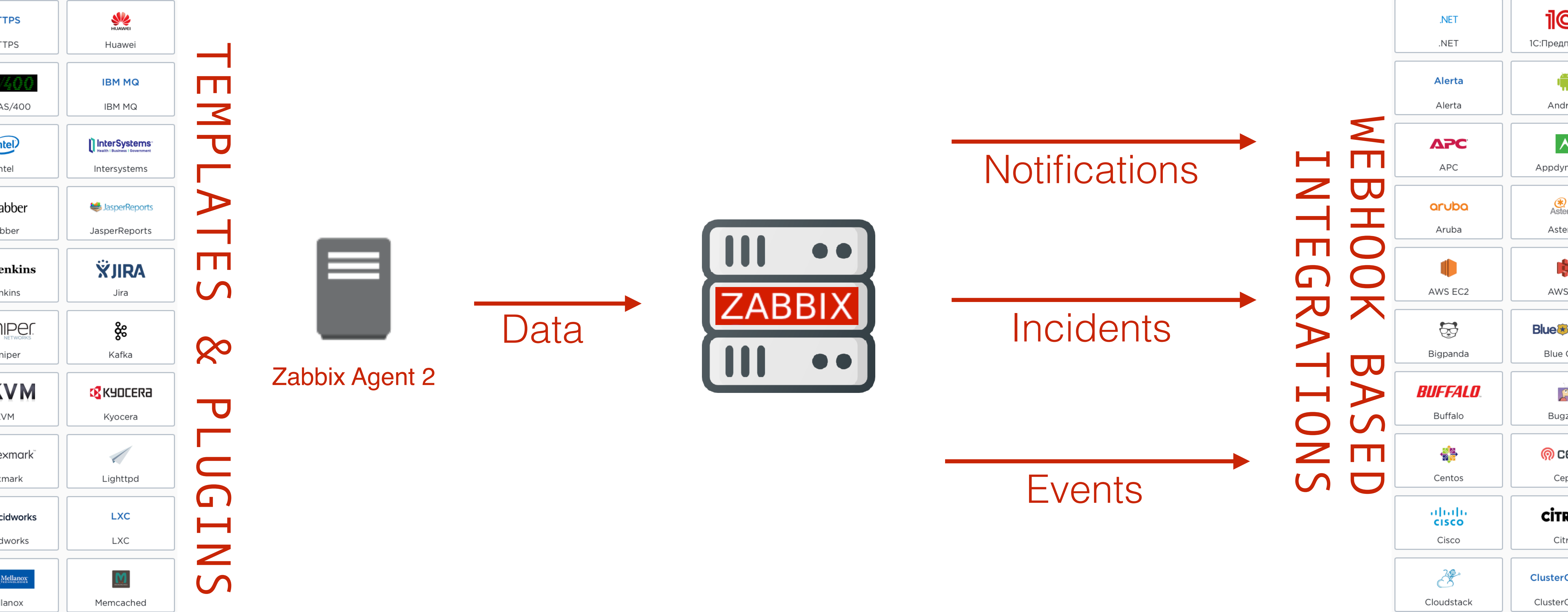
LTS release
May 12, 2020

Available templates for monitoring & integrations

 .NET	 1C:Предприятие	 Active Directory	 ActiveMQ	 Alcatel Lucent	 Cloudstack	 ClusterControl	 Confluence	 Cooling	 CoreOS	 HTTP	 HTTPS	 Huawei
 Alerta	 Android	 Ansible	 Antivirus	 Apache	 Couchbase	 cPanel	 D-Link	 Database monitoring	 Datacom	 IBM AIX	 IBM AS/400	 IBM MQ
 APC	 Appdynamics	 Application monitoring	 Arduino	 Arista	 DB2	 Debian	 Dell	 DNS	 Docker	 Ingress	 Intel	 InterSystems
 Aruba	 Asterisk	 Avaya	 AWS	 AWS CloudWatch	 Drupal	 Elasticsearch	 Eltex	 EMC	 Emerson	 IRC	 Jabber	 JasperReports
 AWS EC2	 AWS S3	 Backup	 Bacula	 Barracuda	 Exim	 Extreme Networks	 F5 Networks	 Facebook Messenger	 Fedora	 JBoss	 Jenkins	 Jira
 Bigpanda	 Blue Coat	 BMC Remedy	 BorgBackup	 Brocade	 firebird	 Firewalls	 Flowdock	 Fortinet	 FreeBSD	 Julia	 Juniper	 Kafka
 Buffalo	 Bugzilla	 C#	 Capacity planning	 Cassandra	 FreshDesk	 Fujitsu Siemens	 Galera cluster	 Geckoboard	 Git	 Kubernetes	 KVM	 Kyocera
 Centos	 Ceph	 Check Point	 Chef	 Chrome extension	 Glassfish	 GLPi	 Go	 Google Apps	 Google Cloud	 Lenovo	 Lexmark	 Lighttpd
 Cisco	 Citrix	 Cloud Foundry	 Cloud monitoring	 Cloudera	 Google Maps	 Grafana	 Graylog	 Hadoop	 HAProxy	 Logstash	 Lucidworks	 LXC
 Cloudstack	 ClusterControl	 Confluence	 Cooling	 CoreOS	 High Availability	 Hipchat	 Hitachi HDS, HNAS	 HP Enterprise	 HP-UX	 Mattermost	 Mellanox	 Memcached

<https://www.zabbix.com/integrations>

Making a platform for high quality solutions



Ticketing



Alerting





Monitoring

Easy to contribute!

3 simple steps

Sign Zabbix Contributor Agreement (ZCA)

<https://www.zabbix.com/developers>

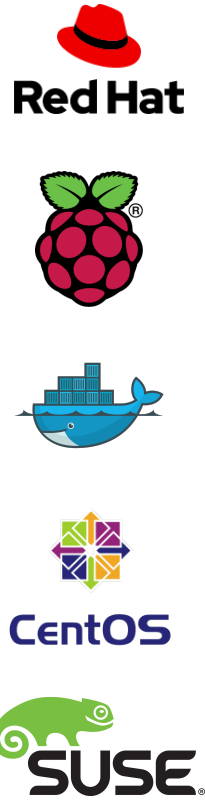
Make Zabbix Pull Request

<https://git.zabbix.com>

Zabbix Dev Team will review and accept if everything is fine

Available everywhere!

Linux distributions and containers







RHEL and CentOS 6, 7 and 8
Debian 8, 9, 10
SuSE 12, 15
16.04 (Xenial), 18.04 (Bionic) and 20.04 (Focal Fossa)
Raspbian 9 (Stretch), 10 (Buster)
Docker


Linux appliance images

ISO
VMWare, VirtualBox
Microsoft Hyper-V
KVM
XEN
LiveCD

Public clouds

 AWS	Zabbix Server 4.4 Mysql + Nginx	 Azure	Zabbix Server 4.4 Mysql + Nginx	 DigitalOcean	Zabbix Server 4.4 Mysql + Nginx	 Google Cloud	Zabbix Server 4.4 Mysql + Nginx
--	------------------------------------	---	------------------------------------	---	------------------------------------	---	------------------------------------



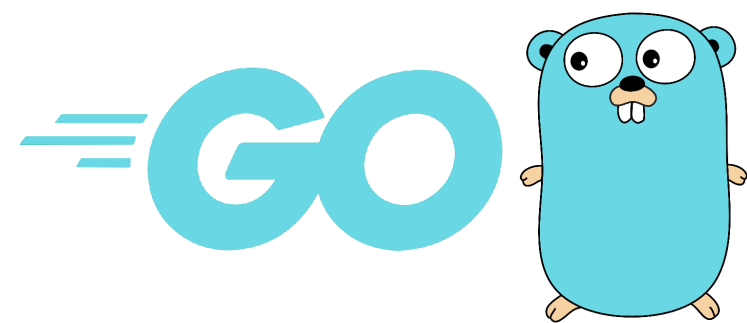


Official support of Zabbix
Agent2 for Linux and
Windows

Most advanced monitoring agent on the market!



New Zabbix Agent
(zabbix_agent2)



Plugin infrastructure

Support of long running scripts

Parallel active checks

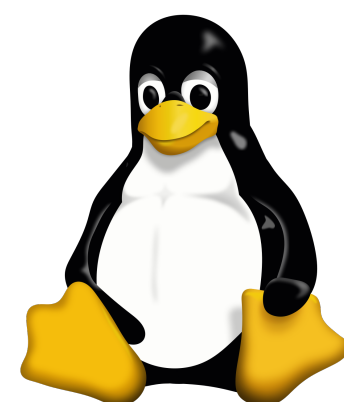
Support of flexible intervals for all checks

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

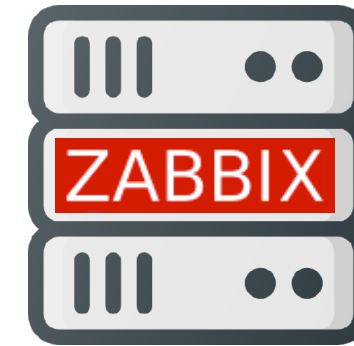


Zabbix Agent2

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



No connection




Zabbix
Server

Use cases

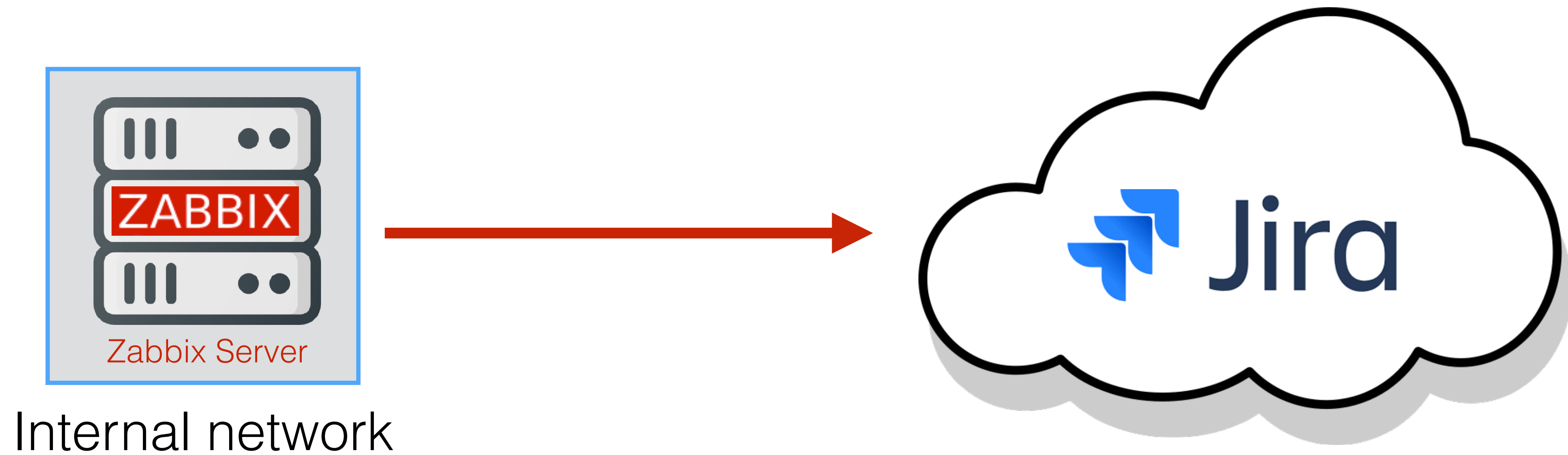
Unstable communications
Monitoring of critical data
Bursts of data

Your data is safe!

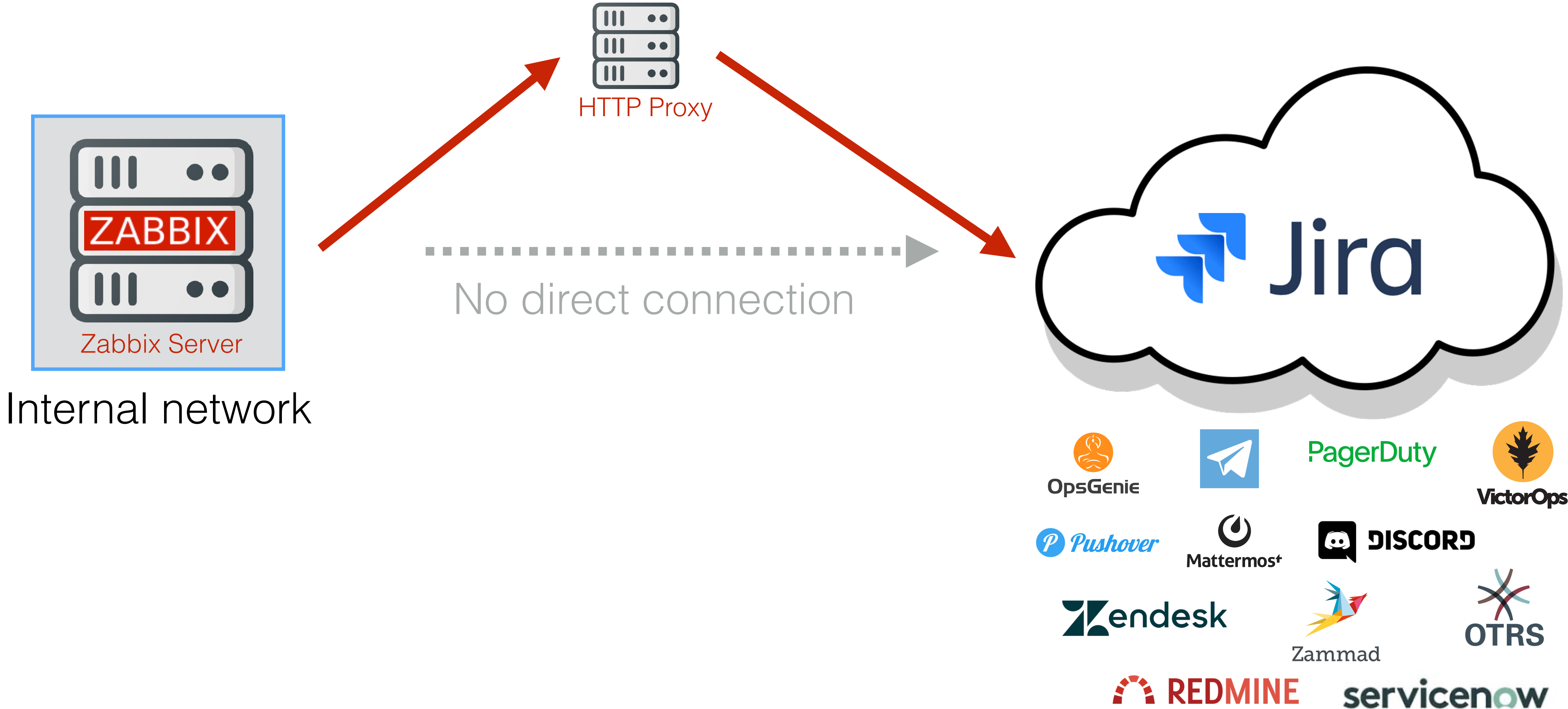


Secure by design

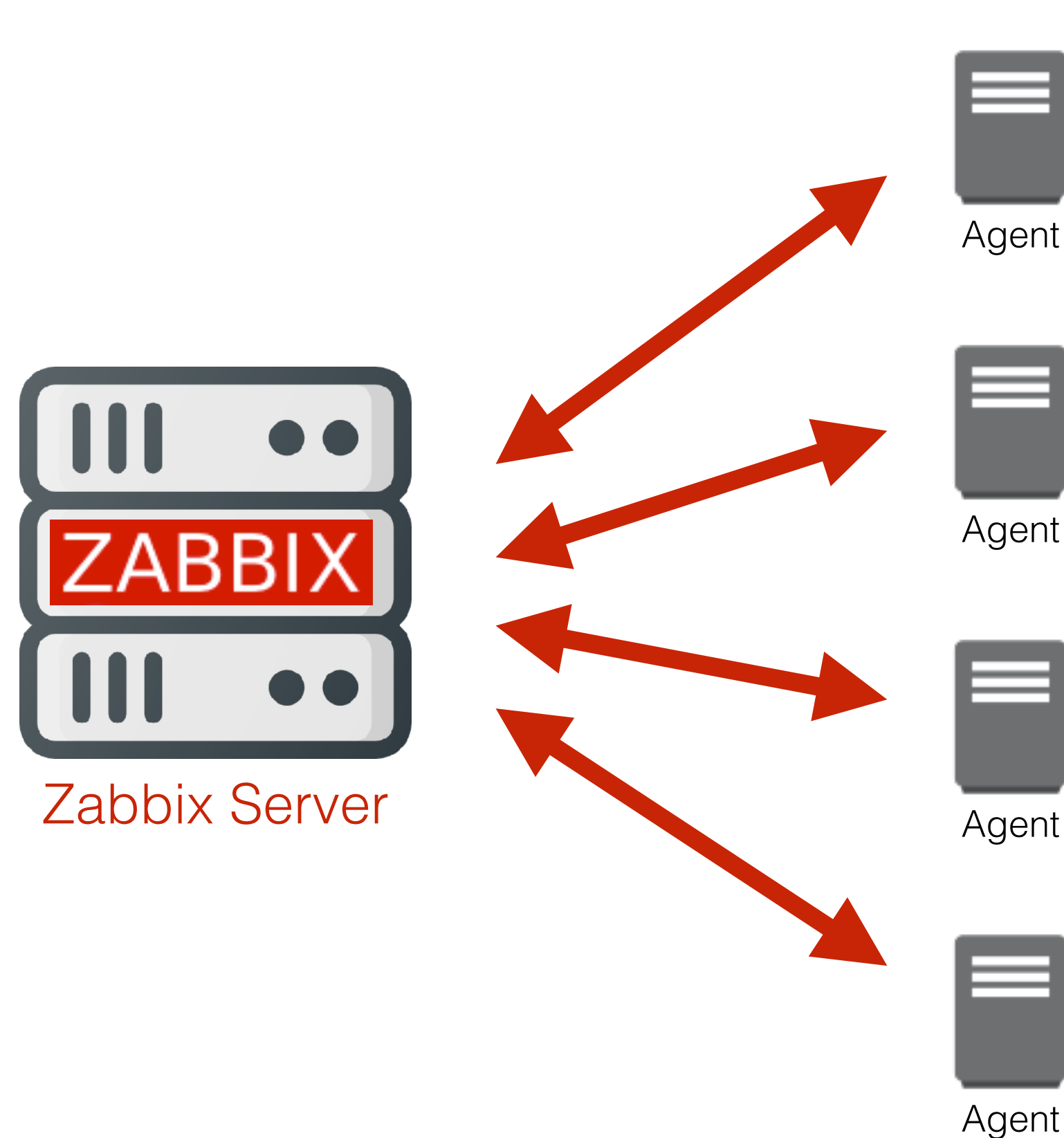
Webhooks over HTTP proxy



Webhooks over HTTP proxy



Restrict available checks on Agent side



Whitelist for MySQL related checks

```
AllowKey=mysql[*]
```

```
DenyKey=*
```

Blacklist to deny all shell scripts

```
DenyKey=system.run[*]
```

Blacklist to deny access to /etc/passwd

```
DenyKey=vfs.file.contents[/etc/passwd,*]
```

Configurable ciphers

Why?

Use cases:

Ability to disable weak ciphers

Ability to have white list of ciphers

Ability to be compliant with internal standards

Override the built-in ciphersuite list for certificate:

TLSCipherCert13 - certificate-based ciphersuite selection criteria for TLS 1.3 (only for OpenSSL 1.1.1 or newer),
TLSCipherCert - certificate-based ciphersuite selection criteria for TLS 1.2/1.3 (for GnuTLS), for TLS 1.2 (OpenSSL).

it override the built-in ciphersuite list for PSK:

TLSCipherPSK13 - PSK-based ciphersuite selection criteria for TLS 1.3 (only for OpenSSL 1.1.1 or newer),
TLSCipherPSK - PSK-based ciphersuite selection criteria for TLS 1.2/1.3 (for GnuTLS), for TLS 1.2 (OpenSSL).

to override the built-in combined ciphersuite list for certificate and PSK:

TLSCipherAll13 - ciphersuite selection criteria for TLS 1.3 (only for OpenSSL 1.1.1 or newer),
TLSCipherAll - ciphersuite selection criteria for TLS 1.2/1.3 (for GnuTLS), for TLS 1.2 (OpenSSL).

Example:

```
TLSCipherCert13=TLS_AES_256_GCM_SHA384
```

Encrypted connection to database



Zabbix Server



Database

ZABBIX

Configure DB connection

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Database port: 0 - use default port

Database name:

User:

Password:

TLS encryption:

TLS key file:

TLS certificate file:

TLS certificate authority file:

With host verification:

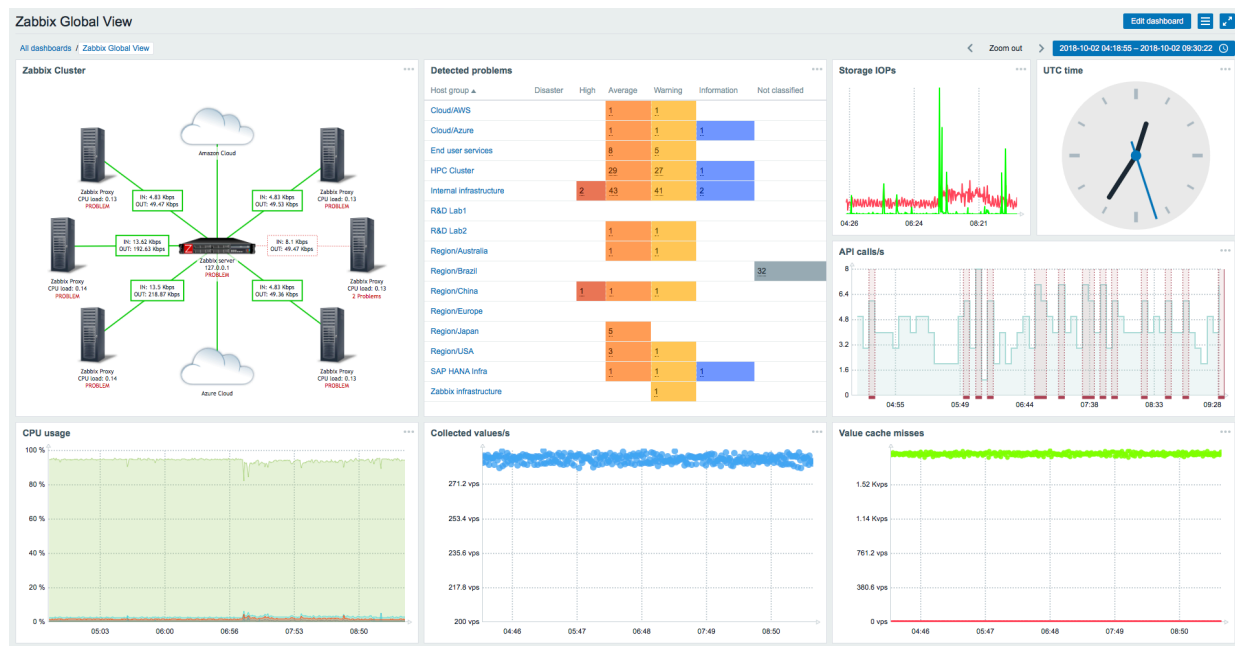
TLS cipher list:

Licensed under [GPL v2](#)

Configurable ciphers



Strong encryption of user password hashes



Zabbix UI








Database

Blowfish instead of MD5

Hash algorithm will be updated on user login

Secret macros

Macros ▾

Macro	Value		Description
{AWS_TOKEN}	 ▾	Token to access AWS API
{SNMP_COMMUNITY}	 ▾	SNMP community for network monitoring
{ZABBIX_URL}	https://zabbix.example.com	 ^	URL of Zabbix UI
{MACRO}	value	 Text	
		 Secret text	

[Add](#)

Use cases

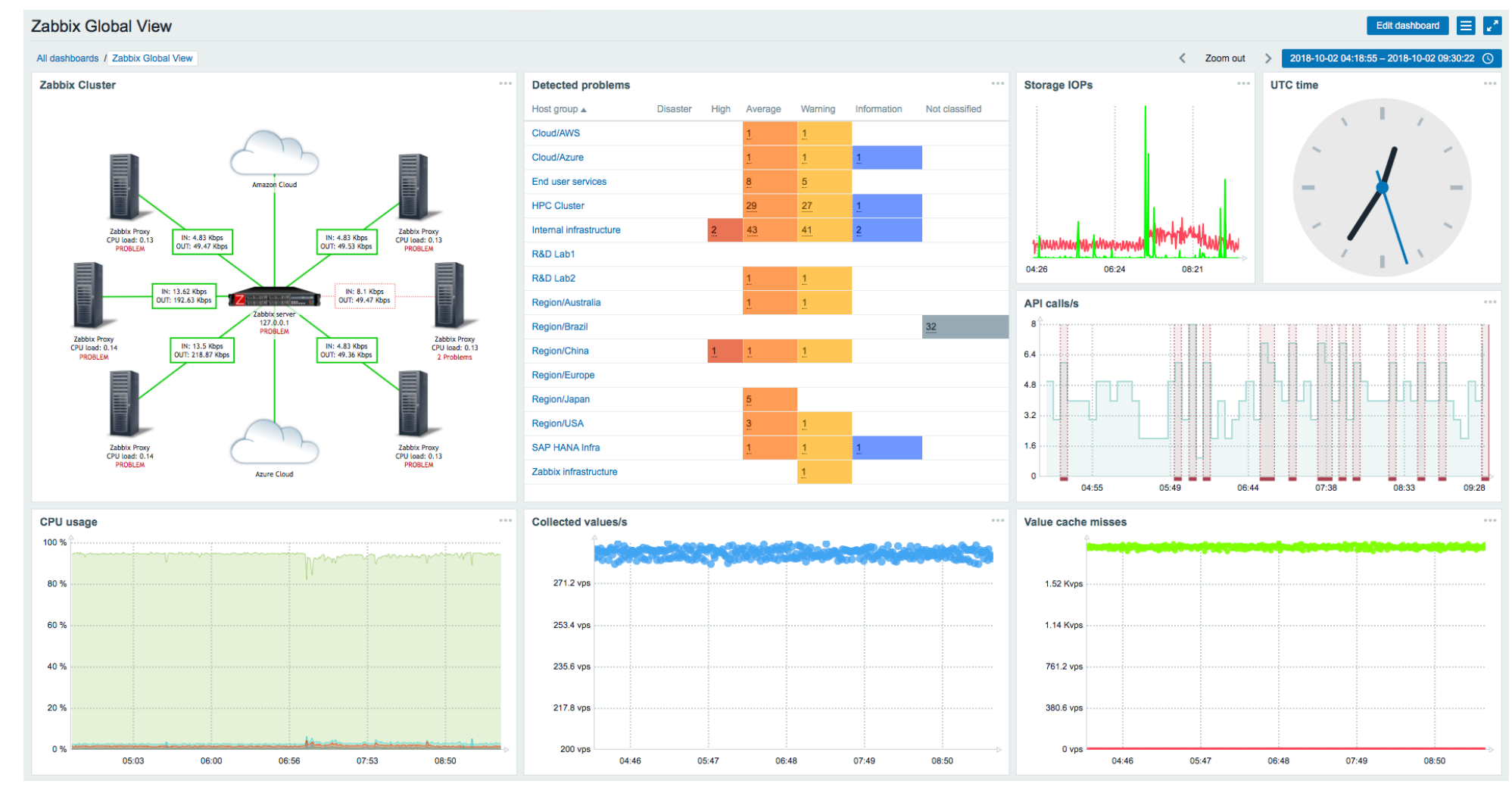
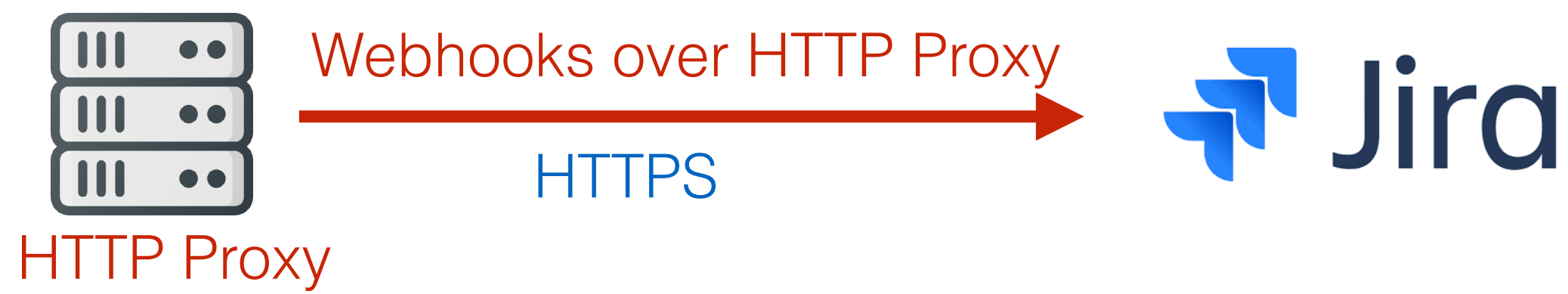
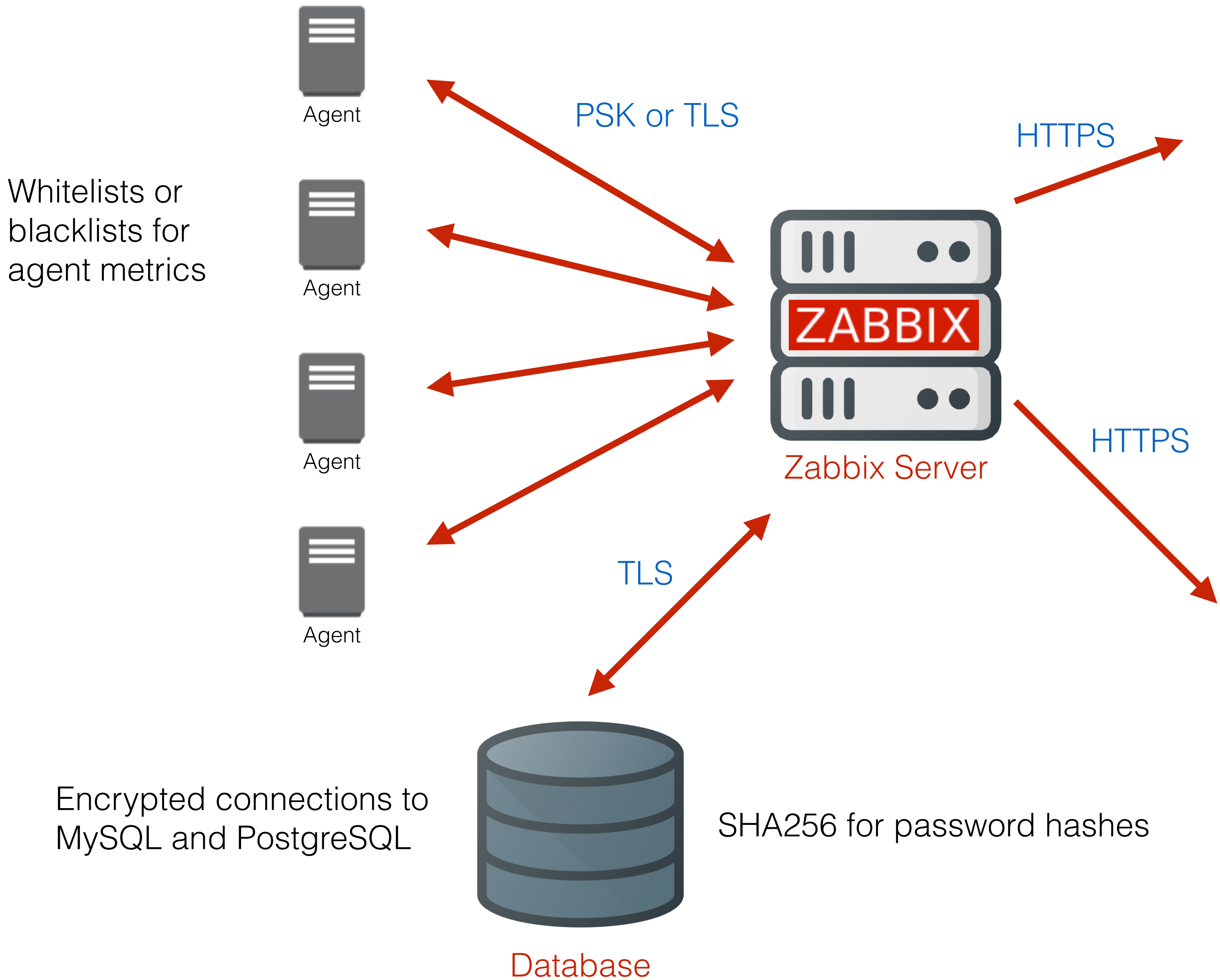
Hide any secrets: passwords, tokens, IDs

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

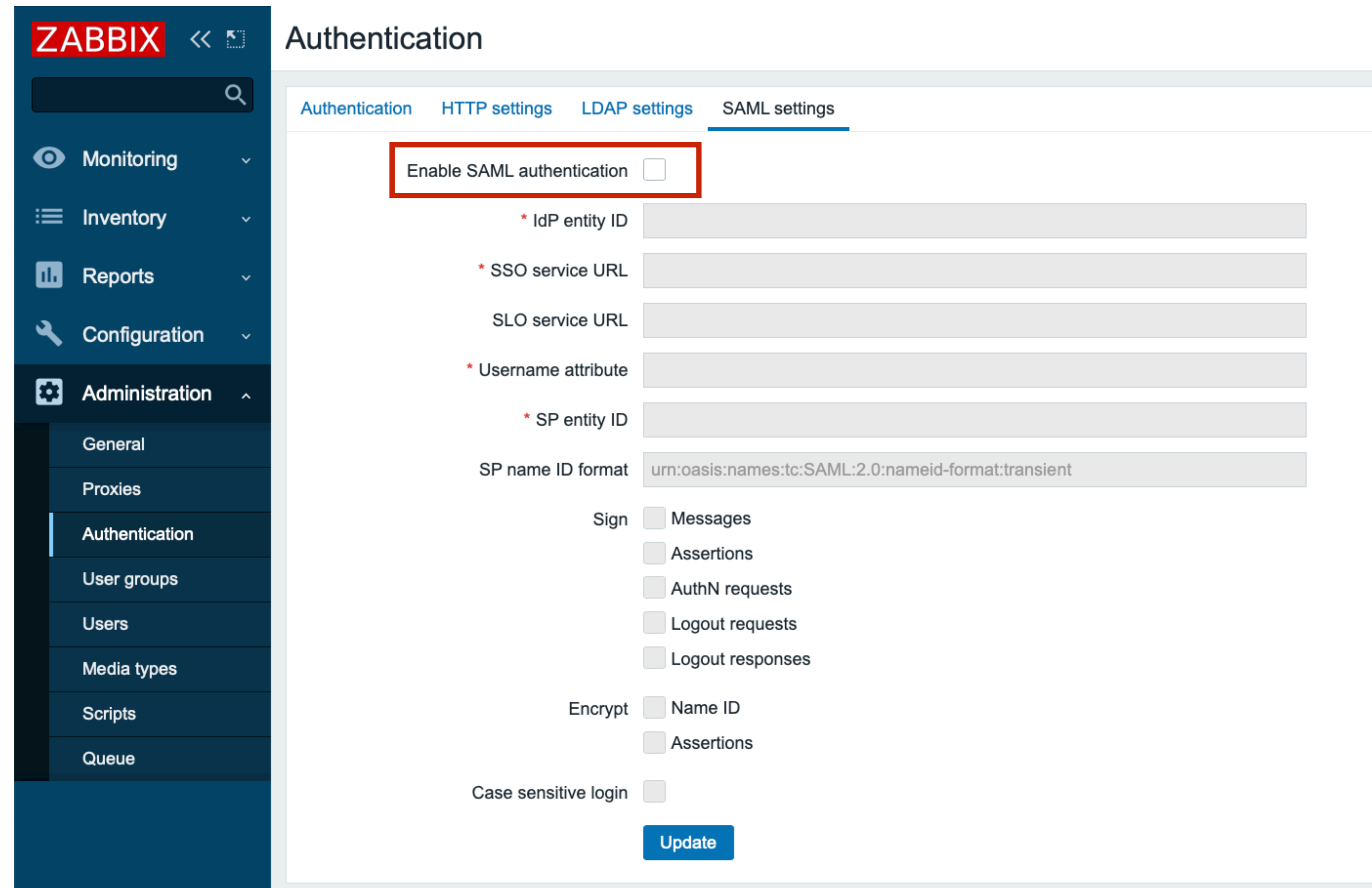


Masking secrets in UI and alerts



SAML authentication for single sign-on

Identity providers



The screenshot shows the ZABBIX web interface for configuring SAML authentication. The left sidebar contains navigation menus for Monitoring, Inventory, Reports, Configuration, and Administration. The main content area is titled 'Authentication' and has tabs for Authentication, HTTP settings, LDAP settings, and SAML settings. The SAML settings page includes a red-bordered checkbox for 'Enable SAML authentication'. Below this are several input fields: '* IdP entity ID', '* SSO service URL', 'SLO service URL', '* Username attribute', '* SP entity ID', and 'SP name ID format' (pre-filled with 'urn:oasis:names:tc:SAML:2.0:nameid-format:transient'). There are also sections for 'Sign' (Messages, Assertions, AuthN requests, Logout requests, Logout responses) and 'Encrypt' (Name ID, Assertions) with checkboxes. A 'Case sensitive login' checkbox and an 'Update' button are at the bottom.





Usability

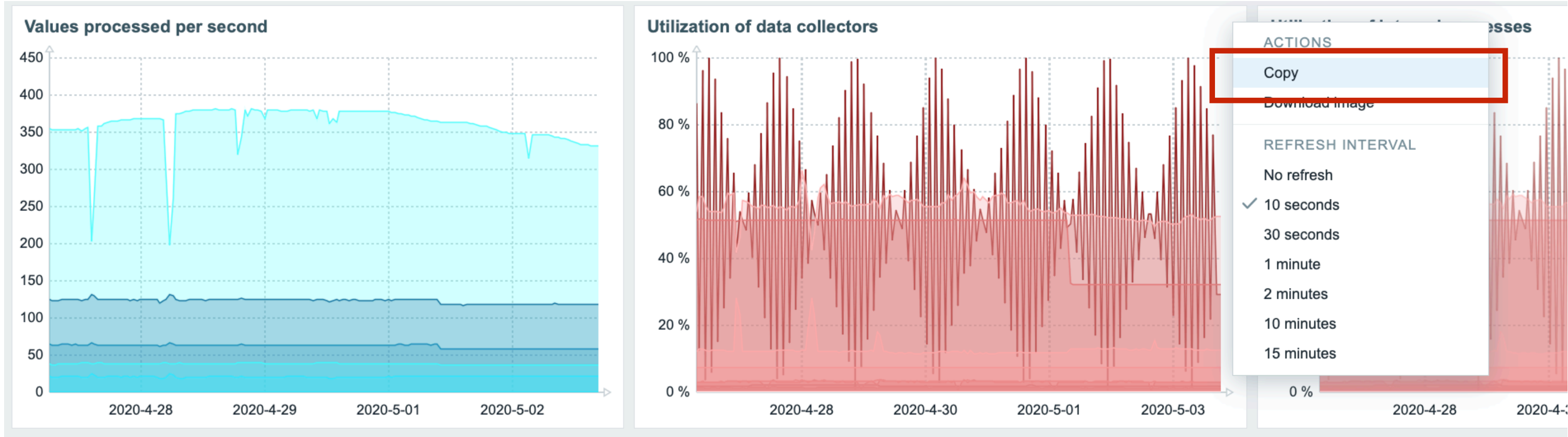
improvements

Optimized for wide screens

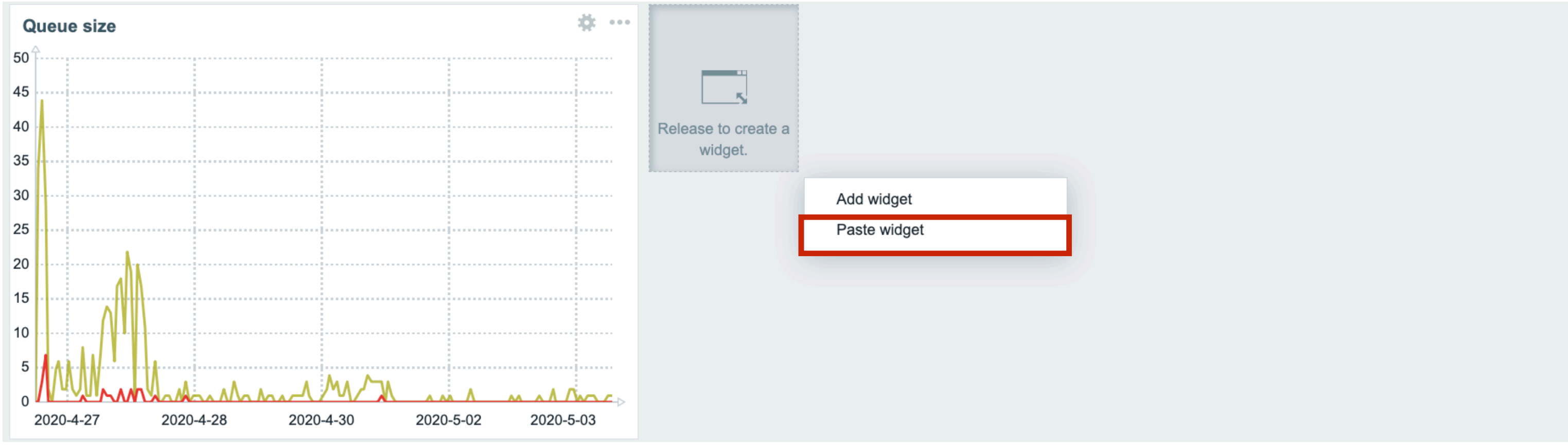
The screenshot displays the Zabbix web interface for the 'Status of Zabbix Server' dashboard. The interface is designed for wide screens, featuring a dark sidebar on the left with navigation options like Monitoring, Inventory, Reports, Configuration, Administration, Support, Share, Help, User settings, and Sign out. The main content area includes a search bar, a breadcrumb trail, and a 'Local time' clock widget. The primary section is 'Zabbix internal problems', which currently shows 'No data found'. Below this are six performance charts: 'Values processed per second' (stacked area chart), 'Utilization of data collectors' (line chart), 'Utilization of internal processes' (line chart), 'Cache usage' (line chart), 'Value cache hits' (area chart), and 'Queue size' (line chart). A vertical red dashed line in all charts marks a specific time point on May 6, 2020. At the bottom, the 'Problems by severity' section is partially visible.

Build dashboards faster

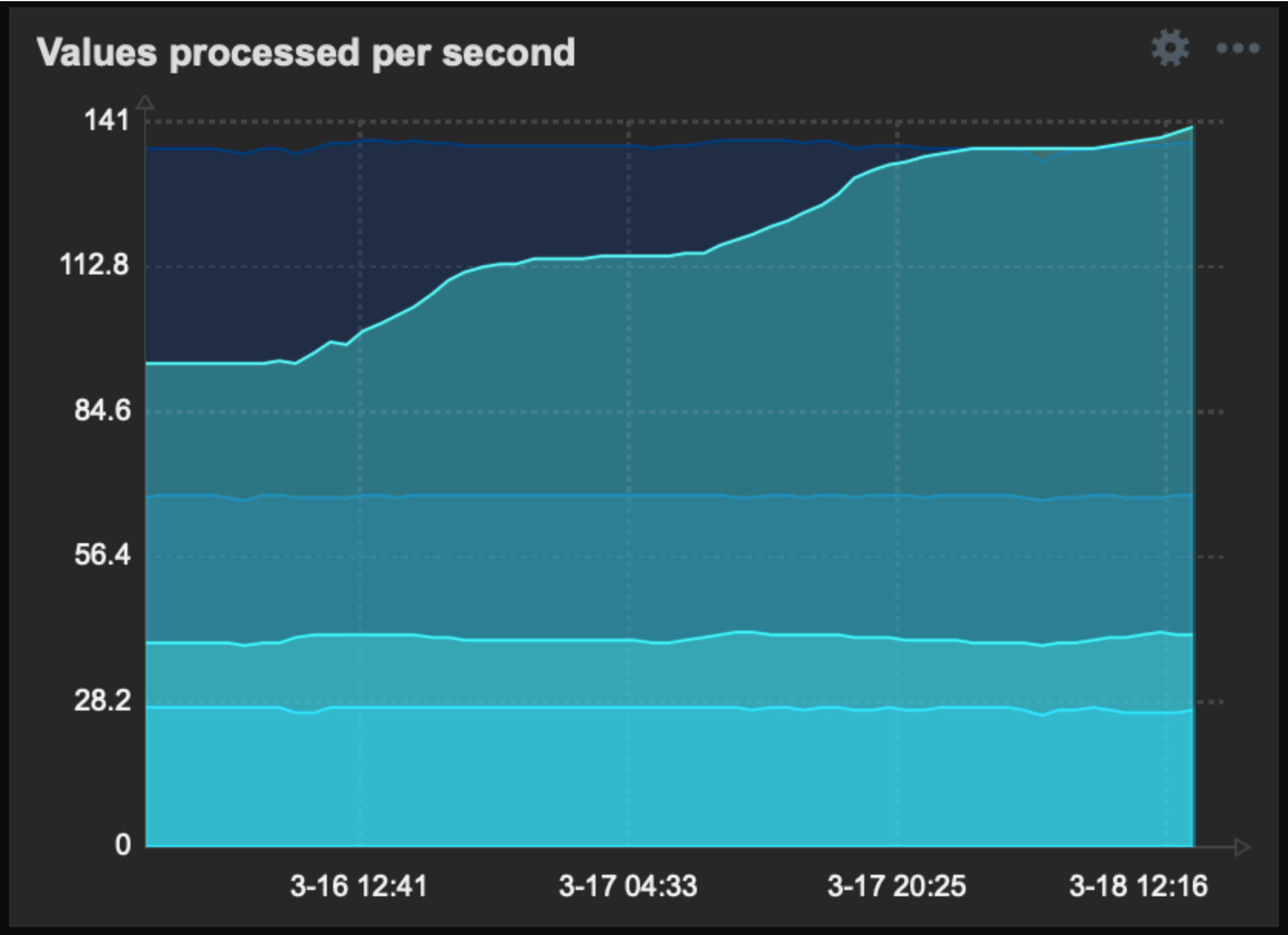
Copy



Paste to the same or different dashboard

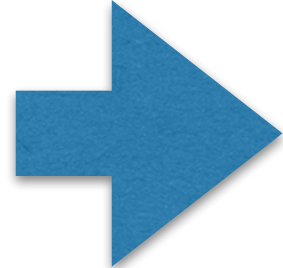


Export graphs as PNG image



A context menu is shown with the following options:

- ACTIONS
 - Download image** (highlighted with a red border)
- REFRESH INTERVAL
 - No refresh
 - ✓ 10 seconds
 - 30 seconds
 - 1 minute
 - 2 minutes
 - 10 minutes
 - 15 minutes



Filter by tags for some widgets

Problem by severity & Problem hosts

Edit widget ✕

Type: **Problems by severity** Show header

Name:

Refresh interval: **Default (1 minute)**

Host groups: Select

Exclude host groups: Select

Hosts: Select

Problem:

Severity: Not classified Warning High Information Average Disaster

Tags: **And/Or** Or

Contains Equals Remove

Contains Equals Remove

[Add](#)

Show:

Layout:

Show operational data:

Show suppressed problems:

Hide groups without problems:

Apply Cancel



UI modules

Create new menu entries

Create new pages

Extend existing functionality

Use and share 3rd party extensions

Permission control

Example:

Structure

```
modules/
  demo_module/
    manifest.json
    Module.php
    actions/
      DemoReportAction.php
    views/
      demo.report.php
  ...
```

manifest.json

```
{
  "manifest_version": 1.0,
  "id": "demo.report",
  "version": "1.0",
  "name": "Custom report",
  "namespace": "Demo",
  "author": "Zabbix",
  "url": "http://www.zabbix.com",
  "description": "Demo report module",
  "actions": {
    "demo.report": {
      "class": "DemoReportAction",
      "view": "demo.report"
    }
  }
}
```

module.php

```
<?php declare(strict_types = 1);

namespace Modules\Demo;

use APP;
use Core\CModule as BaseModule;

class Module extends BaseModule {

    public function init(): void {
        (APP::Component()->get('menu.main'))
            ->find(_('Reports'))
            ->add('Custom report', [
                'action' => 'demo.report'
            ]);
    }
}
```


List of monitored devices

Monitoring->Hosts

Name ▲	Interface	Availability	Tags	Problems	Status	Latest data	Problems	Graphs	Screens	Web
AWS N30	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux Region: us-east-1 Service: Oracle Cluster	1	Enabled	Latest data	Problems 1	Graphs 14	Screens 2	Web
AWS N34	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1	Enabled	Latest data	Problems 2	Graphs 14	Screens 2	Web
AWS N90	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1 1	Enabled	Latest data	Problems 3	Graphs 14	Screens 2	Web
AZ M08	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1	Enabled	Latest data	Problems 2	Graphs 14	Screens 2	Web
AZ M10	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1	Enabled	Latest data	Problems 2	Graphs 14	Screens 2	Web
AZ M18	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1	Enabled	Latest data	Problems 2	Graphs 14	Screens 2	Web
Linux001	127.0.0.1: 10050	ZBX SNMP JMX IPMI	OS: Linux	1 1	Enabled	Latest data	Problems 2	Graphs 14	Screens 2	Web

No more **Monitoring->WEB** and **Monitoring->Graphs**
 Easy navigation to host related resources

List of monitored devices

Advanced filtering options

Hosts
[Filter](#)

Name

Host groups Select

IP

DNS

Port

Severity Not classified Warning High
 Information Average Disaster

Status Any Enabled Disabled

Tags And/Or Or

Contains Equals Remove

[Add](#)

Show hosts in maintenance Show suppressed problems

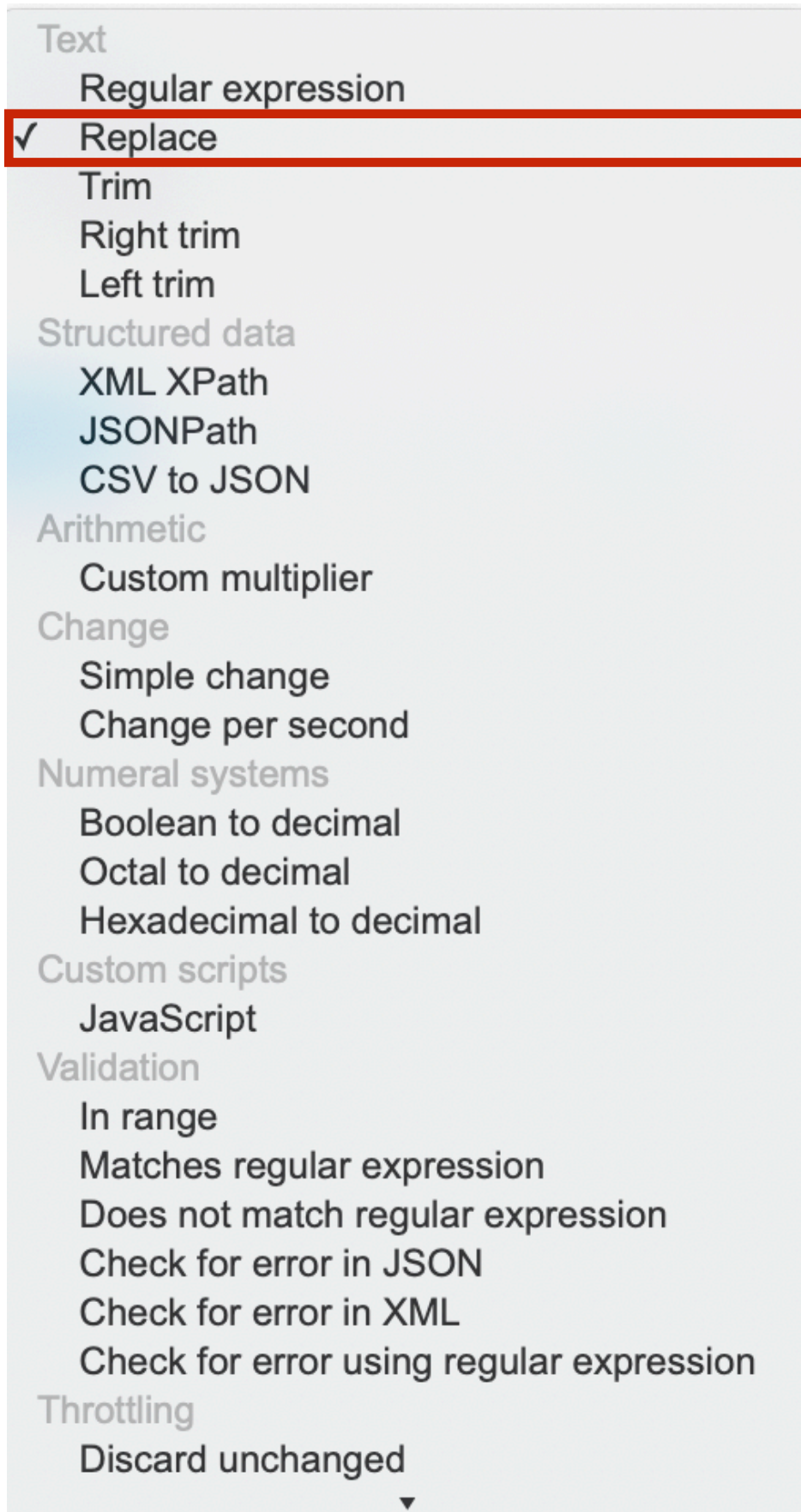
Apply Reset

Name ▲	Interface	Availability	Tags	Problems	Status	Latest data	Problems	Graphs	Screens	Web
AWS N30	127.0.0.1: 10050	ZBX SNMP JMX IPMI	Service: Oracle Cluster OS: Linux Region: us-east-1	1	Enabled	Latest data	Problems 1	Graphs 14	Screens 2	Web
Linux902	127.0.0.1: 10050	ZBX SNMP JMX IPMI	Service: MySQL OS: Linux	1	Enabled	Latest data	Problems 1	Graphs 14	Screens 2	Web

Displaying 2 of 2 found

New preprocessing operator: **Replace**

Configuration->Items



Item Preprocessing

Preprocessing steps	Name	Parameters
1:	Replace	Up 0
2:	Replace	Down 1

[Add](#)

[Update](#) [Clone](#) [Test](#) [Delete](#) [Cancel](#)

Typical use cases

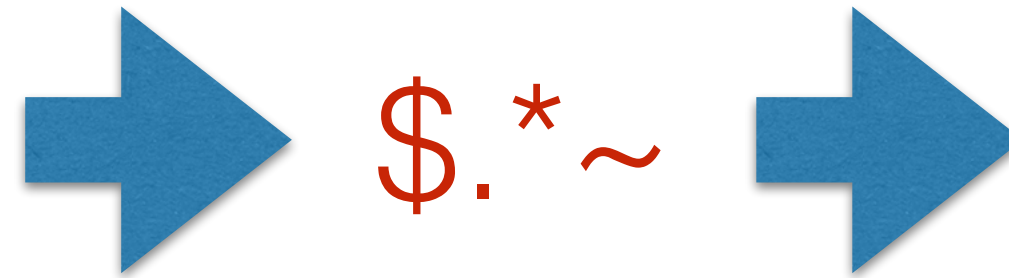
- mappings (text -> numeric, numeric -> text)
- removing characters and strings
- replacing characters and strings
- in many cases easier than dealing with regular expressions!

New operator for JSONPath: ~

It returns property names of matching elements

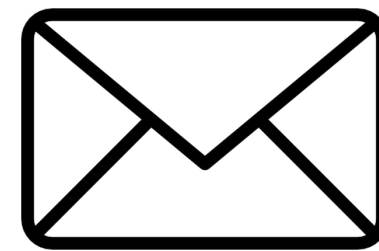
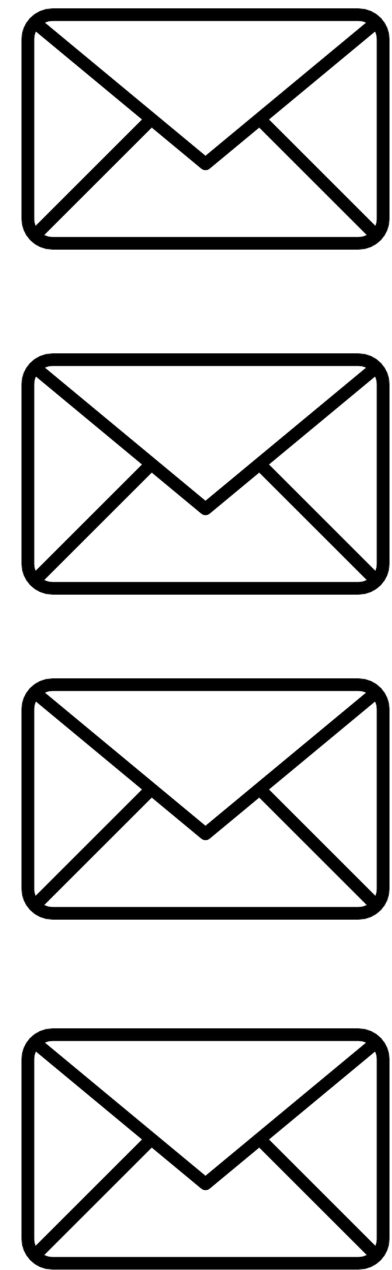


```
{
  "consul": [],
  "content": [
    "2020.01.05",
    "golang"
  ],
  "login": [
    "2019.11.02",
    "java"
  ],
  "mail": [
    "2020.01.02",
    "golang"
  ]
}
```

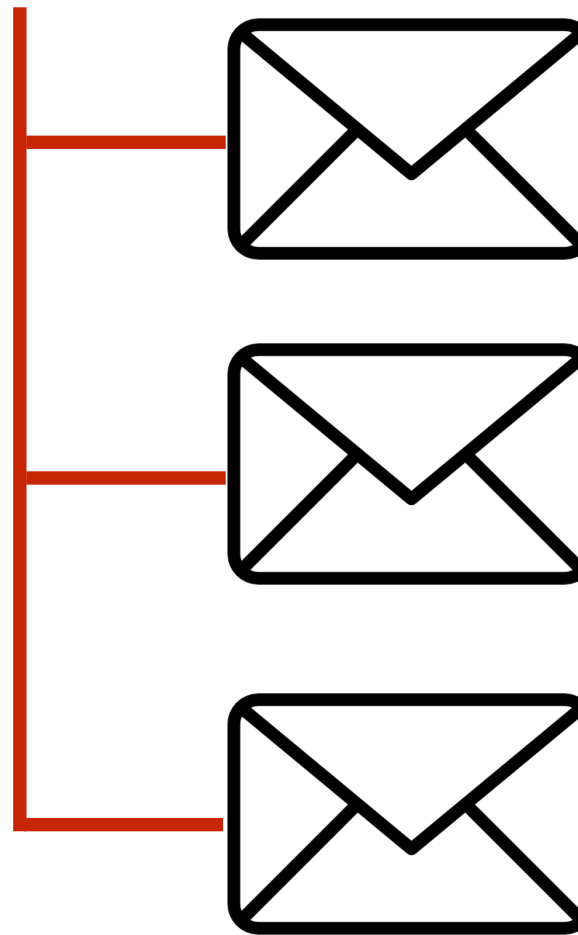


```
[
  "consul",
  "content",
  "login",
  "mail"
]
```

Threading for email notifications



Grouped by event ID + media



User macros for IPMI user name and password

Hosts

All hosts / AWS N30 Enabled ZBX SNMP JMX IPMI Applications 16 Items 84 Triggers 51 Graphs 14 Discovery rules 2 Web scenarios

Host Templates IPMI Tags Macros Inventory Encryption

Authentication algorithm

- Default
- None
- MD2
- MD5
- Straight
- OEM
- RMCP+

Privilege level

- Callback
- User
- Operator
- Admin
- OEM

Username {IPMI.NAME}

Password {IPMI.PASSWORD}

Update Clone Full clone Delete Cancel

Use it with secret macros for extra security!

Mass update of user macros for hosts & templates

Host Templates IPMI Tags **Macros** Inventory Encryption

Macros **Add** Update Remove Remove all

Macro	Value	Description
<input type="text" value="{SERVICE}"/>	<input type="text" value="PostgreSQL cluster"/> T v	<input type="text" value="Our primary production data store"/> Remove

[Add](#)

Update existing

Update Cancel

Host Templates IPMI Tags **Macros** Inventory Encryption

Macros Add **Update** Remove Remove all

Macro	Value	Description
<input type="text" value="{SERVICE}"/>	<input type="text" value="PostgreSQL cluster"/> T v	<input type="text" value="Our primary production data store"/> Remove

[Add](#)

Add missing

Update Cancel

Mass update of user macros for hosts & templates

Host Templates IPMI Tags **Macros** Inventory Encryption

Macros Add Update **Remove** Remove all

Macro

[Remove](#)

[Add](#)

Except selected


Update Cancel

Host Templates IPMI Tags **Macros** Inventory Encryption

Macros Add Update Remove **Remove all**

I confirm to remove all macros

Update Cancel



Message templates for media types

Templates for different message types

Email media type

Media types

Media type Message templates Options

Message type	Template	Actions
Problem	Problem started at {EVENT.TIME} on {EVENT.DATE} Pro...	Edit Remove
Problem recovery	Problem has been resolved at {EVENT.RECOVERY.TIME}...	Edit Remove
Problem update	{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem...	Edit Remove
Discovery	Discovery rule: {DISCOVERY.RULE.NAME} Device IP: {D...	Edit Remove
Autoregistration	Host name: {HOST.HOST} Host IP: {HOST.IP} Agent port:...	Edit Remove
Add		

[Add](#) [Cancel](#)

Message template

Message template

Message type

Subject

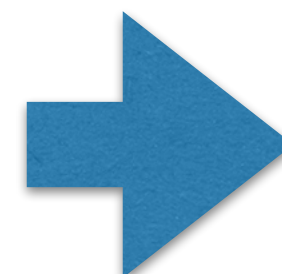
Message

[Update](#) [Cancel](#)

Configure message format in on place!

Before

The screenshot shows the 'Actions' configuration page in a dark theme. The 'Operations' tab is selected. The 'Default operation step duration' is set to '1h'. The 'Default subject' is 'Problem: {EVENT.NAME}'. The 'Default message' field contains a multi-line template: 'Problem started at {EVENT.TIME} on {EVENT.DATE}', 'Problem name: {EVENT.NAME}', 'Host: {HOST.NAME}', 'Severity: {EVENT.SEVERITY}', 'Original problem ID: {EVENT.ID}', and '{TRIGGER.URL}'. The 'Pause operations for suppressed problems' checkbox is checked. Below, there is a table for 'Operations' with columns 'Steps', 'Details', 'Start in', 'Duration', and 'Action'. A 'New' link is under the 'Steps' column. A note at the bottom states: '* At least one operation, recovery operation or update operation must exist.' There are 'Add' and 'Cancel' buttons at the bottom.



After

The screenshot shows the 'Actions' configuration page in a light theme. The 'Operations' tab is selected. The 'Default operation step duration' is set to '1h'. The 'Pause operations for suppressed problems' checkbox is checked. Below, there are three tables for 'Operations', 'Recovery operations', and 'Update operations', each with columns 'Details' and 'Action'. Each table has an 'Add' link under the 'Details' column. A note at the bottom states: '* At least one operation must exist.' There are 'Add' and 'Cancel' buttons at the bottom.

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

Example 1

```
shell> cat test.js  
  
return Math.log(value)  
  
shell> zabbix_js -s test.js -p 10  
  
2.302585092994046
```

Example 2

```
shell> zabbix_js -s test.js -i my.json # reading input from file
```

Example 3

```
shell> cat test.js  
  
Zabbix.Log(3, value) // use Zabbix.Log(log level, text) fo debug purposes  
  
return Math.log(value)
```

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() }
```



Automation & Discovery

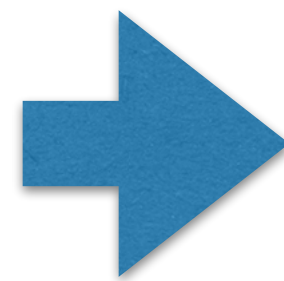
Discovery for JMX counters

New JMX checks

`jmx.get[]`

`jmx.discovery[]`

`jmx.get[]` is similar to the `jmx.discovery[]` item, but it does not turn Java object properties into low-level discovery macro names and therefore can return values without limitations that are associated with LLD macro name generation such as hyphens or non-ASCII characters.



`jmx.get[beans,"com.example:type=*,*"]`

```
[
  {
    "object": "com.example:type=Hello,data-src=data-base,ключ=значение",
    "domain": "com.example",
    "properties": {
      "data-src": "data-base",
      "ключ": "значение",
      "type": "Hello"
    }
  },
  {
    "object": "com.example:type=Atomic",
    "domain": "com.example",
    "properties": {
      "type": "Atomic"
    }
  }
]
```

`jmx.discovery[...]`

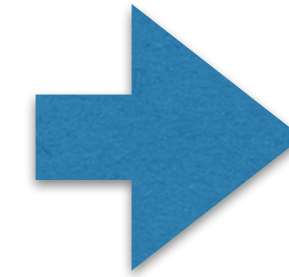
```
[
  {
    "#{#JMXDOMAIN}": "java.lang",
    "#{#JMXTYPE}": "GarbageCollector",
    "#{#JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{#JMXNAME}": "PS Scavenge"
  }
]
```

Discovery for Windows perf counters

Zabbix Agent and Agent2

`perf_counter.discovery[object]`

`perf_counter_en.discovery[object]`



```
[  
  {"#INSTANCE": "0"},  
  {"#INSTANCE": "1"},  
  {"#INSTANCE": "_Total"}  
]
```

Better ODBC monitoring

Current situation

ODBC monitoring work only with data sources (DSN) configured in `/etc/odbc.ini` file.

Now everything can be part of item key

```
db.odbc.select[MySQL db check,, "DRIVER=mysqla;
SERVER=127.0.0.1;
PORT=3306;UID=zabbix;
PWD=zabbix;DATABASE=master;OPTION=3;"]
```

```
db.odbc.select[MySQL db check,, {$CONN_STRING}]
```

Also, if `connection_string` is used, and `User name` field is not empty, it is appended to the connection string as `UID=<user>`. Similar, `Password` field value is appended to the connection_string as `PWD=<password>`.

Items

All hosts / AWS N30 Enabled ZBX SNMP JMX IPMI Applications 16 Items 84 Triggers 51 Graphs 14 Discovery rules 2 Web scenarios

Item Preprocessing

Name

Type Database monitor

Key db.odbc.get[users, "DRIVER=mysqla;SERVER=127.0.0.1; PORT=3306;DATABASE Select

User name {\$DB_USER}

Password {\$DB_PWD}

SQL query select * from users

Type of information Text

Update interval 1h

Type	Interval	Period	Action
Flexible Scheduling	50s	1-7,00:00-24:00	Remove

Add

History storage period Do not keep history Storage period 90d

Can be secret macros!


Discovery of IPMI sensors

ipmi.get

Typical use cases

Simpler templates

```
[{
  "id": "SubTemp12",
  "name": "(7.1).SubTemp12",
  "sensor": {
    "type": "1",
    "text": "temperature"
  },
  "reading": {
    "type": "1",
    "text": "threshold"
  },
  "state": {
    "state": "3",
    "text": "lower Critical - going high"
  },
  "value": "32",
  "units": "C",
  "threshold": {
    "low": {
      "non_crit": "48",
      "crit": "32",
      "non_recover": "16"
    },
    "up": {
      "non_crit": "112",
      "crit": "144",
      "non_recover": "160"
    }
  }
},
{
  "id": "1.8V Switch",
  "name": "(7.1).1.8V Switch",
  "sensor": {
    "type": "2",
    "text": "voltage"
  },
  "reading": {
    "type": "2",
    "text": "discrete_usage"
  },
  "state": {
    "state": "1",
    "text": "transition to active"
  }
}]
```



Test item from UI

For hosts and templates

Test item

Get value from host

Host address Port

Proxy (no proxy)

Value <!DOCTYPE html> Time now

Previous value Prev. time

End of line sequence LF CRLF

Preprocessing steps

Name	Result
1: Regular expression	Zabbix
2: Matches regular expression	Zabbix
3: Replace	1

Result

Result converted to Numeric (unsigned)	1
Result with value map applied	Up

Test item

Get value from agent failed: cannot connect to [[192.168.3.2]:10050]: [111] Connection refused

Get value from host

Host address 192.168.3.2 Port 10050

Proxy (no proxy)

Value 0.000000 Time now

Previous value Prev. time

End of line sequence LF CRLF

Do not forget to test media types too!

Media types Create media type Import Filter

<input type="checkbox"/> Name ▲	Type	Status	Used in actions	Details	Action
<input type="checkbox"/> Discord	Webhook	Enabled			Test
<input type="checkbox"/> Mattermost	Webhook	Enabled			Test
<input type="checkbox"/> Opsgenie	Webhook	Enabled			Test
<input type="checkbox"/> Pushover	Webhook	Enabled			Test

Displaying 4 of 4 found

Support of **user macros**
for host prototypes

User macros for host prototypes

The screenshot displays the Zabbix web interface for configuring host prototypes. The breadcrumb trail is: All hosts / Ild-discoverer-host Enabled [ZBX] [SNMP] [JMX] [IPMI] Discovery list / host.json Item prototypes Trigger prototypes Graph prototypes Host prototypes 1. The 'Macros' tab is selected in the navigation menu. Below the navigation, there are two tabs: 'Host prototype macros' (active) and 'Inherited and host prototype macros'. A table lists the macro configuration:

Macro	Value	Description	
{ \$MACRO }	value	description	Remove

Below the table, there is an 'Add' link and a set of action buttons: Update, Clone, Delete, and Cancel.

Use LLD macros in macro value and description!

Support of **Float64** data types

Float64

Benefits

Compatible with Float64 returned by Prometheus

Execute to upgrade existing installation:

MySQL: `database/mysql/double.sql`

PostgreSQL: `database/postgresql/double.sql`

MySQL: `database/oracle/double.sql`



Scalability

improvements

Zabbix UI

Ready to handle
Millions of devices

Improvements made

No drop-downs for host selections anymore, replaced with host search control

Hardcoded maximum size of the Overview grid

Redesigned Monitoring->Hosts->Graphs (multiselection of graphs, displaying of all graphs, pattern matching)

Introduced paging whenever possible (Monitoring->Hosts->Web)

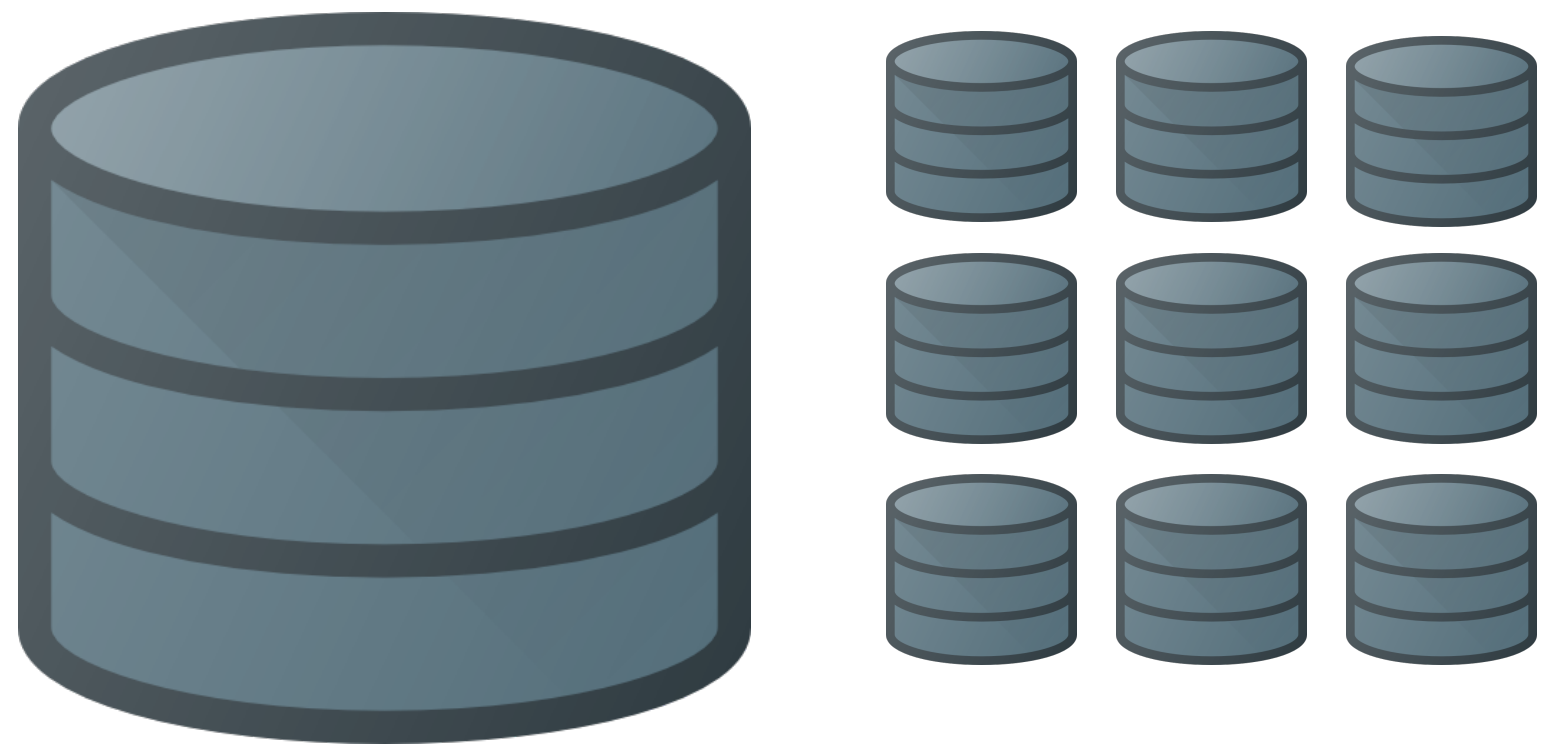


Compression for efficiency

TimescaleDB = PostgreSQL + Extension



TimescaleDB



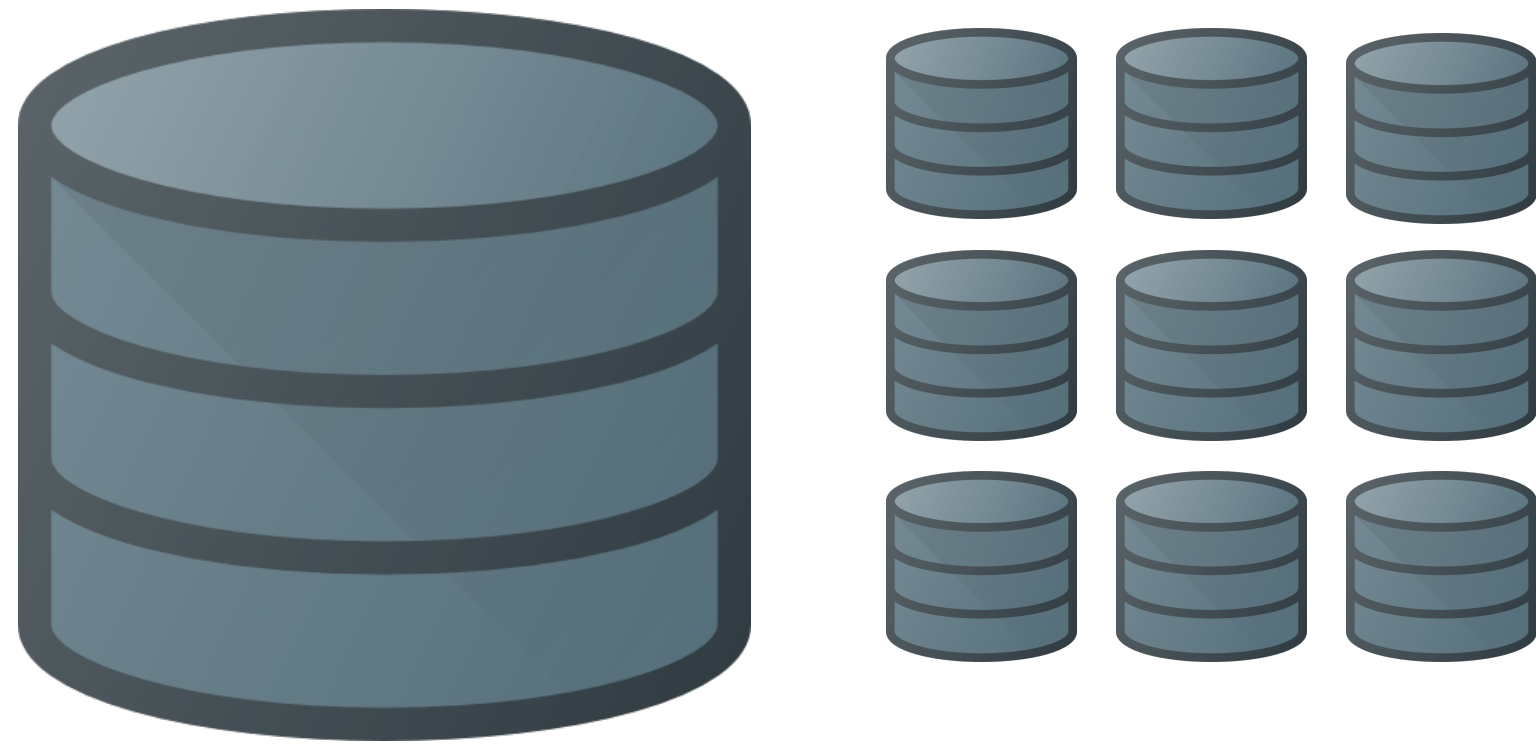
Advantages

- Automatic partitioning
- Zabbix manages removal of old data
- Performance oriented DB
- **Compression!**



TIMESCALE

TimescaleDB



Administration->General->Housekeeping

History and trends compression

Enable compression

* Compress records older than

Lower storage cost



Older than 7 days,
Compressed, read-only



TIMESCALE

Numbers from a production setup

Zabbix 4.x: 355GB

Zabbix 5.0: 43GB

SNMP setting on host **interface** level

Item configuration

- Zabbix agent
- Zabbix agent (active)
- Simple check
- √SNMP agent** One SNMP type
- SNMP trap
- Zabbix internal
- Zabbix trapper
- Zabbix aggregate
- External check
- Database monitor
- HTTP agent
- IPMI agent
- SSH agent
- TELNET agent
- JMX agent
- Calculated
- Dependent item

Host configuration

* Host name

Visible name

* Groups

* Interfaces	Type	IP address	DNS name	Connect to	Port
^	SNMP	<input type="text" value="127.0.0.1"/>	<input type="text"/>	<input checked="" type="radio"/> IP <input type="radio"/> DNS	<input type="text" value="161"/>

* SNMP version

Context name

Security name

Security level

Use bulk requests

Why?

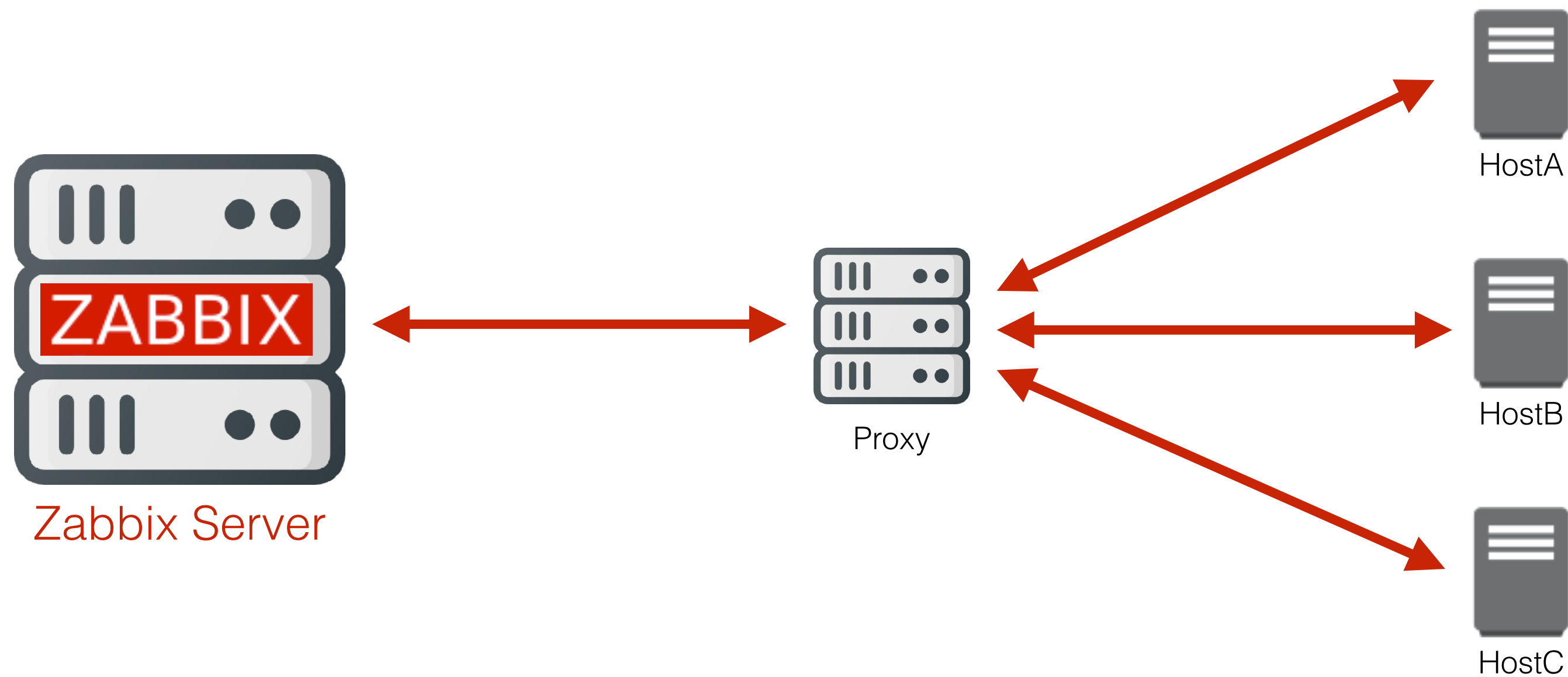
Simplify templates, **one template instead of three**

Easy to manage: SNMP related parameters on interface level

Easy to switch from SNMPv2 -> SNMPv3

Availability monitoring will
respect proxy availability

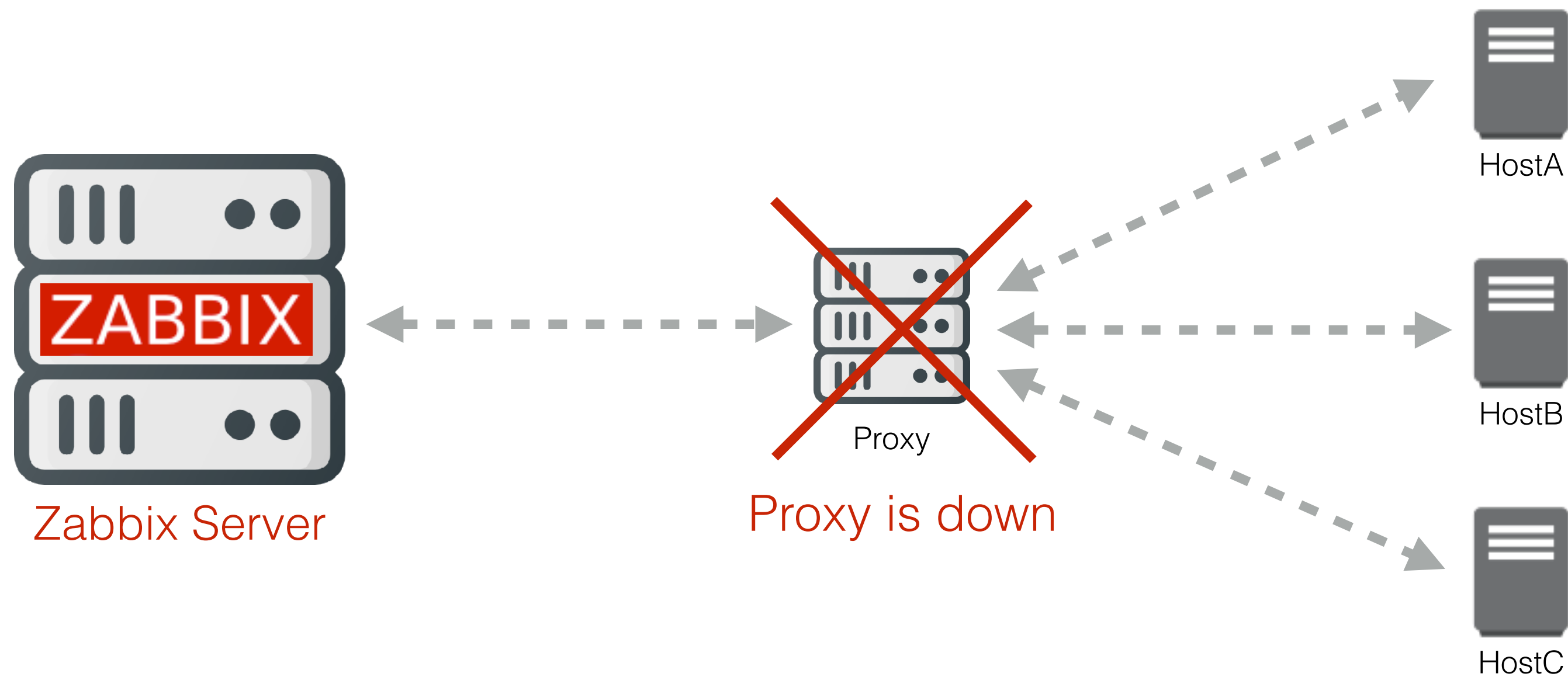
Monitoring host availability



HostA is not available

`{HostA:item.nodata(1m)}=1`

Monitoring host availability



Zabbix 4.x

HostA is not available
HostB is not available
HostC is not available
Proxy is not available

Zabbix 5.0

Proxy is not available

Monitoring host availability

`{HostA:item.nodata(1m)}=1` # respects proxy availability

`{HostA:item.nodata(1m, strict)}=1` # strict check, does not respect proxy availability



Manage LLD rules

globally

Filter for discovery rules

Discovery rules Create discovery rule

All hosts / confluence Enabled ZBX SNMP JMX IPMI Applications 10 Items 114 Triggers 21 Graphs 14 Discovery rules 3 Web scenarios Filter

Host groups: type here to search Type: all

Hosts: confluence Update interval:

Name: Keep lost resources period:

Key:

<input type="checkbox"/>	Host	Name ▲	Items	Triggers	Graphs	Hosts	Key	Interval	Type	Status	Info
<input type="checkbox"/>	confluence	TCP port discovery: Discover TCP ports	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	tcp.port.discover	60	Zabbix agent	Enabled	
<input type="checkbox"/>	confluence	Template_File_System_Discovery: File system discovery	Item prototypes 10	Trigger prototypes 2	Graph prototypes 2	Host prototypes	vfs.fs.discovery	10m	Zabbix agent	Enabled	
<input type="checkbox"/>	confluence	Template_Network: Network discovery	Item prototypes 13	Trigger prototypes	Graph prototypes 1	Host prototypes	net.if.discovery	10m	Zabbix agent	Enabled	

Displaying 3 of 3 found

Useful for: troubleshooting (find all not supported or disabled), mass operations

Ability to
unacknowledge event

Unacknowledge it!

Useful for

fixing mistakes

better workflow between various teams

Update problem

Problem: Low CPU utilization on host machines

Message: Sorry, it was incorrectly acknowledged! This issue should be fixed by DevOps team.

Time	User	User action	Message
2020-05-12 12:10:33	John	✓	I'm working on this problem. To be resolved in 10-15 minutes.

Scope: Only selected problem Selected and all other problems of related triggers 1 event

Change severity: Not classified Information Warning Average High Disaster

Unacknowledge

Close problem

* At least one update operation or message must exist.

Update Cancel



Overrides for LLD rules

Discovery of filesystems

ZABBIX Central NOC

Monitoring

Inventory

Reports

Trigger prototypes

Create trigger prototype

All templates / Template OS Linux / Discovery list / Mounted filesystem discovery / Item prototypes 5 / **Trigger prototypes 2** / Graph prototypes 1 / Host prototypes

<input type="checkbox"/>	Severity	Name ▲	Operational data	Expression	Create enabled	Discover	Tags
<input type="checkbox"/>	Warning	Free disk space is less than 20% on volume {#FSNAME}		{Template OS Linux:vfs.fs.size[{#FSNAME},pfree].last(0)}<20	Yes	Yes	
<input type="checkbox"/>	Warning	Free inodes is less than 20% on volume {#FSNAME}		{Template OS Linux:vfs.fs.inode[{#FSNAME},pfree].last(0)}<20	Yes	Yes	

Displaying 2 of 2 found

Special treating of Oracle related filesystems

Override

Name: Special processing of Oracle directories

Stop processing next overrides if matches:

Filters: Type of calculation: And/Or

Label	Macro	Regular expression	Action
A	{#FSNAME}	matches oradata	Remove

Operations: Condition

Add Cancel

New operation

Object: Trigger prototype

Condition: contains Free disk space

Create enabled: Original

Discover: Original

Severity: Not classified Information Warning Average High Disaster

Tags: Original

Add Cancel

Do not discover temporary filesystems

Override

* Name

Stop processing next overrides if matches

Filters

Type of calculation

Label	Macro		Regular expression	Action
A	<input type="text" value="{#FSNAME}"/>	<input type="text" value="matches"/>	<input type="text" value="tmp"/>	Remove

[Add](#)

Operations

Condition

Trigger prototype equals

Item prototype equals

[Add](#)

New operation

Object

Condition

Create enabled Original

Discover Yes No

Update interval Original

History storage period Original

Trend storage period Original

[Add](#) [Cancel](#)

New macros

ZBXNEXT-1797 support of macro {HOST.ID} in notifications

Can be used to build URLs to Zabbix UI. For example, Latest data:
`{ZABBIX.URL}/zabbix.php?action=latest.view&filter_set=1&filter_hostids%5B0%5D={HOST.ID}`

ZBXNEXT-5369 support of macro {EVENT.TAGSJSON} in notifications

Easier to pass all tags to webhooks

ZBXNEXT-252 support of macro {EVENT.DURATION} in notifications.

Recovery subject “Resolved in **5m**: Service Nginx is down.”

Other improvements

ZBXNEXT-5848 increased size of acknowledge messages to 4K (was 256)

ZBXNEXT-5690 added support of LIBSSH to support newer platforms like RHEL 8

ZBXNEXT-5825 support of Elasticsearch 7.x (7.4, 7.6)

ZBXNEXT-5720 Latest data displays data if filter is not set

ZBXNEXT-1561 increased zabbix_sender time resolution to nanoseconds

ZBXNEXT-1234 Monitoring->Latest data: show data if filter is empty

ZBXNEXT-5734 Base64 processing in JavaScript, functions atob() and btoa()

ZBXNEXT-5604 Do not log system.run[] for local use

ZBXNEXT-4584 New API method to get auditlog

ZBXNEXT-5851 Remote monitoring of versions of Zabbix components

And more

ZBXNEXT-1989 Increased size of item key to 2048 characters (was 255)

ZBXNEXT-3940 Ability to flush SNMP cache, SNMPv3 context changes

ZBXNEXT-5829 Faster hash function for internal operations

ZBXNEXT-2081 Documented how to do filtering for vmware.event monitoring

ZBX-15914 Improved consistency of map labels

Removing legacy to make a better product, faster

ZBXNEXT-5697 No support of Internet Explorer 11 anymore

ZBXNEXT-5592 Dropped support of IBM DB2 database

ZBXNEXT-5716 mbedTLS (former polarSSL) is no longer supported for encryption.
Only OpenSSL and GnuTSL libraries

Minimum supported version for PHP is now 7.2: safer and more strict code

Upgrade!

Just install new server binaries and front-end files

Upgrade!

Important notes

Upgrading history data to Float64 is optional and may take time

Compressed TimescaleDB data is read-only

PHP 7.2 is required

No DB2 support anymore

Read Upgrade Notes for more details!

https://www.zabbix.com/documentation/5.0/manual/installation/upgrade_notes_500

5.2

ETA:
mid-October, 2020

Roadmap

This page contains an incomplete list of planned functionality that can be updated anytime without prior notice.

V 5.2

V 5.4

V 6.0 LTS

Zabbix 5.2

ETA: Q4, 2020

Smart problem and anomaly detection

- Zabbix Insights
A new set of trigger functions for long term analysis of trend data to enable alerts like "Average number of transactions increased by 24% in September".

Make monitoring more secure

- Roles for more granular user permissions
It will control what parts of UI are available and will also introduce fine-grained management of the availability of Zabbix API methods.
- Support storage of secrets in an external Vault
Zabbix will be able to securely keep all sensitive information (like passwords, SNMP community names, tokens, etc.) in a Vault. It will make Zabbix compliant to the most strict security standards.

Smart problem and anomaly detection

- Zabbix Insights

A new set of trigger functions for long term analysis of trend data to enable alerts like “Average number of transactions increased by 24% in September”.

High availability, performance, and scalability

- Zabbix UI and API to support load balancing
- Clean separation of data collection and trigger processing components

Monitoring of IoT devices

- Support data collection using modbus and MQTT protocols

Make monitoring more secure

- Roles for more granular user permissions

It will control what parts of UI are available and will also introduce fine-grained management of the availability of Zabbix API methods.

- Support storage of secrets in an external Vault

Zabbix will be able to securely keep all sensitive information (like passwords, SNMP community names, tokens, etc.) in a Vault. It will make Zabbix compliant to the most strict security standards.

Usability and user experience

- Support of predefined filters for Problems and Hosts for instant switching from one view to another
- Switching to Yaml as the default format for templates and import/export operations
- Selection of timezone for individual users
- Versioning of the templates to simplify the deployment of enhanced templates

It could be used for monitoring automation when templates are managed externally, for example, stored in an external Git repository.



Thank you!