

Achieving 1M NVPS and Other Little Things



The little things



Agent



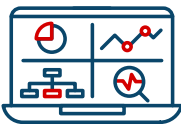
Proxy



Macros



Web monitoring



Dashboards



Configuration

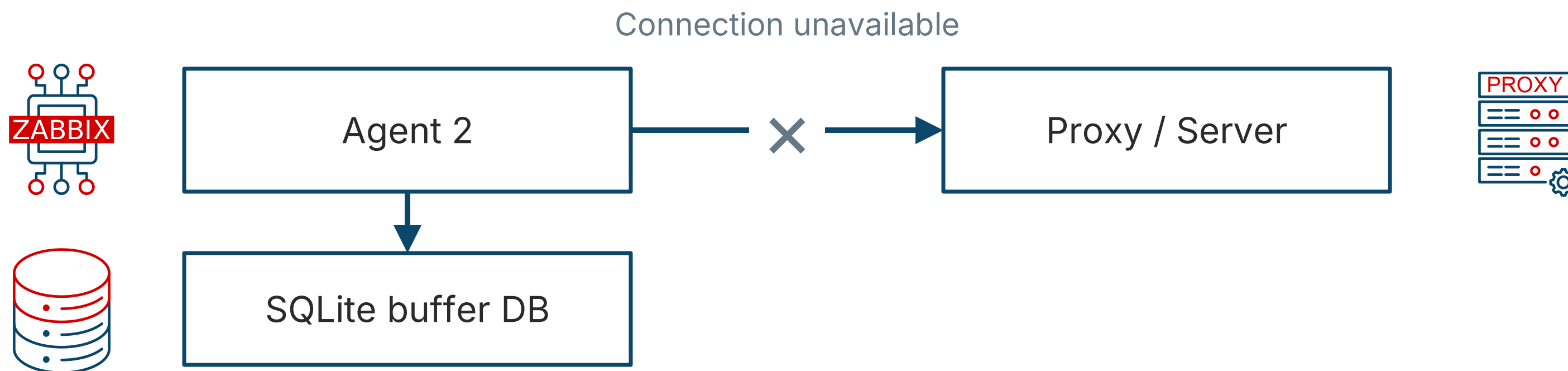
Agents

Persistent buffer on Agent 2

Zabbix 5.0

zabbix_agent2.conf

```
EnablePersistentBuffer=1  
PersistentBufferFile=/var/lib/zabbix/agent2.db  
PersistentBufferPeriod=12h
```



Active-check data can survive a lost connection and an agent restart.

Let the agent normalize its hostname

zabbix_agentd.conf / zabbix_agent2.conf

```
HostnameItem=system.hostname[netbios,lower]
```

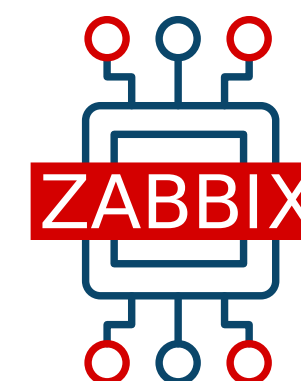
Windows reports

MY-WIN-SERVER

Zabbix hostname

my-win-server

Hostname= must be unset



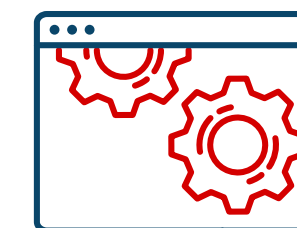
Item aliases

zabbix_agentd.conf / zabbix_agent2.conf

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:..([0-9]+)",,,,1]
```

Item key in Zabbix

zabbix.userid



Also useful for multiple LLD rules

Shorter keys, or several keys for the same underlying check

One key, different platforms

Item key in the template

os.disk.free

Linux: zabbix_agentd.conf / zabbix_agent2.conf

Alias=os.disk.free:vfs.fs.size[/,free]

Windows: zabbix_agentd.conf / zabbix_agent2.conf

Alias=os.disk.free:vfs.fs.size[C:,free]

Several LLD rules, one discovery key

```
zabbix_agentd.conf / zabbix_agent2.conf
```

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Database filesystems

```
vfs.fs.discovery[database]
```

Filter the database mount points

Other filesystems

```
vfs.fs.discovery[other]
```

Apply a different filter and prototypes

Reload user parameters without restarting

Zabbix 6.0

bash · Zabbix agent 2

```
sudo zabbix_agent2 -R userparameter_reload
```

Available in both Zabbix agent and Agent 2

One agent, many host names

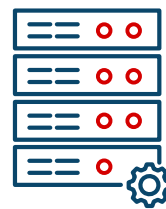
Zabbix 5.2

```
zabbix_agentd.conf / zabbix_agent2.conf
```

```
Hostname=rhel-server01,server01-mariadb,server01-unbound
```

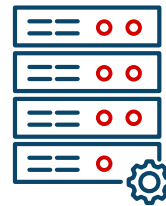


One active agent



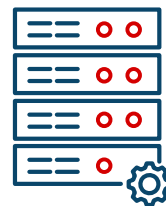
rhel-server01

Operating system



server01-mariadb

MariaDB



server01-unbound

Unbound resolver

Many passive hosts, one agent

Same agent endpoint

192.0.2.10:10050

Template: MySQL by Zabbix agent 2

Different database instances



server01-mariadb-a

{`$MYSQL.DSN`}

tcp://127.0.0.1:**3306**



server01-mariadb-b

{`$MYSQL.DSN`}

tcp://127.0.0.1:**3307**

Agent 2 has a health web endpoint

zabbix_agent2.conf

StatusPort=10055

Status URL

<http://localhost:10055/status>

Plugin status, tasks and current capacity usage.

```
[Agent]
active: true
capacity: 0/1000
check on start: false
tasks: 3
agent.hostmetadata: Returns string with agent host metadata.
agent.hostname: Returns Hostname from agent configuration.
agent.ping: Returns agent availability check result.
agent.variant: Returns agent variant.
agent.version: Version of Zabbix agent.

[Boottime]
active: true
capacity: 0/1000
check on start: false
tasks: 1
system.boottime: Returns system boot time in unixtime.

[Cpu]
active: true
capacity: 0/1000
check on start: false
tasks: 12
system.cpu.discovery: List of detected CPUs/CPU cores, used for low-level discovery.
system.cpu.num: Number of CPUs.
system.cpu.util: CPU utilization percentage.

[DNS]
active: false
capacity: 0/1000
check on start: false
tasks: 0
```

Agent deployment with Host Wizard

Zabbix 7.4

One command on the target

Generate the PSK

Let the installer write the agent configuration

Copy the command and run it

The screenshot shows the Zabbix Host Wizard interface. The 'Hosts' tab is active, and the 'Install Zabbix agent' step is selected. The wizard is in the process of creating a new host named 'LB01 (new)'. The 'Pre-shared key' field is populated with a long hexadecimal string, and a 'Generate new' button is visible. Below this, the user is prompted to select the OS of the monitoring target, with 'Linux' selected. The final step is to set up the Zabbix agent on the target by executing a script. The script is displayed in a text box, which is highlighted with a red border. The script is a single command that uses 'curl' to download the Zabbix agent installer and 'bash' to execute it with specific options for version, server host, hostname, and pre-shared key. The 'Back' and 'Next' buttons are at the bottom right, and a 'Cancel' button is at the bottom left.

Hosts

Host Wizard Create host

Select a template: Linux by Zabbix agent Create or select a host: LB01 (new) Install Zabbix agent Add host interface Configure host

Enter a unique name that Zabbix components will use to recognize the pre-shared key.
Avoid including sensitive data.

* Pre-shared key

5895d6618a5e07cafd47423ec50197338eae1c427dac5818b8dc9eb515992e95 Generate new

Generate a secret pre-shared key hexadecimal string.

3. Select the OS of your monitoring target

Linux ☒ Windows ☐ Other ☐

4. Set up Zabbix agent on your monitoring target by executing the following script
[bash under root]:

```
$(command -v curl || echo $(command -v wget) -O -) https://  
cdn.zabbix.com/scripts/install-zabbix.sh | bash -s -- --version  
8.0 --server-host-stdin --hostname 'LB01' --psk-identity 'LB01'  
PSK' --psk  
5895d6618a5e07cafd47423ec50197338eae1c427dac5818b8dc9eb515992e95
```

Cancel Back Next

Zabbix 8.0.0rc1. © 2001–2026, Zabbix SIA

Proxies

A proxy can buffer in memory

Zabbix 7.0

disk

Database

hybrid

RAM, then database when
needed

memory

RAM only

```
zabbix_proxy.conf
```

```
ProxyBufferMode=hybrid
```

```
ProxyMemoryBufferSize=256M
```

Memory mode eliminates history-buffer database writes; useful on IoT and flash storage.

The proxy can recreate its SQLite database

```
bash · proxy stopped
```

```
sudo rm -f /var/lib/zabbix/proxy.db
```



SQLite

1

Database file is missing

Restart the proxy

2

Proxy creates a fresh database

Monitoring configuration comes from the server

3

Collection resumes

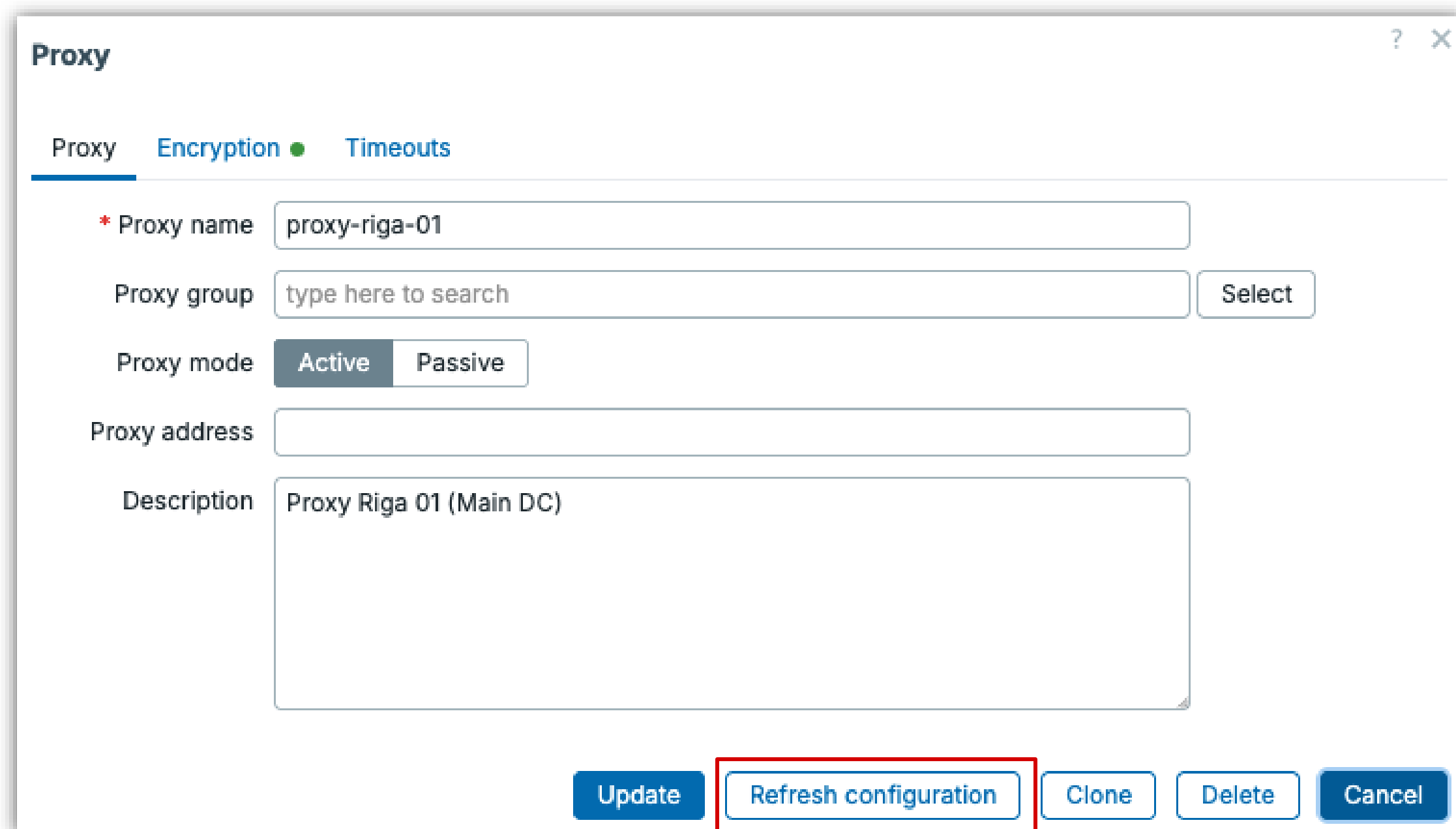
Existing server history is unaffected

Unsent history in the old database is lost.

Refresh proxy configuration from the UI

Zabbix 6.2

Administration / Proxies



The screenshot shows the Zabbix Proxy configuration window. The 'Proxy' tab is selected, and the 'Proxy name' is 'proxy-riga-01'. The 'Proxy group' is 'type here to search' with a 'Select' button. The 'Proxy mode' is 'Active'. The 'Proxy address' is empty. The 'Description' is 'Proxy Riga 01 (Main DC)'. At the bottom, there are five buttons: 'Update', 'Refresh configuration' (highlighted with a red box), 'Clone', 'Delete', and 'Cancel'.

Proxy

Proxy Encryption Timeouts

* Proxy name proxy-riga-01

Proxy group type here to search Select

Proxy mode Active Passive

Proxy address

Description Proxy Riga 01 (Main DC)

Update Refresh configuration Clone Delete Cancel

Refresh configuration without the command line.

Server and proxy versions can differ

Zabbix 6.4

Example: Zabbix 8.0 server

Proxy version	Status	Data	New configuration
8.0	Current	Yes	Yes
7.0 / 7.2 / 7.4	Outdated	Yes	No
Older than 7.0	Unsupported	No	No
Newer than 8.0	Unsupported	No	No

Useful for gradual upgrades. Full functionality requires matching major versions.

Let the proxy talk to Vault itself

Zabbix 7.4

Administration / General / Other

Storage of secrets

Vault provider

HashiCorp Vault

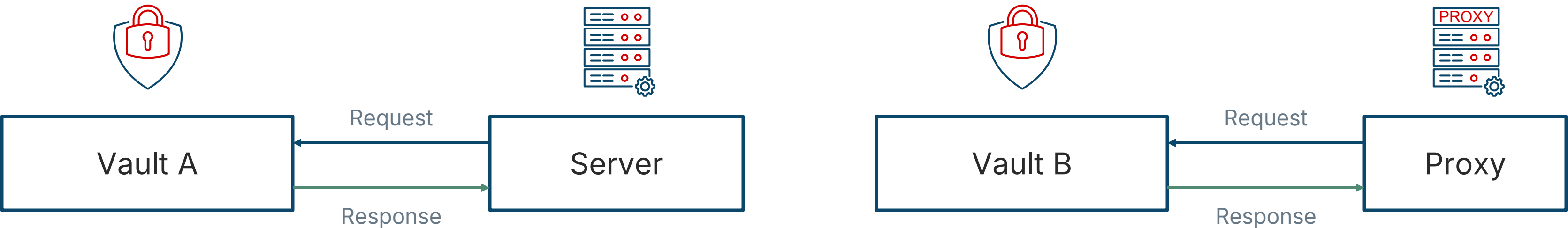
CyberArk Vault

Resolve secret vault macros by ?

Zabbix server

Zabbix server and proxy

- Different vaults for different sites
- Revoke one proxy's access independently



Secrets stay out of configuration transfers.
Give each proxy its own Vault access.

Triggers and macros

Why did this trigger fire?

Zabbix 6.0

Trigger expression

min(/db01/vm.memory.size[pavailable],5m)<10
 or last(/db01/system.swap.size[,pfree])<20

{TRIGGER.EXPRESSION.EXPLAIN}

9.2 < 10



true

{FUNCTION.VALUE1}

Minimum available memory, %

or

34.6 < 20



false

{FUNCTION.VALUE2}

Latest free swap, %

When was this value collected?

Zabbix 7.4

Timestamp macros

Value at the event

{ITEM.VALUE.AGE}
{ITEM.VALUE.DATE}
{ITEM.VALUE.TIME}
{ITEM.VALUE.TIMESTAMP}

Latest value

{ITEM.LASTVALUE.AGE}
{ITEM.LASTVALUE.DATE}
{ITEM.LASTVALUE.TIME}
{ITEM.LASTVALUE.TIMESTAMP}

Notification template

Value: {ITEM.VALUE}
Collected: {ITEM.VALUE.AGE} ago

Resolved notification

Value: 87
Collected: 4m 12s ago

Timestamps in trigger expressions

Zabbix 7.4

<code>firstclock()</code>	Oldest collection time in a period
---------------------------	------------------------------------

lastclock()	Latest or Nth collection time
-------------	-------------------------------

logtimestamp() Timestamp inside a log message

Find a reporting gap after data resumes

```
lastclock(/sensor01/temperature)
-lastclock(/sensor01/temperature,#2)>600
```



Calculate inside an event name

Zabbix 5.4

Trigger expression

Less than 10% free

```
last(/db01/vfs.fs.size[/,pfree])<10
```

Event name

```
Root FS: {{?last(/vfs.fs.size[/,free])  
/1024/1024/1024}.fmtnum(1)} GiB free
```

Resolved event name

Root FS: 6.4 GiB free

Macro functions beyond regsub()

Zabbix 7.0

Formatting

5.2

fmtnum()
fmttime()

Encoding

7.0

btoa()
htmlencode()
urlencode()

Strings

7.0

lowercase()
uppercase()
regrepl()
tr()

Since 7.0: wherever the underlying macro is supported, with minor exceptions

Example using the 7.4 timestamp macro

{{ITEM.VALUE.TIMESTAMP}.fmttime("%Y-%m-%d %H:%M:%S")}

Resolved value

2026-09-25 14:30:00

Regex contexts for user macros

```
{ $FS_FREE_MIN }=10
{ $FS_FREE_MIN:regex:"^/usr/tenants/[^/]+/data$" }=25
```

LLD trigger prototype

```
last(/Linux/vfs.fs.size[{#FSNAME},pfree])
<{ $FS_FREE_MIN:"{#FSNAME}" }
```

Filesystem	Threshold
/ and /home	10%
/usr/tenants/acme/data	25%
/usr/tenants/zeus/data	25%

Avoid overlapping regex contexts. Matching order is undefined.

Web monitoring

Extract JSON and XML in a web scenario

Zabbix 7.0

1. POST /api/v1/auth/login

```
{"access_token":"demo-token-7f2c","token_type":"Bearer"}
```

{token}=jsonpath:\$.access_token

2. Request the protected endpoint

Authorization: Bearer **{token}**

XML alternative

{token}=xmlxpath:/auth/access_token

Web scenario: capture the token

Zabbix 7.0

Extract from the response

{token}
jsonpath:\$.access_token

Read access_token
from the login response.

Step of web scenario

* Name

Login

* URL

https://api.example.com/api/v1/auth/login

Parse

Query fields

Name

Value

name

value

Add

Remove

Post type

Form data

Raw data

Raw post

{"email":"john@example.com","password":"changeme"}

Variables

Name

Value

{token}

jsonpath:\$.access_token

Add

Remove

Headers

Name

Value

Content-Type

application/json

Add

Remove

Follow redirects

☐

Retrieve mode

Body

Headers

Body and headers

* Timeout

15s

Required string

pattern

Web scenario: reuse the token

Zabbix 7.0

Reuse in the next request

Authorization
Bearer {token}

Send the token in
the request header.

Step of web scenario

name

⇒

value

Remove

Add

Post type

Form data

Raw data

Raw post

Variables

Name

name

Value

value

Remove

Add

Headers

Name

Authorization

Value

Bearer {token}

Remove

Content-Type

application/json

Remove

Add

Follow redirects☐Retrieve mode

Body

Headers

Body and headers

* Timeout

15s

Required string

"email"

Update

Cancel

Browser items with ChromeDriver

Zabbix 7.0

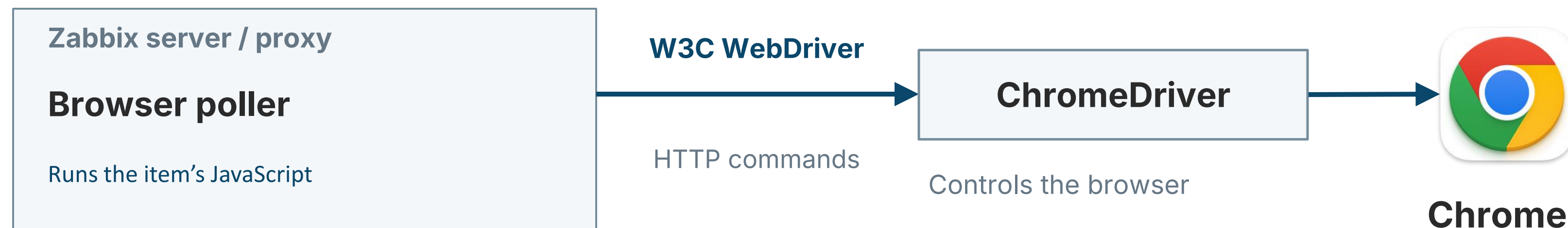
```
zabbix_server.conf / zabbix_proxy.conf
```

```
WebDriverURL=http://127.0.0.1:9515
```

```
bash
```

```
$ ./chromedriver --port=9515 --allowed-ips=127.0.0.1
```

```
ChromeDriver was started successfully on port 9515.
```



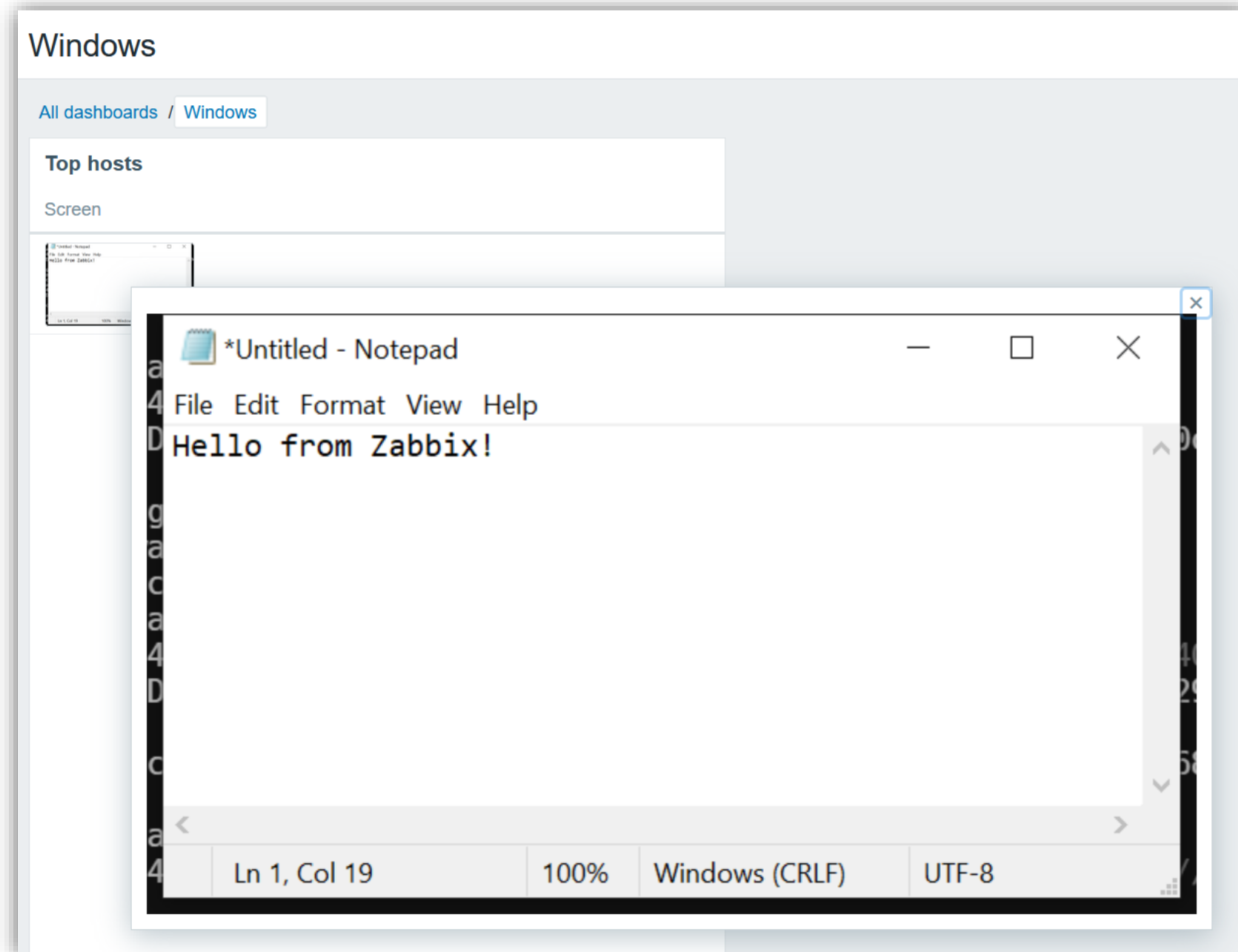
A direct WebDriver endpoint is enough for a small setup.

Application test results in Zabbix

Zabbix 7.0

Browser item (excerpt)

```
var browser = new Browser({
  capabilities: {
    alwaysMatch: {
      platformName: "Windows",
      "appium:automationName": "NovaWindows",
      "appium:app":
        "C:\\\\Windows\\\\System32\\\\notepad.exe",
      "appium:shouldCloseApp": false
    },
    firstMatch: [{}]
  }
});
browser.setElementWaitTimeout(10000);
var editor = browser.findElement(
  "xpath",
  "//*[ @ClassName='Edit' or " +
  "@ClassName='RichEditD2DPT']"
);
editor.click();
editor.sendKeys("Hello from Zabbix!");
```



How many browser pollers?

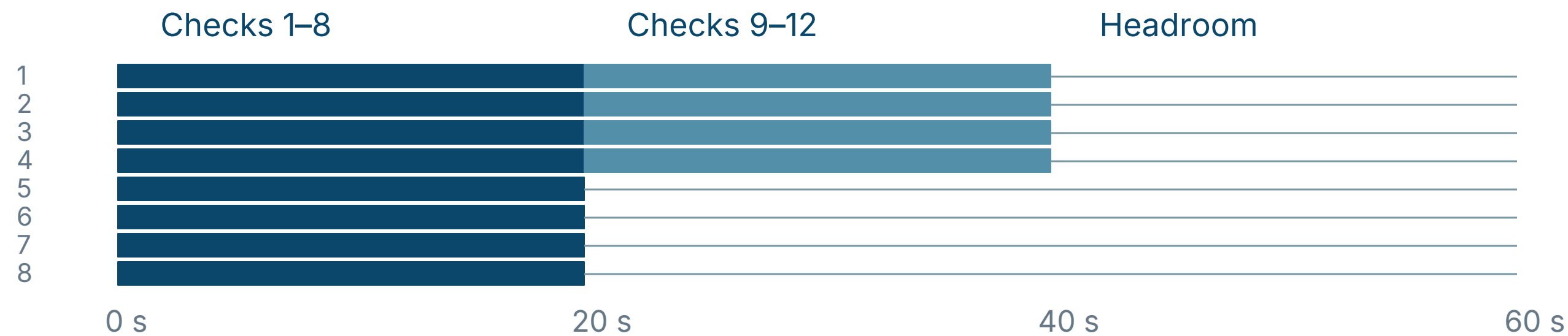
Zabbix 7.0

Synchronous: one running check per browser poller

$12 \text{ checks} \times 20 \text{ s} / 60 \text{ s} = 4 \text{ busy pollers on average}$

zabbix_server.conf / zabbix_proxy.conf

StartBrowserPollers=8



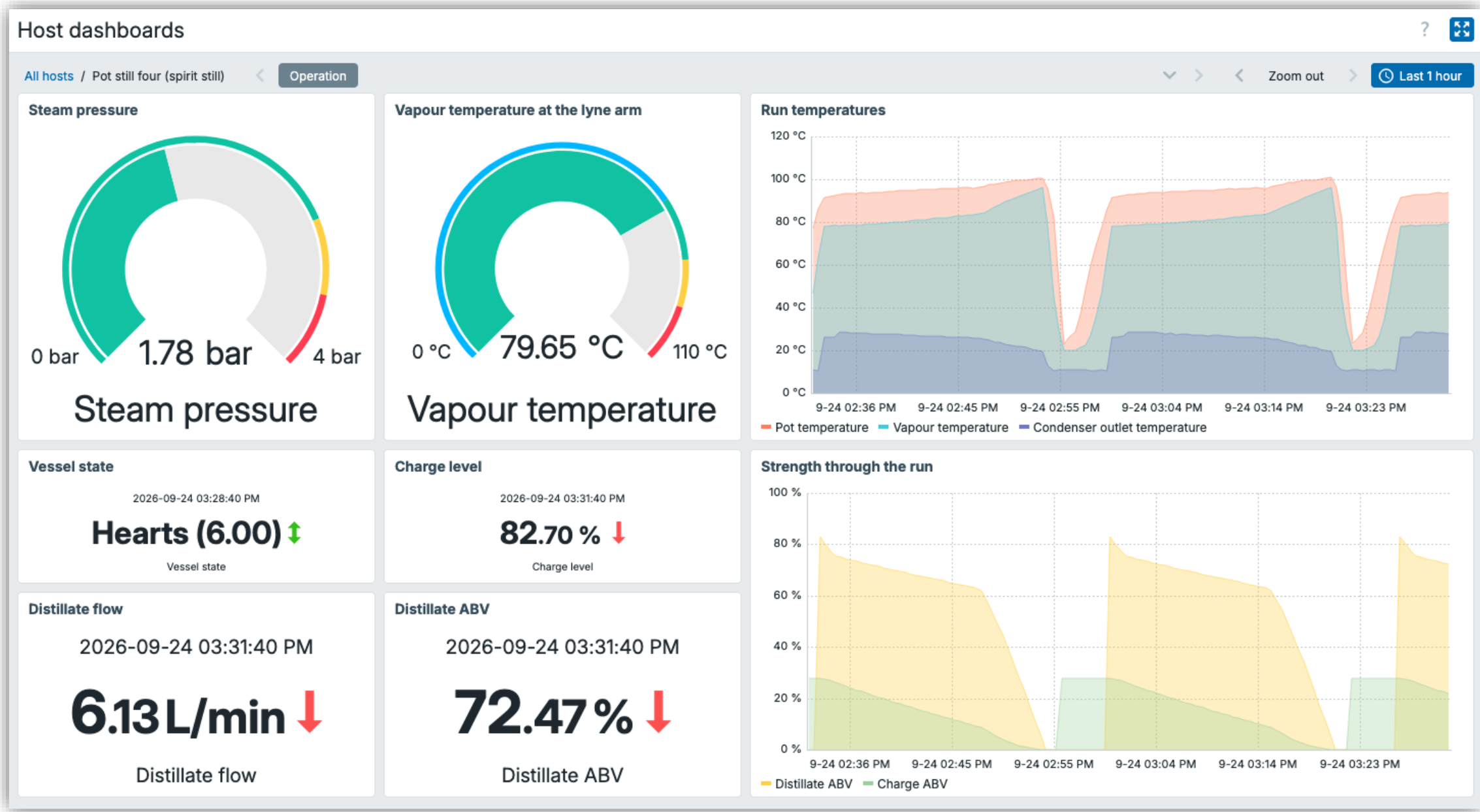
8 pollers: room for slow checks and timeouts.

Dashboards and navigation

All widgets in template dashboards

Zabbix 7.0

All widget types are available in template dashboards



Build the dashboard once and reuse it with the template.

Logs in the Item history widget

Zabbix 7.0

Log view

Logs

All dashboards / Logs

SystemApplicationStart slideshow

/var/log/messages

2026-09-24T18:46:28.478893+03:00 glengoyne systemd[1]: logrotate.service: Deactivated successfully.

2026-09-24T18:46:28.847829+03:00 glengoyne systemd[1]: Finished logrotate.service - Rotate log files.

2026-09-24T18:46:41.781786+03:00 glengoyne systemd[1]: Starting sysstat-collect.service - system activity accounting tool...

2026-09-24T18:46:41.844855+03:00 glengoyne systemd[1]: Finished sysstat-collect.service - system activity accounting tool.

2026-09-24T18:46:41.919202+03:00 glengoyne systemd[1]: sysstat-collect.service: Deactivated successfully.

2026-09-24T18:47:04.677703+03:00 glengoyne systemd[1]: Starting phpsessionclean.service - Clean php session files...

2026-09-24T18:47:04.713063+03:00 glengoyne systemd[1]: phpsessionclean.service: Deactivated successfully.

2026-09-24T18:47:05.318321+03:00 glengoyne systemd[1]: Finished phpsessionclean.service - Clean php session files.

2026-09-24T18:47:33.156261+03:00 glengoyne rsyslogd[2838]: [origin software="rsyslogd" swVersion="8.2406.0" x-pid="2838" x-info="https://www.rsyslog.com"] rsyslogd was H

2026-09-24T18:47:48.138847+03:00 glengoyne kernel: [91567.638847] [UFW BLOCK] IN=enol OUT= MAC=a0:8c:fd:4e:21:7b:f4:1e:57:9a:02:c3:08:00 SRC=185.93.157.36 DST=192.168.1.1

2026-09-24T18:48:02.101917+03:00 glengoyne NetworkManager[662]: <info> [1790264882.1019] dhcp4 (enol): state changed new lease, address=192.168.1.42

2026-09-24T18:48:32.711653+03:00 glengoyne kernel: [91612.211653] [UFW BLOCK] IN=enol OUT= MAC=a0:8c:fd:4e:21:7b:f4:1e:57:9a:02:c3:08:00 SRC=185.202.214.8 DST=192.168.1.1

2026-09-24T18:48:48.108395+03:00 glengoyne systemd-timesyncd[3306]: Contacted time server 185.125.190.57:123 (ntp.ubuntu.com).

2026-09-24T18:49:31.247700+03:00 glengoyne snapd[2929]: storehelpers.go:916: cannot refresh: snap has no updates available: "core22", "firefox", "snapd"

2026-09-24T18:49:54.714932+03:00 glengoyne CRON[22299]: (root) CMD ([-x /usr/lib/php/sessionclean] && if [! -d /run/systemd/system]; then /usr/lib/php/sessionclean;

2026-09-24T18:50:24.086106+03:00 glengoyne dockerd[1194]: time="2026-09-24T18:50:24.086106+03:00" level=info msg="ignoring event" container=4ef2c8201198 module=libcontai

2026-09-24T18:50:39.649322+03:00 glengoyne containerd[3473]: time="2026-09-24T18:50:39.649322+03:00" level=info msg="loading plugin \"io.containerd.event.v1.publisher\".

2026-09-24T18:50:56.867124+03:00 glengoyne containerd[3473]: time="2026-09-24T18:50:56.867124+03:00" level=info msg="loading plugin \"io.containerd.event.v1.publisher\".

2026-09-24T18:52:01.259769+03:00 glengoyne kernel: [91820.759769] [UFW BLOCK] IN=enol OUT= MAC=a0:8c:fd:4e:21:7b:f4:1e:57:9a:02:c3:08:00 SRC=185.107.127.253 DST=192.168.1.1

2026-09-24T18:52:49.103928+03:00 glengoyne CRON[22380]: (root) CMD ([-x /usr/lib/php/sessionclean] && if [! -d /run/systemd/system]; then /usr/lib/php/sessionclean;

2026-09-24T18:53:57.704264+03:00 glengoyne rsyslogd[2838]: [origin software="rsyslogd" swVersion="8.2406.0" x-pid="2838" x-info="https://www.rsyslog.com"] rsyslogd was H

2026-09-24T18:54:13.881219+03:00 glengoyne rsyslogd[2838]: [origin software="rsyslogd" swVersion="8.2406.0" x-pid="2838" x-info="https://www.rsyslog.com"] rsyslogd was H

2026-09-24T18:55:10.240932+03:00 glengoyne containerd[3473]: time="2026-09-24T18:55:10.240932+03:00" level=info msg="loading plugin \"io.containerd.event.v1.publisher\".

Widget settings

edit widget

TypeItem historyShow header

Name/var/log/messages

Refresh intervalDefault (1 minute)

LayoutHorizontalVertical

* Items

Name	Item	Actions
Syslog	glengoyne: syslog	Edit Remove
Add		

* Show lines

25

Override hosttype here to searchSelect

Advanced configuration

New valuesTopBottom

Show timestamp

Show column headerOffHorizontalVertical

Time periodDashboardWidgetCustom

ApplyCancel

Column settings

×

Update column

*

Name

Syslog

*

Item

glengoyne: syslog

×

Select

Base color

Highlights

Regular expression

cannot|fail|error

Remove

BLOCK

Remove

Add

Display

?

As is

HTML

Single line

200

Use monospace font

✓

Display log time

?

Update

Cancel

Graph data sets can use item IDs

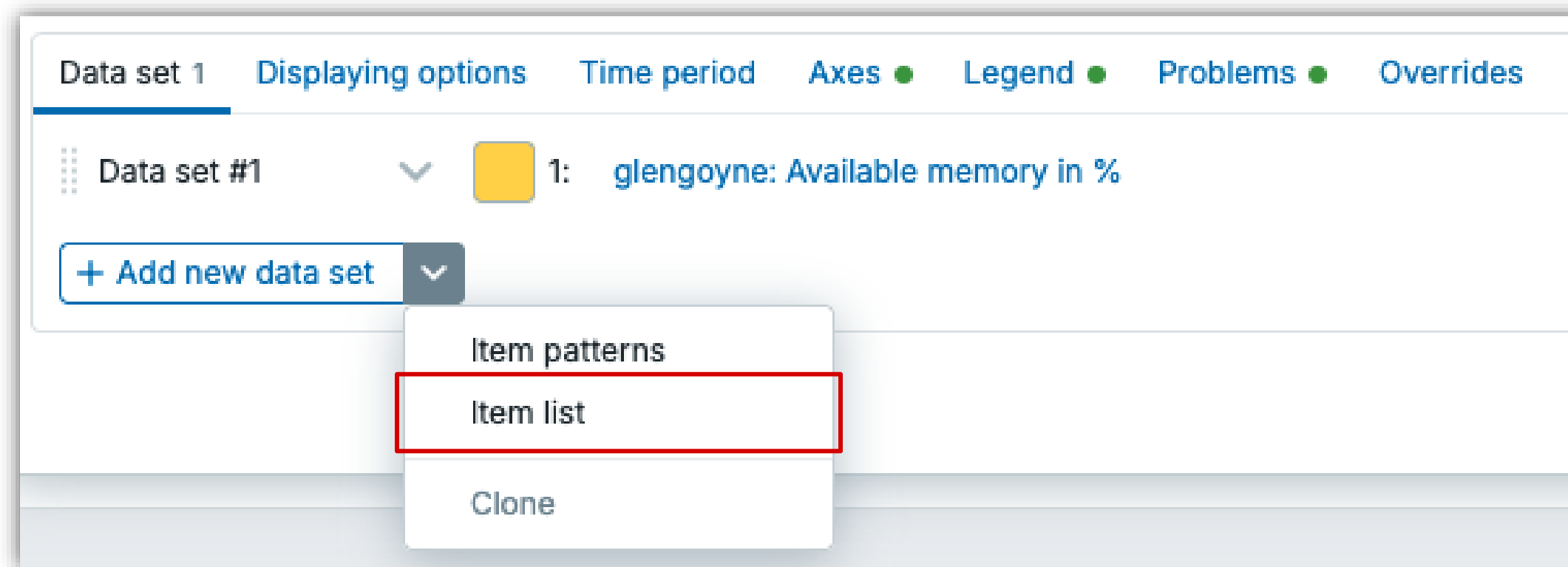
Zabbix 6.2

Item patterns

Match hosts and item names
Useful for a large or changing set of metrics

Item list

Select items in the UI
Zabbix stores their item IDs



Global search as a navigation shortcut

Templates, groups and hosts

Search for part of the name.

Z

Search: prox

?

Hosts

Host	IP	DNS	Monitoring	Configuration
Proxmox VE	192.168.11.25		Latest data Problems Graphs Dashboards Web	Items 145 Triggers 58 Graphs 53 Discovery 5 Web
Zabbix proxy (zabbix-proxy)	192.168.11.108		Latest data Problems Graphs Dashboards Web	Items 106 Triggers 108 Graphs 11 Discovery 4 Web

Displaying 2 of 2 found

Host groups

Host group	Monitoring	Configuration
Proxmox	Latest data Problems Web	Hosts 1

Displaying 1 of 1 found

Templates

Template	Configuration
HAProxy by HTTP	Items 6 Triggers 4 Graphs Dashboards 3 Discovery 6 Web
HAProxy by Zabbix agent	Items 6 Triggers 4 Graphs Dashboards 3 Discovery 6 Web
Proxmox VE by HTTP	Items 3 Triggers 1 Graphs Dashboards Discovery 5 Web
Proxy Availability	Items 1 Triggers 2 Graphs Dashboards Discovery Web
Remote Zabbix proxy health	Items 50 Triggers 37 Graphs Dashboards 1 Discovery Web
Zabbix proxy health	Items 50 Triggers 36 Graphs Dashboards 1 Discovery Web
Zabbix proxy health by Zabbix agent	Items 50 Triggers 37 Graphs Dashboards 1 Discovery Web
Zabbix proxy health by Zabbix agent active	Items 50 Triggers 37 Graphs Dashboards 1 Discovery Web

Displaying 8 of 8 found

Template groups

Template group	Configuration
Reverse proxies	Templates

Displaying 1 of 1 found

Search by part of an IP address

Z

Search: 192.168.14.

?

Search

Grid

Hosts

Host groups

Templates

Template groups

Help

Hosts

Host

IP

DNS

Monitoring

Configuration

DGX Spark 1

192.168.14.79

Latest data

Problems

Graphs

Dashboards

Web

Items 34

Triggers 13

Graphs 5

Discovery 4

Web

Displaying 1 of 1 found

Host groups

Host group

Monitoring

Configuration

No data found

Templates

Template

Configuration

No data found

Template groups

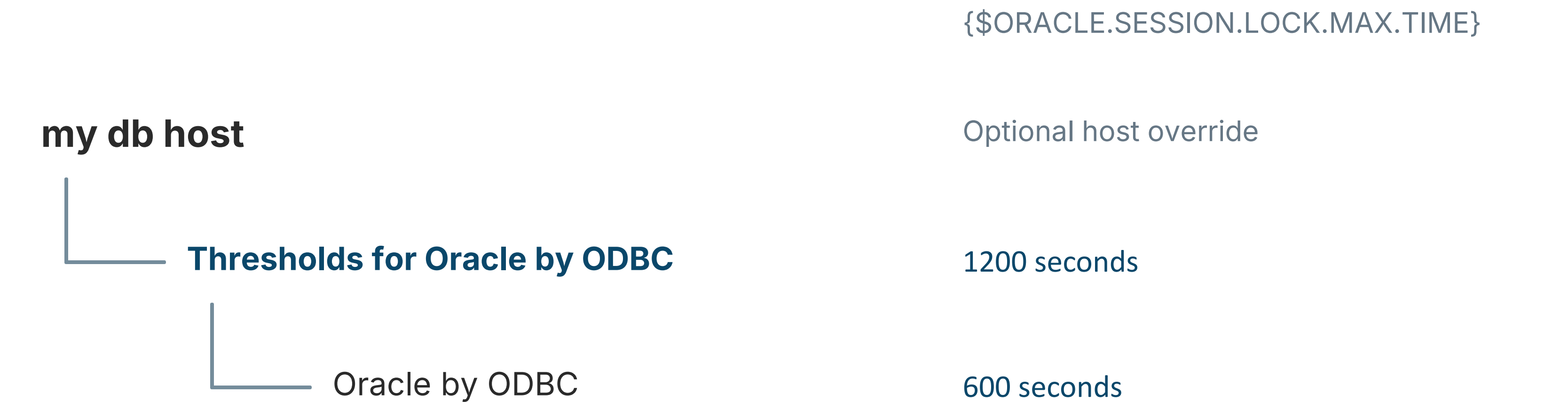
Template group

Configuration

No data found

Configuration and data collection

Macro overrides in a custom template



Host uses 1200 seconds

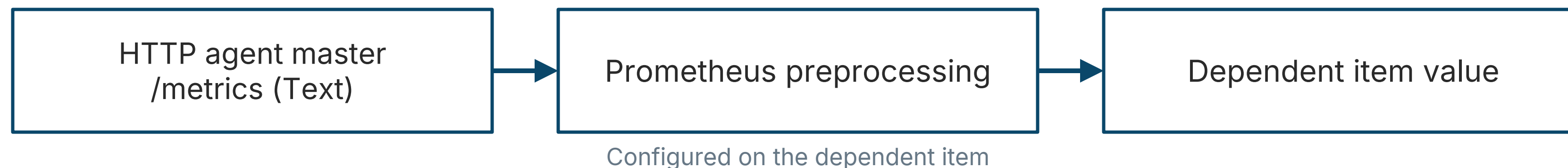
The standard template stays unchanged

Link the host to the custom template, with Oracle by ODBC underneath

Monitoring Prometheus metrics with Zabbix

Application metrics · requests by route and status

```
api_requests_total{route="/login",code="401"} 73  
api_requests_total{route="/login",code="200"} 12480  
api_requests_total{route="/orders",code="500"} 18
```



1. Prometheus pattern — output: Value

```
api_requests_total{route="/login",code="401"}
```

2. Change per second · Numeric (float) · requests/s

Asynchronous polling needs fewer processes

Zabbix 7.0

Up to 1,000 concurrent passive agent checks

Zabbix 6.0

zabbix_server.conf / zabbix_proxy.conf

StartPollers=1000

1,000

processes

≈

Zabbix 7.0

zabbix_server.conf / zabbix_proxy.conf

StartAgentPollers=1
MaxConcurrentChecksPerPoller=1000

1

process

One poller can wait for many responses at once.

Configurable item timeouts

Zabbix 7.0

Global, proxy and item timeouts

Global default

3 seconds

Proxy override

5 seconds

Custom item timeout

Set on a template item; inherited by its hosts

15 seconds

Allowed range: 1–600 seconds

Disable lost LLD resources before deleting

Zabbix 7.0

Discovery rule settings

* Delete lost resources ?

Never

Immediately

After

7d

* Disable lost resources ?

Never

Immediately

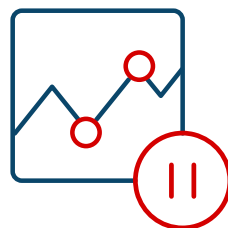
After

Example policy

Disable

After 1h

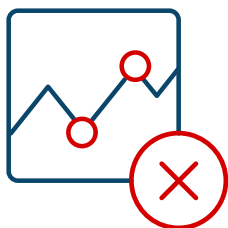
Stop polling



Delete

After 7d

Remove the resource



Configuration from environment variables

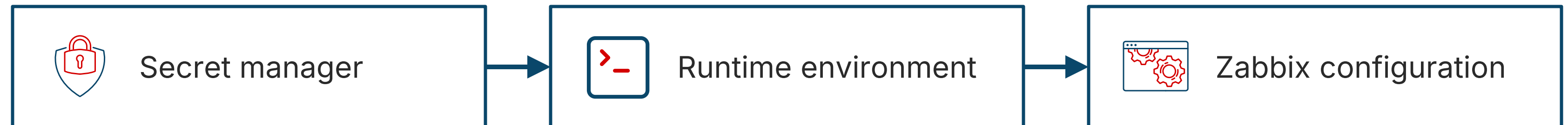
Zabbix 7.2

zabbix_server.conf

DBPassword=**`${ZABBIX_DB_PASSWORD}`**

zabbix_agentd.conf / zabbix_agent2.conf

Hostname=**`${ZABBIX_HOSTNAME}`**



Keep secrets out of static configuration.

Achieving 1M NVPS

Zabbix 8.0

1,000,000

new values every second

Zabbix 8.0 with ClickHouse history storage

Why Clickhouse

Zabbix 8.0

Time series

Very fast inserts and queries by item and time range

Compression

Similar values stored together mean less disk space and I/O

SQL

Familiar filters and aggregations help to reuse existing queries

Retention

TTL removes old history inside the storage engine

A single-node ClickHouse setup

Zabbix 8.0

bash · Docker

```
docker run -d --name clickhouse \  
-p 127.0.0.1:8123:8123 \  
-e CLICKHOUSE_DB=zabbix -e CLICKHOUSE_USER=zabbix \  
-e CLICKHOUSE_PASSWORD='<secret>' \  
-e CLICKHOUSE_DEFAULT_ACCESS_MANAGEMENT=1 \  
-v clickhouse_data:/var/lib/clickhouse \  
--ulimit nofile=262144:262144 \  
clickhouse/clickhouse-server:26.3
```

bash · Zabbix ClickHouse schema directory

```
./history_all.sh --server http://127.0.0.1:8123 \  
--db zabbix --user zabbix --password '<secret>'
```

Connect Zabbix to ClickHouse

Zabbix 8.0

Example: numeric history

zabbix_server.conf

```
HistoryProvider=clickhouse;value_types="uint,dbl",url=http://127.0.0.1:8123,db=zabbix,username=zabbix,password=<secret>
```

zabbix.conf.php

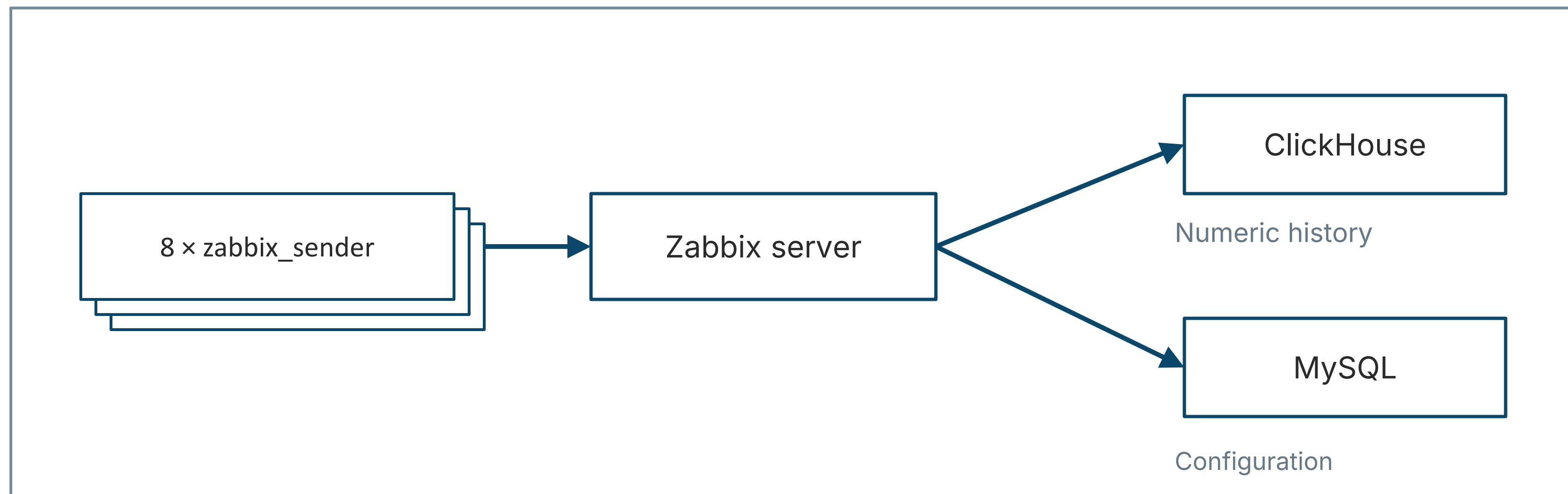
```
$HISTORY_PROVIDERS[] = [  
    'types' => ['uint', 'dbl'], 'provider' => 'clickhouse',  
    'url' => 'http://127.0.0.1:8123', 'db' => 'zabbix',  
    'username' => 'zabbix', 'password' => '<secret>'  
];
```

The test setup

Zabbix 8.0

One laptop

4 performance CPU cores · 32 GB RAM



Configured scale and incoming data

Zabbix 8.0

2,500 hosts · 25,000 application instances

All benchmark items receive data

150,000 triggers

975,000 items

120,000 last()
15,000 avg()
7,500 nodata()
7,500 change()

8 sender streams

Zabbix server settings

Zabbix 8.0

zabbix_server.conf

```
StartTrappers=12
StartDBSyncers=12
CacheSize=8G
HistoryCacheSize=3G
HistoryIndexCacheSize=1G
TrendCacheSize=1G
TrendFunctionCacheSize=256M
ValueCacheSize=3G
DBHost=localhost
DBSocket=/tmp/mysql.sock
```

ClickHouse memory settings

Parameter	Value	Configuration file
max_server_memory_usage	8 GiB	config.xml
max_memory_usage	6 GiB	users.xml / profile
mark_cache_size	1 GiB	config.xml
uncompressed_cache_size	0	config.xml

6 GiB is a per-query limit, within the server memory budget

ClickHouse asynchronous inserts

50 MB threshold · adaptive flushing at 50–200 ms

users.xml · profiles / default

```
<async_insert_max_data_size>50000000</async_insert_max_data_size>  
<async_insert_use_adaptive_busy_timeout>1</async_insert_use_adaptive_busy_timeout>  
<async_insert_busy_timeout_min_ms>50</async_insert_busy_timeout_min_ms>  
<async_insert_busy_timeout_max_ms>200</async_insert_busy_timeout_max_ms>
```

Parameters sent by the Zabbix ClickHouse backend

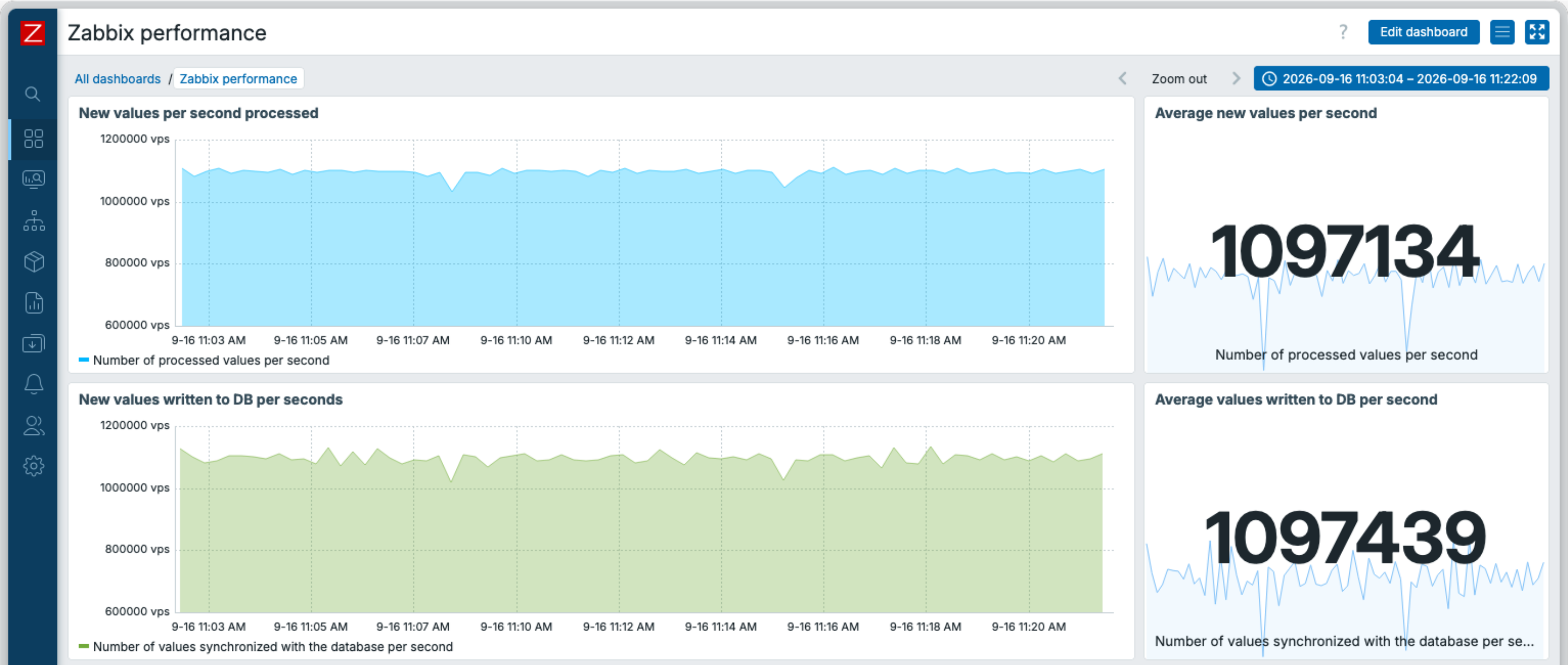
HTTP insert request · query parameters

```
async_insert=1&wait_for_async_insert=0
```

Acknowledgement can arrive before the buffered data is written.

Processing and storage in Zabbix

Zabbix 8.0



1.10M NVPS on a laptop

Zabbix 8.0

15 minutes after warm-up

1,096,237

new values/s processed

1,096,312

new rows/s stored

1,322,201,401 acknowledged values = stored rows

After drain, including warm-up

0 logged insert errors

Peak sampled history cache use: 1.235%

Thank you