

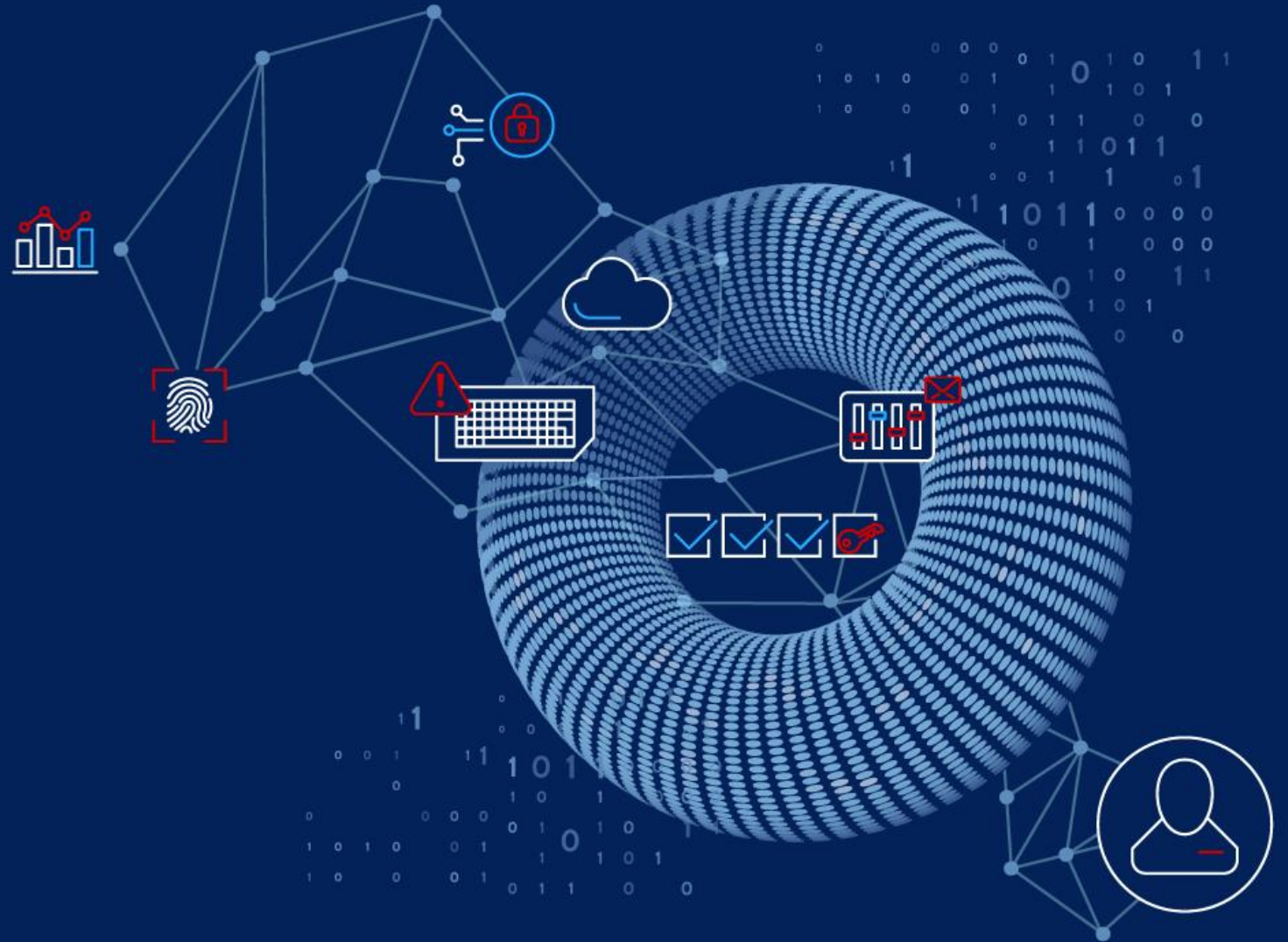
ZABBIX

6.4

INCREMENTAL
CONFIGURATION
UPDATES

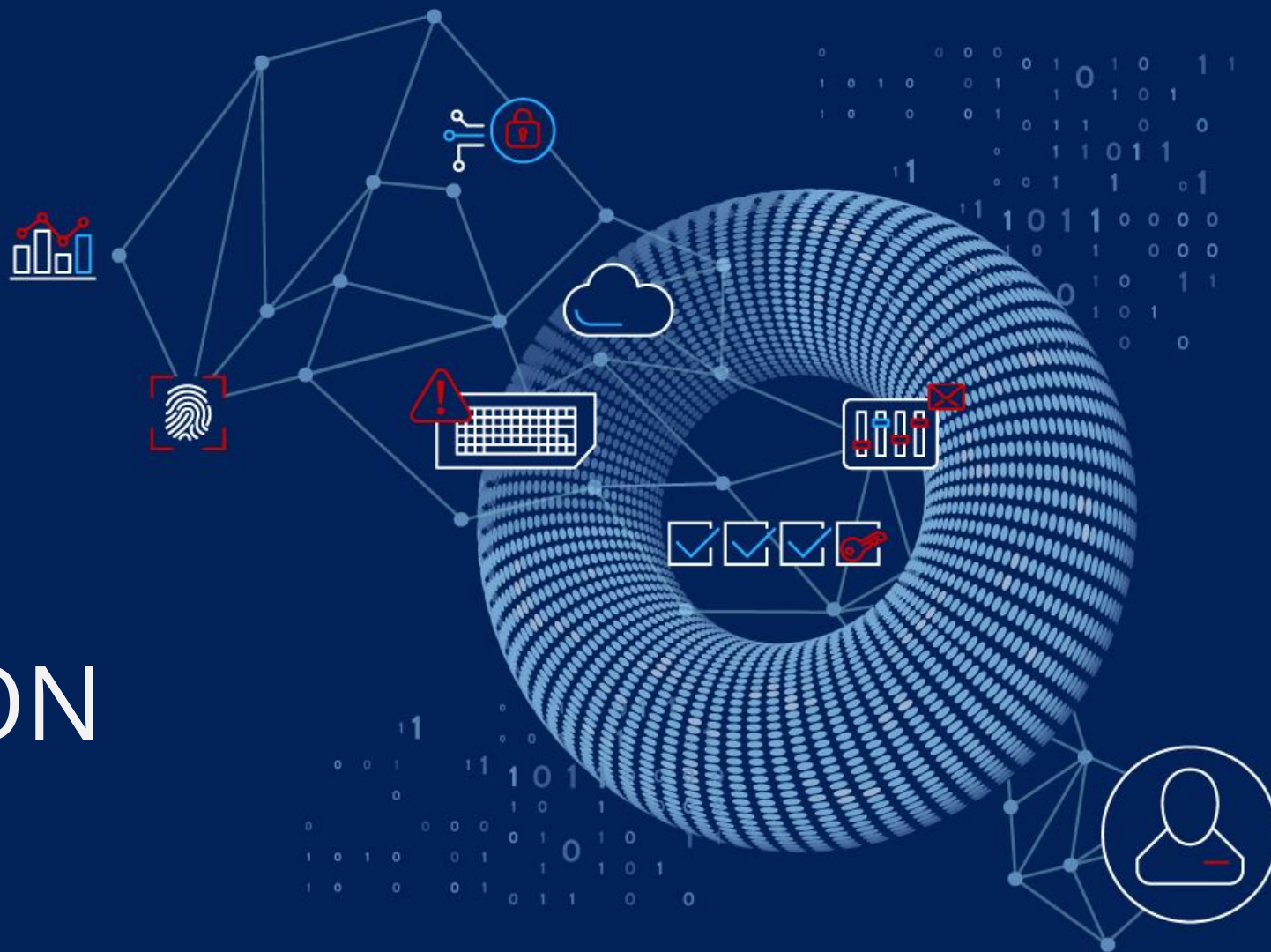


Kaspars Mednis
Chief trainer



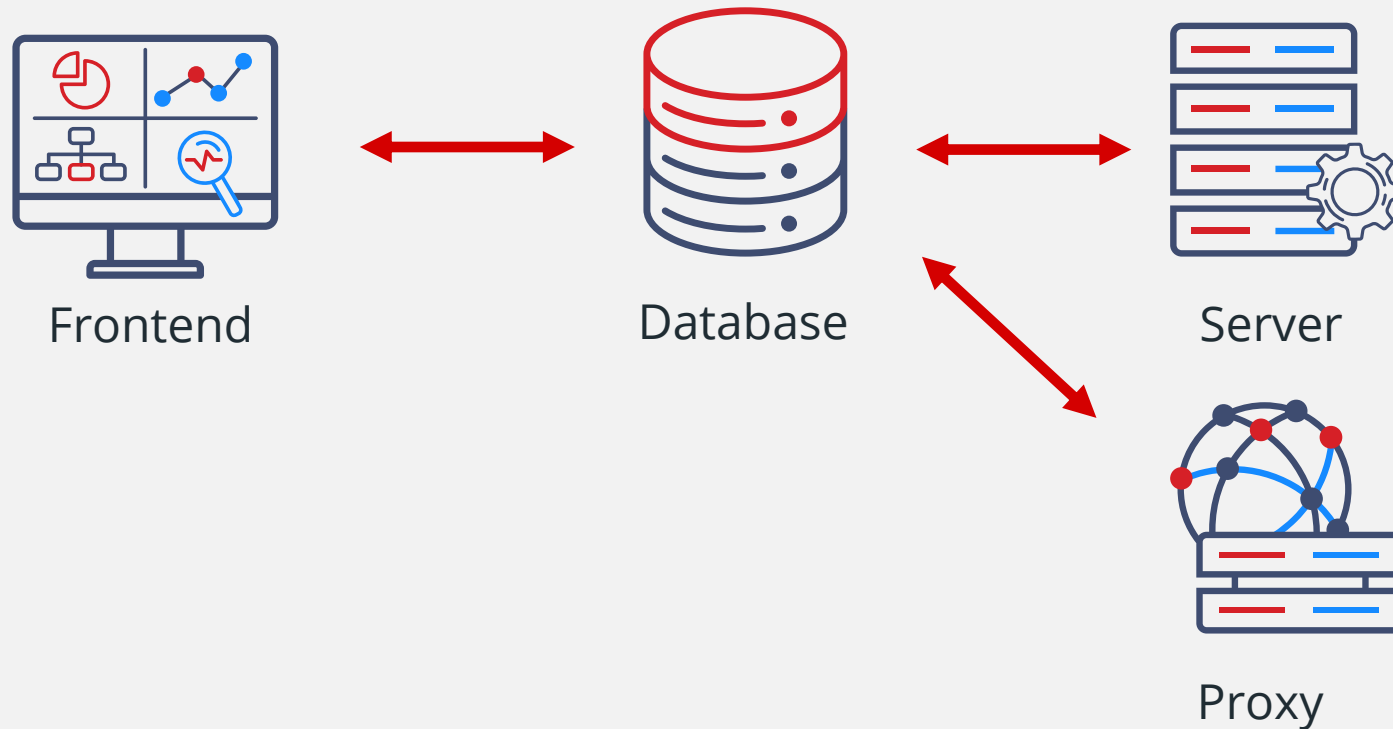
01

CONFIGURATION CACHE



Zabbix monitoring configuration is stored in a **relational database**:

- ⚡ Web frontend always displays the **latest configuration** on demand
- ⚡ Zabbix server and proxies may have **delayed configuration snapshots**



Zabbix server is storing configuration snapshot in a shared memory:

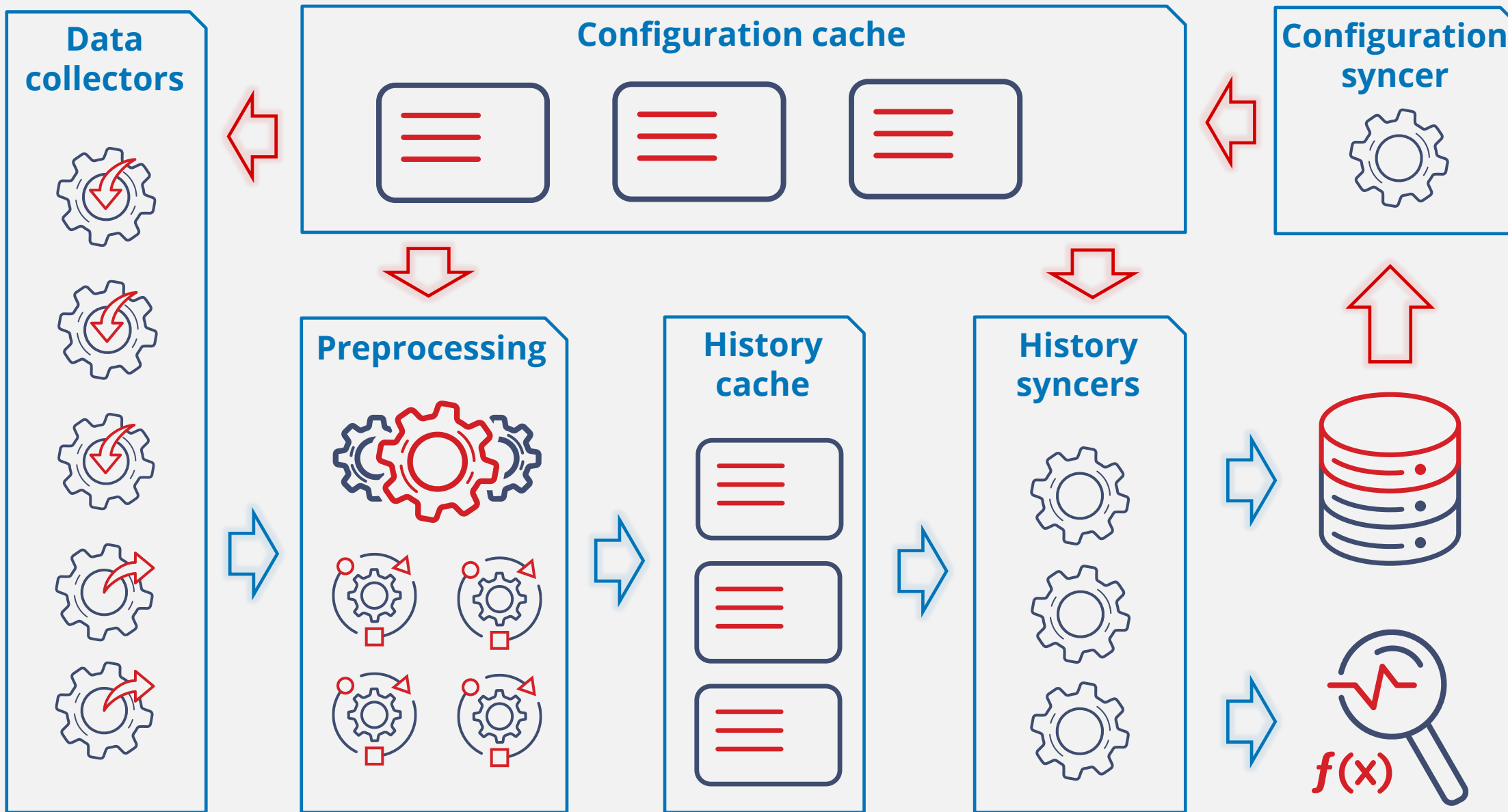
⚡ Memory region is called **Configuration cache**

⚡ The process responsible for updating the cache is named **Configuration syncer**

```
### Option: CacheSize
#       Size of configuration cache, in bytes.
#       Shared memory size for storing host, item and trigger data.
#
# Mandatory: no
# Range: 128K-64G
# Default:
CacheSize=4G
```

```
### Option: CacheUpdateFrequency
#       How often Zabbix will perform update of configuration cache, in seconds.
#
# Mandatory: no
# Range: 1-3600
# Default:
CacheUpdateFrequency=60
```

Configuration cache is used by majority of Zabbix processes

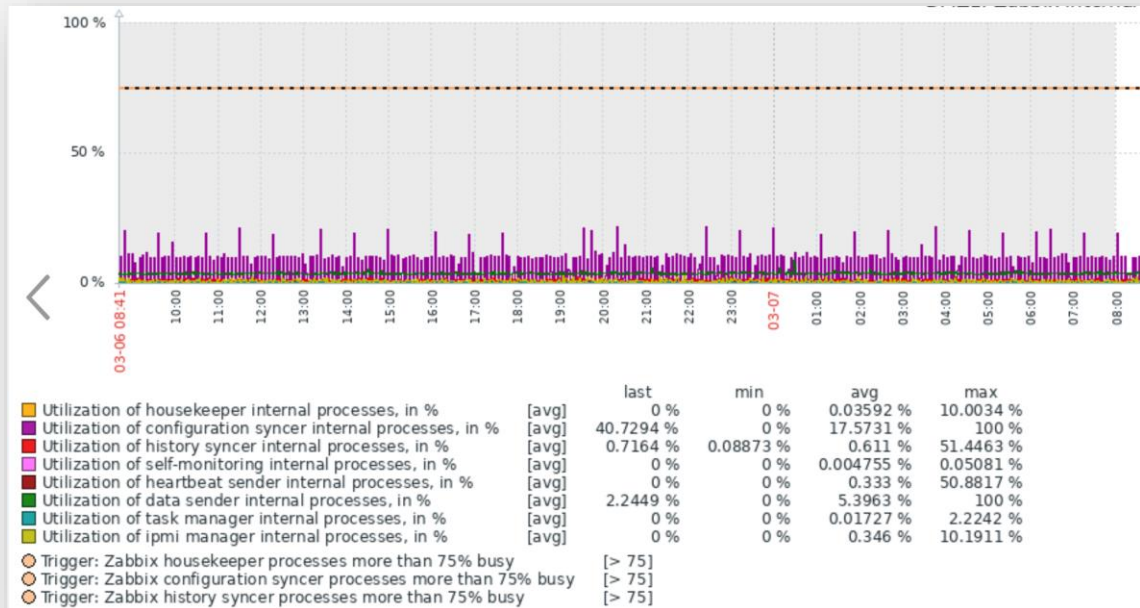


The configuration syncer usage can be seen using:

📡 Zabbix server process list

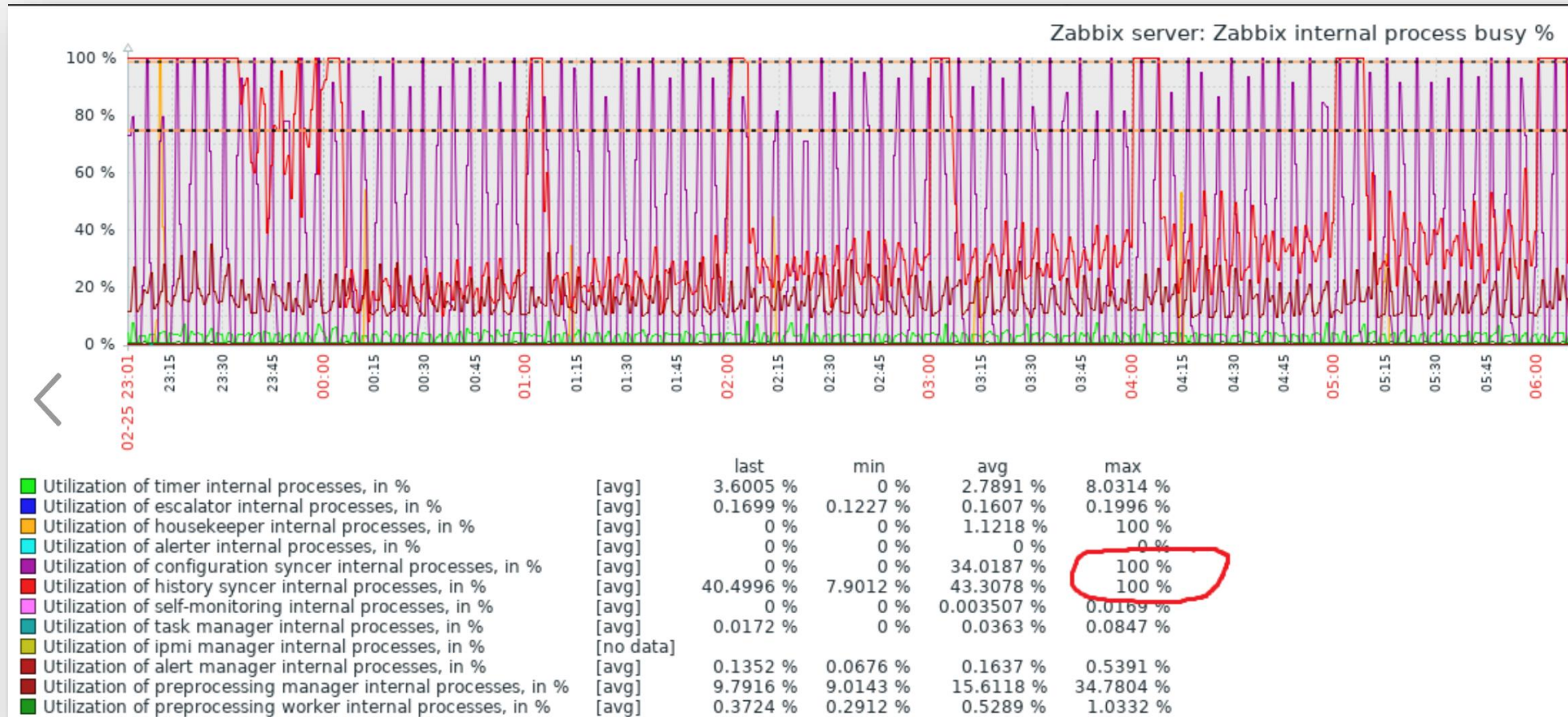
```
1748447 /usr/sbin/zabbix_server -c /etc/zabbix/zabbix_server.conf
1748448 /usr/sbin/zabbix_server: ha manager
1748450 /usr/sbin/zabbix_server: configuration syncer [synced configuration in 23.447738 sec, idle 60 sec]
1748453 /usr/sbin/zabbix_server: alert manager #1 [sent 0, failed 0 alerts, idle 5.018885 sec during 5.018997 sec]
1748483 /usr/sbin/zabbix_server: history syncer #1 [processed 1000 values, 961 triggers in 0.41459 sec, syncing history]
1748484 /usr/sbin/zabbix_server: history syncer #2 [processed 543 values, 435 triggers in 0.330040 sec, syncing history]
1748485 /usr/sbin/zabbix_server: history syncer #3 [processed 0 values, 0 triggers in 0.000034 sec, idle 1 sec]
1748486 /usr/sbin/zabbix_server: history syncer #4 [processed 0 values, 0 triggers in 0.000026 sec, syncing history]
1748487 /usr/sbin/zabbix_server: escalator #1 [processed 0 escalations in 0.014313 sec, idle 1 sec]
.....
```

📡 Zabbix server internal metrics



Before Zabbix 6.2 only full configuration synchronization was possible:

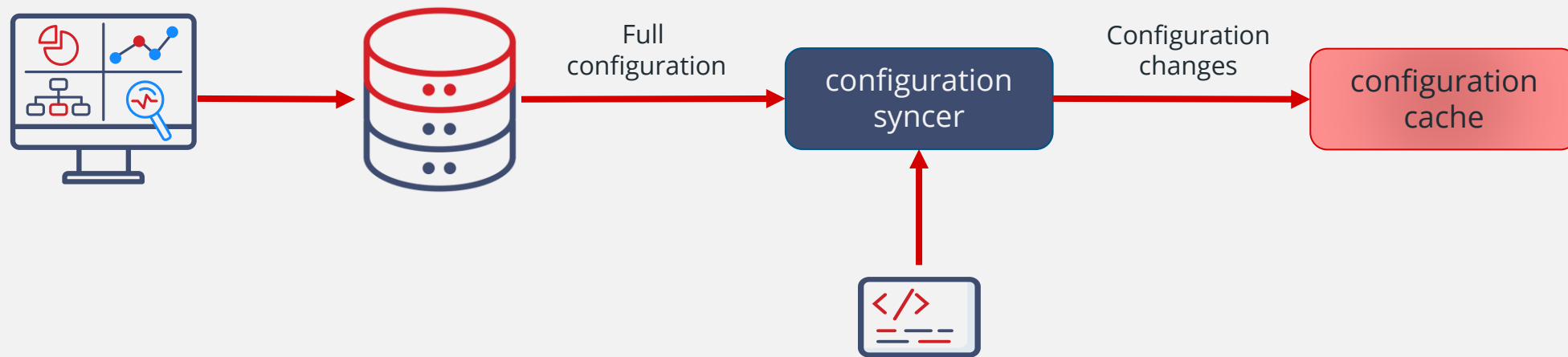
- ⚡ Full configuration sync is performed by executing complex SQL SELECT queries
- ⚡ This can negatively affect database performance
- ⚡ It was recommended to update configuration not too often in large environments



Possible solutions before Zabbix 6.2:

- ⚡ Increase `CacheUpdateFrequency` to 10 minutes or even 1 hour
- ⚡ Just wait... or execute configuration sync on demand if required

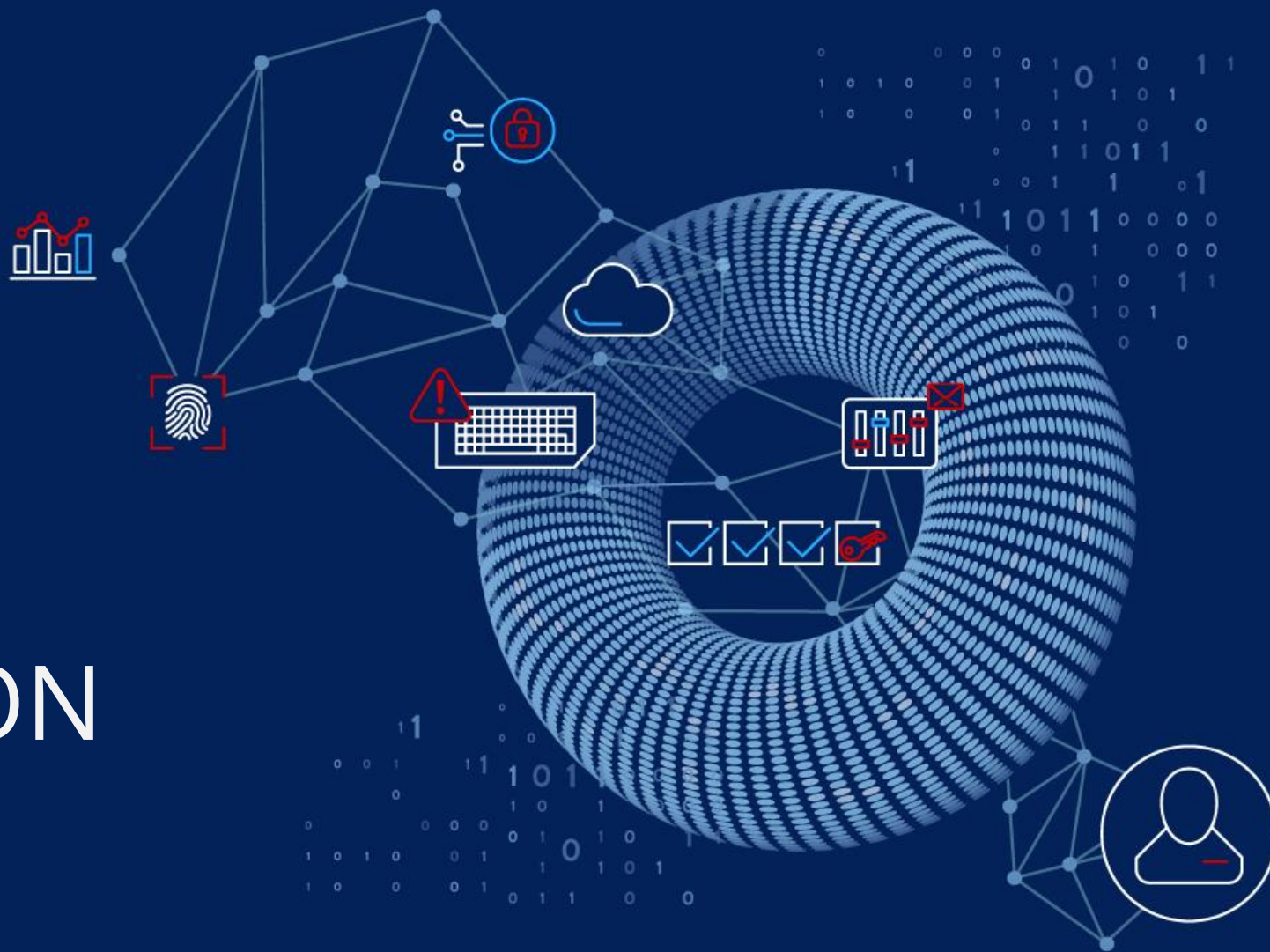
```
# zabbix_server -R config_cache_reload  
Runtime control command was forwarded successfully
```



```
zabbix_server -R config_cache_reload
```


02

CONFIGURATION SYNCER



Configuration syncer process itself was optimized a long time ago:

- ⚡ Full configuration is requested from database every time
- ⚡ Only changes are written to configuration cache

```
DCsync_configuration() config      : sql:0.000819 sync:0.000069 sec (1/0/0).
DCsync_configuration() autoreg     : sql:0.000378 sync:0.000046 sec (1/0/0).
DCsync_configuration() hosts       : sql:1.008107 sync:0.000078 sec (2/0/2).
DCsync_configuration() host_invent: sql:0.012147 sync:0.000061 sec (2/0/2).
DCsync_configuration() templates   : sql:0.004790 sync:0.000082 sec (4/0/4).
DCsync_configuration() globmacros  : sql:0.000486 sync:0.000048 sec (0/0/0).
DCsync_configuration() hostmacros  : sql:0.006122 sync:0.000046 sec (0/0/0).
DCsync_configuration() triggers    : sql:9.814785 sync:0.000110 sec (2/0/3).
DCsync_configuration() trigdeps    : sql:0.018868 sync:0.000052 sec (1/0/0).
DCsync_configuration() trig. tags  : sql:4.581970 sync:0.000055 sec (2/0/2).
DCsync_configuration() host tags   : sql:0.000392 sync:0.000046 sec (0/0/0).
DCsync_configuration() functions   : sql:2.202464 sync:0.000155 sec (4/0/5).
DCsync_configuration() expressions: sql:0.000674 sync:0.000071 sec (0/0/0).
DCsync_configuration() actions     : sql:0.000667 sync:0.000049 sec (0/0/0).
DCsync_configuration() operations  : sql:0.001537 sync:0.000065 sec (0/0/0).
DCsync_configuration() conditions  : sql:0.002381 sync:0.000063 sec (0/0/0).
DCsync_configuration() hgroups     : sql:0.012523 sync:0.000129 sec (0/0/0).
DCsync_configuration() item pproc  : sql:4.276886 sync:0.000045 sec (0/0/0).
.....
DCsync_configuration() reindex     : 0.025953 sec.
DCsync_configuration() total sql   : 23.447738 sec.
DCsync_configuration() total sync  : 0.027991 sec.
```

Starting from Zabbix 6.2 only **configuration changes** are synchronized:

- ⚡ Full synchronization is initialized only on Zabbix server start
- ⚡ After that only configuration changes are read from the database
- ⚡ This results in much **faster** and **lighter** configuration updates
- ⚡ The default **CacheUpdateFrequency** is now just 10 seconds

Before 6.2

full sync

full sync

full sync

Zabbix 6.2 and later

full sync

sync

sync

sync

sync

sync

sync

sync

sync

sync

How this is achieved from the technical perspective:

- ⚡ New table **changelog** is introduced
- ⚡ This table includes all configuration changes to hosts, items, triggers, etc.
- ⚡ On every sync server checks changelog table for new changelogid's

	changelogid	object	objectid	operation	clock
Synced	156138341	1	12242	1	1680679062
	156138342	3	543246	1	1680679062
	156138343	3	543247	1	1680679062
	156138344	3	543248	2	1680679062
	156138345	3	543249	2	1680679062
Not Synced	156138346	8	383949	2	1680679065
	156138347	8	383950	2	1680679065
	156138348	5	86431	3	1680679065
	156138349	5	86431	3	1680679065
	156138350	5	86431	2	1680679065

How the changelog table is maintained:

- ⚡ Database triggers are added to write any changes into **changelog** table
- ⚡ This means, the database itself is responsible for maintaining the changelog

```
create trigger items_insert
after insert
on items for each row
insert into changelog (object,objectid,operation,clock) values (3,new.itemid,1,unix_timestamp());
```

```
create trigger items_update
after update
on items for each row
insert into changelog (object,objectid,operation,clock) values (3,old.itemid,2,unix_timestamp());
```

```
create trigger items_delete
before delete
on items for each row
insert into changelog (object,objectid,operation,clock) values (3,old.itemid,3,unix_timestamp());
```

A few notes about changelog and database triggers:

- ⚡ Available on all supported database engines (MySQL, PostgreSQL, Oracle, SQLite3)
- ⚡ Once the entry is synchronized with cache it is no longer needed in database
- ⚡ Smaller configuration entries like user macros are still fully synced every time
- ⚡ Records older than 1 hour are automatically removed by housekeeper

Additional privileges may be required for zabbix user on database level:

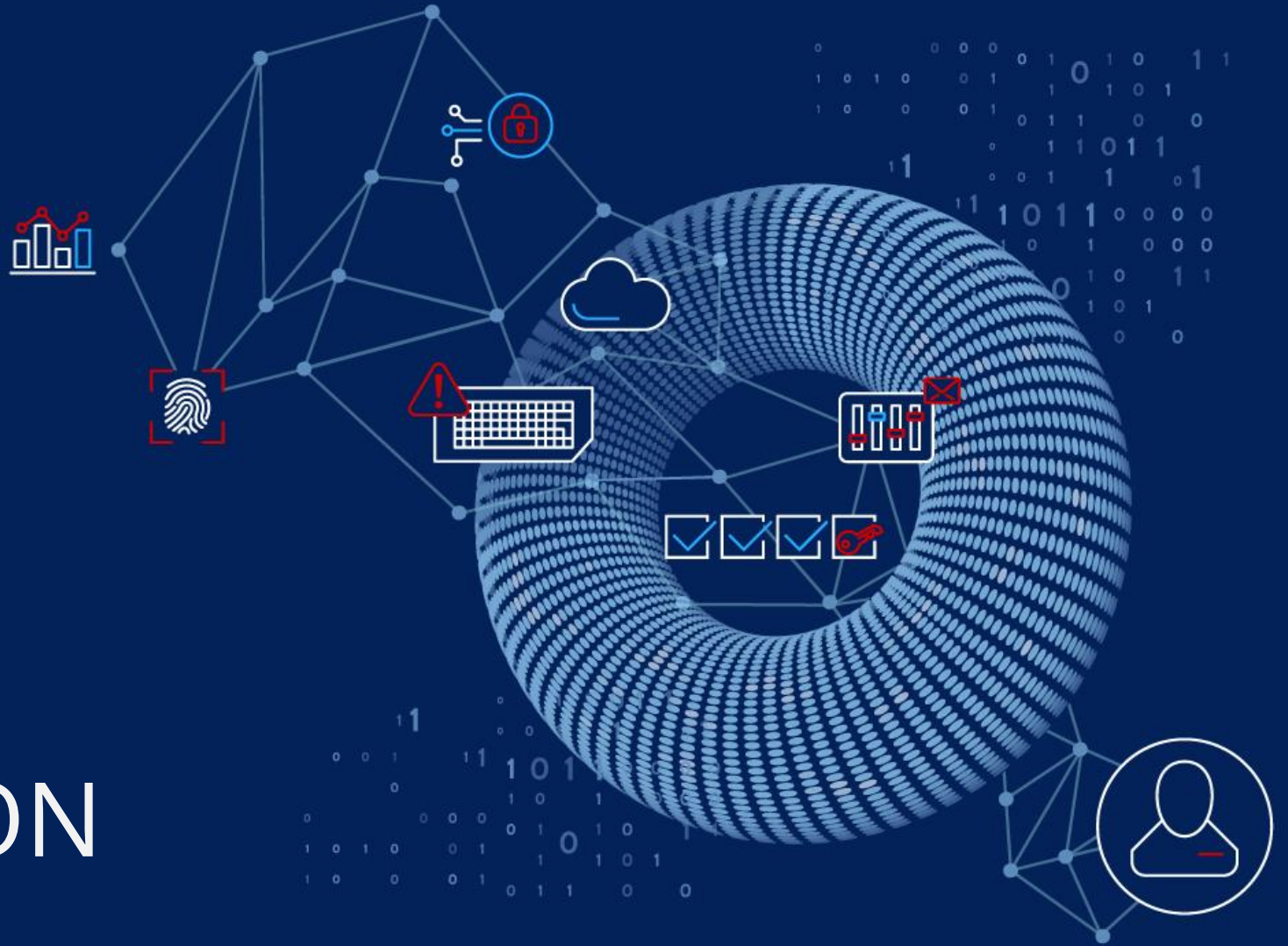
- ⚡ For MySQL the `log_bin_trust_function_creators` must be set to True
- ⚡ This is only required when creating a new database or performing an upgrade

```
set global log_bin_trust_function_creators = 1;
```

You can always restart Zabbix server to initialize full config sync.

03

PROXY CONFIGURATION



The situation with Zabbix proxies initially was also not optimal:

- ⚡ The default configuration update interval is 1 hour
- ⚡ Full synchronization is performed every time over the network
- ⚡ At least - the configuration is compressed before sending

```
### Option: ConfigFrequency
#       How often proxy retrieves configuration data from Zabbix Server in seconds.
#       For a proxy in the passive mode this parameter will be ignored.
# Mandatory: no
# Default:
ConfigFrequency=3600
```

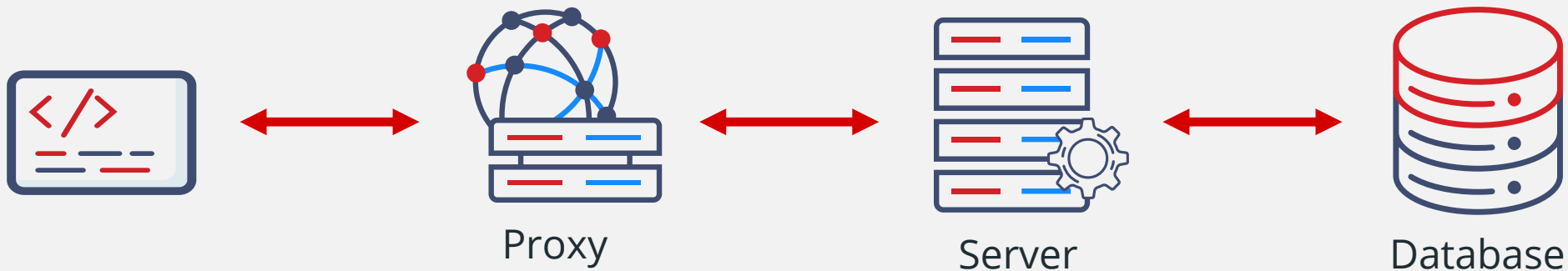
- ⚡ Zabbix server log file shows every synchronization

```
sending configuration data to proxy "Proxy Antwerp" at "10.10.10.10", datalen 3520178,
bytes 3253891 with compression ratio 10.6
sending configuration data to proxy "Proxy Brussels" at "10.10.10.20", datalen 2668090,
bytes 215960 with compression ratio 12.4
sending configuration data to proxy "Proxy Amsterdam" at "10.10.10.30", datalen 941666,
bytes 83642 with compression ratio 11.3
```


It was possible to initialize config sync manually:

- ⚡ Only for active proxies and only from the command line
- ⚡ The command must be executed from the shell of proxy machine

```
# zabbix_proxy -R config_cache_reload  
Runtime control command was forwarded successfully
```



- ⚡ For passive proxies there was no option to force config sync at all

```
# zabbix_proxy -R config_cache_reload  
Cannot perform configuration cache reloading on passive proxy
```

Zabbix 6.2 introduced new features:

- ⚡ Configuration refresh on demand for both active and passive proxies
- ⚡ Command can be executed from Zabbix server command line

```
# zabbix_server -R proxy_config_cache_reload  
Runtime control command was forwarded successfully
```

```
# zabbix_server -R proxy_config_cache_reload="Proxy Antwerp"  
Runtime control command was forwarded successfully
```

⚡ Configuration refresh using Zabbix frontend

<input type="checkbox"/> Name ▲	Mode	Encryption	Last seen (age)	Host count	Item count	Required vps	Hosts
<input type="checkbox"/> Proxy Amsterdam	Active	None	6s	453	12344	240.32	
<input checked="" type="checkbox"/> Proxy Antwerp	Passive	None	6s	53	450	68.89	
<input checked="" type="checkbox"/> Proxy Brussels	Active	None	6s	24	368	127.44	
<input type="checkbox"/> Proxy Eindhoven	Passive	None	6s	146	858	17.37	
<input type="checkbox"/> Proxy Luxembourg	Active	None	6s	221	12104	87.91	

2 selected

Refresh configuration Enable hosts Disable hosts Delete

Displaying 5 of 5 found

Zabbix 6.4 introduces proxy version control and compatibility:

- 🚨 Proxy version and compatibility is displayed in Zabbix frontend
- 🚨 All proxies till the previous LTS release have limited compatibility (will send data)
- 🚨 Configuration updates are possible only for current release

<input type="checkbox"/>	Name ▲	Mode	Encryption	Version	Last seen (age)	Host count	Item count	Required vps
<input type="checkbox"/>	Proxy Amsterdam	Active	None	6.4.0	6s	453	12344	240.32
<input type="checkbox"/>	Proxy Antwerp	Passive	None	6.4.1	6s	53	450	68.89
<input type="checkbox"/>	Proxy Brussels	Active	None	6.4.0	6s	24	368	127.44
<input type="checkbox"/>	Proxy Eindhoven	Passive	None	6.4.1	6s	146	858	17.37
<input type="checkbox"/>	Proxy Gent	Active	None	5.0.22	5s	76	4310	32.5
<input type="checkbox"/>	Proxy Luxembourg	Active	None	6.4.1	6s	5	168	2.93
<input type="checkbox"/>	Proxy Rotterdam	Active	None	6.2.6	6s	89	2165	18.49
<input type="checkbox"/>	Proxy Utrecht	Active	None	6.0.12	6s			

0 selected Refresh configuration Enable hosts Disable hosts Delete

Proxy version is not supported by server version 6.4.1.

Proxy version is outdated, only data collection and remote execution is available with server version 6.4.1.

The main proxy improvement of 6.4 is **proxy instant configuration sync**:

- Based on **object revisions** (32bit counter) in configuration cache
- Revisions are assigned to different entities (hosts, usermacros, etc.)
- Configuration cache size will increase because of this

Each entity has a revision number in configuration cache:

- When entity is updated, revision number is increased
- Updating items or triggers will change revision number for entire host
- Entities with revision number > proxy revision are sent to proxies as updates

Each proxy generates unique session id on start:

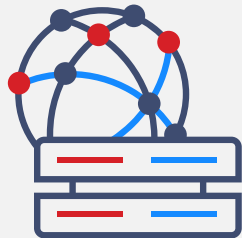
- This identifies current configuration session and is used for incremental updates
- Session id is recreated if proxy is restarted, and full config sync is initialized

Proxies have configuration revisions based on latest updates:

- ⚡ Proxy is sending current revision number when asking for configuration update
- ⚡ Server compares this number with its own cached revision number
- ⚡ Configuration changes are read from database
- ⚡ Server responds with configuration update and new revision number

Active proxy config update example

Config revision = 125



Proxy

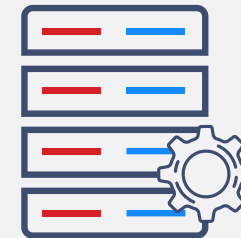
My current config revision is 125



Config update with revision 126



I have entities with
config revision = 126



Server

Active proxy config request example

```
{
  "request": "proxy config",
  "host": "Proxy Amsterdam",
  "version": "6.4.1",
  "session": "fd59a09ff4e9d1fb447de1f04599bcf6",
  "config_revision": 125
}
```

Zabbix server response example

```
{
  "full_sync": 0,
  "data": {
    "hosts": {
      "fields": ["hostid", "host", "status", "ipmi_authtype", "ipmi_privilege", "ipmi_username",
        "ipmi_password", "name", "tls_connect", "tls_accept", "tls_issuer", "tls_subject",
        "tls_psk_identity", "tls_psk"],
      "data": [
        [10084, "Linux server", 0, -1, 2, "", "", "Linux server", 1, 1, "", "", "", ""]
      ]
    },
    "config_revision": 126
  }
}
```

If there are no changes in revision, nothing is sent to the proxy:

- ⚡ Configuration update interval can be dramatically decreased
- ⚡ `ConfigFrequency` parameter is deprecated and replaced with `ProxyConfigFrequency`
- ⚡ The default setting is just 10 seconds

```
### Option: ProxyConfigFrequency
#       How often proxy retrieves configuration data from Zabbix Server in seconds.
#       For a proxy in the passive mode this parameter will be ignored.
#
# Mandatory: no
# Range: 1-3600*24*7
# Default:
ProxyConfigFrequency=10
```

Full configuration will be sent to proxies in the following scenarios:

- ⚡ On proxy restart (different session ID)
- ⚡ On Zabbix server restart (revision = 0)
- ⚡ On Zabbix HA cluster node failover (revision = 0)
- ⚡ If something goes wrong with revision numbers

Initial configuration sync

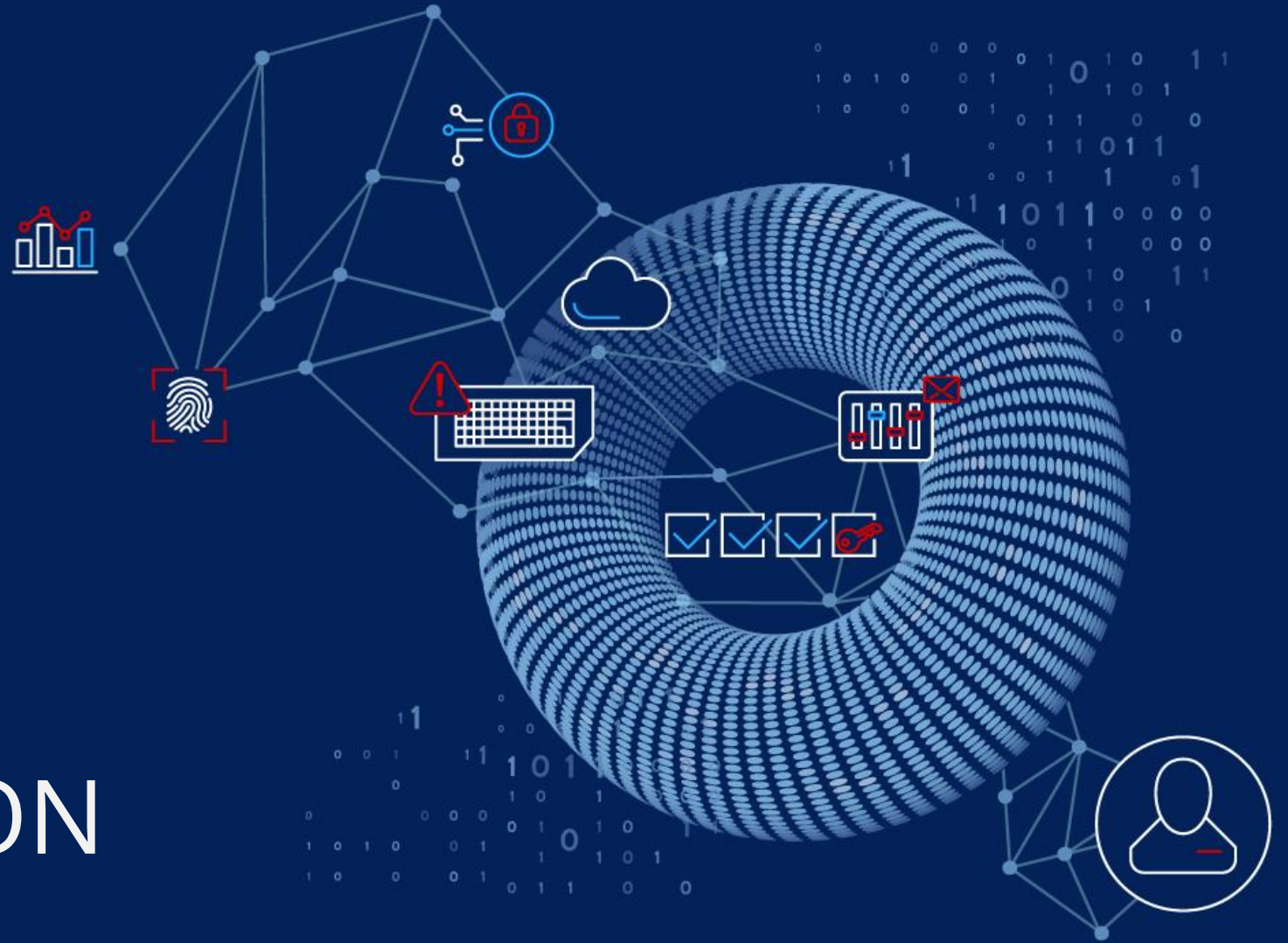
```
16294:20230412:105207.590 sending configuration data to proxy "Proxy Antwerp" at "10.10.10.10", datalen 1963336, bytes 198221 with compression ratio 10.2
```

Incremental configuration update

```
16294:20230412:105317.460 sending configuration data to proxy "Proxy Antwerp" at "10.10.10.10", datalen 24643, bytes 4237 with compression ratio 5.8
```

04

ACTIVE AGENT CONFIGURATION



Before Zabbix 6.4 active agent requested new configuration:

- ⚡ On Zabbix agent start
- ⚡ Every 120 seconds by default

```
### Option: RefreshActiveChecks
#       How often list of active checks is refreshed, in seconds.
#
# Mandatory: no
# Range: 60-3600
RefreshActiveChecks=120
```

This design had a few drawbacks:

- ⚡ Long interval between updates
- ⚡ To force configuration update agent needs to be restarted
- ⚡ Full copy of configuration was sent every time over the network

Zabbix 6.4 has seriously improved this design:

- ⚡ Configuration is sent on Zabbix active agent start
- ⚡ Agent checks for new config updates every 5 seconds by default
- ⚡ The new copy of configuration is sent only if changes are detected
- ⚡ Autoregistration is now cached in the configuration cache

```
### Option: RefreshActiveChecks
#       How often list of active checks is refreshed, in seconds.
#
# Mandatory: no
# Range: 1-86400
RefreshActiveChecks=5
```

Config revision and session IDs are used

- ⚡ Agent sends configuration revision 0 on restart
- ⚡ Full configuration is sent on Zabbix server / proxy restart
- ⚡ Full configuration is sent to pre 6.4 agents (no session id)

Agent config request

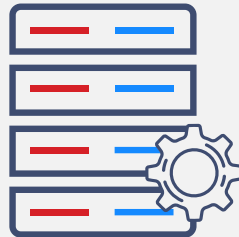
```
{
  "request": "active checks",
  "host": "Linux server",
  "host_metadata": "metadata",
  "config_revision": 0,
  "session": "e3dcbd9ace2c9694e1d7bbd030eeef6e"
}
```



Config request



Server response



```
{
  "response": "success",
  "data": [
    {
      "key": "agent.ping",
      "delay": 60,
      ...
    }
  ],
  "config_revision": 0
}
```

Send config

```
{"response":"success"}
```

No changes

```
{"response":"success"}
```

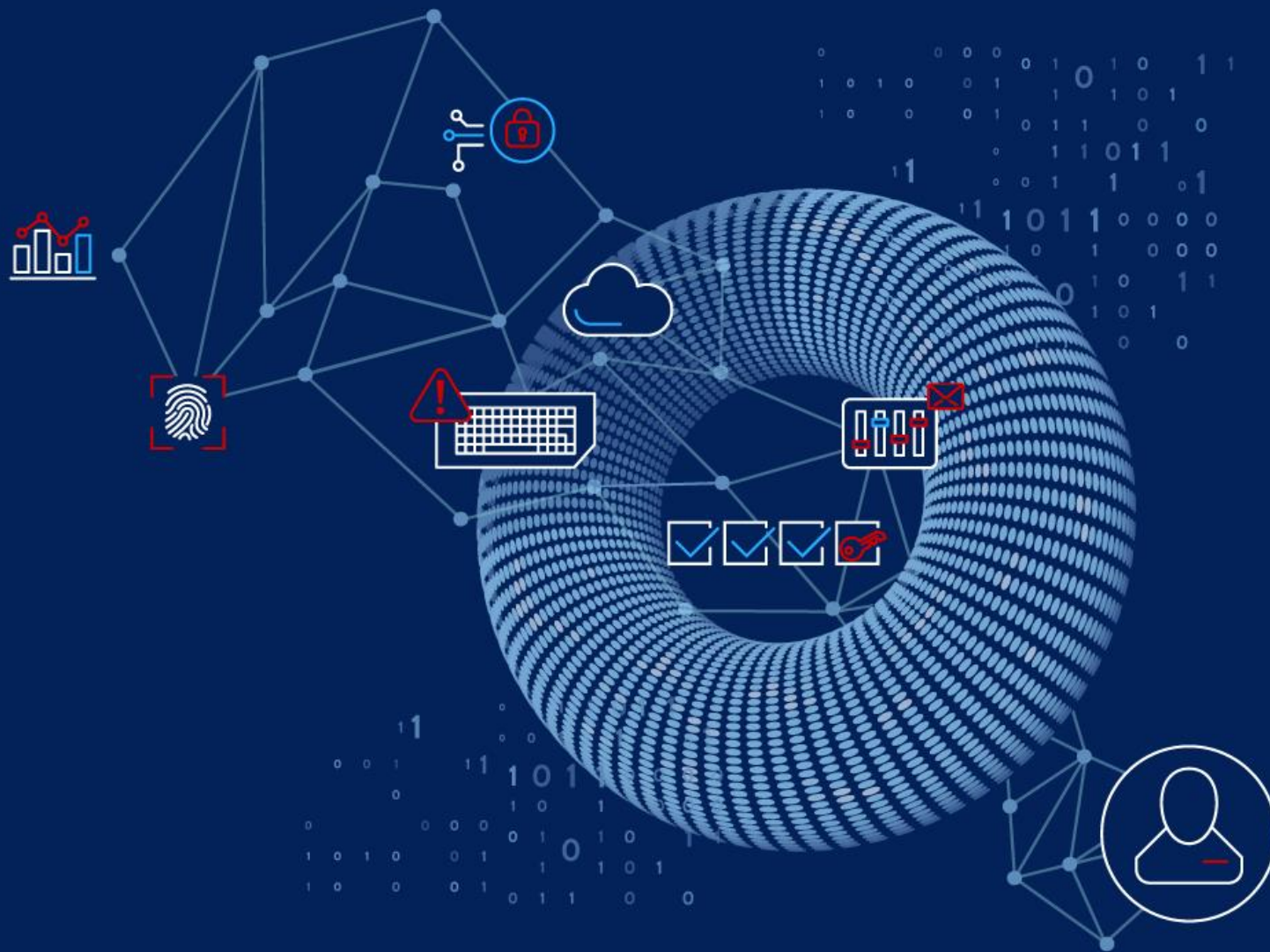
No changes

```
{
  "response": "success",
  "data": [
    {
      "key": "agent.ping",
      "delay": 10,
      ...
    }
  ],
  "config_revision": 1
}
```

Send config

05

CONCLUSION



Time benefits:

- ⚡ Significantly decreased configuration sync times for both server and proxies
- ⚡ The default setting is 10 seconds for both and can be decreased if required

Network traffic benefits:

- ⚡ Reduced network traffic between proxies and server
- ⚡ Proxy pollers are no longer busy sending full configuration updates

Database load benefits:

- ⚡ No full configuration request every 1 minute for server
- ⚡ No full configuration request for every proxy

ZABBIX

6.4

Thank you

