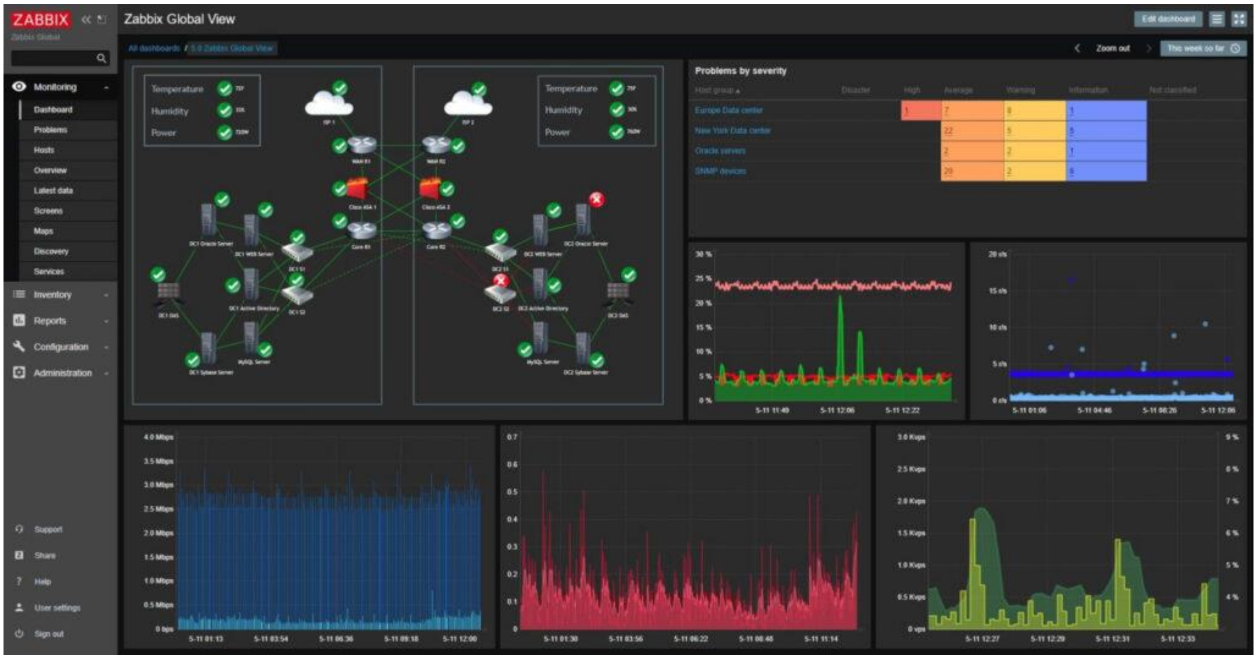
How to Install Zabbix 5 on CentOS 8 / RHEL 8 in 10 minutes!

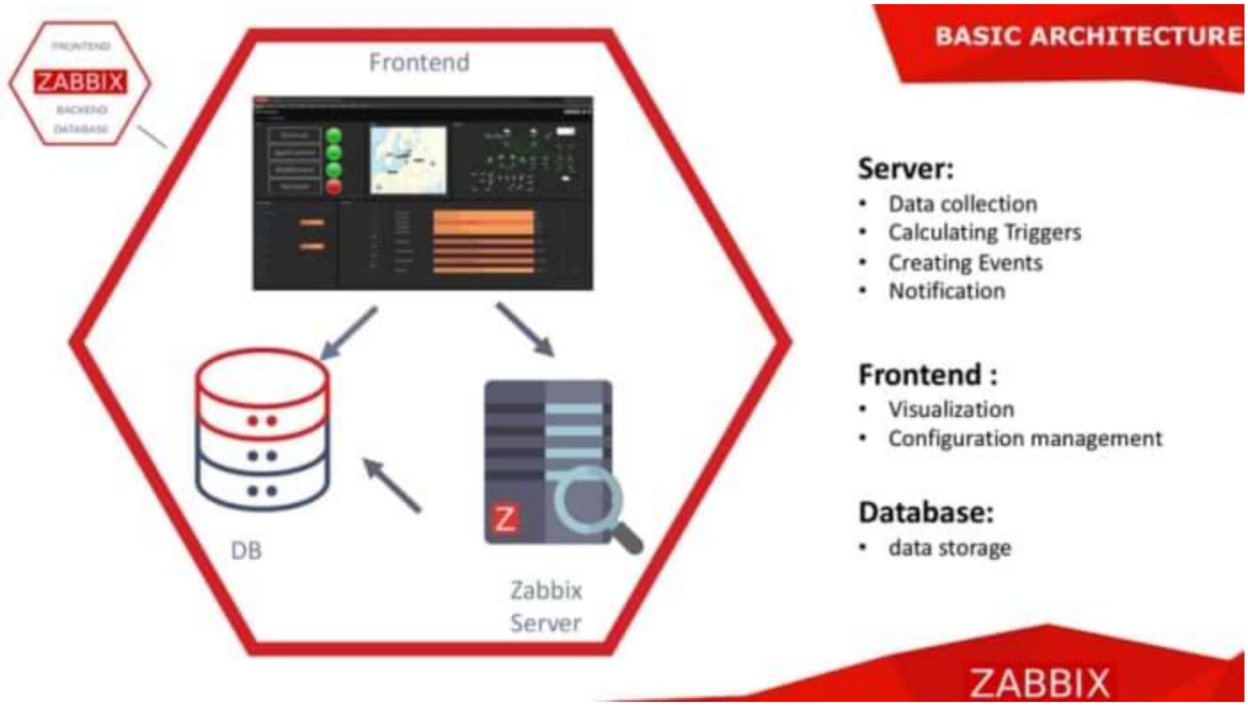
Zabbix server is installable on any Linux distribution, but in this tutorial, you will learn how to install the latest version – Zabbix Server 5.0 on CentOS 8 / RHEL 8.

Zabbix is 100% free open-source ultimate enterprise-level software designed for monitoring availability and performance of IT infrastructure components and services. You can read a case-study about Zabbix popularity and find out more about open-source movement in this article.



Zabbix 5 Dashboard

Enough of talk lets do some work! First, you will install and configure Zabbix server, then a database and lastly the frontend – check the picture bellow for a better understanding of Zabbix architecture.



Picture showing Zabbix architecture

This guide is for installing **Zabbix monitoring system (Server)** on CentOS / RHEL, while guide for installing **Zabbix-Proxy** on CentOS / RHEL can be found on this link.

Step 1: Set SELinux to permissive mode

Configure SELinux to work in permissive mode:

```
setenforce 0 && sed -i 's/^SELINUX=.*/SELINUX=permissive/g' /etc/selinux/config
```

This way, SELinux will not block antyhing, but the audit log will fill up with what would have been denied. And later in step 12, we can create an SELinux policy based on that.

Step 2: Install Zabbix server, frontend, and agent

Setup Zabbix 5 .rpm package on CentOS 8, clean repo and install Zabbix server, frontend, and agent:

```
Zabbix 5.0 LTS version (supported until May 31, 2025)

rpm -Uvh https://repo.zabbix.com/zabbix/5.0/rhel/8/x86_64/zabbix-release-5.0-
1.el8.noarch.rpm
dnf clean all
dnf -y install zabbix-server-mysql zabbix-web-mysql zabbix-apache-conf zabbix-
agent
```

You can find more information about Zabbix's life cycle and release policies on the official website.

Step 3: Install and configure database

In this installation, I will use password rootDBpass as root password and zabbixDBpass as Zabbix password for DB. Consider changing your password for security reasons.

a. Install MariaDB

```
dnf -y install mariadb-server && systemctl start mariadb && systemctl enable
mariadb
```

b. Reset root password for database

Secure MySQL by changing the default password for MySQL root:

```
mysql_secure_installation
Enter current password for root (enter for none): Press the Enter
Set root password? [Y/n]: Y
New password: <Enter root DB password>
Re-enter new password: <Repeat root DB password>
Remove anonymous users? [Y/n]: Y
Disallow root login remotely? [Y/n]: Y
Remove test database and access to it? [Y/n]:  Y
Reload privilege tables now? [Y/n]:  Y
```

c. Create database

```
mysql -uroot -p'rootDBpass' -e "create database zabbix character set utf8
collate utf8_bin;"
mysql  -uroot  -p'rootDBpass'  -e  "grant  all  privileges  on  zabbix.*  to
zabbix@localhost identified by 'zabbixDBpass';"
```

d. Import initial schema and data

Temporary disable strict mode (ZBX-16465) to avoid MySQL error “`ERROR 1118 (42000) at line 1284: Row size too large (> 8126)`”:

```
mysql -uroot -p'rootDBpass' zabbix -e "set global innodb_strict_mode='OFF';"
```

Import database shema for Zabbix server (could last up to 5 minutes):

```
zcat  /usr/share/doc/zabbix-server-mysql*/create.sql.gz  |  mysql  -uzabbix  -
p'zabbixDBpass' zabbix
```

Enable strict mode:

```
mysql -uroot -p'rootDBpass' zabbix -e "set global innodb_strict_mode='ON';"
```

e. Enter database password in Zabbix configuration file

Open `zabbix_server.conf` file with command:

```
sudo nano /etc/zabbix/zabbix_server.conf
```

and add database password in this format anywhere in file:

```
DBPassword=zabbixDBpass
```

Save and exit file (**ctrl+x**, followed by **y** and **enter**).

Step 4: Start Zabbix server and agent processes

```
systemctl restart zabbix-server zabbix-agent
systemctl enable zabbix-server zabbix-agent
```

Step 5: Configure firewall

```
firewall-cmd --add-service={http,https} --permanent
firewall-cmd --add-port={10051/tcp,10050/tcp} --permanent
firewall-cmd –reload
```

Step 6: Configure Zabbix frontend

a. Configure PHP for Zabbix frontend

Edit file “`/etc/php-fpm.d/zabbix.conf`” with command:

```
sudo nano /etc/php-fpm.d/zabbix.conf
```

Uncomment line in zabbix.conf that starts with “`; php_value date.timezone Europe/Riga`” by removing symbol “`;`” and set the right timezone for your country, for example:

```
php_value date.timezone Europe/Amsterdam
```

Save and exit file (**ctrl**+**x**, followed by **y** and **enter**)

b. Restart Apache web server and make it start at system boot

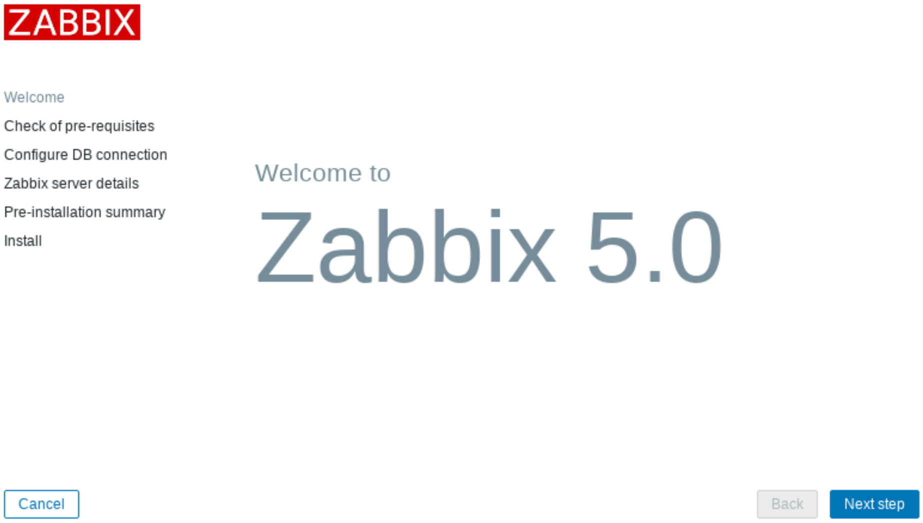
```
systemctl restart httpd php-fpm
systemctl enable httpd php-fpm
```

c. Configure web frontend

Connect to your newly installed Zabbix frontend using URL “*http://server_ip_or_dns_name/zabbix*” to initiate the Zabbix installation wizard.

For example, URL to use would be “*http://192.168.1.161/zabbix*” because Zabbix was installed on the server with IP address 192.168.1.161 (you can find the IP address of your server by typing “`ip a`” command in the terminal).

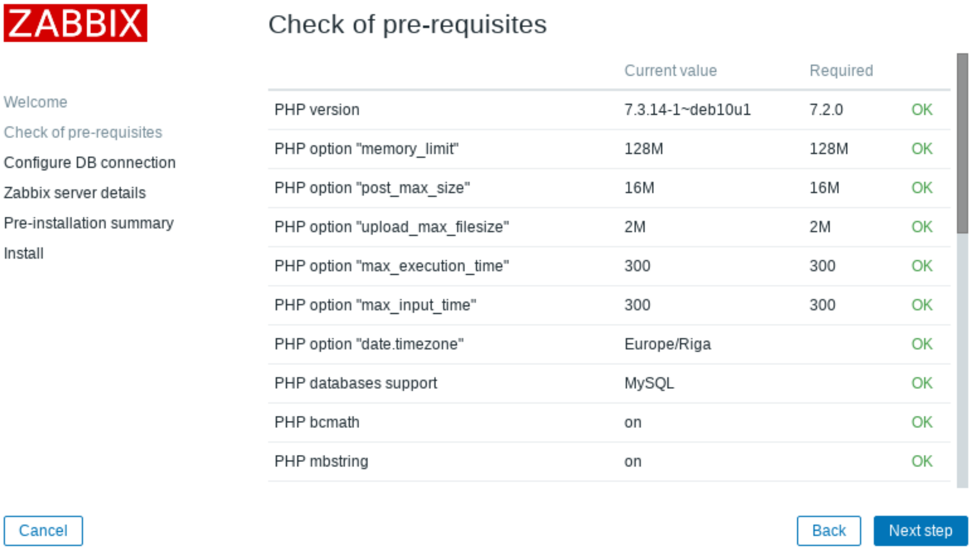
Basically, in this wizard **you only need to enter a password for Zabbix DB user** and for everything else just click “*Next step*“. In this guide, *zabbixDBpass* was used as a database password, but if you set something else, be sure to enter the correct password when prompted by the wizard.



Welcome to

Zabbix 5.0

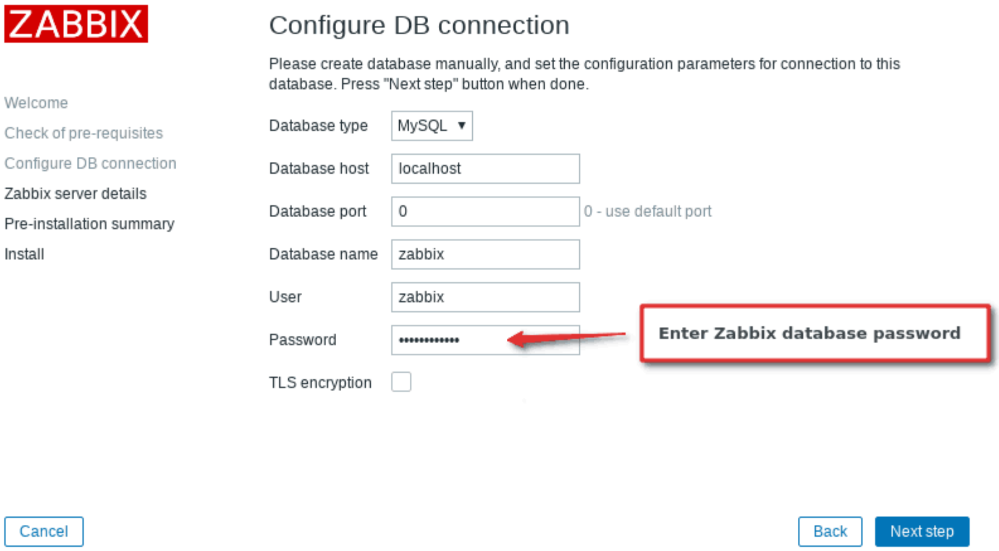
1. Installation step: Welcome screen



Check of pre-requisites

	Current value	Required	
PHP version	7.3.14-1~deb10u1	7.2.0	OK
PHP option "memory_limit"	128M	128M	OK
PHP option "post_max_size"	16M	16M	OK
PHP option "upload_max_filesize"	2M	2M	OK
PHP option "max_execution_time"	300	300	OK
PHP option "max_input_time"	300	300	OK
PHP option "date.timezone"	Europe/Riga		OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK

2. Installation step: Pre-requisites check



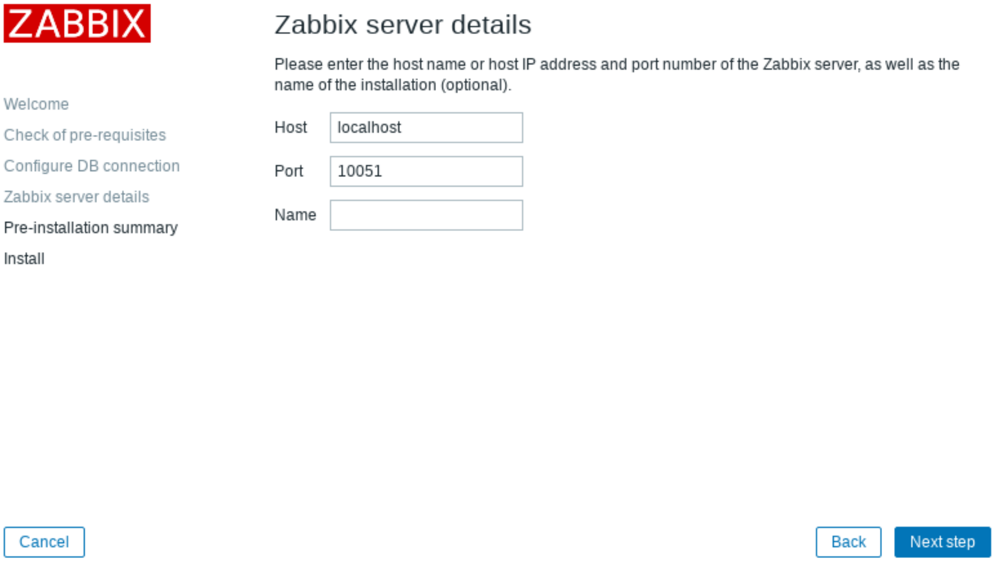
Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type	MySQL ▾
Database host	localhost
Database port	0 (0 - use default port)
Database name	zabbix
User	zabbix
Password	••••••••••••
TLS encryption	□

Enter Zabbix database password

3. Installation step: Configure DB connection

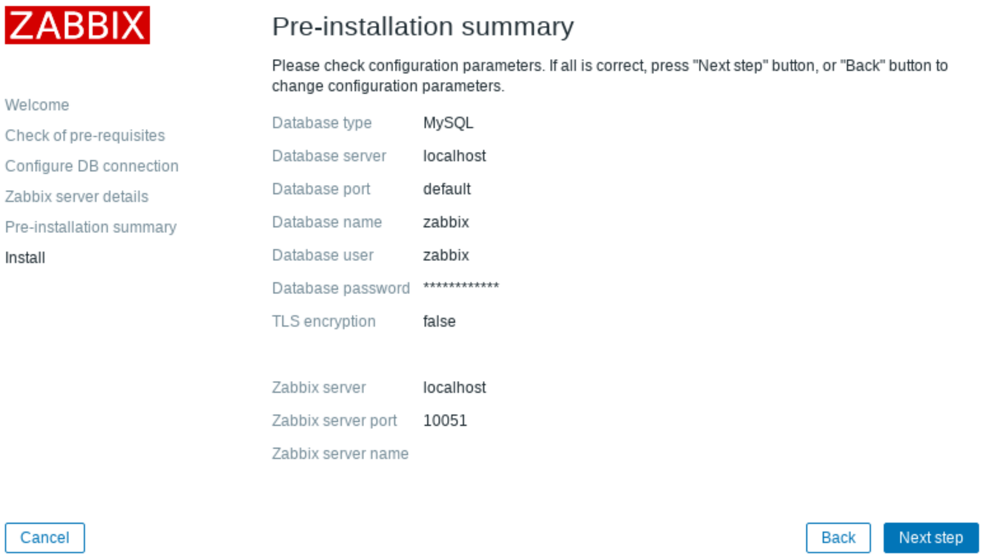


Zabbix server details

Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional).

Host	localhost
Port	10051
Name	

4. Installation step: Configure Zabbix server

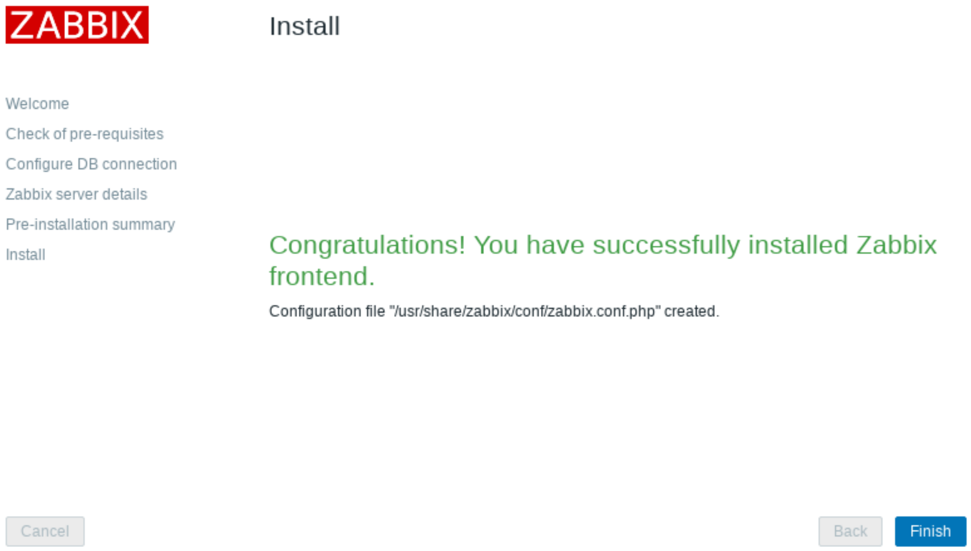


Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Database type	MySQL
Database server	localhost
Database port	default
Database name	zabbix
Database user	zabbix
Database password	************
TLS encryption	false
Zabbix server	localhost
Zabbix server port	10051
Zabbix server name	

5. Installation step: Pre-installation summary



Install

Congratulations! You have successfully installed Zabbix frontend.

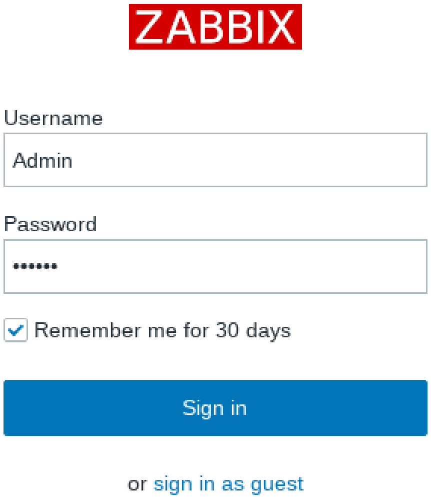
Configuration file "/usr/share/zabbix/conf/zabbix.conf.php" created.

6. Installation step: Finish

That’s it, you have installed Zabbix monitoring system!

Step 7: Login to frontend using Zabbix default login credentials

Use Zabbix default admin username “Admin” and password “zabbix” (without quotes) to login to Zabbix frontend at URL “http://server_ip_or_dns_name/zabbix” via your browser.



Username

Admin

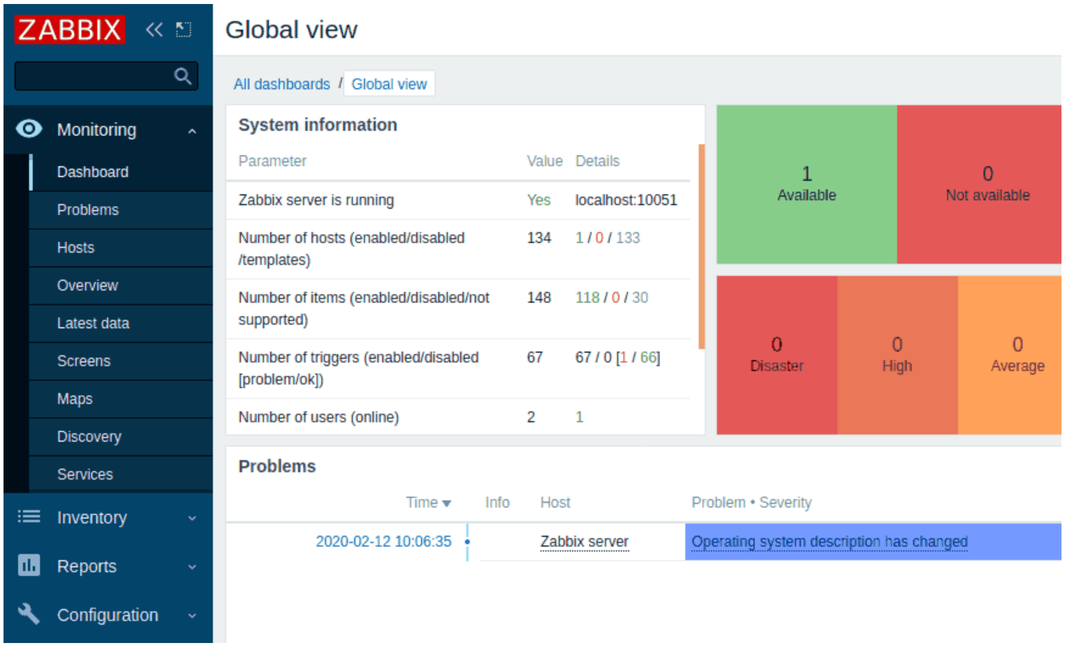
Password

••••••

☑ Remember me for 30 days

ZABBIX LOGIN PAGE

For example, Zabbix was installed on server 192.168.1.161 so you will enter in your browser the URL field http://192.168.1.161/zabbix (you can find the IP address of your server by typing “`ip a`” command in the terminal)



System information

Parameter	Value	Details
Zabbix server is running	Yes	localhost:10051
Number of hosts (enabled/disabled /templates)	134	1 / 0 / 133
Number of items (enabled/disabled/not supported)	148	118 / 0 / 30
Number of triggers (enabled/disabled [problem/ok])	67	67 / 0 [1 / 66]
Number of users (online)	2	1

Problems

Time ▾	Info	Host	Problem • Severity
2020-02-12 10:06:35		Zabbix server	Operating system description has changed

Zabbix 5.0 Dashboard

CONGRATULATIONS!

You have successfully installed Zabbix 5 on CentOS 8 and now you can monitor anything!
No need to change anything else as other steps are optional.

CONTINUE TO LEARN MORE:

How to create MySQL partitions on History and Events tables
Optimizing Zabbix server and MySQL database
Managing Zabbix / MySQL / Apache service
Enable and configure SELinux on Zabbix

Step 8: Create MySQL partitions on History and Events tables

Zabbix's housekeeping process is responsible for deleting old trend and history data. Removing old data from the database using SQL delete query can negatively impact database performance. Many of us have received that annoying alarm "`Zabbix housekeeper processes more than 75% busy`" because of that.

That problem can be easily solved with the database partitioning. Partitioning creates tables for each hour or day and drops them when they are not needed anymore. SQL DROP is way more efficient than the DELETE statement.

You can partition MySQL tables in 5 minutes using this simple guide.

Step 9: Optimizing Zabbix Server (optional)

Don't bother with this optimization if you are monitoring a small number of devices, but if you are planning to monitor a large number of devices then continue with this step.

Open "`zabbix_server.conf`" file with command:

```
sudo nano /etc/zabbix/zabbix_server.conf
```

and add this configuration anywhere in file:

```
StartPollers=100
```

```
StartPollersUnreachable=50
StartPingers=50
StartTrappers=10
StartDiscoverers=15
StartPreprocessors=15
StartHTTPPollers=5
StartAlerters=5
StartTimers=2
StartEscalators=2
CacheSize=128M
HistoryCacheSize=64M
HistoryIndexCacheSize=32M
TrendCacheSize=32M
ValueCacheSize=256M
```

Save and exit file (**ctrl**+**x**, followed by **y** and **enter**).

This is not a perfect configuration, keep in mind that you can optimize it even more. Let's say if you don't use ICMP checks then set the "`StartPingers`" parameter to 1 or if you don't use active agents then set "`StartTrappers`" to 1 and so on. You can find out more about the parameters supported in a Zabbix server configuration file in the official documentation.

If you try to start the Zabbix server you will receive an error "`[Z3001] connection to database 'Zabbix' failed: [1040] Too many connections`" in the log "`/var/log/zabbix/zabbix_server.log`" because we are using more Zabbix server processes than MySQL can handle. We need to increase the maximum permitted number of simultaneous client connections and optimize MySQL – so move to the next step.

Step 10: Optimizing MySQL/MariaDB database (optional)

a. Create custom MySQL configuration file

Create file "`10_my_tweaks.cnf`" with "`sudo nano /etc/my.cnf.d/10_my_tweaks.cnf`" and paste this configuration:

```
[mysqld]
max_connections                = 404
innodb_buffer_pool_size        = 800M

innodb-log-file-size           = 128M
innodb-log-buffer-size         = 128M
innodb-file-per-table          = 1
innodb_buffer_pool_instances   = 8
innodb_old_blocks_time         = 1000
innodb_stats_on_metadata       = off
innodb-flush-method            = O_DIRECT
innodb-log-files-in-group      = 2
```

```
innodb-flush-log-at-trx-commit = 2

tmp-table-size                  = 96M
max-heap-table-size             = 96M
open_files_limit                = 65535
max_connect_errors              = 1000000
connect_timeout                 = 60
wait_timeout                    = 28800
```

Save and exit the file (**ctrl+x**, followed by **y** and **enter**) and set the correct file permission:

```
sudo chown mysql:mysql /etc/my.cnf.d/10_my_tweaks.cnf
sudo chmod 644 /etc/my.cnf.d/10_my_tweaks.cnf
```

Two things to remember!

Configuration parameter **max_connections** must be larger than the total number of all Zabbix proxy processes plus 150. You can use the command below to automatically check the number of Zabbix processes and add 150 to that number:

```
 egrep "^Start.+=[0-9]" /etc/zabbix/zabbix_server.conf | awk -F "=" '{s+=$2} END {print s+150}'
 295
```

The second most important parameter is **innodb_buffer_pool_size**, which determines how much memory can MySQL get for caching InnoDB tables and index data. You should set that parameter to 70% of system memory if only database is installed on server.

However, in this case, we are sharing a server with Zabbix and Apache processes so you should set **innodb_buffer_pool_size** to 40% of total system memory. That would be 800 MB because my CentOS server has 2 GB RAM.

I didn't have any problems with memory, but if your Zabbix proxy crashes because of lack of memory, reduce "`innodb_buffer_pool_size`" and restart MySQL server.

Note that if you follow this configuration, you will receive "`Too many processes on the Zabbix server`" alarm in Zabbix frontend due to the new Zabbix configuration. It is safe to increase the trigger threshold or turn off that alarm (select "Problems" tab → left click on the alarm → select "Configuration" → remove the check from "Enabled" → hit the "Update" button)

b. Restart Zabbix Server and MySQL service

Stop and start the services in the same order as below:

```
sudo systemctl stop zabbix-server
sudo systemctl stop mysql
sudo systemctl start mysql
sudo systemctl start zabbix-server
```

Step 11: How to manage Zabbix / MySQL / Apache service

Sometimes you will need to check or restart Zabbix, MySQL or Apache service – use commands below to do that.

```
Zabbix Server
sudo systemctl <status/restart/start/stop> zabbix-server

MySQL/MariaDB Server
sudo systemctl <status/restart/start/stop> mysql

Apache Server
sudo systemctl <status/restart/start/stop> httpd

PHP FastCGI Process Manager
sudo systemctl <status/restart/start/stop> php-fpm

Zabbix Agent
sudo systemctl <status/restart/start/stop> zabbix-agent
```

Step 12: Enable and configure SELinux on Zabbix

While it is acceptable to disable SELinux in a lab environment, depending on the requirements of the local security IT team, you may need to enable and configure SELinux in your production environment.

At the beginning of this guide, we did not turn off SELinux completely but set to work in the permissive mode which means it will log all the security errors but will not block anything.

If you accidentally left it in enforcing mode then you will receive the “`Zabbix server is not running: the information displayed may not be current`” warning on the Zabbix frontend and “`cannot set resource limit: [13] Permission denied`” in the log file.

Don’t worry, this can be easily fixed, so without further delay, let’s configure SELinux for Zabbix!

a) SELinux: Allow http daemon to connect to Zabbix:

Enable SELinux boolean “`httpd_can_connect_zabbix`” that will allow http daemon to connect to Zabbix:

```
setsebool -P httpd_can_connect_zabbix 1
```

b) SELinux: Allow Zabbix to connect to all TCP ports:

Enable SELinux Boolean “`zabbix_can_network`” that will allow Zabbix to connect to all TCP ports :

```
setsebool -P zabbix_can_network 1
```

c) Set SELinux to work in enforcing mode

Turn on SELinux by setting it to work in enforcing mode:

```
setenforce 1 && sed -i 's/^SELINUX=.*/SELINUX=enforcing/g'
/etc/selinux/config
```

And check SELinux status :

```
# sestatus
SELinux status: enabled
SELinuxfs mount: /sys/fs/selinux
SELinux root directory: /etc/selinux
Loaded policy name: targeted
Current mode: enforcing
Mode from config file: enforcing
Policy MLS status: enabled
Policy deny_unknown status: allowed
Memory protection checking: actual (secure)
Max kernel policy version: 31
```

d) Create additional SELINUX policy for Zabbix

Just in case, we will create an additional SELinux policy for each error in the audit log (“/var/log/audit/audit.log“)

To do this, we will need the policycoreutils-python tool, so let’s install it:

```
dnf -y install policycoreutils-python-utils
```

Create a custom policy package:

```
grep "denied.*zabbix" /var/log/audit/audit.log | audit2allow -M zabbix_policy
```

Install custom SELinux policy package:

```
semodule -i zabbix_policy.pp
```

Well done! You have configured SELinux for Zabbix!

Step 13: Upgrade between minor versions

The following upgrade procedures are about Zabbix upgrade. Zabbix’s team releases new minor versions at least once a month. The main purpose of minor upgrades is to fix bugs (hotfix) and sometimes even bring new functionality. **Therefore, try to do a minor upgrade of Zabbix at least once a month.**

There is no need for backups when doing a minor upgrade, they are completely safe. With this command you can easily upgrade smaller versions of 5.0.x (for example, from 5.0.1 to 5.0.3):

```
dnf upgrade 'zabbix-*'
```

And restart Zabbix server afterward:

```
systemctl restart zabbix-server
```