

## ZABBIX BLOG



How To

## Backups to the rescue!



By Nathan Liefting — September 30, 2022

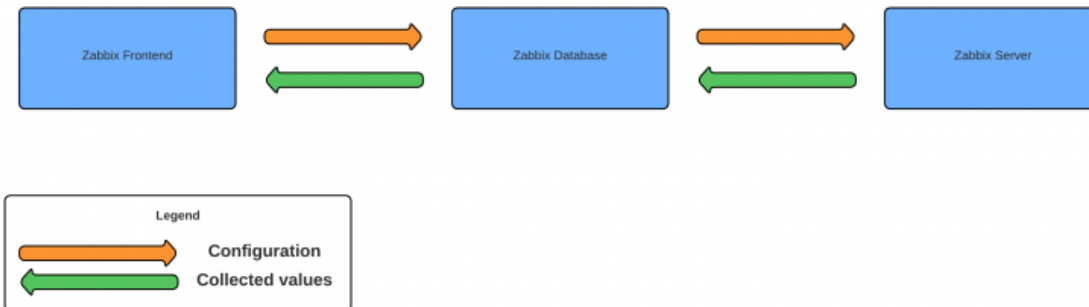
In this blog post, you will learn how to set up backups for your Zabbix environment. There's a wide variety of different options when it comes to taking backups of our Zabbix environment, for us, it will just be a matter of choosing the right fit.

### Table of Contents

- 1. Introduction
- 2. How to
  - 2.1. MySQL backups
  - 2.2. PostgreSQL backups
  - 2.3. What about the configuration files?
  - 2.4. Don't forget the import/export files!
    - 2.4.1. Frontend backups
    - 2.4.2. API backups
  - 2.5. High availability
- 3. Conclusion

taking backups of our Zabbix environment is an important part of running a production Zabbix environment, as we do want to be prepared for a possible issue that might corrupt or even lose our data. It's always a possibility and as such we should be prepared.

For Zabbix, there are a few different methods on how to take backups and it all starts at the database level. Both the Zabbix frontend as well as the Zabbix server write their data into the Zabbix database as we can see in the illustration below:



This means that both our configuration as well as all of our collected values are present in the same Zabbix database and if we take a database backup, we back up (almost) everything we need. So, let's start there and have a look at how we can make a database backup.

## How to

### MySQL backups

Let's start with the most used variant of Zabbix databases: MySQL and it's forks like MariaDB and Percona. All of them can easily be backed up using built-in functionality like the **MySQL Dump** command and we can then use other industry standards to get things going. First, we have to understand the tables in our database though. Most of the tables in your Zabbix environment contain configuration data and as such, they are all important to backup. There are a few tables that we need to consider, however, as they can contain Giga or even Terabytes of data. These are the History, Trends and Events tables:

#### TABLE

history

history\_uint

history\_text

history\_str

history\_log

trends

trends\_uint

events

It is possible to omit these tables from your backup and make smaller, more manageable backups. To make the backup we can then start using tools like **MySQL Dump**:

import command or simply using the `cat` command:

```
[root@summit-zbx-01 ~]# mysqlimport -p --use-threads=4 zabbix /mnt/backup/*.sql
Enter password: █
```

```
[root@summit-zbx-01 ~]# cat /mnt/backup/*.sql | mysql -p zabbix
```

Do not forget, taking and importing large backups can take a long time. This completely depends on your MySQL database performance tuning settings as well as the underlying resources like CPU, Memory and Disk I/O. Also, make sure to check out the MySQL documentation:

MySQL Dump: <https://dev.mysql.com/doc/refman/8.0/en/mysqldump.html> /  
<https://mariadb.com/kb/en/making-backups-with-mysqldump/>

MySQL Import: <https://dev.mysql.com/doc/refman/8.0/en/mysqlimport.html> /  
<https://mariadb.com/kb/en/mysqlimport/>

Alternatively, it's also possible to create backups using tools like **xtrabackup** and **mariadbbackup**.

## PostgreSQL backups

We can actually use the same kinds of methods for the PostgreSQL backups. Keep the required tables in mind and fire away with the built-in tools:

```
[root@summit-zbx-01 ~]# pg_dump zabbix > zabbix_db
```

Then we can restore it by loading the file into postgres:

```
[root@summit-zbx-01 ~]# psql zabbix < zabbix_db
```

## What about the configuration files?

Once we have a database backup, everything is backed up, right? Well, almost everything. With just a database backup we are quite safe, but (and this is oftentimes overlooked) there are a lot of configuration files and perhaps even custom scripts we need to take into account! There are three parts to this story – the Zabbix server, the Zabbix frontend, and also the Zabbix additional components. All of them have their own set of configuration files and locations that are used for storing custom scripts.

The Zabbix frontend location and configuration files can be different, depending on the environment, as we have a few choices to make. Are we running **Apache** or **Nginx**? On what **Linux** distribution? All of these have to be considered when making configuration backups. In general, the locations for the configuration would be:

```
/etc/nginx/
/etc/httpd/
/etc/apache2
```

There's also a symlink to the Zabbix frontend configuration file located in `/etc/zabbix/` but we will get to that one in a bit.

```
/usr/lib/zabbix
```

Let's add them to the list and find a method to back up these files. Crontab is a built-in tool that we can use, but there are definitely other (perhaps better) solutions out there. Let's add the following to cron:

```
0 0 * * 0 sudo find /mnt/backup/config_files/ -mtime +180 -type f -delete
0 0 * * 1-7 cd /mnt/backup/config_files/ && cp -r /usr/lib/zabbix/*
0 0 * * 1-7 cd /mnt/backup/config_files/ && cp -r /etc/zabbix/*
0 0 * * 1-7 cd /mnt/backup/config_files/ && cp -r /etc/nginx/*
```

I also added a **find** command here, which will serve as our roll-over or rotation toll. It will find files older than 180 days and delete them from **/mnt/backup/config\_files/**. Make sure to pick a good (network) folder to store these files as it's important to keep these safe. Feel free to change the number of days you'd like to store the files for.

What about the additional components like Zabbix proxy, Zabbix Java gateway and Zabbix web service (used for PDF reporting)? Well, these have configuration files as well. Make sure to run a backup on the devices running these additional components. As for Zabbix proxies – they have the same file locations as Zabbix server:

```
0 0 * * 0 sudo find /mnt/backup/config_files/ -mtime +180 -type f -delete
0 0 * * 1-7 cd /mnt/backup/config_files/ && cp -r /usr/lib/zabbix/*
0 0 * * 1-7 cd /mnt/backup/config_files/ && cp -r /etc/zabbix/*
```

For Zabbix Java gateway and Zabbix web service, we can omit the **/usr/lib/zabbix/** folder.

## Don't forget the import/export files!

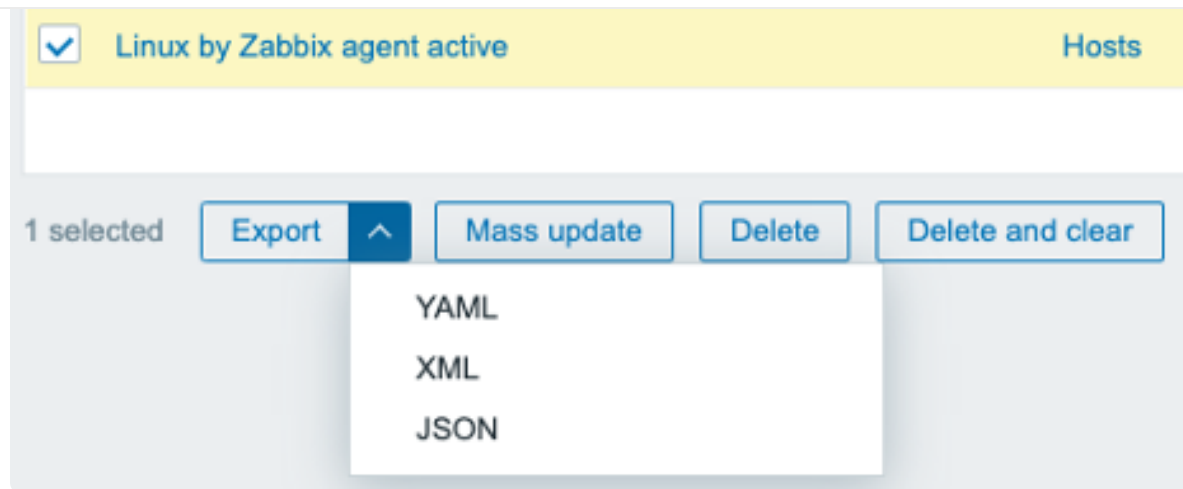
In general, database backups are slow to make, but also slow to import back unless we do not include the history/trends in the backup. But even then, restoring an entire database simply because someone made an error on a single template is a hassle. Zabbix ships with the built-in **frontend export functionality**, allowing us to export (and then import) entire parts of the configuration instantly! We can use these for a number of different parts of the configuration:

- Hosts
- Templates
- Media types
- Maps
- images
- Host groups (API ONLY)
- Template groups (API ONLY)

All of these are available through the Zabbix API allowing us to choose whether we do a manual configuration backup from the frontend, as well as providing us with automation options using that API. You could even manage and update your Zabbix configuration from GIT entirely if you write the right scripts for this.

## Frontend backups

To run an export from the frontend simply go to one of the supported sections like **Configuration** | **Templates** and select the export data format. When selecting multiple entities, keep in mind that they will



We can then make our edits and import files from the frontend as well:

## Import

? x

\* Import file  zbx\_export\_templates (12).yaml

Rules	Update existing	Create new	Delete missing
Template groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Host groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Value mappings	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template dashboards	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template linkage		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Web scenarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

For Templates this will even result in a nice **diff** pop-up window, detailing all the **changes**, **deletes** and **additions** to the templates:



Request body

```
1  {
2      "jsonrpc": "2.0",
3      "method": "configuration.export",
4      "params": {
5          "options": {
6              "hosts": [
7                  "10161"
8              ]
9          },
10         "format": "yaml"
11     },
12     "auth": "038e1d7b1735c6a5436ee9eae095879e",
13     "id": 1
14 }
```

The response will then look something like this:

```
1  {
2      "jsonrpc": "2.0",
3      "result": "zabbix_export:\n version: '6.2'\n date: '2022-09-19T14:47:45Z'\n host_groups:\n
         -\n      uuid: e0dc612216ab4f74917fd04aa38eb8\n      name: Servers/OS/Windows\n hosts:\n
         -\n      host: 'BUF mssql01'\n      name: 'BUF mssql01'\n      proxy:\n      name:\n
         customer1-proxy-1\n      groups:\n      -\n      name: Servers/OS/Windows\n
         inventory_mode: DISABLED\n",
4      "id": 1
5  }
```

But this feature really starts to shine once we combine it with our own automation scripts. Use it wisely!

## High availability

So, what about high availability? Isn't that some form of a backup?

Well yes and no. High availability is not an "IT backup" in the form of making sure we can recover something that is broken. But it is a backup in the way that if a Zabbix server instance fails, another one takes over for it. HA is somewhat out of scope for this blog post, but it's still worth mentioning. There are several solutions to set up Zabbix as a full high availability cluster. For MySQL we can use a **Primary/Primary** setup, for the frontend we can use load balancing techniques like **HAProxy** and for the Zabbix server, we can use the built-in high availability method. Combine all of these together and you'll definitely be able to serve your every (production ready!) need.

## Conclusion

To conclude, there are many options to start taking backups of our Zabbix environment. It all starts at the database and these backups are definitely vital to keep things safe in case of disaster. When making the backups, do not forget about the configuration files and custom scripts as well as the frontend backup option. Combining all of these solutions will safeguard our environment, but if that isn't enough – do not forget about industry standards like **snapshots**. Even further safeguarding our environment on multiple levels.

<https://oicts.com>**Opensource ICT Solutions****Tags:**

Api

Backups

Database

**About Nathan Liefing**

IT consultant &amp; Zabbix Trainer at Opensource ICT Solutions

[← Prev Post](#)[Next Post →](#)☐ Subscribe ▼

Connect with

[↗ Login](#)

Please login to comment

**0 COMMENTS****Categories**

Handy Tips

Technical

How To

Integrations

**Zabbix resources**[www.zabbix.com](http://www.zabbix.com)

Integrations &amp; Templates

Documentation

Webinars

**Professional services**

Technical Support

Training

Consulting

Case Study

