Monitoring Your FileMaker Server

Getting Started with Zabbix

By Wim Decorte, Senior Technical Solution Architect and
Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Contents

This document is one in a series of guides that walk you through installing, configuring, and using Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix.

Why Zabbix?

Needless to say, we believe that monitoring your FileMaker Server is crucial for a stable, performant and secure deployment of your FileMaker app.

The FileMaker Server admin console overhaul with version 17 removed the live statistics viewer and live event log viewer.

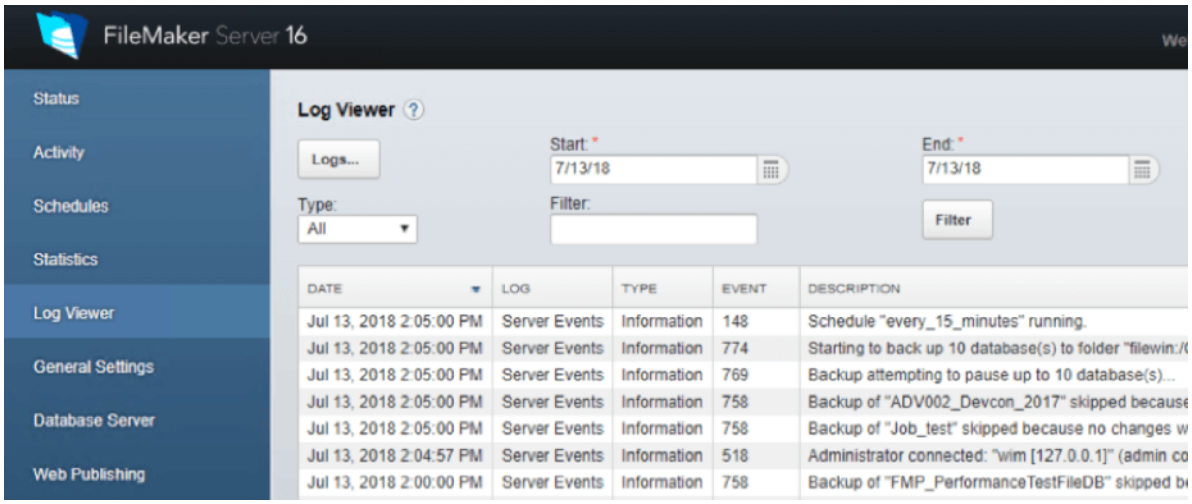


Figure 1. Admin Console - Log Viewer

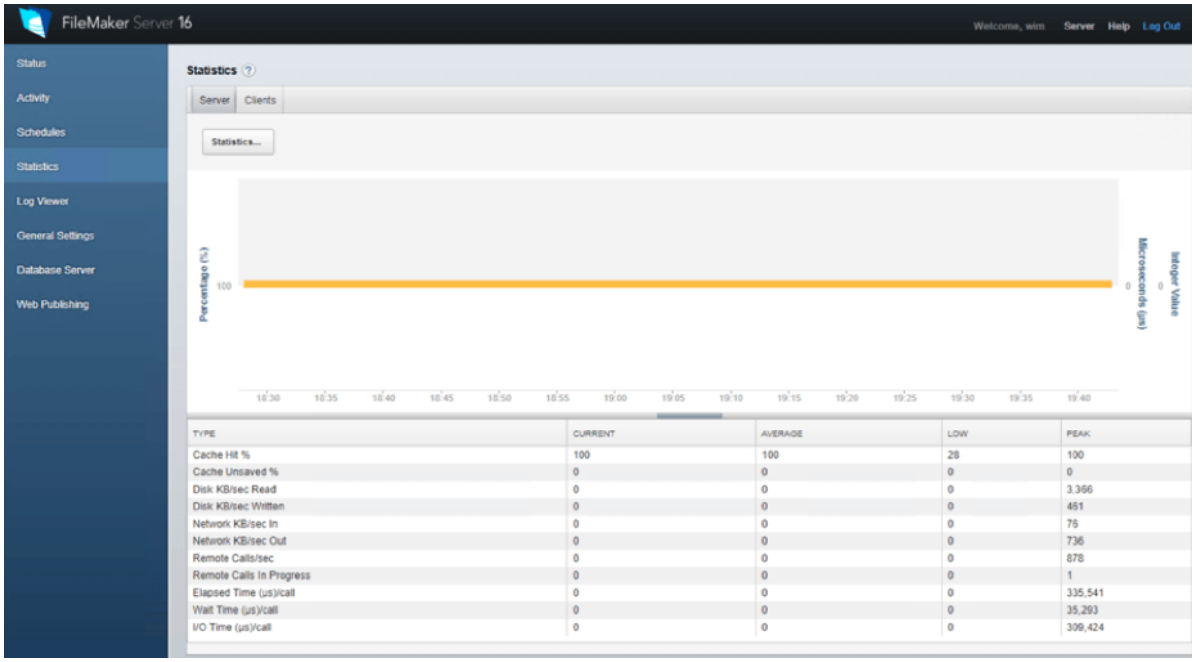


Figure 2. Admin Console - Statistics

And that, at first, seems like a giant step backwards. But it isn't really.

In my 2018 DevCon session dedicated to a review of the new Admin Console and the updated Command Line and new Admin API, I had mentioned some of the available tools to monitor your server, with Zabbix as one of them. All of the tools that I had listed pretty much deliver the same functionality. However, here at Soliant Consulting, we settled on Zabbix for a variety of reasons that you can see listed in the slide deck for our 2019 DevCon presentation.

In fact, those tools, and Zabbix chief among them, give us the ability to do much more than we could ever do with the old-style FileMaker Server Admin Console. With Zabbix you have very fine-grained control over the things you want to monitor, and just as important: on the actions you want to have taken if something is off. You can have Zabbix automatically restart a process such as the server's scripting engine, the Data API, or web publishing. You can create intricate escalation rules if problems do not get resolved quickly. And you can build your own dashboards to see just the things that you want to focus on the things that are important to you.

So how do you get started?

Let's tackle the elephant in the room: you'll need a Linux server. Zabbix Server runs only on Linux.

We may lose you here, but stick around for a bit longer. We are convinced that this should not be a stumbling block for anyone.

We fully recognize that not everyone wants to have to go through multiple steps of what seem like complex and incomprehensible command line wizardry just to get to the starting line.

Here are your three options to set up a Zabbix Server as we see them:

Appliance

If you do not feel comfortable working in Linux but still want to explore what Zabbix has to offer, we suggest downloading one of the pre-built appliances.

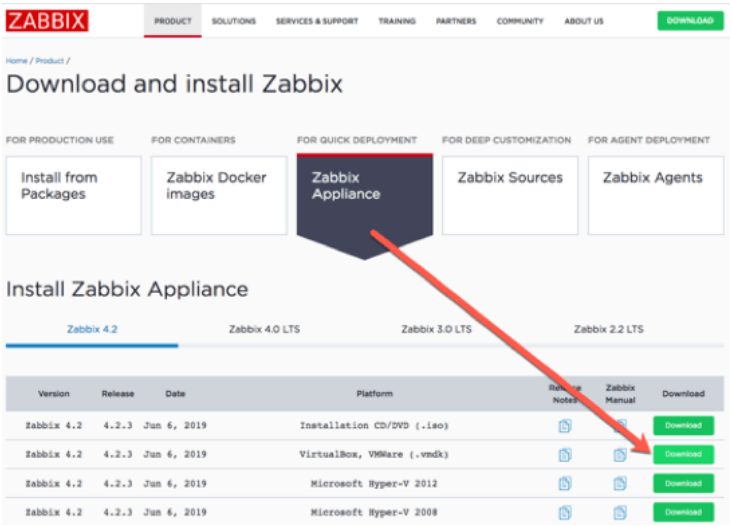


Figure 3. Download a pre-build appliance.

Such an appliance is basically a virtual machine with Linux already installed and Zabbix installed on top of that. It's very much plug-and-play and ready to go. What is left for you to do is to import the Zabbix template that we put together for our DevCon presentation and install the Zabbix agent on your FileMaker Server.

Follow along with the "2a – Zabbix as an Appliance" guide to easily create a working Zabbix installation. We do note that Zabbix mentions on the appliance download page that it is intended for testing rather than production, but we believe that is production-ready for the purpose of monitoring FileMaker Servers.

Note that if you are familiar with Docker, you can also deploy Zabbix that way.

Do-it-yourself

If you do feel comfortable with Linux, or you want to expand your skill set then there are some good resources for you to follow along with:

- FileMaker Inc. has a white paper that describes how to install Zabbix. Those instructions are for Zabbix 3.4. The current version is 4.2 but the instructions are still valid.
- We have our own guide (2b - Zabbix Full Installation) where we have captured many of our lessons-learned, and it is more detailed than the FileMaker Inc. white paper.

Commercial Product

If you want good server monitoring without the learning curve of a new tool and are willing to forgo the ability to tweak everything yourself and potentially expand the monitoring to other aspects of your infrastructure then consider using one of the commercial products like the Nutshell Console. This product that is built specifically to monitor just a FileMaker Server.

But in the Old Console You Didn't Have to Learn Anything!?

That is true. But what we had in the old Admin Console was also fairly limited. Using a tool that was designed to this task well is far superior that what we had before.

As with most things, the unfamiliarity can be a little off-putting, and that is what we try to counter with these guides. There is a learning curve to using Zabbix, as there is with any decent monitoring tool, but going through the learning curve is very worthwhile.

What Versions of FileMaker Server Apply?

Monitoring a FileMaker Server with Zabbix works with all versions of FileMaker Server. There are a few items dependent on the Admin API, but the vast majority of items we monitor on our FileMaker Servers are generic and work on all versions.

Official Documentation

Please refer to the Zabbix documentation to dive deeper for any of the topics covered in this white paper and the ones that follow it. This documentation is very good and thorough.

Monitoring Your FileMaker Server

Zabbix as an Appliance

By Wim Decorte, Senior Technical Solution Architect and
Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

This document is one in a series of guides that walk you through installing, configuring, and using Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix.

Zabbix Appliance

We're assuming that you have read part 1, Getting Started with Zabbix to Monitor Your FileMaker Server.

The Zabbix Appliance is a pre-installed Linux server plus Zabbix server. Its purpose is to remove all hurdles for those not inclined to do the initial installation themselves. It allows you to get up and running with Zabbix with almost no effort at all.

The Zabbix website mentions that it should be used for testing rather than production, but we believe it is production-ready for the purpose of monitoring FileMaker Servers.

There are, of course, some trade-offs, but they are minimal. Mainly you do not get to choose the type of back-end database that Zabbix will use (it will be MySQL version 8 as per Zabbix 4.2.3) nor the flavor of Linux for the underlying operating system (it is Ubuntu 16.0.4 as per July 2019). Chances are that you are not interested very much in this anyway.

The first step is to download the Appliance in your preferred format from the Zabbix downloads page. For this demo we will use the image that works with both VirtualBox and VMware.

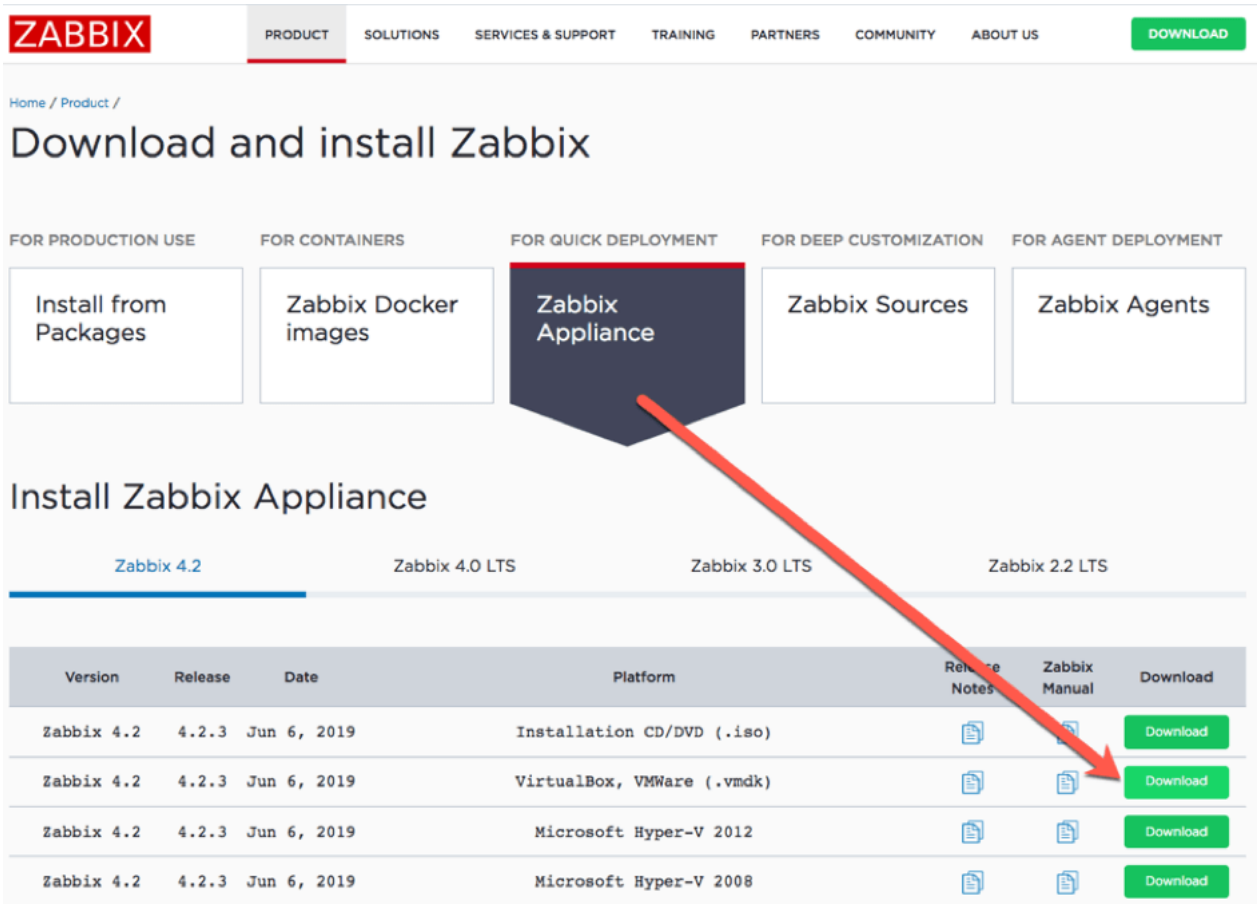


Figure 1. Download the Appliance

The download expands into a set of files, that can be opened directly in VMware Fusion (or VMware Workstation on Windows or hosted in your VMware environment).

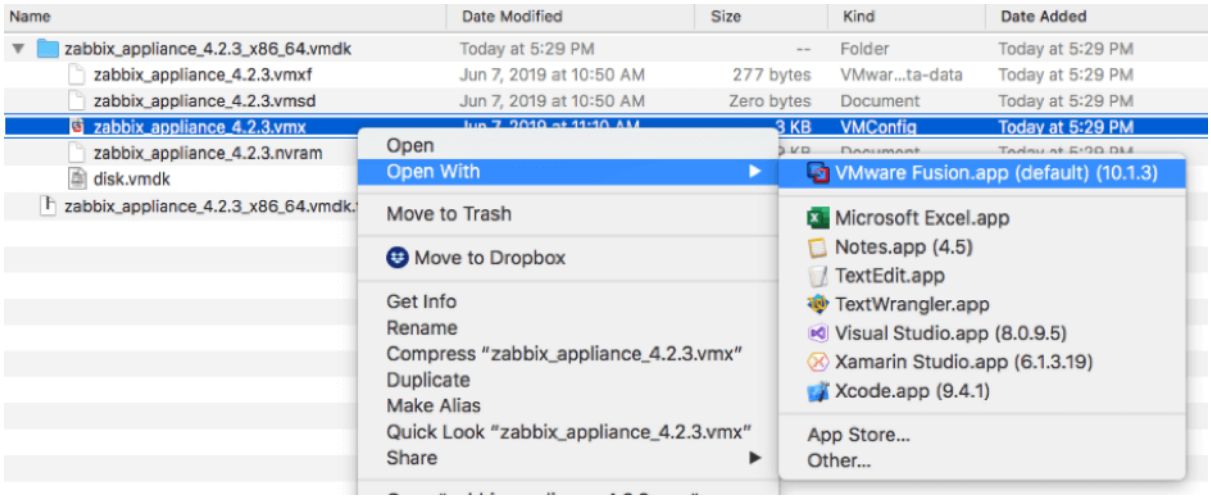


Figure 2. Open the Appliance directly in VMware Fusion

When the virtual machine starts, it shows a command prompt:

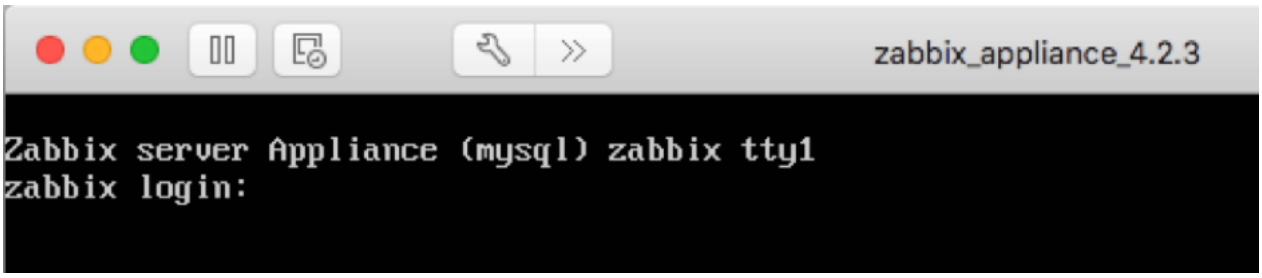


Figure 3. Command prompt

For now, we're mainly interested in logging into Zabbix itself, so we do not need to do anything in that command prompt window.

I configured the VMware settings so that the virtual machine shows up as a separate machine on my network:

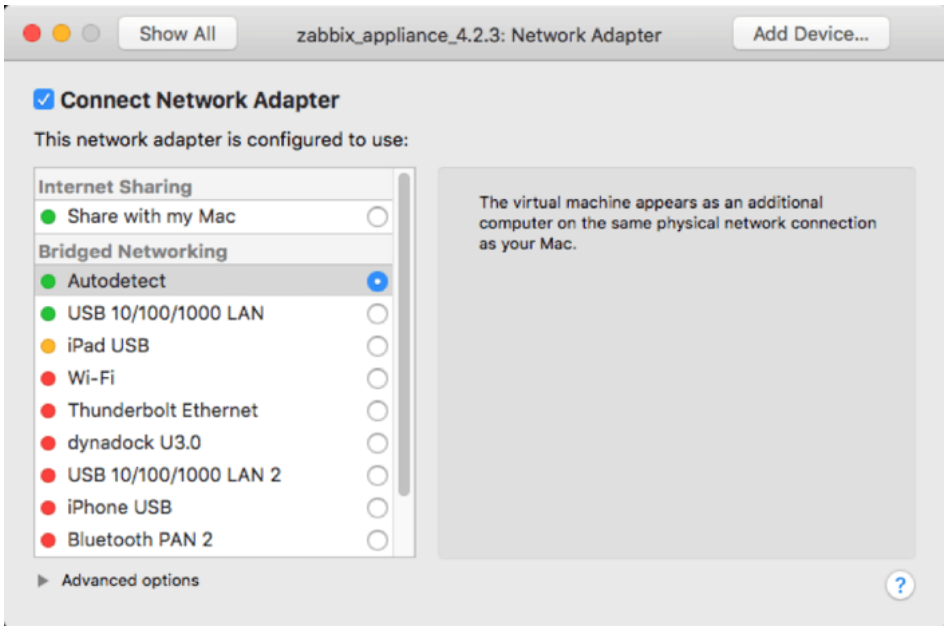


Figure 4. Configuring the VMware settings

At this point, the virtual machine picked up a dynamic IP address through DHCP. All I need to do is figure out what the IP address is (I use IP Scanner Pro for this, but your DHCP server's console would also provide you this information[1]) so that I can point my browser to it and log in:

http://<ip_address>/zabbix

The default credentials for the Zabbix frontend are username Admin (capital A!) and password zabbix:

[1] Or if you are not averse to a little bit of command line use, in VMware Fusion's window, log into Linux with the default credentials for the operating system (username: appliance, password: Zabbix) and use the "ifconfig" command to show the network card configuration. See the yellow highlighted command the IP address listed:

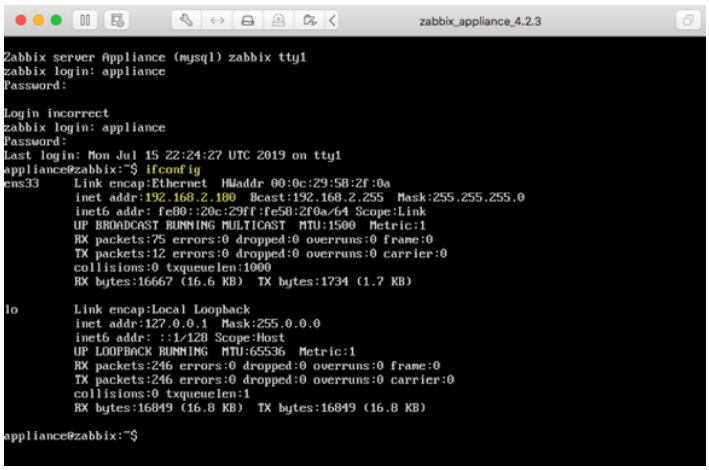


Figure 6. inet6 addr:192.168.2.180

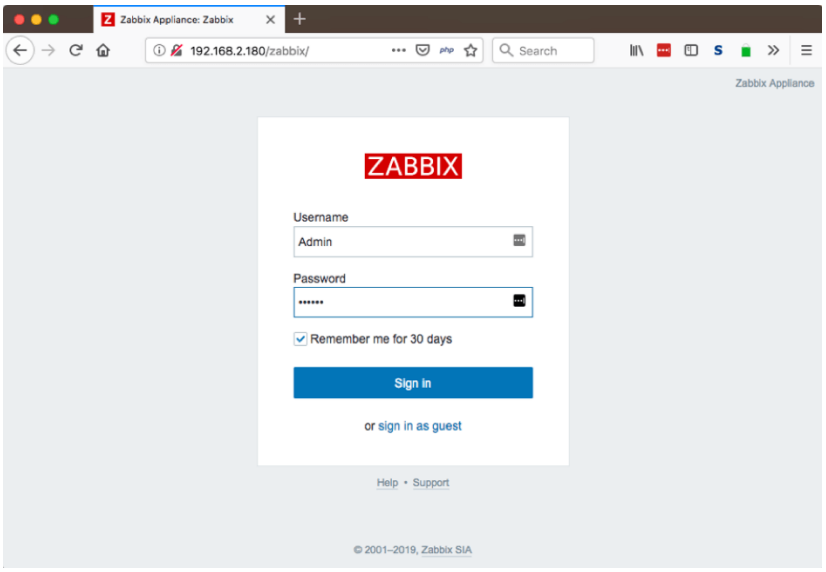


Figure 5. Use default credentials to log into Zabbix.

And with that we are in the Zabbix admin console. From this point on, it is just a matter of importing the templates that monitor various items of a FileMaker Server and adding your FileMaker Server as a host to monitor. Those steps are explained in one of the next guides in this series.

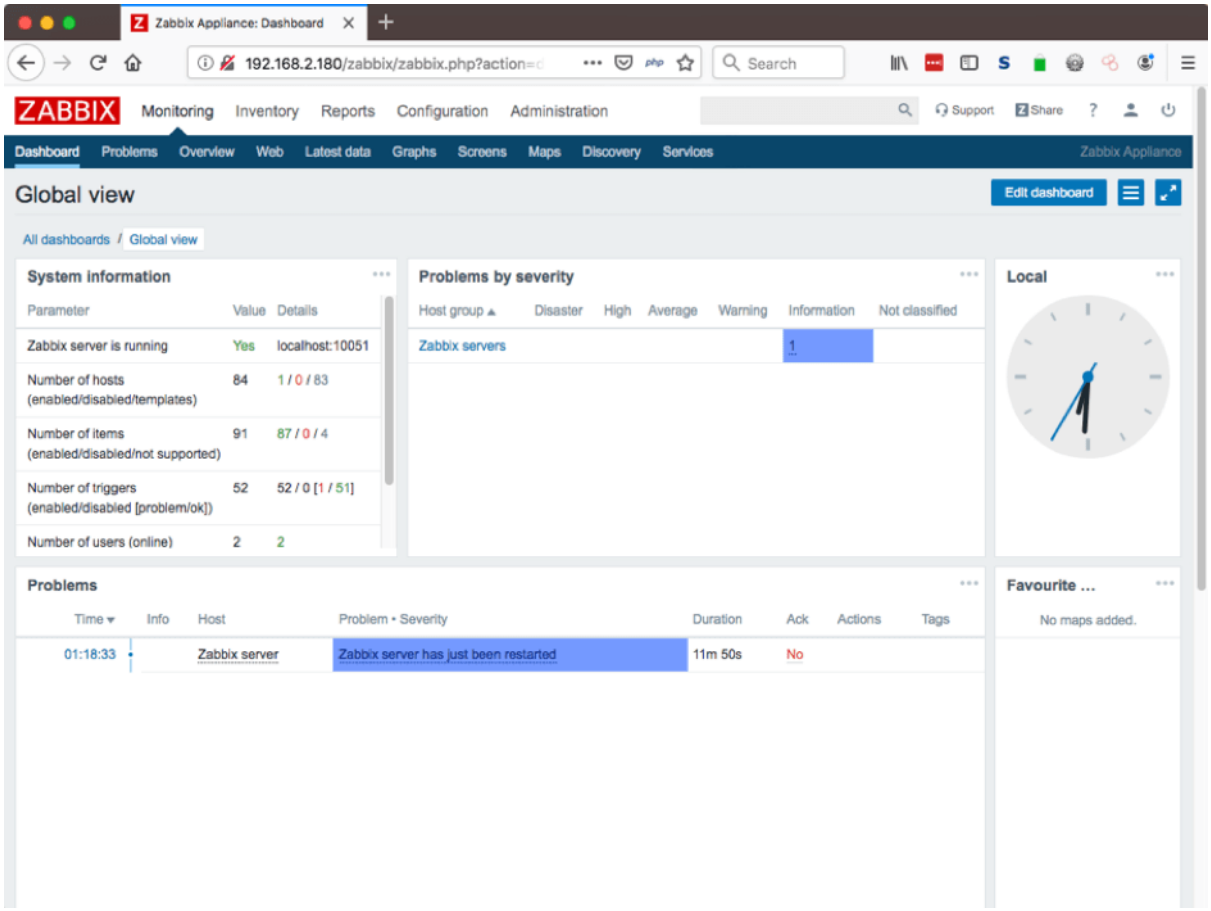


Figure 7. Zabbix admin console

The Zabbix Appliance manual (https://www.zabbix.com/documentation/4.2/manual/appliance) covers some useful details, such as which ports are open on the Linux firewall in case you need to tweak those.

There is one important OS configuration change you'll need sooner rather than later, and that is to switch your virtual machine from a dynamic to a static IP address. Your FileMaker Server will need to be able to reach it at all times so a dynamically changing IP address will not work well.

In the VMware Fusion window, log into the OS with user 'appliance' and password 'zabbix' and run the following command to install the nano text editor, one of the easiest to use:

```
sudo apt-get install nano
```

With nano in place we will update Ubuntu's network configuration and tell nano to open the configuration file with this command:

sudo nano /etc/network/interfaces

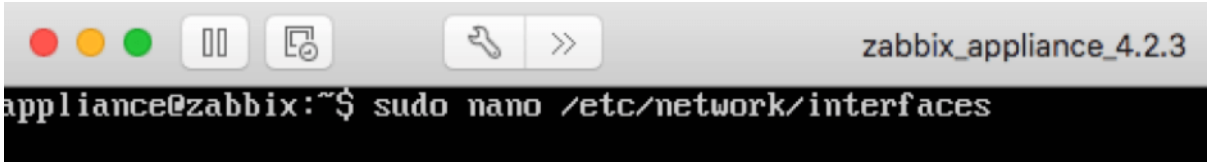


Figure 8. Command line – telling nano to open the configuration file

The default settings will look like this:

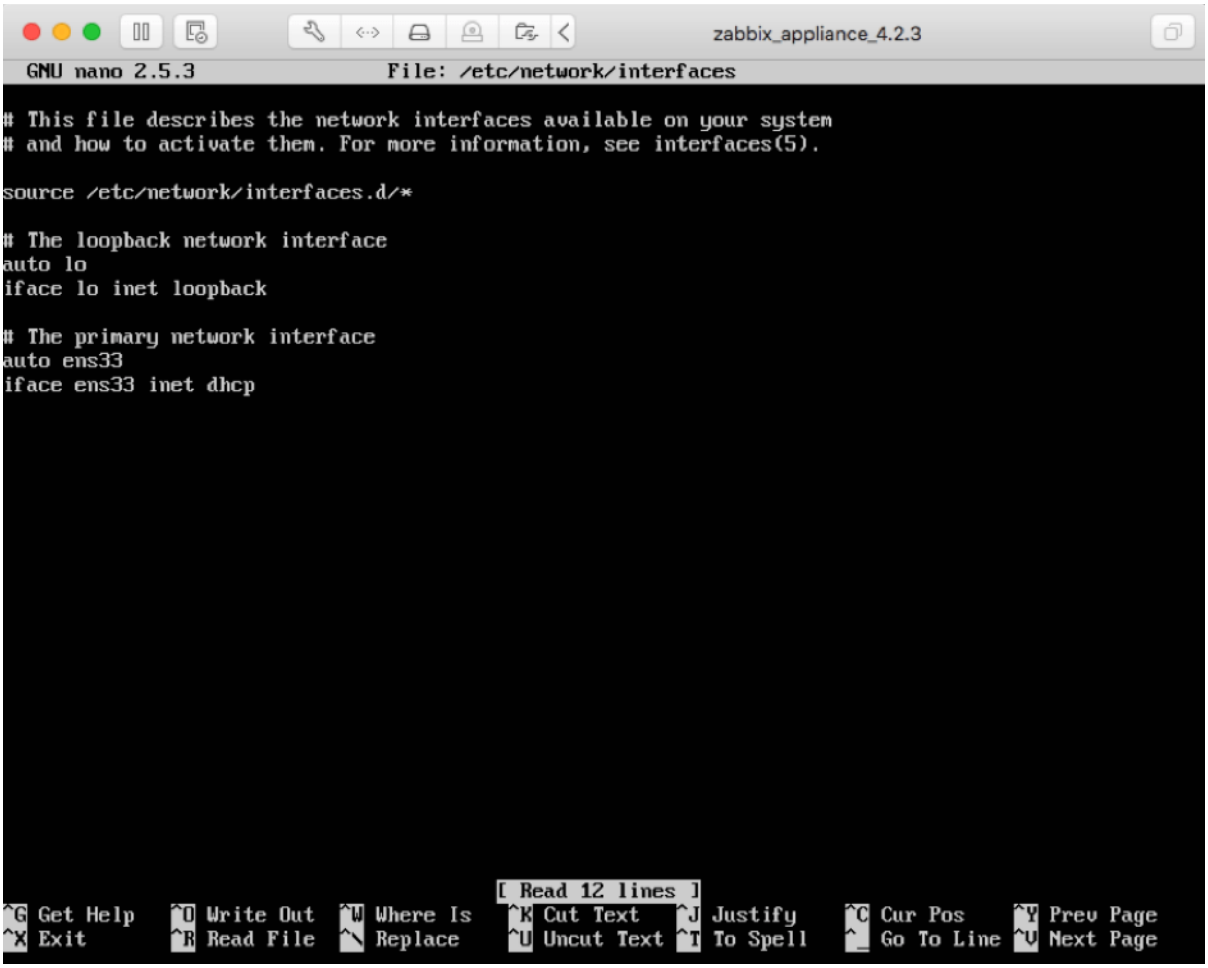


Figure 9. Default settings

We are interested in the last line as it sets the network card to use DHCP.

Change the word “dhcp” to “static”:

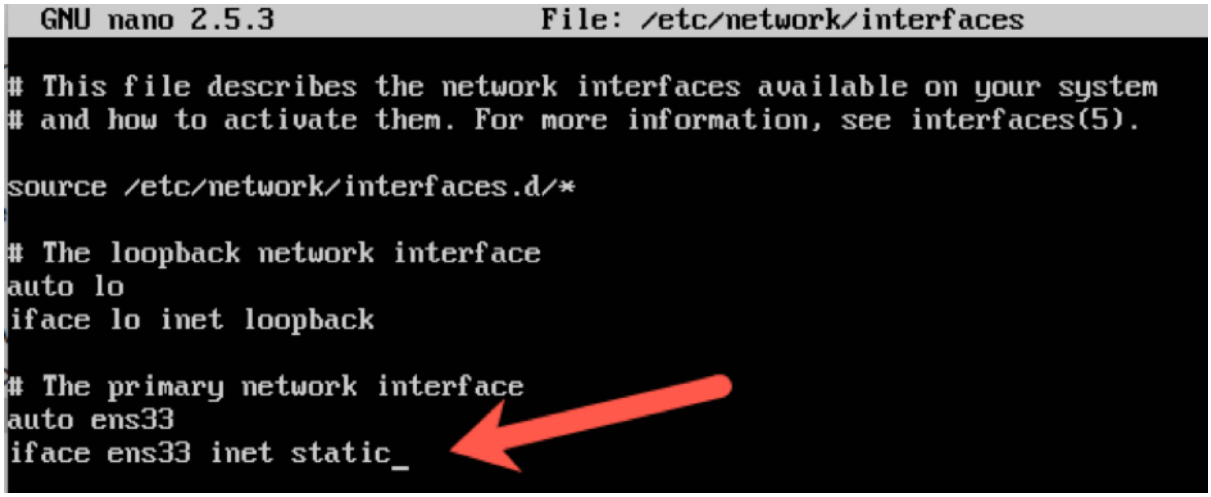


Figure 10. Change "dhcp" to "static"

And add 3 lines below it but with a static IP address, netmask and gateway that fits your network:

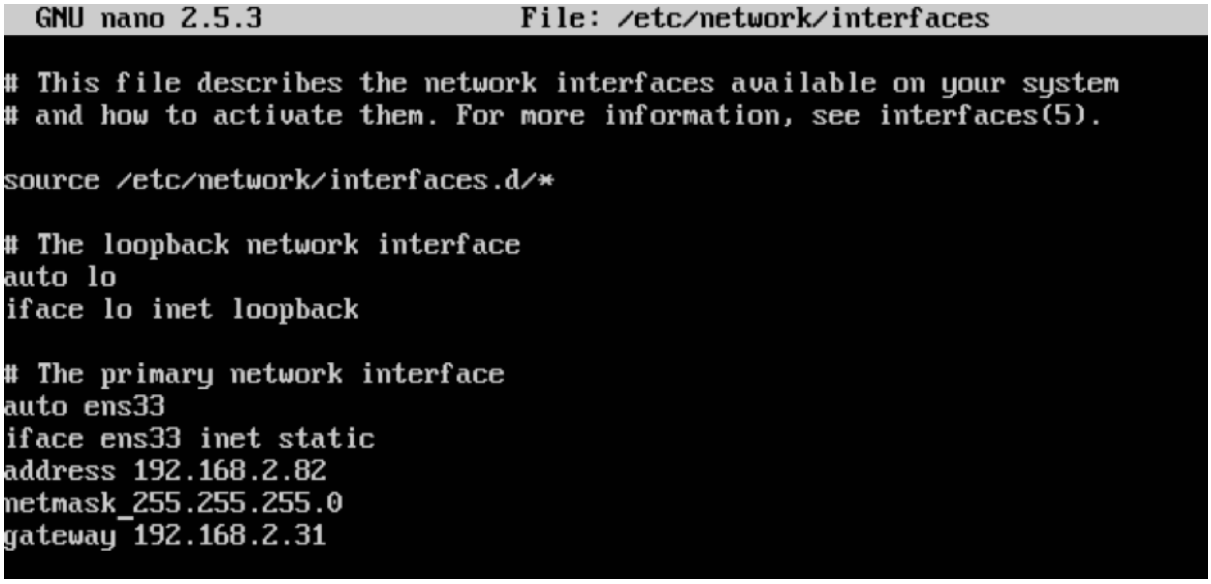


Figure 11. Add the static IP address, netmask, and gateway

Then hit control-o (oh, not zero) to be asked to save the file:

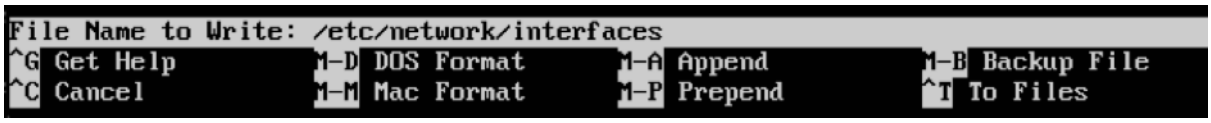


Figure 12. Save the file

Hit enter and then control-x to quit the text editor. This should drop you back to the command line prompt. As a final command, reboot your instance to have the settings take effect:

```
sudo reboot now
```

Now you can use that new static IP address in your browser to work with the Zabbix admin console:

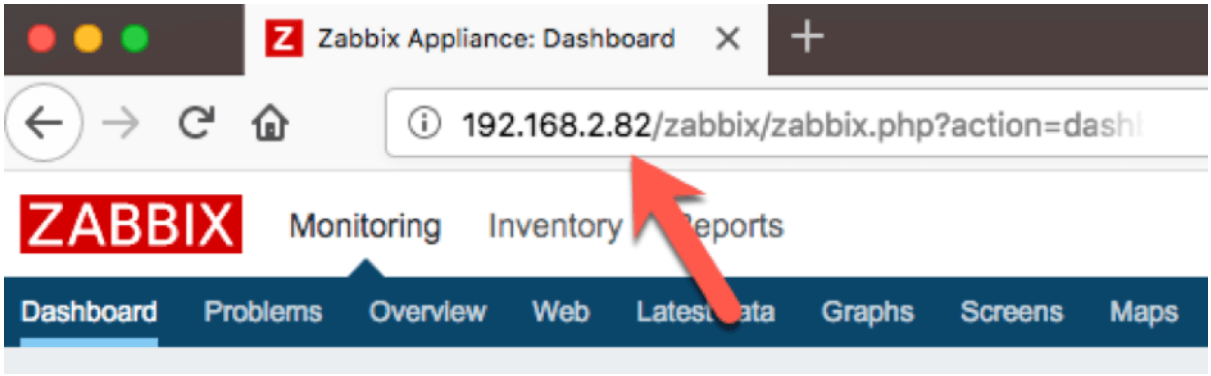


Figure 13. Use the new static IP address

The virtual machine is pre-configured to use 4 cores and 4GB of memory. Depending on the number of FileMaker Servers you want to monitor (or other servers / devices you intend to monitor), you can scale that down significantly. For my purposes I have it set to use 2 cores and 1GB of RAM, since I will only need to work with two to four FileMaker Servers. To adjust the resources for the virtual machine, stop the instance and use the VMware preferences to choose the number of processors and memory you want to assign to the machine.

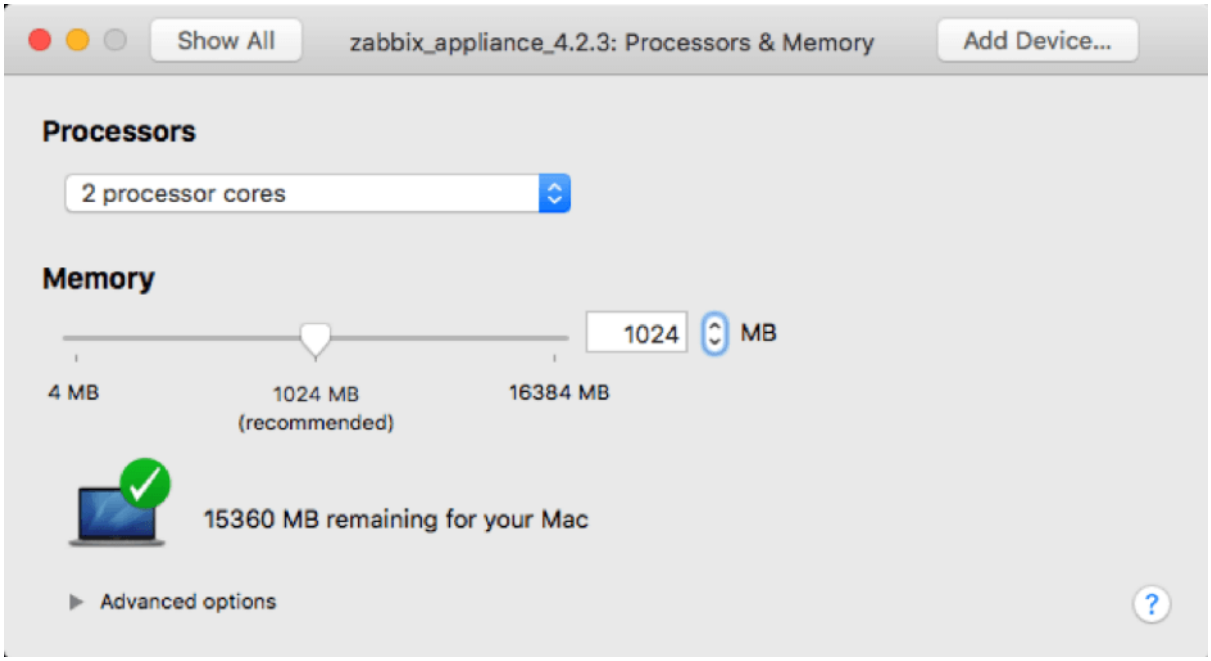


Figure 14. Set the number of cores and memory.

Monitoring Your FileMaker Server

Zabbix: Full Installation from Scratch

By Wim Decorte, Senior Technical Solution Architect
and Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Contents

This document is one in a series of guides that walk you through installing, configuring, and using Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix.

This guide will walk you through installing Zabbix Server. These steps assume you are somewhat proficient with Linux and the use of the command line or that you are willing to expand your skill set in this area. If this does not describe you, see the guide (2a – Zabbix as an Appliance) about using the Zabbix Server Appliance.

Choose Your Linux

Zabbix Server only runs on Linux, so we will have to pick a flavor of Linux we are comfortable with that is supported by Zabbix.

The Zabbix download page guides you neatly through the available choices of operating systems, versions of the chosen operating system, and backend-databases you want Zabbix Server to use to store its data:

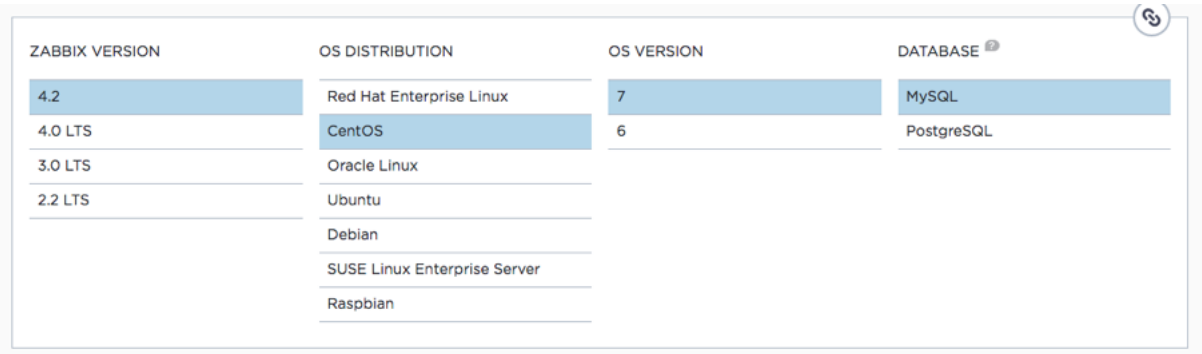


Figure 1. Zabbix download

We chose CentOS, because it is also the operating system used for FileMaker Cloud, so whatever skills we pick up working with CentOS will serve us well both in working with the underpinnings of Zabbix Server and FileMaker Cloud. CentOS is also the operating system used in FileMaker Inc's installation guide.

Note that the white paper that FileMaker Inc. published with the release of FileMaker 18 back in May of 2019 uses Zabbix Server version 3.x. The current version of as July

2019 is Zabbix 4.2 and its installation instructions are just slightly different. But the FMI guide is still a good reference[1].

Selecting the OS, version, and database on the Zabbix download page, update the command line information further down on the page that you need for the installation of Zabbix Server and its components:

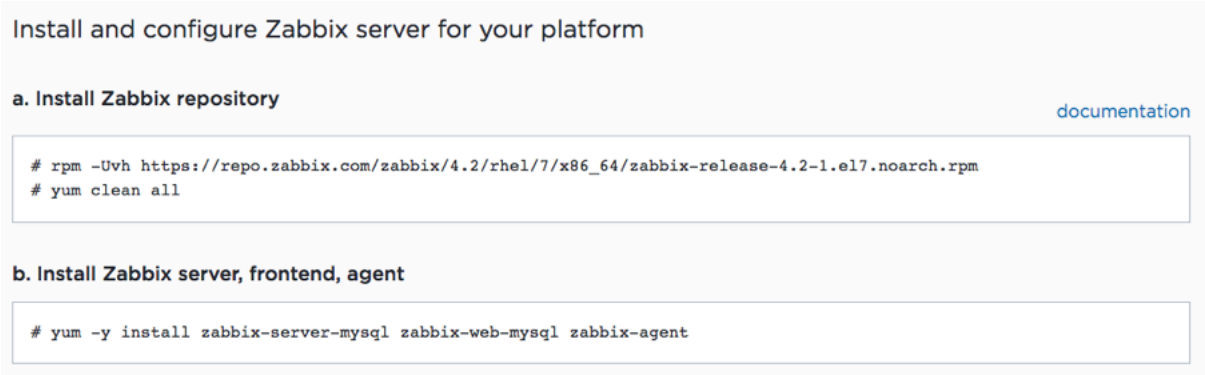


Figure 2. Updated command line information

Those steps assume that you already have a running Linux server of your chosen version, so that you can copy and paste these commands into the command line terminal. Let's take a step back and get one up and running.

There are a couple of different ways to get a running CentOS depending on whether you want it on-premise on physical hardware, on-premise as a virtual machine, or in the cloud.

Choose Your Location

For an on-premise install on physical hardware, you can grab the ISO installer directly from https://centos.org/:

[1] For complete reference, the official installation guide is here:
https://www.zabbix.com/documentation/4.2/manual/installation/install_from_packages/rhel_centos

and the one that FileMaker Inc has put together:
https://community.filemaker.com/en/s/article/Using-Zabbix-for-Monitoring-FileMaker-Server

Between those two and this guide you are reading now, you should have all the information you need to get to a successful Zabbix Server installation.

Figure 3. Download CentOS

You can also use that ISO image to install it on-premise as a virtual machine. Or, if you want to cut out a few steps, you can download a pre-built virtual machine image from https://www.osboxes.org/centos.

Figure 4. Pre-built virtual image from CentOS

Setting Up CentOS On AWS

If you prefer a cloud server, pre-built images are available as well in the cloud provider's market place. For AWS, centos.org provides an official instance for CentOS 7:

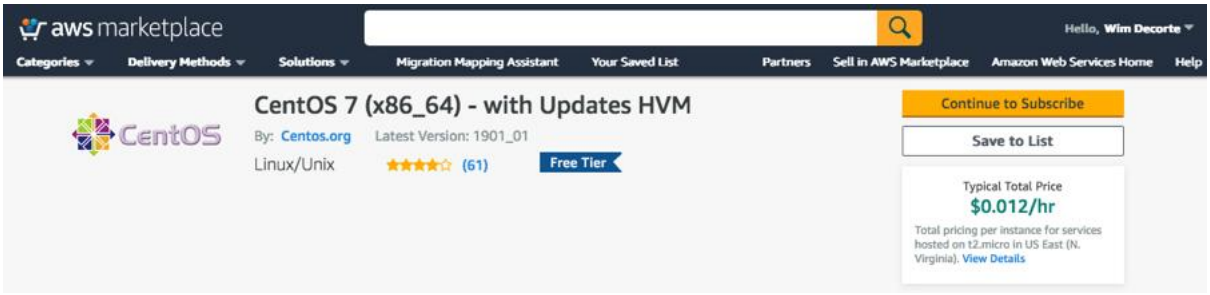


Figure 5. Pre-built image for AWS

The default instance type is a t2.micro (one virtual CPU and 1GB of RAM with 8GB of disk space), which is very cheap and more than capable of handling a Zabbix server monitoring multiple FileMaker Server hosts.

For this guide we will use AWS, since it is easy to spin up a new instance (and abandon it) without having to make changes to our in-house infrastructure.

Once you go through the few steps required in AWS to have the machine launched, you will see it in your EC2 console. The “Name” column will be empty initially. I named it centOS7:

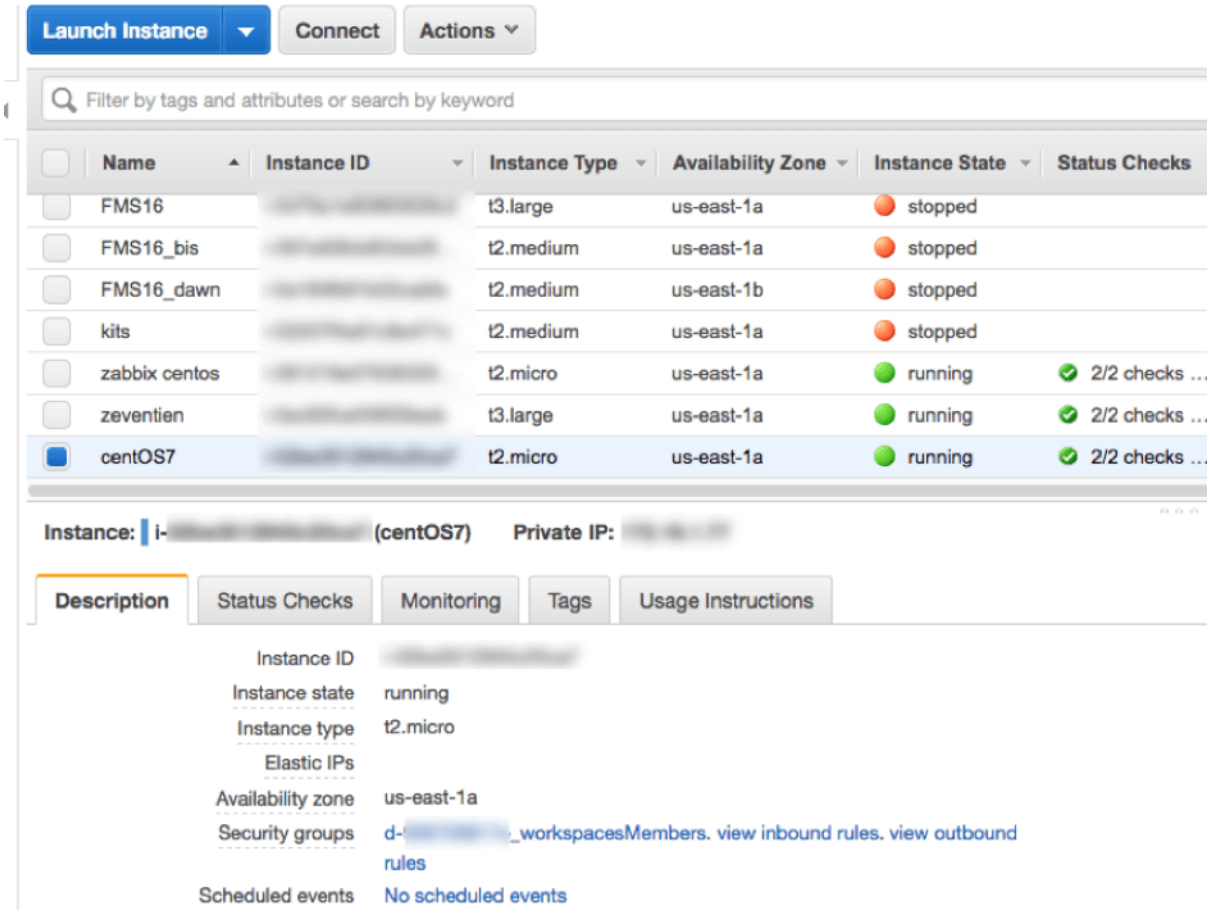


Figure 6. EC2 console

The first thing we will do is click on the security group at the bottom of that screenshot to view and update the AWS 'firewall' so that we can use SSH to remote into the instance.

Specifically, we want to work with the "inbound rules." By default, there will be none, so we will add the ones that we need:

- Port 22 for SSH
- Port 10051 for incoming data from the Zabbix agents that we will deploy later
- Ports 80 and 443 for access to the browser-based Zabbix admin console

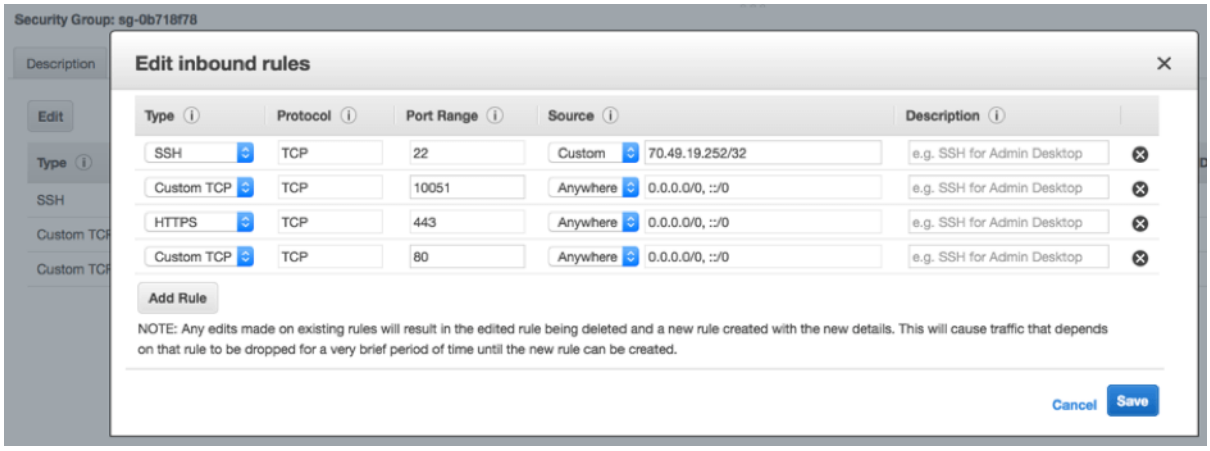


Figure 7. Set up the inbound rules

SSH To Server

At this point, we have a working Linux server. To connect to it, open your favorite SSH client (on macOS I am just using Terminal) and issue the proper SSH command:

ssh -i /Users/wimdecorte/Documents/projects/ETS18/zabbix_resources/wim_ets_15.pem centos@xxx.xxx.xxx.xxx

The path to the .pem file is required by AWS to allow SSH connections. “centos” is the default username for CentOS, and what comes after the @ is the public IP address or DNS name of your Linux server.

The first time you log in you will be asked for confirmation to connect and then will see the command prompt of an SSH session on your Linux server.

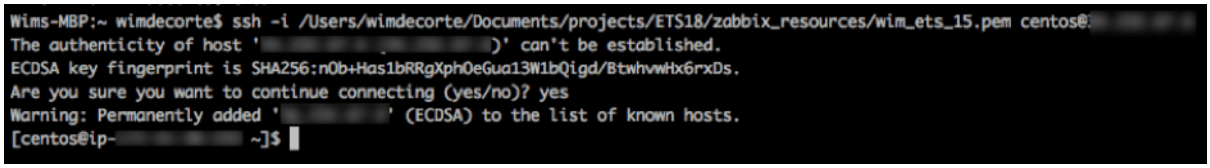


Figure 8. Command prompt of an SSH session

Install nano

The very first thing we will do is install “nano,” our favorite Linux text editor.

Type in:

sudo yum install nano

(yum stands for YellowDog Updater, Modified and is the CentOS default software manager. It is used to install new software and system updates).

As with all installations and updates, you will see a bit of an overview of what will happen, and you will be asked to confirm with “Y” that you want to proceed:

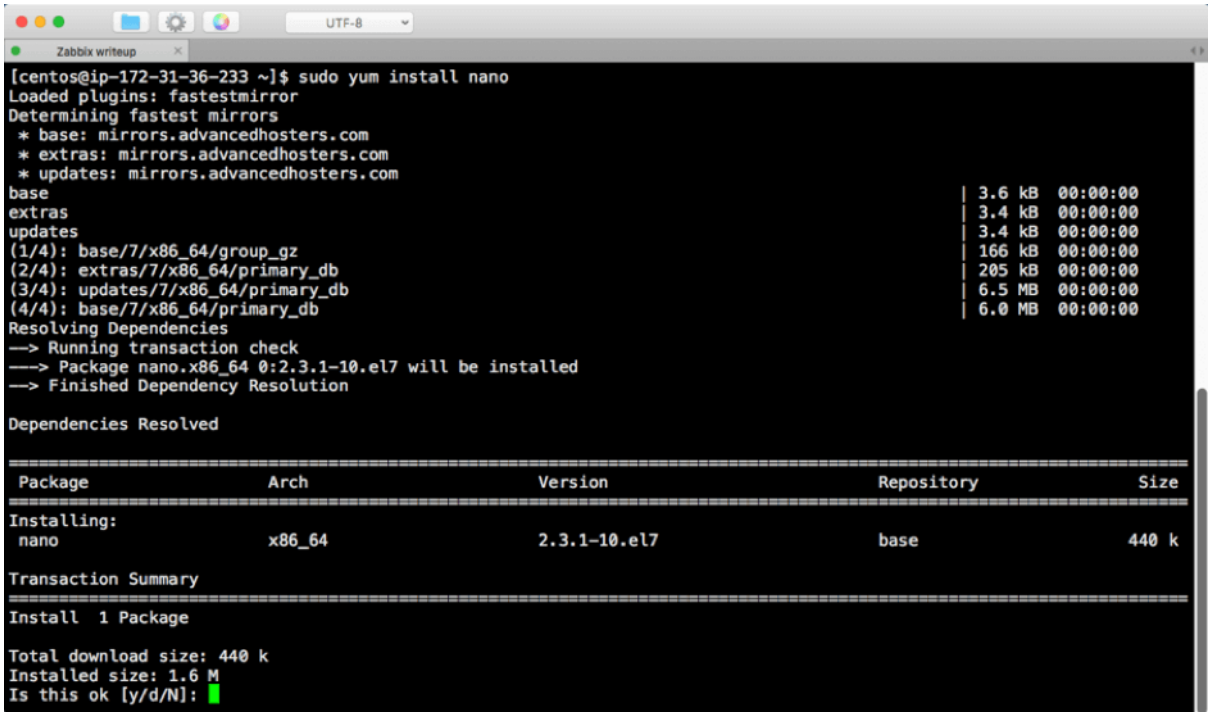


Figure 9. Type “Y” to proceed

A few seconds later, we will be finished:

Figure 10. Nano install is complete

SELinux Configuration Change

Second on our to-do list is to change a security setting in CentOS’s default configuration. By default, CentOS has SELinux enabled, which will get in the way of Zabbix Server functioning properly, so we need to adjust it:

Type in:

```
sudo nano /etc/selinux/config
```

Change the line that starts with “SELINUX” to read “SELINUX=permissive”

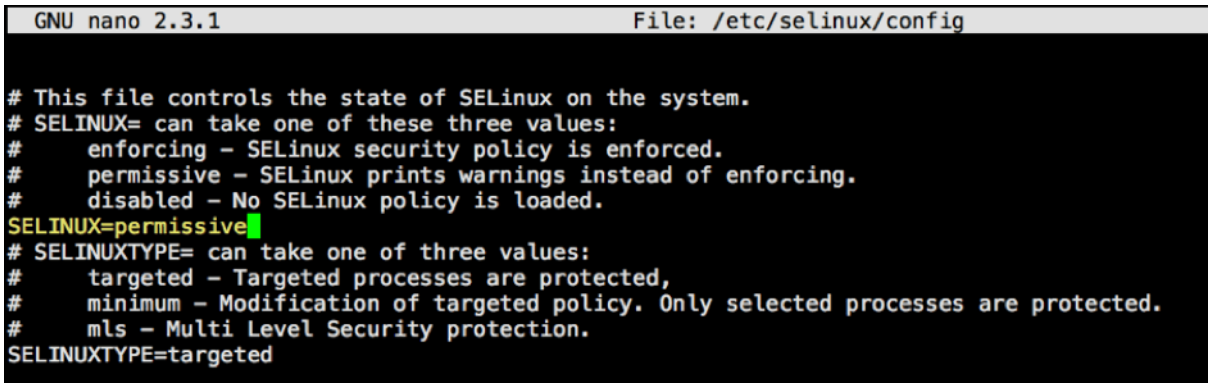


Figure 11. Line changed to “SELINUX=permissive”

Press control-o and then enter to save the changes and control-x to quit the text editor.

Install Zabbix

Now we can go back to the instructions on the [Zabbix download page] that tell us how to install Zabbix Server:

Figure 12. Install Zabbix Server

The first set of commands is basically telling ‘yum’ where the installers are located and to clean its internal database of available software locations.

Remember to run all of these commands as ‘super user’ by prefixing them with ‘sudo’. The first command will look like this:

Figure 13. Prefix all commands with “sudo”

The result of running the first two commands will look like this:

```
[centos@ip-          /]$ sudo rpm -Uvh https://repo.zabbix.com/zabbix/4.2/rhel/7/x86_64/zabbix-release-4.2-1.el7.n
oarch.rpm
Retrieving https://repo.zabbix.com/zabbix/4.2/rhel/7/x86_64/zabbix-release-4.2-1.el7.noarch.rpm
warning: /var/tmp/rpm-tmp.TFAp95: Header V4 RSA/SHA512 Signature, key ID a14fe591: NOKEY
Preparing...                          ################################# [100%]
Updating / installing...
   1:zabbix-release-4.2-1.el7         ################################# [100%]
[centos@ip-          /]$ sudo yum clean all
Loaded plugins: fastestmirror
Cleaning repos: base extras updates zabbix zabbix-non-supported
Cleaning up list of fastest mirrors
[centos@ip-          /]$
```

Figure 14. After running commands under "a. Install Zabbix repository"

The third command on the Zabbix downloads page (under b) is where the actual installation happens:

sudo yum -y install zabbix-server-mysql zabbix-web-mysql zabbix-agent

That one will run for a little while, install everything needed, and report back what it has done:

```
Installed:
  zabbix-agent.x86_64 0:4.2.4-1.el7  zabbix-server-mysql.x86_64 0:4.2.4-1.el7  zabbix-web-mysql.noarch 0:4.2.4-1.el7

Dependency Installed:
  OpenIPMI-libs.x86_64 0:2.0.23-2.el7                 OpenIPMI-modalias.x86_64 0:2.0.23-2.el7
  apr.x86_64 0:1.4.8-3.el7_4.1                        apr-util.x86_64 0:1.5.2-6.el7
  centos-logos.noarch 0:70.0.6-3.el7.centos           dejavu-fonts-common.noarch 0:2.33-6.el7
  dejavu-sans-fonts.noarch 0:2.33-6.el7               fontpackages-filesystem.noarch 0:1.44-8.el7
  fping.x86_64 0:3.10-1.el7                           httpd.x86_64 0:2.4.6-89.el7.centos
  httpd-tools.x86_64 0:2.4.6-89.el7.centos            libX11.x86_64 0:1.6.5-2.el7
  libX11-common.noarch 0:1.6.5-2.el7                  libXau.x86_64 0:1.0.8-2.1.el7
  libXpm.x86_64 0:3.5.12-1.el7                        libjpeg-turbo.x86_64 0:1.2.90-6.el7
  libtool-ltdl.x86_64 0:2.4.2-22.el7_3                libxcb.x86_64 0:1.13-1.el7
  libxslt.x86_64 0:1.1.28-5.el7                       libzip.x86_64 0:0.10.1-8.el7
  mailcap.noarch 0:2.1.41-2.el7                       net-snmp-libs.x86_64 1:5.7.2-37.el7
  php.x86_64 0:5.4.16-46.el7                          php-bcmath.x86_64 0:5.4.16-46.el7
  php-cli.x86_64 0:5.4.16-46.el7                      php-common.x86_64 0:5.4.16-46.el7
  php-gd.x86_64 0:5.4.16-46.el7                       php-ldap.x86_64 0:5.4.16-46.el7
  php-mbstring.x86_64 0:5.4.16-46.el7                 php-mysql.x86_64 0:5.4.16-46.el7
  php-pdo.x86_64 0:5.4.16-46.el7                      php-xml.x86_64 0:5.4.16-46.el7
  t1lib.x86_64 0:5.1.2-14.el7                         unixODBC.x86_64 0:2.3.1-11.el7
  zabbix-web.noarch 0:4.2.4-1.el7

Complete!
[centos@ip-          /]$
```

Figure 15. Showing what has been done

Install MySQL

The next step is to configure the underlying MySQL (c and d on the Zabbix downloads page). But this is where the instructions may fail; MySQL is likely not installed at this point yet. You can try the command, but if an error comes back, follow the steps below to install MySQL:

Add the MySQL 8.0 repository to your server:

sudo yum install https://dev.mysql.com/get/mysql80-community-release-el7-2.noarch.rpm

Then install MySQL itself:

sudo yum -y install mysql-community-server

This one is a fairly hefty download and will take a while. After it is installed, start the MySQL service:

sudo systemctl start mysqld

Enable it to auto-start when the machine starts:

sudo systemctl enable mysqld

At any time, you can check if MySQL is running by using this command:

sudo systemctl status mysqld

When all is well, you will see an output from that 'status' command, confirming MySQL is up and running

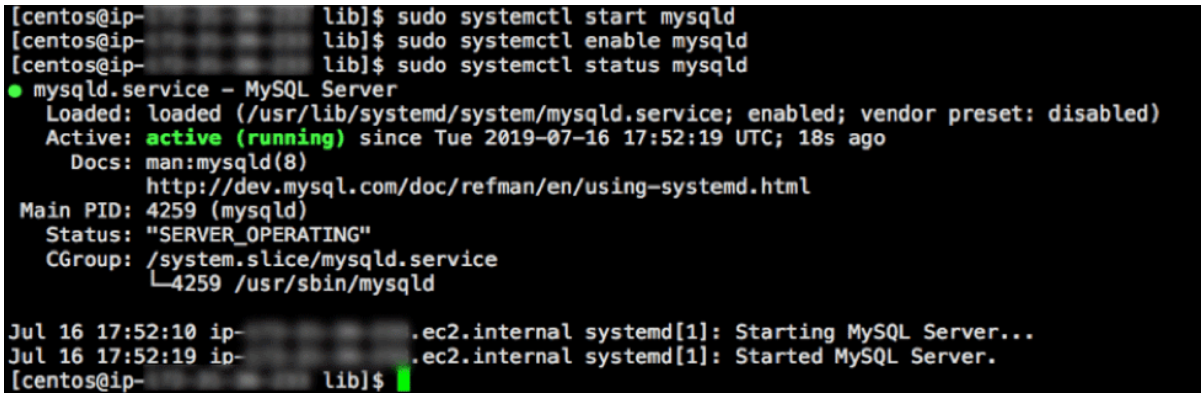


Figure 16. MySQL up and running

MySQL installs with a temporary password that we need to grab before we do anything else. To do this, type in:

sudo nano /var/log/mysqld.log

and make note that the master user is 'root' with the password listed there.

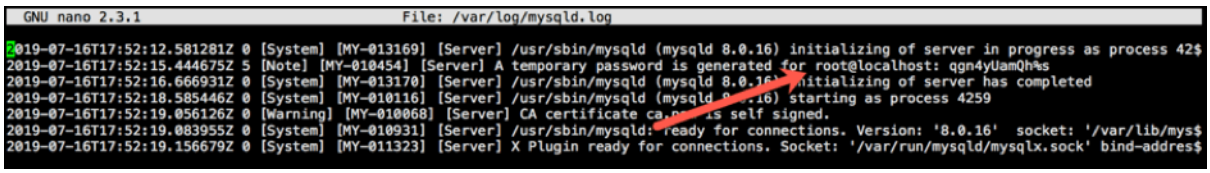


Figure 17. Make note of the temporary password generated for master user

Hit control-x to quit the text editor.

Create the Zabbix Database

Now we are ready to resume our Zabbix installation following the step c instructions on the Zabbix downloads page to create the required database:

The first command is to enter ‘mysql mode’:

```
sudo mysql -uroot -p
```

At the password prompt, use the password that you retrieved earlier. After doing so, you will find yourself at a MySQL prompt.

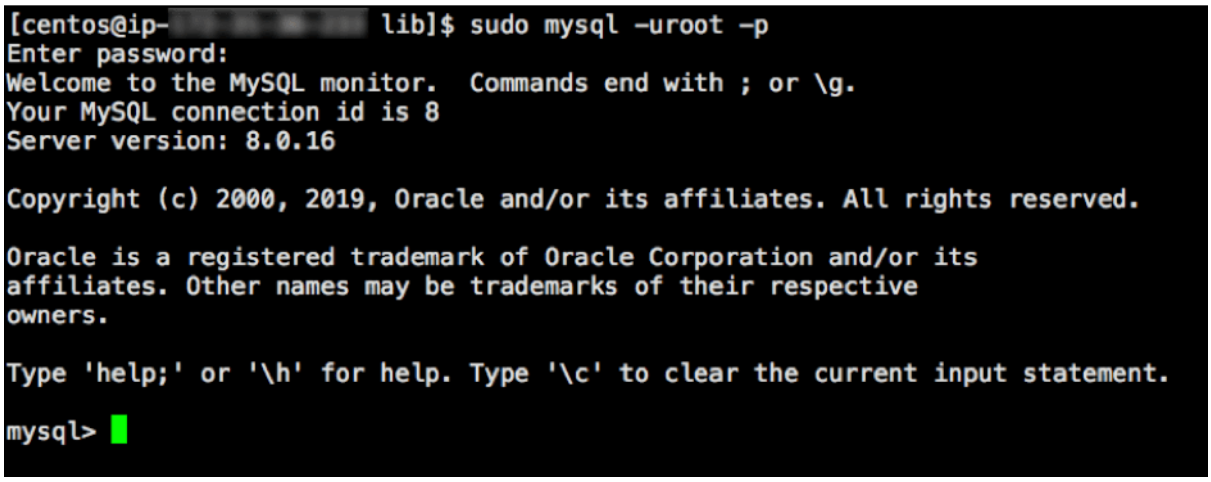


Figure 18. Use the temporary password

Before we execute the commands listed on the Zabbix page, we need to change the password for the root user, since that temporary password cannot be used beyond this point. (The commands listed in step c will fail if you try.)

Type in:

```
alter user root@localhost identified by 'someNewPassword';
```

The new password is in-between the single quotes. Don't forget to add the ";" at the end. This is how MySQL knows you are done with the command.

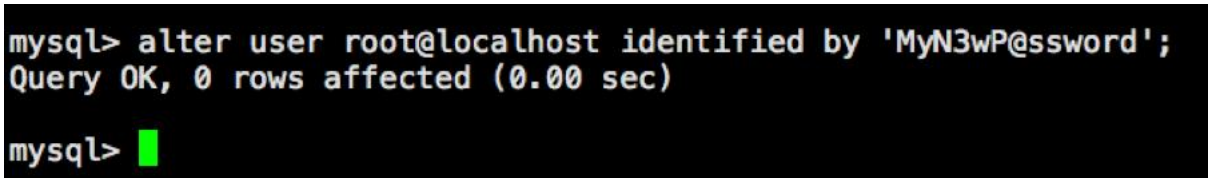


Figure 19. Temporary password is replaced

Now type in (or copy/paste) the first of the three separate MySQL commands of step c to create the database named "zabbix.."

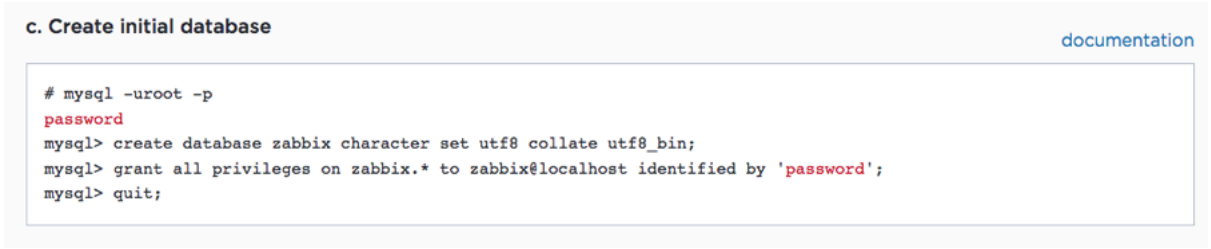


Figure 20. Create the database named "zabbix"

Before we do the next command, we actually have to create the Zabbix user in MySQL. Older versions of MySQL allow you to use the GRANT command to both create the user and assign rights to that user, but that is no longer supported in newer version of MySQL (versions 8+).

Type in the following to create a MySQL "zabbix" user with a password you specify:

CREATE USER zabbix@localhost IDENTIFIED WITH mysql_native_password BY 'MyOtherN3wP@ssword';

Then instead of the command shown in the Zabbix instructions, use this slightly modified one to grant rights to the "zabbix" database for the "zabbix" user:

GRANT ALL PRIVILEGES ON zabbix.* TO zabbix@localhost WITH GRANT OPTION;

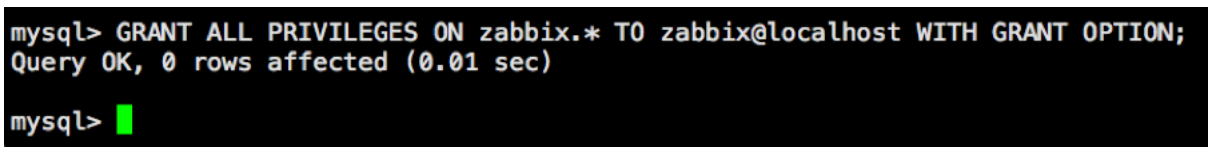


Figure 21. Grant rights to the "zabbix" database for the "zabbix" user

The last command is easy and just drops out of the MySQL mode and back into Linux:

Figure 22. Quit MySQL and return back into Linux

The last item in step c is to import the schema for the "zabbix" database:

sudo zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p Zabbix

Figure 23. Import schema for the "zabbix" database

Note that the password requested here is for the newly created "zabbix" user.

Configure Zabbix Server

The next step in the Zabbix instructions calls for an edit to the Zabbix config file to make sure that Zabbix knows the MySQL password for the "zabbix" user:

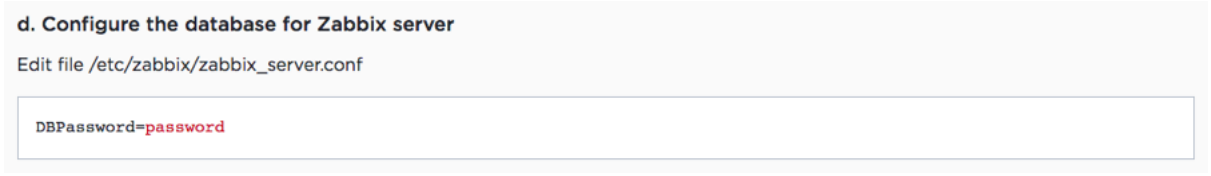


Figure 24. Edit the Zabbix config file

Type in:

sudo nano /etc/zabbix/zabbix_server.conf

to open the Zabbix configuration file and scroll down to the entry for the database password:

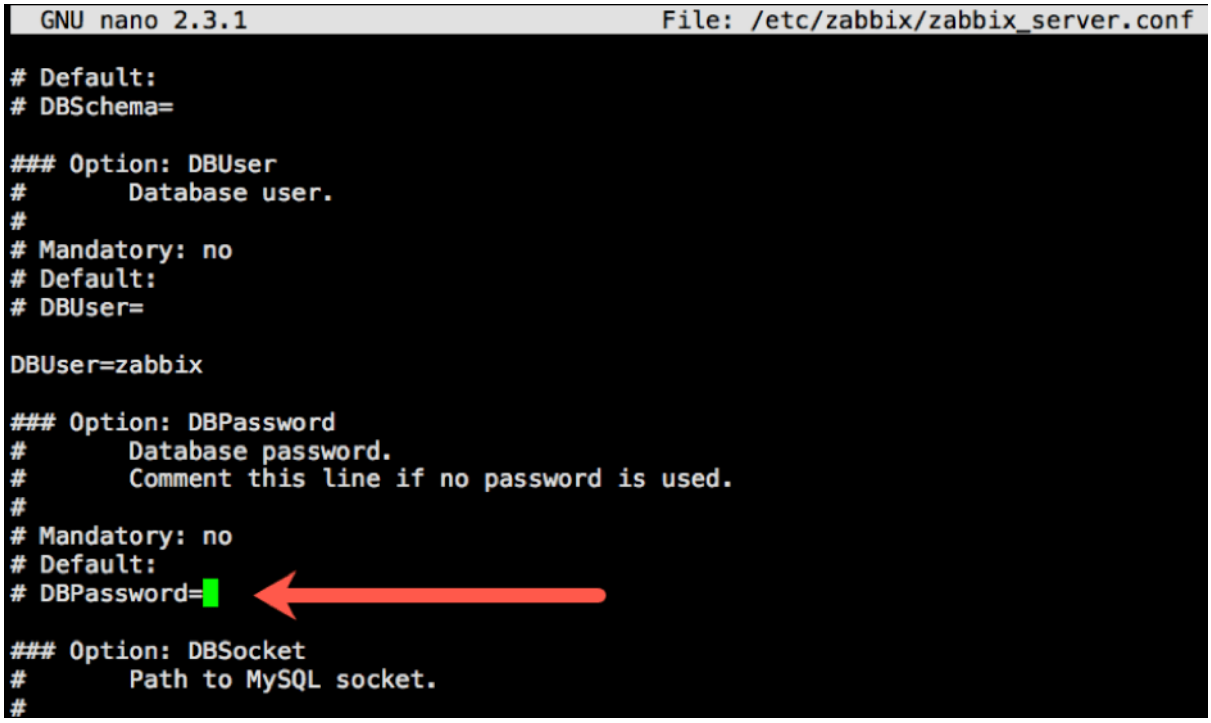


Figure 25. Scroll to entry for the database password

Remove the "#" at the start of the line and add the password for the Zabbix MySQL user.

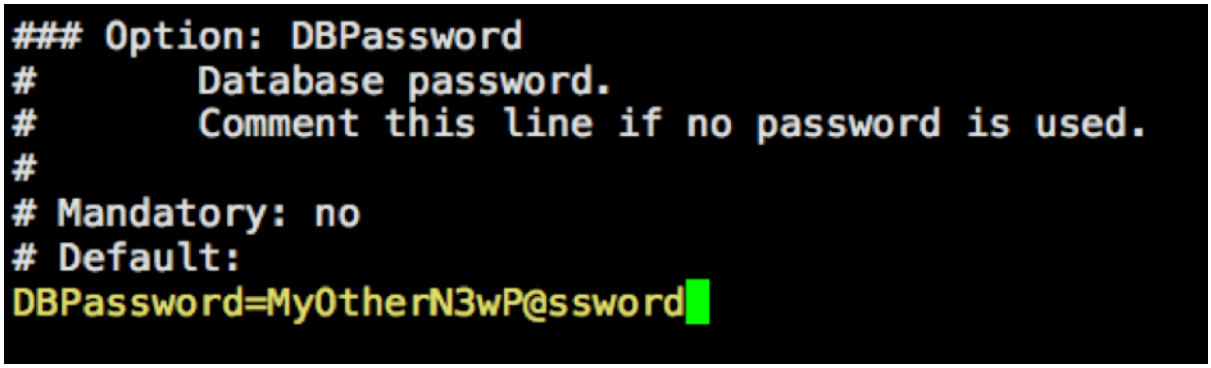


Figure 26. # symbol removed from start of the line

Hit control-o and then enter to save the change and then control-x to quit nano.

At this point, much of the heavy lifting is done; the next step listed on the Zabbix download page is to update the time zone that will be used by Zabbix:

e. Configure PHP for Zabbix frontend

Edit file /etc/httpd/conf.d/zabbix.conf, uncomment and set the right timezone for you.

```
# php_value date.timezone Europe/Riga
```

Figure 27. Update the time zone

Type in

sudo nano /etc/httpd/conf.d/zabbix.conf

and scroll down to the time zone setting:

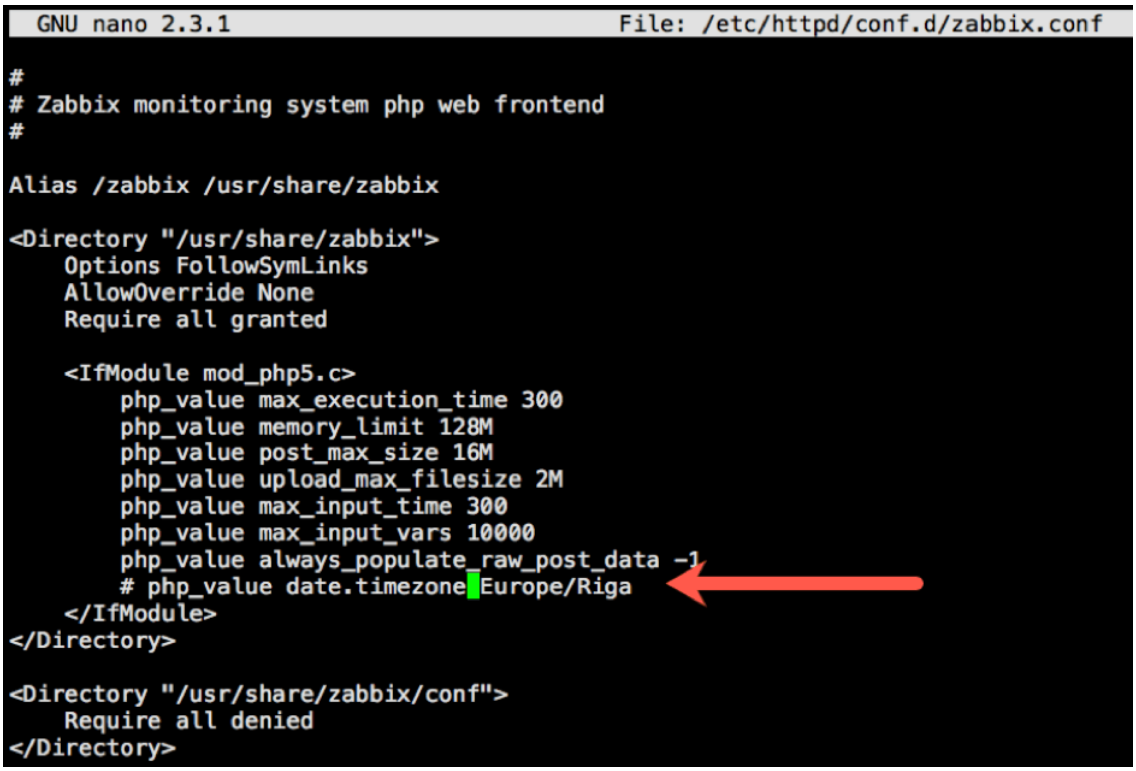


Figure 28. Time zone setting

And change it to your time zone. All supported time zones are listed here: https://www.php.net/manual/en/timezones.php Since we are on the East Coast, we changed it to America/New_York and removed the "#" at the start of the line:

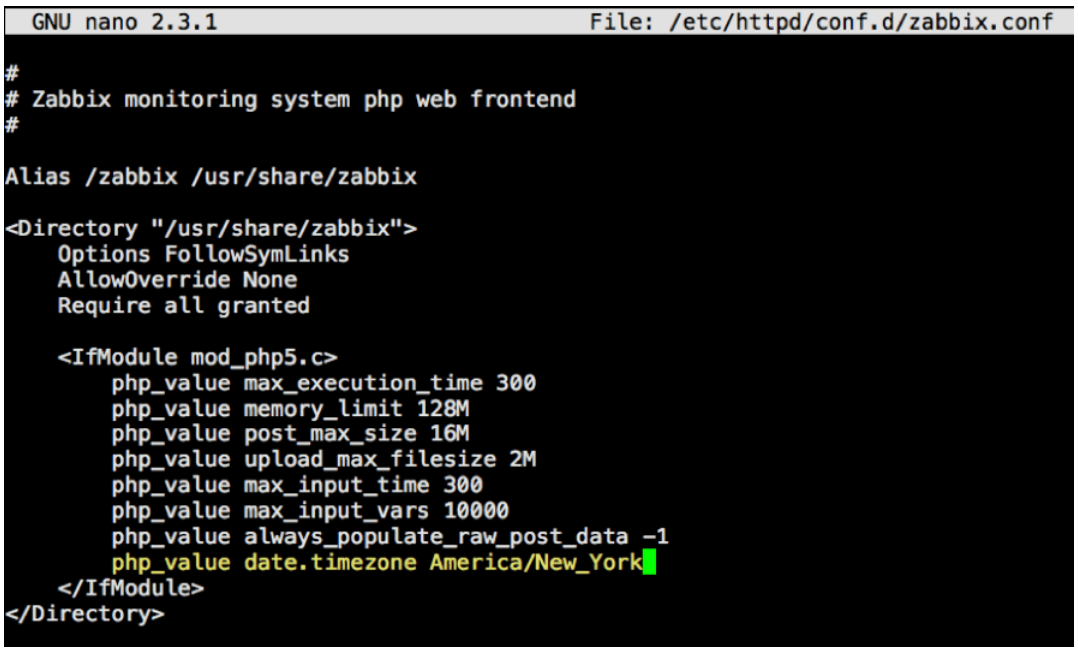


Figure 29. # symbol removed from start of the time zone line

Hit control-o and then enter to save the change and control-x to exit the text editor.

At this point we can start the Zabbix server so that all the changes we have made take effect.

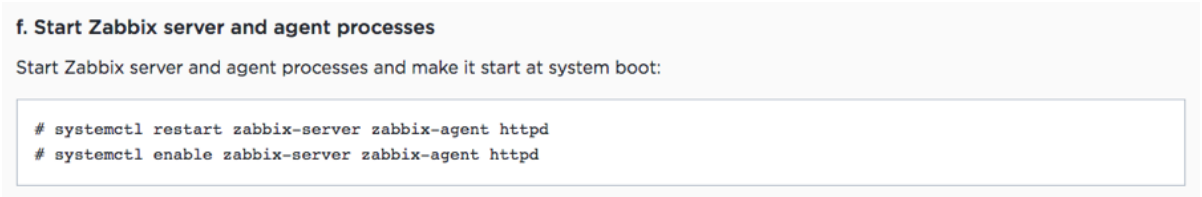


Figure 30. Start the Zabbix server

sudo systemctl restart zabbix-server zabbix-agent httpd

This command is actually restarting three services:

1. The Zabbix server
2. The Zabbix agent (each Zabbix server also monitors itself)
3. The web server (https)

The 2nd command under step F ensures that all three of these services will auto-start when the machine restarts:

sudo systemctl enable zabbix-server zabbix-agent httpd

At this stage you will want to confirm that all three of these services are properly working by running these three checks:

sudo systemctl status zabbix-server
sudo systemctl status zabbix-agent
sudo systemctl status httpd

After each command you should see the service “in the green” next to ‘Active’:

```
[centos@ip-          ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2019-07-16 20:37:49 UTC; 6min ago
     Docs: man:httpd(8)
           man:apachectl(8)
 Main PID: 3795 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─3795 /usr/sbin/httpd -DFOREGROUND
           ├─3857 /usr/sbin/httpd -DFOREGROUND
           ├─3858 /usr/sbin/httpd -DFOREGROUND
           ├─3859 /usr/sbin/httpd -DFOREGROUND
           ├─3861 /usr/sbin/httpd -DFOREGROUND
           └─3862 /usr/sbin/httpd -DFOREGROUND

Jul 16 20:37:49 ip-172-31-36-233.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Jul 16 20:37:49 ip-172-31-36-233.ec2.internal systemd[1]: Started The Apache HTTP Server.
[centos@ip-          ~]$
```

Figure 31. Service shown in green

Configure Zabbix Frontend

Going back to following the instructions shown on the Zabbix download page, we can now turn our attention to the Zabbix frontend by opening it in the browser:

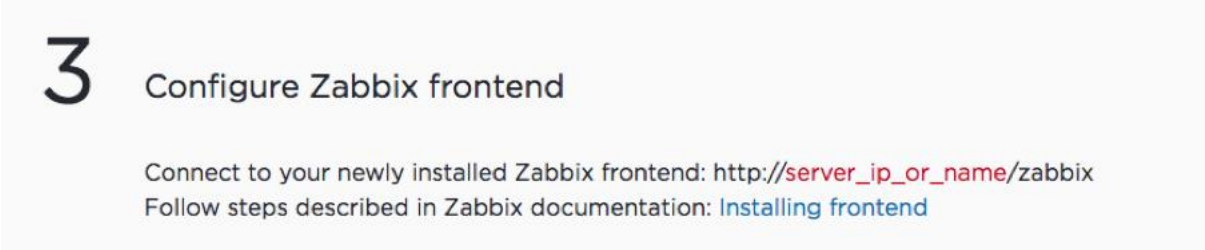


Figure 32. Open Zabbix frontend

Use the URL as indicated in your favorite browser, and you should see this:

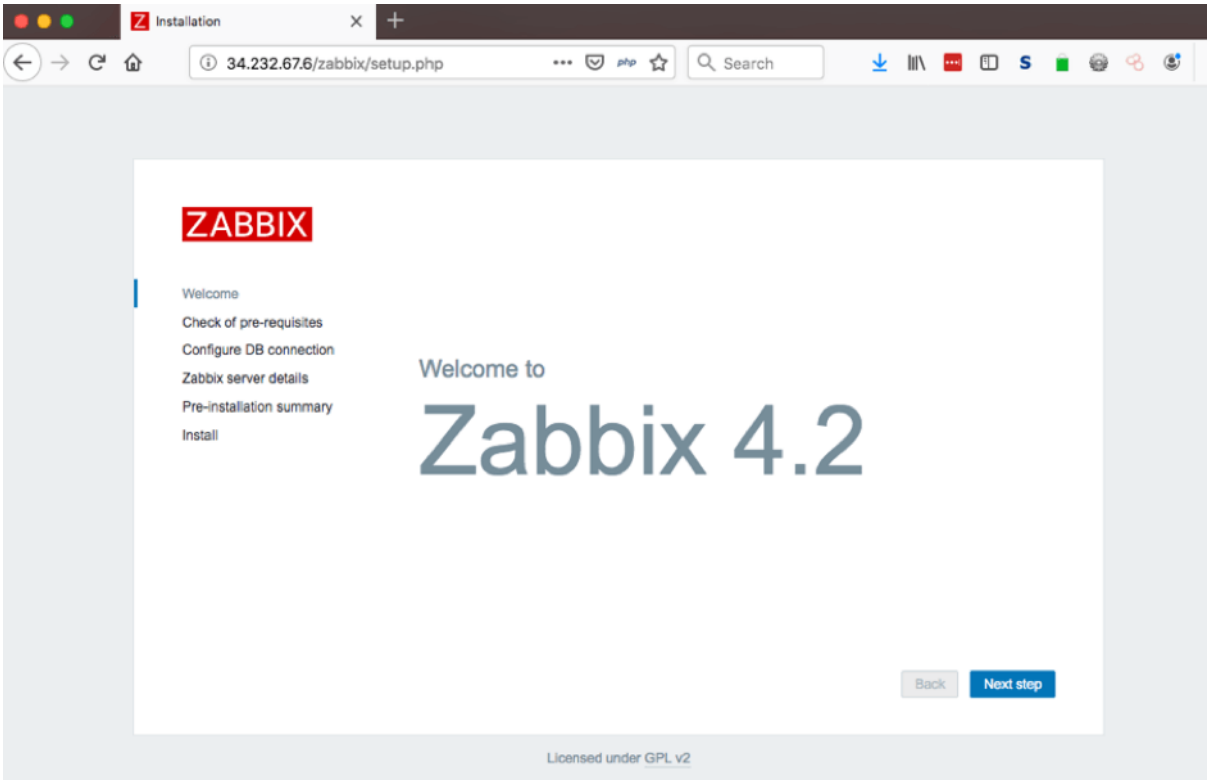


Figure 33. Zabbix frontend in browser

“Next Step” brings us to an overview of the PHP pre-requisites, and if we did the installation correctly, everything here should be in the green:

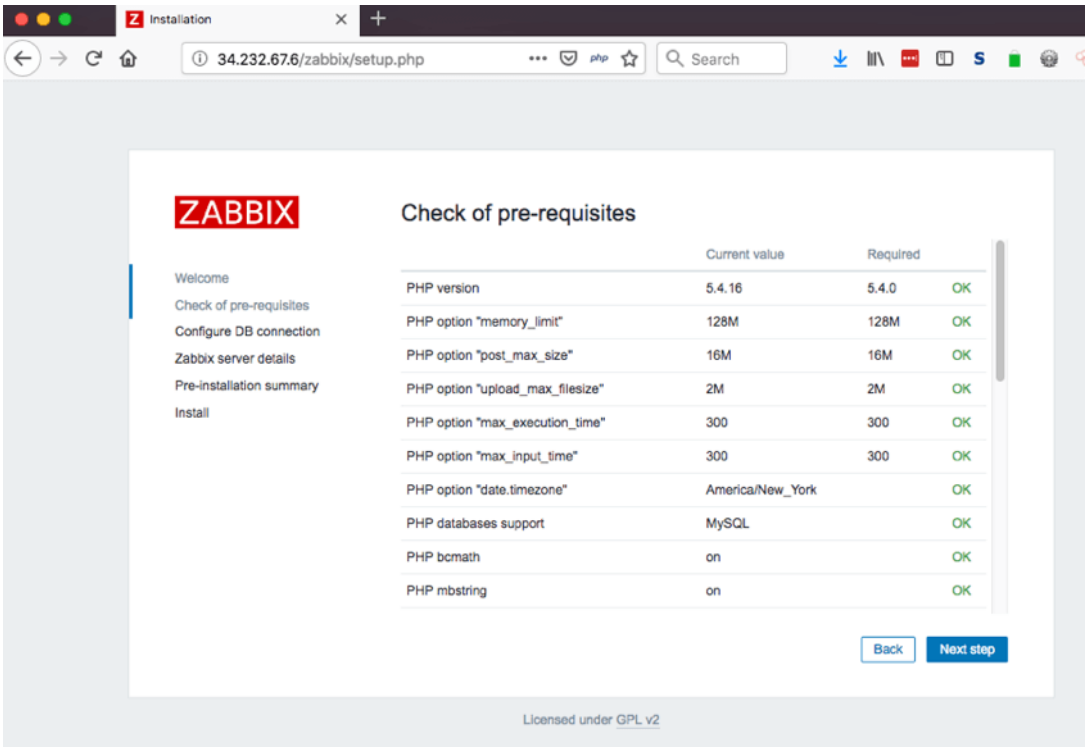


Figure 34. PHP prerequisites

The following step is a confirmation of the Zabbix MySQL database. Enter the password here for the "zabbix" user that was added earlier on.

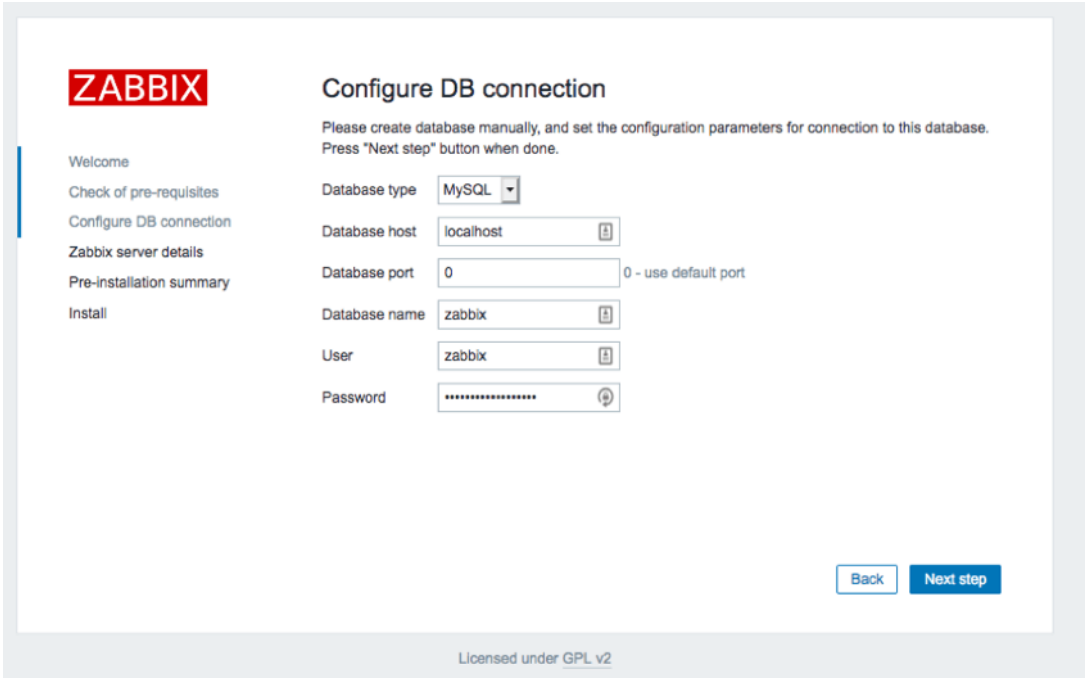


Figure 35. Enter password for the “zabbix” user

Provide a name for your Zabbix installation:

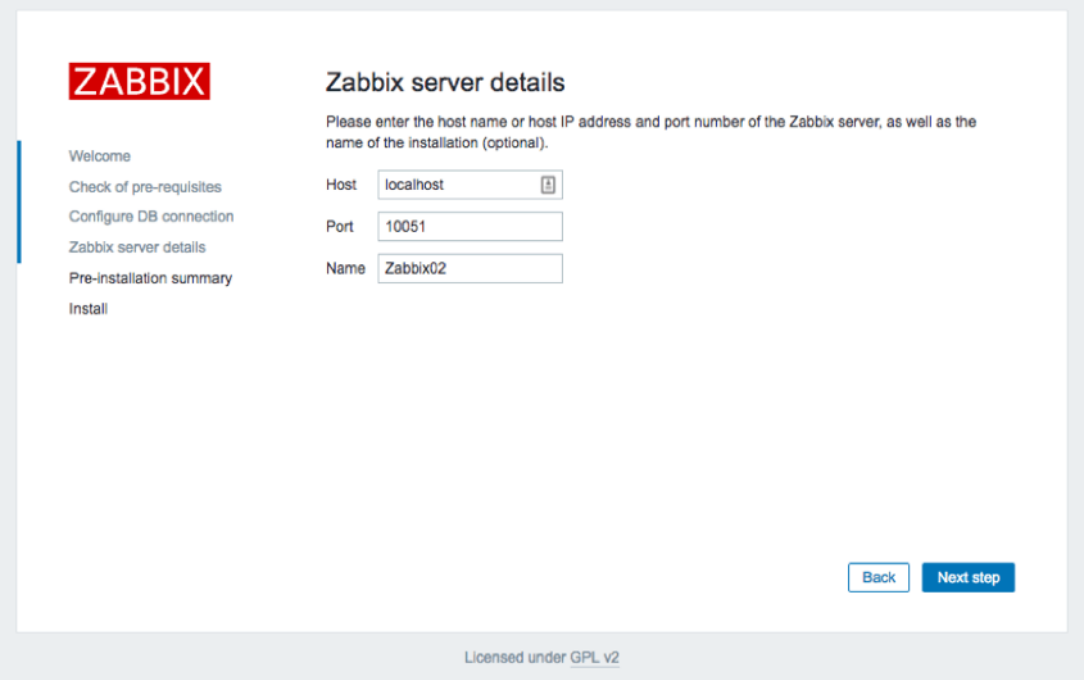


Figure 36. Enter name for the Zabbix installation

And you get one last chance to confirm all the settings:

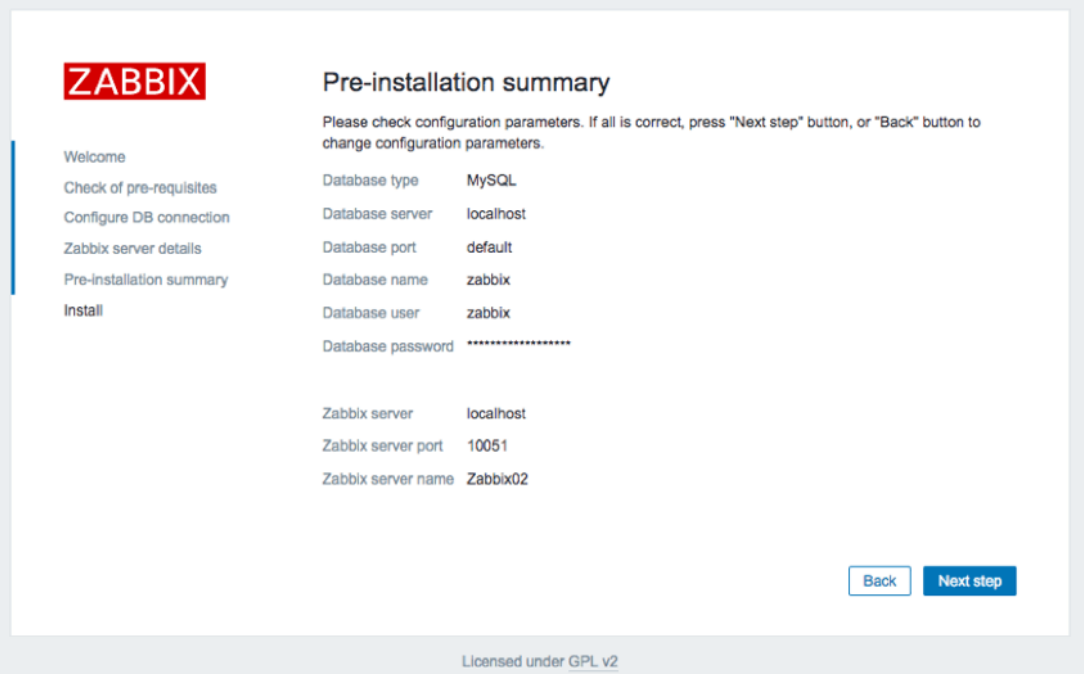


Figure 37. Pre-installation summary

And we are done:

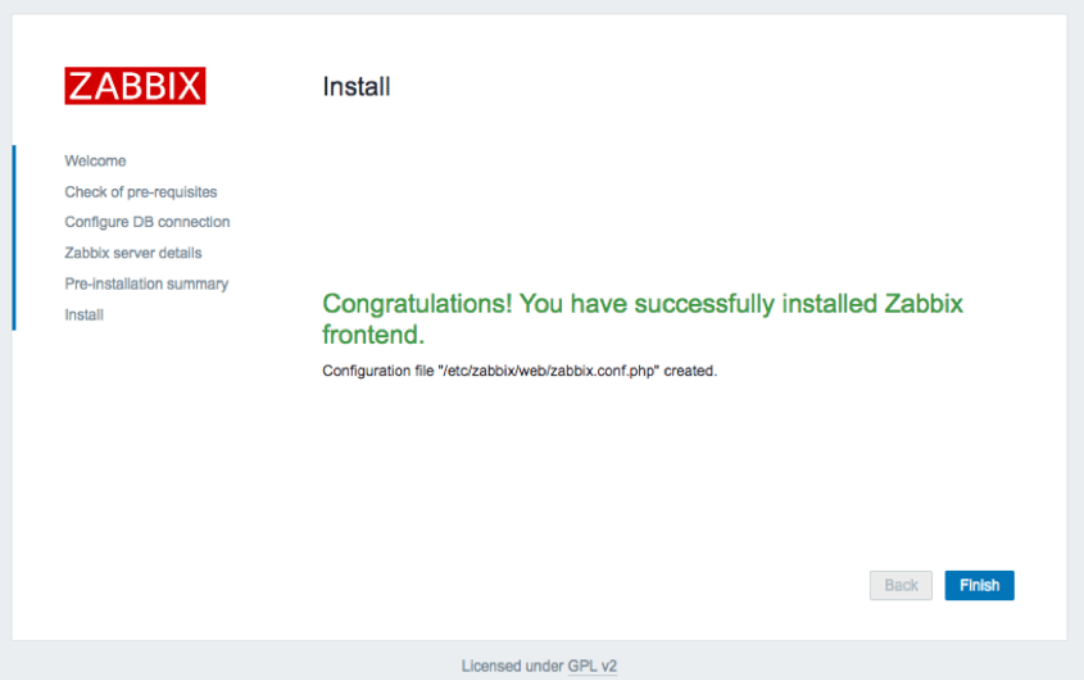


Figure 38. Zabbix frontend installation completed

Now you can log in to the Zabbix admin console. The default credentials are username Admin (with a capital!) and password zabbix.

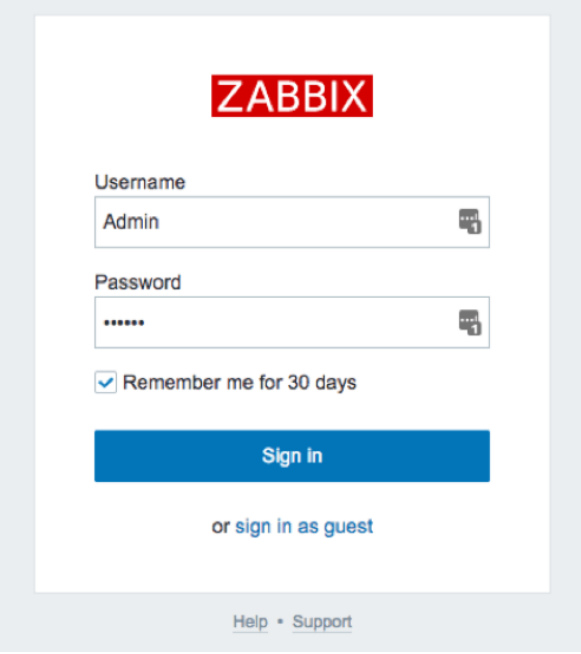


Figure 39. Log into the Zabbix admin console

Congratulations, you have a fully functional Zabbix Server:

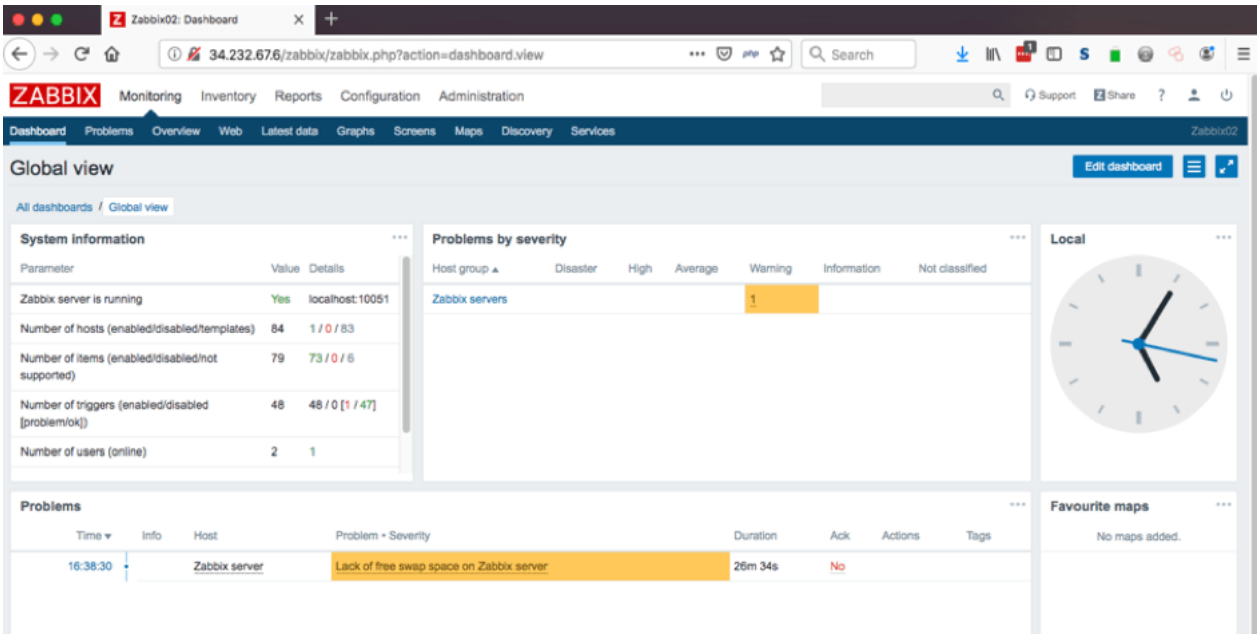


Figure 40. Zabbix Server

Next Steps

The next guides in this series explain how to install Zabbix agents on your FileMaker Servers (3 – Zabbix Agents) and how to add those FileMaker Servers as hosts to monitor here in the Zabbix Server (4 – Zabbix Configuration).

Disable MySQL Binary Logging

But there is one more important change that we want to make to MySQL before Zabbix Server starts to collect data.

We are running on an AWS t2.micro with 8GB of disk space, and at the end of the installation, we have just more than half of that disk space still available:

To check, type the following command and look at the Use % of the root directory (/):

```
df
```

```
[centos@ip-          ~]$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/xvda1       8377344 3410852   4966492  41% /
devtmpfs          483740       0    483740   0% /dev
tmpfs             506596       0    506596   0% /dev/shm
tmpfs             506596   13140    493456   3% /run
tmpfs             506596       0    506596   0% /sys/fs/cgroup
tmpfs             101320       0    101320   0% /run/user/1000
```

Figure 41. Disk space used

MySQL will collect binary logs (in folder /var/lib/mysql/) that will very quickly fill up that disk space. Those binary logs are only required if you intend to replicate this particular MySQL instance with others, and for our purpose, we do not. If you do want that default MySQL behavior, you will need to increase the disk size for this server.

In our deployment we want to disable those binary logs.

Type in:

sudo nano /etc/my.cnf

and scroll down to the section indicated in Figure 42:

```
 GNU nano 2.3.1                                File: /etc/my.cnf

# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/8.0/en/server-configuration-defaults.html

[mysqld]
#
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove the leading "# " to disable binary logging
# Binary logging captures changes between backups and is enabled by
# default. It's default setting is log_bin=binlog
# disable_log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M
#
```

Figure 42. Scroll down to "# disable_log_bin"

Remove the "#" at the start of the line so that "disable_log_bin" becomes active:

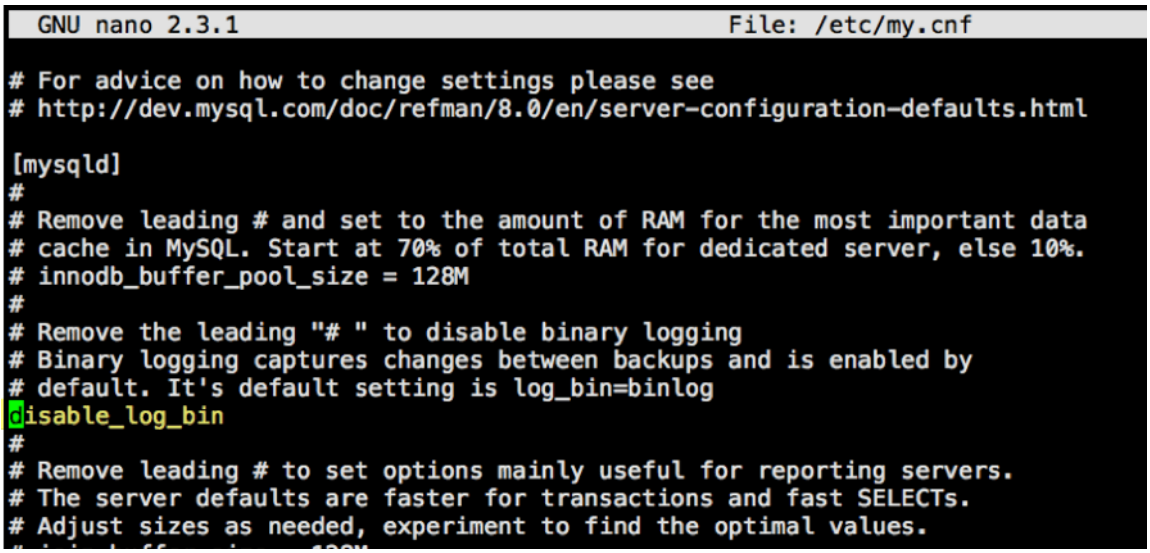


Figure 43. Remove the "#" from the line

Hit control-o and then enter to save the changes and then control-x to quit the text editor.

Restart MySQL for the change to take effect:

```
sudo systemctl restart mysqld
```

On to the next guide and installing Zabbix Agents (3 – Zabbix Agents).

Monitoring Your FileMaker Server

Installing Zabbix Agent

By Wim Decorte, Senior Technical Solution Architect
and Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Contents

This document is one in a series of guides that walk you through installing, configuring, and using Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix.

Do We Need an Agent?

Zabbix agents are responsible for collecting data from the host (FileMaker Server) being monitored. While Zabbix server can monitor servers and devices without the presence of an agent on the host, the amount of data you can collect, and its relevance, would be much less.

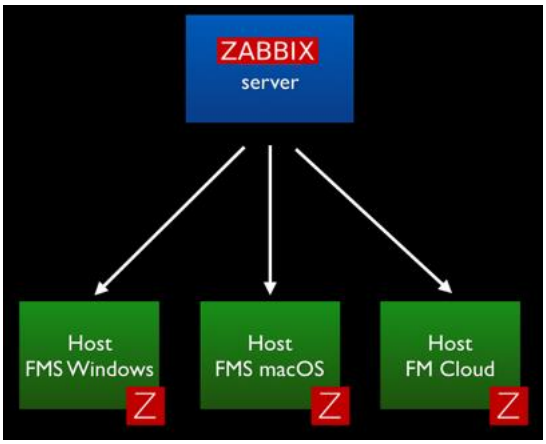


Figure 1. Zabbix Server

The agent is a small piece of software that runs completely in the background as a service/daemon. It is designed to be lightweight so that its monitoring activity does not affect the host that it is monitoring. These agents exist for all three of the platforms that matter for us: Windows, macOS and CentOS (FileMaker Cloud).

The Zabbix agent footprint is small. As an example, the screenshots below are from one of our Zabbix servers that monitors four development FileMaker Servers. Over the course of three months, the processor time for the Zabbix agent did not exceed 1% and used about 20MB of memory.

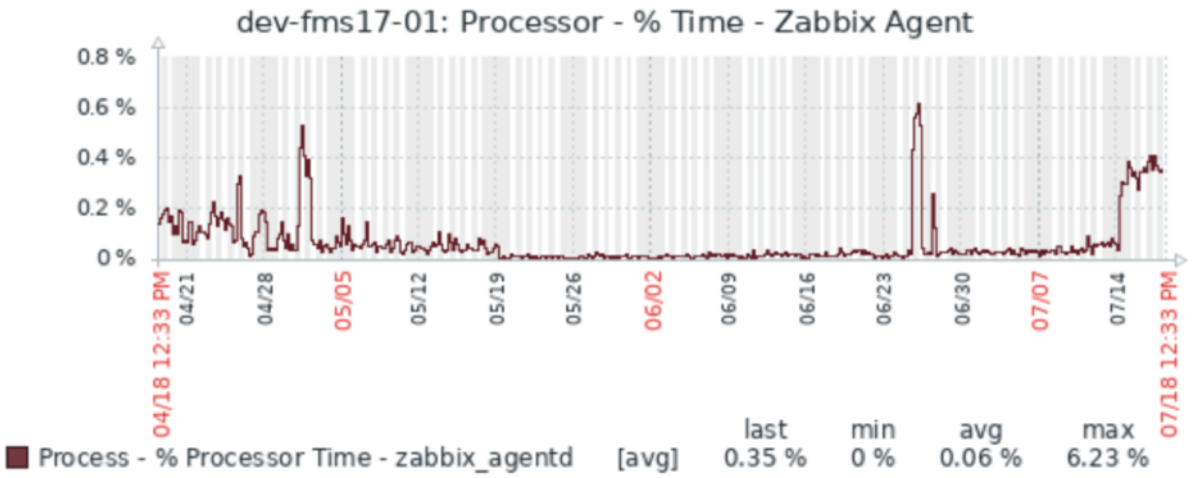


Figure 2. Zabbix Agent - processor time

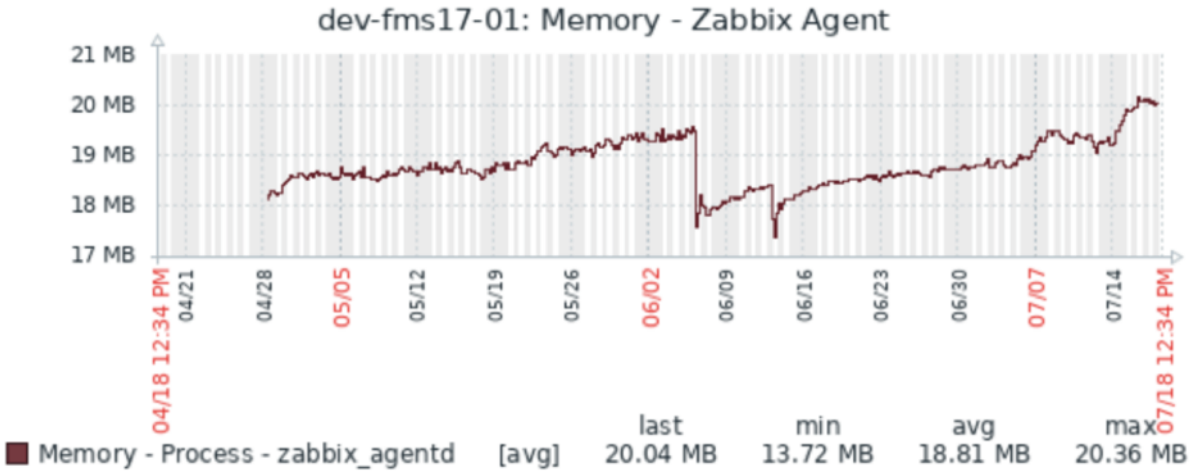


Figure 3. Zabbix Agent – memory

Active or Passive Agent and Firewall ports

Agents can operate in two modes – active or passive – and the difference can matter to you in terms of whether you are comfortable with opening an extra port on the FileMaker Server.

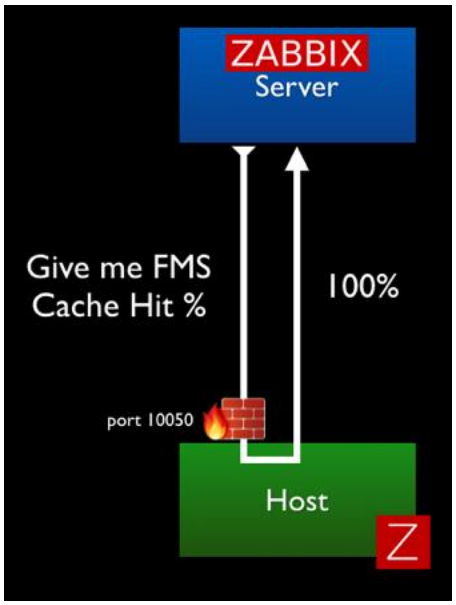


Figure 4. Zabbix Server – passive mode

In Passive mode, the agent does not do anything at all until it is asked to do something by the Zabbix server. The communication originates from the Zabbix server and requires port 10050[1] to open on the FileMaker Server to allow that incoming traffic.

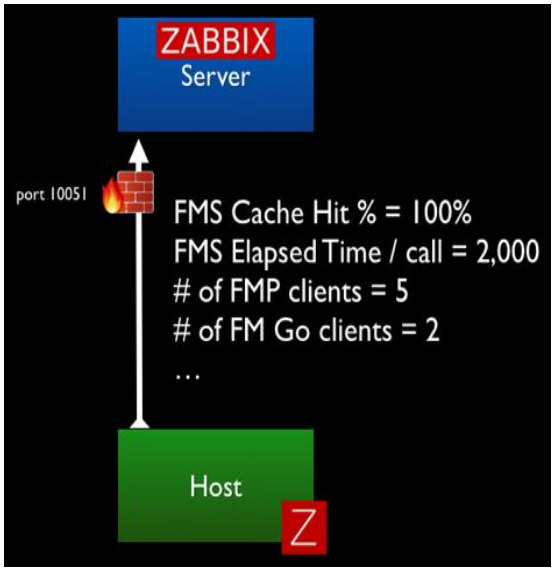


Figure 5. Zabbix Server – active move

In Active mode, the agent collects all of the required data on its own (based on the interval set for each item it collects data for) and sends that data to the Zabbix server. In this scenario all communication originates from the FileMaker Server; no ports need to be opened on the FileMaker Server. The Zabbix port 10051 needs to be open on the Zabbix server to accept the incoming data.

To use all of Zabbix's functionality, including the ability for Zabbix server to send remote commands to your FileMaker Server (for instance to restart the scripting engine), you'll need to allow traffic in both directions.

[1] These ports can be customized as we will show later.

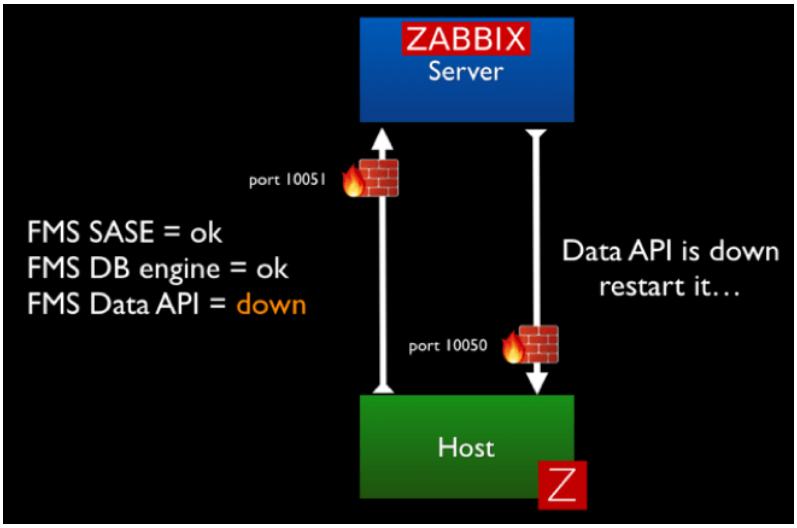


Figure 6. Allow traffic in both directions

Based on your security requirements, you can decide to forego some of the functionality around triggering remote actions and opt for a more locked-down deployment.

Installing the Agent

For macOS and Windows you can download the agent from the Zabbix download page. For FileMaker Cloud the installation is done through the CentOS software manager command line.

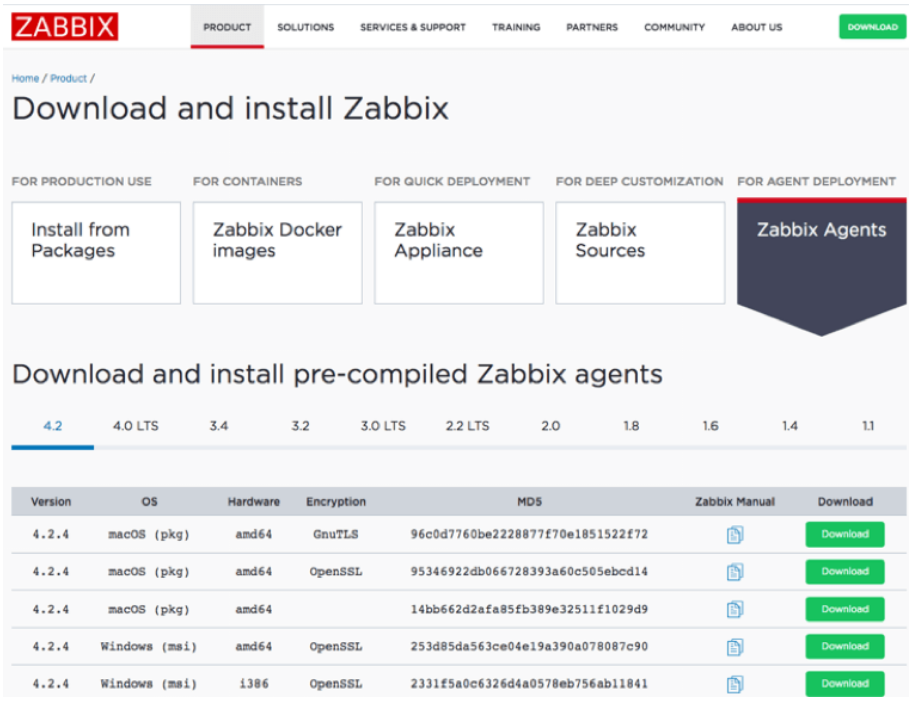


Figure 7. Zabbix download page

Note that you have multiple choices per platform depending on the encryption engine (GnuTLS, OpenSSL, no encryption). The main reason for offering different encryption engine options is so that if a vulnerability were to be discovered in one encryption platform, we can fairly seamlessly switch to another. In that sense, you can pick whichever one you prefer. There is no functional difference between the choices.

Installing on Windows

The Zabbix agent for Windows comes as a standard installer with the usual wizard that will walk you through some of the needed basic configuration details.

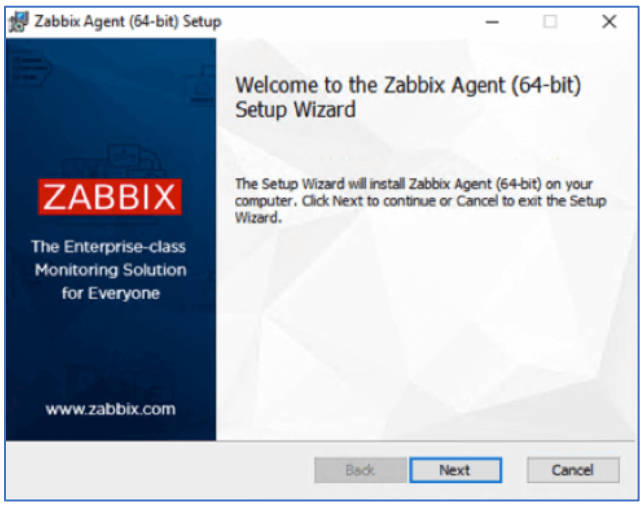


Figure 8. Zabbix Agent Windows installer

The choices you make on the next screen can all be modified in the Zabbix agent config file as will be shown later in this guide.

The Host Name gets set by default to the host name of your Windows machine. You can change it to something meaningful, provided that it is unique. The Host Name will be shown on the Zabbix server dashboard and is used when you set up a new host to monitor on your Zabbix server.

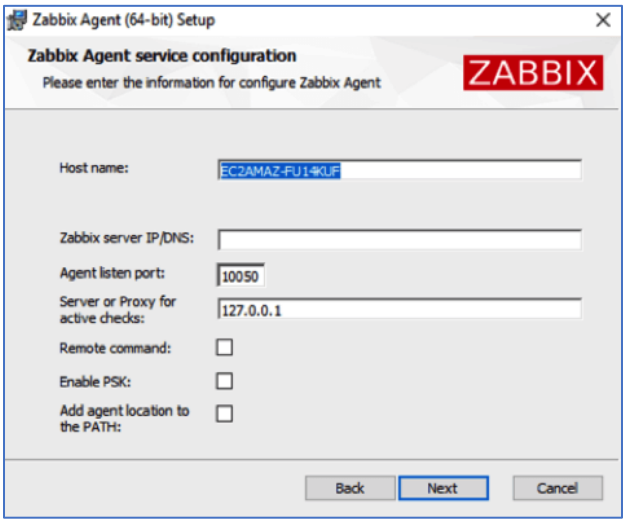


Figure 9. You can change the default host name

The **Zabbix server DNS name or IP address** will be used to inform the agent where to send data and as a security measure so that the agent will only respond to incoming traffic (passive requests for data, remote commands) from the Zabbix server(s) listed.

Port 10050 is the default port on the Agent-side to listen to those incoming requests. This port needs to be opened on your FileMaker Server's firewall or your perimeter firewall for your network and then forwarded from your router to your FileMaker Server. If your FileMaker Server is hosted on AWS or a similar provider, remember to adjust the inbound rules there. If you would rather not use the default port, you can adjust it here (or later by modifying the config file).

Typically, you would use the same DNS name or IP address for the **Server or Proxy for Active Checks** as you have for the Zabbix server DNS name earlier. This setting decides where the Agent will send the data it collects for Active items (where the agent does not get prompted by the server to collect data). In complex deployments, you could use a different Zabbix server or a Zabbix proxy for these active checks.

The **Remote command** toggle is to decide whether you will allow this Agent to accept remote commands from the Zabbix server listed. We do use this functionality in our FileMaker Server templates to restart processes like the FileMaker Server scripting engine, Data API, or Web Publishing Engine if they have stopped running.

By **enabling PSK**, you encrypt the traffic between the Agent and the Server through a Pre-shared Key. This security scheme is similar to how most Wi-Fi networks work.

Enabling the option to **add the agent location to the PATH** will ensure that you can use the Zabbix agent command line commands from anywhere on the machine without first having to navigate to where those executables are. That is similar to how "fmsadmin" works on your FileMaker Server; the FileMaker Server installer does this automatically.

For our deployment, the configuration looks like Figure 10.

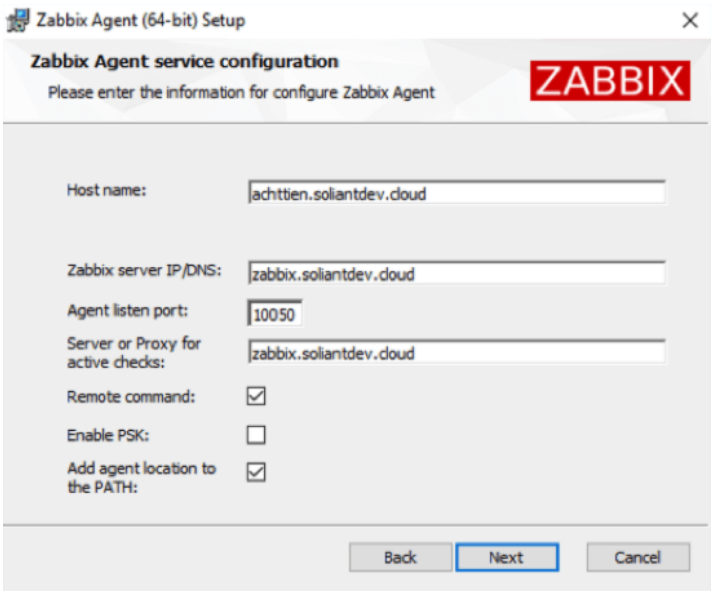


Figure 10. Deployment configuration

The core of the Zabbix agent is the 'Agent Daemon'. By default, the installer will also install the Zabbix Sender and Zabbix Get, which are command line tools to manually initiate sending data to the Zabbix server or retrieve information from the Zabbix server about what active items for which to collect data.

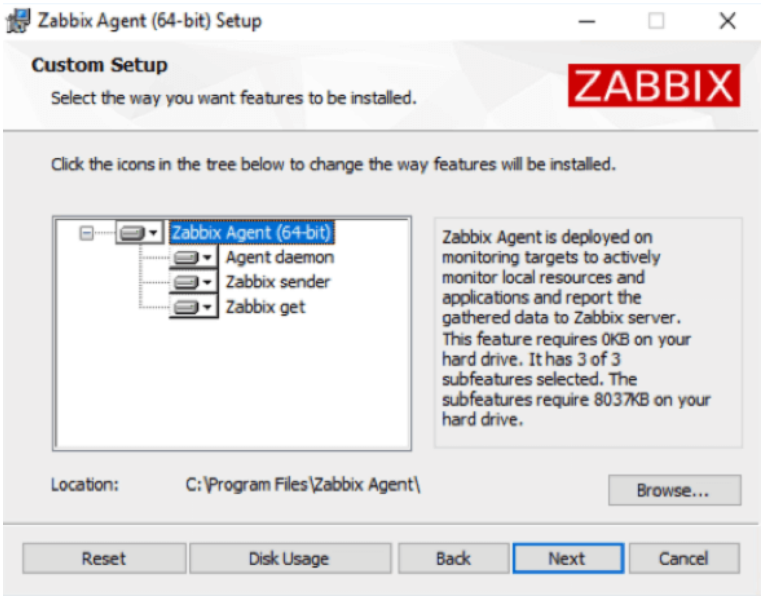


Figure 11. Custom setup

When the installer has completed, you will find the Zabbix agent listed among the Windows services. Like most background services, it runs under the "local system" account.

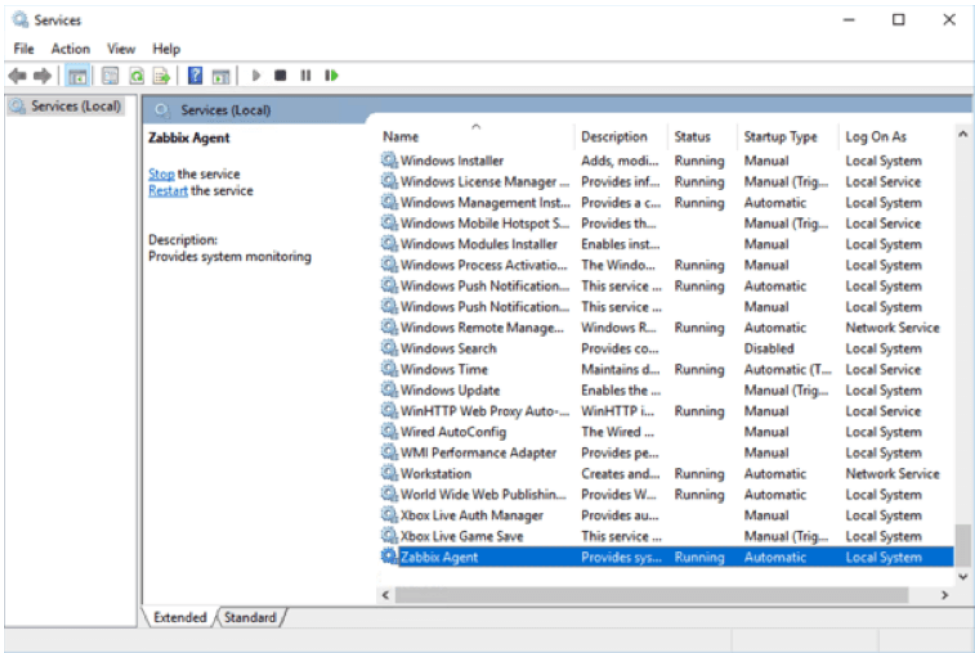


Figure 12. Zabbix Agent shown under Windows services

The log file for troubleshooting is in the Zabbix agent install location under “Program Files”:

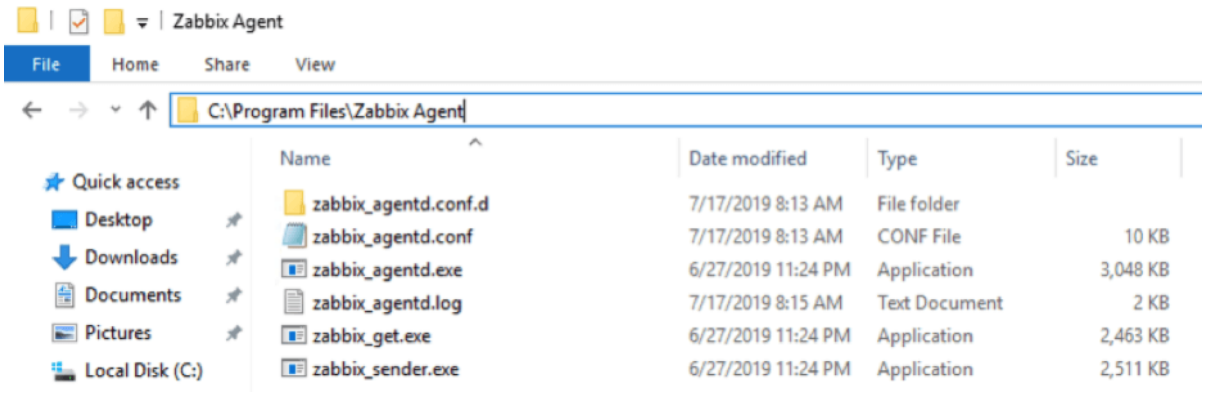


Figure 13. Log file

Later in this guide, we will make some modifications to the **zabbix_agentd.conf** file, located in this same folder, to further tweak our deployment.

Installing on macOS

Similarly, on macOS the Agent’s installer, will walk you through the standard wizard:

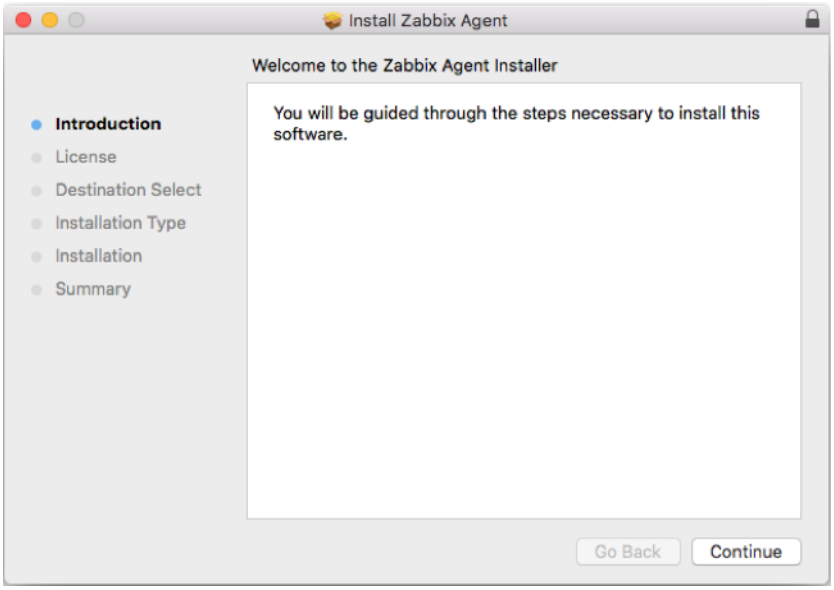


Figure 14. Zabbix Agent macOS installer

But it will not provide any options to change configuration settings up-front. We will show you how to modify the config file to set the relevant options.

The macOS installer adds a Zabbix user account responsible for running the daemon. This will be relevant later on when we make our configuration changes.

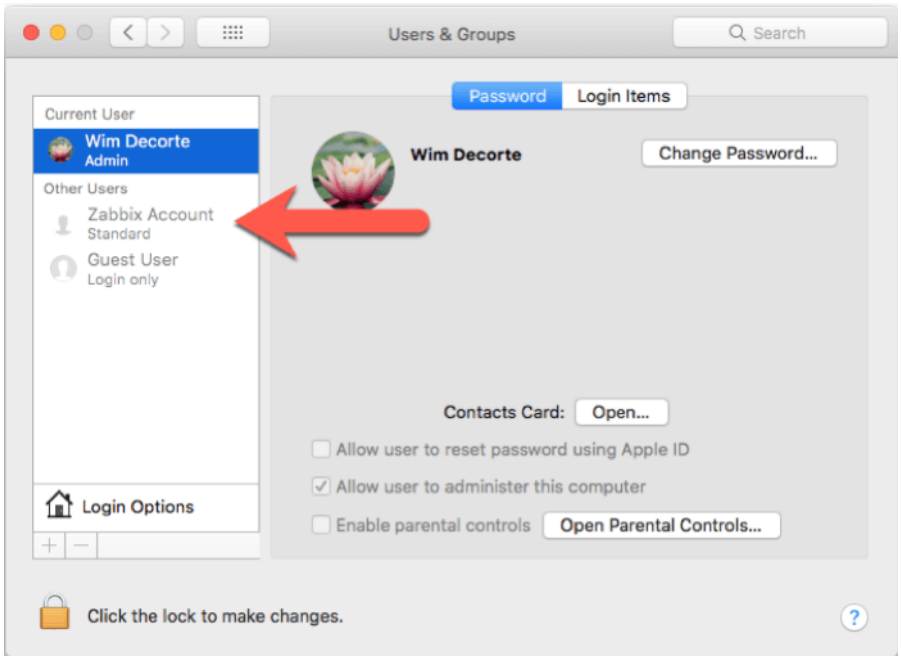


Figure 15. Zabbix user account is added during installation

Adding the zabbix user to sudoers

As part of our Zabbix template, we use some of the macOS and FileMaker Server command line functionality to collect (and take action on) data for items we monitor. As such, the Zabbix agent user needs the right level of privileges to execute those commands.

To make this work seamlessly through the security features available in macOS, we will use the sudoers file.

First off, open Terminal and type in this command to create a new file in the sudoers folder:

```
sudo nano /etc/sudoers.d/zabbix_nopasswd
```

In the nano text editor window:

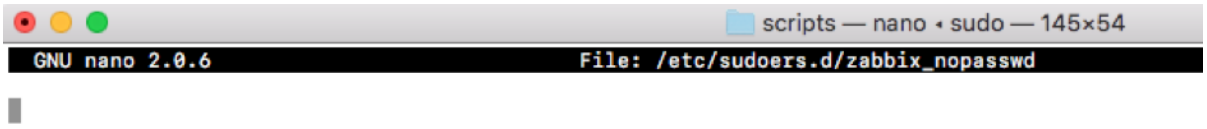


Figure 16. Nano text editor

Type in or paste in the following line:

```
zabbix ALL=(ALL) NOPASSWD: ALL
```

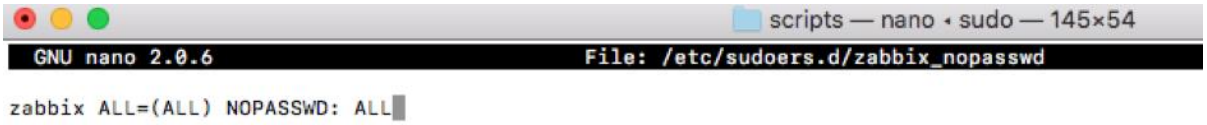


Figure 17. Edit file added to sudoers folder

Hit control-o and then enter to save the file and then control-x to quit out of the nano text editor and return back to the command line.

Type in the following command to restrict that new file's access level so that it is read-only for the owner of the file and the group to which the owner belongs. (This further protects it from inadvertent changes.)

```
sudo chmod 0440 /etc/sudoers.d/zabbix_nopasswd
```

With this done, we'll instruct macOS to read this new file when evaluating the rights of a certain user to run commands as Super-User (aka the **su** in sudo).

The safest way to do this is through the command line utility **visudo**. **vi** is another text editor, and **visudo** is made specifically to be the text editor for everything that has to do with changes to the super-user do (sudo) levels.

```
sudo visudo
```

The Terminal window will change to the **vim** text editor (a variant of the **vi** text editor) – which most people find difficult to work with unless you use it very often.

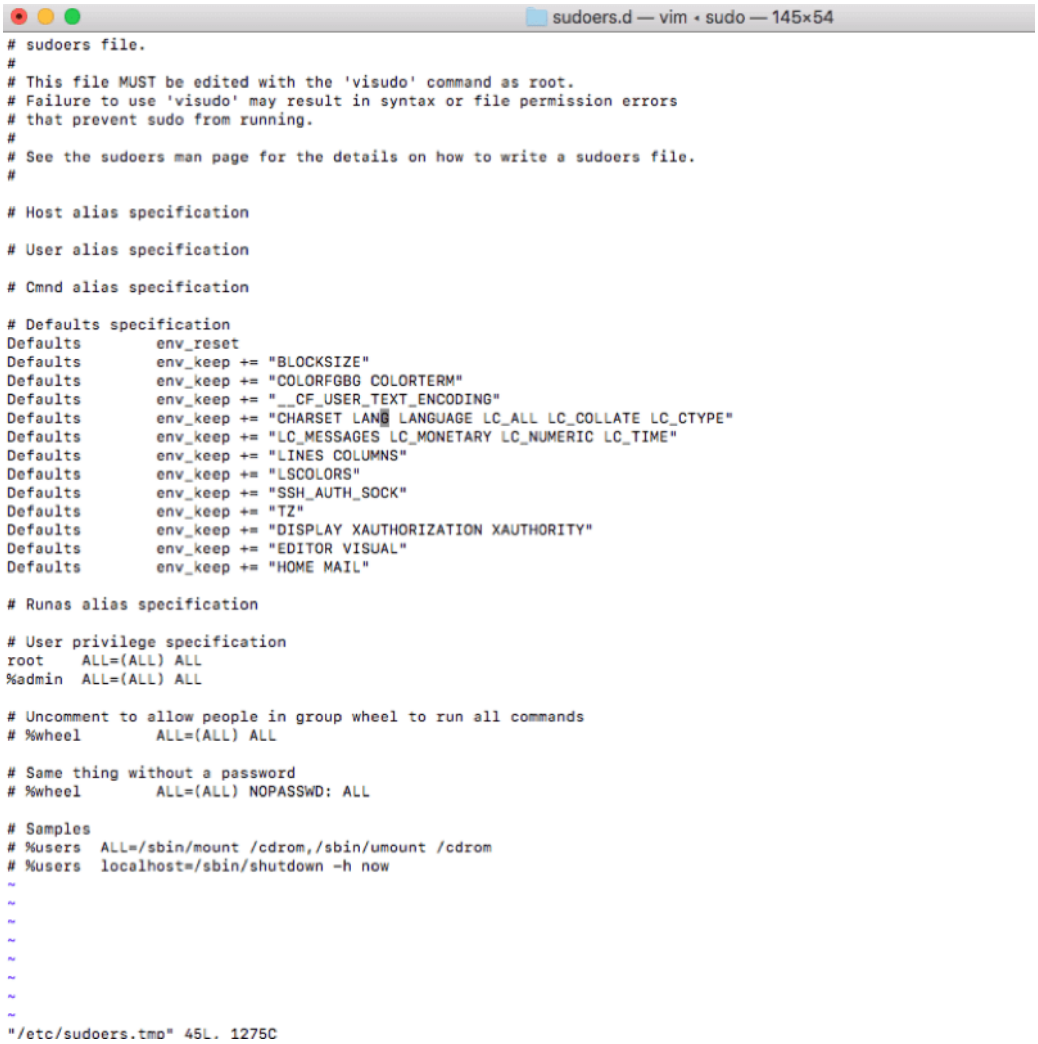


Figure 18. Vim text editor

Using the arrow keys, scroll down to the bottom of the file and hit **i** on your keyboard to enter edit (insert) mode. The bottom of the window will confirm that you are in edit mode:

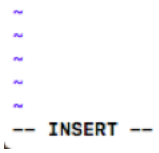


*Figure 19. Scroll to the bottom of the file and hit the "**i**" key*

Add the following two lines:

Read drop-in files from /etc/sudoers.d (## indicates a comment line; # does not)
#includedir /etc/sudoers.d

To exit edit mode, hit **escape** on your keyboard and type in **:wq** and then enter to save the document and quit vim. This will place you back on the command line.

The syntax of the file we have just added to the sudoers folder is crucial to the operating system. To ensure you did not make any syntax errors use this command:

sudo visudo -cf /etc/sudoers.d/zabbix_nopasswd

When all is well, you should see "parsed OK" in the result of that command:

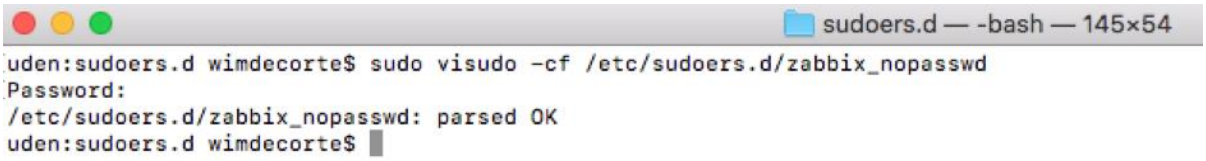


Figure 20. "parsed OK" is shown when done

Python requests module

As part of our template, we will use a small Python script on the FileMaker Server machine (PowerShell on Windows) to communicate with the FileMaker Server Admin API and retrieve configuration settings. The Admin API is only available in FileMaker Server 18 (and in 17 until its expiry on September 27, 2019).

Recent versions of macOS have Python 2.7.10 installed by default[2] so we made sure that the Python script is compatible with that – somewhat old – version of Python. To make the REST request to the Admin API, we want to use Python's requests module.

[2] See https://opensource.apple.com/, for each version of macOS, you can click through to see what version of Python was installed. Python 2.7.10 is included in all versions since 10.10 (Yosemite). Because FileMaker Server 18 requires macOS 10.13 (High Sierra) or 10.14 (Mojave) and 17 requires macOS 10.12 (Sierra) or 10.13 (High Sierra), we know that the right version of Python is available on all macOS servers running FileMaker Server that support the Admin API.

That module, however, is missing from the standard macOS Python installation and so is Python's software package installer (pip).

First, we need to install **pip**:

sudo python -m ensurepip --default-pip

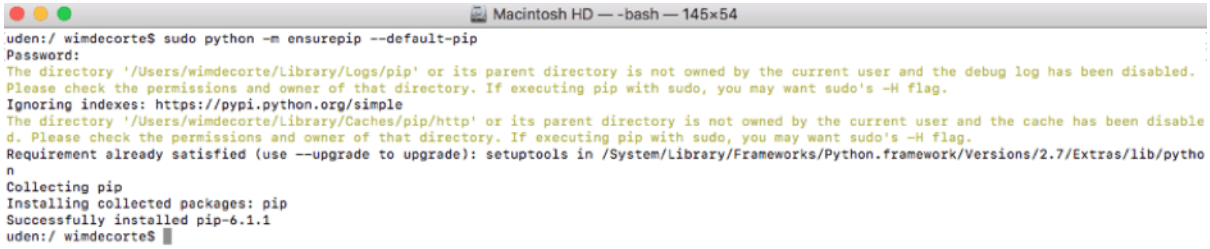


Figure 21. Installing pip

And with pip installed, we can install the requests module:

sudo python -m pip install requests

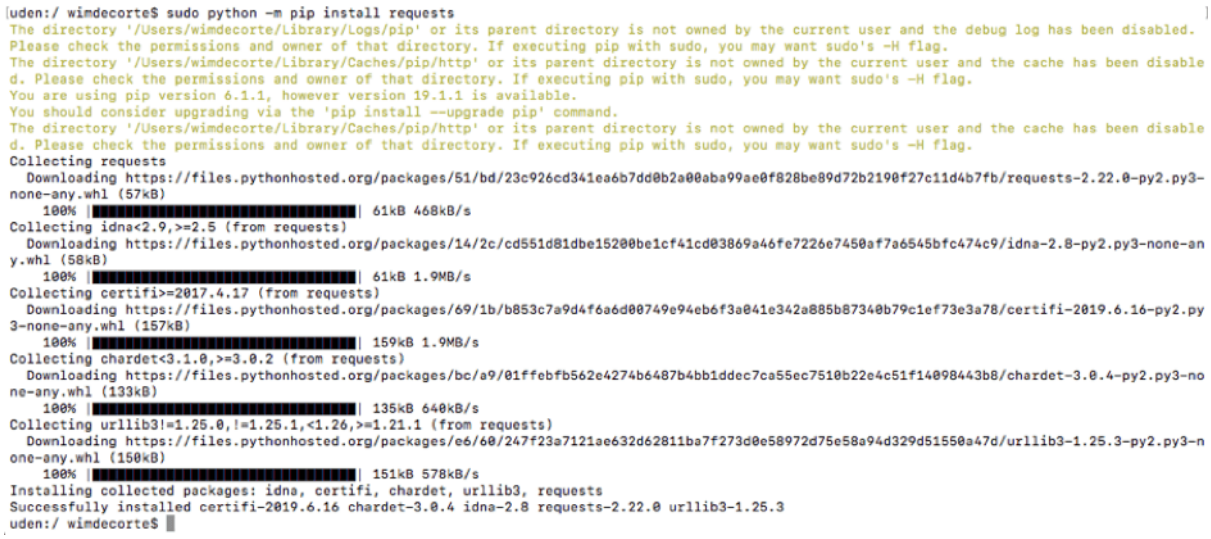


Figure 22. Installing the request module

Starting, Stopping the agent and where to find the log file

To start the agent, use this command in Terminal:

sudo launchctl start com.zabbix.zabbix_agentd

Or, use **stop** to stop the agent, particularly after making changes to the Zabbix agent config file which necessitates an agent restart.

The log file is in this folder: **/var/log/Zabbix/Zabbix_agentd.log** and contains very useful troubleshooting information.

Installing on FileMaker Cloud

FileMaker Cloud runs on Linux CentOS. The Zabbix downloads page does not offer a pre-compiled agent for that operating system. Instead, all software installations on CentOS are done through its built-in command line software package manager: yum[3].

Since we need access to the command line, we need to establish an SSH connection to the server. FileMaker Cloud instances do not allow this by default, so we need to change the inbound rules in the AWS EC2 console. Select your FileMaker Cloud instance and click on the security group that applies to it:

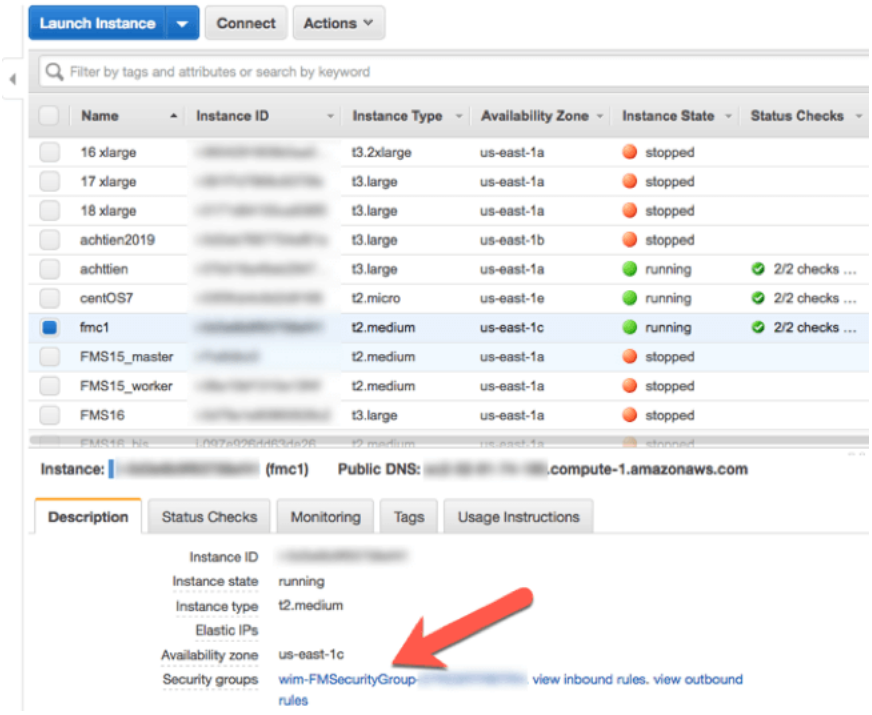


Figure 23. Click on the security group for the selected FileMaker Cloud instance

In the security group settings, select inbound rules and adjust them so that:

- Port 22 (SSH) is allowed but only from your IP address
- Port 10050 is allowed but only from the IP address of your Zabbix server

[3] Yum = Yellow dog Updater, Modified. If you are familiar with other flavors of Linux, it is the equivalent of "apt-get".

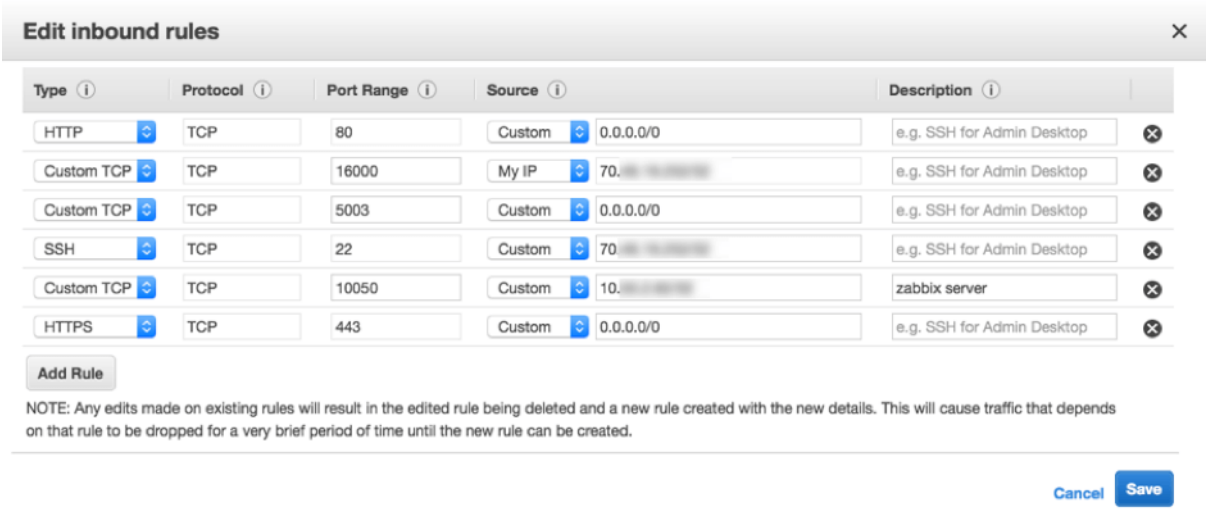


Figure 24. Inbound rules

With this done, we can now open Terminal on macOS or your favorite SSH client on Windows and connect to the FileMaker Cloud instance:

ssh -i /Users/wimdecorte/Documents/projects/ETS18/zabbix_resources/wim_ets_15.pem centos@<IP or DNS name of your FileMaker Cloud instance>

All SSH connections to AWS instances require the use of the pem file (certificate) that was used to create the instance. You can do this by specifying the **-i** and the path to that pem file. **centos** is the default user name to log into CentOS Linux.

Before we go on, we have to mention a big caveat: any and all configuration changes that we make from this point forward may get lost through the automatic updates that happen on FileMaker Cloud instances. There is nothing that can be done about this, since that is the architecture of FileMaker Cloud. A FileMaker Cloud instance consists of a number of drives, one of which holds your FileMaker Data and all the FileMaker Server configuration settings. The other drives hold the Linux operating system and its configuration, and those drives get replaced from time to time with Linux system updates.

What does this mean for Zabbix monitoring? After a FileMaker Cloud upgrade, you may have to redo the steps in this section, so it is a good idea to save a copy of the configuration file after making changes to it.

The very first thing we will do is install nano, our favorite Linux text editor. We will need it to change the Zabbix agent configuration.

Type in:

sudo yum install nano

As with all installations and updates, you will see a bit of an overview of what will happen, and you will be asked to confirm with “Y” that you want to proceed:

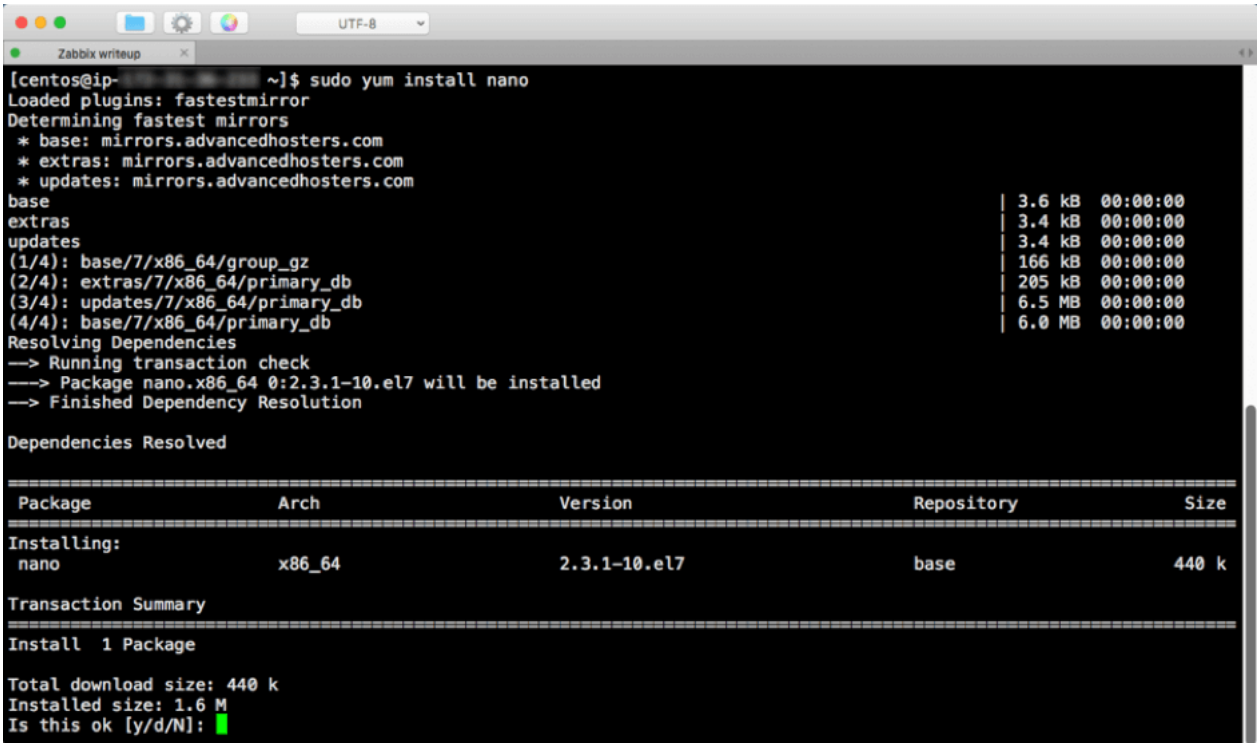


Figure 25. Type “Y” to proceed

A few seconds later, we will be done:

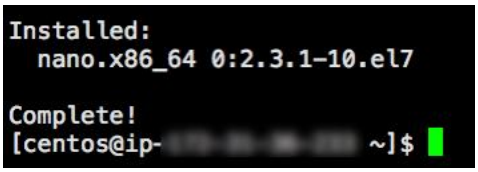


Figure 26. Nano installation completed

Type in these two commands:

sudo firewall-cmd --zone=public --add-port=10050/tcp –permanent

sudo firewall-cmd --reload

You can check what ports are open with this command, to confirm that the port is now open:

sudo firewall-cmd --list-ports

Figure 27. View ports that are open

Yum, the software package manager used by CentOS, keeps a list of repositories with available software that can be installed. The Zabbix repository is not listed by default, so we will need to add it with this command:

sudo rpm -Uvh https://repo.zabbix.com/zabbix/4.2/rhel/7/x86_64/zabbix-release-4.2-1.el7.noarch.rpm

followed by this command to tell yum to do some internal housekeeping:

sudo yum clean all

And finally, we can run the command to install the Zabbix agent:

sudo yum install -y zabbix-agent

And these two commands to start it and set it to auto-start whenever the machine boots:

sudo systemctl start zabbix-agent

sudo systemctl enable zabbix-agent

The next section of this guide will step you through the Zabbix agent configuration.

Configuration changes for Zabbix agent

On Windows, the configuration file will be in **C:\Program Files\Zabbix Agent** unless you changed the installation location during the install. On macOS you will find the configuration file in **/usr/local/etc/zabbix/**. And on FileMaker Cloud it is located in **/etc/zabbix/**.

The configuration file is always named **zabbix_agentd.conf**, and its content is the same on all platforms.

On Windows, the installer will have asked for some configuration options already. However, this will not have happened on macOS and FileMaker Cloud, so we will review all the changes here that make our Zabbix server installation work, specifically for monitoring a FileMaker Server.

On Windows, we usually install Notepad++, which allows us to create a custom 'language' that colors all the comments in green for easy reading:

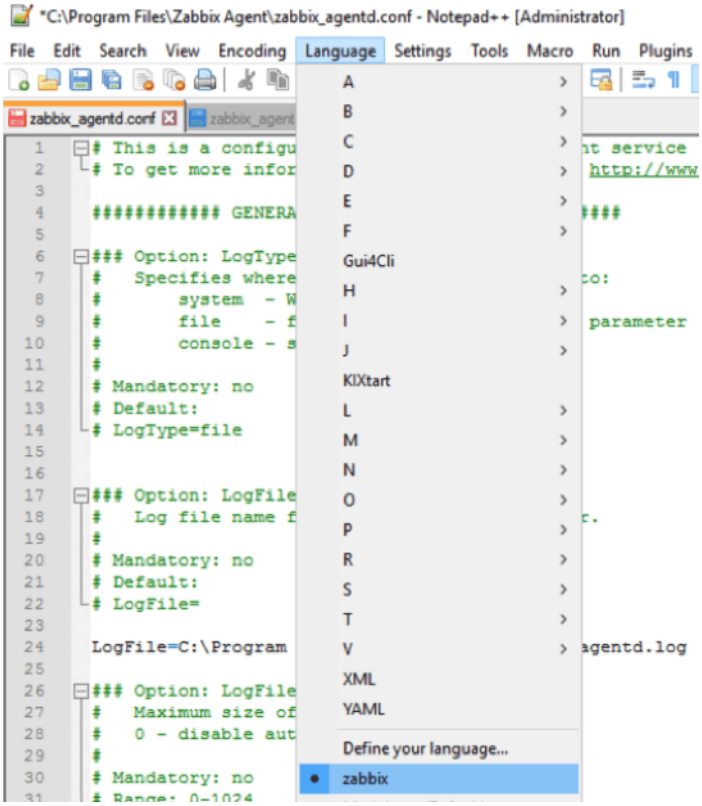


Figure 28. Notepad ++ on. Windows

You can download that language file at https://github.com/soliantconsulting/FileMaker-Server-Zabbix-Templates.

On macOS and FileMaker Cloud, you can edit the config file from the command line by using the nano text editor:

macOS:

```
sudo nano /usr/local/etc/zabbix/zabbix_agentd.conf
```

FileMaker Cloud:

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

Using the command line on both macOS and FileMaker Cloud ensures that the privileges on the file do not change. On macOS, you could certainly use your favorite text editor but make sure that the privileges do not change from what they need to be for the Zabbix agent to work properly:

```
uden:zabbix wimdecorte$ pwd
/usr/local/etc/zabbix
uden:zabbix wimdecorte$ ls -al
total 24
drwxr-xr-x  4 root  wheel    136 Jul 17 08:30 .
drwxr-xr-x  3 root  wheel    102 Jun 27 05:32 ..
drwxr-xr-x  4 root  wheel    136 Jul 17 08:30 zabbix_agentd
-rw-r--r--  1 root  wheel  10837 Jun 27 05:32 zabbix_agentd.conf
uden:zabbix wimdecorte$
```

Figure 29. Ensure privileges do not change

Enable Remote Commands

This setting allows Zabbix server to send commands to the FileMaker server as part of a configured Action; for instance, to restart the FileMaker Server Scripting Engine process when it fails. If you enable this setting, we also recommend enabling the setting that logs each executed remote command. But note that doing so will result in the FileMaker Server admin console credentials being included in the agent log for all of the items and remote actions that rely on the fmsadmin utility. (We will cover items and actions in more detail in the Zabbix Configuration white paper.)

Note that from a security point of view, the Zabbix agent will only accept remote commands from servers listed in the “Active” section (see later).

```
55
56  ### Option: EnableRemoteCommands
57  #   Whether remote commands from Zabbix server are allowed.
58  #   0 - not allowed
59  #   1 - allowed
60  #
61  # Mandatory: no
62  # Default:
63  # EnableRemoteCommands=0
64  EnableRemoteCommands=1
65
66  ### Option: LogRemoteCommands
67  #   Enable logging of executed shell commands as warnings.
68  #   0 - disabled
69  #   1 - enabled
70  #
71  # Mandatory: no
72  # Default:
73  LogRemoteCommands=1
74
```

Figure 30. Remote commands enabled

Set Zabbix server & the port that the Agent listens to

These settings are relevant for passive checks, where Zabbix server talks to the agent to ask it to collect data for a monitored item or to run a remote command.

We have left the port setting at its default of 10050, but this is where you can change it. The port is also specified in the Zabbix frontend and, as was discussed earlier, in the firewall settings. If you end up changing it in the configuration file, don't forget to also change it in those other places.

```
77  ### Option: Server
78  #   List of comma delimited IP addresses, optionally in CIDR notation, or DNS
79  #   Incoming connections will be accepted only from the hosts listed here.
80  #   If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.
81  #   '0.0.0.0/0' can be used to allow any IPv4 address.
82  #   Example: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.domain
83  #
84  # Mandatory: yes, if StartAgents is not explicitly set to 0
85  # Default:
86  # Server=
87
88  Server=zabbix.soliantdev.cloud
89
90  ### Option: ListenPort
91  #   Agent will listen on this port for connections from the server.
92  #
93  # Mandatory: no
94  # Range: 1024-32767
95  # Default:
96  # ListenPort=10050
97
```

Figure 31. This is where the port setting can be changed

The Zabbix agent will only listen to requests from the server that is listed here.

Set Zabbix server to send data to

The previous section determines which Zabbix server the agent will listen to, and this section defines which Zabbix server the agent will send its data to for Active[4] items.

```
117
118  ### Option: ServerActive
119  #    List of comma delimited IP:port (or DNS name:por
120  #    If port is not specified, default port is used.
121  #    IPv6 addresses must be enclosed in square bracke
122  #    If port is not specified, square brackets for IP
123  #    If this parameter is not specified, active check
124  #    Example: ServerActive=127.0.0.1:20051,zabbix.dom
125  #
126  # Mandatory: no
127  # Default:
128  # ServerActive=
129
130  ServerActive=zabbix.soliantdev.cloud
131
```

Figure 32. Send data for Active items to the specified Zabbix server

Hostname

The hostname will be used to reference the FileMaker server on which the Agent is running. The same name will be used when setting up the monitored host in the Zabbix frontend. It needs to be unique among all the servers monitored by the Zabbix server. Using the DNS name of the FileMaker Server is an easy way to ensure that.

```
131
132  ### Option: Hostname
133  #    Unique, case sensitive hostname.
134  #    Required for active checks and must match hostname as configured on the server.
135  #    Value is acquired from HostnameItem if undefined.
136  #
137  # Mandatory: no
138  # Default:
139  # Hostname=
140
141  Hostname=achttien.soliantdev.cloud
142
```

Figure 33. Define the host name

[4] Active vs. Passive is described earlier in this document.

Advanced Parameters – Timeout

The timeout setting is located a lot further down in the config file, and it specifies how long the Zabbix agent is going to spend on any one request. The default is three seconds, but we will ask it to do some things that could take longer as you will see later.

```
223
224 ### Option: Timeout
225 #    Spend no more than Timeout seconds on processing.
226 #
227 # Mandatory: no
228 # Range: 1-30
229 # Default:
230 Timeout=30
231
```

Figure 34. Setting the timeout

User Defined Monitored Parameters – allow unsafe parameters

This setting sounds scarier than it is. It allows us to send certain characters which Zabbix considers unsafe – such as slashes and spaces – as parameters to remote commands that the Agent will execute.

```
258
259 ####### USER-DEFINED MONITORED PARAMETERS #######
260
261 ### Option: UnsafeUserParameters
262 #   Allow all characters to be passed in arguments to user-defined parameters.
263 #   The following characters are not allowed:
264 #   \ ' " ` * ? [ ] { } ~ $ ! & ; ( ) < > | # @
265 #   Additionally, newline characters are not allowed.
266 #   0 - do not allow
267 #   1 - allow
268 #
269 # Mandatory: no
270 # Range: 0-1
271 # Default:
272 UnsafeUserParameters=1
273
```

Figure 35. Allow unsafe parameters

User Defined Monitored Parameters – UserParameter

This configuration option will be discussed at length in the Zabbix Configuration white paper.

The “scripts” folder and the "fms_config.ps1" PowerShell script referenced in the screenshot are items that we deployed to the FileMaker Server machine; they are not part of the default Zabbix agent installation.

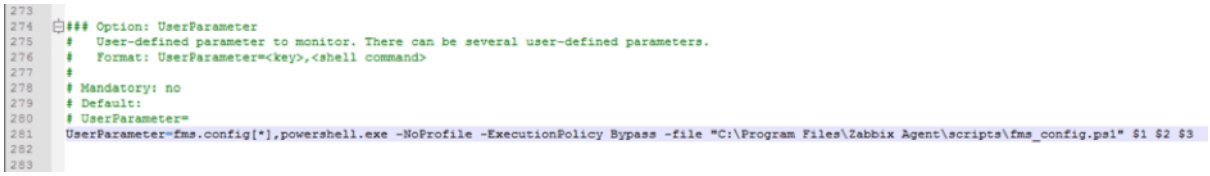


Figure 36. Set the UserParameter

The UserParameter configuration is largely the same on macOS and FileMaker Cloud, except that a Python script is called instead of a PowerShell script. The path to the scripts folder and the call syntax are also different between macOS and FileMaker Cloud.

FileMaker Cloud

```
### Option: UserParameter
#       User-defined parameter to monitor. There can be several user-defined parameters.
#       Format: UserParameter=<key>,<shell command>
#       See 'zabbix_agentd' directory for examples.
#
# Mandatory: no
# Default:
# UserParameter=

UserParameter=fms.config[*],/etc/zabbix/scripts/fms_config.py $1 $2 $3
```

Figure 37. UserParameter configuration in FileMaker Cloud

macOS

```
### Option: UserParameter
#       User-defined parameter to monitor. There can be several user-defined parameters.
#       Format: UserParameter=<key>,<shell command>
#       See 'zabbix_agentd' directory for examples.
#
# Mandatory: no
# Default:
# UserParameter=

UserParameter=fms.config[*],python /usr/local/etc/zabbix/scripts/fms_config.py $1 $2 $3
```

Figure 38. UserParameter in macOS

The PowerShell and Python script files are available https://github.com/soliantconsulting/FileMaker-Server-Zabbix-Templates.

Restart Zabbix agent Service

Whenever you make changes to the configuration file, you will need to restart the agent for those changes to take effect.

On Windows, use the Windows Services Control Panel to restart the agent. On macOS, use these commands:

```
sudo launchctl stop com.zabbix.zabbix_agentd

sudo launchctl start com.zabbix.zabbix_agentd
```

And on CentOS (FileMaker Cloud) use this command:

```
sudo systemctl restart zabbix-agent
```

The next guide in the series will walk you through how to import the FileMaker Server templates into the Zabbix admin console and configure your first FileMaker Server to be monitored.

Monitoring Your FileMaker Server

Zabbix Configuration

By Wim Decorte, Senior Technical Solution Architect
and Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Content

This document is one in a series of guides that walk you through how to install, configure, and use Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix. By this point, you should already have Zabbix server installed on a Linux machine – either as an appliance or installed from scratch – as well as the Zabbix agent installed on one or more FileMaker Server machines.

Zabbix Templates

Zabbix monitors servers and other devices (called hosts) by collecting data (called items) from those hosts. You can tag items with applications which are words or phrases you specify and are used to group and organize the items. Triggers are then used to evaluate the collected item data, and actions are set up to run in response to the triggers. The types of actions that can be taken include sending notifications and issuing remote commands (which are to be performed on the monitored hosts). In addition to automated actions, you can view the collected data in graphs, screens (which can be configured to show multiple graphs and other elements), and dashboards.

Some (although not all) of these configuration entities can be exported as an XML template and shared across different Zabbix installations. Both FileMaker Inc. and Soliant Consulting have released a set of templates that can greatly speed up the time it takes to set up your Zabbix server and monitor your FileMaker Server(s).

- FileMaker Inc. templates: https://community.filemaker.com/en/s/article/Using-Zabbix-for-Monitoring-FileMaker-Server
- Soliant Consulting templates: https://github.com/soliantconsulting/FileMaker-Server-Zabbix-Templates

As you download the templates, you will note that even though there are multiple templates, they are all contained within a single XML file.

Here is a partial list of the configuration entities that have and have not been included in the Soliant Consulting template:

Included:

- Host Groups – used for organizing hosts and templates

- Applications
- Items
- Triggers
- Graphs
- Host Screens – used to display data from a single host

Not included:

- Actions (because they depend greatly on your priorities and users/email environment)
- Global Screens – used to display data from multiple hosts
- Dashboards

Importing Templates

To import a template, log into the Zabbix frontend and navigate to **Configuration** > **Templates**.

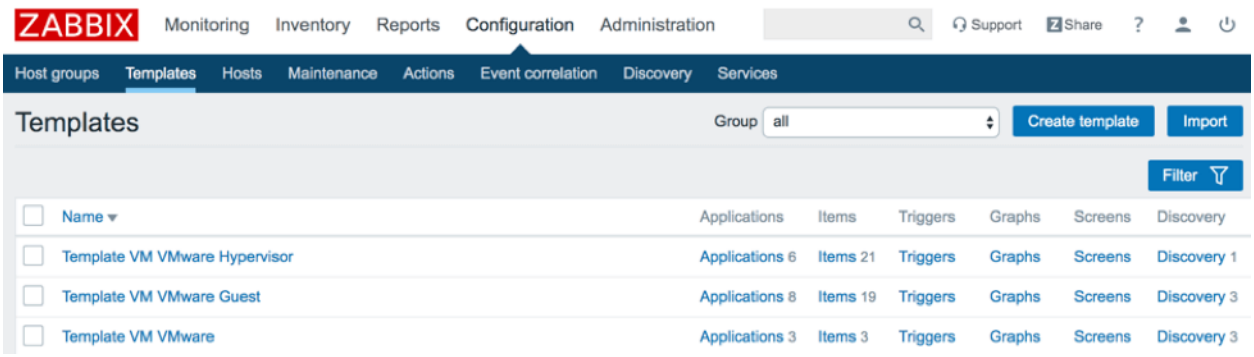


Figure 1. Use the Zabbix frontend to import templates

Click **Import** to go to the import screen, select the template XML file, leave the default import rules, and click **Import**.

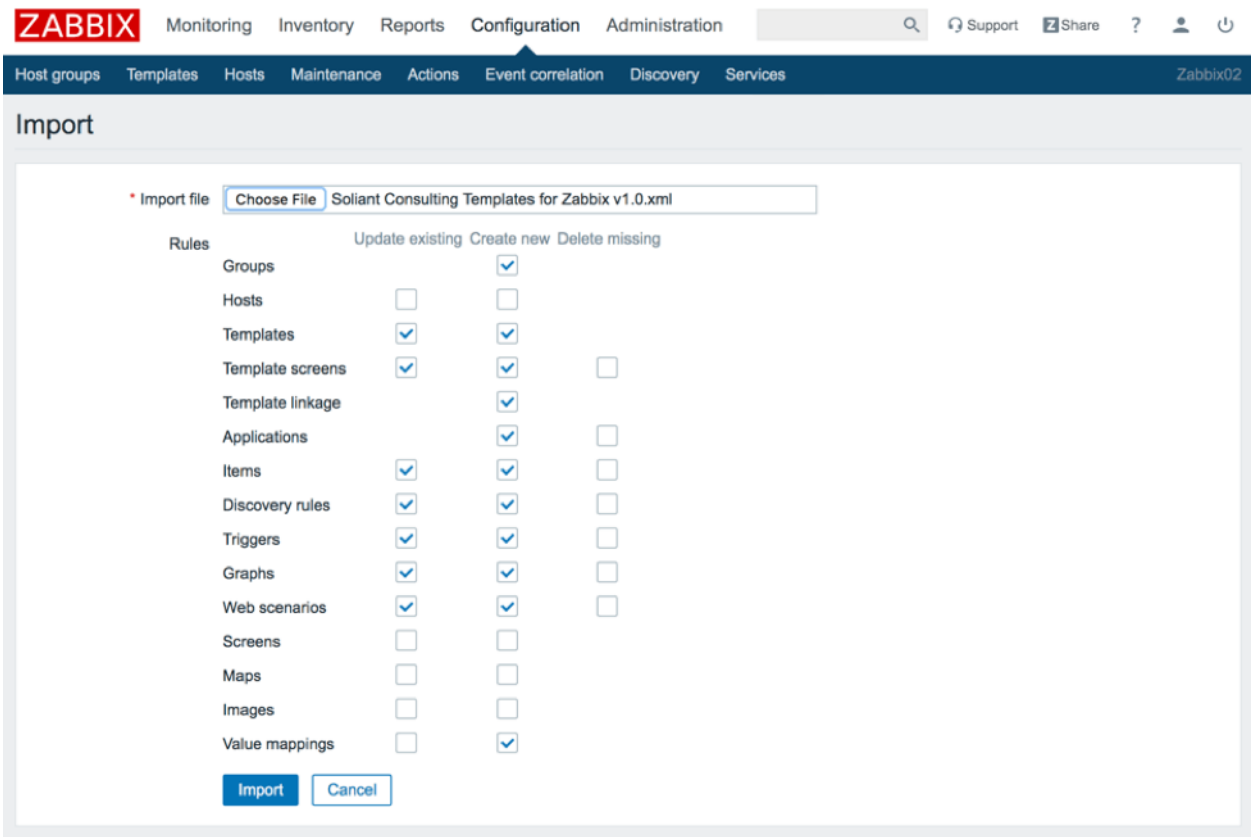


Figure 2. Import the template XML file

Navigate once again to **Configuration** > **Templates** and select **Templates/Soliant** as the group to only display the templates that were imported and to exclude the predefined templates that come by default with a Zabbix installation.

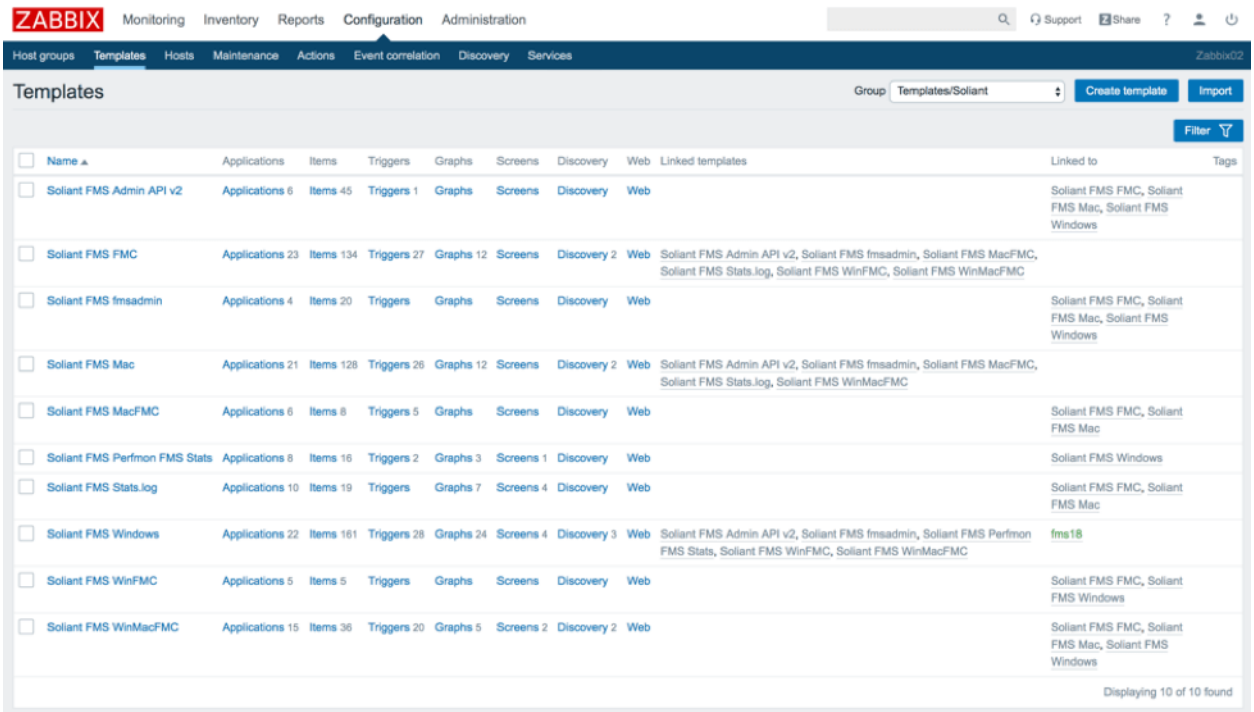


Figure 3. Template list shows only the templates that were imported

Template Organizational Scheme

To strike a balance between complexity and flexibility, and to minimize the amount of customization required after importing, we've organized the Soliant templates in these two tiers:

- Top-level templates – Intended to be linked to hosts.
- Component templates – Nested inside one or more top-level templates.

There are three top-level templates – one for each FileMaker Server platform: Windows, macOS, and FileMaker Cloud. These are the only ones that you would select when adding a template to your FileMaker Server host.

The component templates exist to account for variation in cross-platform compatibility. For example, the **perf_counter** item collects data from the Windows Performance Monitor (Perfmon) tool, so this item will only work with a Windows host.

Using component templates also facilitates configuring Zabbix to monitor different FileMaker Server versions and configurations. For example, as of the writing of this white paper, the Admin API v2 is only supported with FileMaker Server 18, so items that rely on the Admin API v2 are grouped in their own template.

The following schematic visualizes how the top-level and component templates fit together:

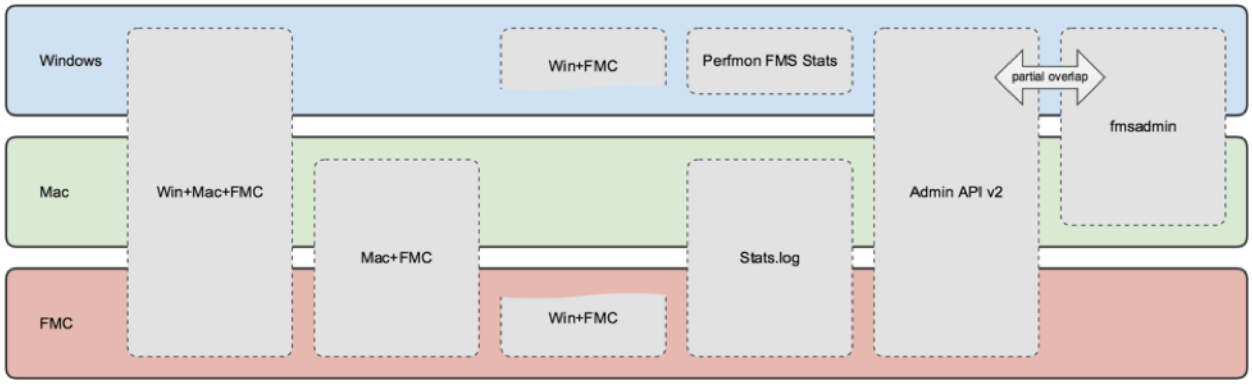


Figure 4. Schematic of top-level and component templates

To view just the top-level templates, select Templates/Soliant/Top-Level in the Group dropdown. Similarly, you can select Templates/Soliant/Component to view just those templates.

The names of the three top-level templates make their intention self-explanatory: when deciding which one to link to the host you wish to monitor, choose the one that corresponds to that host's operating system:

The component templates deserve a bit more explanation:

- **Soliant FMS WinMacFMC** – Includes items that work on all three FileMaker Server platforms.
- **Soliant FMS WinFMC** – Includes items that work on Windows and FileMaker Cloud but not on macOS.
- **Soliant FMS MacFMC** – Includes items that work on macOS and FileMaker Cloud but not on Windows.
- **Soliant FMS Admin API v2** – Includes FileMaker Server configuration data collected from the Admin API v2.
- **Soliant FMS fmsadmin** – Includes FileMaker Server configuration data collected from the fmsadmin CLI utility. The fmsadmin CLI tool has diminished capabilities when running on FileMaker Cloud, so for this reason, the fmsadmin component template is only included with the Windows and macOS top-level templates.
- **Soliant FMS Perfmon FMS Stats** – Includes Windows Performance Monitor (Perfmon) counters that have counterparts in the FileMaker Server Stats.log file.

- **Soliant FMS Stats.log** – Includes items collected from the Stats.log file.

Each top-level template has been linked to all eligible component templates so that it inherits their configuration entities (items, triggers, etc.). For example, the **Mac** template has been linked to the **WinMacFMC** and **MacFMC** templates (among others) but not to **WinFMC**, since the items in that template will not work on macOS.

The **Admin API v2** and **fmsadmin** templates both collect FileMaker Server configuration data but through different mechanisms. Some of the items overlap, and others don't. Review the items available in each template and decide which one you'd like to use. Note that to use the Admin API v2, your host must be FileMaker Server 18 with a custom SSL certificate. Zabbix server must also be able to reach the host via port 443. You can also choose to keep both sets of items enabled, in which case, you will end up collecting some redundant data. (You are not restricted to enabling/disabling items at the template level; you can also disable individual items within a template. We'll cover this in more detail in a following section.)

Similarly, both of the **Perfmon FMS Stats** and **Stats.log** templates collect FileMaker Server statistics but through different mechanisms. The data collected with either mechanism is essentially identical. Our Windows top-level template makes use of the Perfmon FMS Stats component template. However, as Perfmon does not work on macOS and Linux, the other two top-level templates use the Stats.log component template.

Adding a host

To add a host, navigate to **Configuration** > **Hosts**.

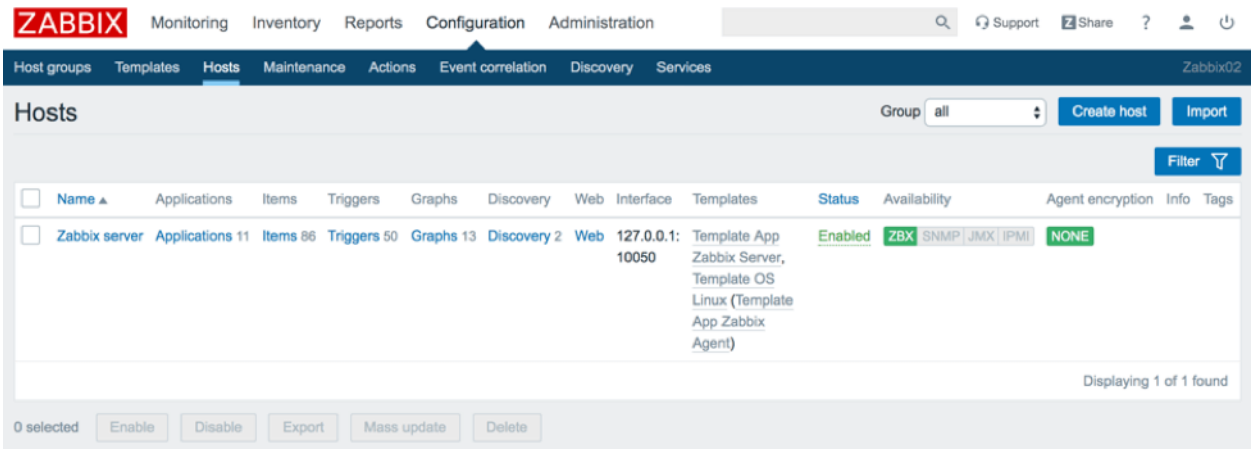


Figure 5. Adding a host

Click **Create host** and take the following steps on the **Host** tab:

- Enter the hostname (as set in the agent's config file) and optionally a visible name (a 'friendly' name).
- Select **FMS** as the group. (The FMS group will have been created as part of importing the Soliant Consulting template. You can select a different group if you would like.)
- Specify either a DNS name or an IP address or both and select the one that you would like Zabbix to use as the default means of communicating with the host.

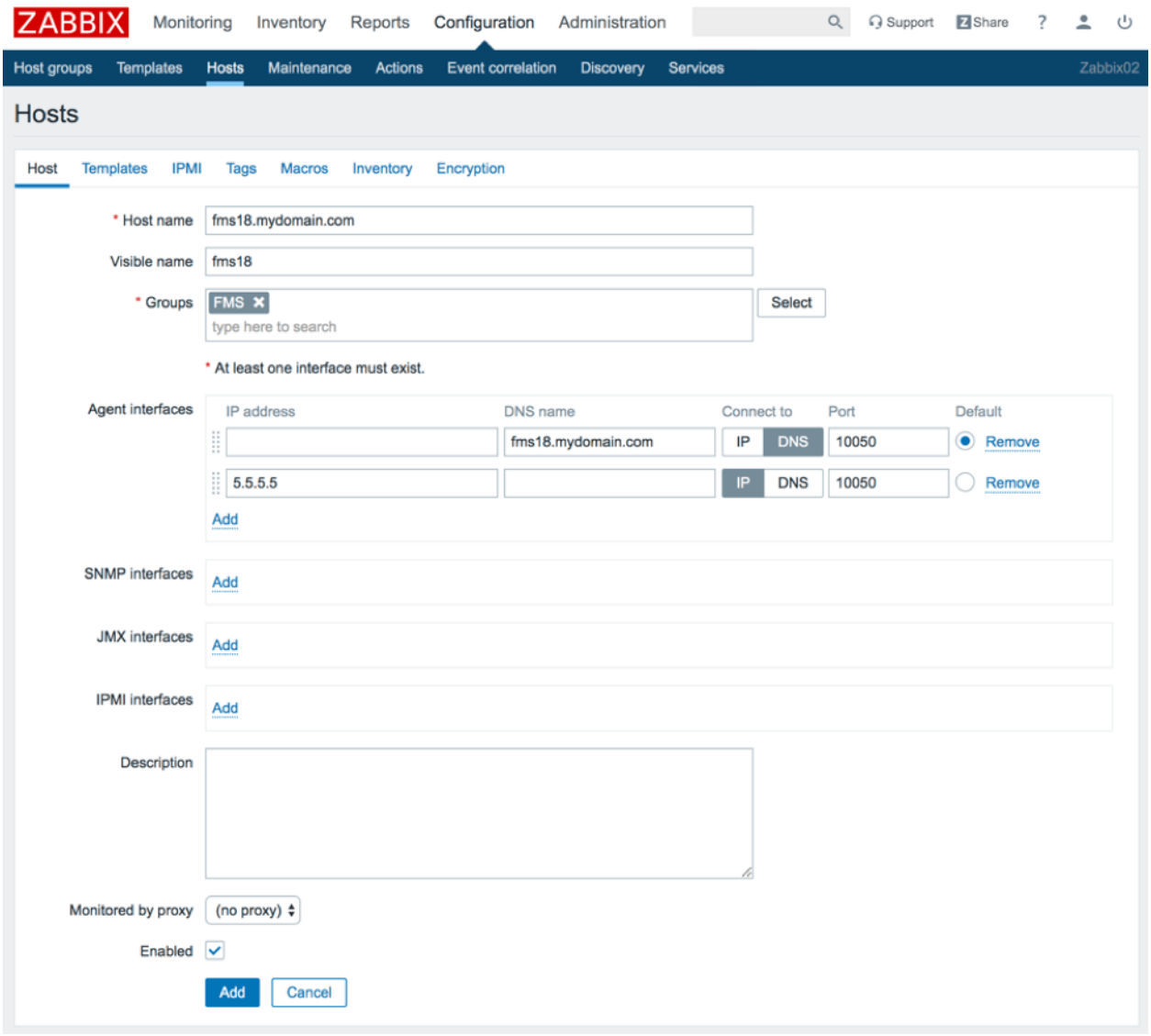


Figure 6. Specify the host properties

On the **Templates** tab, select the top-level template you would like to use. We will use the Windows template in the examples shown in this document. Don't forget to click the blue **Add** link to add the selected template to this list of linked templates.

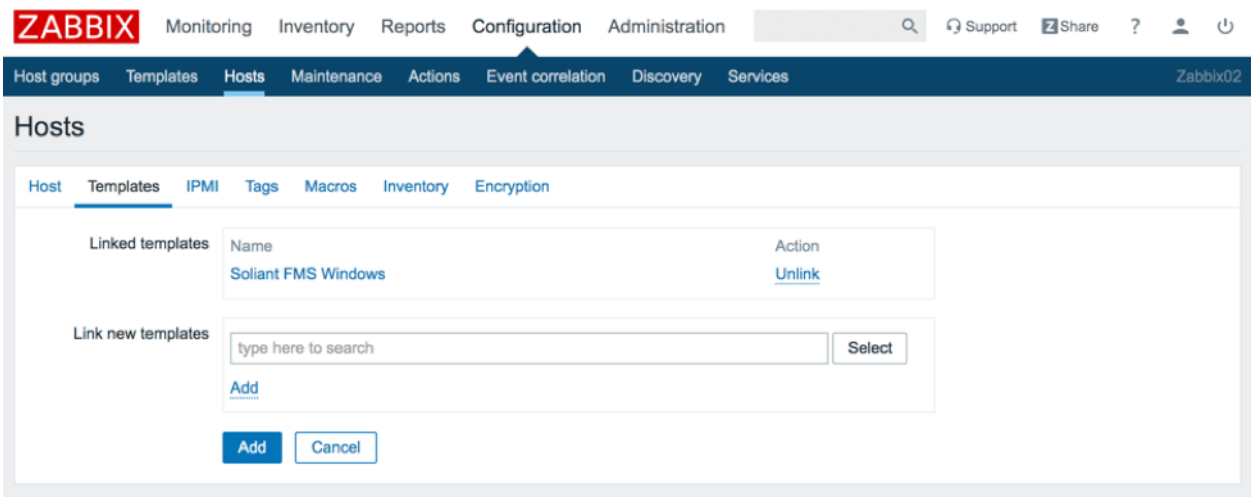


Figure 7. Select the top-level template to use

Switch to the **Macros** tab and then to the **Inherited and host macros** sub-tab. Macros are variables which make it possible to specify values in items, triggers, and other places that would otherwise have to be hardcoded. The top-level template selected during the previous step will have several macros already defined.

Click **Change** next to each macro whose value needs to change to accurately reflect the environment of the host that is being added.

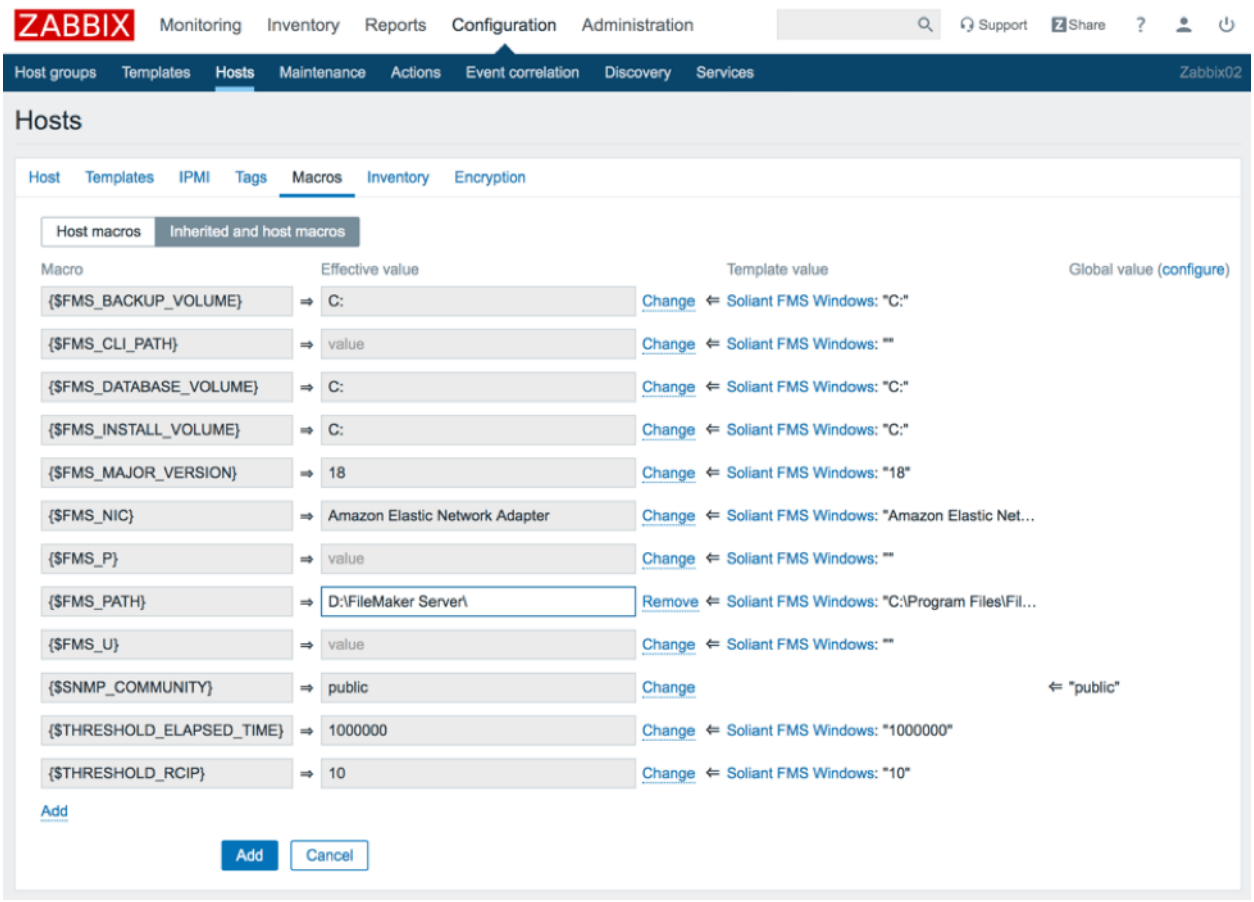


Figure 8. Change the macro's effective value

{FMS_U} and {FMS_P} are the credentials you would use to log into the FileMaker Server admin console.

Click the blue **Add** button to add the host.

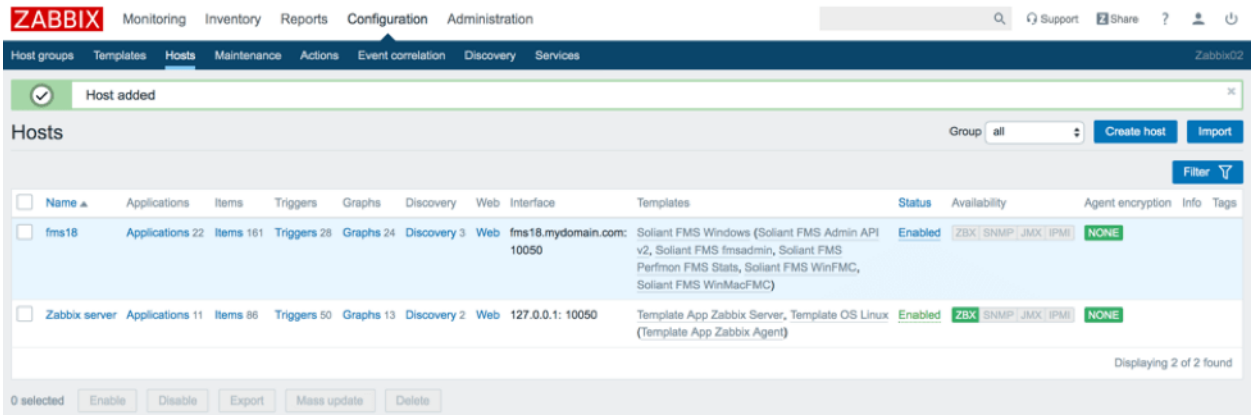


Figure 9. List displays host that was added

Items

Now that you have added a host and linked it to a template, you will want to finetune which items are enabled and how they are configured for data collection, the severity classification for certain triggers, and triggers important to your deployment. For instance: you may not care that the Data API is not running, or you may want a notification when the user load exceeds 20 users.

We will provide a few examples of how to do this, but there may be quite a few other customizations which could be appropriate for your situation. Our advice is to review the list of all configuration entities (applications, items, triggers, graphs, etc.) attached to the host and decide which ones require additional customization.

Item Status – Disabling an Individual Item

One of the items collects data from the Web Publishing Engine (WPE) log. If your FileMaker Server installation does not have WPE enabled, you could choose to disable this item. Doing so will lighten the load for the Zabbix server (by a tiny amount). If there was a trigger tied to this item, then disabling the item would also have the effect of disabling the trigger.

Navigate to **Configuration** > **Hosts** and click **Items** next to the host you want to configure. (An alternate way to arrive at the items screen is to click on the hostname first and then click on Items in the following screen.)

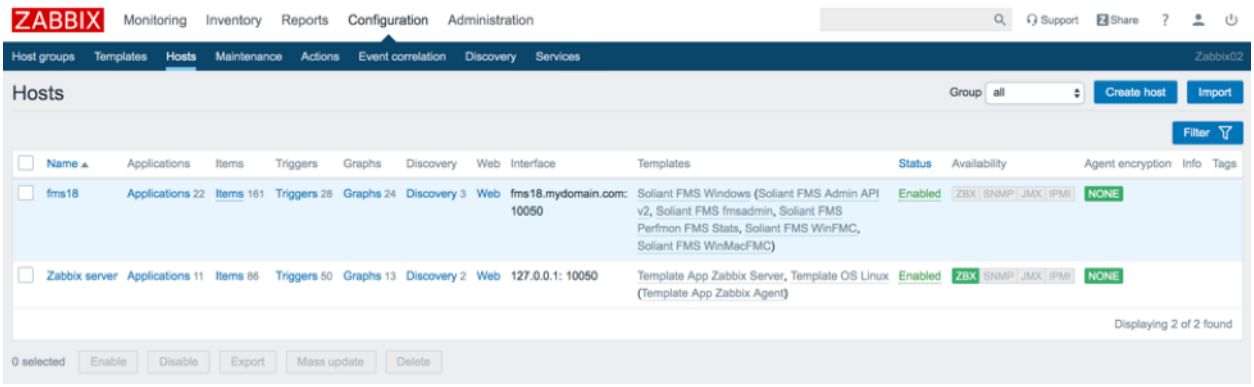


Figure 10. Click "Items" to the right of the host you want to configure

If the search/filter area is not shown, click **Filter** to display it. (Clicking Filter will toggle the display of that section.) Type "wpe" in the **Name** field and click **Apply**.

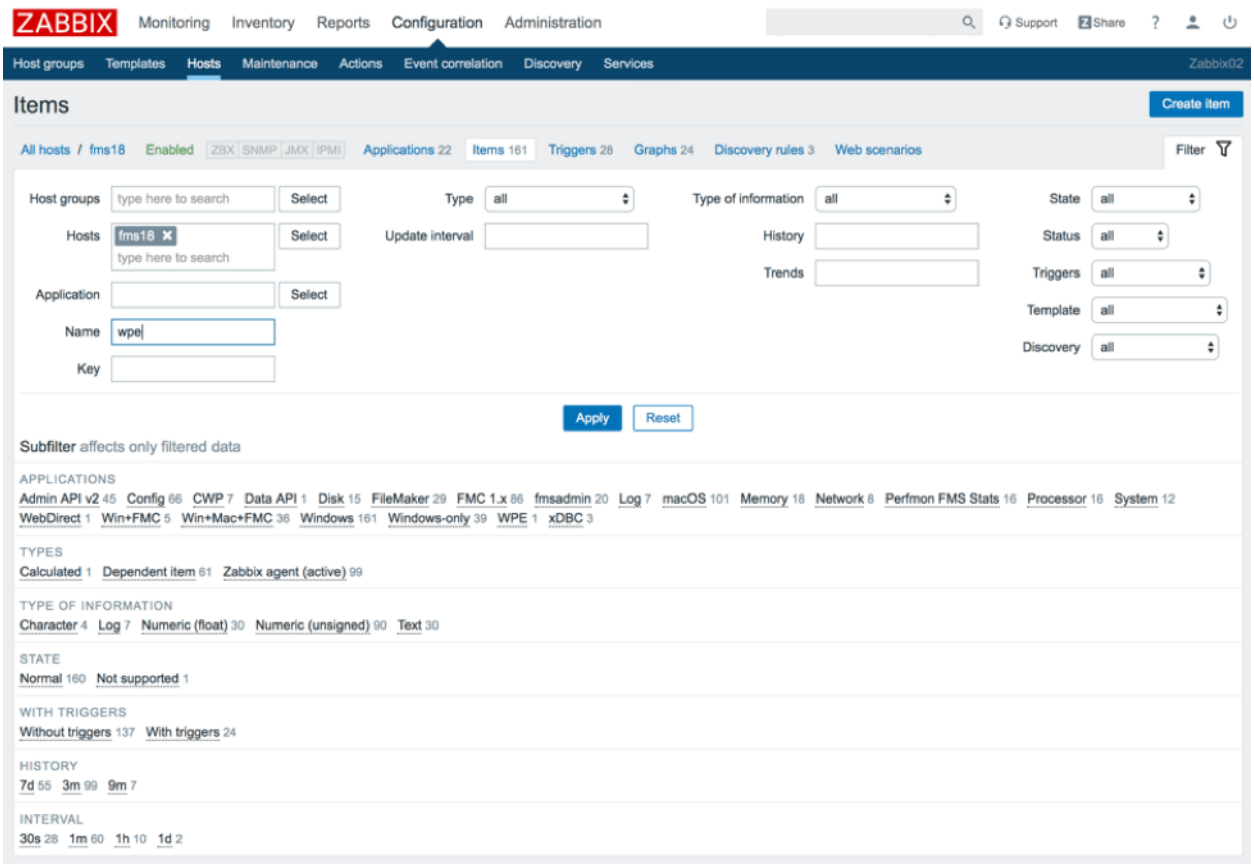


Figure 11. Filter to find a specific item

This search should yield a single item, named **Log - wpe**. Click the green **Enabled** link in the Status column to disable this item. (The filter section has been hidden in the screenshot below to conserve screen space.)

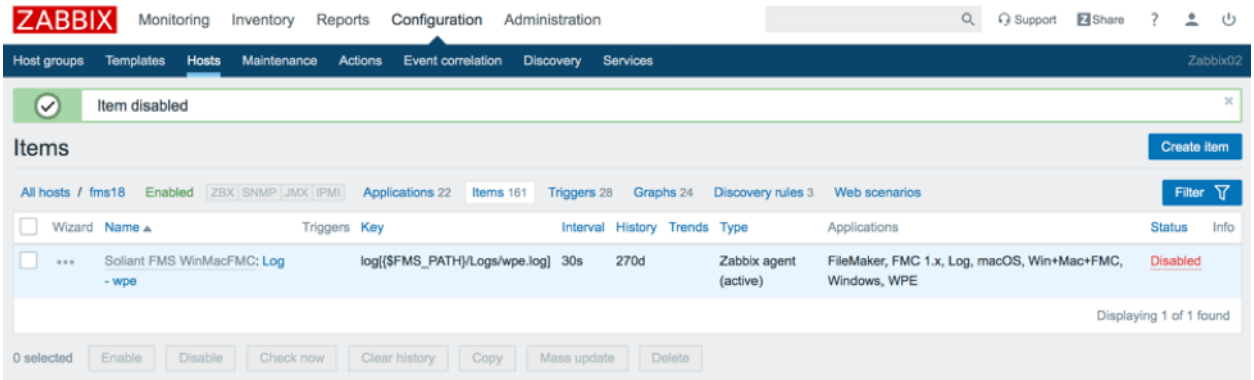


Figure 12. Item has been disabled

Using Mass Update to Disable a Group of Items

If your FileMaker Server installation does not support the Admin API v2, you may want to disable those items (and all associated triggers). To do so, navigate to **Configuration** > **Hosts** and click **Items** next to the host you want to modify. This time we will search for the items we want to disable by filtering by the **Admin API v2** application.

In Zabbix parlance, applications are just tags defined by you (or the template you are using). They provide a means of categorizing items, which makes it easier to filter for items. They are also used for grouping items when displaying a list of collected item values.

Click on the **Admin API v2** application in the filter area. The filter will be applied immediately. (Note that if you were to then click on another application filter, the effect would be an OR search, not an AND search; i.e., items that have been tagged with either one or the other application will be displayed.)

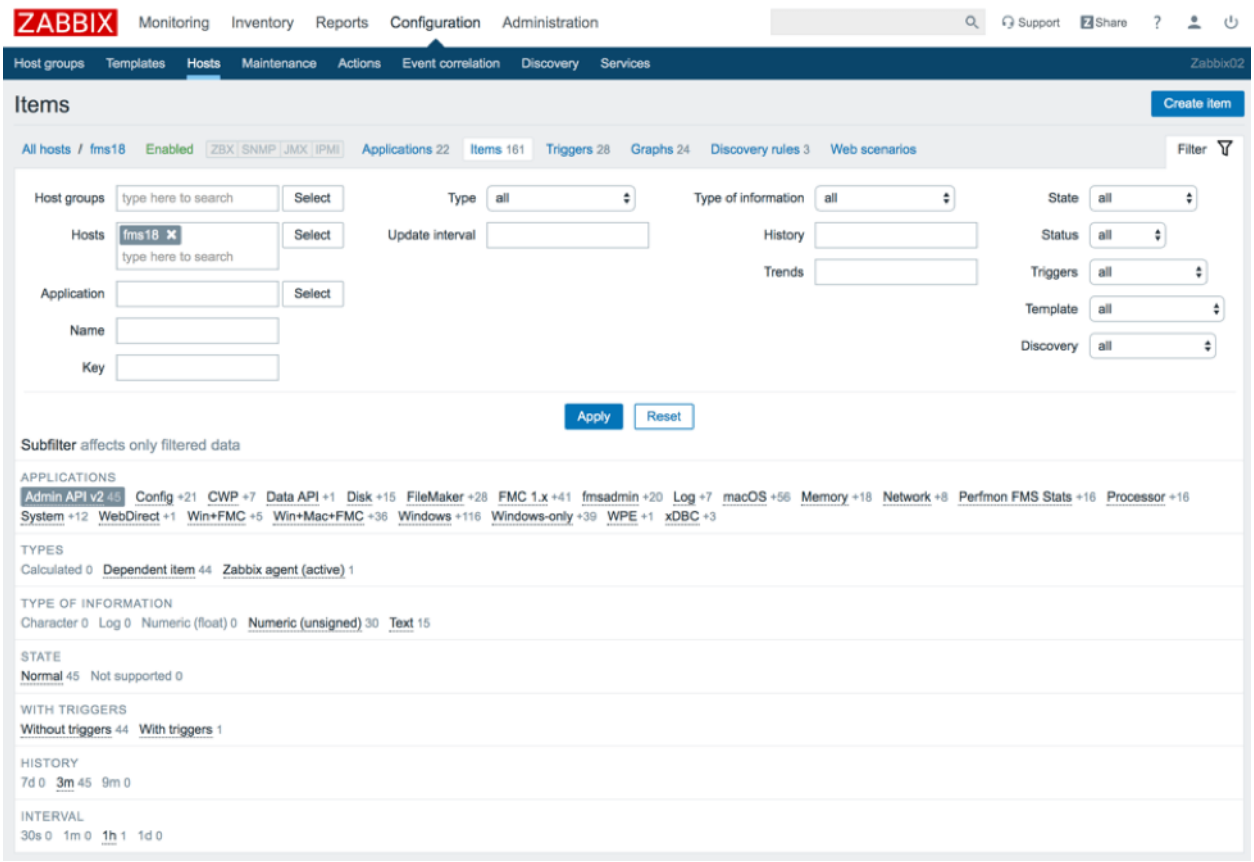


Figure 13. Items filtered by Admin API v2

Click the checkbox at the top of the list to select all items in the search results.

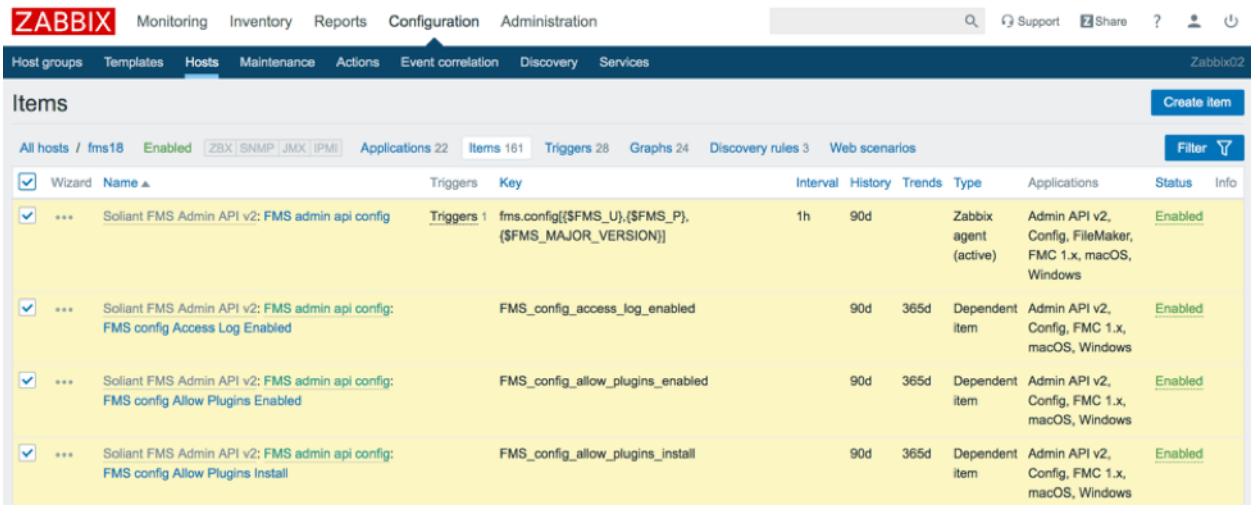


Figure 14. Checkbox marked to select all items in the search results

Then scroll to the bottom of the page and click **Mass update**.

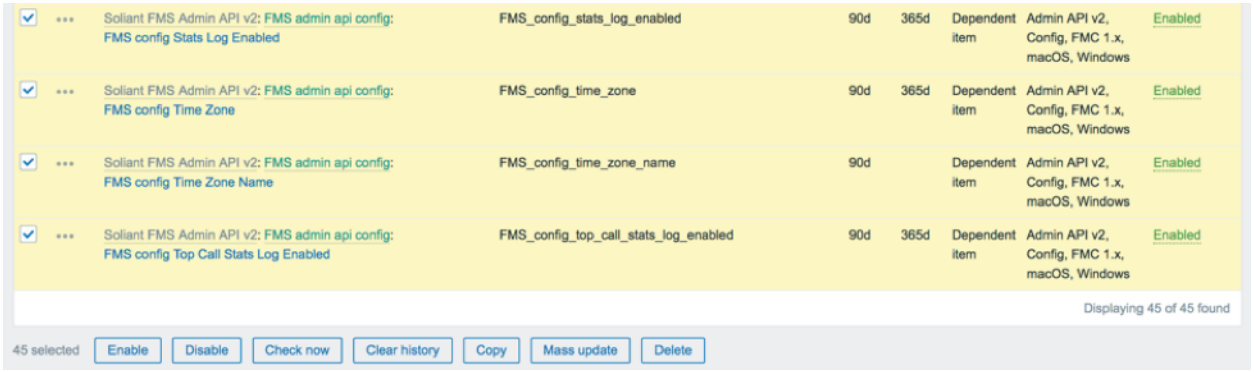


Figure 15. Click the "Mass update" button

Scroll down to the bottom of the mass update screen, click the **Status** checkbox, and select **Disabled** in the dropdown. Then click **Update**.

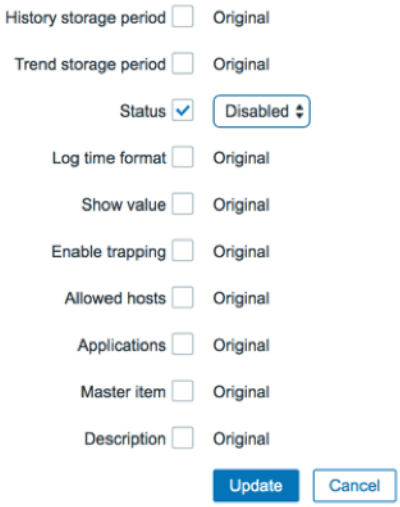


Figure 16. "Status" checkbox marked and "Disabled" selected from the dropdown

All of the items that were selected will now be disabled.

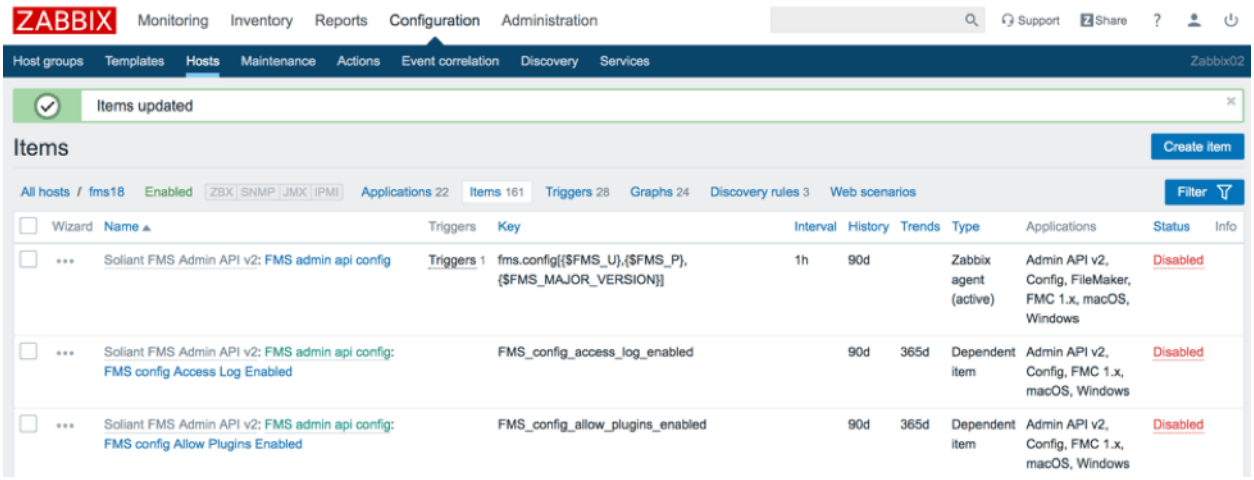


Figure 17. All items are disabled

Mass Update can be used in this way to make many different kinds of changes to a group of selected items. We will cover several of these in the next few sections:

- Item type
- Time intervals
- Applications

Item Type

Most of the items in our templates are configured to use the passive-mode **Zabbix agent** type. (See the **Zabbix Agent** white paper for a discussion of active and passive modes.)

One advantage of having the items run in passive mode is that you can collect item data on demand as opposed to having to wait for the collection interval to pass. To do so, select the items you want to collect, scroll down to the bottom of the page, and click **Check now**. Using passive mode does however require that port 10500 is open on the host. If you prefer not to do that, you can change all of the items to active mode.

Changing the type of an item works a little differently from changing the enabled/disabled status. You must do this at the template level instead of the host level, so you will need to navigate to **Configuration** > **Templates** and click on **Items** next to the template whose items you want to modify. If you try to change the item type from a host context, you will notice that the **Type** field is displayed with a gray background to indicate that it is not editable.

Once you arrive at the item detail screen (from a template context), filter by **Zabbix agent** type (don't skip this step!), select all items, click **Mass Update**, select the **Type** checkbox, select **Zabbix agent (active)** to indicate you want the agent to run in active mode, and click **Update**.

Time Intervals

There are three kinds of time intervals to be aware of:

- The **update interval** controls how frequently the data is collected.
- The **history storage period** controls how long the individual data points are kept.
- The **trend storage period** controls how long summarized data is kept.

After the history storage period elapses, numeric data is summarized on an hourly basis to conserve storage space. The individual data points are discarded, and the minimum, maximum, average, and the total number of values are kept for every hour. Note, this is done only for numeric data types. Non-numeric data (log entries, configuration settings, etc.) is discarded after the history storage period elapses.

You can adjust the time intervals for an individual item or, using Mass Update, for a group of items. As was the case with item statuses, time intervals can be changed both for a host or for a template.

Applications

You can add/edit/delete applications – both for an individual host or for a template – to change how the items are tagged to suit your needs. You can prepare the application tags you want to use in the **Configuration** > **Templates** > **Applications or Configuration** > **Hosts** > **Applications** area. It is also possible to create new applications on the fly when using Mass Update.

Let's take a closer look at this using an example. The template already has an application tag for "Processor." Some of the items tagged with this application measure the % processor time for both the overall processor and for the individual FileMaker Server processes. Suppose we want to be able to quickly filter on just these items for one of the hosts we are monitoring.

To do this, navigate to **Configuration** > **Hosts**, click **Items** next to the host where you want to make the change, type "% processor time" in the **Name** field, and click Apply. Select the items you want to modify and click **Mass Update**. Select the **Applications** checkbox and then select the **Add** tab. Type "% Processor Time" and then select the option in the dropdown to create the new application. Then click **Update**.

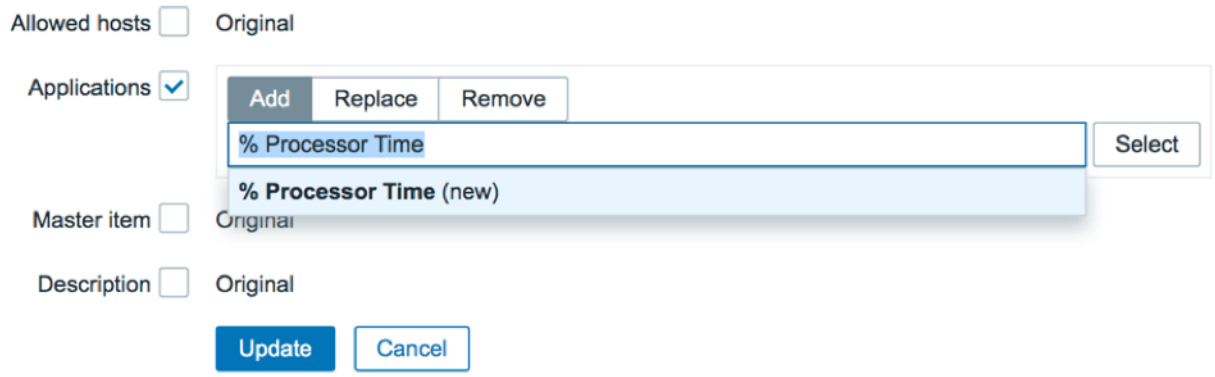


Figure 18. Creating a new application with % processor time

A new % Processor Time application will be created, and all of the items that were selected will be tagged with it.

User parameters

Each item is uniquely defined by an item key which describes the type of data that will be collected. Most items in the Soliant Consulting template make use of item keys that are predefined by Zabbix, but there is one item – belonging to the **Admin API v2** component template – which uses a custom item key that we have defined in the Zabbix agent configuration file.

Such an item is called a **user parameter**. The agent configuration file defines not only the item key for a user parameter but also the command that will be used to collect the item data.

In our template, we have named this item "FMS admin api config," and, for a Windows host, we define it in the agent configuration file as follows:

```
UserParameter=fms.config[*],powershell.exe -NoProfile -ExecutionPolicy Bypass -file "C:\Program Files\zabbix-agent\scripts\fms_config.ps1" $1 $2 $3
```

In order for this item to function properly, the fms_config.ps1 file has to exist in the specified path on the host. (The agent configuration file will follow the same format for macOS and FileMaker Cloud hosts although the file and path will be different.) Our agent installation guide provides instructions on how to set this up.

All of the other items belonging to the Admin API v2 component template rely on this item – their item types are configured as Dependent Items, which means they derive their value from another item. So, if you want to make use of the Admin API v2 items, this user parameter item will need to be set up properly on your host.

Triggers

Zabbix triggers evaluate data that has been collected using items. The evaluation is done using a **problem expression**. This is a logical expression that you can create so that if it evaluates as true, we know there is a problem. In short; triggers determine whether there is a problem or not.

For example, the following item key collects the percentage of disk space that is free on the C: drive.

```
vfs.fs.size[C:,pfree]
```

We can use the following problem expression to indicate that there is a problem when the most recently collected value is less than 5%. Note that the item key is referenced inside of the expression.

```
vfs.fs.size[C:,pfree].last(0)<5
```

To declare that the problem has been resolved, we can rely on the same problem expression, or we can use a separate **recovery expression**, or we can do neither, which means the problem would have to be closed manually.

Trigger Status

As was the case with items, triggers can be disabled – both individually and using Mass Update. Review the list of triggers provided in the template you are using and disable the ones that don't apply for your host. For instance, if you do not have the Data API enabled, disable the **FMS process not running - Data API - Windows** trigger.

Trigger Severity

Each trigger is assigned a severity, which is a designation that you make. The severity of a problem is a subjective assessment that can vary depending on the situation, so for this reason, our advice is to review the severities of the triggers defined in the template and change them as you deem appropriate.

Modifying Triggers

Let's take a look at how we would go about changing the severity of a trigger. The FileMaker Server event log records a series of messages depending on what events transpire on the server. The list of possible messages can be viewed here. Several of the messages are warnings about possible consistency check issues. For example:

634 – Warning – Database "%1" consistency check skipped by administrator, database opened. Use of this database could result in data corruption.

If you are getting notifications about event log errors, but no warnings, you would miss this message. However, consistency check warnings are indicators of possible serious issues. Our template has a trigger which will create a problem event if it notices this (or similar) message in the event log.

Suppose you are reviewing the triggers used by your host, you notice that the severity of this particular trigger has been set as **Average**, and you would like to change it to **High**.

To see the list of triggers used by a host, navigate to **Configuration** > **Hosts** and click **Triggers** next to the host you are interested in. Type "consistency" in the **Name** search field and click **Apply**, or simply look for "Possible consistency check issue" in the list of all triggers shown. (In Figure 18, the filter section has been toggled off, which you can do by clicking on the Filter tab.)

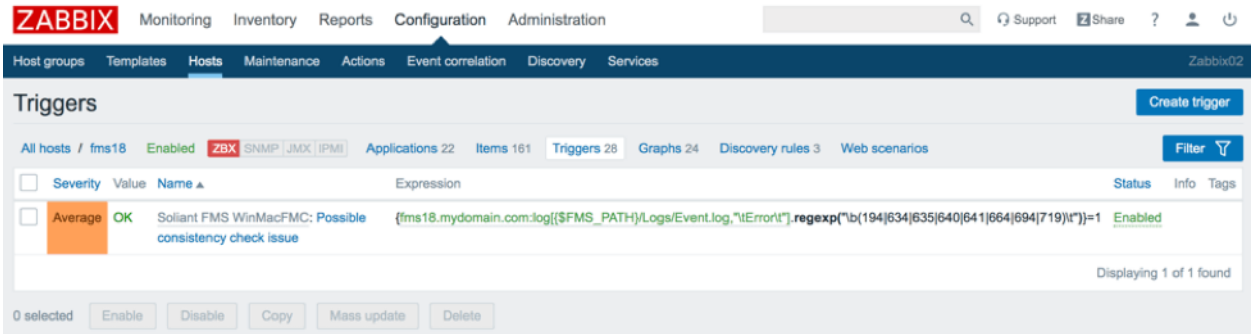


Figure 19. Triggers List with filter section toggled off

Click on the trigger name to view the details.

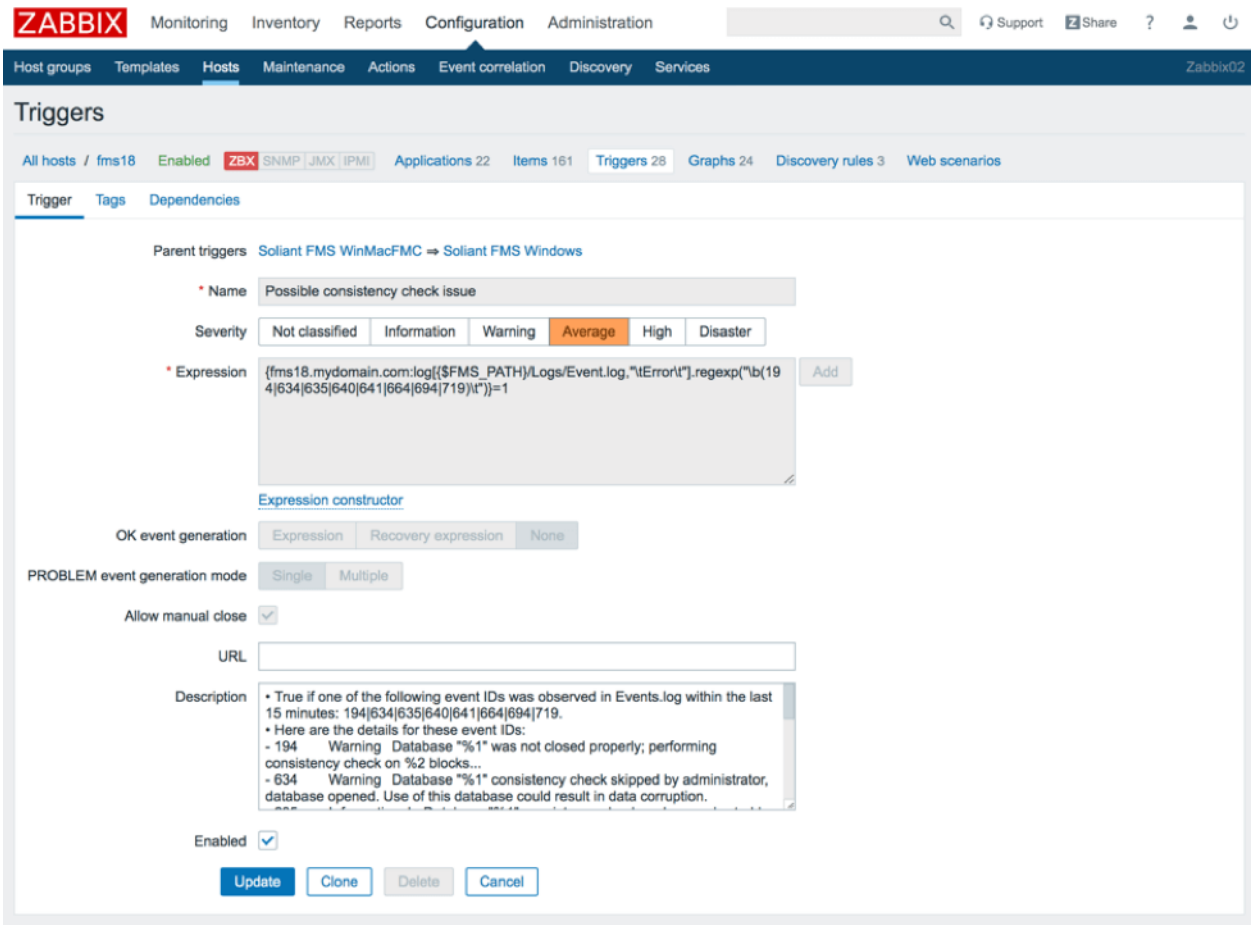


Figure 20. Trigger details

To change the severity, simply select a new value and click **Update**.

Note that some of the trigger attributes – for example, name and problem expression – are shown with a gray background, which means they are not editable. Because this trigger is inherited by the host through a template, these attributes can only be modified in the template. In this example, this trigger was originally defined in the **Soliant FMS WinMacFMC** template, which is inherited by the **Soliant FMS Windows** template.

Click the blue **Soliant FMS WinMacFMC** link at the top to view this trigger from the context of that template. All of the attributes will now be editable but do keep in mind that changing the values will affect the behavior for all hosts that use this template.

Users

It is a good idea to create a dedicated account for each user who will log into the Zabbix frontend.

Navigate to **Administration** > **Users**, click **Create user**, and fill in the user details. We'll use the provided **Zabbix administrators** group for our new user.

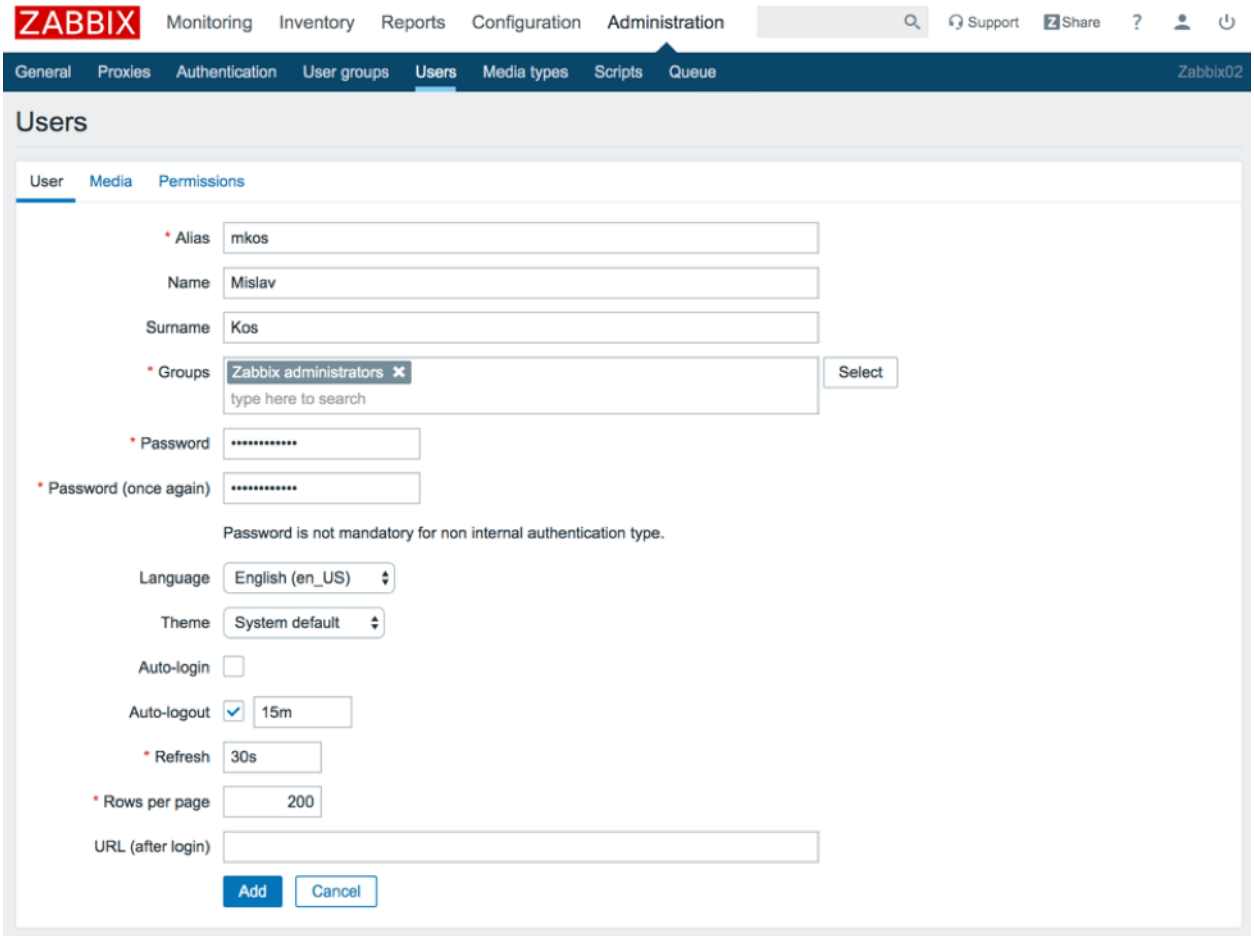


Figure 21. Create user and add to the Zabbix administrators group

Switch to the **Media** tab, click **Add** next to **Media**, and enter your email address.

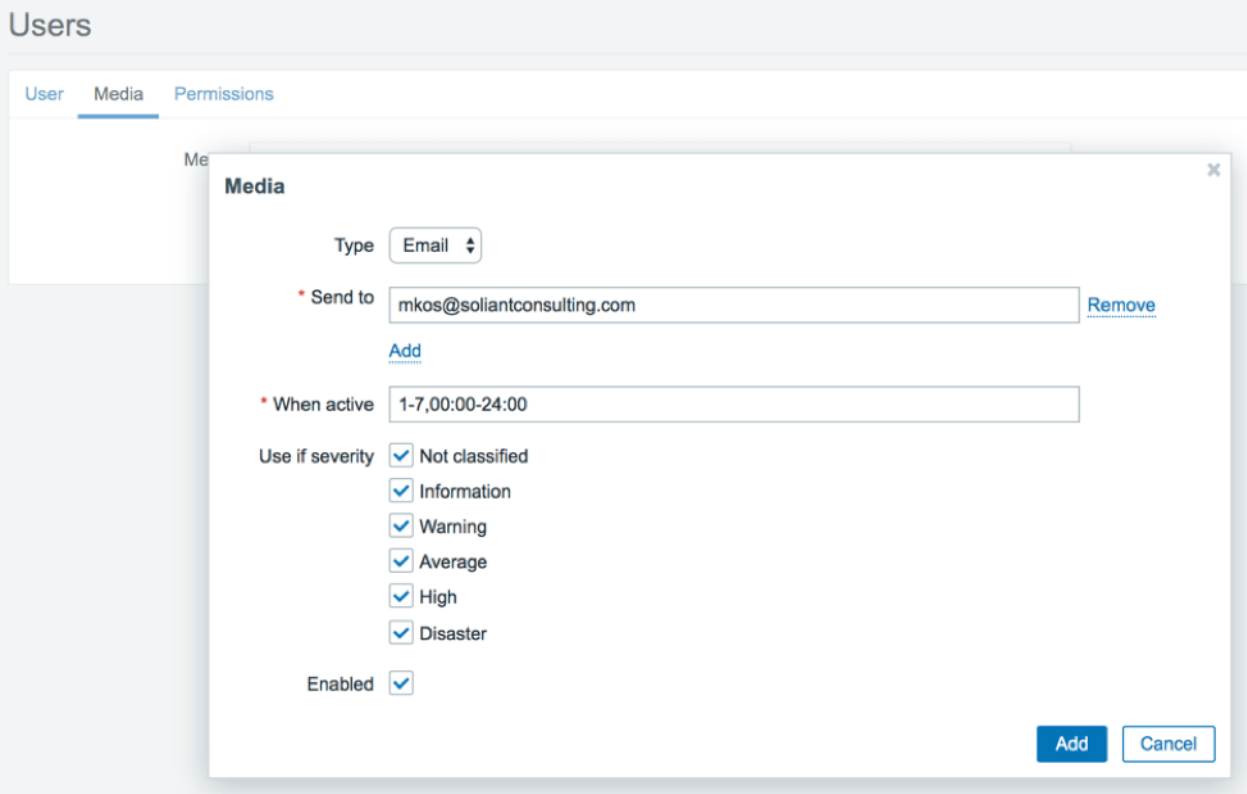


Figure 22. Enter email address in the Media popup

Click **Add**.

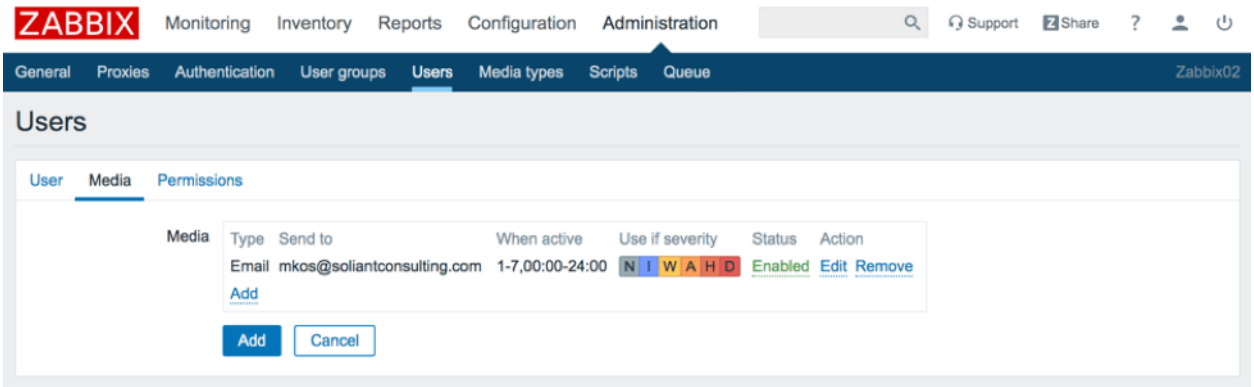


Figure 23. Email address added in field on the Media tab

Click **Add** again to add the user.

User Groups

Zabbix already provides a **Zabbix administrators** group which can be used for your accounts. But we will create a new user group here, both to illustrate how this is done, and because we will need this new group later to send out text alerts.

Navigate to **Administration** > **User groups**, click **Create user group**, and specify a **Group name**. The **Frontend access** field allows you to specify if the accounts in these groups will be internal or LDAP (Active Directory). In our case, we are disabling access to the frontend (a.k.a. the Zabbix web interface), because accounts belonging to this user group will be used solely for sending out text messages. The reason for this will be explained in a later section.

> *Note: If you decide to use LDAP for your user group, you will still have to create individual user accounts in Administration > Users for every user who needs access. This is different from what you may be used to with FileMaker where you can simply create one externally authenticated account for an entire Active Directory group which then delegates authentication for all users belonging to that group to Active Directory.*

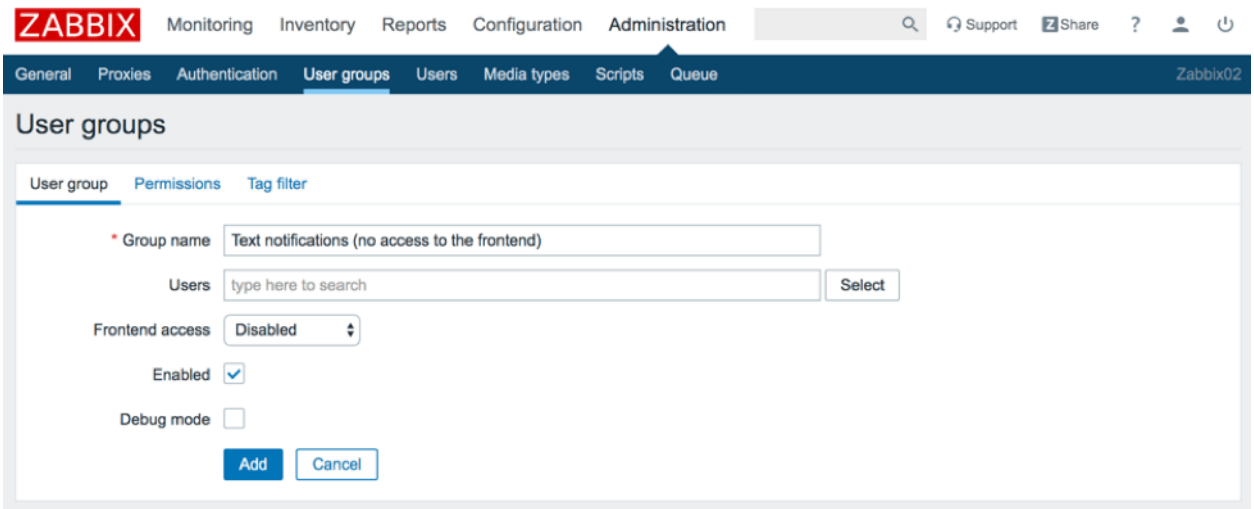


Figure 24. Creating a user group with Frontend access disabled

Switch to the **Permissions** tab, specify the **FMS** host group (which was added as part of importing the Soliant template), select to **Include subgroups**, and select **Read** permissions.

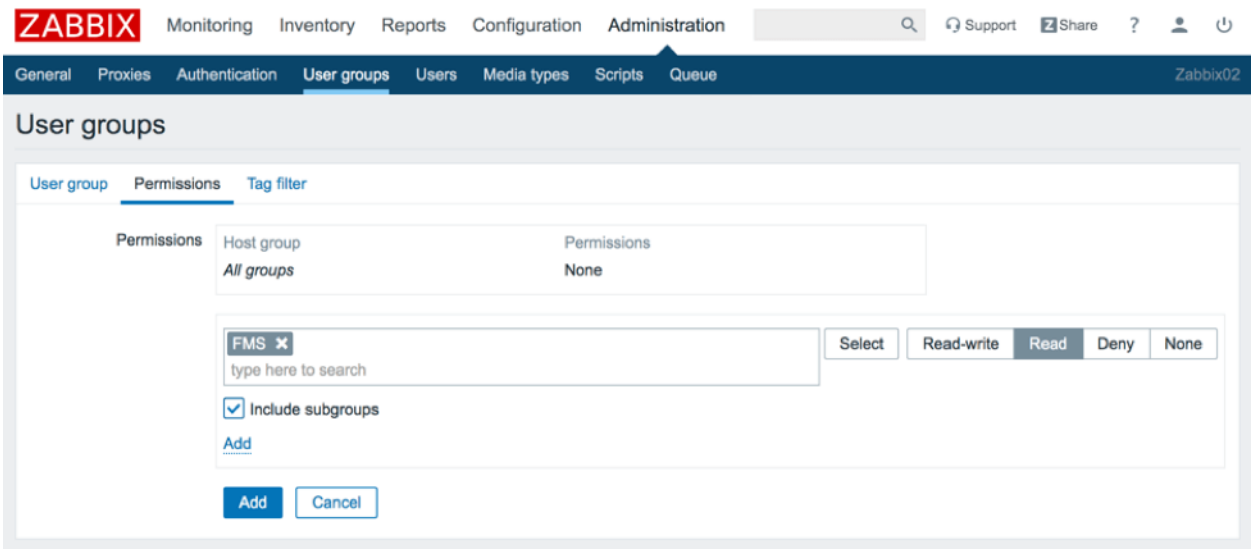


Figure 25. Setting User group permissions to include subgroups and read permissions

Click the blue **Add** link to add the host group. (Don't click the blue **Add** button yet at this point.)

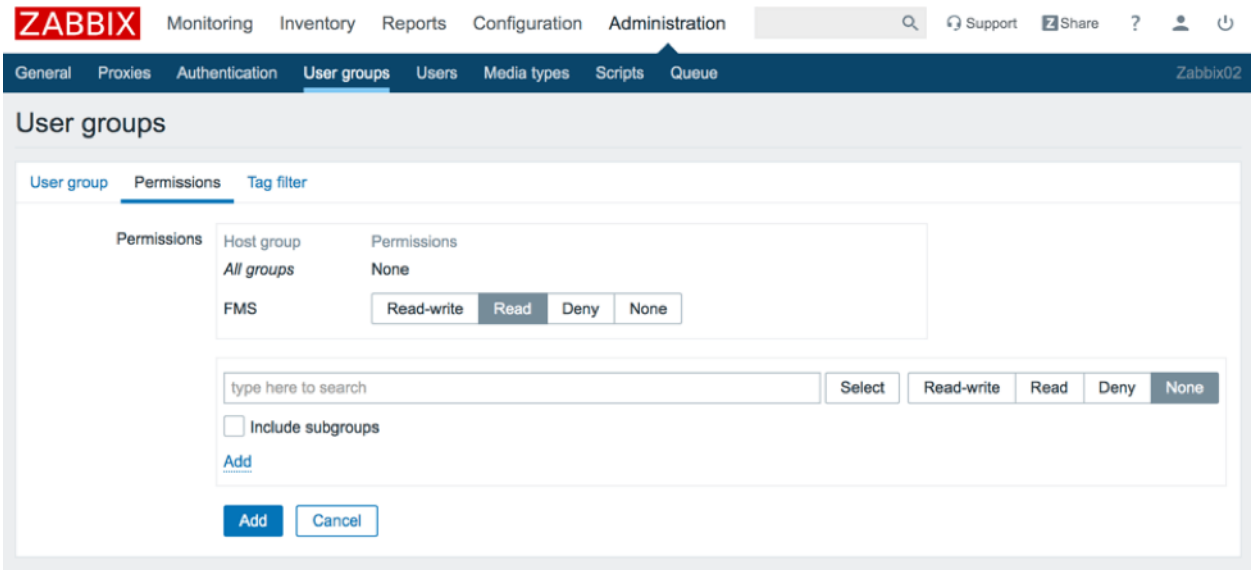


Figure 26. Host group added

Now click the blue **Add** button to add the new user group.

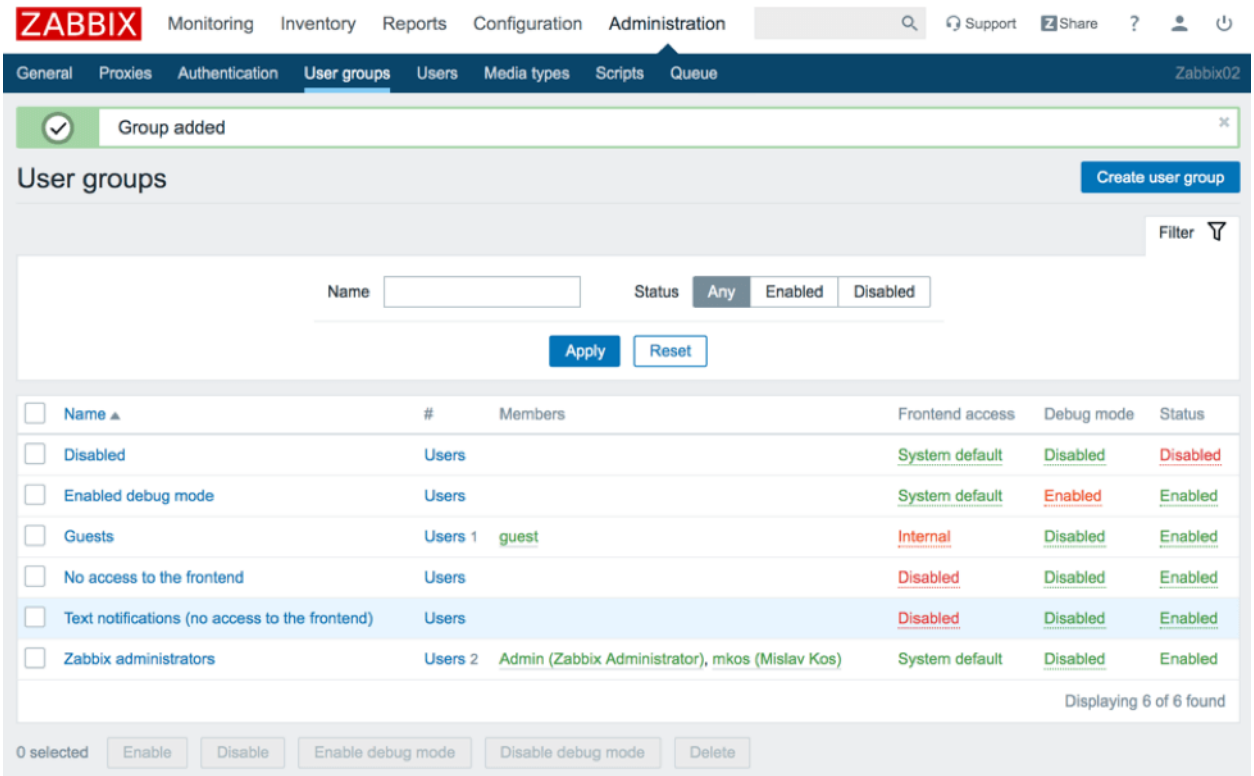


Figure 27. New user group is displayed in the User groups list

Actions

Having problems identified automatically is nice but not very useful if you have to log in to the Zabbix frontend in order to see that a problem occurred.

Creating actions solves this issue. There are two types of actions:

- Notifications – Sent via email, text, or even chat (Slack, RingCentral Glip, etc.)
- Remote commands – Executed on the host; e.g., restart FileMaker Script Engine

Unfortunately, actions cannot be shared using a template, so you will have to create them from scratch.

Email Notifications

To enable email notification messages, we first have to enter the SMTP email server information. Navigate to **Administration** > **Media types** and click on the **Email** media type. Enter the information for your SMTP server and click **Update**.

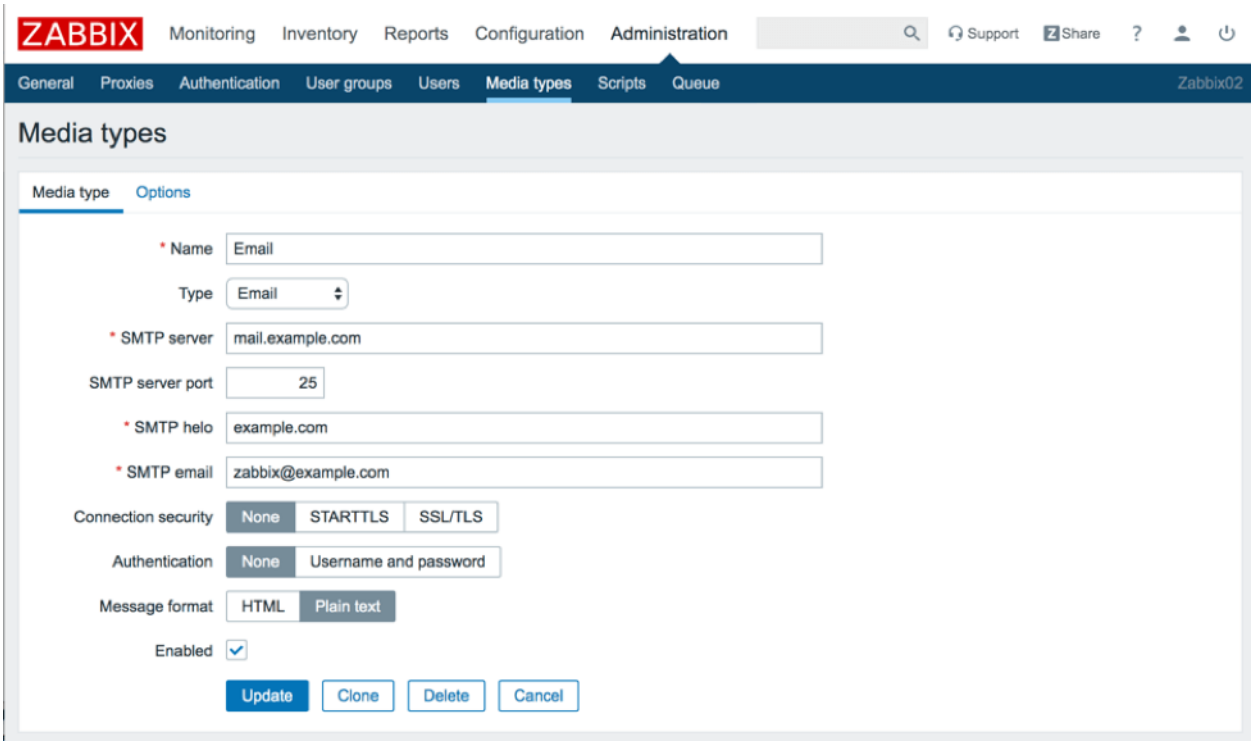


Figure 28. Add information for your SMTP server

We have already specified an email address for one of the users in the **Zabbix administrators** group in the Users section of this white paper, so now we are ready to create a new action to send an email (using the **Email** media type) to all users belonging to the **Zabbix administrators** user group. As an example, we will set up the action to send emails for triggers whose severity is average or higher.

Navigate to **Configuration** > **Actions** and click **Create** action. In the **Action** tab of the detail screen, give the action a name and specify a new condition where trigger severity is greater than or equal to average.

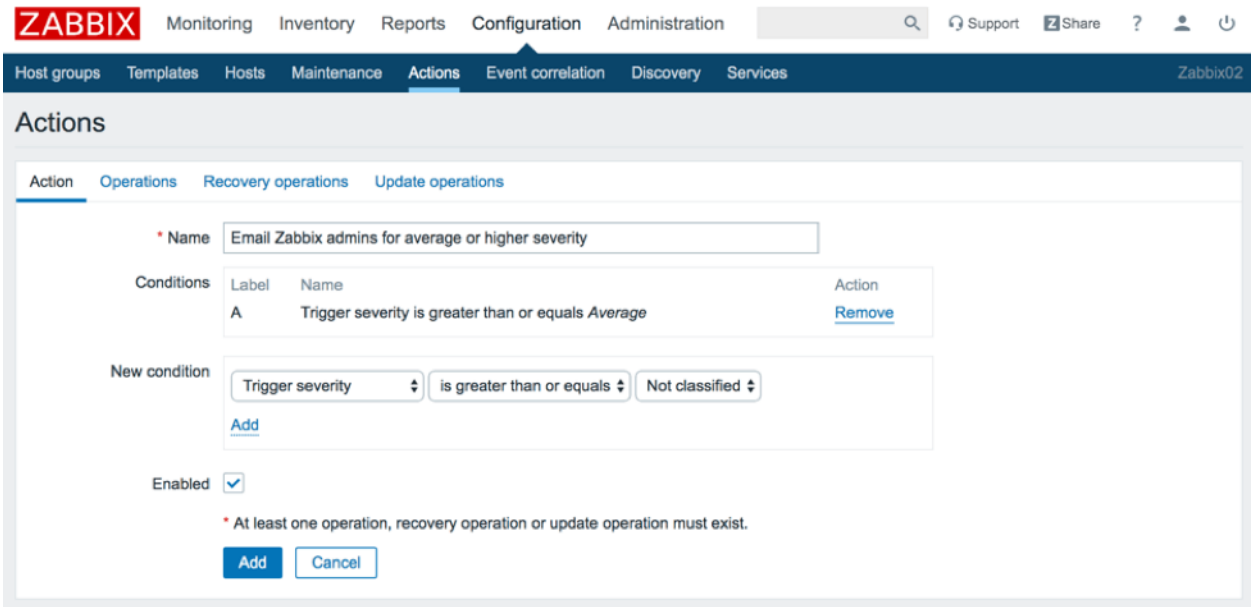


Figure 29. Add the action name and new condition

Click the blue **Add** link to add the new condition, but don't click the blue **Add** button yet. Instead, switch to the **Operations** tab and add the following to the bottom of the default message to provide additional helpful information in the emails that will be sent out:

Item value: {ITEM.VALUE}

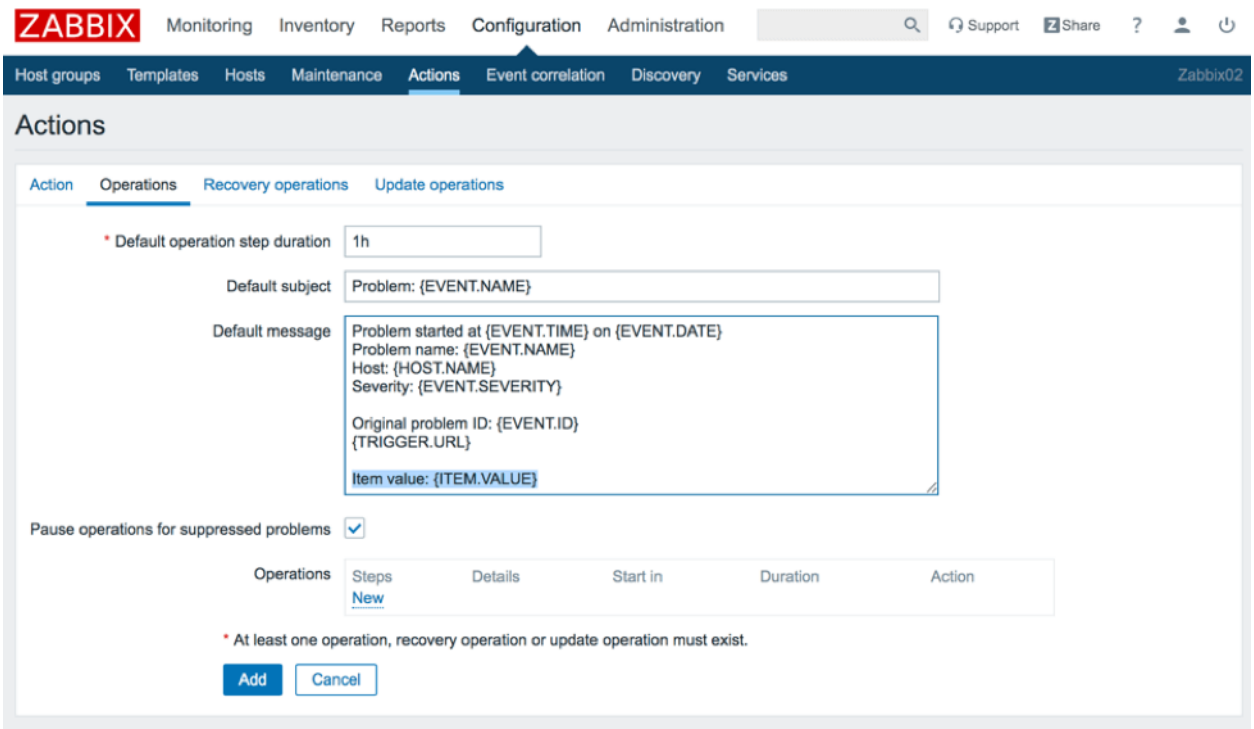


Figure 30. Item value added to the bottom of the default message

Click **New** to create a new operation step. Then click **Add** next to **Send to User groups** and select the **Zabbix administrators** user group. The **Operation type** is already set to **Send a message**, so we can leave that is. If we wanted to configure a remote command instead of sending a message, this is where we would specify that. Select the **Email** media type as the **Send only to** value.

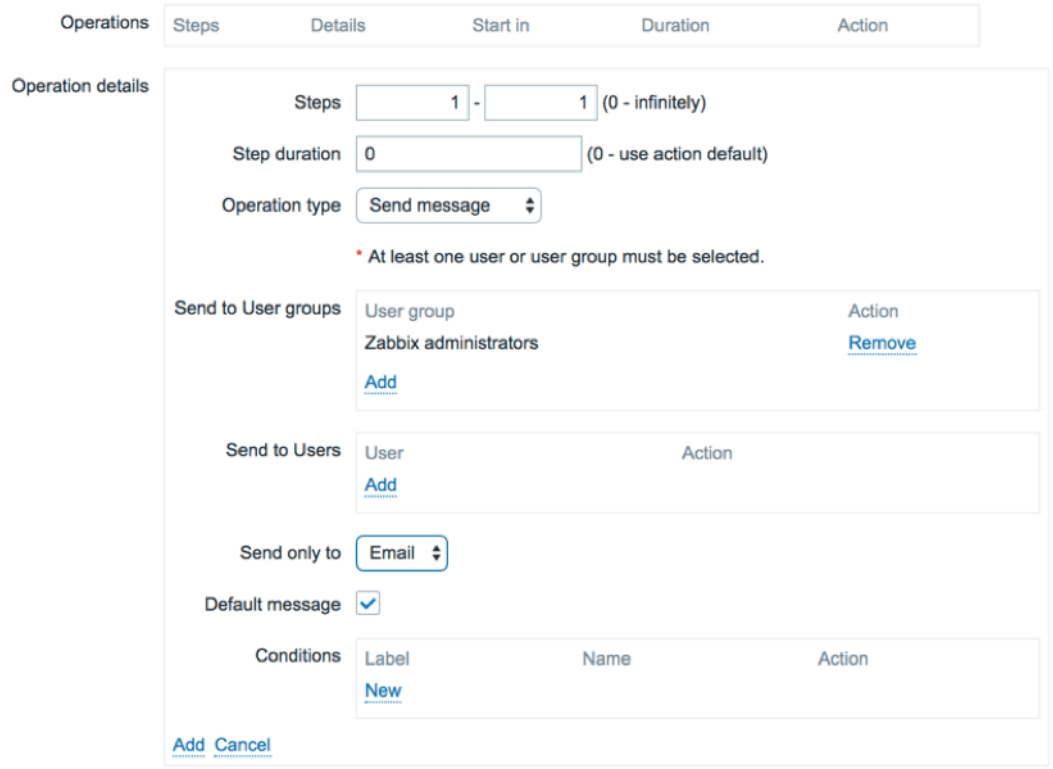


Figure 31. Add the new operation step with 'Send only to' set to Email

Click the blue **Add** link to add the new operation step.

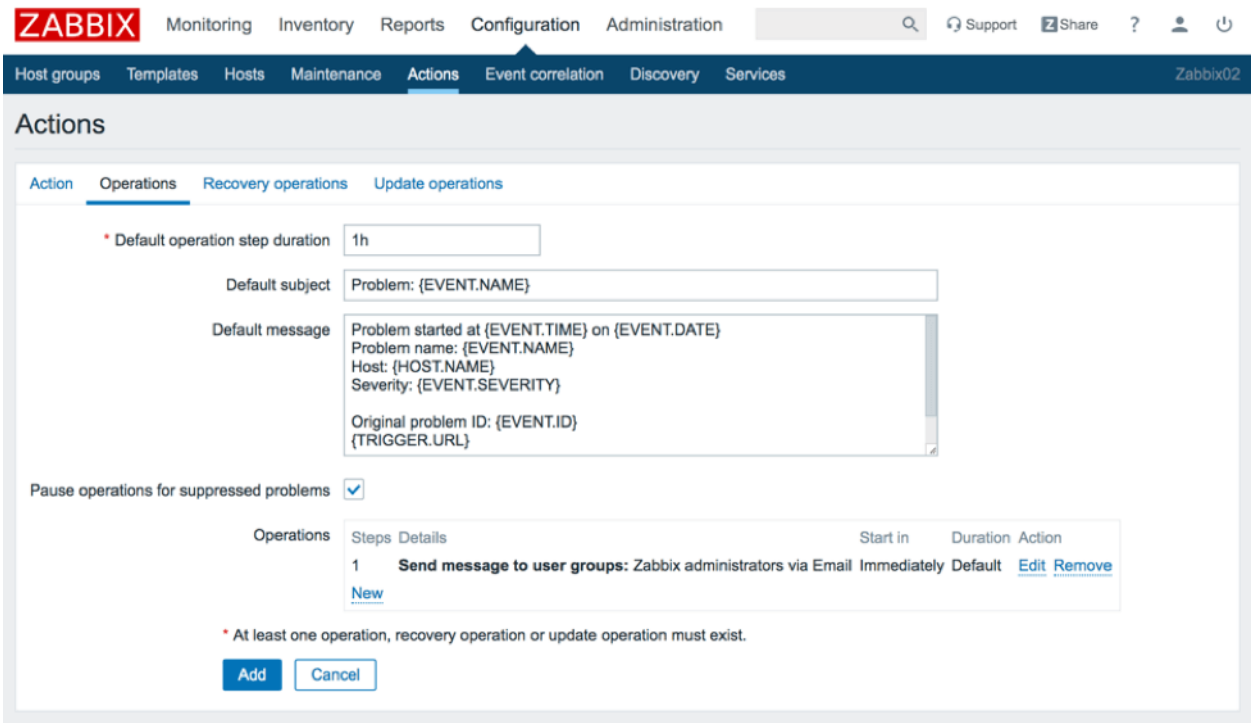


Figure 32. New operation step is shown in the 'Operations' field

The Operations tab defines the operations (i.e., actions) that will be taken when a problem is first *identified*. The **Recovery operations** and **Update operations** tabs define the actions that will be taken when a problem is *resolved* or *updated*. (We haven't covered problem updates yet, so we'll do so briefly here. Once a problem is created, it can be updated by navigating to **Monitoring** > **Problems** and working with the problem there. For example, an update can take the form of acknowledging the problem. Doing so lets others know that you are on the case.)

It's a good idea to receive email notifications not just for when a problem is identified but also for when it is resolved or updated. Repeat the steps documented above to create the same operation step in the recovery and update tabs. Finally, click **Add** to add the action.

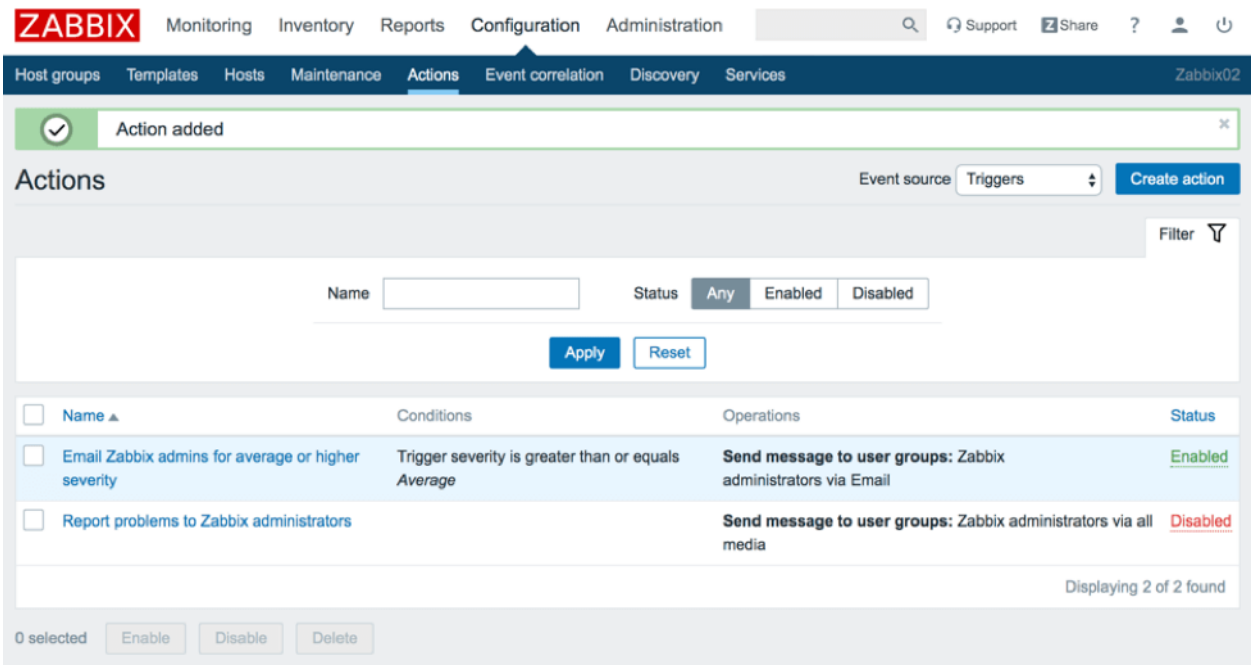


Figure 33. New action is added and displayed in the Actions list

Text Notifications

The predefined SMS media type can be used to send text message notifications, but this approach requires that a GSM modem is connected to the Zabbix server. Fortunately, there is an alternative, more practical way of sending text alerts.

We have already specified a **Text notifications (no access to the frontend)** user group in the **User Groups** section of this white paper. Now we'll need to create a new user account belonging to that group.

Navigate to **Administration** > **Users** and click **Create new user**. Enter in the basic user information and attach this user to the user group we created earlier.

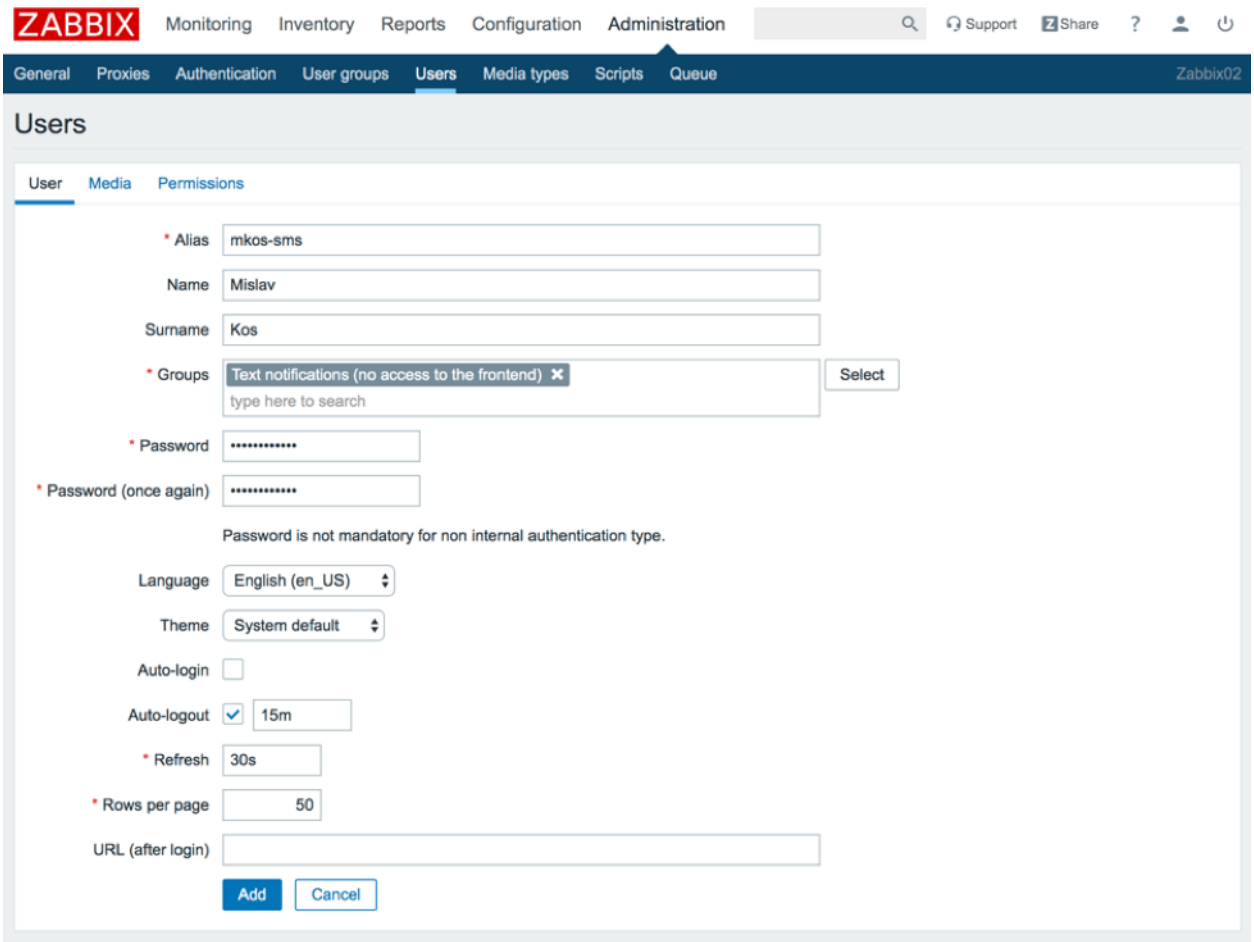


Figure 34. Enter information for new user

Switch to the **Media** tab and click the blue **Add** link next to Media. We'll keep the **Type** as **Email** even though we are setting up text alerts. Many telecoms provide an SMS gateway which can be used to send SMS to mobile phones via email. Construct the email address by using your phone number as the local-part of the email address (i.e., the part preceding the @ symbol) and the SMS gateway as the domain. For example, the email address for the +1-312-555-1234 phone number, serviced by T-Mobile, will take the form of 3125551234@tmomail.net. Here is a partial list of SMS gateways that are available.

Carrier	SMS Gateway
Alltel	@message.allte.com
AT&T	@txt.att.net
Boost Mobile	@sms.myboostmobile.com

Carrier	SMS Gateway
Cricket	@sms.cricketwireless.net
Cricket Wireless	@sms.cricketwireless.net
Google Fi (Project Fi)	@msg.fi.google.com
Metro PCS	@mymetropcs.com
Project Fi	@msg.fi.google.com
Republic Wireless	@text.republicwireless.com
Sprint	@messaging.sprintpcs.com
T-Mobile	@tmomail.net
U.S. Cellular	@email.uscc.net
Verizon	@vtext.com
Virgin Mobile	@vmobl.com
Xfinity Mobile	@vtext.com

Consider changing **When active** to a time period that corresponds to regular business hours. If you plan on enabling a separate action that sends out email notifications, you will still get those 24/7. In that case, it may not be necessary to also receive those same messages as text alerts during night hours.

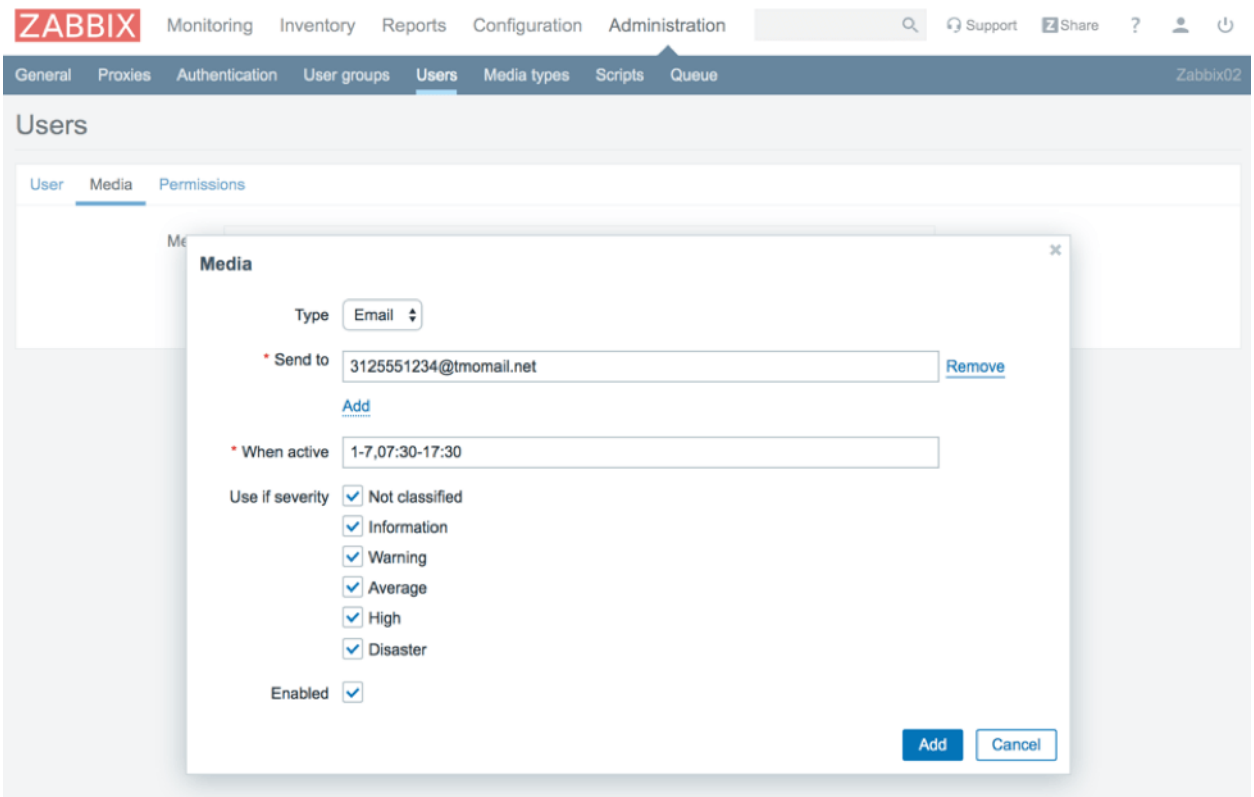


Figure 35. Change 'When active' to correspond with regular business hours

Click **Add** to add the media and then click **Add** again to add the new user.

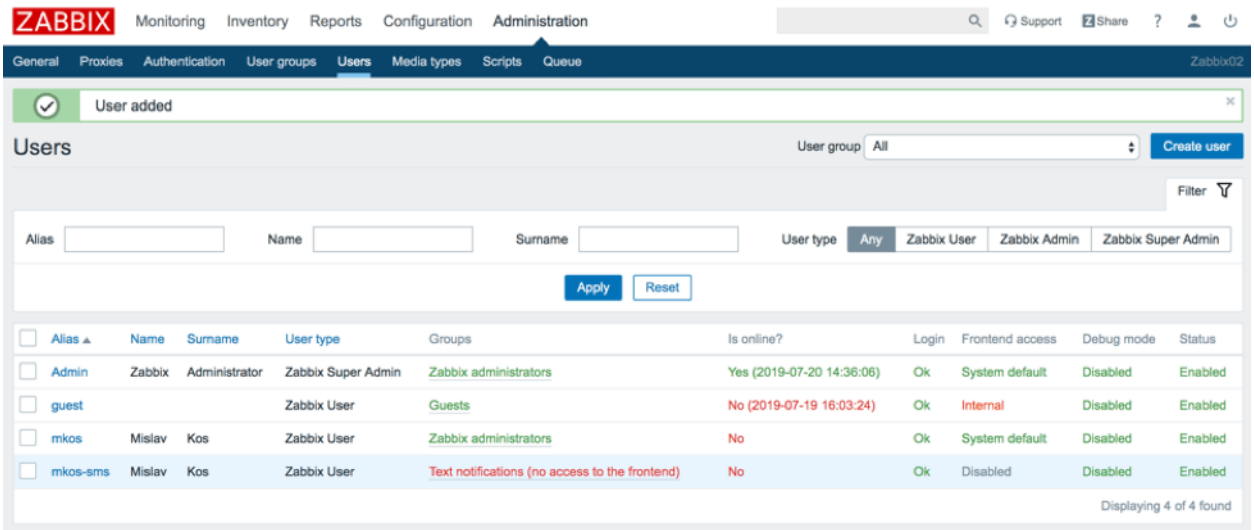


Figure 36. New user is added to the Users list

Now add a new action the same way as before, but this time use the Text notifications (no access to the frontend) user group instead of Zabbix administrators.

Remote Commands

As an example, we will set up an action that will start the FileMaker script engine if we notice that it is not running.

Navigate to **Configuration** > **Actions** and click **Create** action. Give the action a name. Select **Trigger** in the first **New condition** dropdown. Click **Select**, and select the **FMS process not running - Script Engine - Windows**, and click Select. Click the blue **Add** link to add the new condition.

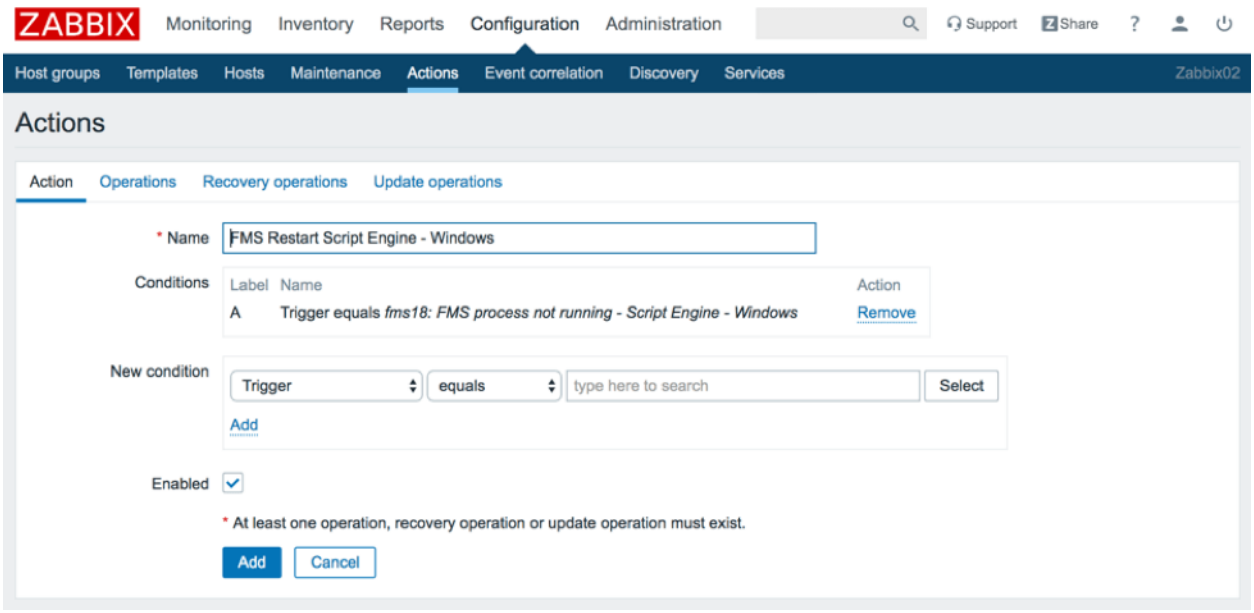


Figure 37. Create new action to start the FileMaker script engine if it's not running

Switch to the Operations tab and change the default duration to 60 seconds and optionally modify the message.

In a moment, we will add two operations to this action. The first will issue the remote command, and the second will send an email notification. Those two commands must happen in sequence with the first one finishing before the second one can happen. For this reason, we change the duration from 1 hour to 1 minute, so that we don't have to wait an hour before the email notification is sent.

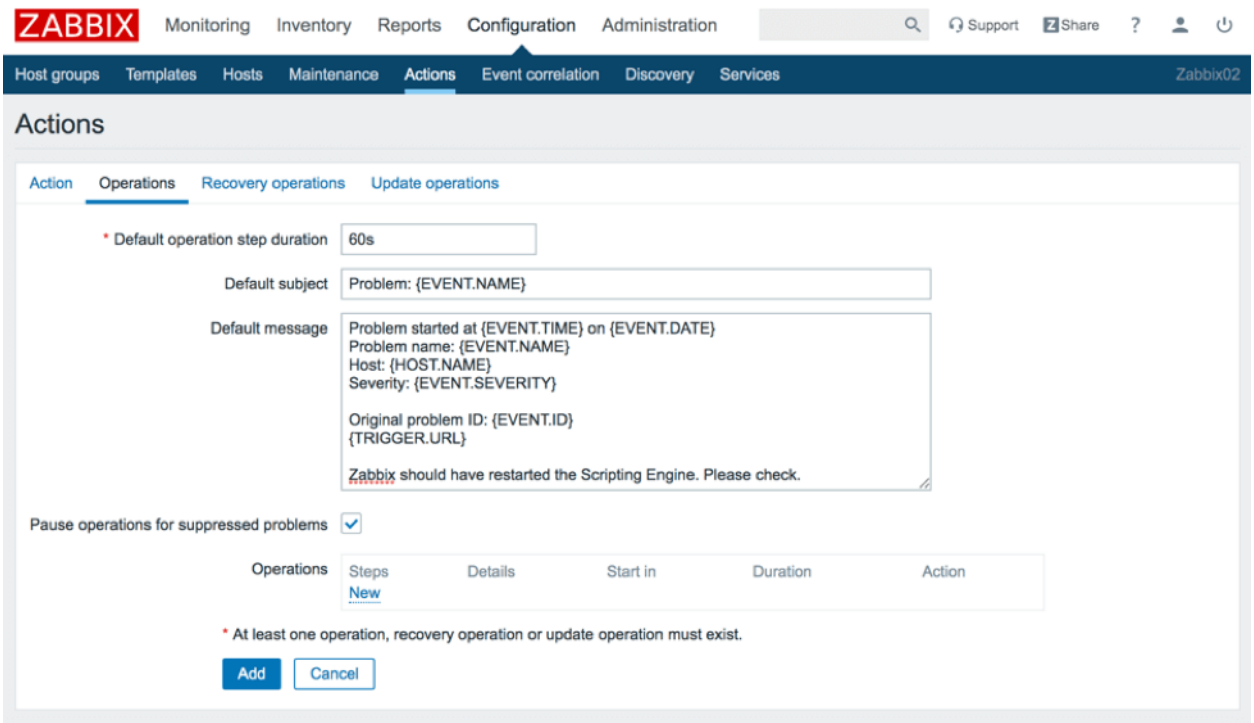


Figure 38. Default duration changed to 60 seconds and optionally modify the message

Click **New** to begin adding a new operation. Change the end step to 5 to indicate that you would like to try starting the script engine process up to five times. If the start command succeeds on the first try, the remaining four iterations of this step will be canceled. Change **Operation type** to **Remote command**. Click **New** next to **Target list**. Keep **Current host** selected as the **Target** and click the blue **Add** link to add the target. Enter the following command in the **Commands** field:

```
fmsadmin start fmse
```

On macOS, you have to include the full path to fmsadmin since the Zabbix agent uses a different shell than macOS does. macOS uses bash as the default, and the Zabbix agent uses sh. The fmsadmin command is not registered in the shell, so it will not be accessible from any location like it is in bash.

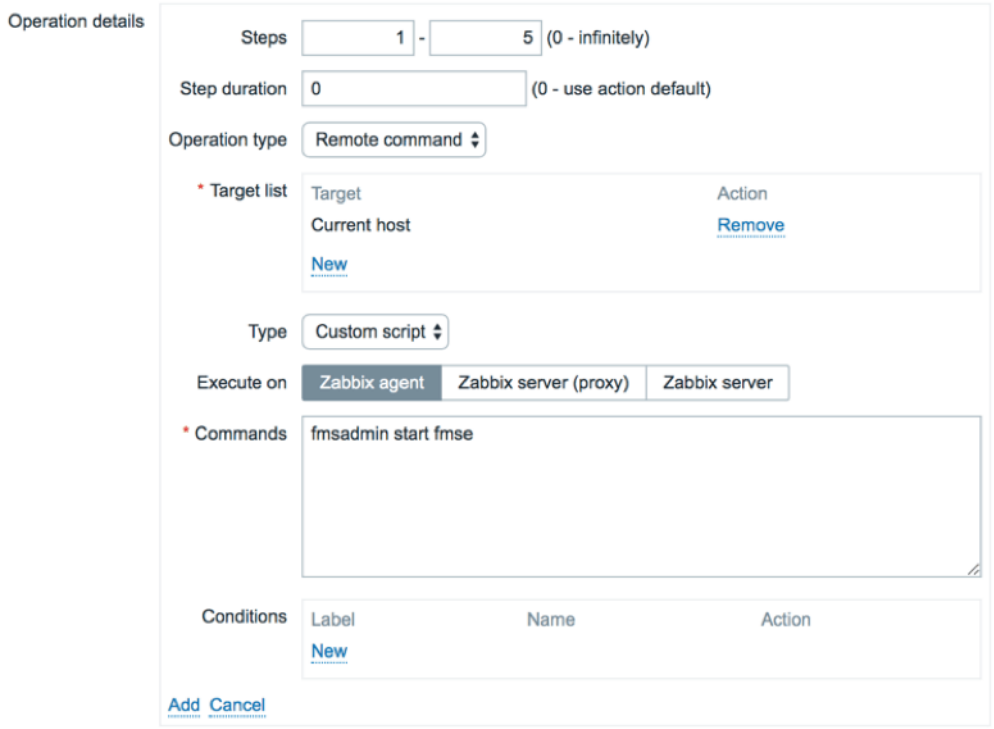


Figure 39. Change the end step to 5 for the new operation

Click the blue **Add** link to add the operation. Then click **New** to add another operation to send an email notification to accompany the first operation. Fill out the details similar to what is shown in Figure 40.

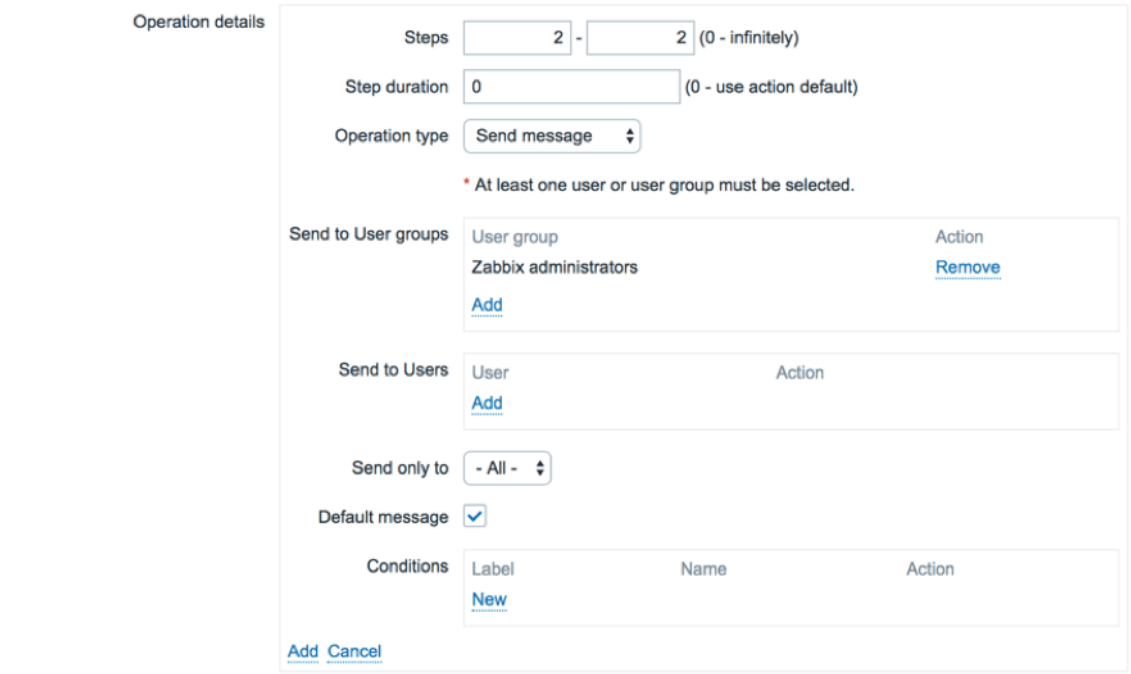


Figure 40. Add another operation with details as shown above

Once you click the blue **Add** link to add the second operation, the two operations should display as shown here:

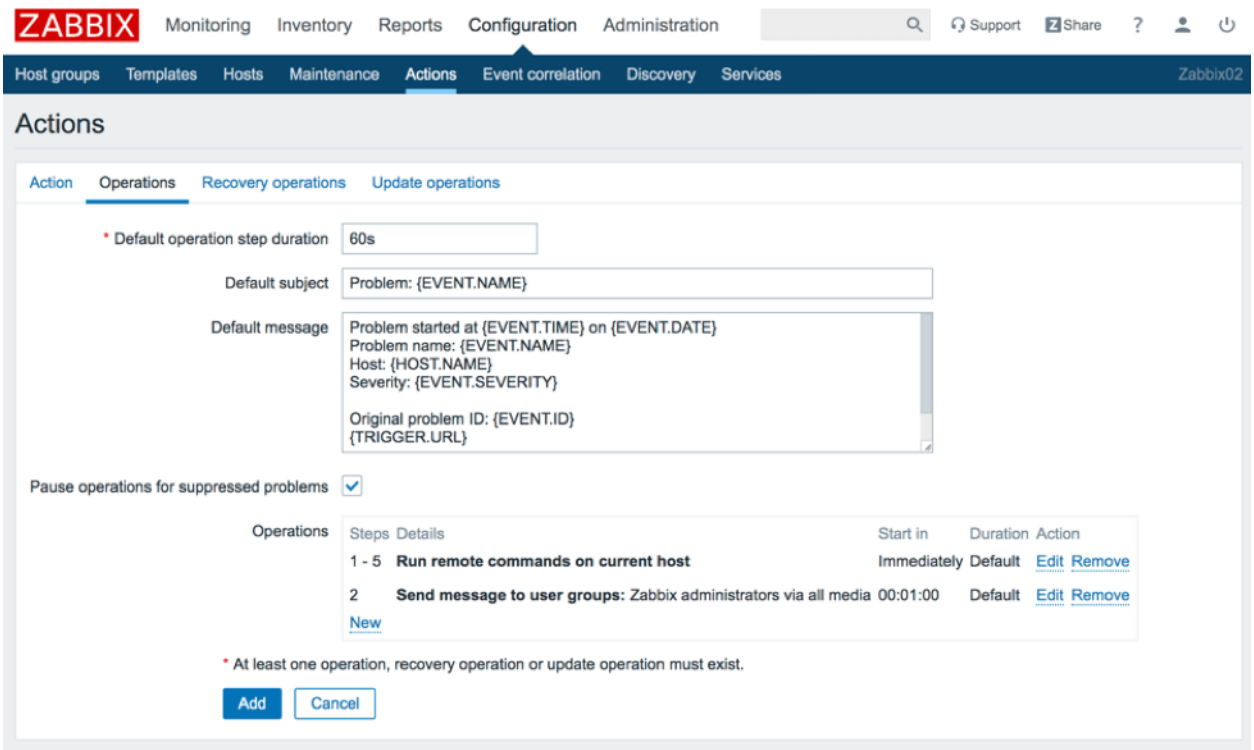


Figure 41. Two operations should be shown in the 'Operations' field

Click the blue **Add** button to add the action.

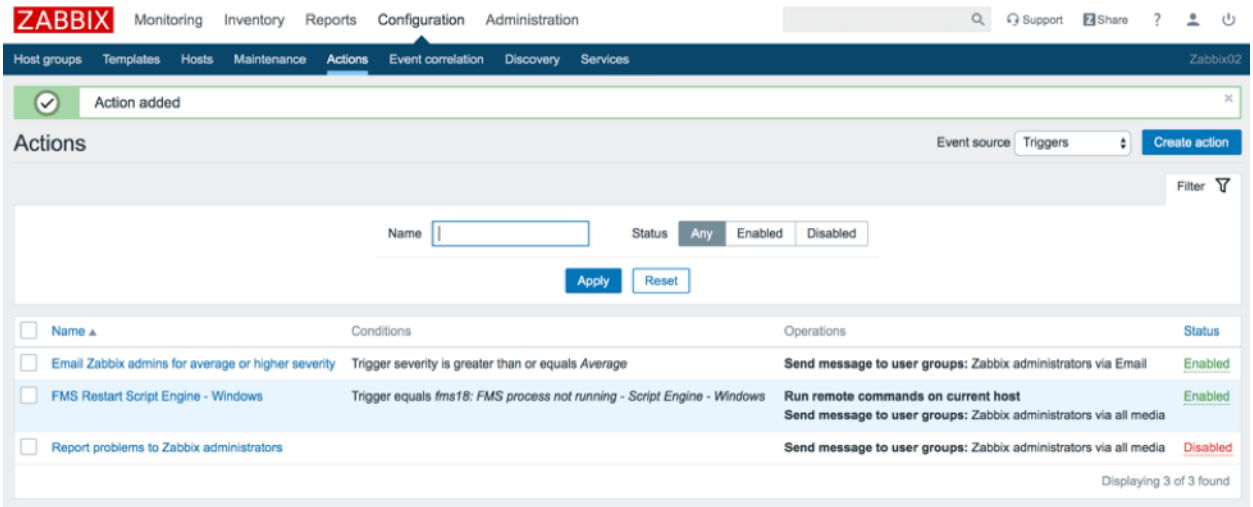


Figure 42. New action is added to the Actions list

Other Considerations

Change the number of entries shown in a list

The Soliant Consulting template has many items which get broken up into multiple pages when viewed in a list. To have them displayed on a single page, go to your user profile and change the **Rows per page** to a higher number; e.g., 200.

Securing Zabbix

If you haven't already done so, change the default password used by the default Admin account. You can do so quickly by clicking on the person silhouette icon in the upper right and then clicking the **Change password** button.

Review the Zabbix documentation to see what other security measures make sense for your situation.

Monitoring Your FileMaker Server

Using Zabbix

By Wim Decorte, Senior Technical Solution Architect
and Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Contents

The Zabbix Configuration white paper covered how to use the Zabbix web frontend to configure and administer Zabbix. This white paper will cover how to use the Zabbix web frontend to view the item data that has been collected and the problems and actions that have been identified and taken. Not all of the sections will be covered. Our aim is to provide a reasonably thorough overview of the parts that will be most useful to you as a FileMaker developer or administrator. Please refer to the Zabbix documentation if you would like to learn additional details about a particular section.

Our full set of guides is available at https://www.soliantconsulting/filemaker-zabbix.

Monitoring Items and Problems

Overview

The **Monitoring** > **Overview** section shows either **item** or **trigger** data depending on the Type that is selected.

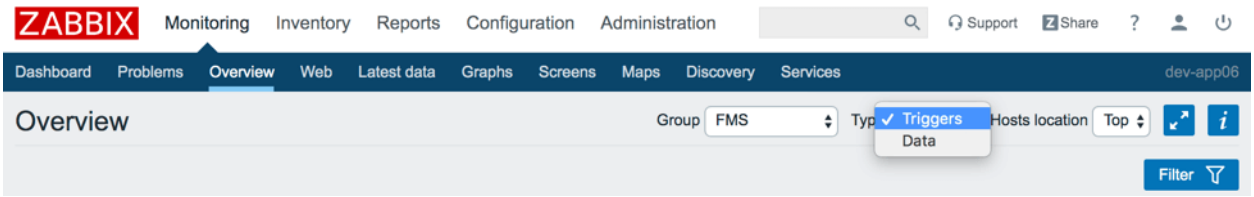


Figure 1. Use the 'Type' dropdown to change what is shown in the Overview

The **item data overview** shows a list of the most recently collected values – similar to the **Monitoring** > **Latest data** section, which we will cover next – for the hosts in the selected host group. The list can be filtered further by application. Clicking on the filter tab will toggle the display of the filter section.

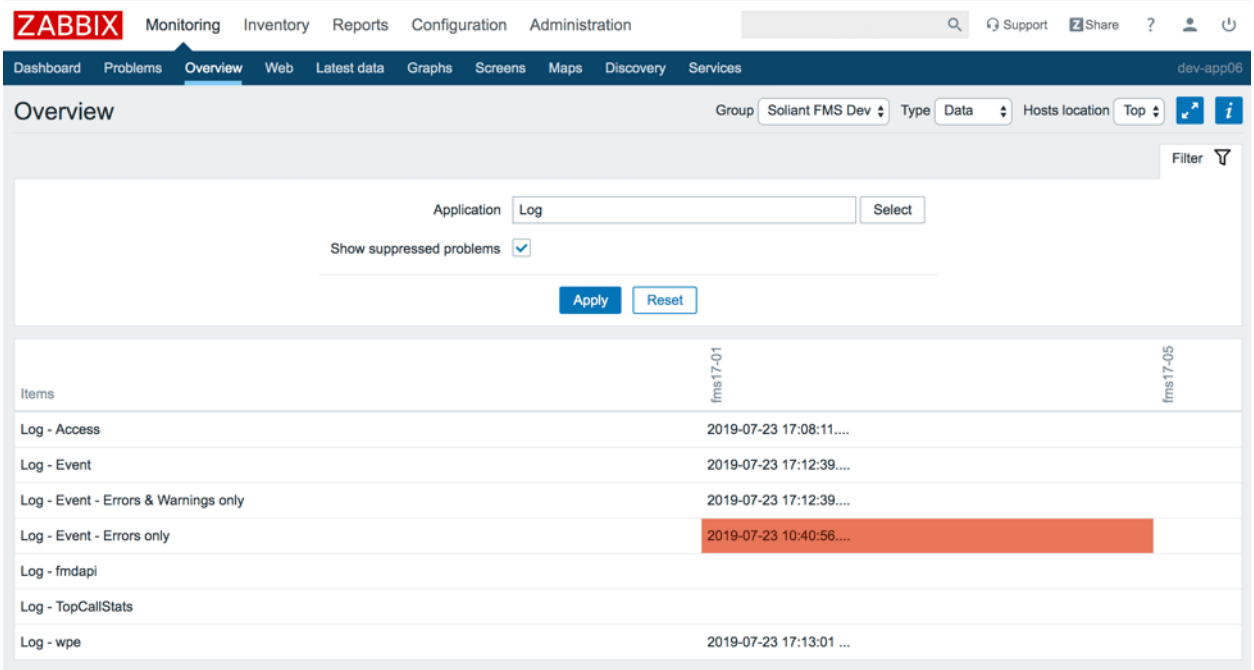


Figure 2. The data overview can be filtered by application

The **trigger overview** shows a list of triggers for the hosts in the selected host group along with their states and severity. Green is used when the **trigger status** is **OK**. The other colors are used to indicate severity for triggers in the **Problem** state. A checkmark indicates that the problem has been acknowledged. The filtering section provides several additional filtering options that are not available for the item data overview.

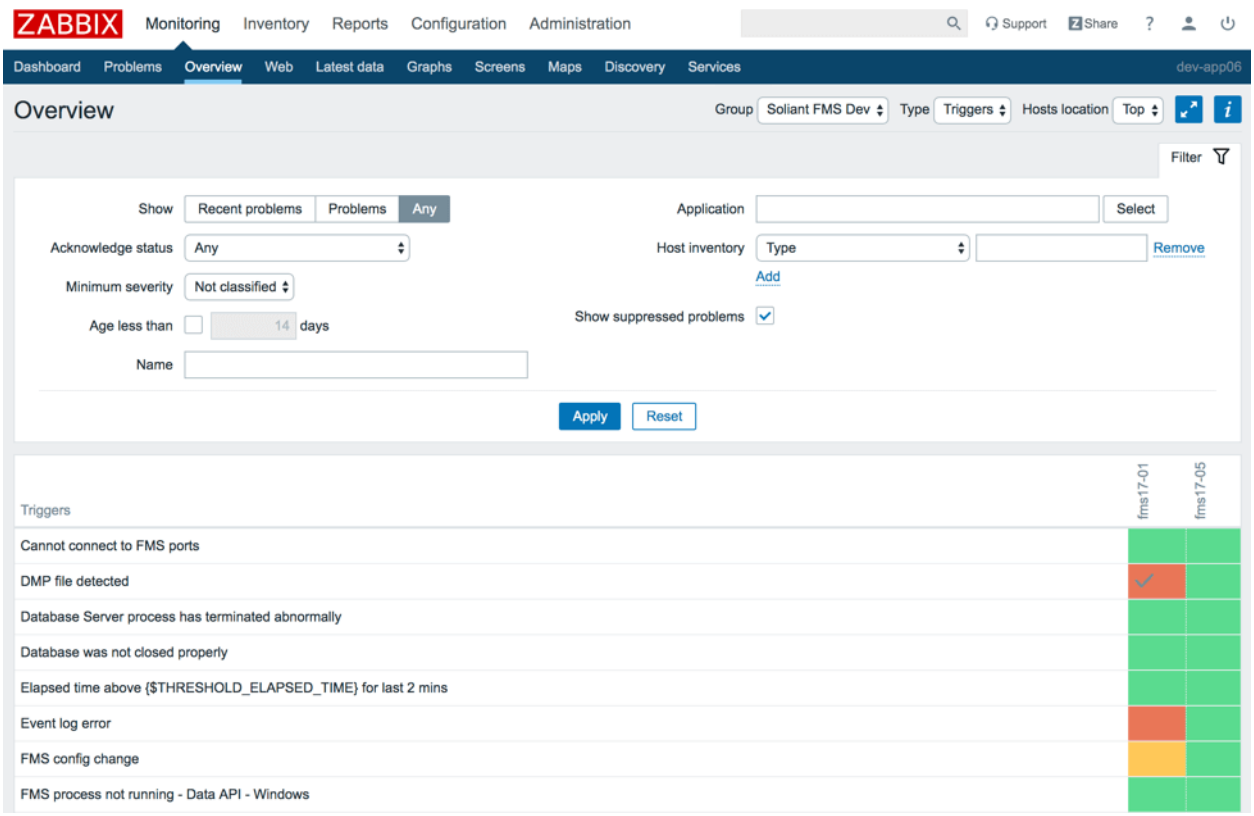


Figure 3. The trigger overview shows trigger states and severity

Latest Data

The **Monitoring** > **Latest** data section shows the most recently collected item data and the time at which it was collected. The values are grouped by host and application. Somewhat confusingly, if an item has been tagged with multiple applications, it will appear in the list multiple times. The filtering section allows for searching by host group, host, application, and name.

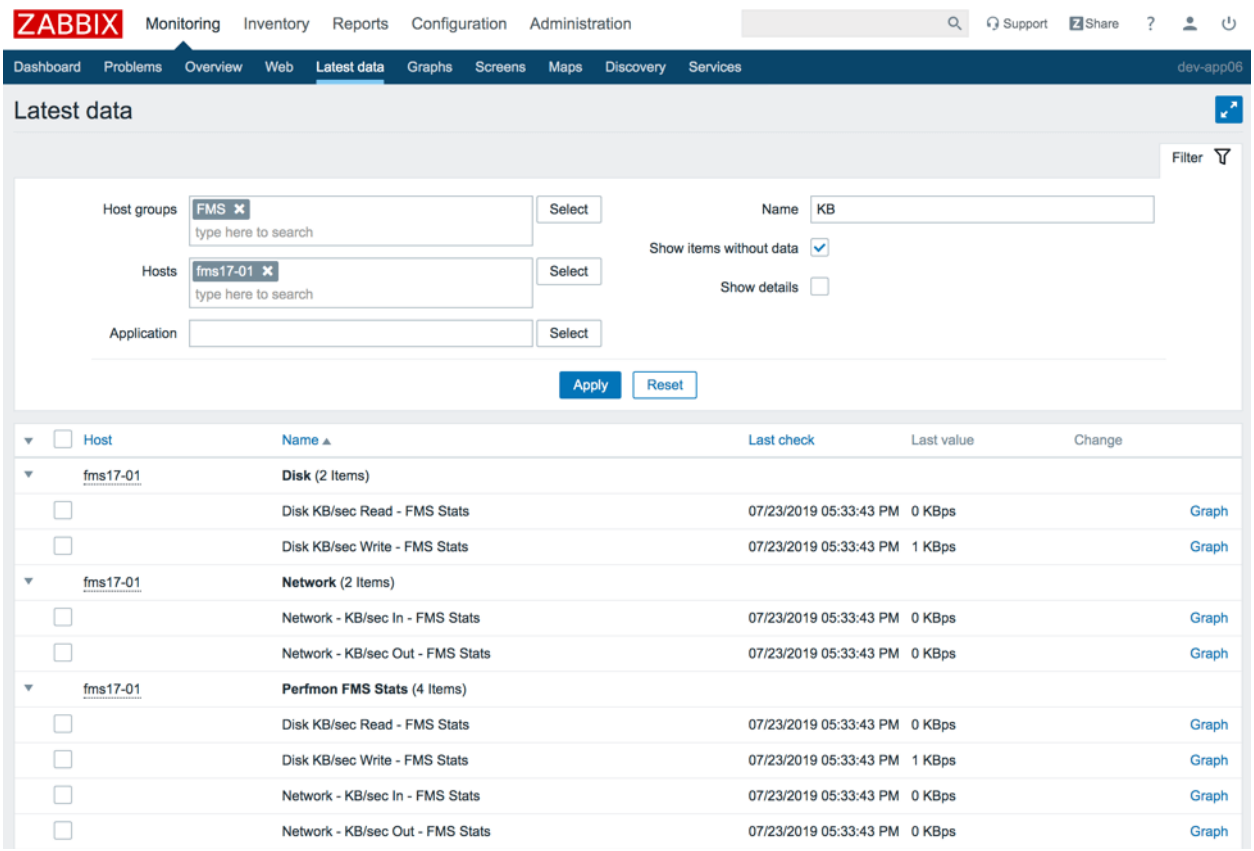


Figure 4. Latest data can be filtered by host group, host, application, and name

This section is useful for confirming that the item data is being collected as expected. A history of the values can be seen in an ad-hoc graph by clicking on the **Graph** link. Text-based item data will have a **History** link instead to view a list of all collected data for that item. Once you switch to the detail for an item – by clicking Graph or History – the time period can be adjusted using the filter section.

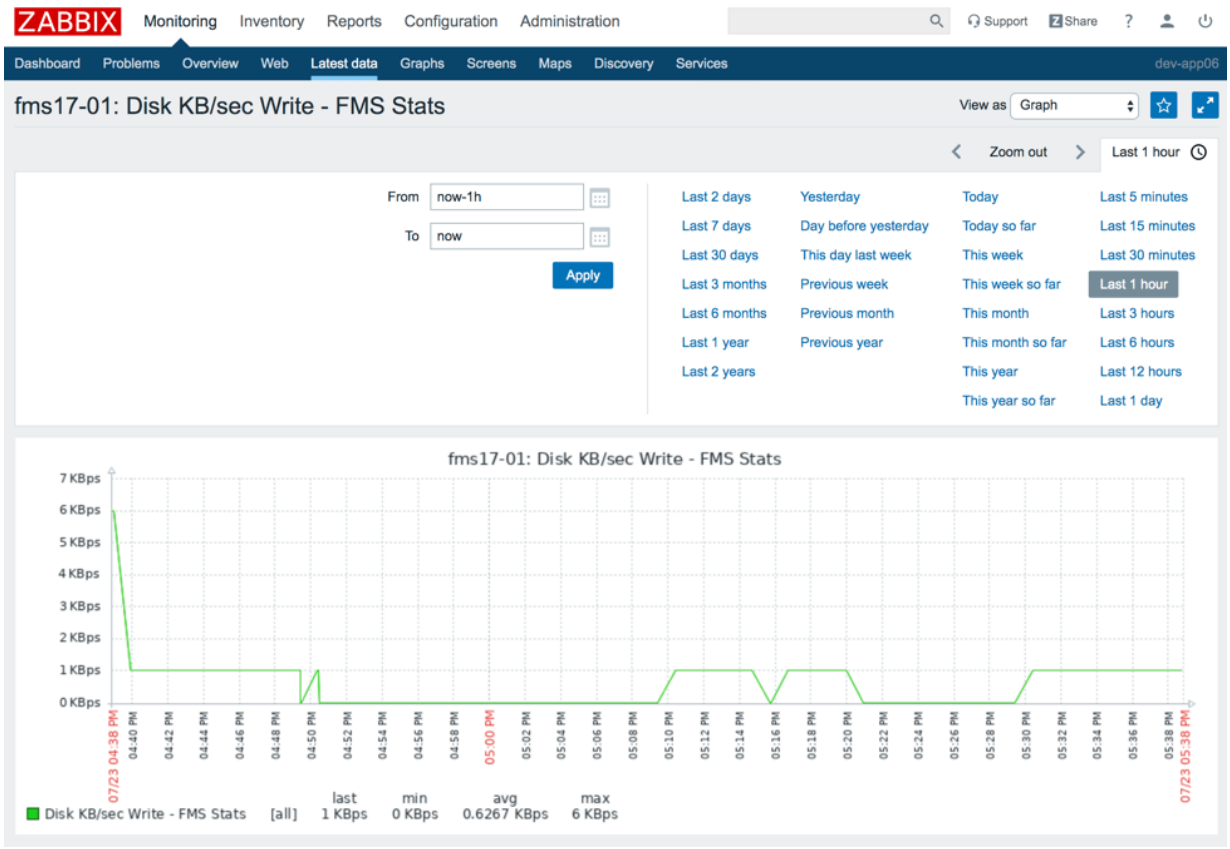


Figure 5. Clicking on the 'Graph' link opens an ad-hoc graph

Data shown in a graph can also be viewed as a list of values by changing the **View as** option.

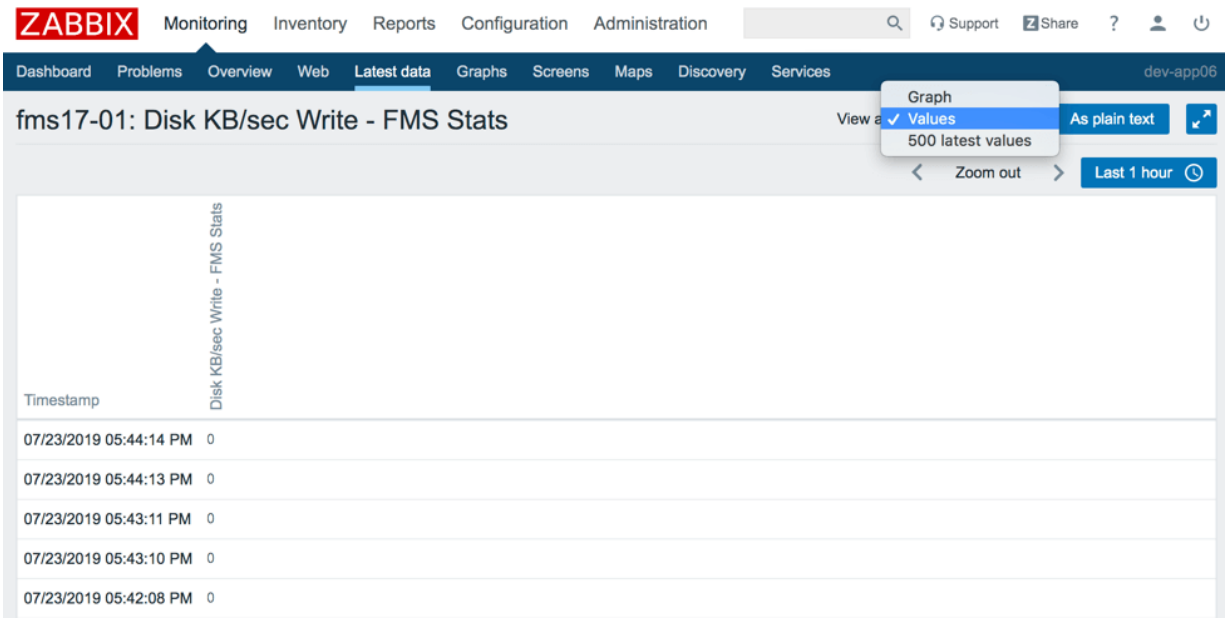


Figure 6. Select 'Values' in the 'View as' dropdown to change from a graph to a list

The data shown can be quickly transferred to Excel or some other software for further analysis by clicking the **As plain text** button and then copy-and-pasting the values.

fms17-01: Disk KB/sec Write - FMS Stats

```
07/23/2019 05:45:18 PM 1563921918 0
07/23/2019 05:45:17 PM 1563921917 0
07/23/2019 05:44:14 PM 1563921854 0
07/23/2019 05:44:13 PM 1563921853 0
07/23/2019 05:43:11 PM 1563921791 0
07/23/2019 05:43:10 PM 1563921790 0
07/23/2019 05:42:08 PM 1563921728 0
07/23/2019 05:42:08 PM 1563921728 0
07/23/2019 05:41:05 PM 1563921665 0
07/23/2019 05:41:05 PM 1563921665 0
07/23/2019 05:40:02 PM 1563921602 0
07/23/2019 05:40:02 PM 1563921602 0
07/23/2019 05:38:59 PM 1563921539 6
07/23/2019 05:38:58 PM 1563921538 6
07/23/2019 05:37:56 PM 1563921476 1
07/23/2019 05:37:55 PM 1563921475 1
07/23/2019 05:36:53 PM 1563921413 1
07/23/2019 05:36:52 PM 1563921412 1
07/23/2019 05:35:51 PM 1563921351 1
07/23/2019 05:35:49 PM 1563921349 1
07/23/2019 05:34:47 PM 1563921287 1
07/23/2019 05:34:47 PM 1563921287 1
07/23/2019 05:33:43 PM 1563921223 1
07/23/2019 05:33:43 PM 1563921223 1
```

Figure 7. Use the 'As plain text' button to view the data as plain text

Problems

The **Monitoring** > **Problems** section will display the problems that have been generated by triggers.

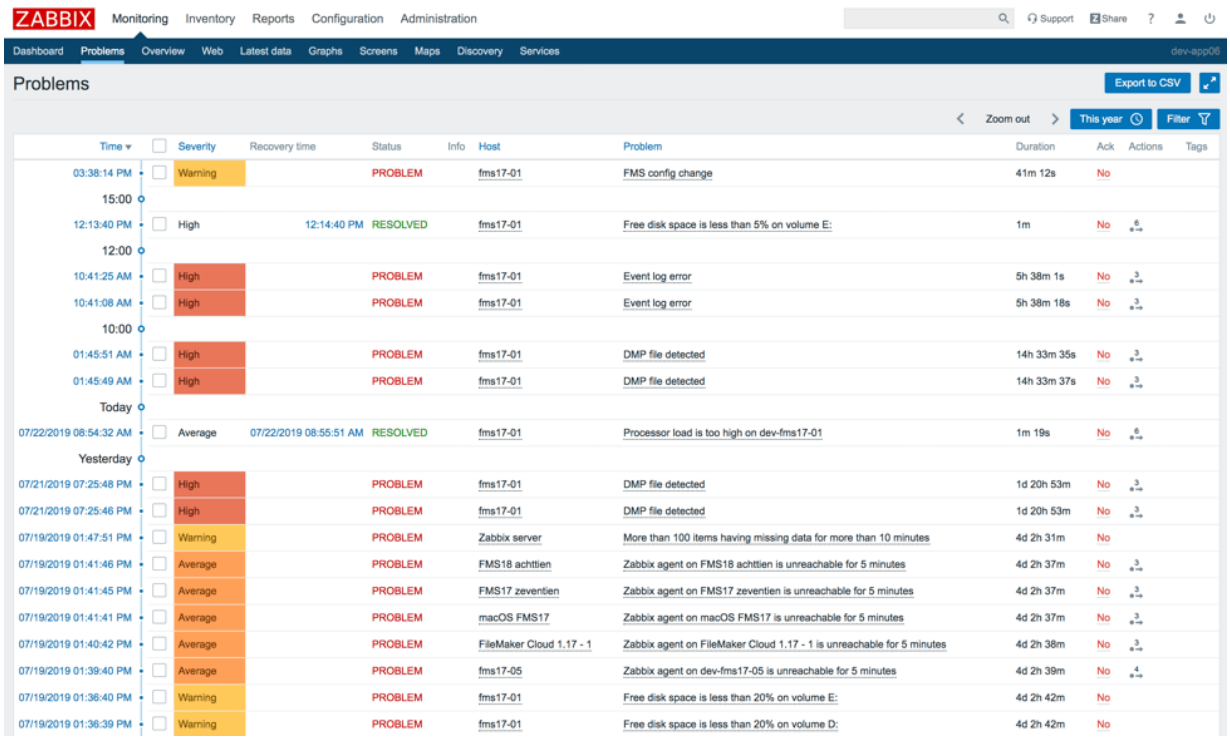


Figure 8. Problems list

The filter section provides many criteria through which the data can be winnowed down. The default filtering option will show only **Recent problems,** so if you are not seeing problems that you expect to see, make sure to set the **Show** option to **Problems** or **History**. The data can be sorted in different ways by clicking on the column header. The timeline view can be very helpful in breaking up the data into meaningful chunks, so make sure the **Show timeline** option is enabled.

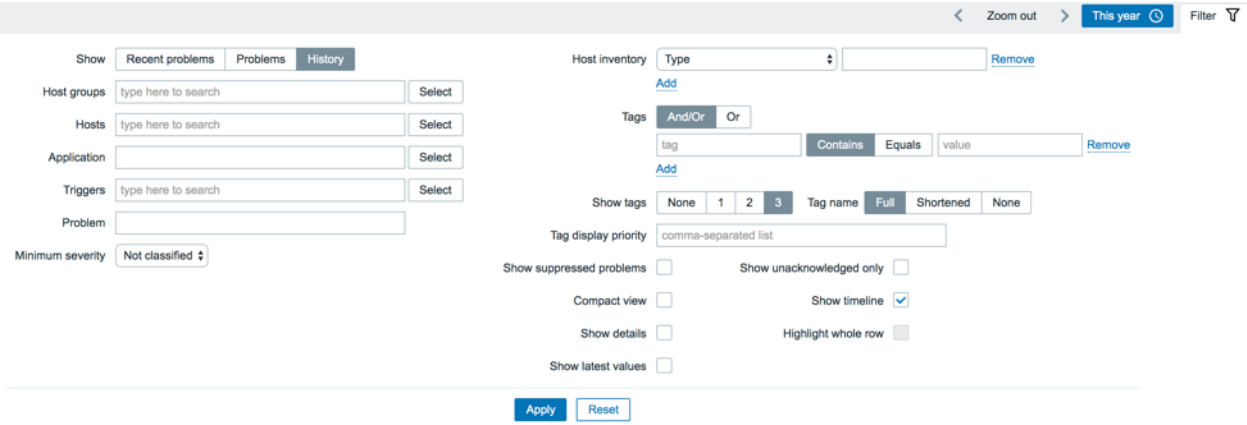


Figure 9. Use the filter to show more than the default 'Recent problems'

Time column: Click on the problem creation timestamp to view details about the problem event, including information about the trigger that generated the problem, a list of automated actions taken, and a list of times when this problem occurred previously.

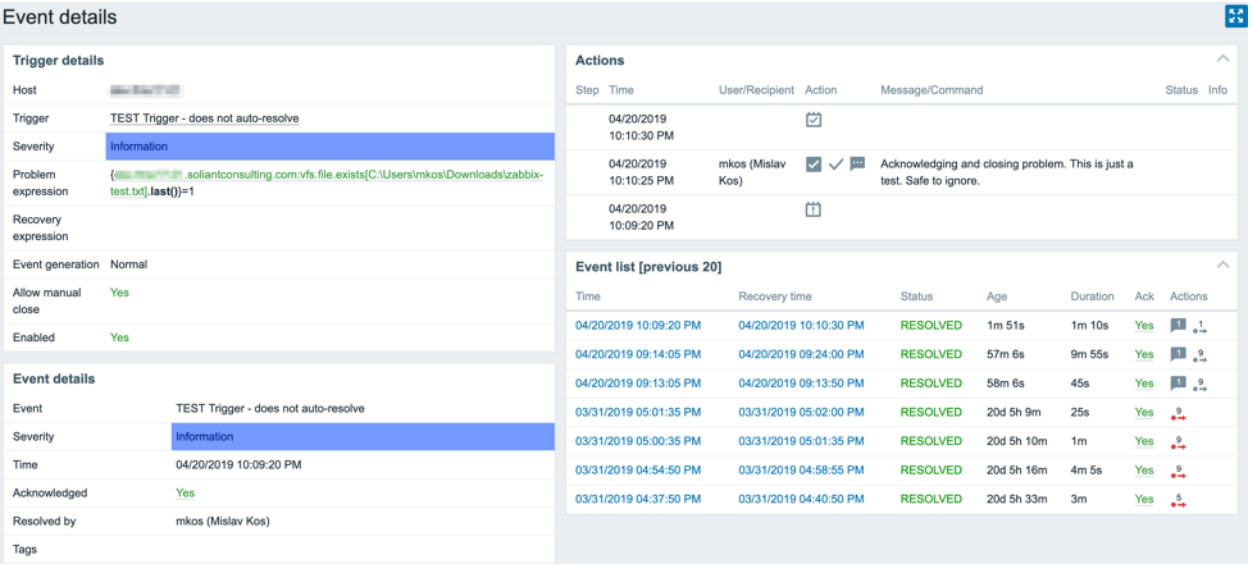


Figure 10. View problem event details by clicking on the problem creation timestamp

Host column: Clicking on the host name brings up the host menu which makes it possible to jump to other Monitoring sections with the filter automatically adjusted for that host. This popup host menu is available in the same way – by clicking on the host name – in several other sections of the frontend as well where the host name is displayed with a dotted underline.

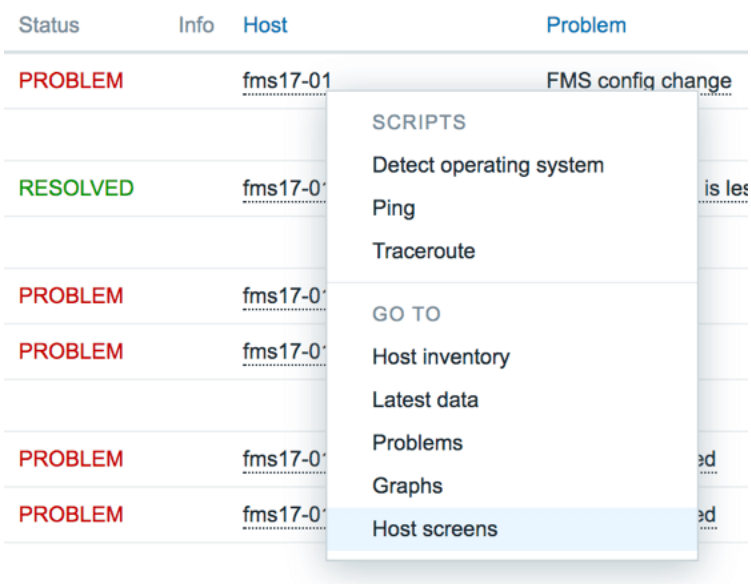


Figure 11. Clicking on the host name opens the host menu

Ack column: You can click on the Yes/No value in the **Ack** (Acknowledge) column of the problem list to bring up a screen where you can update the problem. For example, you can add a message to keep notes about the problem, you can acknowledge the problem (to let your teammates know that you are working on it), or you can close the problem (if the trigger settings allow for manual resolution).

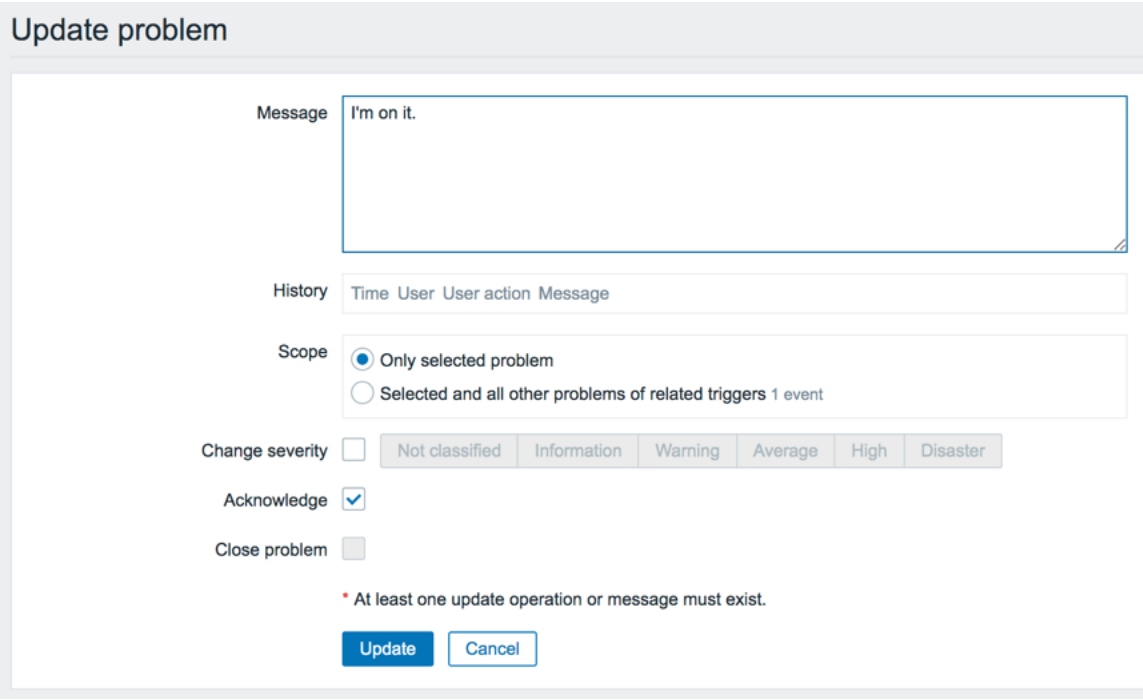


Figure 12. Click on 'Yes' or 'No' in the Ack column to update the problem.

Actions column: This column shows a count of how many actions – notifications or remote commands – were taken in response to a problem. Hover or click on the count to see the list of actions.

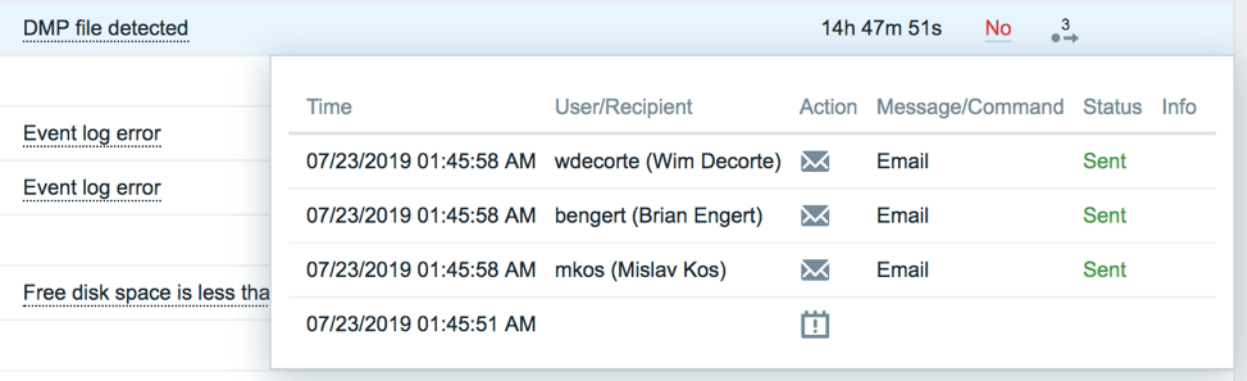


Figure 13. Clicking on the count in the Action column opens the popup showing the list of actions taken

Dashboard

Zabbix makes it possible to create multiple dashboards, each of which can display summaries of various types of data. Dashboards cannot be shared using templates – discussed in the Zabbix Configuration white paper – so you will need to create your own from scratch or modify the **Global view** dashboard that comes predefined when Zabbix is installed.

You can view a list of available dashboards by navigating to **Monitoring** > **Dashboard**. Click on any dashboard that is shown in the list to view its details. You can also create new or edit existing dashboards from here.

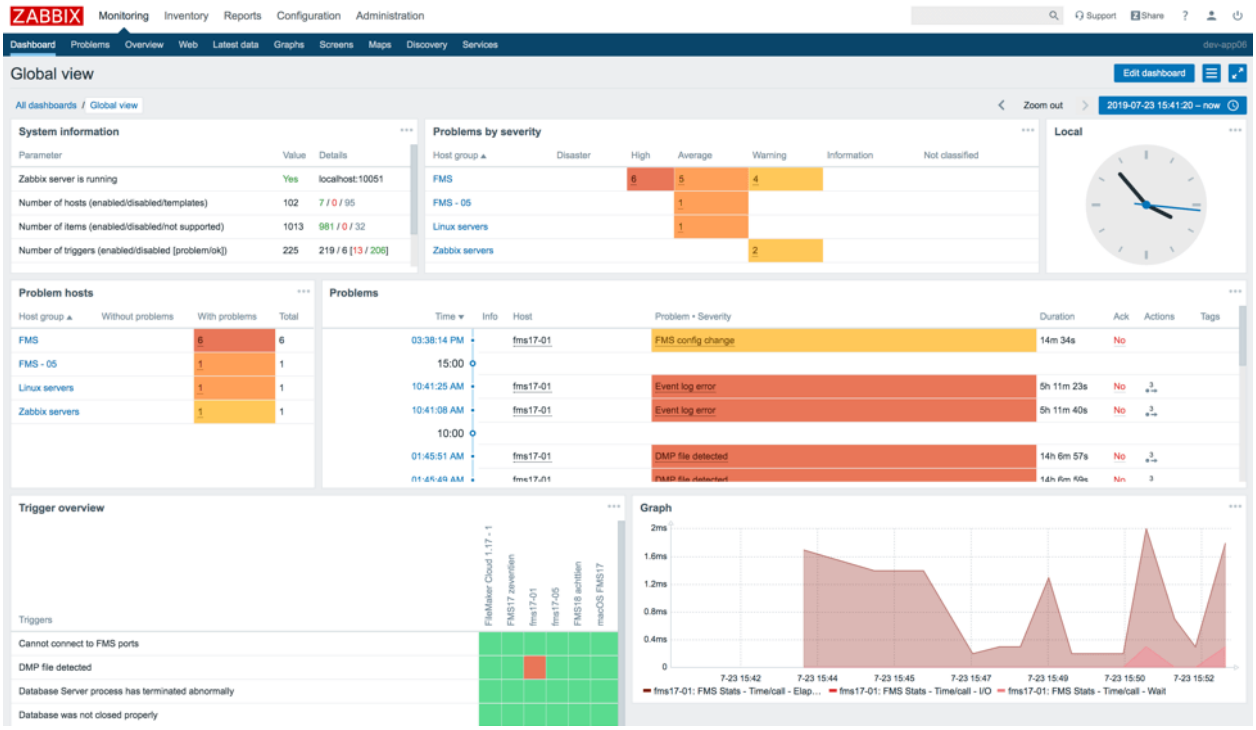


Figure 14. Click on any dashboard shown in the dashboard list to view its details

As is the case with many other Monitoring areas in the web frontend, when viewing a dashboard, the time filter can be adjusted using the time period selector. You can choose from quite a few pre-defined time filters, or you can specify a custom range.

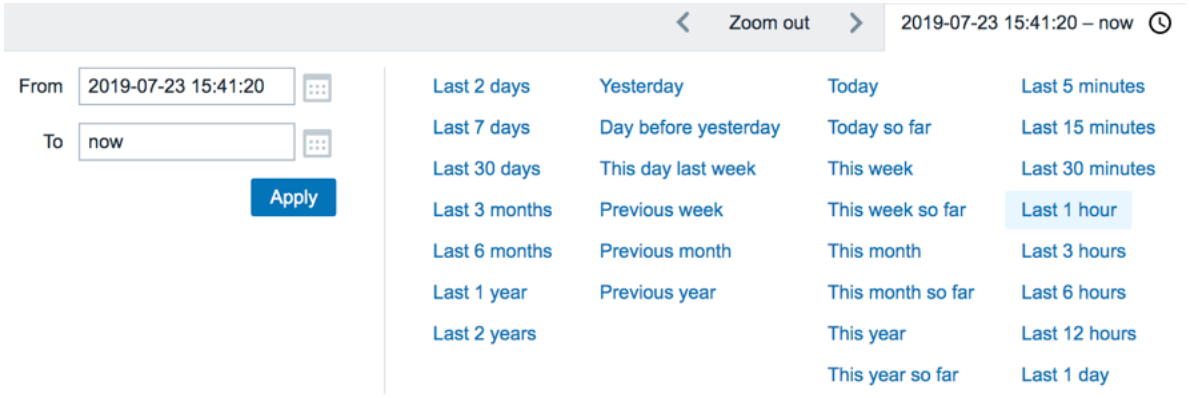


Figure 15. Use pre-defined filters or enter a custom range

Here's a partial list of widgets that can be shown in a dashboard:

- Data overview – Shows latest item data filtered by host group, application, and other criteria
- Shortcuts to graphs and screens you marked as favorites

- Graphs – Custom graphs created for the dashboard (these are different from the graphs defined for a host or template)
- Problems – filtered list or cross-tab summarized by severity
- Trigger overview – list of trigger states (OK or problem) for a group of hosts
- Action log – list of recent actions taken (notifications or remote commands)

Graphs

In the **Monitoring** > **Latest data** section, we saw how ad-hoc **graphs** can be created to view item data. Graphs can also be pre-defined for a particular host or, more generally, for a template (which would then be inherited by the hosts using that template). These pre-defined graphs can be viewed in **screens** – which we will cover in the next section – where they will typically be grouped with other information, or they can be viewed individually in the **Monitoring** > **Graphs** section.

To view one of these pre-defined graphs, navigate to **Monitoring** > **Graphs**, select a host group and a host, and then select one of the available graphs. You can adjust the time period using the section at the top. You can also click-and-drag to quickly zoom in on the portion of the graph you're interested in.

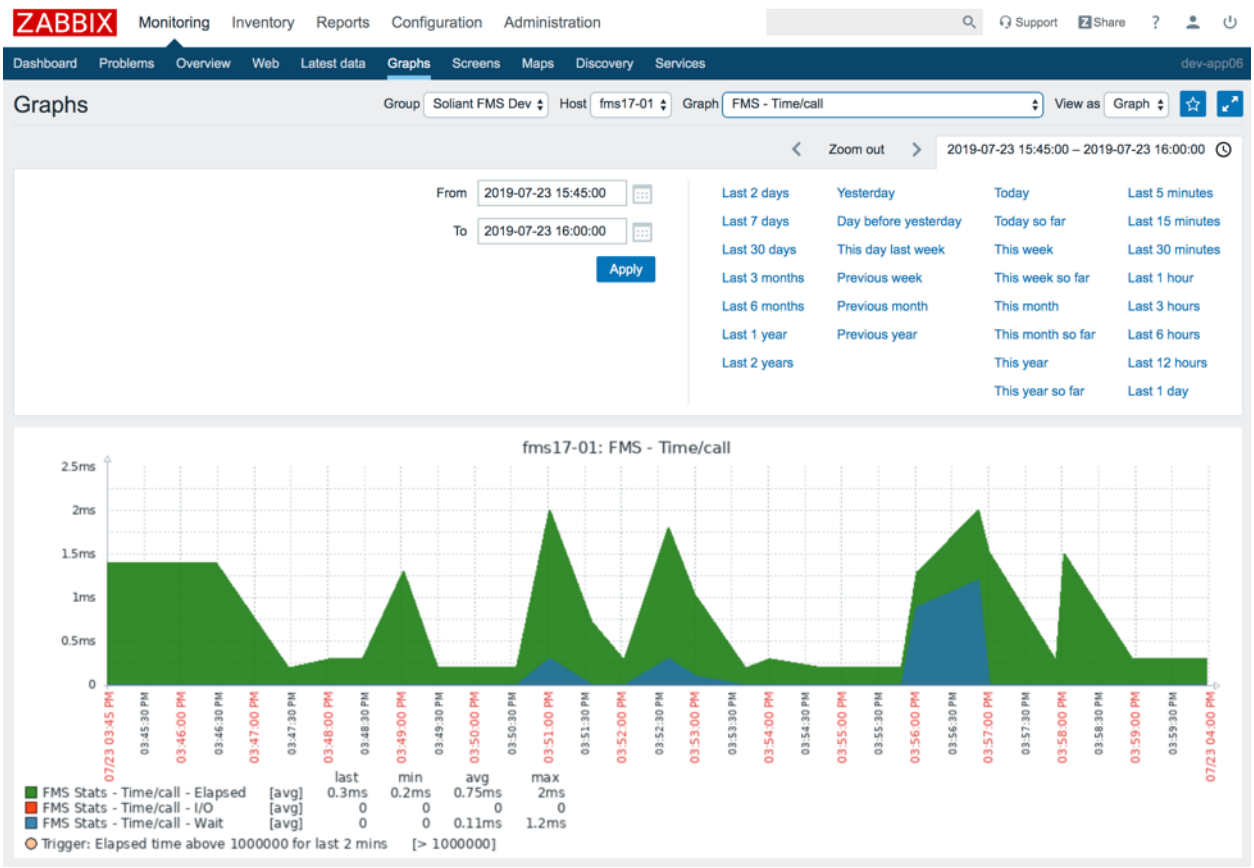


Figure 16. The graph time period can be changed using the time period selector or by clicking-and-dragging on the graph

The data values displayed in the graph can be viewed in a table by changing the **View as** option.

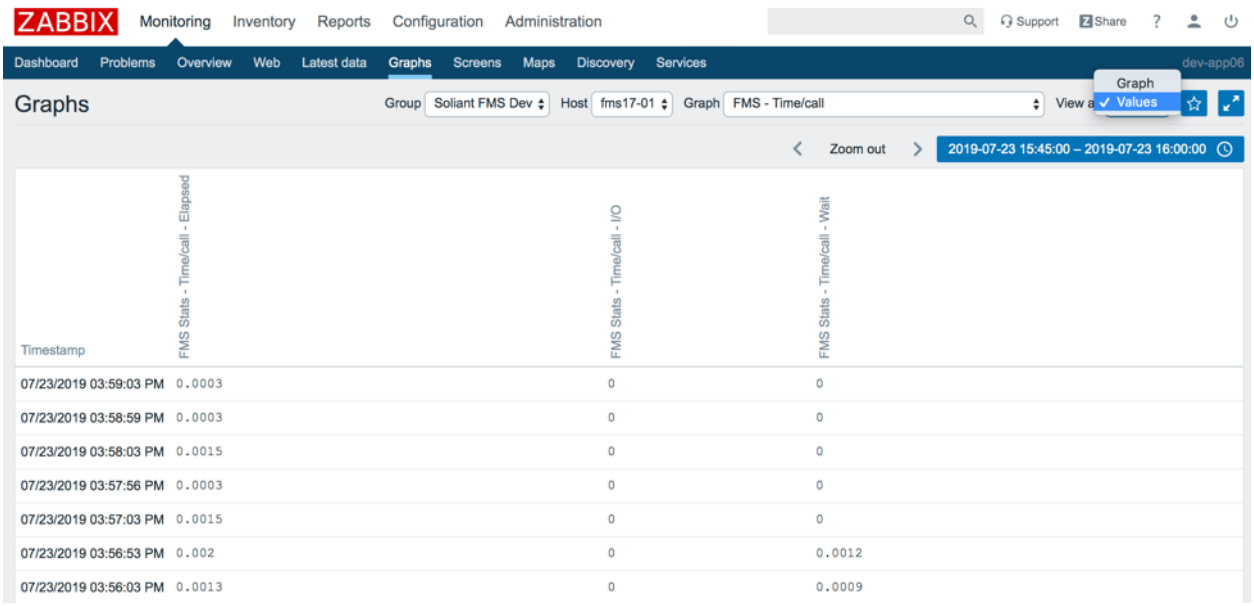


Figure 17. Select 'Values' on the View as dropdown to change the graph to a table

Screens

Like dashboards, Zabbix screens display different pieces of information on a single page. In fact, screens can include much of the same components as dashboards:

- Data overview
- Graphs
- Problems – list or cross-tab summarized by severity
- Trigger overview
- Action log

There are two kinds of screens. **Global screens** can display information from multiple hosts, and **host screens** are meant to display information from just one host.

Global screens are managed and viewed from **Monitoring** > **Screens**. As was the case with the Dashboard section, this section can be viewed as a list of global screens or as the detail of an individual screen – somewhat similar to how FileMaker has form and list views for any given layout.

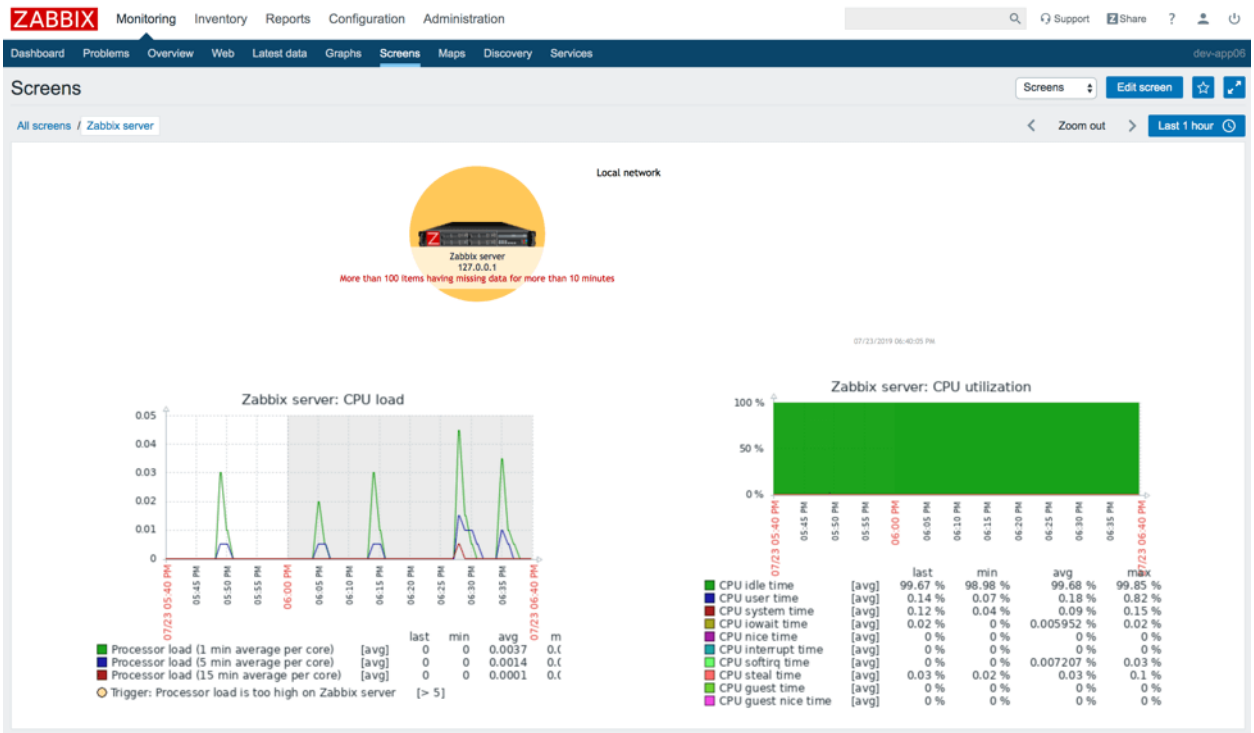


Figure 18. View Global screens as a list or detail of an individual screen

Host screens are managed within a template and viewed by selecting the **Host screens** option in a host menu which is a popup menu that can be accessed in various places by clicking on the host name. (We already touched upon host menus briefly in the Problems section of this white paper.) Some of these places include:

- Monitoring > Problems
- Monitoring > Overview
- Monitoring > Latest data

For example, in the **Monitoring** > **Problems** section, you can click on the name of a host and then on **Host screens** to access the screens available for that host.

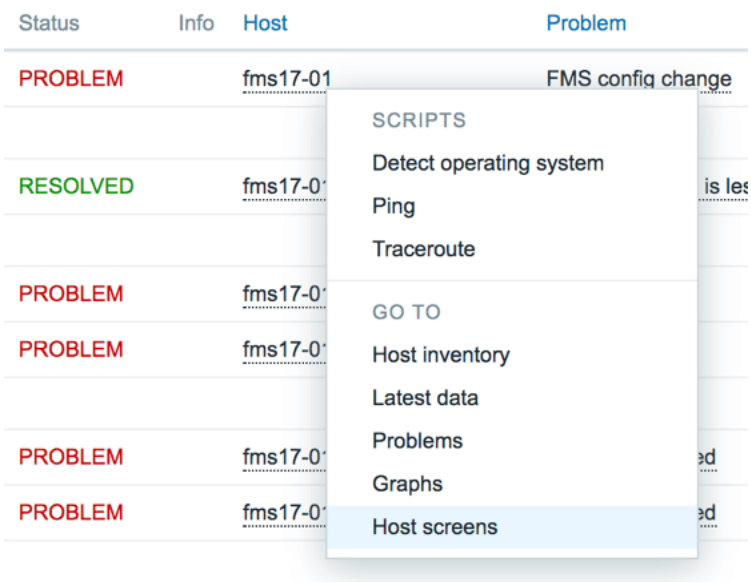


Figure 19. Click on 'Host screens' in the popup to view available screens for the host

Doing so will display one of the available host screens – whichever one was viewed most recently. You can switch to the other screens available for the host using the dropdown menu located in the upper-right section of the screen.

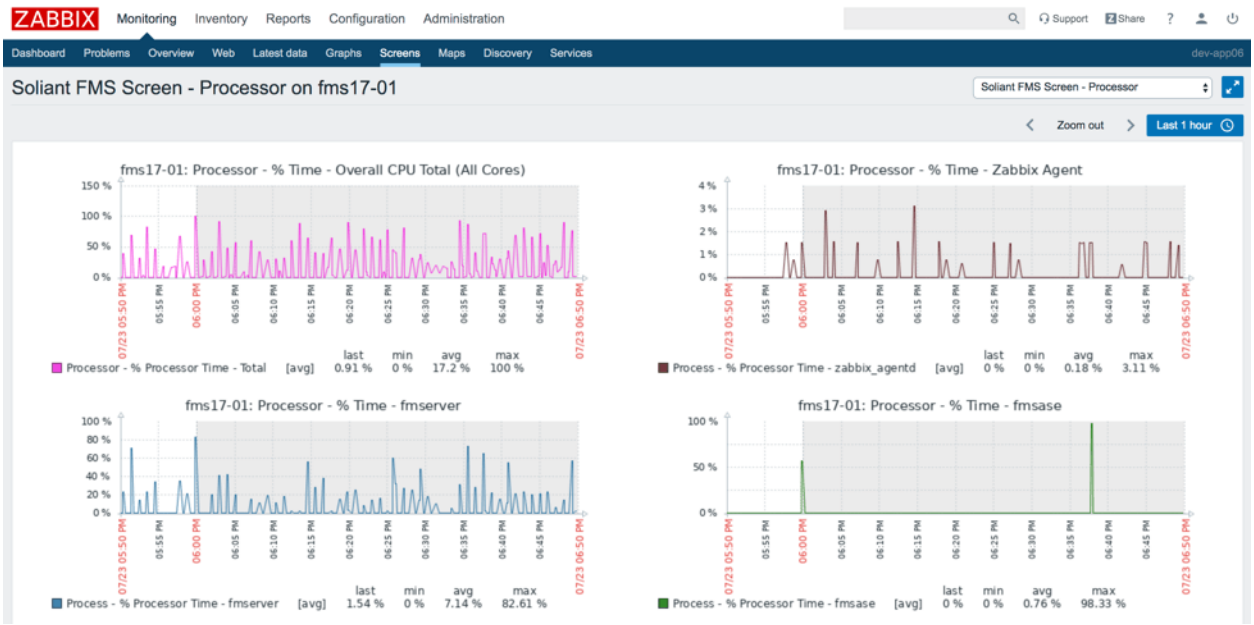


Figure 20. Available host screen

Host screens can be shared across Zabbix installations using templates. Sharing of global screens is less feasible. Global screen configurations can be saved by exporting them to an XML file, and they can be loaded by importing that XML file, but that file will

contain hardcoded references to specific hosts, so importing the file to another Zabbix server which doesn't have those same hosts defined will not work.

Zabbix Reports

Availability Report

The availability report shows what percentage of time each trigger spent in the OK and Problem states. The report data can be filtered by time period, host, and host group.

In the example below, we can see that the Zabbix agent on our FileMaker Cloud 1.17 development server has been unreachable 18.6% of the time.

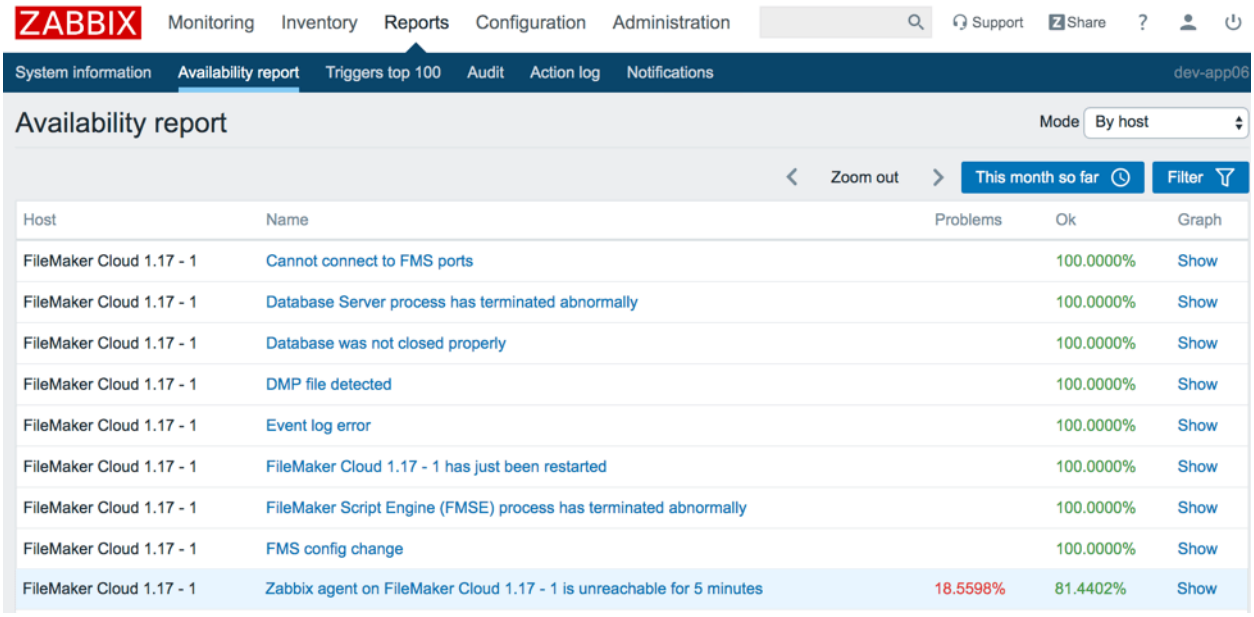


Figure 21. Availability report

You can click on the **Show** link to view the data in more detail and in a chart format.

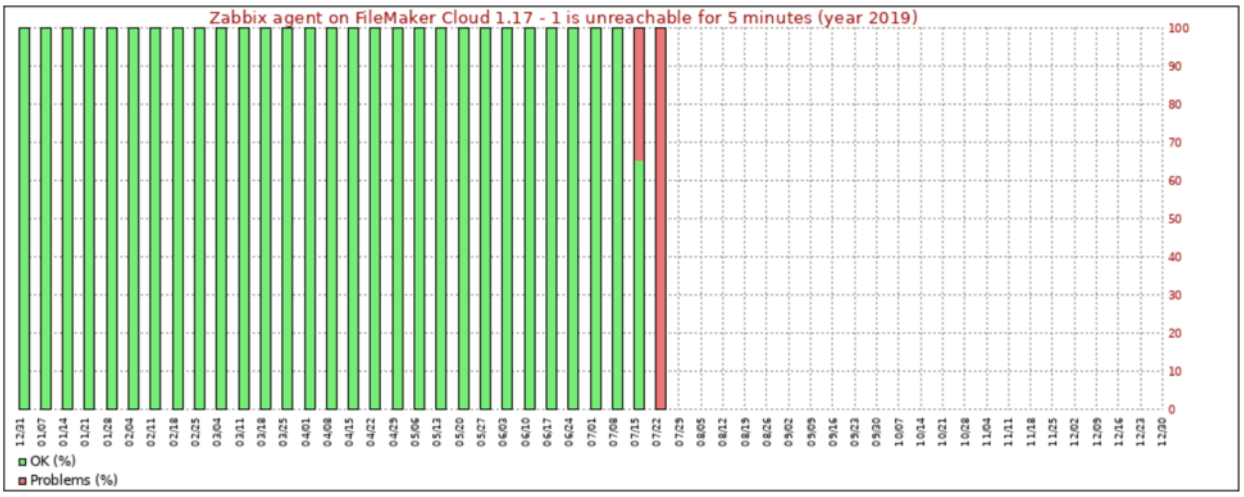


Figure 22. Clicking the 'Show' link shows a 100% stacked column chart for the trigger availability

Triggers Top 100

This report shows up to 100 triggers that have changed their state the most frequently in the selected time period. The report data can be filtered by time period, host, host group, and trigger severity.

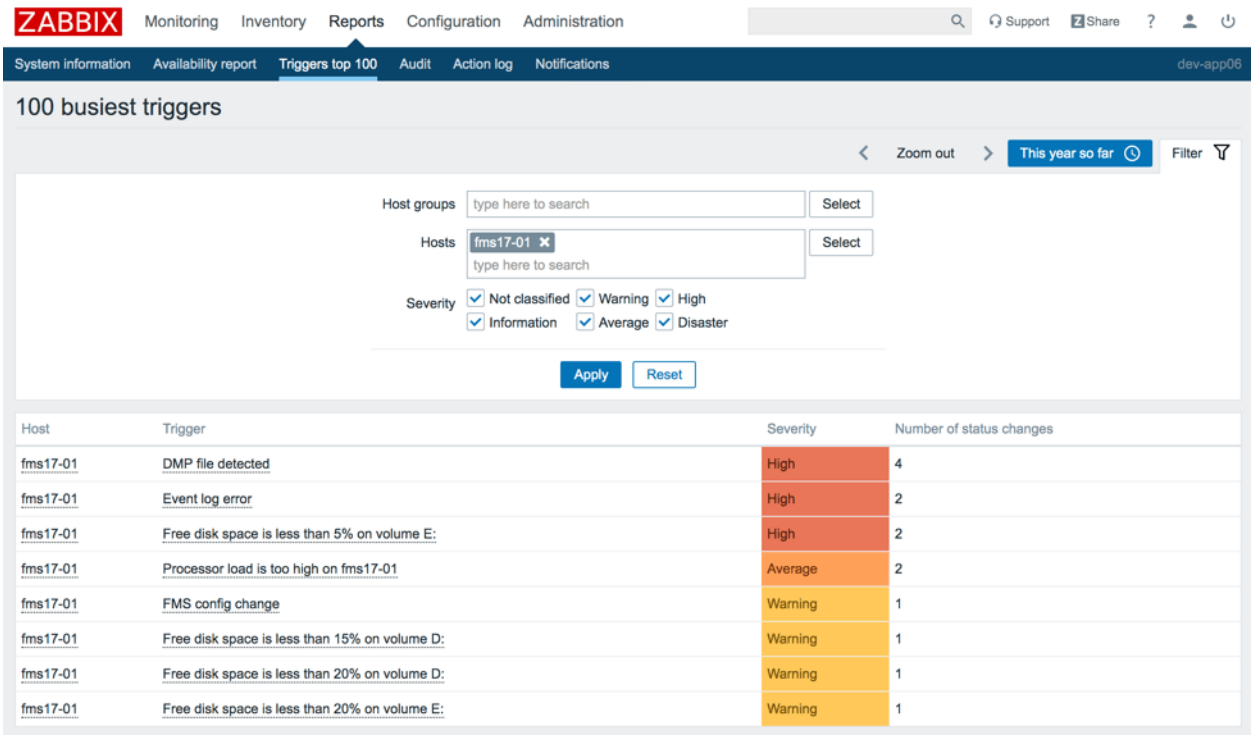


Figure 23. The top100 triggers can be filtered by time period, host, host group, and trigger severity

As is the case in other areas of the web frontend, clicking on the information shown with a dotted underline – in this case, host name and trigger name – will reveal contextual menus that can be used to jump to related views of the data.

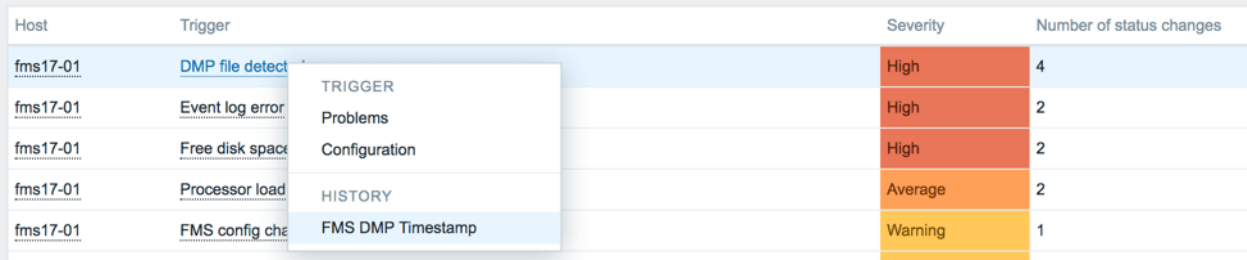


Figure 24. Open contextual menus by clicking on the host or trigger name

Audit

The Audit log is different from the other reports in that its purpose is not to inform about the monitored hosts. Instead, this report shows activity that has taken place within the web frontend. This includes information about user logins and changes made to templates, items, triggers, and other resources.

For example, we can see a list of updates made to screens.

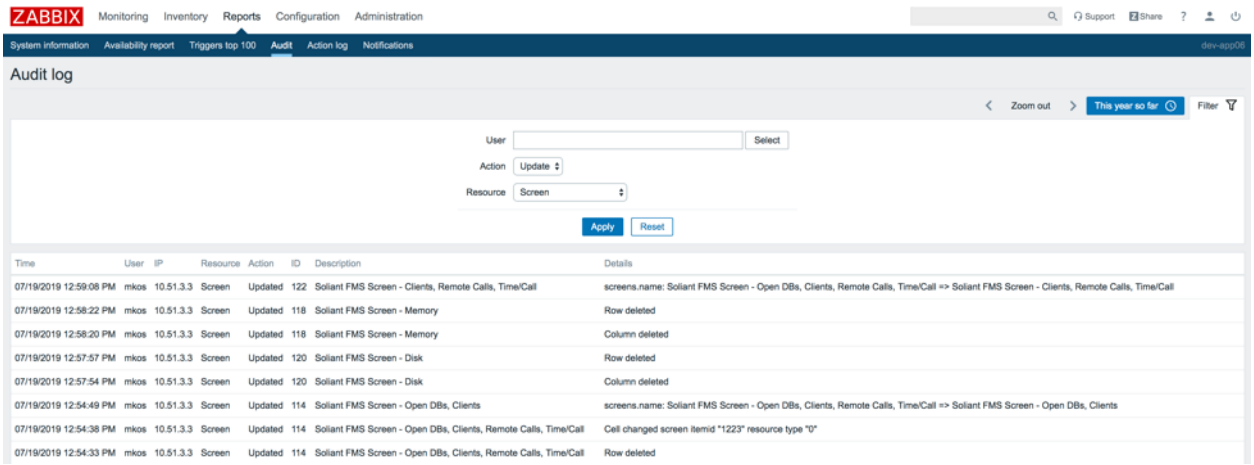


Figure 25. Filter the Audit log to view changes made to screens and other resources

In addition to the usual time period selector, the information can be filtered by the type of action and by the resource that was added, modified, or deleted.

Action Log

The Action Log displays a list of actions – notification emails or remote commands – that were taken, for example in response to triggers. The actions can be filtered by notification recipient and time period.

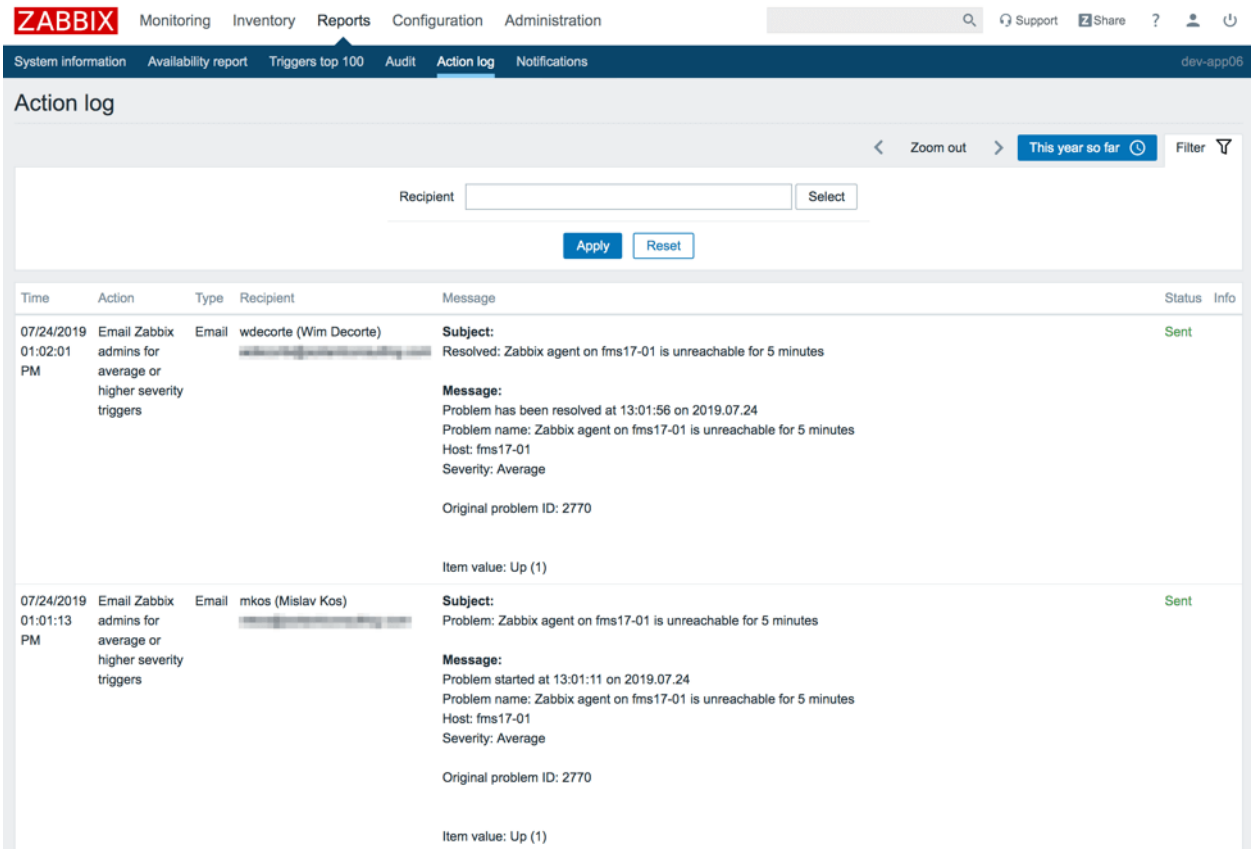
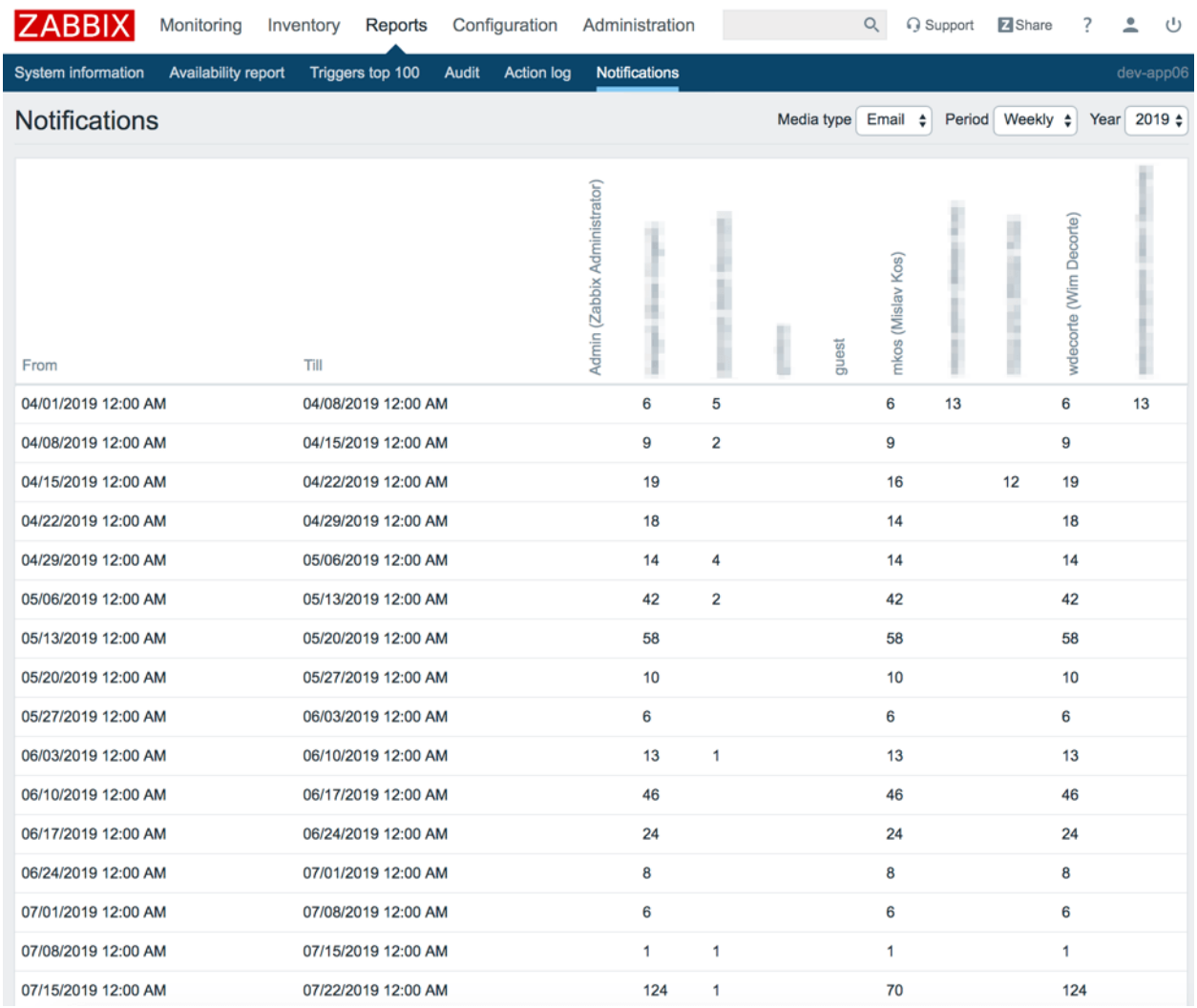


Figure 26. The Action log can be filtered by notification recipient and time period

Notifications

The notifications report displays a summary of notifications that were sent out for the selected year, summarized daily, weekly, monthly, or yearly. Each recipient will have their notification totals displayed in a separate column.



From	Till	Admin (Zabbix Administrator)		guest	mkos (Mislav Kos)			wdecorte (Wim Decorte)	
04/01/2019 12:00 AM	04/08/2019 12:00 AM	6	5		6	13		6	13
04/08/2019 12:00 AM	04/15/2019 12:00 AM	9	2		9			9	
04/15/2019 12:00 AM	04/22/2019 12:00 AM	19			16		12	19	
04/22/2019 12:00 AM	04/29/2019 12:00 AM	18			14			18	
04/29/2019 12:00 AM	05/06/2019 12:00 AM	14	4		14			14	
05/06/2019 12:00 AM	05/13/2019 12:00 AM	42	2		42			42	
05/13/2019 12:00 AM	05/20/2019 12:00 AM	58			58			58	
05/20/2019 12:00 AM	05/27/2019 12:00 AM	10			10			10	
05/27/2019 12:00 AM	06/03/2019 12:00 AM	6			6			6	
06/03/2019 12:00 AM	06/10/2019 12:00 AM	13	1		13			13	
06/10/2019 12:00 AM	06/17/2019 12:00 AM	46			46			46	
06/17/2019 12:00 AM	06/24/2019 12:00 AM	24			24			24	
06/24/2019 12:00 AM	07/01/2019 12:00 AM	8			8			8	
07/01/2019 12:00 AM	07/08/2019 12:00 AM	6			6			6	
07/08/2019 12:00 AM	07/15/2019 12:00 AM	1	1		1			1	
07/15/2019 12:00 AM	07/22/2019 12:00 AM	124	1		70			124	

Figure 27. Notifications report

Monitoring Your FileMaker Server

Maintaining Your Zabbix Server and Agents

By Wim Decorte, Senior Technical Solution Architect
and Mislav Kos, Senior Technical Project Lead
Soliant Consulting, Inc.

July 29, 2019

Table of Contents

This document is one in a series of guides that walk you through installing, configuring, and using Zabbix to monitor your FileMaker servers. The full set of guides is available at https://www.soliantconsulting.com/filemaker-zabbix.

Zabbix Server

As with any server, we recommend you keep an eye on available updates, especially the security ones for both the operating system and Zabbix.

And for Zabbix itself, since it is under active development, there are worthwhile upgrades that deliver powerful new functionality. For instance, we decided to be early adopters of Zabbix 4.2 because it offers the ability to process item data in JavaScript.

Before running a system update or Zabbix update, make sure to stop Zabbix server itself.

For both our CentOS Zabbix server and for the Ubuntu Zabbix server appliance that command is:

```
sudo systemctl stop zabbix-server
```

One reason that we favor virtual machines for servers like this is we can very easily take a full machine snapshot at this point; we have that to fall back on if any of the updates below produces an undesired result. We highly recommend you do this before running any updates.

At the end of the update process, you can use **systemctl** to start Zabbix server again or consider rebooting the machine with

```
sudo reboot now
```

Operating System

On Linux, system and software updates are typically delivered through the native package manager for the flavor of Linux. That package manager is **yum** for CentOS (the Zabbix server operating system we used in these guides) and its big brother Red Hat Enterprise Linux as well as for Oracle Linux. Some of the other supported Linux

versions such as Debian, Ubuntu and Raspbian have **apt-get** as their native package manager. SUSE Linux uses **zypper**.

If you have been following our guides, then you either have a Zabbix server running CentOS or a Zabbix server appliance running Ubuntu. If you chose another Linux version, then we are assuming you are familiar enough with its package manager to have it check for updates.

On our CentOS machine we can check for available updates by running:

```
yum check-update
```

Yum will check the various software repositories.

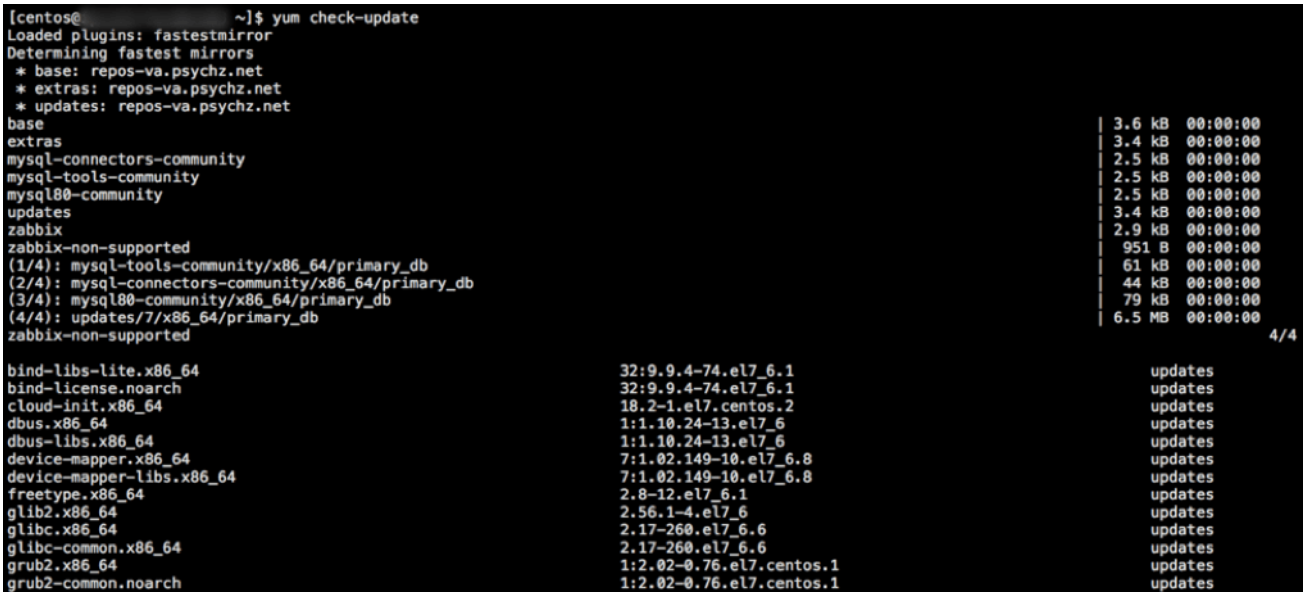


Figure 1. Yum checks the various software repositories

Linux updates are very much unlike updates to either Windows or macOS: they come in the shape of a great number of updates to individual components, not in one update that bundles all of these into convenient overall operating system versions and sub-versions.

It is unlikely that you have the time or energy to check what each of the listed updates means or assess its impact on your deployment. This is why we recommend using a virtual machine and make liberal use of its snapshotting capabilities.

While it is interesting to run the check-update command first just to get a sense of the updates, most typically you would just run

sudo yum update

This command will also show you the list of updates that are about to be installed and their combined total download size and will ask for your confirmation to proceed:

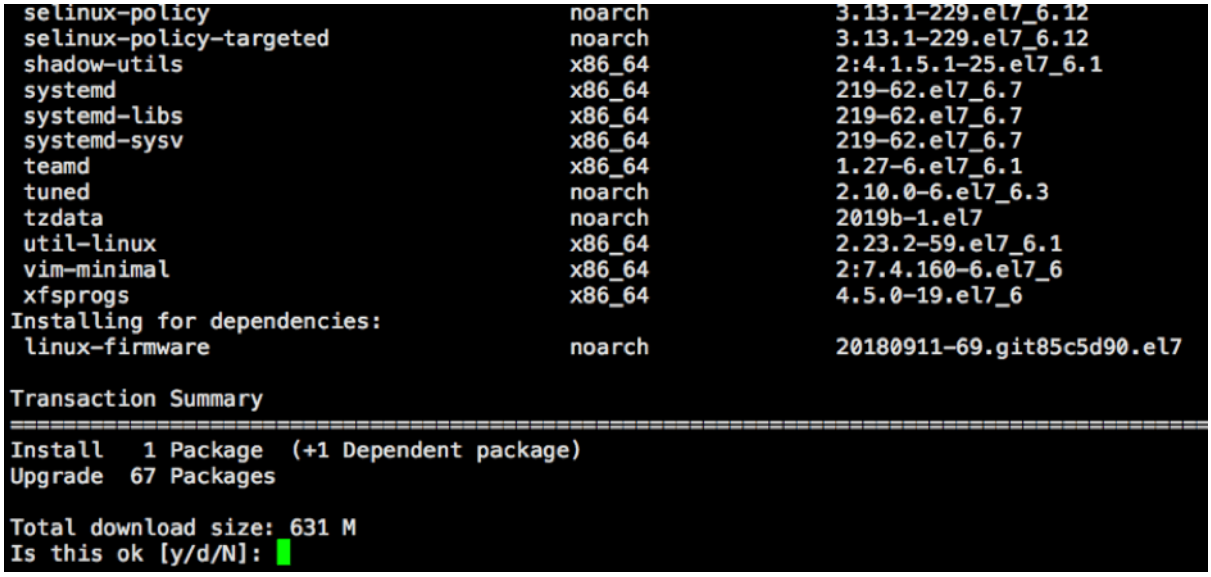


Figure 2. Type 'Y' to proceed with the update

If you scroll through the list and see Zabbix items or MySQL items (when that is your chosen database for Zabbix), and you have not run the snapshot backup, consider backing out at this point and running the snapshot.

On the Ubuntu Zabbix appliance, here are two commands to run:

sudo apt-get update

To update the list of software packages and

sudo apt-get upgrade

To update the actual software installed on your server.

Zabbix software

Updates to Zabbix server within the same major version are also done through the native package manager.

For CentOS that would look like this:

sudo yum update 'zabbix-*'

And for the Ubuntu Zabbix server appliance:

sudo apt-get upgrade 'zabbix-*'

To upgrade Zabbix server between major versions, follow the procedure as outlined here:

https://www.zabbix.com/documentation/4.2/manual/installation/upgrade

Note that when you land on the Zabbix documentation web site, you can pick the version of the documentation that is relevant to you in the header of the page. At the time of writing this, this page would show the instructions of upgrading to Zabbix 4.2.

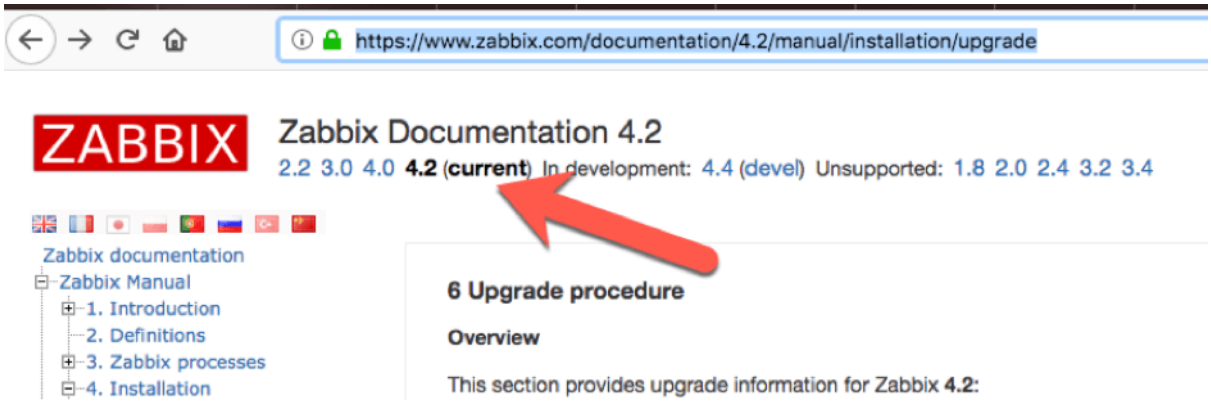


Figure 3. Zabbix Documentation version shown as 4.2

Zabbix Agents

Zabbix agents do not need to be on the same exact version number as the Zabbix server that they interact with. However, we still recommended you keep an eye on the agent releases to determine if you need or want that particular update.

Windows

Updating the Zabbix agent on Windows is as easy as downloading the new installer from the Zabbix download page. Before you run the installer though, make sure to stop the Zabbix agent service in the Windows Services Control Panel. Running the installer will not overwrite the changes made earlier to the agent configuration file.

When the installer is finished, check back in the Services Control Panel to make sure the agent is running again. If not, start it from there.

macOS

Here too the updater comes in the form of a new installer. First, though, you want to stop the agent from the Terminal:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

If the installer does not start the agent again when it is done, use the same command but with **start** instead of **stop** to launch it.

You can check whether the Zabbix agent is running with the **Activity Monitor**. Set View to show all processes and filter the list by entering part of the Zabbix name:

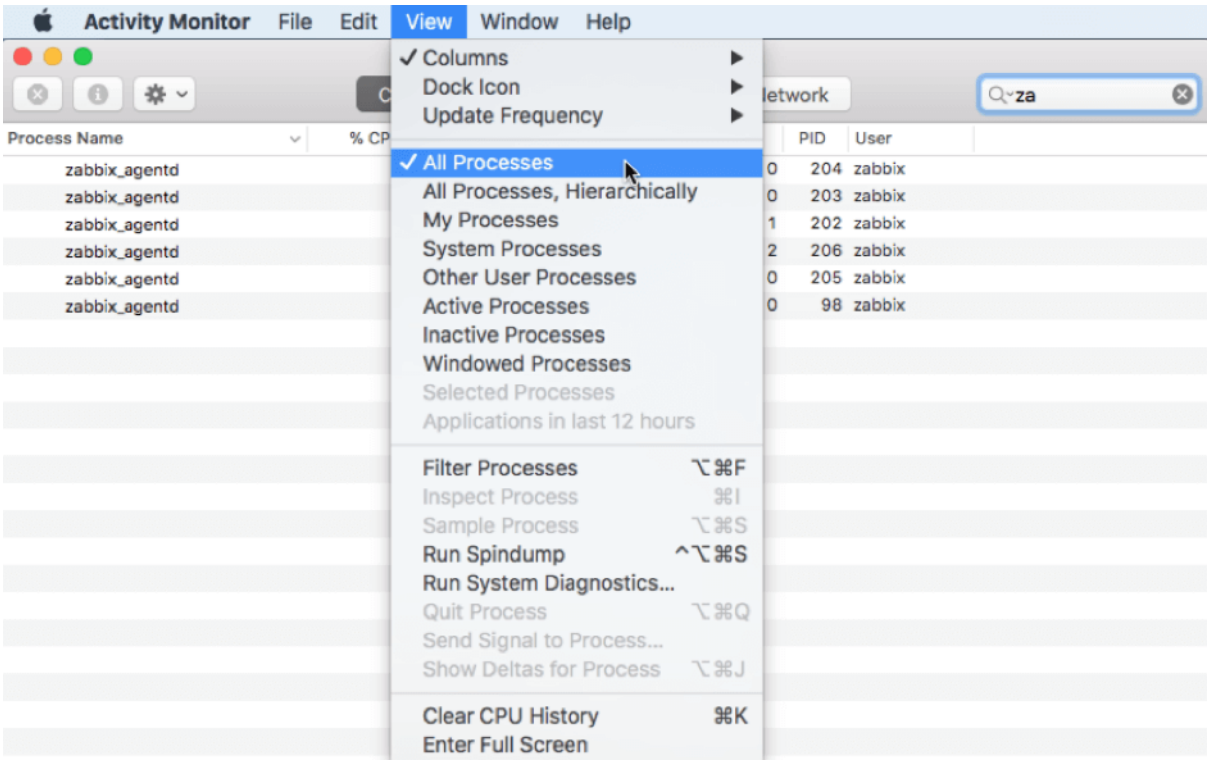


Figure 4. Use the Activity Monitor to check if the Zabbix agent is running

FileMaker Cloud (CentOS)

Given that FileMaker Inc. is in control of the operating system part of your FileMaker Cloud installation, you do **NOT** want to run any operating system updates, as that could render FileMaker Cloud inoperable.

If you want to upgrade the Zabbix agent, then run these three commands:

sudo systemctl stop zabbix-agent

sudo yum update 'zabbix-agent-*'

sudo systemctl start zabbix-agent

You can check whether the Zabbix agent is running by using this command:

sudo ps -u zabbix -o state,comm

Figure 5. Check whether the Zabbix agent is running by entering the sudo command shown above

This asks Linux to list all the processes (ps) that are owned by user Zabbix and return its state and what the command or executable is that created the process (comm). If the lines start with R, D or S, then all is well.

In our Zabbix template we use this type of command to check on all the running processes that make up FileMaker Server.