

CheckMK - Open Source Network, Server, and Machine Monitoring

<https://www.youtube.com/embed/7OnhuCsR7jg>

CheckMK is a network and systems monitoring solution that can span from a small home user's needs into the enterprise with ease. There are many of these solutions out there, and all of them have their pros and cons, but for many of them, the con is that they are complicated to setup, maintain, and keep running.

Thus far, this is where CheckMK seems to be a little different. The setup is pretty straight forward, their documentation is outstanding, and the maintenance of the system seems to be quite easy.

If you've been looking for a monitoring solution for your various devices and networks, then this may just be the solution for you. As always, CheckMK is open source, and available in a free (community / raw) version, but the open source software is supported by the option to have more functionality / capability with their professional and enterprise level offerings as well. So, if you like CheckMK, and think it would be a good fit for your business, consider contributing to open source by giving them some business with their paid tiers as well.

What You'll Need

- Either a Server (I Used and LXC Container on Proxmox) or Docker and Docker-Compose
- If you want to use this across the cloud, then you'll also want a VPN between your various networks, or a domain name that points to the public IP of the network where you'll run CheckMK, and a reverse proxy like NGINX-Proxy-Manager.
- About 20 minutes of your time to get started.

Installation

In order to install CheckMK on Docker, jump down to the "Installing On Docker" section below. If you are installing on a regular Linux based server, then keep reading.

There are a pretty basic set of steps to get this installed actually. I used Ubuntu, but if you are using a different base distro, like Fedora, Cnetos, etc, then you can check [their instructions here](#).

Installing on Ubuntu

Make sure you're logged in with a non-root user that has sudo privileges. If you don't know what this means, essentially, make a new user who's not root, then give that user sudo privileges by adding them to the sudo group.

```
adduser < your user name >
```

Enter the password, and re-type it when prompted.

After completing the `adduser` portion, add the user to the "sudo" group with:

```
usermod -aG sudo < your user name >
```

Now logout from the root account, and login with your new user account.

If you need to run something with elevated privileges, just put `sudo` in front of the command. You'll be prompted for your user's password occasionally in order to run anything as "sudo".

To find the version for your distro and release (mine is Ubuntu 20.04 LTS), gotto <https://checkmk.com/de/download> and select through the options down the page to filter down to your needs.



Below that, you'll find a "wget" command, which is perfect for downloading a package to your server directly when logged in via SSH.

In my case the command is:

```
wget https://download.checkmk.com/checkmk/2.1.0p8/check-mk-raw-2.1.0p8_0.focal_amd64.deb
```

Once downloaded, we'll install it using the command:

```
sudo apt install ./check-mk-raw-2.1.0p8_0.focal_amd64.deb
```

Let the installer run, and you should get some output toward the end showing that the system is installed. Afterward, check that it's installed by running:

```
omd version
```

You should get a version number for the CheckMK you just installed.

Next, you need to create a "Monitoring" site. You can name the site anything, but we'll stick with what their documentation says, and call it "monitoring":

```
omd create monitoring
```

Note, you may need `sudo` for this step.

Once the monitoring site is created, you'll get a default user of `cmkadmin`, and a password that is autogenerated. Make sure to copy the password somewhere so you don't lose it. You can change it via the web ui later.

Now, we need to start our site into a running state.

```
omd start monitoring
```

Again, you may need to use `sudo` in front of that command.

You should see output similar to this:

```
Starting mkeventd...OK
Starting liveproxysd...OK
Starting mknotifyd...OK
Starting rrdcached...OK
Starting cmc...OK
Starting apache...OK
Starting dcd...OK
Starting redis...OK
Initializing Crontab...OK
```

You should now be able to access your CheckMK server via your favorite modern web browser at the IP address of your host system, and the location `/cmk`:

```
http://192.168.10.42/cmk
```

 was the address I used.

Installing on Docker

If you are like me, and you prefer to containerize anything and everything due to the simplicity of upkeep, updates, and maintenance, then you'll be more interested in running CheckMK in docker. I went through their instructions, and took a bit from here and there, and decided I wanted to create a nice docker-compose file instead of using their docker run, so that's what we'll be using. Before that, a quick section on installing Docker and Docker-Compose if you don't already have it. If you do, then skip down to the 'Installing CheckMK in Docker' section below.

Installing Docker-CE and Docker-Compose

If you already have Docker and Docker-Compose installed, feel free to skip down to the next section.

You may want to install some pre-requisite software as well:

Debian / Ubuntu

```
sudo apt install git curl wget
```

Fedora / Redhat

```
dnf install git curl wget
```

Arch

```
sudo pacman -Sy git curl wget
```

You can easily install Docker-CE, Docker-Compose, Portainer-CE, and NGinX Proxy manager by using this quick install script I created and maintain on Github. Just use the command:

```
wget https://gitlab.com/bmcgonag/docker\_installs/-/raw/main/install\_docker\_nproxyman.sh
```

To download the script to your desired host.

Change the permissions to make the script executable:

```
chmod +x ./install_docker_nproxyman.sh
```

and then run the script with the command:

```
./install_docker_nproxyman.sh
```

When run, the script will prompt you to select your host operating system, then will ask you which bits of software you want to install.

Simply enter 'y' for each thing you want to install.

For instance, you may want to answer 'y' to NGinX Proxy Manager, and Portainer-CE if you don't already use these in your system.

At some point, you'll be asked for your super user (sudo) password as well.

Allow the script to complete installation.

At this point, you might want to log out and back in, as this will allow you to use the `docker` and `docker-compose` commands without the need of `sudo` in front of them.

Installing CheckMK on Docker

I like to keep all of my docker containers in a certain folder structure. Make a "docker" folder, then inside that make a new "checkmk" folder.

```
mkdir -p docker/checkmk
```

Next, move into the checkmk folder we just created, and make a new file called "docker-compose.yml"

```
cd docker/checkmk
```

```
nano docker-compose.yml
```

Now paste the contents of the code-block below into that file:

```
version: '3.3'
services:
  check-mk-raw:
    ports:
      - '8080:5000'
    tmpfs: '/opt/omd/sites/cmktmp:uid=1000,gid=1000'
    volumes:
      - './monitoring:/omd/sites'
      - '/etc/localtime:/etc/localtime:ro'
    container_name: monitoring
    restart: always
    image: 'checkmk/check-mk-raw:2.0.0-latest'
```

You may want to change the left side of the port mapping in the file from 8080 to another port if 8080 is already in use on your host machine. Remember to only change the left side, leave the 5000 alone.

When done, save the file with CTRL + O, press Enter to confirm, and then exit the nano text editor with CTRL + X.

Now, we'll pull down CheckMK and start the container with the command:

```
docker-compose up -d && docker-compose logs -f
```

This command starts the container running first, then shows us log output as the container is started. We can watch for any startup errors.

Additionally, by watching the logs, we can see the admin username `cmkadmin` and the default password create for our admin user in the logging. We'll need this to login for the first time.

As long as there are no errors you can get out of the logs view with CTRL + C.

Now, go to your host machine's IP and port 8080 (unless you changed it in the docker-compose file above, then use the port you set).

I used `http://192.168.10.112:8080`.

Now log in, and get ready to setup your Folders and host machines.

For more on how to setup a host machine (client) and the folder structure see the CheckMK docs as well as the video linked at this top of this post.

Support my Channel and Content

Support my Channel and ongoing efforts through Patreon:

<https://www.patreon.com/bePatron?u=234177>

Revision #1

Created 30 September 2022 16:16:38 by Brian McGonagill

Updated 30 September 2022 16:17:48 by Brian McGonagill