

Installing Docker

Docker is a service that allows you to quickly host containerized applications tailored for your hardware.

- [Running the Install Script](#)
- [Graphic Card Integration](#)
- [Creating Your First Container](#)

Running the Install Script

[Docker](#) provides a convenient script for quickly installing their software and getting it ready.

You can find the [official Docker installation guide](#) on their website.

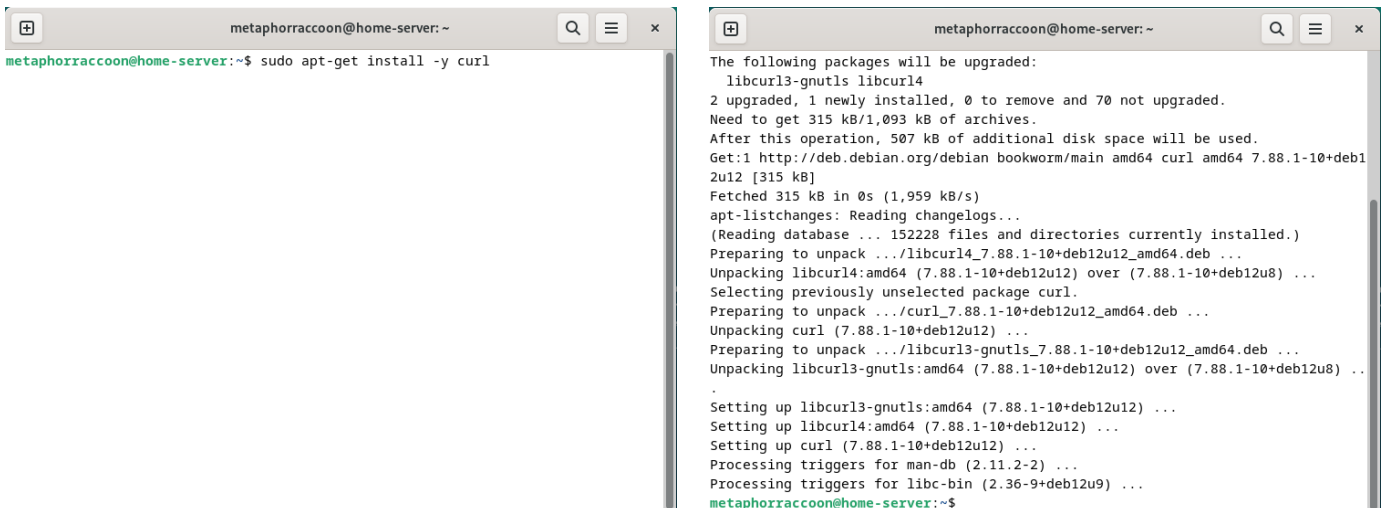
Preparation

First, we need to open our terminal to install a small application called *curl* that can be used to download files through the terminal.

Enter this command and hit enter to automatically install the curl software:

```
sudo apt-get install -y curl
```

You will be prompted to enter your password.

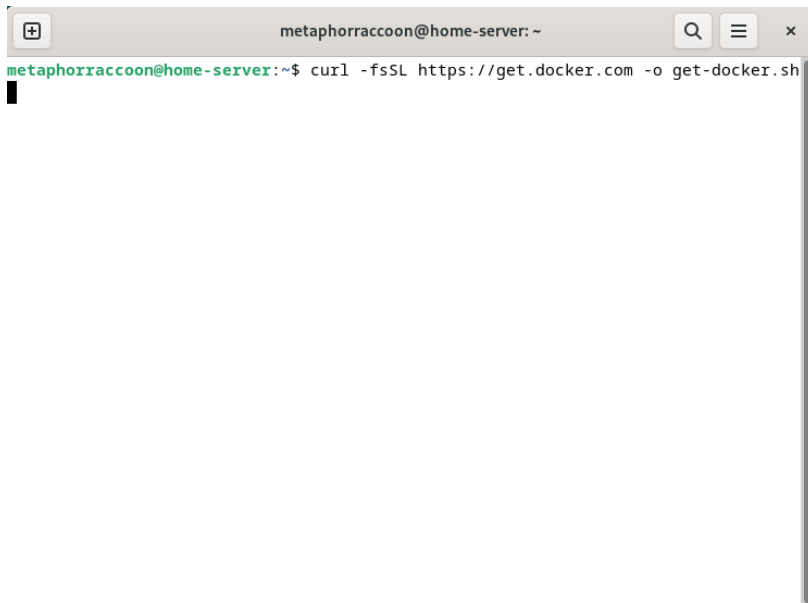


```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ sudo apt-get install -y curl  
  
The following packages will be upgraded:  
  libcurl3-gnutls libcurl4  
2 upgraded, 1 newly installed, 0 to remove and 70 not upgraded.  
Need to get 315 kB/1,093 kB of archives.  
After this operation, 507 kB of additional disk space will be used.  
Get:1 http://deb.debian.org/debian bookworm/main amd64 curl amd64 7.88.1-10+deb12u12 [315 kB]  
Fetched 315 kB in 0s (1,959 kB/s)  
apt-listchanges: Reading changelogs...  
(Reading database ... 152228 files and directories currently installed.)  
Preparing to unpack .../libcurl4_7.88.1-10+deb12u12_amd64.deb ...  
Unpacking libcurl4:amd64 (7.88.1-10+deb12u12) over (7.88.1-10+deb12u8) ...  
Selecting previously unselected package curl.  
Preparing to unpack .../curl_7.88.1-10+deb12u12_amd64.deb ...  
Unpacking curl (7.88.1-10+deb12u12) ...  
Preparing to unpack .../libcurl3-gnutls_7.88.1-10+deb12u12_amd64.deb ...  
Unpacking libcurl3-gnutls:amd64 (7.88.1-10+deb12u12) over (7.88.1-10+deb12u8) ...  
.  
Setting up libcurl3-gnutls:amd64 (7.88.1-10+deb12u12) ...  
Setting up libcurl4:amd64 (7.88.1-10+deb12u12) ...  
Setting up curl (7.88.1-10+deb12u12) ...  
Processing triggers for man-db (2.11.2-2) ...  
Processing triggers for libc-bin (2.36-9+deb12u9) ...  
metaphorraccoon@home-server:~$
```

Installation

Now, enter the following command to use *curl* to download the script that installs Docker and its dependencies:

```
curl -fsSL https://get.docker.com -o get-docker.sh
```



```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ curl -fsSL https://get.docker.com -o get-docker.sh
```

Once the download is complete, you can enter this command to run the script we just downloaded:

```
sh get-docker.sh
```



```
metaphorraccoon@home-server: ~  
metaphorraccoon@home-server:~$ sh get-docker.sh
```

The Docker setup process should proceed automatically and prompt you about further actions once it's finished. We do not need to follow these additional instructions.

```
metaphorraccoon@home-server: ~
docker-init:
  Version:      0.19.0
  GitCommit:   de40ad0
=====

To run Docker as a non-privileged user, consider setting up the
Docker daemon in rootless mode for your user:

  dockerd-rootless-setuptool.sh install

Visit https://docs.docker.com/go/rootless/ to learn about rootless mode.

To run the Docker daemon as a fully privileged service, but granting non-root
users access, refer to https://docs.docker.com/go/daemon-access/

WARNING: Access to the remote API on a privileged Docker daemon is equivalent
to root access on the host. Refer to the 'Docker daemon attack surface'
documentation for details: https://docs.docker.com/go/attack-surface/
=====

metaphorraccoon@home-server:~$
```

We have successfully installed the Docker Container Engine on our server.

Graphic Card Integration

Graphic cards handle images and videos much more efficiently than a traditional processor. While using Docker, some media services – such as [Plex](#), [Jellyfin](#) or [Stash](#) – will run much more efficiently if they have direct access to your graphics card.

This process is different depending on the manufacturer of your graphics card.

AMD/ATI

You will need to install the ROCm software in order to use your graphics card within Docker. The process for [installing ROCm on Debian](#) is detailed in their documentation.

Once installed, the graphics card can be attached to Docker containers.

Intel

Integrated graphics, like those in modern Intel processors, are innately available to Docker containers by providing access.

Nvidia

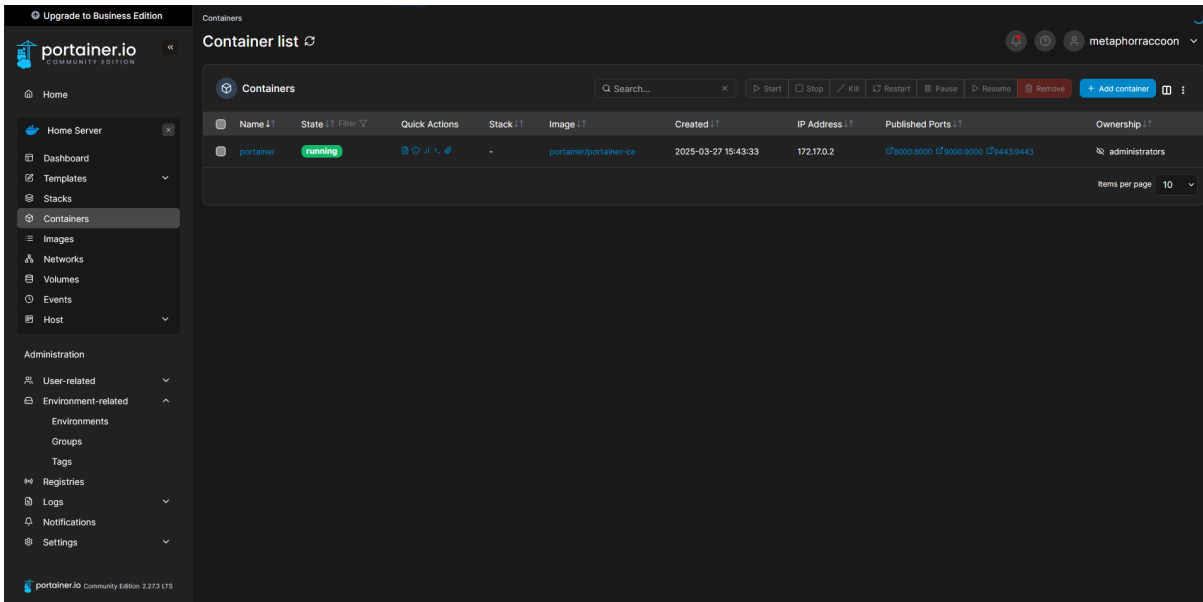
You will need to install the [Nvidia Container Toolkit](#) in order to use your graphics card with Docker containers.

This requires that the [proprietary Nvidia drivers are installed](#) and working correctly.

The process for [installing the required software](#) is detailed in their documentation. Once installed, the graphics card can be attached to Docker containers.

Creating Your First Container

We will be installing [Portainer](#), an open-source web application that makes it easy to manage Docker from your browser.



Starting Portainer

We are adding port 9000 which isn't used by default. This allows Portainer to be accessed through HTTP instead of HTTPS.

You can find the [official Portainer installation guide](#) on their website.

Using the terminal, we will issue the command to start our new Docker container:

```
sudo docker run -d -p 8000:8000 -p 9000:9000 -p 9443:9443 --name=portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v /srv/portainer:/data portainer/portainer-ce
```

Docker will download the image and start the container.

```
metaphorraccoon@development:~$ sudo docker run -d --name=portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v /srv/portainer:/data portainer/portainer-ce
Unable to find image 'portainer/portainer-ce:latest' locally
latest: Pulling from portainer/portainer-ce
e2e06b27b87e: Pull complete
1fed1531b45b: Pull complete
04de093ad5ed: Pull complete
86a7cce72d42: Pull complete
e09df2601140: Pull complete
eae3ebf29ea8: Pull complete
c12aa3fbd31a: Pull complete
f111bda3f9a6: Pull complete
81021110ed01: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:7f10a26bfdda3fc58295ea09b860117ecd86a642d66fb94ce1f27a4c221d4649
Status: Downloaded newer image for portainer/portainer-ce:latest
135058bbd3756f850fc6c30f22c76dd3a811d9c1f63d06caf7396df474a61a2
metaphorraccoon@development:~$
```

Once the command has finished,

we can verify that Portainer is operational by querying Docker's running containers:

```
sudo docker ps
```

```
metaphorraccoon@home-server:~$ sudo docker ps
[sudo] password for metaphorraccoon:
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS
US            PORTS                                NAMES
c14737e6a097  portainer/portainer-ce             "/portainer"           About a minute ago Up A
bout a minute  0.0.0.0:8000->8000/tcp, [::]:8000->8000/tcp, 0.0.0.0:9000->9000/
tcp, [::]:9000->9000/tcp, 0.0.0.0:9443->9443/tcp, [::]:9443->9443/tcp  portaine
r
metaphorraccoon@home-server:~$
```

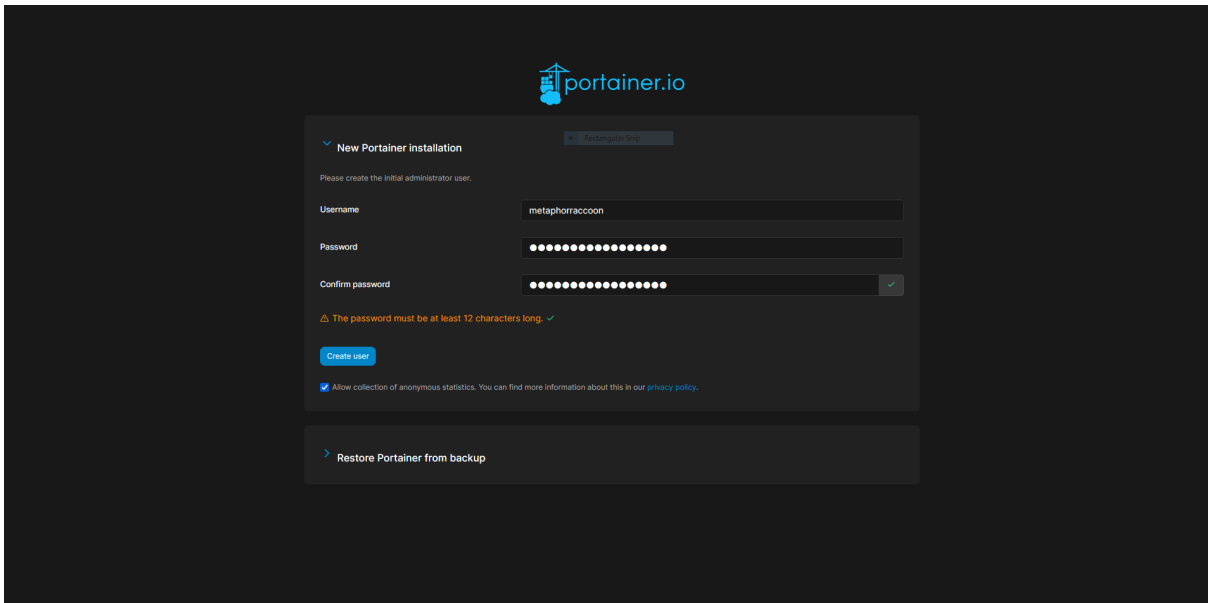
The command should return our running Portainer container and show that it's available under four different ports.

Creating an Admin Account

In order to configure our instance of Portainer, we will need to open a web browser.

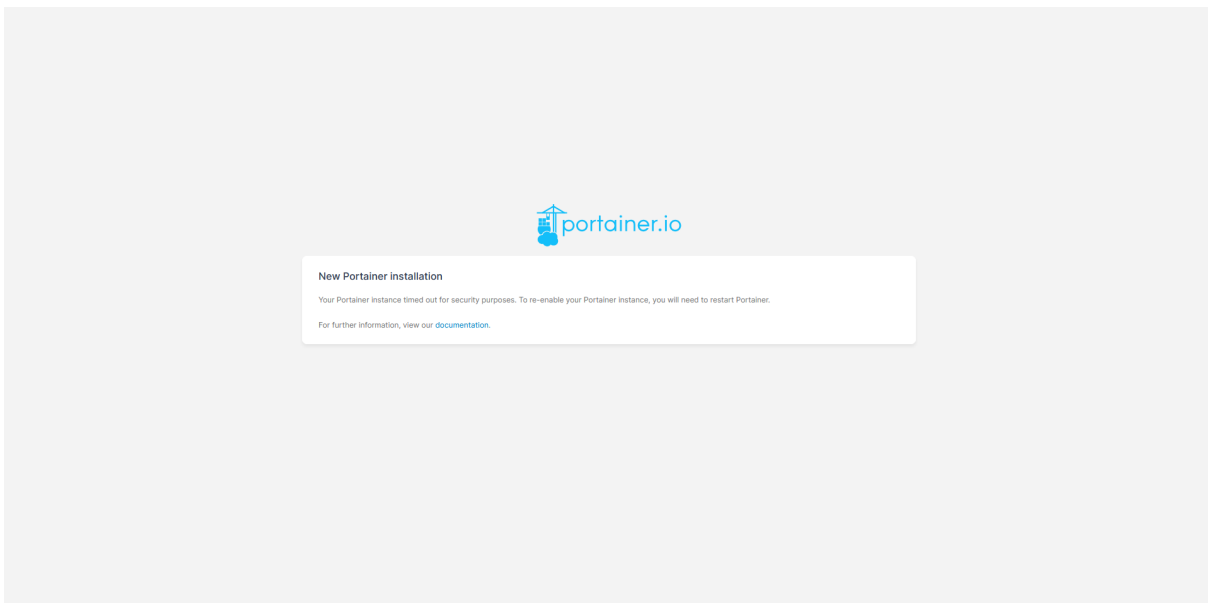
We recommend [Mozilla Firefox](#), but [Google Chrome](#) will also work.

Once the web browser is open, go to <http://localhost:9000/> to load Portainer.



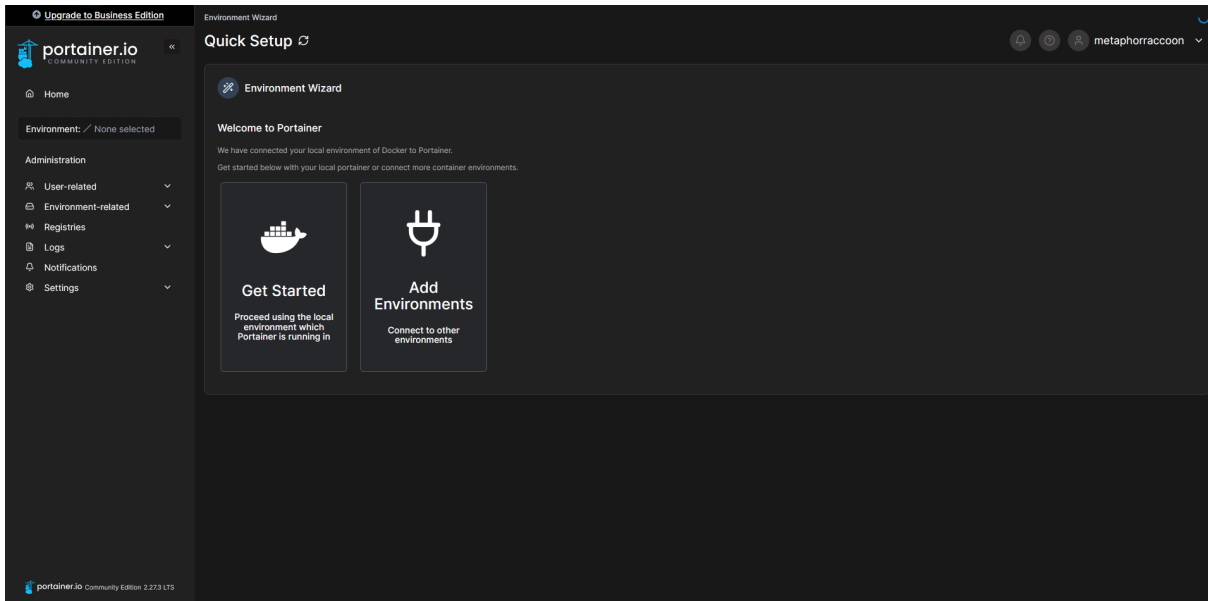
Here, we are instructed to create an administrator account and set a secure password. This will be used in the future to login to Portainer.

The account creation wizard will time-out if not completed quickly.
If this happens, restart Portainer with the command: `docker restart portainer`.



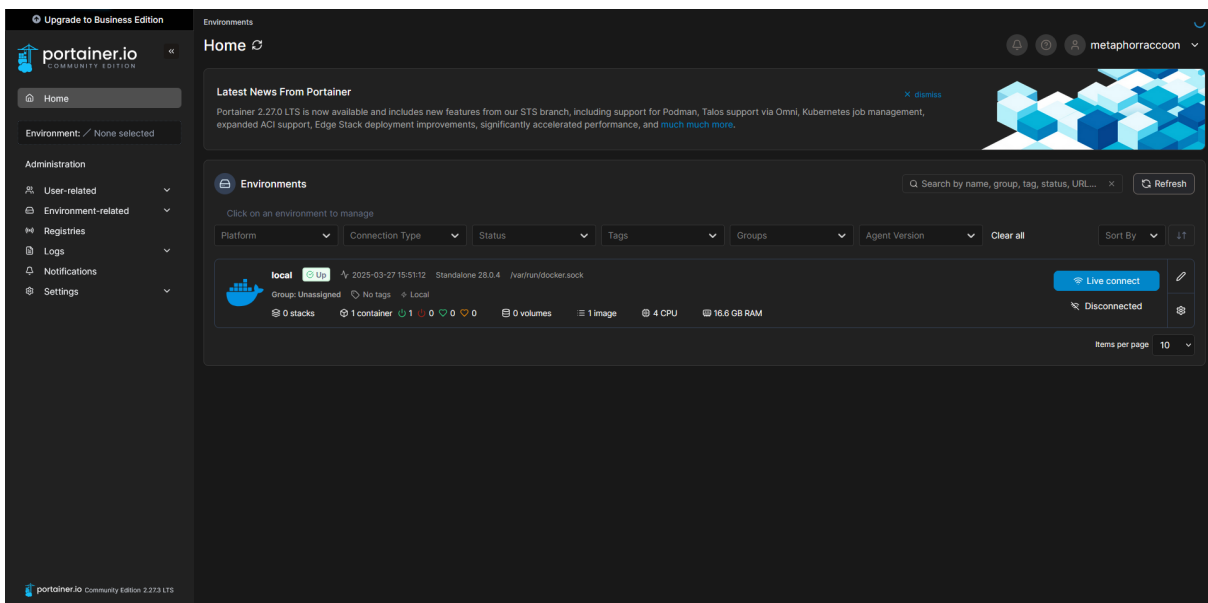
Adding Docker

Once we have created an account, we will be able to select our Docker service. If everything is working as expected, you should be able to click 'Get Started' to use the local Docker service.

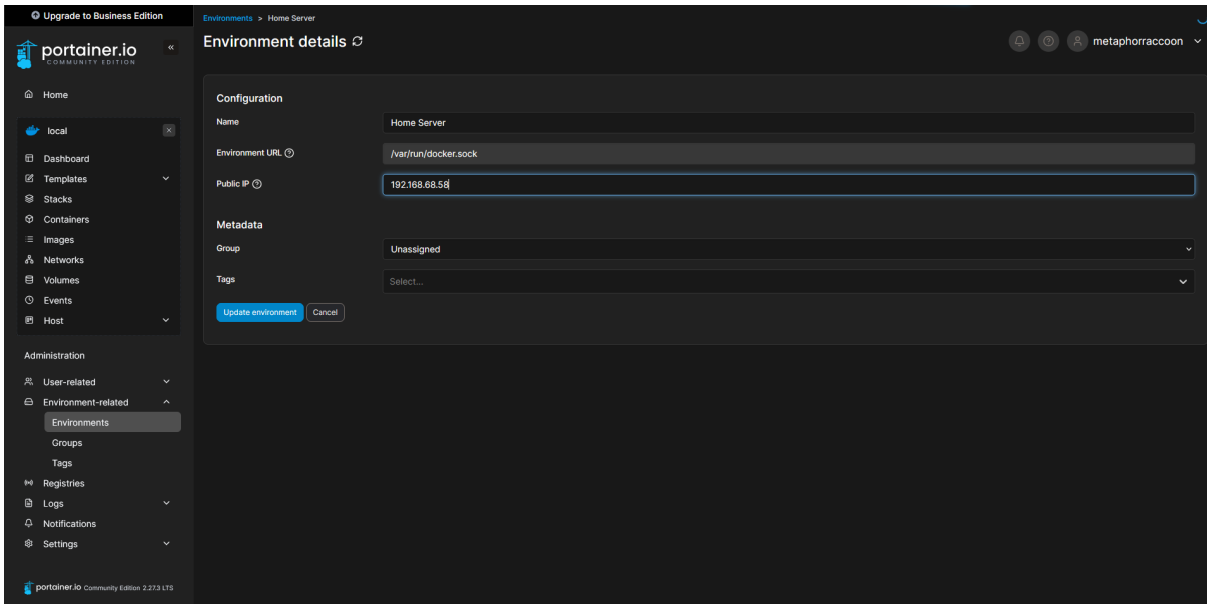


This will bring you to a list of Docker installations that are available to Portainer.

You can use the [Portainer Agent](#) to control Docker on additional servers.

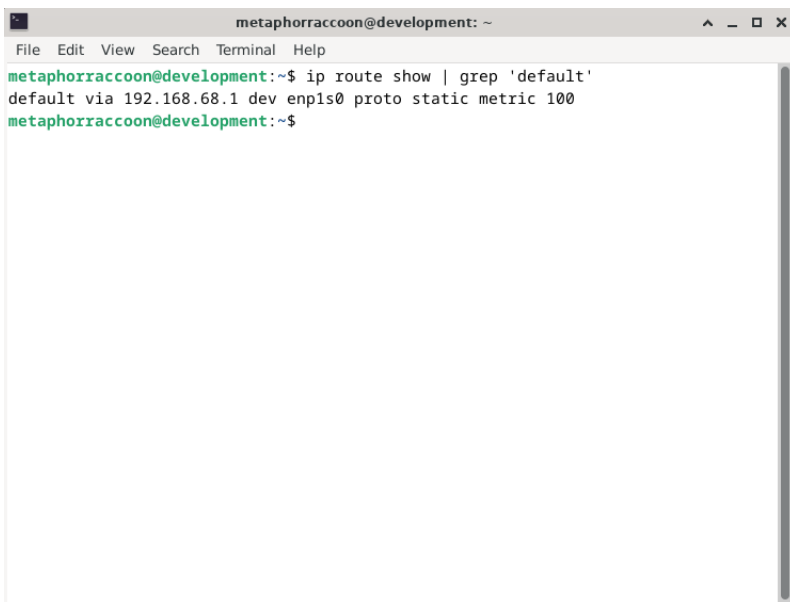


Now, we can configure the name of our Docker server, for personal preference, as well as set the root IP address of our server so we can quickly access our services from Portainer. Select the pencil icon next to your "local" Environment.

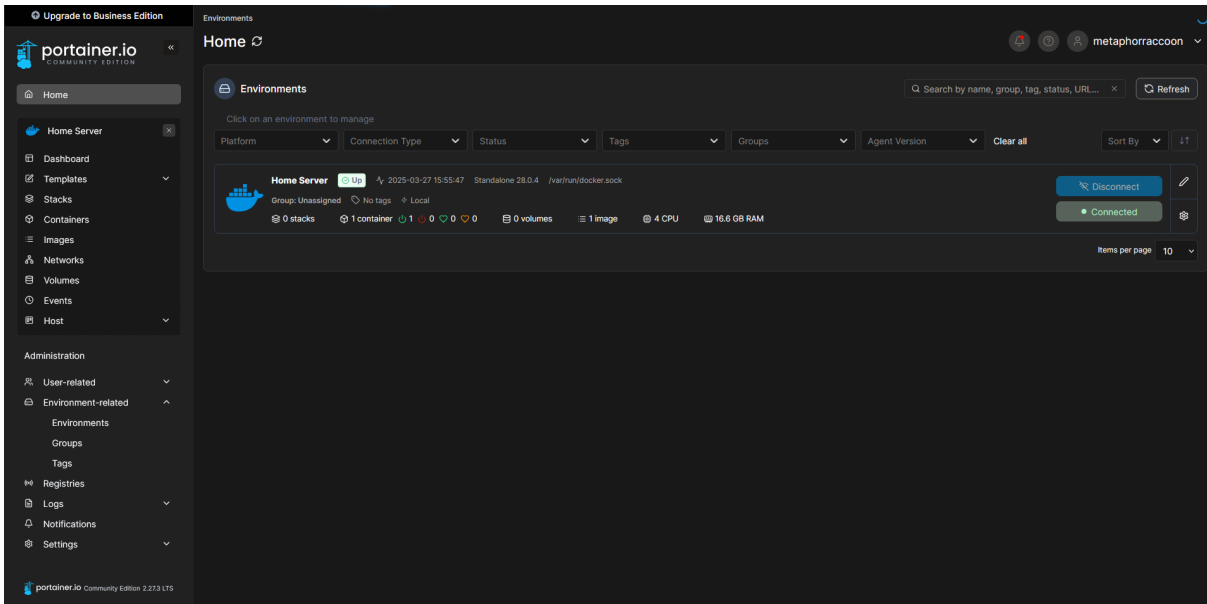


If you are unsure about the LAN IP address [we selected earlier](#), you can run the command below to obtain it:

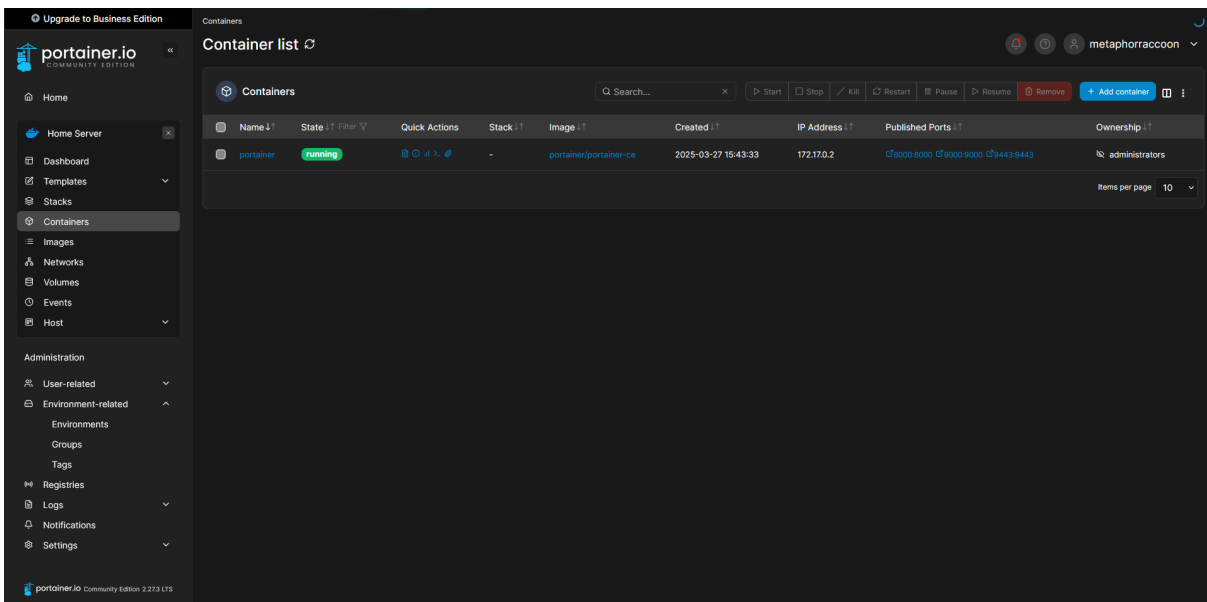
```
ip -o route show | grep default
```



Once you have completed that, you can save the settings to apply them. When returning to the home screen, we can see that our settings were updated.



Clicking on our Docker environment - named "Home Server" here - will pull up a list of our running containers. You can see the first and only one listed is our newly created Portainer installation.



Clicking on "8000:8000" under assigned ports will open this port using your web browser. In this instance, it should open Portainer in a new tab.

- Dashboard
- Templates
- Stacks
- Containers
- Images
- Networks
- Volumes
- Events
- Host

Administration

- User-related
- Environment-related
 - Environments
 - Groups
 - Tags
- Registries
- Logs
- Notifications
- Settings

Environments

Search by name, group, tag, status, URL... Refresh

Click on an environment to manage

Platform Connection Type Status Tags Groups Agent Version Clear all Sort By

Home Server 2025-03-27 15:55:47 Standalone 28.0.4 /var/run/docker.sock

Group: Unassigned No tags Local

0 stacks 1 container 1 0 0 0 0 volumes 1 Image 4 CPU 16.6 GB RAM

Disconnect Connected

Items per page 10