Manjaro Linux Docker installation

linuxconfig.org/manjaro-linux-docker-installation

30 June 2020

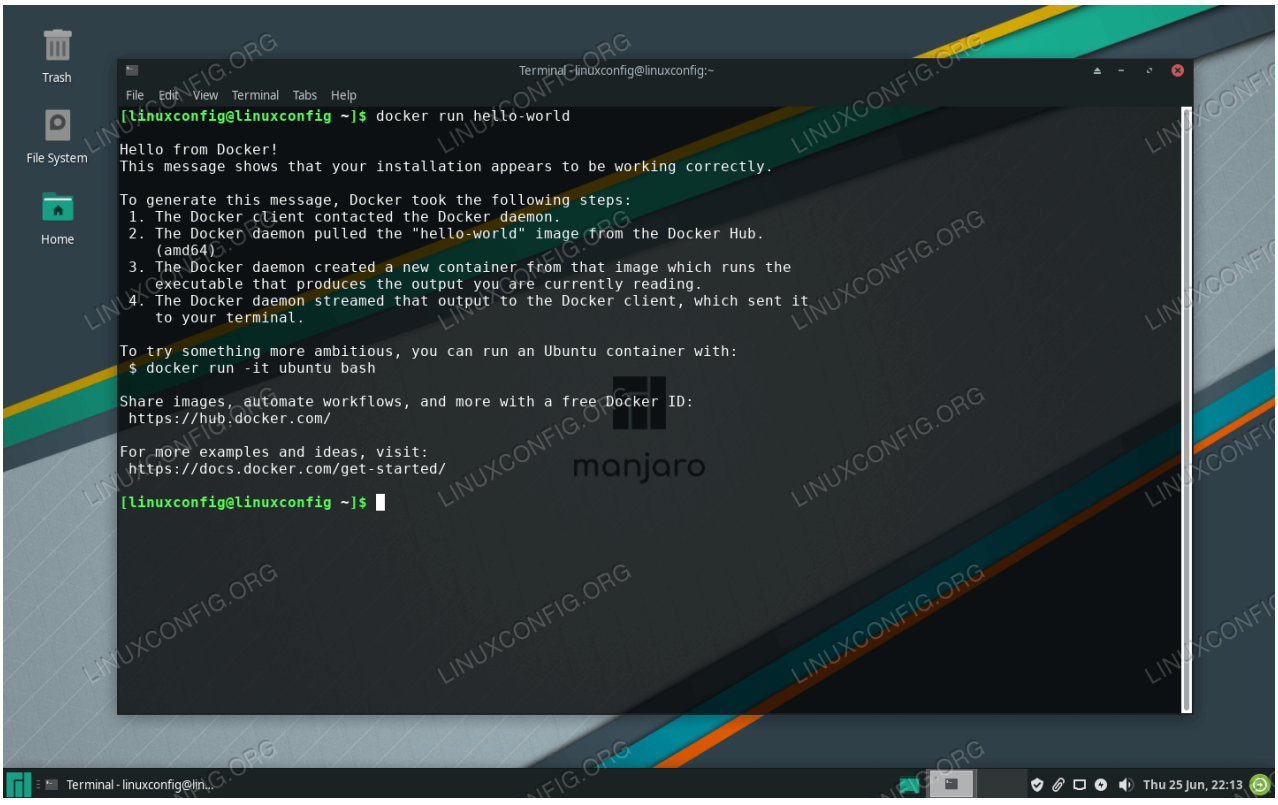
Docker is a tool that is used to run software in a container. It's a great way for developers and users to worry less about compatibility with an operating system and dependencies because the contained software should run identically on any system.

Docker is available for download and installation on Manjaro as well as most other distributions of Linux. After Docker is installed, you can use it to install software packages much the same way you would use your distro's package manager to download an app. The difference of using Docker is that everything is more automated, with compatability and dependencies no longer being potential issues.

In this guide, we'll show you how to install Docker on Manjaro Linux and get started with installing containerized software.

In this tutorial you will learn:

- How to install Docker
- How to run Docker without root
- How to search for a Docker image
- How to install a Docker image
- How to run a Docker image
- How to monitor Docker with various commands



Docker running a container image

Category	Requirements, Conventions or Software Version Used
System	Manjaro Linux
Software	Docker
Other	Privileged access to your Linux system as root or via the `sudo` command.
Conventions	**#** – requires given linux commands to be executed with root privileges either directly as a root user or by use of `sudo` command **$** – requires given linux commands to be executed as a regular non-privileged user

Software Requirements and Linux Command Line Conventions

Install Docker

To get started installing Docker, the first thing you should do is open a terminal and make sure that Manjaro is up to date.

```
$ sudo pacman -Syu
```

Next, execute the following command to install Docker:

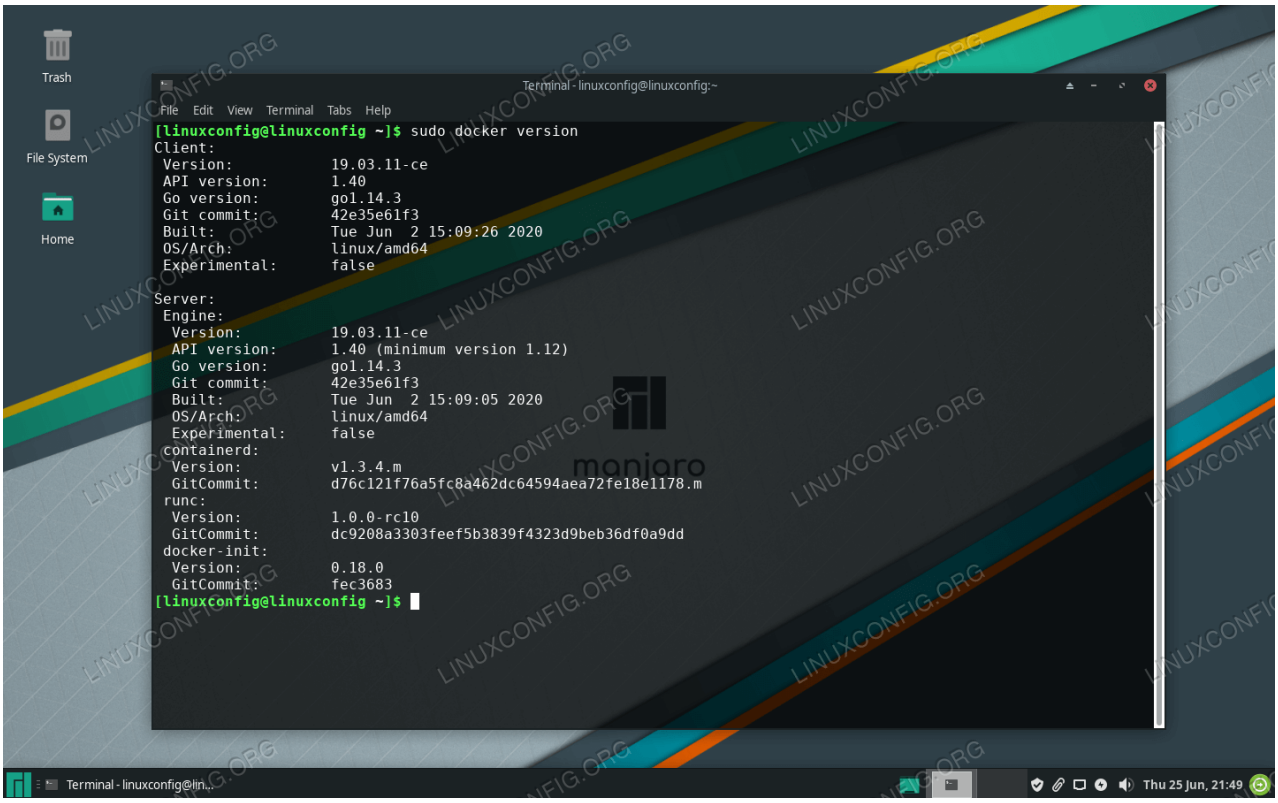
```
$ sudo pacman -S docker
```

DID YOU KNOW?
You can also install the development version of Docker from the AUR, but it's not recommended to use that release in a production environment.
Once installation is completed, start the Docker service and, optionally, enable it to run whenever the system is rebooted:

```
$ sudo systemctl start docker.service
$ sudo systemctl enable docker.service
```

You can verify that Docker is installed and gather some information about the current version by entering this command:

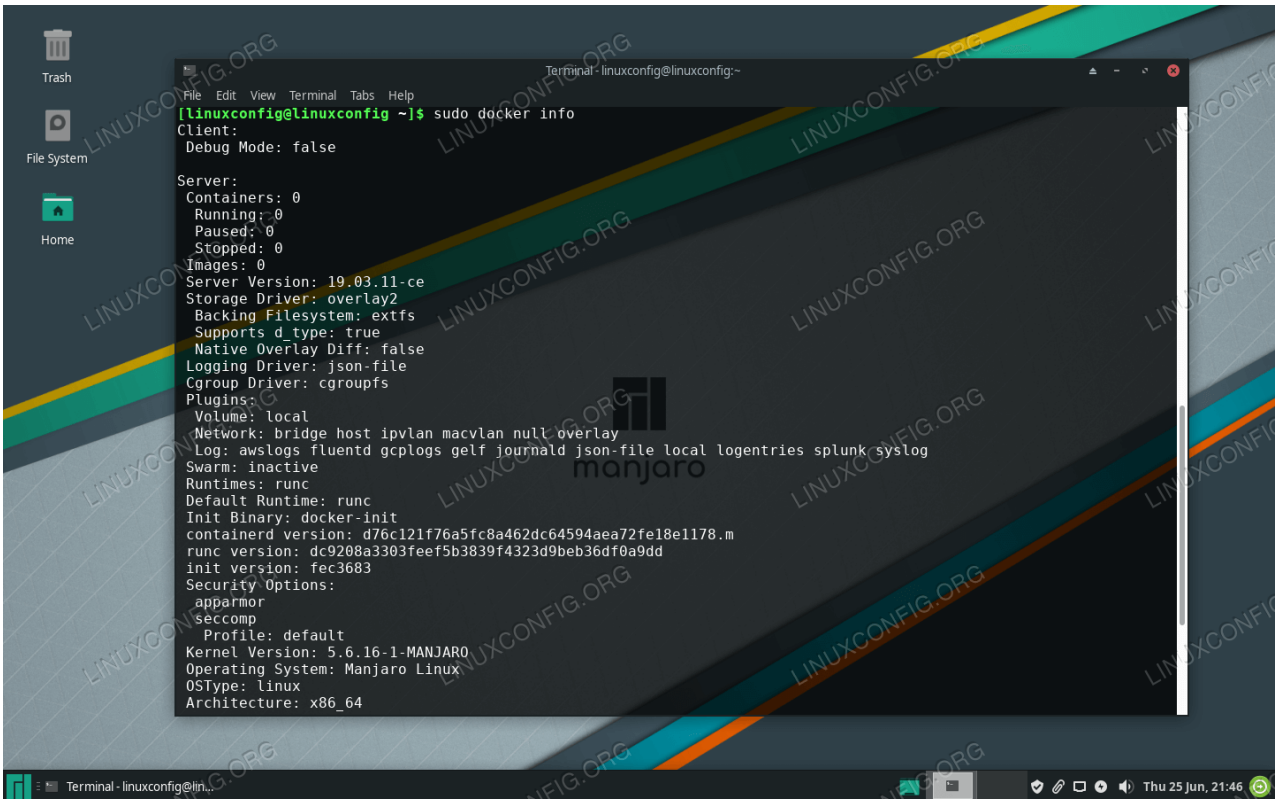
```
$ sudo docker version
```



Output of docker version command

There is also a quick and easy way to see how many Docker containers are currently running and see some of Docker's configured options by entering:

```
$ sudo docker info
```



Output of docker info command showing its configuration

Run Docker without root

By default, you'll have to use `sudo` or login to root anytime you want to run a Docker command. This next step is optional, but if you'd prefer the ability to run Docker as your current user, add your account to the `docker` group with this command:

```
$ sudo usermod -aG docker $USER
```

You'll need to reboot your system for those changes to take effect.

```
$ reboot
```

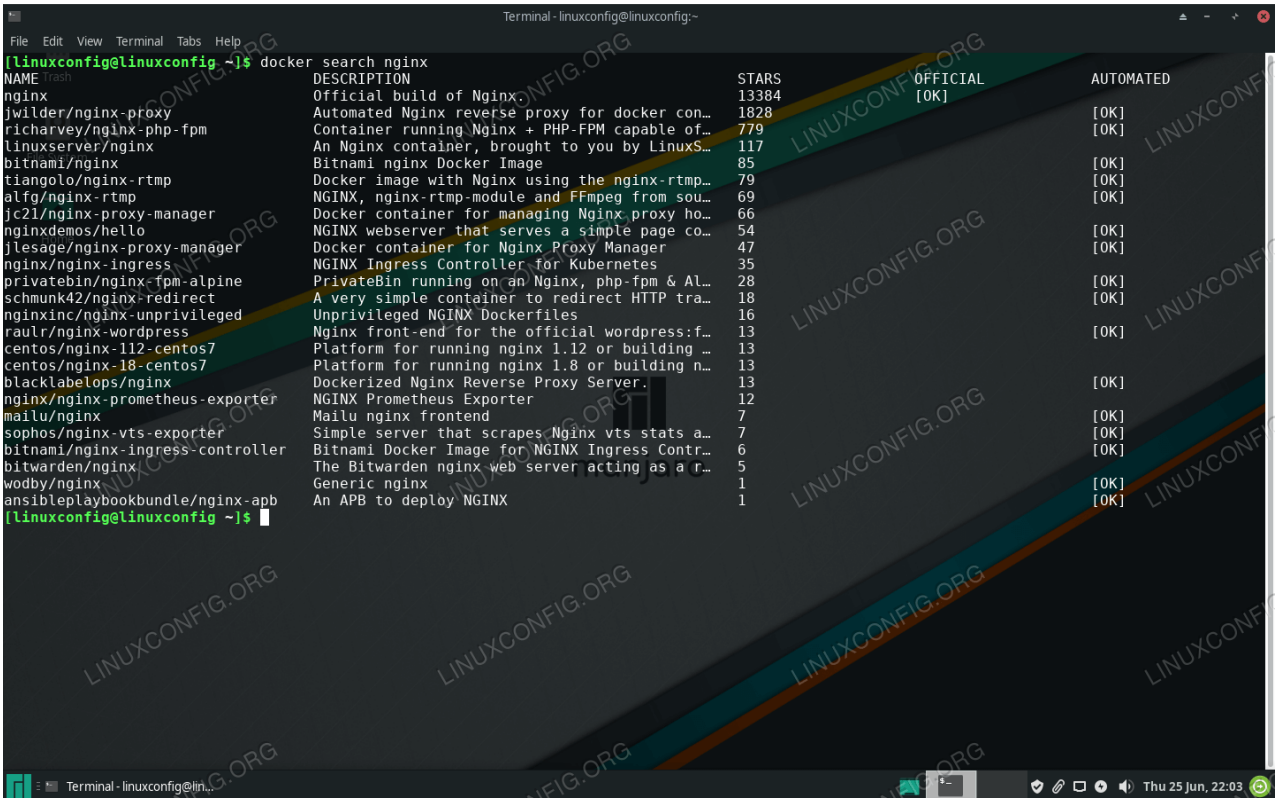
Searching for a Docker image

Now you're ready to install images with Docker. If you already know the name of an image that you'd like to install, you can move on to the next section. If you need to search through Docker for the desired software, you can use the following command syntax:

```
$ docker search [name]
```

For example, let's try searching for `nginx`, which is popular web server software.

```
$ docker search nginx
```



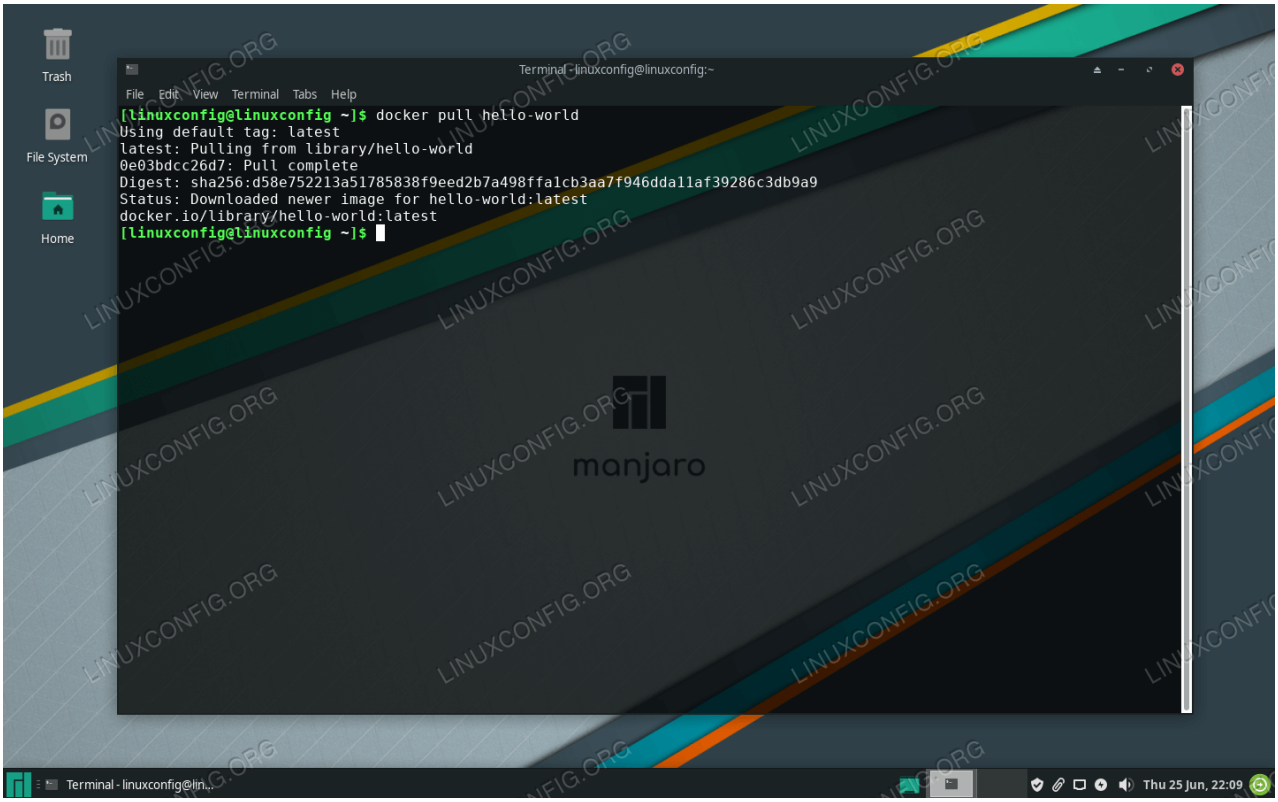
Docker can search for any available container images

As you can see, there is one official image for nginx (indicated by the `OFFICIAL` column) simply called `nginx`. There are also other releases available, and you would have to read their descriptions to see what they do differently from the official image.

Install a Docker image

Once you know which image you'd like to install, you can use the following command to instruct Docker to download the desired software. Just as an example, we'll install the `hello-world` package which can be used to make sure that Docker is able to download and run images successfully.

```
$ docker pull hello-world
```

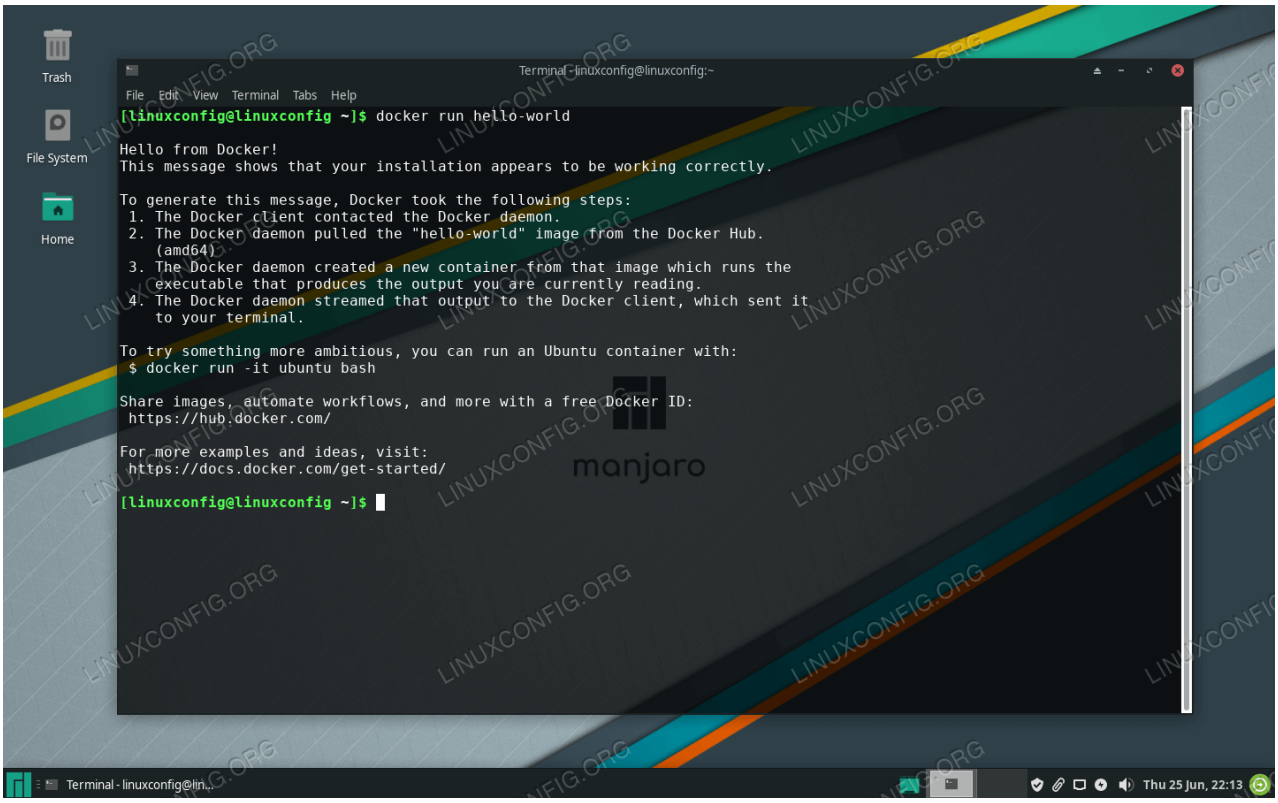


Docker downloading a container image
The output in the screenshot above indicates that Docker was able to find and download the image we specified.

Running a Docker image

Now that the image is downloaded, run it with the following command:

```
$ docker run hello-world
```



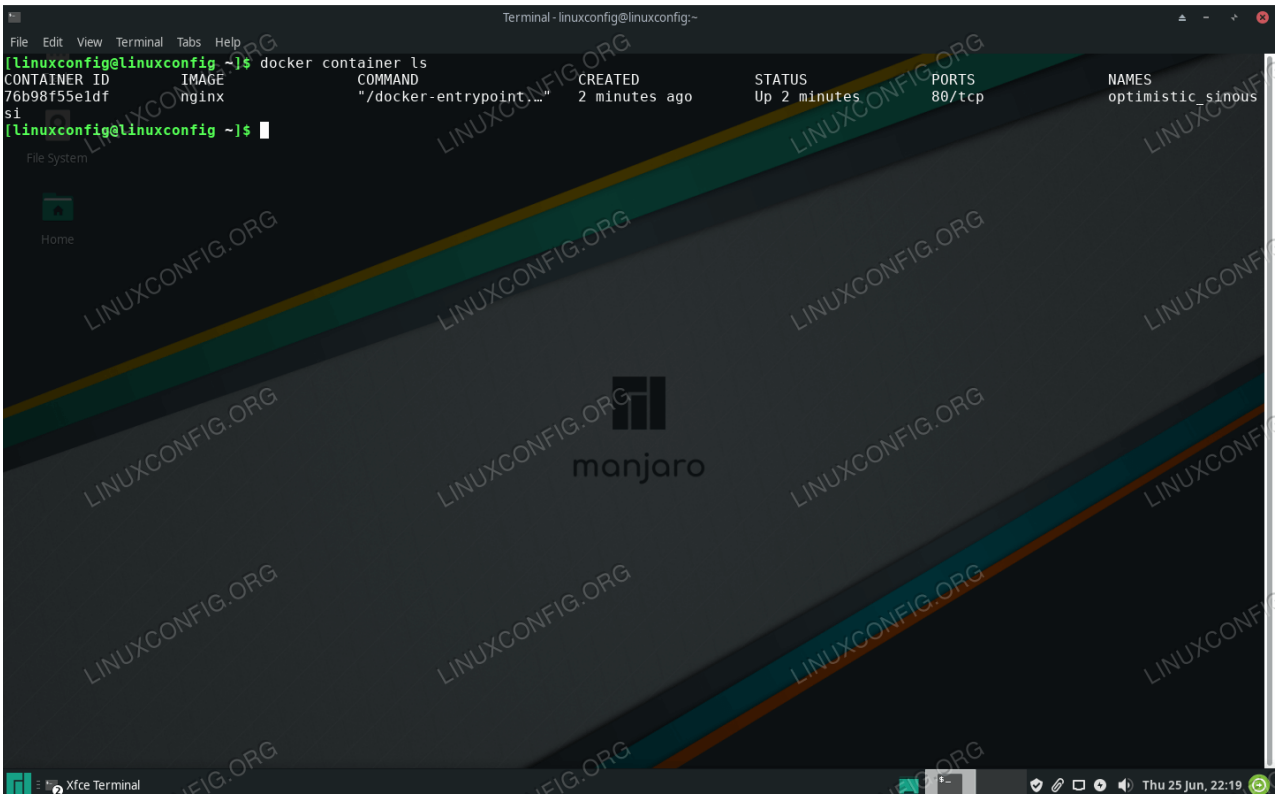
Docker was able to run the hello-world image successfully

Monitoring Docker

Docker gives us a lot of options to see what's going on with our images, how many system resources they're using, etc. The following commands come in handy for monitoring Docker and our installed images.

To see which Docker containers are running and check their current status, type:

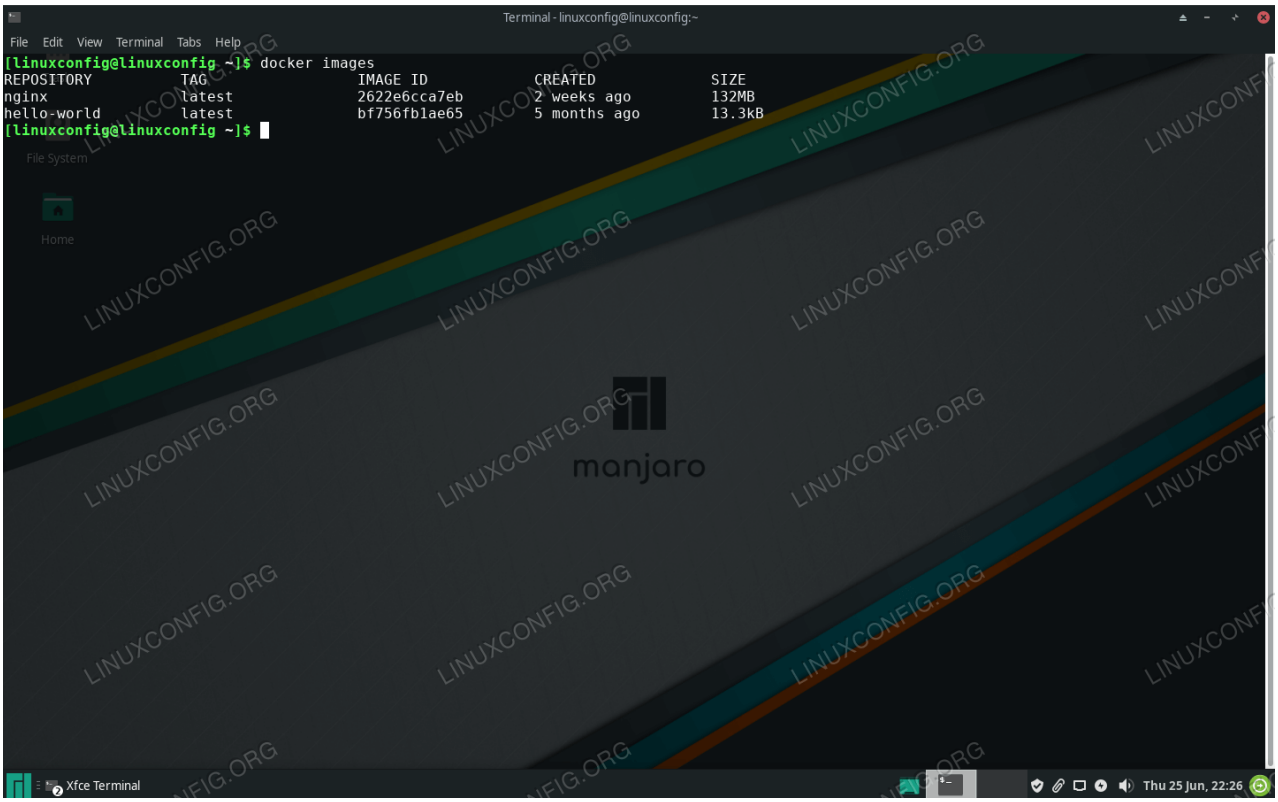
```
$ docker container ls
```



The docker container ls command shows currently running containers
To see a list of all the Docker images installed, type:

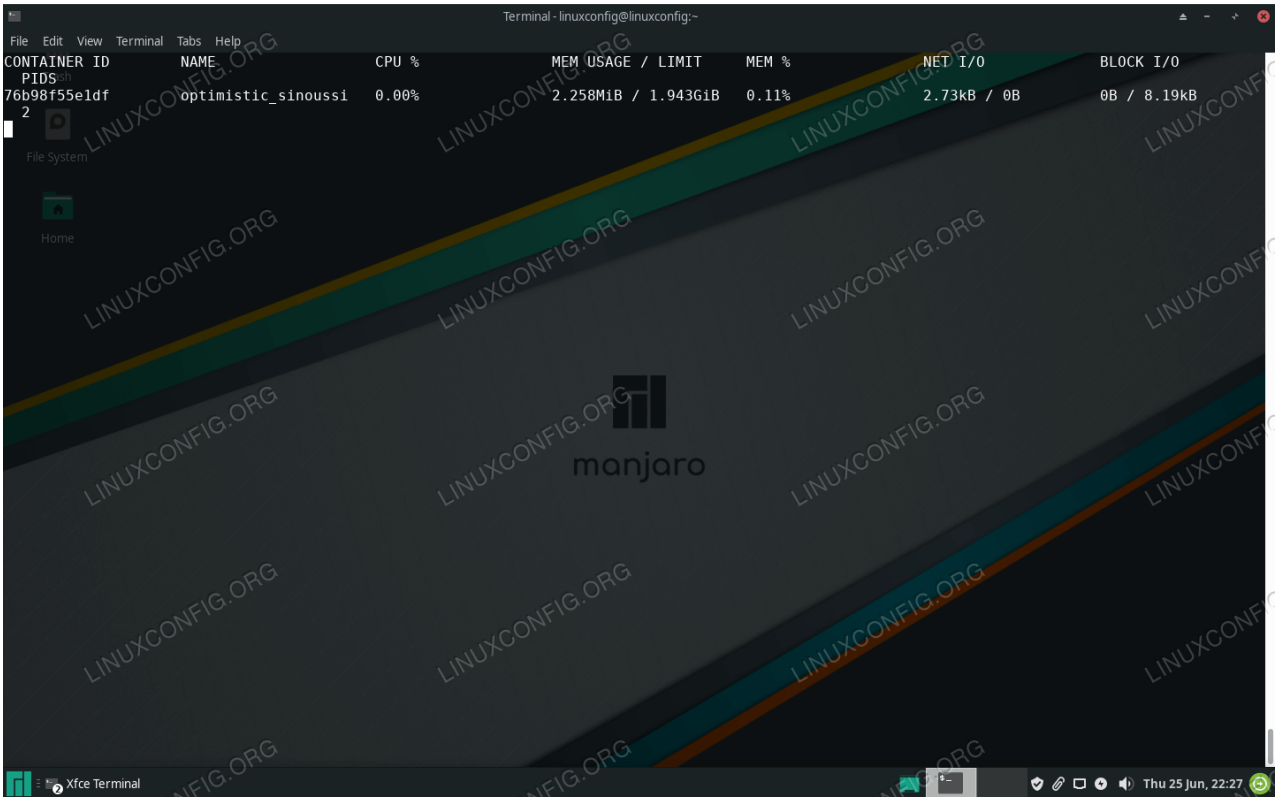
```
$ docker images
```

AD



The docker images command shows which images are currently installed in Docker
To see the current CPU, RAM, and network usage of running images, type:

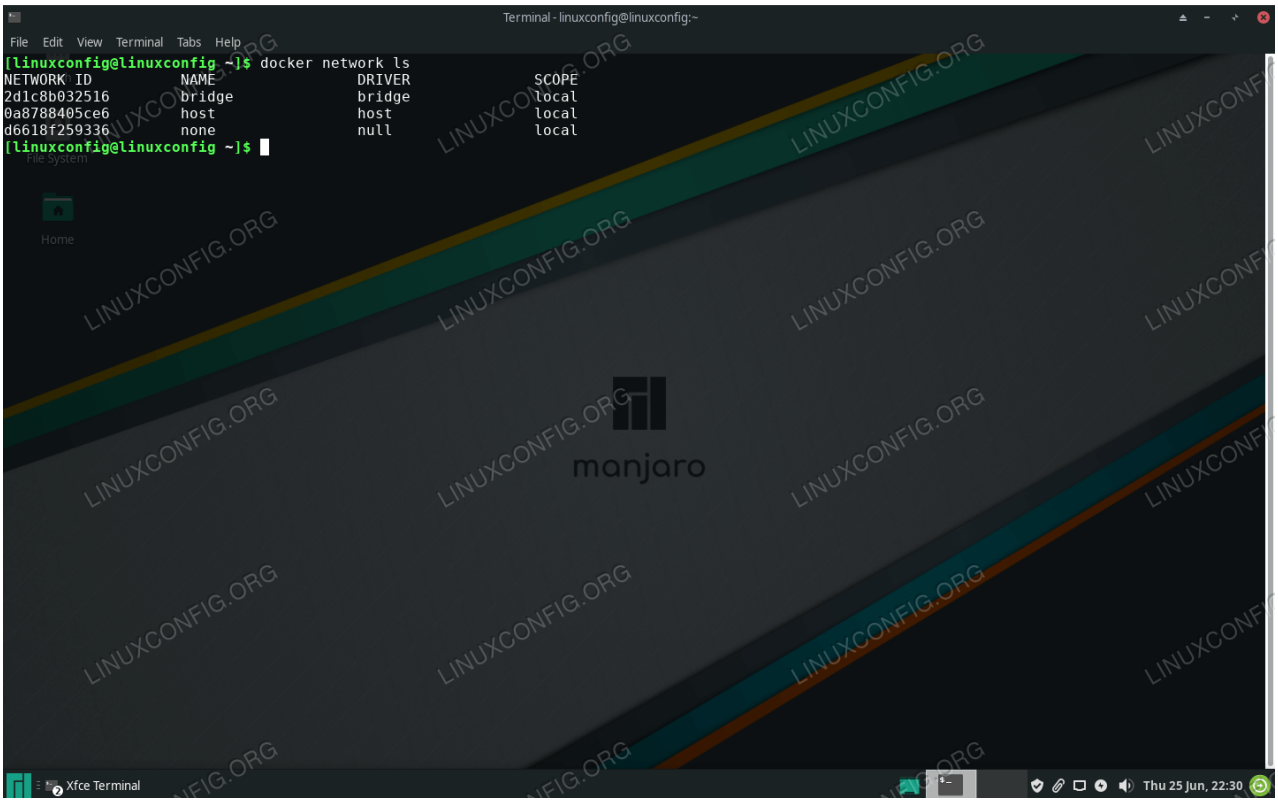
```
$ docker stats
```



See the current system usage of Docker images with docker stats command
To see Docker's network configuration, type:

AD

```
$ docker network ls
```



See currently configured networks in Docker with docker network ls

Closing Thoughts

Docker is pretty easy to use on Manjaro. Like most new things, you just need to learn the command syntax that comes with it, which is very simple anyway. Now that Docker is up and running on your Manjaro system, you'll be able to search for and install new container images as you please. Chances are that you'll find this a lot more convenient than manually installing or building certain packages on your own.