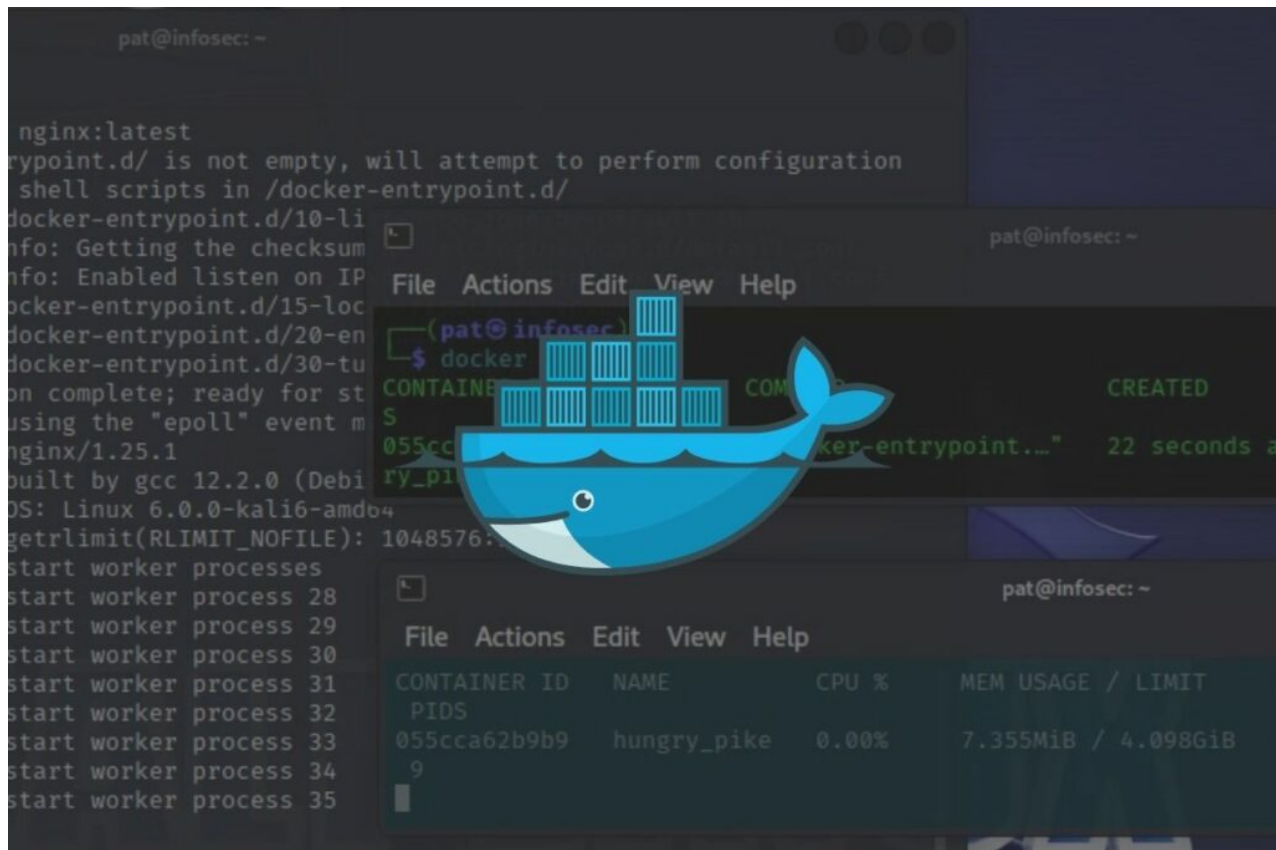


Getting Started with Docker on Kali Linux: A complete guide

 infosecscout.com/docker-on-kali-linux

Patrick Fromaget



```
pat@infosec: ~  
  
nginx:latest  
rypoint.d/ is not empty, will attempt to perform configuration  
shell scripts in /docker-entrypoint.d/  
docker-entrypoint.d/10-li  
nfo: Getting the checksum  
nfo: Enabled listen on IP  
docker-entrypoint.d/15-loc  
docker-entrypoint.d/20-en  
docker-entrypoint.d/30-tu  
on complete; ready for st  
using the "epoll" event m  
nginx/1.25.1  
built by gcc 12.2.0 (Debi  
OS: Linux 6.0.0-kali6-amd64  
getrlimit(RLIMIT_NOFILE): 1048576:  
start worker processes  
start worker process 28  
start worker process 29  
start worker process 30  
start worker process 31  
start worker process 32  
start worker process 33  
start worker process 34  
start worker process 35  
  
$ docker  
CONTAINER ID   NAME          CPU %     MEM USAGE / LIMIT   CREATED  
055cca62b9b9   hungry_pike   0.00%     7.355MiB / 4.098GiB  22 seconds a  
9
```

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT
055cca62b9b9	hungry_pike	0.00%	7.355MiB / 4.098GiB

Nowadays, more and more projects rely on Docker to run. It's often an easier option to set up something without interfering with the existing system, but it doesn't mean it's an easy concept or application. If it's a bit unclear to you, just keep reading, and you'll learn how to install and use Docker on Kali Linux in this article.

Docker is a service allowing to install and run applications in separate containers, isolated from the main operating system. It's used to simplify deployment, security and scalability. Luckily, it's available in the main repository on Kali Linux, so the installation is not too complicated.

Let's start with a brief theory introduction, before talking about the installation and taking an example to make everything clear. Please don't hesitate to use the table of contents below if you already know a bit about Docker.

Master Linux Commands

Your essential Linux handbook

Want to level up your Linux skills? Here is the perfect solution to become efficient on Linux. **20% off today!**

[Download now](#)

What is Docker?

Think of Docker as a magical box that contains everything an application needs to work correctly.

Instead of installing each piece of the application separately on your computer, which can be tricky, Docker puts all those pieces together in one big box, called a container.

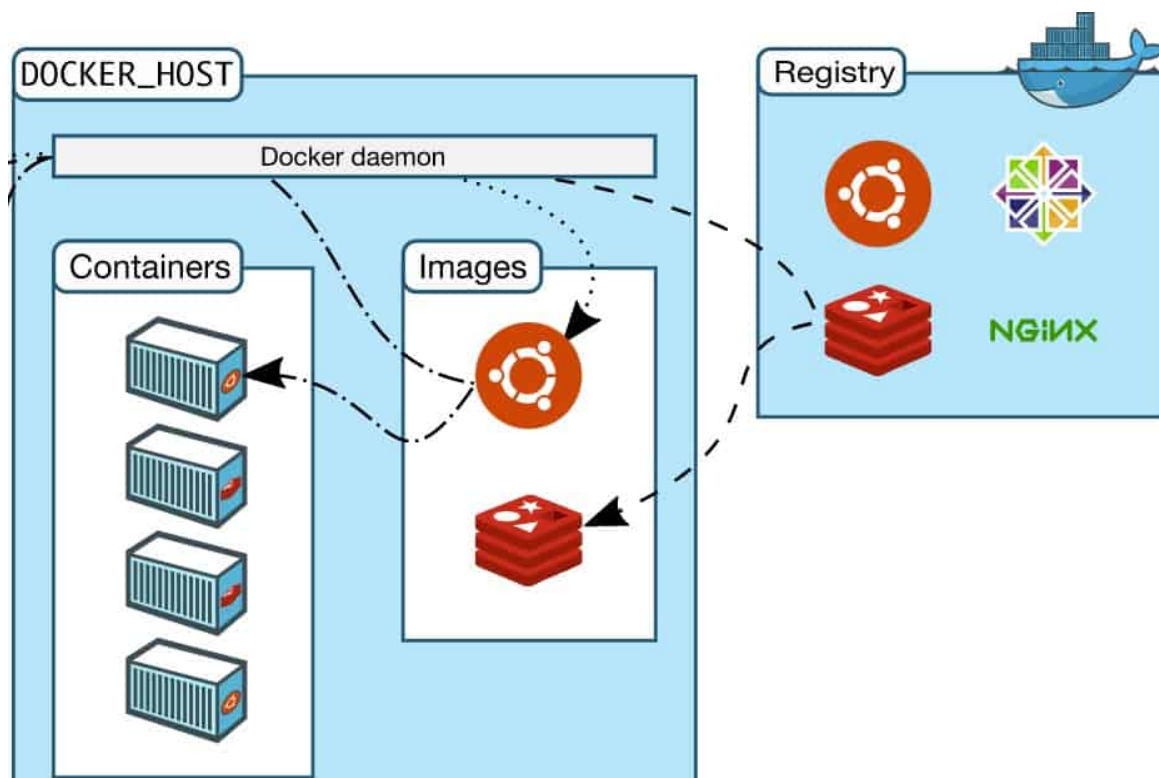
Hide your IP address and location with a free VPN:

[Try it for free now](#), with advanced security features.

2900+ servers in 65 countries. It's free. Forever.

Developers create this special box by gathering all the necessary stuff for the application to run smoothly. Then they share this box with users like you. When you get the box, you can open it and everything inside will work just as the developer intended.

This way, you don't have to worry about installing complicated software or dealing with headaches. Docker makes it super easy to run applications without any fuss!



At the end of this article, I'll give you a real-life example. But here's the basic idea: Imagine you want to install a complex service or application on your Kali Linux system. It can be tough to get everything working together flawlessly, and you might spend a lot of time setting everything up.

But here is the cool part: If a developer has already built this project and put everything together in a Docker container, he can share it with you.

When you'll start this container on your machine, it will work exactly as if it were on the developer computer. It's like magic! You can get the same project running in no time.

How to install Docker on Kali Linux

Docker is not pre-installed on Kali Linux. A service is required to create or run containers, so we need to install it first. Hopefully, this is not complicated. Unlike on other systems, Docker is available in Kali Linux repositories.

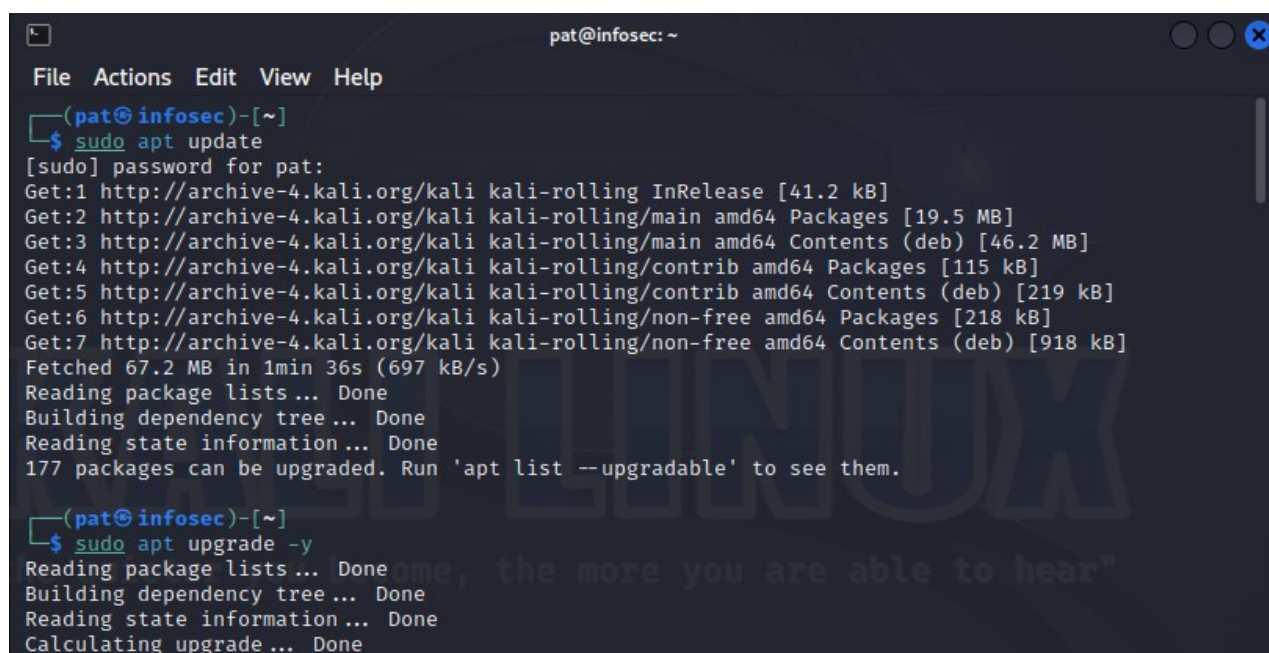
Update your system

As for any installation tutorial, the first step is to make sure your system is up-to-date. It might take a few minutes to download and install all the new packages version, but it will avoid most issues later on.

Open a terminal and type these commands:

```
sudo apt update
```

```
sudo apt upgrade -y
```

A screenshot of a terminal window titled 'pat@infosec: ~'. The terminal shows the execution of two commands. The first command is 'sudo apt update', which prompts for the password 'pat:' and then displays the output of the update process, including fetching package lists and building the dependency tree. The second command is 'sudo apt upgrade -y', which also prompts for the password and shows the output of the upgrade process, including reading package lists and calculating the upgrade. The terminal output for the first command is: [pat@infosec]-[~]
\$ sudo apt update
[sudo] password for pat:
Get:1 http://archive-4.kali.org/kali kali-rolling InRelease [41.2 kB]
Get:2 http://archive-4.kali.org/kali kali-rolling/main amd64 Packages [19.5 MB]
Get:3 http://archive-4.kali.org/kali kali-rolling/main amd64 Contents (deb) [46.2 MB]
Get:4 http://archive-4.kali.org/kali kali-rolling/contrib amd64 Packages [115 kB]
Get:5 http://archive-4.kali.org/kali kali-rolling/contrib amd64 Contents (deb) [219 kB]
Get:6 http://archive-4.kali.org/kali kali-rolling/non-free amd64 Packages [218 kB]
Get:7 http://archive-4.kali.org/kali kali-rolling/non-free amd64 Contents (deb) [918 kB]
Fetched 67.2 MB in 1min 36s (697 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
177 packages can be upgraded. Run 'apt list --upgradable' to see them.
The output for the second command is: [pat@infosec]-[~]
\$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done

Once done, we can move to the Docker installation in itself. I would suggest a computer restart if there were a lot of packages installed, just to make sure everything is still working properly before installing new things:

```
sudo reboot
```

Install the Docker package

As mentioned earlier, Docker can be installed directly with the Kali Linux package manager (APT), so you just need one command line:

```
sudo apt install docker.io
```

```
pat@infosec: ~  
File Actions Edit View Help  
  
(pat@infosec)-[~]  
$ sudo apt install docker.io  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  cgroupfs-mount containerd criu libintl-perl libintl-xs-perl libmodule-find-perl  
  libmodule-scandeps-perl libproc-processtable-perl libsort-naturally-perl needrestart runc  
  tini  
Suggested packages:  
  containernetworking-plugins docker-doc aufs-tools btrfs-progs debootstrap rinse rootlesskit  
  xfsprogs zfs-fuse | zfsutils-linux  
The following NEW packages will be installed:  
  cgroupfs-mount containerd criu docker.io libintl-perl libintl-xs-perl libmodule-find-perl  
  libmodule-scandeps-perl libproc-processtable-perl libsort-naturally-perl needrestart runc  
  tini  
0 upgraded, 13 newly installed, 0 to remove and 7 not upgraded.  
Need to get 66.7 MB of archives.  
After this operation, 269 MB of additional disk space will be used.  
Do you want to continue? [Y/n]
```

All the requirements will be installed automatically. Just wait a few minutes to download and install everything and it should be fine.

Allow the main user to use Docker

By default, only root can use Docker commands. It might make sense on some servers, but in general, for a Kali Linux user, we need the right to use it from the main session too.

Instead of switching to root or using sudo all the time, I recommend adding the permission for your current user to use Docker commands.

To do this, add your current user to the “docker” group:

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

Once done, you need to exit your current session. If you use SSH, exit and reconnect. If you use the GUI, you need to log out and log in again (or restart the computer).

A quick test you can do is to run this command:

```
docker ps
```

If you get something like that, it means you have the permissions:

```
pat@infosec: ~  
File Actions Edit View Help  
  
(pat@infosec)-[~]  
$ docker ps  
CONTAINER ID    IMAGE    COMMAND    CREATED    STATUS    PORTS    NAMES
```

But if you get an error message like “Permission denied while trying to connect to the Docker daemon socket”, then you still don’t have the permission. If you already did the previous commands and restarted, try to [check the documentation here](#).

Testing the Docker setup

A quick test we can do to make sure is working is to download and run a test container, named “hello-word”.

You can do this with this command:

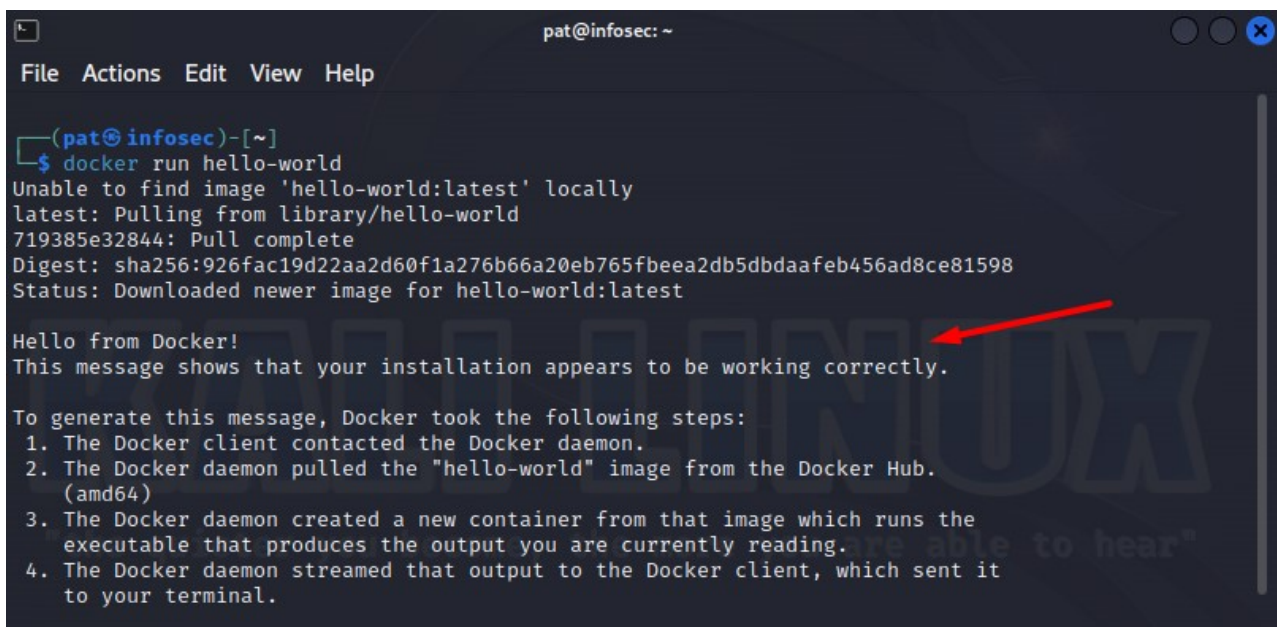
```
docker run hello-world
```

Master Ethical Hacking Skills!

Join the [Complete Ethical Hacking Course Bundle](#) and step into the world of cybersecurity.

Learn to think like a hacker and protect systems with this comprehensive course.

Docker will first try to run it, then see that it's not available locally, so it will be downloaded and started. This application does nothing unless saying “Hello from Docker!”, but it's a good test to make sure everything is installed properly.

A screenshot of a terminal window titled 'pat@infosec: ~'. The terminal shows the command 'docker run hello-world' being executed. The output indicates that the 'hello-world:latest' image was not found locally and was pulled from the Docker Hub. The terminal then displays 'Hello from Docker!' and a message stating that the installation appears to be working correctly. A red arrow points to the 'Hello from Docker!' message. Below this, the terminal lists the steps Docker took to generate the message: 1. The Docker client contacted the Docker daemon. 2. The Docker daemon pulled the "hello-world" image from the Docker Hub. (amd64) 3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading. 4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal. The terminal window has a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'.

Your first steps with Docker on Kali Linux

Docker commands

The main command with Docker is simply “docker”, but then there are many actions and parameters to do everything with it. Here is a summary of the main commands you’ll use all the time:

- **Monitor the running containers:**
`docker ps`
- **Display the current version of Docker:**
`docker version`
- **Download a new image:**
`docker pull [IMAGE]`

- Run an image (and download it if not existing on your local system):

```
docker run [IMAGE]
```

- Search for an image in the Docker repository:

```
docker search [X]
```

```

pat@infosec: ~
File Actions Edit View Help

(pat@infosec)-[~]
$ docker search plex
NAME                                DESCRIPTION                                STARS    OFF
linuxserver/plex                    A Plex Media Server container, brought to yo... 1706
plexinc/pms-docker                  Official Plex Media Server Docker Repo          1544
linuxserver/plexpy                  DEPRECATED PLEASE USE https://hub.docker.com... 197
lsioarmhf/plex                      ARMHF based Linuxserver.io image of plex         58
jaymoulin/plex                      Plex Media Server Docker container (Multiarc... 56

```

- Show the usage statistics:

```
docker stats
```

- Display the list of all the Docker commands:

```
docker help
```

Example: Nginx

Now that Docker is installed and tested, you're almost ready to start any project you were initially interested in. But let's take a full example to make sure you understand the steps, and don't lose time later on.

Let's say you want to install Nginx (a web server), but in a Docker container instead of the usual Debian package.

Here are the steps:

- Use the website or the “search” command to check the corresponding images available:

```
docker search nginx
```

```

pat@infosec: ~
File Actions Edit View Help

(pat@infosec)-[~]
$ docker search nginx
NAME                                DESCRIPTION                                STARS    OFFICIAL  AUTOMATED
nginx                               Official build of Nginx.                  18775    [OK]
linuxserver/nginx                  An Nginx container, brought to you by LinuxS... 204
bitnami/nginx                      Bitnami nginx Docker Image                168      [OK]
nginxproxy/acme-companion          Automated ACME SSL certificate generation fo... 116

```

- There are many results, but the first one is generally the best. So let's install and run it with:

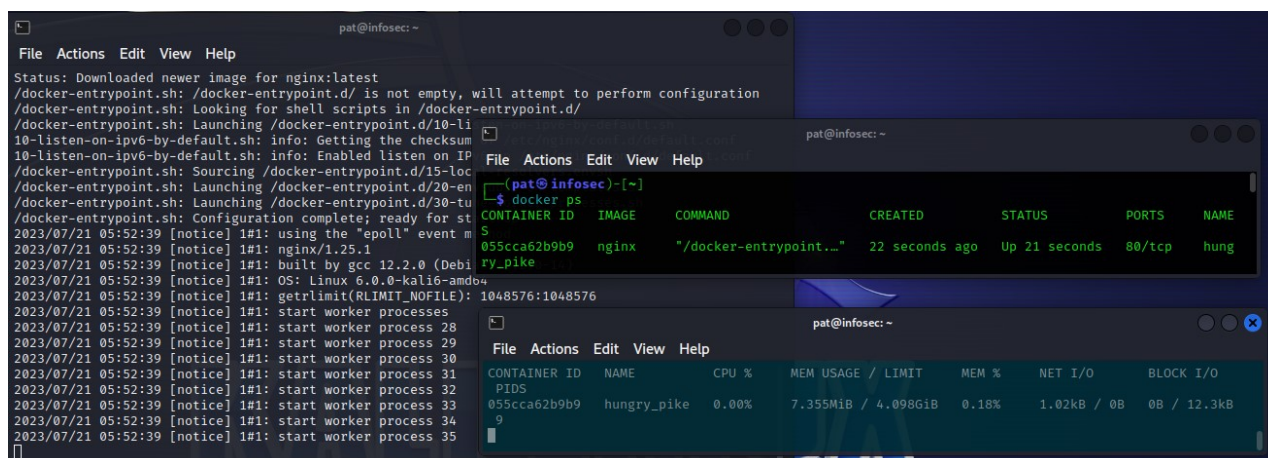
`docker run nginx`

```
(pat@infosec)-[~]
$ docker run nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
faef57eae888: Downloading 7.129MB/29.12MB
76579e9ed380: Downloading 4.254MB/41.46MB
cf707e233955: Download complete
91bb7937700d: Download complete
4b962717ba55: Download complete
f46d7b05649a: Download complete
103501419a0a: Download complete
```

- As for the hello-world example, it will check if it's already installed, and if not, download it.
Then, it will start the container.
You can use "`docker ps`" or "`docker stats`" to check if it's running.
- **One thing I didn't tell you, is that "`docker run [X]`" will start the container in the current session.** If you close your SSH connection or terminal, the application will stop.

To run it in the background, you need to use the "-d" option:

`docker run -d nginx`



And there you have it! Now you're all set to start using your application. Depending on the image you choose, there might be some extra steps involved. No worries, though! Just check out the documentation related to that specific image for more details.

For example, let's say you picked the Nginx build. If you [visit their page](#), they'll give you some additional parameters and configuration options for your new web server.

I hope things are starting to make more sense now. But you'll probably need a bit more of practice to get used to it, it's normal. Go ahead with your Docker project, and you'll get better and better each time you use it.

Whenever you're ready for more security, here are things you should think about:

- **Break free from Gmail**: You should be able to choose what happens to your data. With Proton, only you can read your emails. [Get private email](#).
- **Protect yourself online**: Use a high-speed Swiss VPN that safeguards your privacy. Open-source, no activity logs. [Get Proton VPN risk-free](#).
- **Master Linux commands**: A sure method to learn (and remember) Linux commands. Useful ones only, one at a time, with clear explanations. [Download the e-book](#).