# How to Install Docker and Docker Compose on Debian 11

## Prerequisites

- A server running Debian 11.
- A root password is configured on the server.

## Getting Started

First, it is recommended to update your system package cache to the latest version. You can update them using the following command:

```
apt-get update -y
```

Once you are done, install other required dependencies using the following command:

```
apt-get install apt-transport-https software-properties-common ca-certificates curl gnupg lsb-release -y
```

## Install Docker

By default, the latest version of Docker is not included in the Debian 11 official repository. So you will need to add the Docker CE repository to the APT. You can add it using the following command:

```
curl -fsSL https://download.docker.com/linux/debian/gpg | apt-key add -
add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/debian $(lsb_release -cs)
```

Once the Docker repository is added, update the repository and install the Docker CE with the following command:

```
apt-get update -y
apt-get install docker-ce docker-ce-cli -y
```

After the installation, verify the Docker CE version using the following command:

```
docker version
```

You should get the following output:

```
Client: Docker Engine - Community
 Version:           20.10.8
 API version:       1.41
 Go version:        go1.16.6
 Git commit:        3967b7d
 Built:             Fri Jul 30 19:54:22 2021
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          20.10.8
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.16.6
  Git commit:       75249d8
  Built:            Fri Jul 30 19:52:31 2021
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.4.9
  GitCommit:        e25210fe30a0a703442421b0f60afac609f950a3
 runc:
  Version:          1.0.1
  GitCommit:        v1.0.1-0-g4144b63
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
```

# Manage Docker Services

You can manage the Docker service easily using the systemd utility.

To start a Docker service, run the following command:

```
systemctl start docker
```

To restart a Docker service, run the following command:

```
systemctl restart docker
```

To stop a Docker service, run the following command:

```
systemctl stop docker
```

To enable the Docker service to start at system reboot, run the following command:

```
systemctl enable docker
```

To check the Docker status, run the following command:

```
systemctl status docker
```

You should see the status of Docker in the following output:

```
? docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2021-09-10 07:19:35 UTC; 27s ago
TriggeredBy: ? docker.socket
       Docs: https://docs.docker.com
   Main PID: 29018 (dockerd)
      Tasks: 7
     Memory: 32.6M
        CPU: 407ms
     CGroup: /system.slice/docker.service
             ??29018 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 10 07:19:34 debian11 dockerd[29018]: time="2021-09-10T07:19:34.809035575Z" level=info msg="s
Sep 10 07:19:34 debian11 dockerd[29018]: time="2021-09-10T07:19:34.809219999Z" level=info msg="c
Sep 10 07:19:34 debian11 dockerd[29018]: time="2021-09-10T07:19:34.809410545Z" level=info msg="C
Sep 10 07:19:34 debian11 dockerd[29018]: time="2021-09-10T07:19:34.897972507Z" level=info msg="L
Sep 10 07:19:35 debian11 dockerd[29018]: time="2021-09-10T07:19:35.186940748Z" level=info msg="D
Sep 10 07:19:35 debian11 dockerd[29018]: time="2021-09-10T07:19:35.298681937Z" level=info msg="L
Sep 10 07:19:35 debian11 dockerd[29018]: time="2021-09-10T07:19:35.356364773Z" level=info msg="D
Sep 10 07:19:35 debian11 dockerd[29018]: time="2021-09-10T07:19:35.357524464Z" level=info msg="D
Sep 10 07:19:35 debian11 systemd[1]: Started Docker Application Container Engine.
Sep 10 07:19:35 debian11 dockerd[29018]: time="2021-09-10T07:19:35.401626151Z" level=info msg="A
```

# Run a Container Using Docker

You can use the **docker run** command to download any image and run it inside the container.

For example, run the following command to download Debian image and run a container:

```
docker run --rm -it --name test debian:latest /bin/sh
```

You should get the following output:

```
Unable to find image 'debian:latest' locally
latest: Pulling from library/debian
955615a668ce: Pull complete
Digest: sha256:08db48d59c0a91afb802ebafc921be3154e200c452e4d0b19634b426b03e0e25
Status: Downloaded newer image for debian:latest
#
```

Run the following command to exit from the Debian container

```
#exit
```

# Install Docker Compose

Although you can install Docker Compose from the official Debian repositories, it is several minor versions behind the latest release, so in this tutorial you'll install it from Docker's GitHub repository. The command that follows is slightly different than the one you'll find on the Releases page. By using the `-o` flag to specify the output file first rather than redirecting the output, this syntax avoids running into a "permission denied" error caused when using `sudo`.

Check the current release and, if necessary, update it in the command that follows:

```
sudo curl -L https://github.com/docker/compose/releases/download/1.25.3/docker-compose-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose
```

Next we'll set the permissions:

```
sudo chmod +x /usr/local/bin/docker-compose
```

Then we'll verify that the installation was successful by checking the version:

```
docker-compose --version
```

This will print out the version we installed:

You should see the following output:

```
docker-compose version 1.29.2, build 5becea4c
docker-py version: 5.0.0
CPython version: 3.7.10
OpenSSL version: OpenSSL 1.1.0l  10 Sep 2019
```

# Docker Compose Commands

To run a Docker compose file, run the following command:

```
docker-compose up -d
```

To stop all running containers, run the following command:

```
docker-compose down
```

To pause and unpause the running container, run the following command:

```
docker-compose pause
docker-compose unpause
```

To list all running containers, run the following command:

```
docker-compose ps
```

To check the logs of running services, run the following command:

```
docker-compose logs
```

---

Revision #2
Created 2024-03-02 22:25:33 UTC by Slitzer
Updated 2025-06-01 22:49:55 UTC by Slitzer