

Install Docker Engine on CentOS

Estimated reading time: 10 minutes

To get started with Docker Engine on CentOS, make sure you meet the prerequisites, then install Docker.

Prerequisites

OS requirements

To install Docker Engine, you need a maintained version of CentOS 7. Archived versions aren't supported or tested.

The `centos-extras` repository must be enabled. This repository is enabled by default, but if you have disabled it, you need to re-enable it (<https://wiki.centos.org/AdditionalResources/Repositories>).

The `overlay2` storage driver is recommended.

Uninstall old versions

Older versions of Docker were called `docker` or `docker-engine`. If these are installed, uninstall them, along with associated dependencies.

```
$ sudo yum remove docker \
    docker-client \
    docker-client-latest \
    docker-common \
    docker-latest \
    docker-latest-logrotate \
    docker-logrotate \
    docker-engine
```

It's OK if `yum` reports that none of these packages are installed.

The contents of `/var/lib/docker/`, including images, containers, volumes, and networks, are preserved. The Docker Engine package is now called `docker-ce`.

Installation methods

You can install Docker Engine in different ways, depending on your needs:

- Most users set up Docker's repositories and install from them, for ease of installation and upgrade tasks. This is the recommended approach.
- Some users download the RPM package and install it manually and manage upgrades completely manually. This is useful in situations such as installing Docker on air-gapped systems with no access to the internet.
- In testing and development environments, some users choose to use automated convenience scripts to install Docker.

Install using the repository

Before you install Docker Engine for the first time on a new host machine, you need to set up the Docker repository. Afterward, you can install and update Docker from the repository.

SET UP THE REPOSITORY

Install the `yum-utils` package (which provides the `yum-config-manager` utility) and set up the **stable** repository.

```
$ sudo yum install -y yum-utils

$ sudo yum-config-manager \
    --add-repo \
    https://download.docker.com/linux/centos/docker-ce.repo
```

Optional: Enable the nightly or test repositories.

These repositories are included in the `docker.repo` file above but are disabled by default. You can enable them alongside the stable repository. The following command enables the **nightly** repository.

```
$ sudo yum-config-manager --enable docker-ce-nightly
```

To enable the **test** channel, run the following command:

```
$ sudo yum-config-manager --enable docker-ce-test
```

You can disable the **nightly** or **test** repository by running the `yum-config-manager` command with the `--disable` flag. To re-enable it, use the `--enable` flag. The following command disables the **nightly** repository.

```
$ sudo yum-config-manager --disable docker-ce-nightly
```

Learn about **nightly** and **test** channels (/engine/install/).

INSTALL DOCKER ENGINE

1. Install the *latest version* of Docker Engine and containerd, or go to the next step to install a specific version:

```
$ sudo yum install docker-ce docker-ce-cli containerd.io
```

If prompted to accept the GPG key, verify that the fingerprint matches `060A 61C5 1B55 8A7F 742B 77AA C52F EB6B 621E 9F35`, and if so, accept it.

Got multiple Docker repositories?

If you have multiple Docker repositories enabled, installing or updating without specifying a version in the `yum install` or `yum update` command always installs the highest possible version, which may not be appropriate for your stability needs.

Docker is installed but not started. The `docker` group is created, but no users are added to the group.

2. To install a *specific version* of Docker Engine, list the available versions in the repo, then select and install:

- a. List and sort the versions available in your repo. This example sorts results by version number, highest to lowest, and is truncated:

```
$ yum list docker-ce --showduplicates | sort -r

docker-ce.x86_64 3:18.09.1-3.el7          docker-ce-stable
docker-ce.x86_64 3:18.09.0-3.el7          docker-ce-stable
docker-ce.x86_64 18.06.1.ce-3.el7         docker-ce-stable
docker-ce.x86_64 18.06.0.ce-3.el7         docker-ce-stable
```

The list returned depends on which repositories are enabled, and is specific to your version of CentOS (indicated by the `.el7` suffix in this example).

- b. Install a specific version by its fully qualified package name, which is the package name (`docker-ce`) plus the version string (2nd column) starting at the first colon (`:`), up to the first hyphen, separated by a hyphen (`-`). For example, `docker-ce-18.09.1`.

```
$ sudo yum install docker-ce-18.09.1 docker-ce-cli-18.09.1 containerd.io
```

Docker is installed but not started. The `docker` group is created, but no users are added to the group.

3. Start Docker.

```
$ sudo systemctl start docker
```

4. Verify that Docker Engine is installed correctly by running the `hello-world` image.

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

This command downloads a test image and runs it in a container. When the container runs, it prints an informational message and exits.

Docker Engine is installed and running. You need to use `sudo` to run Docker commands. Continue to Linux postinstall (/engine/install/linux-postinstall/) to allow non-privileged users to run Docker commands and for other optional configuration steps.

UPGRADE DOCKER ENGINE

To upgrade Docker Engine, follow the installation instructions, choosing the new version you want to install.

Install from a package

If you cannot use Docker's repository to install Docker, you can download the `.rpm` file for your release and install it manually. You need to download a new file each time you want to upgrade Docker Engine.

1. Go to <https://download.docker.com/linux/centos/> (<https://download.docker.com/linux/centos/>) and choose your version of CentOS. Then browse to `x86_64/stable/Packages/` and download the `.rpm` file for the Docker version you want to install.

Note: To install a **nightly** or **test** (pre-release) package, change the word `stable` in the above URL to `nightly` or `test`. Learn about **nightly** and **test** channels (/engine/install/).

2. Install Docker Engine, changing the path below to the path where you downloaded the Docker package.

```
$ sudo yum install /path/to/package.rpm
```

Docker is installed but not started. The `docker` group is created, but no users are added to the group.

3. Start Docker.

```
$ sudo systemctl start docker
```

4. Verify that Docker Engine is installed correctly by running the `hello-world` image.

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

This command downloads a test image and runs it in a container. When the container runs, it prints an informational message and exits.

Docker Engine is installed and running. You need to use `sudo` to run Docker commands. Continue to Post-installation steps for Linux (/engine/install/linux-postinstall/) to allow non-privileged users to run Docker commands and for other optional configuration steps.

UPGRADE DOCKER ENGINE

To upgrade Docker Engine, download the newer package file and repeat the installation procedure, using `yum -y upgrade` instead of `yum -y install`, and pointing to the new file.

Install using the convenience script

Docker provides convenience scripts at get.docker.com (<https://get.docker.com/>) and test.docker.com (<https://test.docker.com/>) for installing edge and testing versions of Docker Engine - Community into development environments quickly and non-interactively. The source code for the scripts is in the `docker-install` repository (<https://github.com/docker/docker-install>). **Using these scripts is not recommended for production environments**, and you should understand the potential risks before you use them:

- The scripts require `root` or `sudo` privileges to run. Therefore, you should carefully examine and audit the scripts before running them.
- The scripts attempt to detect your Linux distribution and version and configure your package management system for you. In addition, the scripts do not allow you to customize any installation parameters. This may lead to an unsupported configuration, either from Docker's point of view or from your own organization's guidelines and standards.
- The scripts install all dependencies and recommendations of the package manager without asking for confirmation. This may install a large number of packages, depending on the current configuration of your host machine.
- The script does not provide options to specify which version of Docker to install, and installs the latest version that is released in the "edge" channel.
- Do not use the convenience script if Docker has already been installed on the host machine using another mechanism.

This example uses the script at get.docker.com (<https://get.docker.com/>) to install the latest release of Docker Engine - Community on Linux. To install the latest testing version, use test.docker.com (<https://test.docker.com/>) instead. In each of the commands below, replace each occurrence of `get` with `test` .

⚠ Warning:

Always examine scripts downloaded from the internet before running them locally.

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

<output truncated>

If you would like to use Docker as a non-root user, you should now consider adding your user to the “docker” group with something like:

```
sudo usermod -aG docker your-user
```

Remember to log out and back in for this to take effect!

⚠ Warning:

Adding a user to the “docker” group grants them the ability to run containers which can be used to obtain root privileges on the Docker host. Refer to [Docker Daemon Attack Surface \(/engine/security/security/#docker-daemon-attack-surface\)](/engine/security/security/#docker-daemon-attack-surface) for more information.

Docker Engine - Community is installed. It starts automatically on `DEB` -based distributions. On `RPM` -based distributions, you need to start it manually using the appropriate `systemctl` or `service` command. As the message indicates, non-root users can’t run Docker commands by default.

✔ Note:

To install Docker without root privileges, see [Run the Docker daemon as a non-root user \(Rootless mode\) \(/engine/security/rootless/\)](/engine/security/rootless/).

Rootless mode is currently available as an experimental feature.

UPGRADE DOCKER AFTER USING THE CONVENIENCE SCRIPT

If you installed Docker using the convenience script, you should upgrade Docker using your package manager directly. There is no advantage to re-running the convenience script, and it can cause issues if it attempts to re-add repositories which have already been added to the host machine.

Uninstall Docker Engine

1. Uninstall the Docker Engine, CLI, and Containerd packages:

```
$ sudo yum remove docker-ce docker-ce-cli containerd.io
```

2. Images, containers, volumes, or customized configuration files on your host are not automatically removed. To delete all images, containers, and volumes:

```
$ sudo rm -rf /var/lib/docker
```

You must delete any edited configuration files manually.

Next steps

- Continue to [Post-installation steps for Linux \(/engine/install/linux-postinstall/\)](/engine/install/linux-postinstall/).
- Review the topics in [Develop with Docker \(/engine/develop/\)](/engine/develop/) to learn how to build new applications using Docker.

[requirements \(/search/?q=requirements\)](/search/?q=requirements), [apt \(/search/?q=apt\)](/search/?q=apt), [installation \(/search/?q=installation\)](/search/?q=installation), [centos \(/search/?q=centos\)](/search/?q=centos), [rpm \(/search/?q=rpm\)](/search/?q=rpm), [install \(/search/?q=install\)](/search/?q=install), [uninstall \(/search/?q=uninstall\)](/search/?q=uninstall), [upgrade \(/search/?q=upgrade\)](/search/?q=upgrade), [update \(/search/?q=update\)](/search/?q=update)