Lesson 07 Demo 01

Installing Docker Community Edition

Objective: To install Docker Community Edition (CE) on Ubuntu for containerization and managing Docker containers efficiently

Tools required: Docker

Prerequisites: Ubuntu configuration

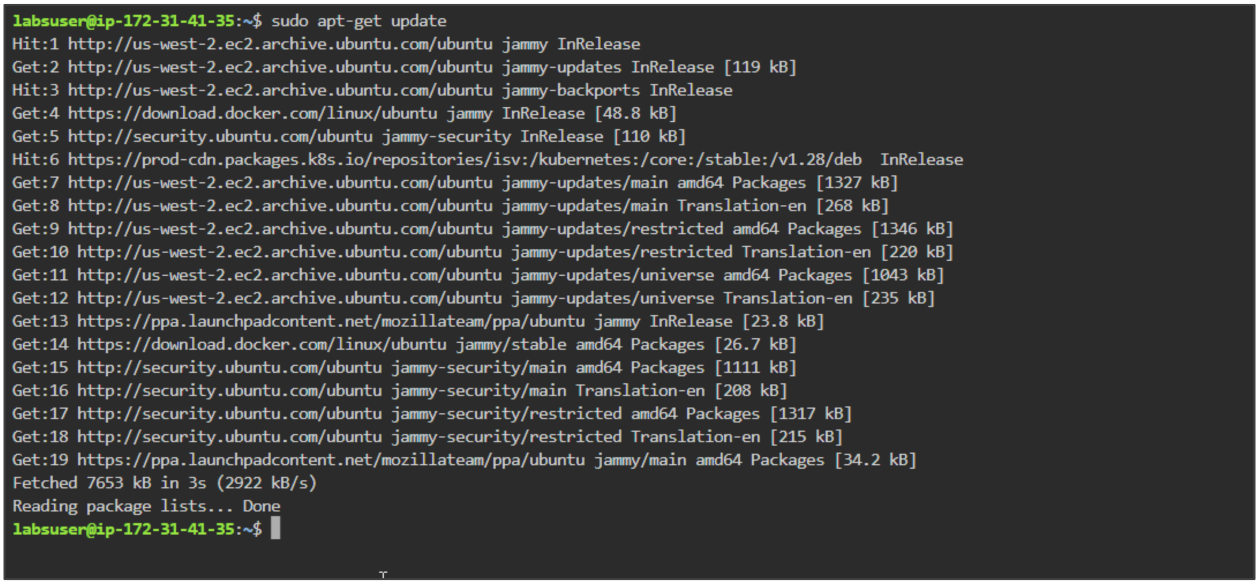
Note: Docker is pre-installed on your machine. In case it does not work or is unavailable, you may follow the provided steps below to install it.

Steps to be followed:

1. Install the Docker CE from Docker repository
2. Verify the correctly installed Docker Engine

Step 1: Install the Docker CE from Docker repository

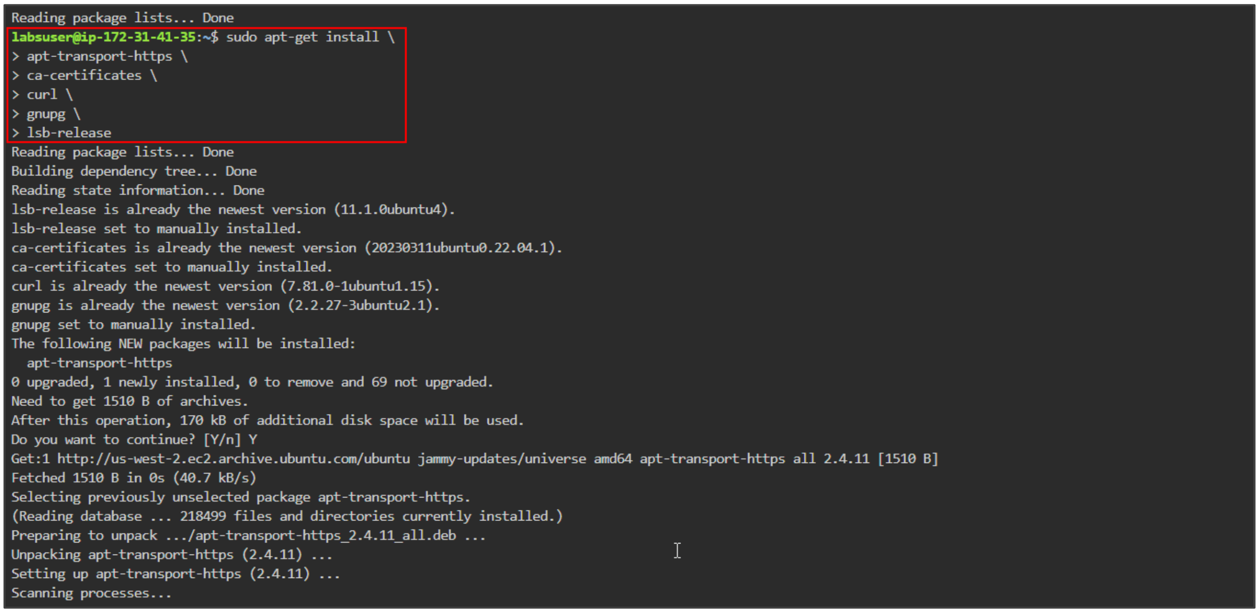
1.1 Use the following command to update the apt package:
sudo apt-get update



```
labsuser@ip-172-31-41-35:~$ sudo apt-get update
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Get:7 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1327 kB]
Get:8 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [268 kB]
Get:9 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1346 kB]
Get:10 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [220 kB]
Get:11 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1043 kB]
Get:12 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [235 kB]
Get:13 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy InRelease [23.8 kB]
Get:14 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [26.7 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1111 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [208 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1317 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [215 kB]
Get:19 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy/main amd64 Packages [34.2 kB]
Fetched 7653 kB in 3s (2922 kB/s)
Reading package lists... Done
labsuser@ip-172-31-41-35:~$
```

1.2 Use the following command to install packages to allow the apt to use a repository over HTTPS:

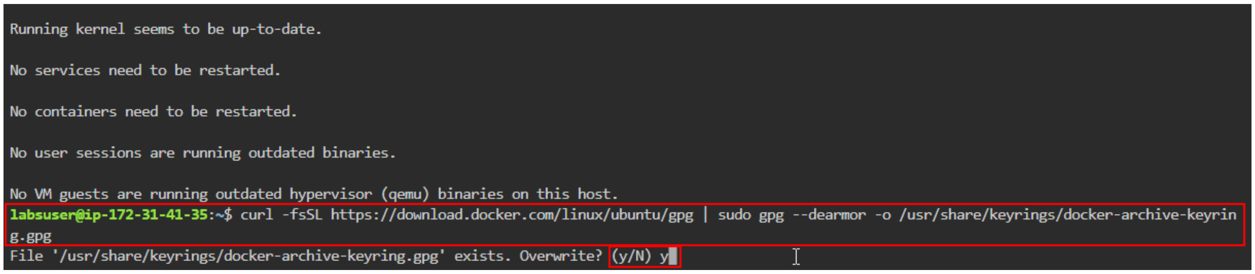
**sudo apt-get install **
**apt-transport-https **
**ca-certificates **
**curl **
**gnupg **
lsb-release



```
Reading package lists... Done
labsuser@ip-172-31-41-35:~$ sudo apt-get install \
> apt-transport-https \
> ca-certificates \
> curl \
> gnupg \
> lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.15).
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 69 not upgraded.
Need to get 1510 B of archives.
After this operation, 170 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.11 [1510 B]
Fetched 1510 B in 0s (40.7 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 218499 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.11_all.deb ...
Unpacking apt-transport-https (2.4.11) ...
Setting up apt-transport-https (2.4.11) ...
Scanning processes...
```

1.3 Use the following curl command to add Docker’s official GPG key:

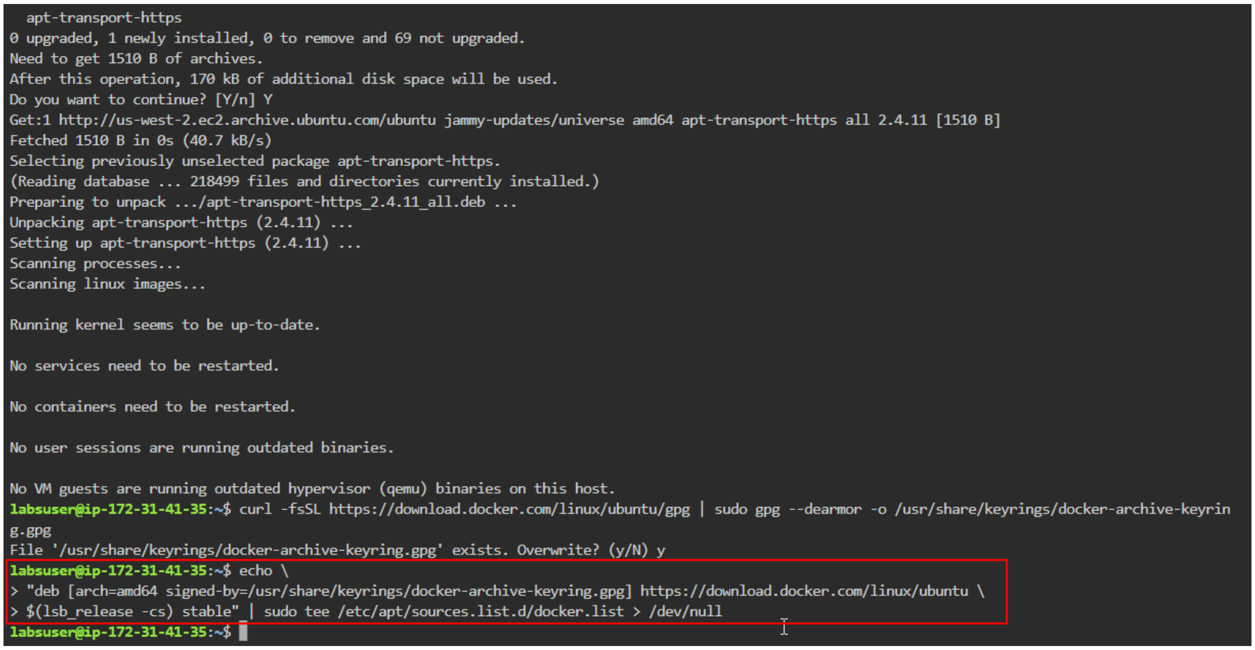
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg



```
Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
labsuser@ip-172-31-41-35:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyrin
g.gpg
File '/usr/share/keyrings/docker-archive-keyring.gpg' exists. Overwrite? (y/N) y
```

1.4 Use the following commands to set up a stable repository as shown in the screenshot below:

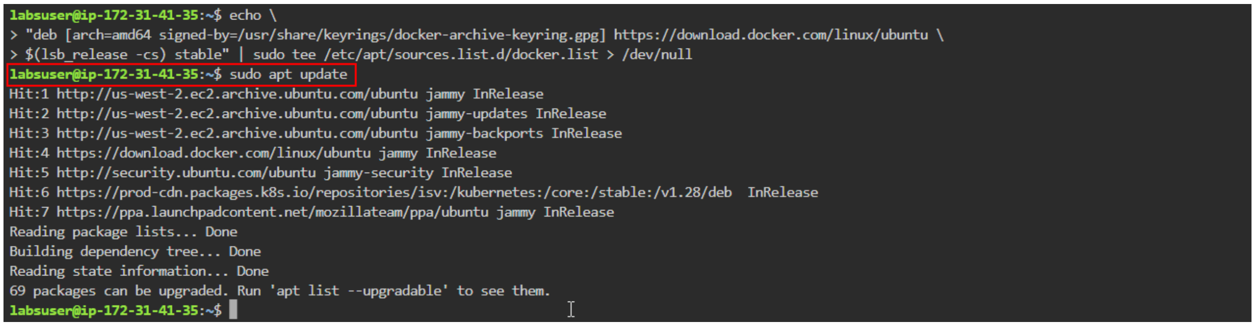
**echo **
**"deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu **
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null



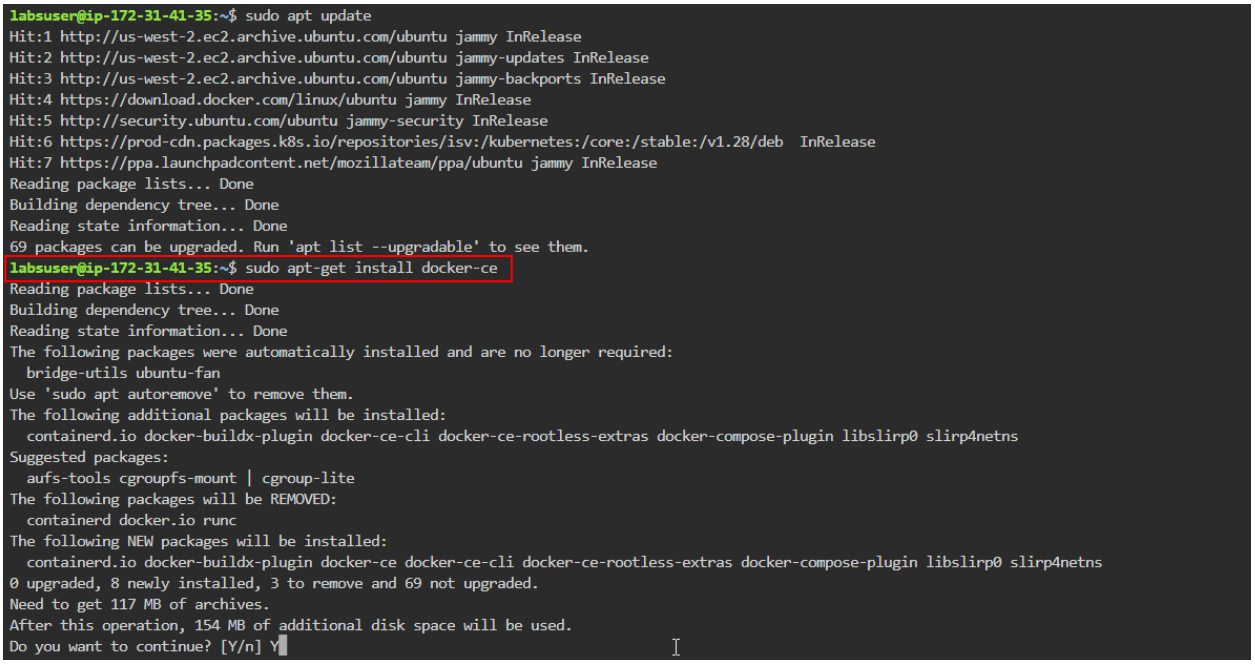
```
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 69 not upgraded.
Need to get 1510 B of archives.
After this operation, 170 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.11 [1510 B]
Fetched 1510 B in 0s (40.7 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 218499 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.11_all.deb ...
Unpacking apt-transport-https (2.4.11) ...
Setting up apt-transport-https (2.4.11) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
labsuser@ip-172-31-41-35:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyrin
g.gpg
File '/usr/share/keyrings/docker-archive-keyring.gpg' exists. Overwrite? (y/N) y
labsuser@ip-172-31-41-35:~$ echo \
> "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
labsuser@ip-172-31-41-35:~$
```

1.5 Use the following commands to install the latest version of Docker CE and check the version:

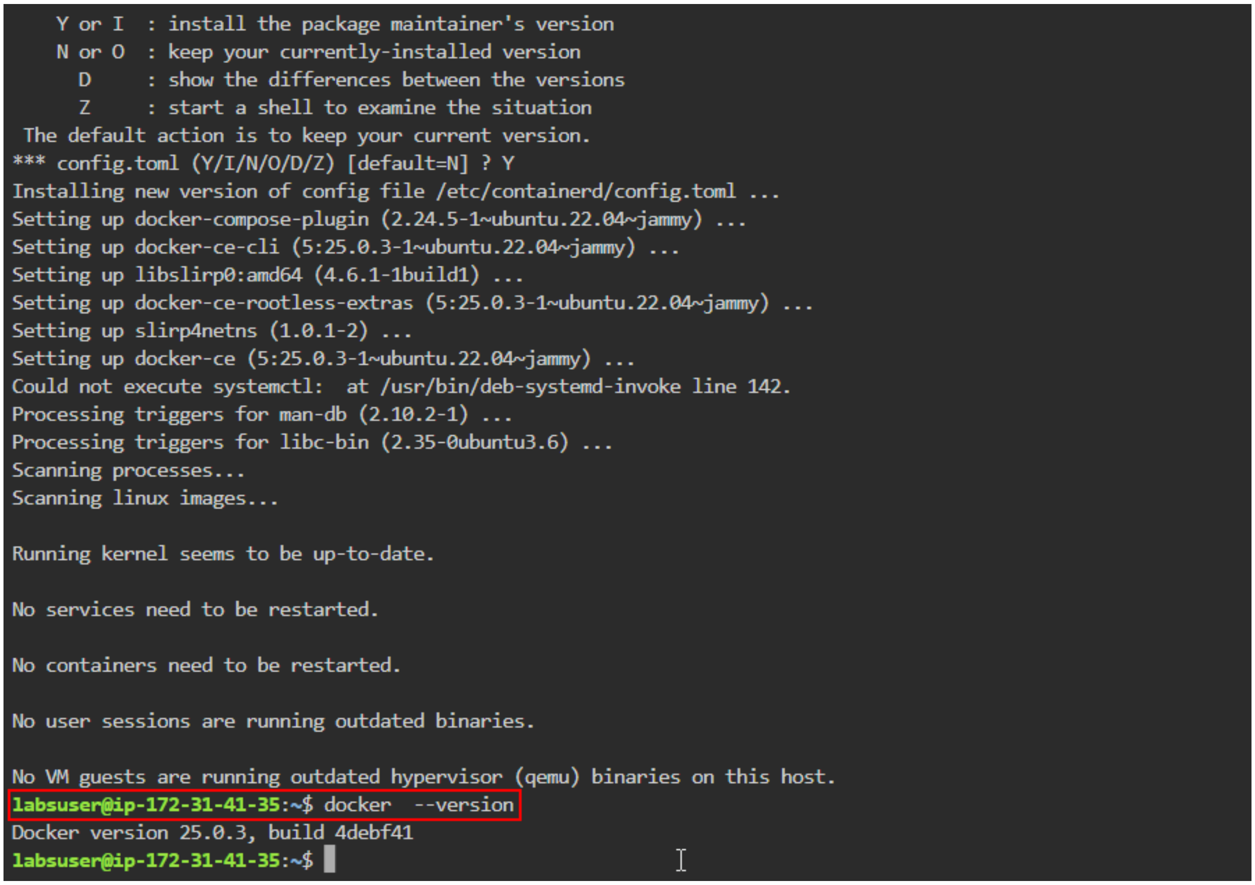
sudo apt update
sudo apt-get install docker-ce
docker --version



```
labsuser@ip-172-31-41-35:~$ echo \
> "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
labsuser@ip-172-31-41-35:~$ sudo apt update
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Hit:7 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
69 packages can be upgraded. Run 'apt list --upgradable' to see them.
labsuser@ip-172-31-41-35:~$
```



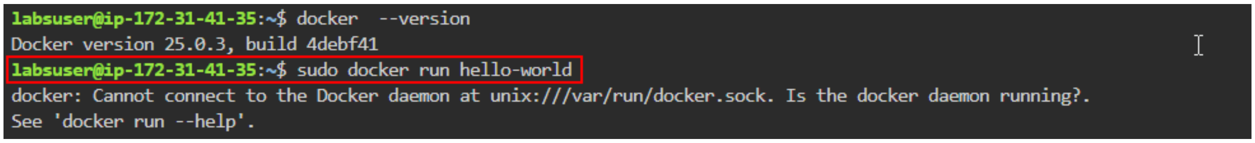
```
labsuser@ip-172-31-41-35:~$ sudo apt update
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Hit:7 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
69 packages can be upgraded. Run 'apt list --upgradable' to see them.
labsuser@ip-172-31-41-35:~$ sudo apt-get install docker-ce
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bridge-utils ubuntu-fan
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  containerd.io docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libslirp0 slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following packages will be REMOVED:
  containerd docker.io runc
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libslirp0 slirp4netns
0 upgraded, 8 newly installed, 3 to remove and 69 not upgraded.
Need to get 117 MB of archives.
After this operation, 154 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```



```
    Y or I  : install the package maintainer's version
    N or O  : keep your currently-installed version
      D     : show the differences between the versions
      Z     : start a shell to examine the situation
 The default action is to keep your current version.
*** config.toml (Y/I/N/O/D/Z) [default=N] ? Y
Installing new version of config file /etc/containerd/config.toml ...
Setting up docker-compose-plugin (2.24.5-1~ubuntu.22.04~jammy) ...
Setting up docker-ce-cli (5:25.0.3-1~ubuntu.22.04~jammy) ...
Setting up libslirp0:amd64 (4.6.1-1build1) ...
Setting up docker-ce-rootless-extras (5:25.0.3-1~ubuntu.22.04~jammy) ...
Setting up slirp4netns (1.0.1-2) ...
Setting up docker-ce (5:25.0.3-1~ubuntu.22.04~jammy) ...
Could not execute systemctl:  at /usr/bin/deb-systemd-invoke line 142.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
labsuser@ip-172-31-41-35:~$ docker  --version
Docker version 25.0.3, build 4debf41
labsuser@ip-172-31-41-35:~$
```

Step 2: Verify the correctly installed Docker engine

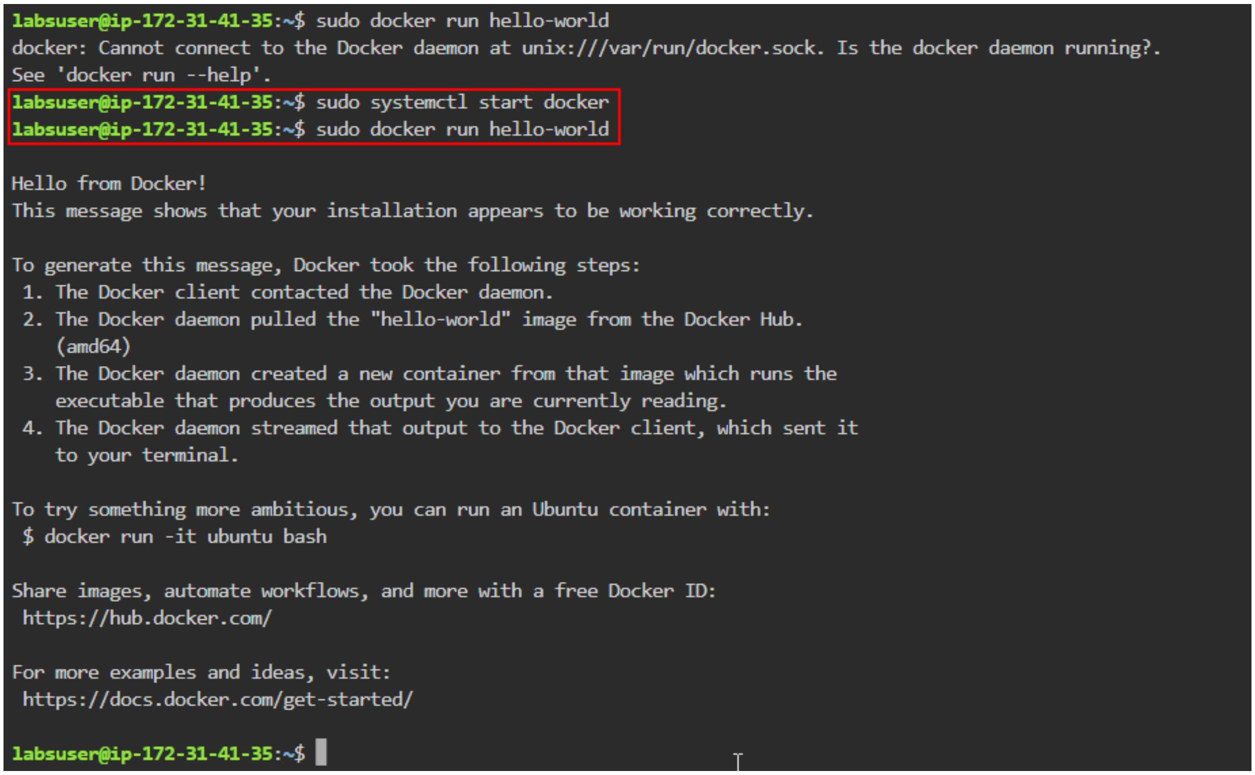
2.1 Use the following command to verify the Docker engine:
sudo docker run hello-world



```
labsuser@ip-172-31-41-35:~$ docker  --version
Docker version 25.0.3, build 4debf41
labsuser@ip-172-31-41-35:~$ sudo docker run hello-world
docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.
See 'docker run --help'.
```

Note: The above error message indicates that the Docker daemon is not running on your system. The Docker daemon is responsible for managing Docker containers and images.

2.2 Start the Docker daemon and verify its execution using the following command:
sudo systemctl start docker
sudo docker run hello-world



```
labsuser@ip-172-31-41-35:~$ sudo docker run hello-world
docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.
See 'docker run --help'.
labsuser@ip-172-31-41-35:~$ sudo systemctl start docker
labsuser@ip-172-31-41-35:~$ sudo docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 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.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

labsuser@ip-172-31-41-35:~$
```

By following these steps, you have successfully installed Docker Community Edition on Ubuntu for containerization and efficient management of Docker containers.