
NETWORKING & SYSTEM ADMINISTRATION LAB**Experiment No: 24****Aim**

Docker installation.

Procedure

Step 1: Open the terminal on Ubuntu.

Step 2: Remove any Docker files that are running in the system, using the following command:

\$ sudo apt-get remove docker docker-engine docker.io

After entering the above command, you will need to enter the password of the root and press enter.

```
mca@S37:~$ sudo apt-get remove docker docker-engine docker.io
[sudo] password for mca:
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package 'docker-engine' is not installed, so not removed
Package 'docker' is not installed, so not removed
Package 'docker.io' is not installed, so not removed
The following packages were automatically installed and are no longer required:
  debhelper dh-autoreconf dh-strip-nondeterminism distro-info
  libarchive-cpio-perl libfile-stripnondeterminism-perl libmail-sendmail-perl
  libpcre16-3 libpcre3-dev libpcre32-3 libpcrecpp0v5 libssl-dev libssl-doc
  libsys-hostname-long-perl php-common php-pear php-xml php7.2-cli
  php7.2-common php7.2-json php7.2-opcache php7.2-readline php7.2-xml
  pkg-php-tools po-debconf shtool x11proto-dri2-dev x11proto-gl-dev
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
```

Name: Shefany Shanavas

Roll No: 37

Batch: B

Date: 23/05/2022

Step 3: Check if the system is up-to-date using the following command:

\$ sudo apt-get update

```
mca@S37:~$ sudo apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu bionic-proposed InRelease
Err:5 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu bionic InRelease
 403 Forbidden [IP: 185.125.190.52 80]
Ign:6 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 InRelease
Hit:7 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:9 http://archive.canonical.com/ubuntu bionic InRelease
Get:10 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release [2,495 B]
Get:8 http://ppa.launchpad.net/webupd8team/java/ubuntu bionic InRelease [15.4 kB]
Get:11 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release.gpg [801 B]
Err:11 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release.gpg
The following signatures were invalid: EXPKEYSIG 58712A2291FA4AD5 MongoDB 3.6 Release Signing Key <packaging@mongodb.com>
Reading package lists... Done
E: Failed to fetch http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu/dists/bionic/InRelease 403 Forbidden [IP: 185.125.190.52 80]
E: The repository 'http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu bionic InRelease' is not signed.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
E: Repository 'http://ppa.launchpad.net/webupd8team/java/ubuntu bionic InRelease' changed its 'Label' value from 'Oracle Java (JDK) 8 / 9 Inst
aller PPA' to 'Oracle Java (JDK) 8 Installer PPA (DISCONTINUED)'
N: This must be accepted explicitly before updates for this repository can be applied. See apt-secure(8) manpage for details.
W: An error occurred during the signature verification. The repository is not updated and the previous index files will be used. GPG error: ht
tps://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release: The following signatures were invalid: EXPKEYSIG 58712A2291FA4AD5 MongoDB 3.
6 Release Signing Key <packaging@mongodb.com>
```

Step 4: Install Docker using the following command:

\$ sudo apt install docker.io

You'll then get a prompt asking you to choose between y/n - choose y

```
mca@S37:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  debhelper dh-autoreconf dh-strip-nondeterminism distro-info libarchive-cpio-perl libfile-stripnondeterminism-perl libmail-sendmail-perl
  libpcre16-3 libpcre3-dev libpcre32-3 libpcrecpp0v5 libssl-dev libssl-doc libsys-hostname-long-perl php-common php-pear php-xml php7.2-cli
  php7.2-common php7.2-json php7.2-opcache php7.2-readline php7.2-xml pkg-php-tools po-debconf shtool x11proto-dri2-dev x11proto-glx-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils containerd pigz runc ubuntu-fan
Suggested packages:
  aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd docker.io pigz runc ubuntu-fan
0 upgraded, 6 newly installed, 0 to remove and 41 not upgraded.
Need to get 74.2 MB of archives.
After this operation, 360 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Setting up pigz (2.4-1) ...
Setting up docker.io (20.10.7-0ubuntu5~18.04.3) ...
Adding group 'docker' (GID 130) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /lib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
Processing triggers for systemd (237-3ubuntu10.52) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
ureadahead will be reprofiled on next reboot
```

Step 5: Install all the dependency packages using the following command:

\$ sudo snap install docker

```
mca@S37:~$ sudo snap install docker
docker 20.10.14 from Canonical* installed
```

Step 6: Before testing Docker, check the version installed using the following command:

\$ docker --version

```
mca@S37:~$ docker --version
Docker version 20.10.7, build 20.10.7-0ubuntu5~18.04.3
```

Step 7: Pull an image from the Docker hub using the following command:

\$ sudo docker run hello-world

Here, *hello-world* is the docker image present on the Docker hub.

```
mca@S37:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:80f31da1ac7b312ba29d65080fddf797dd76acfb870e677f390d5acba9741b17
Status: Downloaded newer image for hello-world:latest

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/
```

Step 8: Check if the docker image has been pulled and is present in your system using the following command:

\$ sudo docker images

```
mca@S37:~$ sudo docker images
REPOSITORY      TAG         IMAGE ID      CREATED        SIZE
hello-world      latest      feb5d9fea6a5  8 months ago  13.3kB
```

Step 9: To display all the containers pulled, use the following command:

\$ sudo docker ps -a

```
mca@S37:~$ sudo docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS              PORTS          NAMES
269499d226a5   hello-world  "/hello"                2 minutes ago  Exited (0) 2 minutes ago           pedantic_euler
```

Step 10: To check for containers in a running state, use the following command:

\$ sudo docker ps

```
mca@S37:~$ sudo docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS              PORTS          NAMES
```

Successfully installed Docker on Ubuntu!