NETWORKING & SYSTEM ADMINISTRATION LAB

Name: VYSHNAVI BABU S

Roll No: 55

Batch: B

Date: 23-05-2022

Experiment No.: 22

Aim

Install Docker in Ubuntu.

Procedure

1. Open the terminal on Ubuntu.
2. Remove any Docker files that are running in the system, using the following command:

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

```
mca@S54:~$ 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 gimp-data i965-va-driver
  libaacs0 libamd2 libarchive-cpio-perl libavcodec57 libavformat57 libavutil55
  libbabl-0.1-0 libbdplus0 libblas3 libbluray2 libcamd2 libccolamd2
  libcholmod3 libchromaprint1 libcrystalhd3 libfile-stripnondeterminism-perl
  libgegl-0.3-0 libgfortran4 libgimp2.0 libgme0 libgsm1 liblapack3
  libmail-sendmail-perl libmetis5 libmng2 libopenjp2-7 libopenmpt0 libpcre16-3
  libpcre3-dev libpcre32-3 libpcrecpp0v5 libshine3 libsnappy1v5 libsoxr0
  libssh-gcrypt-4 libssl-dev libssl-doc libswresample2 libswscale4
  libsys-hostname-long-perl libumfpack5 libva-drm2 libva-x11-2 libva2
  libvdpau1 libx264-152 libx265-146 libxvidcore4 libzvbi-common libzvbi0
  mesa-va-drivers mesa-vdpau-drivers 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 va-driver-all vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
mca@S54:~$
```

After entering the above command, enter the password of the root and press enter.

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

$ sudo apt-get update

```
mca@S54:~$ sudo apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://dl.google.com/linux/chrome/deb stable InRelease [1,811 B]
Get:3 http://ppa.launchpad.net/j-4321-i/ppa/ubuntu bionic InRelease [15.9 kB]
Ign:4 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 InRelease
Err:2 http://dl.google.com/linux/chrome/deb stable InRelease
  The following signatures couldn't be verified because the public key is not available: NO_PUBKEY 78BD65473CB3BD13
Get:5 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release [2,495 B]
Err:6 http://ppa.launchpad.net/jonathonf/python-3.6/ubuntu bionic InRelease
  403  Forbidden [IP: 185.125.190.52 80]
Get:7 https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 Release.gpg [801 B]
Err:7 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>
Get:9 http://ppa.launchpad.net/j-4321-i/ppa/ubuntu bionic/main amd64 Packages [2,888 B]
Get:10 http://ppa.launchpad.net/j-4321-i/ppa/ubuntu bionic/main i386 Packages [2,888 B]
Get:11 http://ppa.launchpad.net/j-4321-i/ppa/ubuntu bionic/main Translation-en [2,452 B]
Get:8 http://ppa.launchpad.net/webupd8team/java/ubuntu bionic InRelease [15.4 kB]
Reading package lists... 10%
```

4. Install Docker using the following command:

$ sudo apt install docker.io

```
mca@S54:~$ 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 gimp-data i965-va-driver
  libaacs0 libamd2 libarchive-cpio-perl libavcodec57 libavformat57 libavutil55
  libbabl-0.1-0 libbdplus0 libblas3 libbluray2 libcamd2 libccolamd2
  libcholmod3 libchromaprint1 libcrystalhd3 libfile-stripnondeterminism-perl
  libgegl-0.3-0 libgfortran4 libgimp2.0 libgme0 libgsm1 liblapack3
  libmail-sendmail-perl libmetis5 libmng2 libopenjp2-7 libopenmpt0 libpcre16-3
  libpcre3-dev libpcre32-3 libpcrecpp0v5 libshine3 libsnappy1v5 libsoxr0
  libssh-gcrypt-4 libssl-dev libssl-doc libswresample2 libswscale4
  libsys-hostname-long-perl libumfpack5 libva-drm2 libva-x11-2 libva2
  libvdpau1 libx264-152 libx265-146 libxvidcore4 libzvbi-common libzvbi0
  mesa-va-drivers mesa-vdpau-drivers 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 va-driver-all vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils cgroupfs-mount ubuntu-fan
Suggested packages:
  aufs-tools btrfs-tools debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils cgroupfs-mount docker.io ubuntu-fan
0 upgraded, 4 newly installed, 0 to remove and 9 not upgraded.
Need to get 30.1 MB of archives.
After this operation, 137 MB of additional disk space will be used.
Do you want to continue? [Y/n] yes
Get:1 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 bridge-utils amd64 1.5-15ubuntu1 [30.1 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic/universe amd64 cgroupfs-mount all 1.4 [6,320 B]
Get:3 http://in.archive.ubuntu.com/ubuntu bionic/universe amd64 docker.io amd64 17.12.1-0ubuntu1 [30.1 MB]
Get:4 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 ubuntu-fan all 0.12.10 [34.7 kB]
Fetched 30.1 MB in 1s (24.4 MB/s)
```

We will get a prompt asking you to choose between y/n - choose *y*

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

$ sudo snap install docker

```
mca@S54:~$ sudo snap install docker
docker 20.10.14 from Canonical* installed
mca@S54:~$
```

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

$ docker –version

```
mca@S54:~$  docker --version
Docker version 17.12.1-ce, build 7390fc6
mca@S54:~$
```

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

$ sudo docker run hello-world

```
mca@S54:~$ 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/
```

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

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

$ sudo docker images

```
mca@S54:~$  sudo docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
hello-world         latest              feb5d9fea6a5        8 months ago        13.3kB
mca@S54:~$
```

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

$ sudo docker ps –a

```
mca@S54:~$ sudo docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                      PORTS               NAMES
911e0cff8c21        hello-world         "/hello"            34 seconds ago      Exited (0) 32 seconds ago                       condescending_
```

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

$ sudo docker ps

```
mca@S54:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
mca@S54:~$
```

We have successfully installed Docker on Ubuntu!