

**Name** - Manav Pahilwani

**Roll No** - 37

**Class** - D11AD

### **Experiment 8**

**Aim** - To study and Implement Containerization using Docker

**Theory** - Containerization is a popular approach to application deployment that allows applications to be packaged and run in a lightweight and isolated environment. Docker is a widely used platform for containerization that provides a set of tools and technologies for building, deploying, and managing containerized applications.

At its core, Docker is based on the concept of a container, which is an isolated environment that encapsulates an application and all of its dependencies. Each container runs its own operating system, providing a level of isolation and security that is not possible with traditional virtual machines. Containers can be easily moved between environments, making them ideal for modern, cloud-based applications that require rapid deployment and scaling.

Docker consists of several key components, including the Docker engine, which is responsible for building and running containers, and Docker Hub, which is a central repository for Docker images. Docker images are preconfigured containers that can be used as a starting point for building containerized applications.

Some key benefits of containerization using Docker include improved application portability, faster deployment times, and better resource utilization. Docker also provides a high degree of flexibility and control, allowing developers to easily manage and scale containerized applications across a variety of environments.

**Conclusion** - containerization using Docker is a powerful approach to application deployment that provides a range of benefits for modern, cloud-based applications. By leveraging the tools and technologies provided by Docker, developers can build and deploy containerized applications quickly and easily, while also ensuring a high degree of portability, flexibility, and security.

**Implementation** -

# Installing docker

```
EC2 Instance Connect x +
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-045169c7096af984a...
Personal Id CoinDCX - Crypto E... YouTube
AWS Services Q Search [Alt+S] N. Virginia ManavPahlwani

ubuntu@ip-172-31-52-173:~$ sudo apt-get remove docker docker-engine docker.io containerd runc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package 'docker.io' is not installed, so not removed
E: Unable to locate package docker
E: Unable to locate package docker-engine

ubuntu@ip-172-31-52-173:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 KB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [221 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [948 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [205 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [13.7 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [424 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [107 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [584 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [890 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [117 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [18.1 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [24.1 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [5312 B]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [444 B]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [40.7 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [9690 B]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [392 B]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [19.5 kB]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [14.0 kB]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [392 B]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [692 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [142 kB]

I-045169c7096af9849 (Docker)
PublicIPs: 100.25.204.129 PrivateIPs: 172.31.52.173

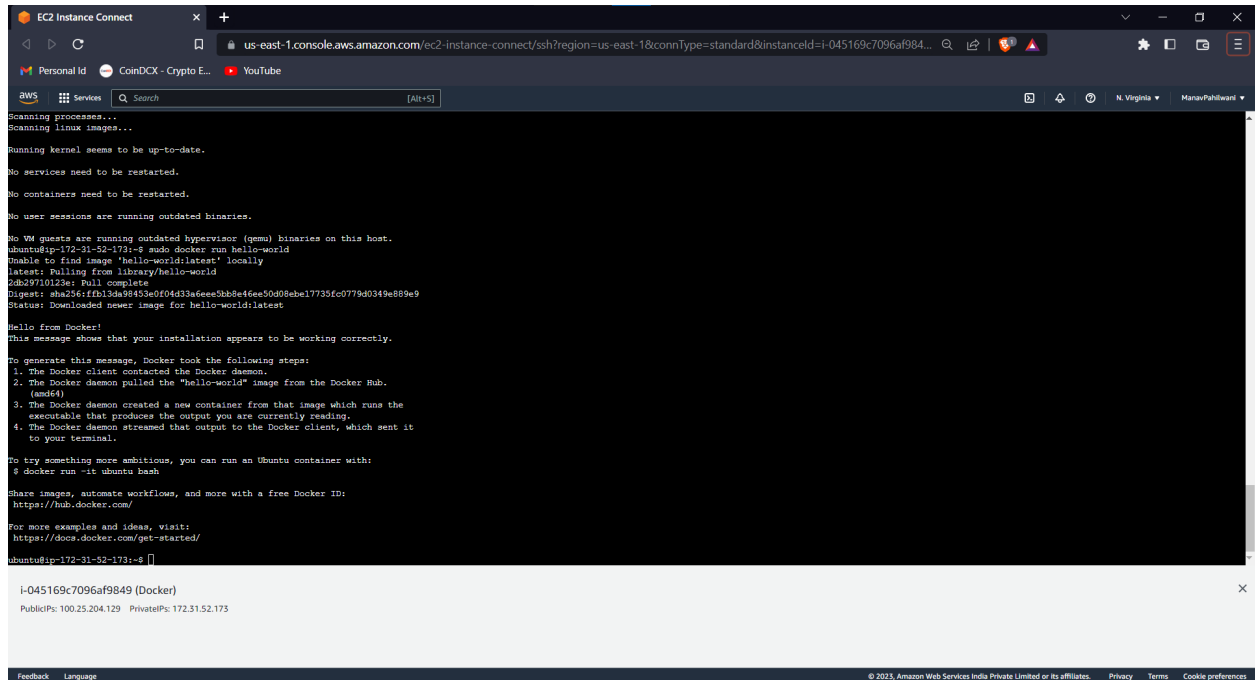
Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences
```

```
EC2 Instance Connect x +
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-045169c7096af984a...
Personal Id CoinDCX - Crypto E... YouTube
AWS Services Q Search [Alt+S] N. Virginia ManavPahlwani

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-52-173:~$ sudo mkdir -m 0755 -p /etc/apt/keyrings
ubuntu@ip-172-31-52-173:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
ubuntu@ip-172-31-52-173:~$ sudo \
  'deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable' | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
ubuntu@ip-172-31-52-173:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.9 kB]
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [13.6 kB]
Fetched 62.5 kB in 1s (62.0 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-52-173:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin libltdl7 liblrip0 pigs slurp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following new packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin libltdl7 liblrip0 pigs slurp4netns
0 upgraded, 11 newly installed, 0 to remove and 31 not upgraded.
Need to get 112 MB of archives.
After this operation, 401 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 pigs amd64 2.6-1 [63.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libltdl7 amd64 2.4.6-1build2 [39.6 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 liblrip0 amd64 4.6.1-1build1 [61.5 kB]
Get:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.18-1 [28.2 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 slurp4netns amd64 1.0.1-2 [28.2 kB]
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-buildx-plugin amd64 0.10.2-1-ubuntu.22.04-jammy [25.9 MB]
Get:7 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli amd64 5:23.0.1-1-ubuntu.22.04-jammy [19.2 MB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce amd64 5:23.0.1-1-ubuntu.22.04-jammy [22.0 MB]
Get:9 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-rootless-extras amd64 5:23.0.1-1-ubuntu.22.04-jammy [8760 kB]
Get:10 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-compose-plugin amd64 2.16.0-1-ubuntu.22.04-jammy [10.2 MB]
Get:11 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-scan-plugin amd64 0.23.0-ubuntu-jammy [1623 kB]

I-045169c7096af9849 (Docker)
PublicIPs: 100.25.204.129 PrivateIPs: 172.31.52.173

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences
```

A screenshot of a web browser window displaying an AWS EC2 Instance Connect terminal session. The browser's address bar shows the URL 'us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-045169c7096af984...'. The terminal output shows the following steps: scanning processes and Linux images, confirming the kernel is up-to-date, and checking for services, containers, and user sessions that need restarting. It then reports that no VM guests are running outdated hypervisor (qemu) binaries. The user runs 'sudo docker run hello-world'. The terminal shows it couldn't find the image locally, so it pulled 'latest' from the Docker Hub library. The pull is complete with digest 'sha256:ffb13da98453e0f04d33a6eee5bb8e46ee50d08ebe17735fc0779d0349e889e9'. The output is 'Hello from Docker!'. A detailed message explains the steps Docker took to generate this output. Finally, it suggests running 'docker run -it ubuntu bash' for a more ambitious task. A metadata box at the bottom of the terminal shows the instance ID 'i-045169c7096af9849 (Docker)' and IP addresses 'PublicIP: 100.25.204.129 PrivateIP: 172.31.52.173'. The browser's footer shows '© 2025, Amazon Web Services India Private Limited or its affiliates.' and links for Feedback, Language, Privacy, Terms, and Cookie preferences.

```
ubuntu@ip-172-31-52-173:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:ffb13da98453e0f04d33a6eee5bb8e46ee50d08ebe17735fc0779d0349e889e9
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/>

## Exploring Docker Commands :-

```
root@ip-172-31-52-173:/home/ubuntu# docker info
Client:
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.10.2
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.16.0
    Path:     /usr/libexec/docker/cli-plugins/docker-compose
  scan: Docker Scan (Docker Inc.)
    Version:  v0.23.0
    Path:     /usr/libexec/docker/cli-plugins/docker-scan

Server:
 Containers: 1
  Running: 0
  Paused: 0
  Stopped: 1
 Images: 1
 Server Version: 23.0.1
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Using metacopy: false
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: systemd
 Cgroup Version: 2
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: io.containerd.runc.v2 runc
 Default Runtime: runc
 Init Binary: docker-init
```

```
root@ip-172-31-52-173:/home/ubuntu# docker version
Client: Docker Engine - Community
 Version:           23.0.1
 API version:       1.42
 Go version:        go1.19.5
 Git commit:        a5ee5b1
 Built:             Thu Feb  9 19:47:01 2023
 OS/Arch:           linux/amd64
 Context:           default

Server: Docker Engine - Community
 Engine:
  Version:           23.0.1
  API version:       1.42 (minimum version 1.12)
  Go version:        go1.19.5
  Git commit:        bc3805a
  Built:             Thu Feb  9 19:47:01 2023
  OS/Arch:           linux/amd64
  Experimental:      false
 containerd:
  Version:           1.6.18
  GitCommit:        2456e983eb9e37e47538f59ea18f2043c9a73640
 runc:
  Version:           1.1.4
  GitCommit:        v1.1.4-0-g5fd4c4d
 docker-init:
  Version:           0.10.0
```

```
ubuntu@ip-172-31-52-173:~$ docker images
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://42fvar%2Frun%2Fdocker.sock/v1.24/images/json": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-52-173:~$ sudo su
root@ip-172-31-52-173:/home/ubuntu# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
hello-world    latest    feb5d9fea6a5  18 months ago  13.3kB
root@ip-172-31-52-173:/home/ubuntu#
```

```
root@ip-172-31-52-173:/home/ubuntu# docker ps --all
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS              PORTS          NAMES
bf0885a26a00   hello-world    "/hello"                23 minutes ago Exited (0) 23 minutes ago          epic_goldwasser
root@ip-172-31-52-173:/home/ubuntu#
```

```
root@ip-172-31-52-173:/home/ubuntu# docker ps --all
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS              PORTS          NAMES
bf0885a26a00   hello-world    "/hello"                23 minutes ago Exited (0) 23 minutes ago          epic_goldwasser
root@ip-172-31-52-173:/home/ubuntu# docker network ls
NETWORK ID     NAME          DRIVER   SCOPE
88d153fa1d15   bridge        bridge   local
b88d24b2cbb0   host          host     local
169267e87703   none          null     local
root@ip-172-31-52-173:/home/ubuntu#
```