

TASK 4: DOCKER

Submitted by :-

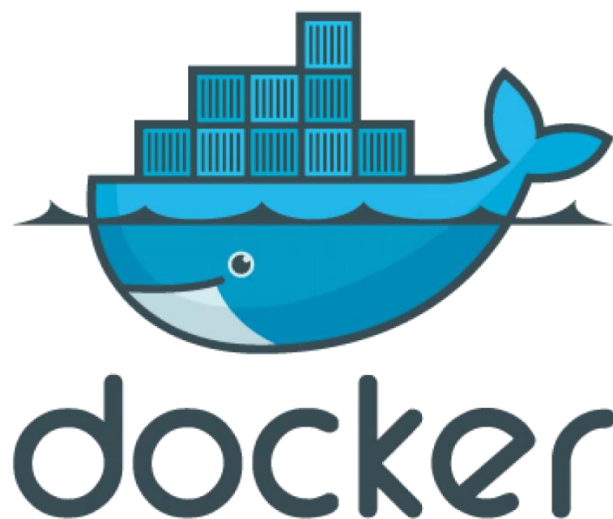
Aditya Kumar Verma

Guided by:

Ashish Mittal Sir

What is Docker?

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.



Docker features

Docker lets developers access these native containerization capabilities using simple commands, and automate them through a work-saving application programming interface (API). Compared to LXC, Docker offers:

- **Improved and seamless container portability:**

While LXC containers often reference machine-specific configurations, Docker containers run without modification across any desktop, data center and cloud environment.

- **Even lighter weight and more granular updates:**

With LXC, multiple processes can be combined within a single container. This makes it possible to build an application that can continue running while one of its parts is taken down for an update or repair.

- **Automated container creation:**

Docker can automatically build a container based on application source code.

- **Container versioning:**

Docker can track versions of a container image, roll back to previous versions, and trace who built a version and how. It can even upload only the deltas between an existing version and a new one.

- **Container reuse:**

Existing containers can be used as base images—essentially like templates for building new containers.

- **Shared container libraries:**

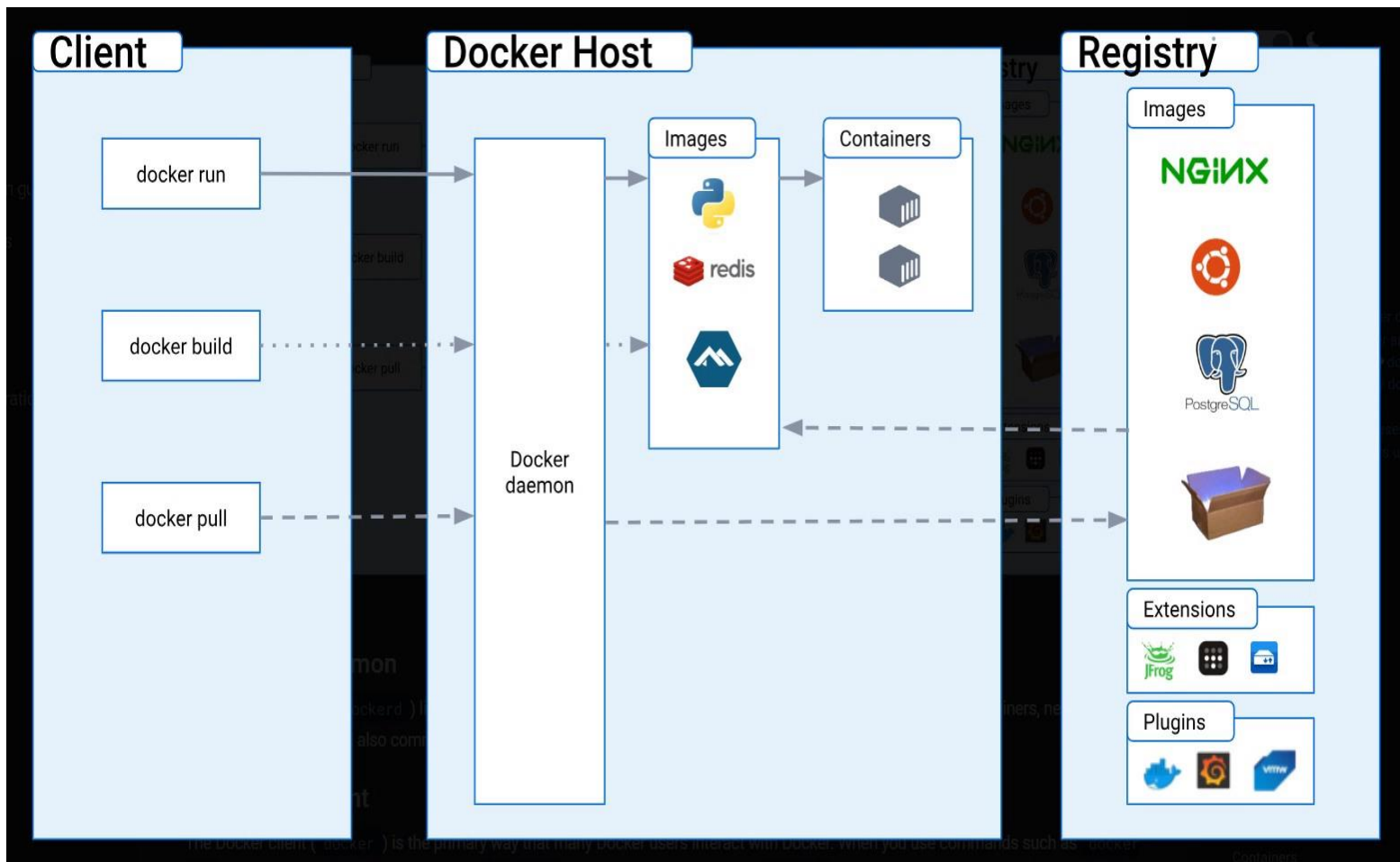
Developers can access an open-source registry containing thousands of user-contributed containers.

Today Docker containerization also works with Microsoft Windows and Apple MacOS. Developers can run Docker containers on any operating system, and most leading cloud providers, including Amazon Web Services (AWS), Microsoft Azure, and IBM Cloud offer specific services to help developers build, deploy and run applications containerized with Docker.



Docker Architecture

Docker uses a client-server architecture. The Docker client talks to the Docker daemon, which does the heavy lifting of building, running, and distributing your Docker containers. The Docker client and daemon can run on the same system, or you can connect a Docker client to a remote Docker daemon. The Docker client and daemon communicate using a REST API, over UNIX sockets or a network interface. Another Docker client is Docker Compose, that lets you work with applications consisting of a set of containers.



TASK 4 :

- **docker install**
- **service start**
- **docker pull**
- **docker run**
 - (1) os based image
 - (2) application based images:
 - (2.1)httpd (2.2)mysql

Step 1: Create a EC2 Instance on AWS and connect it to web server

Launch an instance | EC2 Mana x

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Gmail My groups YouTube LinkedIn Dashboard | EC2 M... Learn & Build Inspace/26thmay... OCI Dashboard OCI Certification CISCO Netacad Data Science Maste... Other bookmarks

aws Services Search [Alt+S]

EC2 > Instances > Launch an instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name
Docker_Practice1 [Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Li [Browse more AMIs](#)
aws Mac ubuntu Microsoft Red Hat SUS Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)
Amazon Linux 2023 AMI [Free tier eligible](#)

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.1.2...[read more](#)
ami-0b9ecf71fe947bbdd

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel [Launch instance](#) [Review commands](#)

CloudShell Feedback Language

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

Step 2: After launching an EC2 instance, connect it to the launched instance using command.

The screenshot displays the AWS Management Console interface. At the top, the browser address bar shows the URL: `ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:v=3;$case=tags:true%5Cclient:false;$regex=tags:false%5Cclient:false;sort=instanceState`. The console header includes the AWS logo, a search bar, and navigation links for various services. The left sidebar contains a navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Limits, Instances, Images, Elastic Block Store, and Network & Security.

The main content area is titled "Instances (1/8) Info". It features a table listing EC2 instances. The first instance, "Docker_Practice1", is highlighted in blue and has a status of "Running". The table columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 address, and Elastic IP.

Below the table, the details for the selected instance "Docker_Practice1" are shown. The "Details" tab is active, displaying the "Instance summary" section. This section includes the Instance ID, IPV6 address, Hostname type, IP name, Answer private resource DNS name, Auto-assigned IP address, and IAM Role. A tooltip indicates that the "Public IPv4 address" has been copied.

The "Monitoring" tab is also visible, showing the instance's state as "Running". The "Tags" tab is also present. The bottom of the console shows the footer with copyright information and links for Privacy, Terms, and Cookie preferences.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Docker_Practice1	i-0046350e77778aa25	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	ec2-13-232-16-153.ap-...	13.232.16.153	-
Apache2	i-05340bfc565a6de77	Stopped	t2.micro	-	No alarms	ap-south-1a	-	-	-
NGINX	i-06242677188df93fb	Stopped	t2.micro	-	No alarms	ap-south-1a	-	-	-
General Use 2.0	i-09f1029d9dee60e8b	Stopped	t2.micro	-	No alarms	ap-south-1a	-	-	-
General Use	i-0f042d81122a694cb	Stopped	t2.micro	-	No alarms	ap-south-1a	-	-	-
Apache	i-096d27b95d1c194c6	Stopped	t2.micro	-	No alarms	ap-south-1b	-	-	-
Docker_Practice	i-0e7a39c9e302b71e6	Stopped	t2.micro	-	No alarms	ap-south-1b	-	-	-
Don't Use	i-0fc19ed369c8eafea	Terminated	t2.micro	-	No alarms	ap-south-1b	-	-	-

Instance: i-0046350e77778aa25 (Docker_Practice1)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary Info

Instance ID: i-0046350e77778aa25 (Docker_Practice1)

IPV6 address: -

Hostname type: IP name: ip-172-31-5-49.ap-south-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Auto-assigned IP address: 13.232.16.153 [Public IP]

IAM Role: -

Instance state: Running

Private IP DNS name (IPv4 only): ip-172-31-5-49.ap-south-1.compute.internal

Instance type: t2.micro

VPC ID: vpc-0eab6d6e2b45cf919

Subnet ID: -

Private IPv4 addresses: 172.31.5.49

Public IPv4 DNS: ec2-13-232-16-153.ap-south-1.compute.amazonaws.com [open address]

Elastic IP addresses: -

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. [Learn more]

Auto Scaling Group name: -

Step 3: Install the Docker to your server

```
ec2-user@ip-172-31-5-49:~$ ssh -i Downloads\LnB.pem ec2-user@13.232.16.153
#_
##### Amazon Linux 2023
#####
##### \####
##### \###
##### \#/ --- https://aws.amazon.com/linux/amazon-linux-2023
##### V# ' ->
##### /m/'

[ec2-user@ip-172-31-5-49 ~]$ sudo yum update
Last metadata expiration check: 0:05:28 ago on Sat Jul 1 11:50:27 2023.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-5-49 ~]$ sudo yum install docker
Last metadata expiration check: 0:05:46 ago on Sat Jul 1 11:50:27 2023.
Dependencies resolved.

===== Package Arch Version Repository Size
=====Installing:
docker x86_64 20.10.23-1.amzn2023.0.1 amazonlinux 42 M

Installing dependencies:
containerd x86_64 1.6.19-1.amzn2023.0.1 amazonlinux 31 M
iptables-libse x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 401 k
iptables-nft x86_64 1.8.8-3.amzn2023.0.2 amazonlinux 183 k
libcgroup x86_64 3.0-1.amzn2023.0.1 amazonlinux 75 k
libnetfilter_conntrack x86_64 1.0.8-2.amzn2023.0.2 amazonlinux 58 k
libnftnl x86_64 1.0.1-19.amzn2023.0.2 amazonlinux 30 k
libnftnl x86_64 1.2.2-2.amzn2023.0.2 amazonlinux 84 k
pigz x86_64 2.5-1.amzn2023.0.3 amazonlinux 83 k
runc x86_64 1.1.7-1.amzn2023.0.1 amazonlinux 3.0 M

Transaction Summary
=====Install 10 Packages

Total download size: 77 M
Installed size: 300 M
Is this ok [y/N]: y
Downloading Packages:
(1/10): libnftnl-1.0.1-19.amzn2023.0.2.x86_64.rpm | 30 kB 00:00
(2/10): iptables-nft-1.8.8-3.amzn2023.0.2.x86_64.rpm | 183 kB 00:00
(3/10): libcgroup-3.0-1.amzn2023.0.1.x86_64.rpm | 75 kB 00:00
(4/10): libnftnl-1.2.2-2.amzn2023.0.2.x86_64.rpm | 84 kB 00:00
(5/10): runc-1.1.7-1.amzn2023.0.1.x86_64.rpm | 3.0 MB 00:00
(6/10): iptables-libse-1.8.8-3.amzn2023.0.2.x86_64.rpm | 401 kB 00:00
(7/10): pigz-2.5-1.amzn2023.0.3.x86_64.rpm | 83 kB 00:00
(8/10): libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64.rpm | 58 kB 00:00
(9/10): docker-20.10.23-1.amzn2023.0.1.x86_64.rpm | 42 MB 00:01
```

```
ec2-user@ip-172-31-5-49:~  
[ec2-user@ip-172-31-5-49 ~]$ sudo service docker start  
Redirecting to /bin/systemctl start docker.service  
[ec2-user@ip-172-31-5-49 ~]$ docker info  
Client:  
Context: default  
Debug Mode: false  
Plugins:  
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)  
  
Server:  
ERROR: Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/info": dial unix /var/run/dock  
er.sock: connect: permission denied  
errors pretty printing info  
[ec2-user@ip-172-31-5-49 ~]$ history  
  1 sudo yum update  
  2 sudo yum install docker  
  3 sudo service docker start  
  4 docker info  
  5 history  
[ec2-user@ip-172-31-5-49 ~]$ sudo usermod -a -G docker 13.232.16.153  
usermod: user '13.232.16.153' does not exist  
[ec2-user@ip-172-31-5-49 ~]$ sudo systemctl start docker  
[ec2-user@ip-172-31-5-49 ~]$ sudo systemctl enable docker  
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.  
[ec2-user@ip-172-31-5-49 ~]$ docker info  
Client:  
Context: default  
Debug Mode: false  
Plugins:  
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)  
  
Server:  
ERROR: Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/info": dial unix /var/run/dock  
er.sock: connect: permission denied  
errors pretty printing info  
[ec2-user@ip-172-31-5-49 ~]$ docker --version  
Docker version 20.10.23, build 7155243  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker info  
Client:  
Context: default  
Debug Mode: false  
Plugins:  
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)  
  
Server:  
Containers: 0  
  Running: 0  
  Paused: 0  
  Stopped: 0  
Images: 0
```

```
ec2-user@ip-172-31-5-49:~  
  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 io.containerd.runtime.v1.linux runc  
Default Runtime: runc  
Init Binary: docker-init  
containerd version: 1e1ea6e986c6c86565bc33d52e34b81b3e2bc71f  
runc version: f19387a6bec4944c770f7668ab51c4348d9c2f38  
init version: de40ad0  
Security Options:  
  seccomp  
    Profile: default  
  cgroupns  
Kernel Version: 6.1.34-58.102.amzn2023.x86_64  
Operating System: Amazon Linux 2023  
OSType: linux  
Architecture: x86_64  
CPUs: 1  
Total Memory: 949.8MiB  
Name: ip-172-31-5-49.ap-south-1.compute.internal  
ID: HX2U:VSQ0:ZXFG:7ZUH:W4FP:HVLN:VYRI:PGVV:7SAC:S72T:VCUW:HD4R  
Docker Root Dir: /var/lib/docker  
Debug Mode: false  
Registry: https://index.docker.io/v1/  
Labels:  
Experimental: false  
Insecure Registries:  
  127.0.0.0/8  
Live Restore Enabled: false  
  
[ec2-user@ip-172-31-5-49 ~]$ history  
  1 sudo yum update  
  2 sudo yum install docker  
  3 sudo service docker start  
  4 docker info  
  5 history  
  6 sudo usermod -a -G docker 13.232.16.153  
  7 sudo systemctl start docker  
  8 sudo systemctl enable docker  
  9 docker info  
 10 docker --version  
 11 sudo docker info  
 12 history  
[ec2-user@ip-172-31-5-49 ~]$ |
```

--- Docker service successfully installed ---

Step 4: Start the Docker service

```
ec2-user@ip-172-31-5-49:~  
[ec2-user@ip-172-31-5-49 ~]$ sudo systemctl start docker  
[ec2-user@ip-172-31-5-49 ~]$ sudo systemctl enable docker  
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.  
[ec2-user@ip-172-31-5-49 ~]$ docker info  
Client:  
Context: default  
Debug Mode: false  
Plugins:  
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)  
  
Server:  
ERROR: Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/info": dial unix /var/run/dock  
er.sock: connect: permission denied  
errors pretty printing info  
[ec2-user@ip-172-31-5-49 ~]$ docker --version  
Docker version 20.10.23, build 7155243  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker info  
Client:  
Context: default  
Debug Mode: false  
Plugins:  
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)  
  
Server:  
Containers: 0  
  Running: 0  
  Paused: 0  
  Stopped: 0  
Images: 0  
Server Version: 20.10.23  
Storage Driver: overlay2  
  Backing Filesystem: xfs  
  Supports d_type: true  
  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 io.containerd.runtime.v1.linux runc  
Default Runtime: runc  
Init Binary: docker-init  
containerd version: 1e1ea6e986c6c86565bc33d52e34b81b3e2bc71f  
runc version: f19387a6bec4944c770f7668ab51c4348d9c2f38  
init version: de40ad0  
Security Options:
```

```
Command Prompt  
init version: de40ad0  
Security Options:  
  seccomp  
    Profile: default  
  cgroupns  
Kernel Version: 6.1.34-58.102.amzn2023.x86_64  
Operating System: Amazon Linux 2023  
OSType: linux  
Architecture: x86_64  
CPUs: 1  
Total Memory: 949.8MiB  
Name: ip-172-31-5-49.ap-south-1.compute.internal  
ID: RX2U:VSQD:ZXFG:7ZUH:W4FP:HVLN:VYRI:PGVV:7SAC:S72T:VCUW:HD4R  
Docker Root Dir: /var/lib/docker  
Debug Mode: false  
Registry: https://index.docker.io/v1/  
Labels:  
Experimental: false  
Insecure Registries:  
  127.0.0.0/8  
Live Restore Enabled: false  
  
[ec2-user@ip-172-31-5-49 ~]$ history  
  1 sudo yum update  
  2 sudo yum install docker  
  3 sudo service docker start  
  4 docker info  
  5 history  
  6 sudo usermod -a -G docker 13.232.16.153  
  7 sudo systemctl start docker  
  8 sudo systemctl enable docker  
  9 docker info  
 10 docker --version  
 11 sudo docker info  
 12 history
```

Step 5: Docker Pull & Run OS Based Image - Amazon Linux

```
Command Prompt
[ec2-user@ip-172-31-5-49 ~]$ sudo docker pull amazonlinux
Using default tag: latest
latest: Pulling from library/amazonlinux
a802d1401e24: Pull complete
Digest: sha256:28004508959f548dd98278f746c577a41b8340dc295e51d309e721e4e217bb85
Status: Downloaded newer image for amazonlinux:latest
docker.io/library/amazonlinux:latest
[ec2-user@ip-172-31-5-49 ~]$ sudo docker run -it amazonlinux
bash-5.2# cat /etc/os-release
NAME="Amazon Linux"
VERSION="2023"
ID="amzn"
ID_LIKE="fedora"
VERSION_ID="2023"
PLATFORM_ID="platform:al2023"
PRETTY_NAME="Amazon Linux 2023"
ANSI_COLOR="0;33"
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2023"
HOME_URL="https://aws.amazon.com/linux/"
BUG_REPORT_URL="https://github.com/amazonlinux/amazon-linux-2023"
SUPPORT_END="2028-03-01"
bash-5.2# yum install
usage: yum install [-c [config file]] [-q] [-v] [--version] [--installroot [path]]
                  [--nodocs] [--noplugins] [--enableplugin [plugin]]
                  [--disableplugin [plugin]] [--releasever RELEASEVER]
                  [--setopt SETOPTS] [--skip-broken] [-h] [--allowerase]
                  [-b | --nobest] [-C] [-R [minutes]] [-d [debug level]] [--debugsolver]
                  [--showduplicates] [-e ERRORLEVEL] [--obsoletes]
                  [--rpmverbosity [debug level name]] [-y] [--assumeno]
                  [--enablerepo [repo]] [--disablerepo [repo] | --repo [repo]]
                  [--enable | --disable] [-x [package]] [--disableexcludes [repo]]
                  [--repofrompath [repo,path]] [--noautoremove] [--nogpgcheck]
                  [--color COLOR] [--refresh] [-4] [-6] [--destdir DESTDIR]
                  [--downloadonly] [--comment COMMENT] [--bugfix] [--enhancement]
                  [--newpackage] [--security] [--advisory ADVISORY] [--bz BUGZILLA]
                  [--cve CVES] [--sec-severity {Critical,Important,Moderate,Medium,Low}]
                  [--forcearch ARCH]
                  PACKAGE [PACKAGE ...]
yum install: error: the following arguments are required: PACKAGE
bash-5.2# yum install httpd
Amazon Linux 2023 repository                      9.1 MB/s | 14 MB      00:01
Last metadata expiration check: 0:00:04 ago on Sat Jul 1 12:15:51 2023.
Dependencies resolved.
===== Package Arch Version Repository Size
Installing:
httpd x86_64 2.4.56-1.amzn2023 amazonlinux 48 k
Installing dependencies:
apr x86_64 1.7.2-2.amzn2023.0.2 amazonlinux 129 k
apr-util x86_64 1.6.3-1.amzn2023.0.1 amazonlinux 98 k
cracklib x86_64 2.9.6-27.amzn2023.0.2 amazonlinux 82 k

cracklib x86_64 2.9.6-27.amzn2023.0.2 amazonlinux 82 k
dbus x86_64 1:1.12.24-1.amzn2023.0.2 amazonlinux 8.3 k
dbus-broker x86_64 32-1.amzn2023.0.2 amazonlinux 174 k
dbus-common noarch 1:1.12.24-1.amzn2023.0.2 amazonlinux 15 k
device-mapper x86_64 1.02.185-1.amzn2023.0.4 amazonlinux 140 k
device-mapper-libs x86_64 1.02.185-1.amzn2023.0.4 amazonlinux 180 k
generic-logos-httpd noarch 18.0.0-12.amzn2023.0.3 amazonlinux 19 k
gzip x86_64 1.12-1.amzn2023.0.1 amazonlinux 160 k
httpd-core x86_64 2.4.56-1.amzn2023 amazonlinux 1.4 M
httpd-filesystem noarch 2.4.56-1.amzn2023 amazonlinux 15 k
httpd-tools x86_64 2.4.56-1.amzn2023 amazonlinux 82 k
kmod-libs x86_64 29-2.amzn2023.0.5 amazonlinux 62 k
libargon2 x86_64 20171227-9.amzn2023.0.2 amazonlinux 29 k
libbrotli x86_64 1.0.9-4.amzn2023.0.2 amazonlinux 315 k
libbrotli-libs x86_64 5.3.28-49.amzn2023.0.2 amazonlinux 756 k
libbconf x86_64 0.4.0-1.amzn2023.0.2 amazonlinux 28 k
libfdisk x86_64 2.37.4-1.amzn2023.0.3 amazonlinux 154 k
libpwquality x86_64 1.4.4-6.amzn2023.0.2 amazonlinux 106 k
libseccomp x86_64 2.5.3-1.amzn2023.0.2 amazonlinux 71 k
libsemanage x86_64 3.4-5.amzn2023.0.2 amazonlinux 121 k
libutempter x86_64 1.2.1-4.amzn2023.0.2 amazonlinux 26 k
mailcap noarch 2.1.49-3.amzn2023.0.3 amazonlinux 33 k
pam x86_64 1.5.1-8.amzn2023.0.3 amazonlinux 543 k
shadow-utils x86_64 2:4.9-12.amzn2023.0.2 amazonlinux 1.1 M
systemd x86_64 252.4-1161.amzn2023.0.4 amazonlinux 4.2 M
systemd-libs x86_64 252.4-1161.amzn2023.0.4 amazonlinux 629 k
systemd-pam x86_64 252.4-1161.amzn2023.0.4 amazonlinux 332 k
util-linux x86_64 2.37.4-1.amzn2023.0.3 amazonlinux 2.2 M
util-linux-core x86_64 2.37.4-1.amzn2023.0.3 amazonlinux 433 k
xkeyboard-config noarch 2.33-1.amzn2023.0.2 amazonlinux 779 k
Installing weak dependencies:
apr-util-openssl x86_64 1.6.3-1.amzn2023.0.1 amazonlinux 17 k
cryptsetup-libs x86_64 2.6.1-1.amzn2023.0.1 amazonlinux 489 k
diffutils x86_64 3.8-1.amzn2023.0.2 amazonlinux 370 k
libxkbcommon x86_64 1.3.0-1.amzn2023.0.2 amazonlinux 141 k
mod_http2 x86_64 2.0.11-2.amzn2023 amazonlinux 150 k
mod_lua x86_64 2.4.56-1.amzn2023 amazonlinux 62 k
qrencode-libs x86_64 4.1.1-2.amzn2023.0.2 amazonlinux 66 k
systemd-networkd x86_64 252.4-1161.amzn2023.0.4 amazonlinux 621 k
systemd-resolved x86_64 252.4-1161.amzn2023.0.4 amazonlinux 284 k

Transaction Summary
=====Install 42 Packages

Total download size: 16 M
Installed size: 57 M
Is this ok [y/N]: y
Downloading Packages:
(1/42): libsemanage-3.4-5.amzn2023.0.2.x86_64.rpm 2.1 MB/s | 121 kB 00:00
(2/42): systemd-resolved-252.4-1161.amzn2023.0.4.x86_64.rpm 11 MB/s | 284 kB 00:00
```

```
Command Prompt
(2/42): systemd-resolved-252.4-1161.amzn2023.0.4.x86_64.rpm 11 MB/s 284 kB 00:00
(3/42): device-mapper-1.02.185-1.amzn2023.0.4.x86_64.rpm 7.5 MB/s 140 kB 00:00
(4/42): libdb-5.3.28-49.amzn2023.0.2.x86_64.rpm 46 MB/s 756 kB 00:00
(5/42): dbus-1.12.24-1.amzn2023.0.2.x86_64.rpm 8.3 kB/s 8.3 kB 00:01
(6/42): gzip-1.12-1.amzn2023.0.1.x86_64.rpm 11 MB/s 160 kB 00:00
(7/42): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm 2.9 MB/s 98 kB 00:00
(8/42): libfdisk-2.37.4-1.amzn2023.0.3.x86_64.rpm 14 MB/s 154 kB 00:00
(9/42): libeconf-0.4.0-1.amzn2023.0.2.x86_64.rpm 2.4 MB/s 28 kB 00:00
(10/42): httpd-2.4.56-1.amzn2023.x86_64.rpm 40 kB/s 48 kB 00:01
(11/42): qrencode-libs-4.1.1-2.amzn2023.0.2.x86_64.rpm 5.2 MB/s 66 kB 00:00
(12/42): apr-1.7.2-2.amzn2023.0.2.x86_64.rpm 13 MB/s 129 kB 00:00
(13/42): pam-1.5.1-8.amzn2023.0.3.x86_64.rpm 43 MB/s 543 kB 00:00
(14/42): systemd-pam-252.4-1161.amzn2023.0.4.x86_64.rpm 162 kB/s 332 kB 00:02
(15/42): libargon2-20171227-9.amzn2023.0.2.x86_64.rpm 1.3 MB/s 29 kB 00:00
(16/42): device-mapper-libs-1.02.185-1.amzn2023.0.4.x86_64 105 kB/s 180 kB 00:01
(17/42): libutempter-1.2.1-4.amzn2023.0.2.x86_64.rpm 527 kB/s 26 kB 00:00
(18/42): util-linux-core-2.37.4-1.amzn2023.0.3.x86_64.rpm 32 MB/s 433 kB 00:00
(19/42): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm 26 MB/s 315 kB 00:00
(20/42): shadow-utils-4.9-12.amzn2023.0.2.x86_64.rpm 41 MB/s 1.1 MB 00:00
(21/42): httpd-core-2.4.56-1.amzn2023.x86_64.rpm 623 kB/s 1.4 MB 00:02
(22/42): httpd-tools-2.4.56-1.amzn2023.x86_64.rpm 58 kB/s 82 kB 00:01
(23/42): dbus-broker-32-1.amzn2023.0.2.x86_64.rpm 15 MB/s 174 kB 00:00
(24/42): diffutils-3.8-1.amzn2023.0.2.x86_64.rpm 24 MB/s 370 kB 00:00
(25/42): cracklib-2.9.6-27.amzn2023.0.2.x86_64.rpm 12 MB/s 82 kB 00:00
(26/42): libseccomp-2.5.3-1.amzn2023.0.2.x86_64.rpm 7.5 MB/s 71 kB 00:00
(27/42): mod_httpd-2.0.11-2.amzn2023.x86_64.rpm 88 kB/s 150 kB 00:01
(28/42): systemd-libs-252.4-1161.amzn2023.0.4.x86_64.rpm 43 MB/s 629 kB 00:00
(29/42): systemd-networkd-252.4-1161.amzn2023.0.4.x86_64.r 289 kB/s 621 kB 00:02
(30/42): systemd-252.4-1161.amzn2023.0.4.x86_64.rpm 56 MB/s 4.2 MB 00:00
(31/42): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm 472 kB/s 17 kB 00:00
(32/42): libklibc-common-1.3.0-1.amzn2023.0.2.x86_64.rpm 82 kB/s 141 kB 00:01
(33/42): kmod-libs-29-2.amzn2023.0.5.x86_64.rpm 6.1 MB/s 62 kB 00:00
(34/42): libpwquality-1.4.4-6.amzn2023.0.2.x86_64.rpm 8.8 MB/s 106 kB 00:00
(35/42): util-linux-2.37.4-1.amzn2023.0.3.x86_64.rpm 63 MB/s 2.2 MB 00:00
(36/42): mod_lua-2.4.56-1.amzn2023.x86_64.rpm 69 kB/s 62 kB 00:00
(37/42): httpd-filesystem-2.4.56-1.amzn2023.noarch.rpm 16 kB/s 15 kB 00:00
(38/42): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 20 kB/s 19 kB 00:00
(39/42): cryptsetup-libs-2.6.1-1.amzn2023.0.1.x86_64.rpm 263 kB/s 489 kB 00:01
(40/42): mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm 28 kB/s 33 kB 00:01
(41/42): dbus-common-1.12.24-1.amzn2023.0.2.noarch.rpm 17 kB/s 15 kB 00:00
(42/42): xkeyboard-config-2.33-1.amzn2023.0.2.noarch.rpm 370 kB/s 779 kB 00:02
Total 1.6 MB/s | 16 MB 00:10

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Installing : systemd-libs-252.4-1161.amzn2023.0.4.x86_64 1/42
  Installing : apr-1.7.2-2.amzn2023.0.2.x86_64 2/42
```

```
Command Prompt
Installing : systemd-libs-252.4-1161.amzn2023.0.4.x86_64 1/42
Installing : apr-1.7.2-2.amzn2023.0.2.x86_64 2/42
Installing : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 3/42
Installing : apr-util-1.6.3-1.amzn2023.0.1.x86_64 4/42
Installing : util-linux-core-2.37.4-1.amzn2023.0.3.x86_64 5/42
Running scriptlet: util-linux-core-2.37.4-1.amzn2023.0.3.x86_64 5/42
Installing : mailcap-2.1.49-3.amzn2023.0.3.noarch 6/42
Installing : libfdisk-2.37.4-1.amzn2023.0.3.x86_64 7/42
Installing : httpd-tools-2.4.56-1.amzn2023.x86_64 8/42
Installing : xkeyboard-config-2.33-1.amzn2023.0.2.noarch 9/42
Installing : libklibc-common-1.3.0-1.amzn2023.0.2.x86_64 10/42
Installing : dbus-common-1:1.12.24-1.amzn2023.0.2.noarch 11/42
Running scriptlet: dbus-common-1:1.12.24-1.amzn2023.0.2.noarch 11/42
Installing : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 12/42
Installing : kmod-libs-29-2.amzn2023.0.5.x86_64 13/42
Installing : libseccomp-2.5.3-1.amzn2023.0.2.x86_64 14/42
Installing : diffutils-3.8-1.amzn2023.0.2.x86_64 15/42
Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 16/42
Installing : libargon2-20171227-9.amzn2023.0.2.x86_64 17/42
Installing : qrencode-libs-4.1.1-2.amzn2023.0.2.x86_64 18/42
Installing : libeconf-0.4.0-1.amzn2023.0.2.x86_64 19/42
Installing : gzip-1.12-1.amzn2023.0.1.x86_64 20/42
Installing : cracklib-2.9.6-27.amzn2023.0.2.x86_64 21/42
Installing : libdb-5.3.28-49.amzn2023.0.2.x86_64 22/42
Installing : libpwquality-1.4.4-6.amzn2023.0.2.x86_64 23/42
Installing : pam-1.5.1-8.amzn2023.0.3.x86_64 24/42
Installing : libsemanage-3.4-5.amzn2023.0.2.x86_64 25/42
Installing : shadow-utils-2:4.9-12.amzn2023.0.2.x86_64 26/42
Running scriptlet: libutempter-1.2.1-4.amzn2023.0.2.x86_64 27/42
Installing : libutempter-1.2.1-4.amzn2023.0.2.x86_64 27/42
Installing : util-linux-2.37.4-1.amzn2023.0.3.x86_64 28/42
warning: /etc/adjtime created as /etc/adjtime.rpmnew

Running scriptlet: util-linux-2.37.4-1.amzn2023.0.3.x86_64 28/42
Running scriptlet: dbus-broker-32-1.amzn2023.0.2.x86_64 29/42
Installing : dbus-broker-32-1.amzn2023.0.2.x86_64 29/42
Running scriptlet: dbus-broker-32-1.amzn2023.0.2.x86_64 29/42
Installing : dbus-1:1.12.24-1.amzn2023.0.2.x86_64 30/42
Installing : systemd-resolved-252.4-1161.amzn2023.0.4.x86_64 31/42
Running scriptlet: systemd-resolved-252.4-1161.amzn2023.0.4.x86_64 31/42
Installing : cryptsetup-libs-2.6.1-1.amzn2023.0.1.x86_64 32/42
Installing : device-mapper-libs-1.02.185-1.amzn2023.0.4.x86_64 33/42
Installing : device-mapper-1.02.185-1.amzn2023.0.4.x86_64 34/42
Installing : systemd-networkd-252.4-1161.amzn2023.0.4.x86_64 35/42
Running scriptlet: systemd-networkd-252.4-1161.amzn2023.0.4.x86_64 35/42
Installing : systemd-252.4-1161.amzn2023.0.4.x86_64 36/42
Running scriptlet: systemd-252.4-1161.amzn2023.0.4.x86_64 36/42
Creating group 'input' with GID 104.
Creating group 'kvm' with GID 36.
Creating group 'render' with GID 105.
```

```
Command Prompt
Creating group 'render' with GID 185.
Creating group 'sgx' with GID 186.
Creating group 'systemd-journal' with GID 190.
Creating group 'systemd-network' with GID 192.
Creating user 'systemd-network' (systemd Network Management) with UID 192 and GID 192.
Creating group 'systemd-oom' with GID 999.
Creating user 'systemd-oom' (systemd Userspace OOM Killer) with UID 999 and GID 999.
Creating group 'systemd-resolve' with GID 193.
Creating user 'systemd-resolve' (systemd Resolver) with UID 193 and GID 193.

Installing      : systemd-pam-252.4-1161.amzn2023.0.4.x86_64      37/42
Running scriptlet: httpd-filesystem-2.4.56-1.amzn2023.noarch      38/42
Installing      : httpd-filesystem-2.4.56-1.amzn2023.noarch      38/42
Installing      : httpd-core-2.4.56-1.amzn2023.x86_64            39/42
Installing      : mod_httpd-2.0.11-2.amzn2023.x86_64             40/42
Installing      : mod_lua-2.4.56-1.amzn2023.x86_64               41/42
Installing      : httpd-2.4.56-1.amzn2023.x86_64                42/42
Running scriptlet: httpd-2.4.56-1.amzn2023.x86_64                42/42
Running scriptlet: pam-1.5.1-8.amzn2023.0.3.x86_64              42/42
Running scriptlet: systemd-resolved-252.4-1161.amzn2023.0.4.x86_64 42/42
Running scriptlet: httpd-2.4.56-1.amzn2023.x86_64                42/42
Verifying        : systemd-pam-252.4-1161.amzn2023.0.4.x86_64      1/42
Verifying        : httpd-2.4.56-1.amzn2023.x86_64                2/42
Verifying        : libsemanage-3.4-5.amzn2023.0.2.x86_64         3/42
Verifying        : systemd-resolved-252.4-1161.amzn2023.0.4.x86_64 4/42
Verifying        : device-mapper-1.02.185-1.amzn2023.0.4.x86_64   5/42
Verifying        : libdb-5.3.28-49.amzn2023.0.2.x86_64           6/42
Verifying        : dbus-1:1.12.24-1.amzn2023.0.2.x86_64          7/42
Verifying        : gzip-1.12-1.amzn2023.0.1.x86_64               8/42
Verifying        : apr-util-1.6.3-1.amzn2023.0.1.x86_64           9/42
Verifying        : libfdisk-2.37.4-1.amzn2023.0.3.x86_64        10/42
Verifying        : libecconf-0.4.0-1.amzn2023.0.2.x86_64        11/42
Verifying        : qrencode-libs-4.1.1-2.amzn2023.0.2.x86_64     12/42
Verifying        : httpd-core-2.4.56-1.amzn2023.x86_64          13/42
Verifying        : apr-1.7.2-2.amzn2023.0.2.x86_64              14/42
Verifying        : pam-1.5.1-8.amzn2023.0.3.x86_64              15/42
Verifying        : device-mapper-libs-1.02.185-1.amzn2023.0.4.x86_64 16/42
Verifying        : libargon2-20171227-9.amzn2023.0.2.x86_64     17/42
Verifying        : httpd-tools-2.4.56-1.amzn2023.x86_64         18/42
Verifying        : libutempter-1.2.1-4.amzn2023.0.2.x86_64      19/42
Verifying        : util-linux-core-2.37.4-1.amzn2023.0.3.x86_64 20/42
Verifying        : libbrotli-1.0.9-4.amzn2023.0.2.x86_64        21/42
Verifying        : shadow-utils-2:4.9-12.amzn2023.0.2.x86_64    22/42
Verifying        : systemd-networkd-252.4-1161.amzn2023.0.4.x86_64 23/42
Verifying        : mod_httpd-2.0.11-2.amzn2023.x86_64           24/42
Verifying        : dbus-broker-32-1.amzn2023.0.2.x86_64         25/42
Verifying        : diffutils-3.8-1.amzn2023.0.2.x86_64          26/42
Verifying        : crackmapexec-2.9.6-27.amzn2023.0.2.x86_64    27/42
Verifying        : libseccomp-2.5.3-1.amzn2023.0.2.x86_64       28/42
Verifying        : libxkbcommon-1.3.0-1.amzn2023.0.2.x86_64     29/42
```

```
Command Prompt
Verifying        : dbus-common-1:1.12.24-1.amzn2023.0.2.noarch    41/42
Verifying        : xkeyboard-config-2.33-1.amzn2023.0.2.noarch   42/42

Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-1.6.3-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
crackmapexec-2.9.6-27.amzn2023.0.2.x86_64
cryptsetup-libs-2.6.1-1.amzn2023.0.1.x86_64
dbus-1:1.12.24-1.amzn2023.0.2.x86_64
dbus-broker-32-1.amzn2023.0.2.x86_64
dbus-common-1:1.12.24-1.amzn2023.0.2.noarch
device-mapper-1.02.185-1.amzn2023.0.4.x86_64
device-mapper-libs-1.02.185-1.amzn2023.0.4.x86_64
diffutils-3.8-1.amzn2023.0.2.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
gzip-1.12-1.amzn2023.0.1.x86_64
httpd-2.4.56-1.amzn2023.x86_64
httpd-core-2.4.56-1.amzn2023.x86_64
httpd-filesystem-2.4.56-1.amzn2023.noarch
httpd-tools-2.4.56-1.amzn2023.x86_64
kmod-libs-29-2.amzn2023.0.5.x86_64
libargon2-20171227-9.amzn2023.0.2.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
libdb-5.3.28-49.amzn2023.0.2.x86_64
libecconf-0.4.0-1.amzn2023.0.2.x86_64
libfdisk-2.37.4-1.amzn2023.0.3.x86_64
libpwquality-1.4.4-6.amzn2023.0.2.x86_64
libseccomp-2.5.3-1.amzn2023.0.2.x86_64
libsemanage-3.4-5.amzn2023.0.2.x86_64
libutempter-1.2.1-4.amzn2023.0.2.x86_64
libxkbcommon-1.3.0-1.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_httpd-2.0.11-2.amzn2023.x86_64
mod_lua-2.4.56-1.amzn2023.x86_64
pam-1.5.1-8.amzn2023.0.3.x86_64
qrencode-libs-4.1.1-2.amzn2023.0.2.x86_64
shadow-utils-2:4.9-12.amzn2023.0.2.x86_64
systemd-252.4-1161.amzn2023.0.4.x86_64
systemd-libs-252.4-1161.amzn2023.0.4.x86_64
systemd-networkd-252.4-1161.amzn2023.0.4.x86_64
systemd-pam-252.4-1161.amzn2023.0.4.x86_64
systemd-resolved-252.4-1161.amzn2023.0.4.x86_64
util-linux-2.37.4-1.amzn2023.0.3.x86_64
util-linux-core-2.37.4-1.amzn2023.0.3.x86_64
xkeyboard-config-2.33-1.amzn2023.0.2.noarch

Complete!
bash-5.2# history
 1 cat /etc/os-release
 2 yum install
 3 yum install httpd
 4 history
```


Step 6: Docker Pull & Run Application Based Images for httpd

```
ec2-user@ip-172-31-5-49:~  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker pull httpd  
Using default tag: latest  
latest: Pulling from library/httpd  
5b5fe70539cd: Pull complete  
1d40567696ba: Pull complete  
d6cb3e372b06: Pull complete  
fbc80fe62958: Pull complete  
3da840f9e96e: Pull complete  
Digest: sha256:f499227681dfff576d6ae8c49550c57f11970b358ee720bb8557b9fa7daf3a06d  
Status: Downloaded newer image for httpd:latest  
docker.io/library/httpd:latest  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker run -d -p 80:80 httpd  
6fa3e01eeb57d57904b8a18cd4a671f60989ece5eae514a2a65dc8ff4f53817c  
[ec2-user@ip-172-31-5-49 ~]$
```

Step 7: Docker Pull & Run Application Based Images for mysql

```
[ec2-user@ip-172-31-5-49 ~]$ sudo docker pull mysql  
Using default tag: latest  
latest: Pulling from library/mysql  
46ef68baacb7: Pull complete  
94c1114b2e9c: Pull complete  
ff05e3f38802: Pull complete  
41cc3fcd9912: Pull complete  
07bbc8bdf52a: Pull complete  
6d88f83726a9: Pull complete  
cf5c7d5d33f7: Pull complete  
9db3175a2a66: Pull complete  
feadeb27fa9: Pull complete  
cf91e7784414: Pull complete  
b1770db1c329: Pull complete  
Digest: sha256:15f069202c46cf861ce429423ae3f8dfa6423306fbf399eaeef36094ce30dd75c  
Status: Downloaded newer image for mysql:latest  
docker.io/library/mysql:latest  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker run -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD=mysqlpassword mysql  
399aaf35b5b73625f8a706c645791f8132b7330762c177f0e61123d60c39aeff  
[ec2-user@ip-172-31-5-49 ~]$ sudo docker images  
REPOSITORY TAG IMAGE ID CREATED SIZE  
amazonlinux latest 24a9e1bb56d4 10 days ago 144MB  
mysql latest 91b53e2624b4 2 weeks ago 565MB  
httpd latest ad303d7f80f9 2 weeks ago 168MB  
[ec2-user@ip-172-31-5-49 ~]$
```


Step 8: Docker Info.

```
ec2-user@ip-172-31-5-49:~$ sudo docker info
Client:
Context:      default
Debug Mode: false
Plugins:
  buildx: Docker Buildx (Docker Inc., 0.0.0+unknown)
Server:
Containers: 3
  Running: 2
  Paused: 0
  Stopped: 1
Images: 3
Server Version: 20.10.23
Storage Driver: overlay2
  Backing Filesystem: xfs
  Supports d_type: true
  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 io.containerd.runtime.v1.linux runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 1e1ea6e986c6c86565bc33d52e34b81b3e2bc71f
runc version: f19387a6bec4944c77b6f7668ab51c4348d9c2f38
init version: de40ad0
Security Options:
  seccomp
   Profile: default
  cgroupns
Kernel Version: 6.1.34-58.102.amzn2023.x86_64
Operating System: Amazon Linux 2023
OSType: linux
Architecture: x86_64
CPUs: 1
Total Memory: 949.8MiB
Name: ip-172-31-5-49.ap-south-1.compute.internal
ID: KX2U:VSQO:ZXFG:7ZUH:W4FP:HVLN:VYRI:PGVV:7SAC:S72T:VCUW:HD4R
Docker Root Dir: /var/lib/docker
Debug Mode: false
Registry: https://index.docker.io/v1/
Labels:
Experimental: false
Insecure Registries:
  127.0.0.0/8
Live Restore Enabled: false
```

Step 9: Docker History (so far)

```
[ec2-user@ip-172-31-5-49 ~]$ history
 1 sudo yum update
 2 sudo yum install docker
 3 sudo service docker start
 4 docker info
 5 history
 6 sudo usermod -a -G docker 13.232.16.153
 7 sudo systemctl start docker
 8 sudo systemctl enable docker
 9 docker info
10 docker --version
11 sudo docker info
12 history
13 sudo docker pull amazonlinux
14 sudo docker run -it amazonlinux
15 docker --version
16 cls
17 clear
18 sudo docker pull httpd
19 sudo docker run -d -p 80:80 httpd
20 sudo docker pull mysql
21 sudo docker run -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD=mysqlpassword mysql
22 sudo docker images
23 sudo docker info
24 exit
25 history
26 exit
27 history
[ec2-user@ip-172-31-5-49 ~]$
```

----- COMPLETED -----