

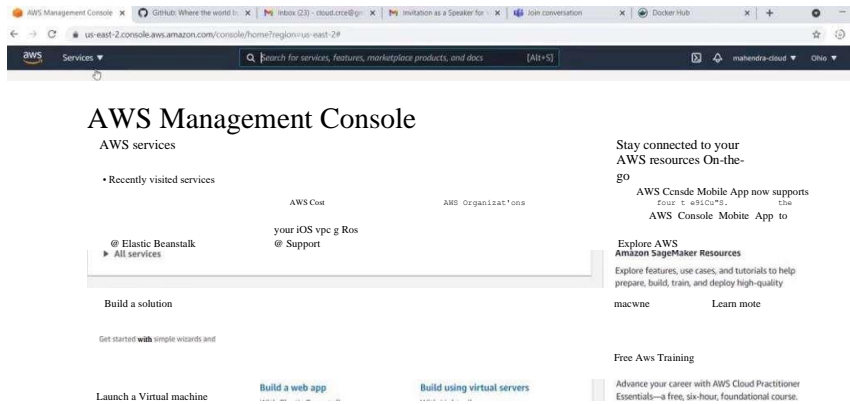
Experiment No.8

Aim: To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.

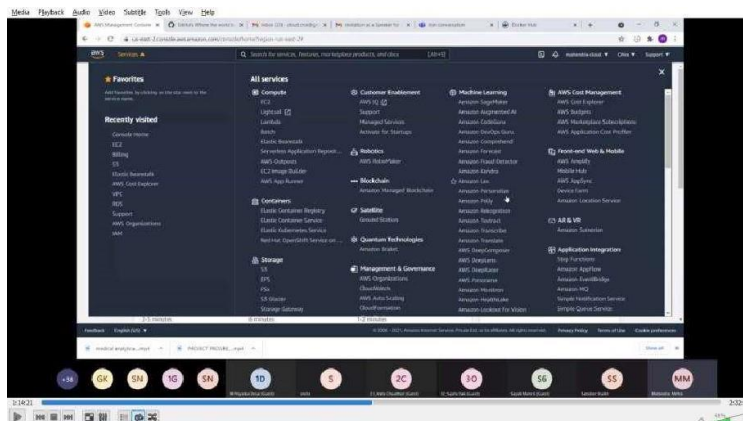
Theory :

Steps are as follows:

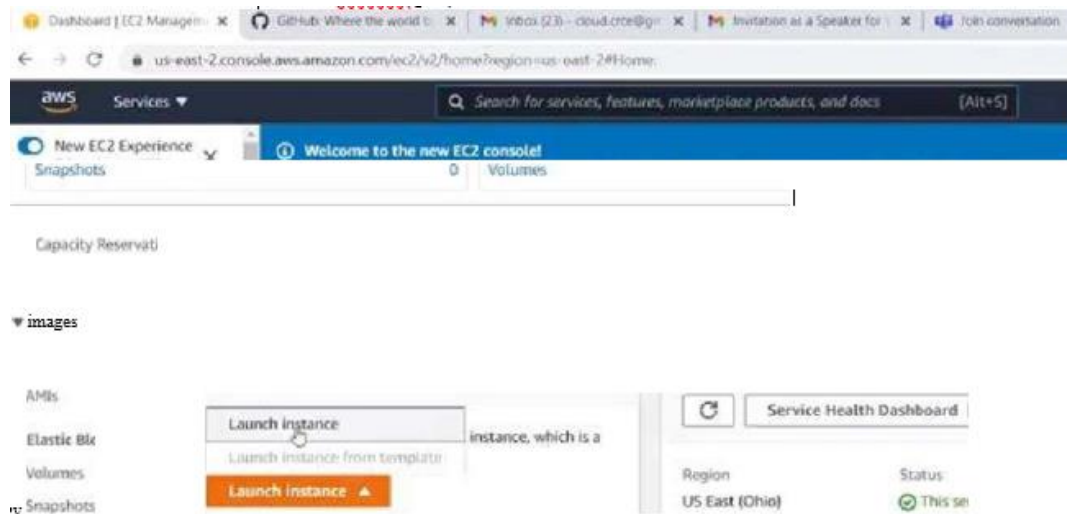
1. Create AWS account by paying Rs. 2/2
2. Login into AWS Console as root user.
3. Following screen will display



4 . Click on Services:



6. Click on "Launch Instances". As Current Instances running shows as 0".



7. Select "Ubuntu" and press "Select" button:



10. Click " Add Tags"
11. Skip this page as it is and click on "Configure Security Group".

1 Choose AMI 2 Choose Instance Type 3 Configure Instance 4 Add Storage 5 Add Tags 6 Configure Security Group 7 Review

Step 6: Configure Security Group

A security group is a software firewall that controls inbound and outbound traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group
☐ Select an existing security group

Security group name:
Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

Add Rule

Warning

Rules source 0.0.0.0/0 allows any IP addresses to access your instance, which is not recommended. It is recommended to restrict access to specific IP addresses only.

Cancel Previous Review and Launch

12. Write Security Group name as "vcet-te"
13. Click on "Add Rule" and select HTTP and Keep Source as "Anywhere" and One more add rule as "HTTPS" and keep source "Anywhere" as well.
14. Click on "Review and Launch"

aws services Ohio

Step 7: Review Instance Launch

You can't go back to edit your instance's configuration. You can go back to edit your instance's configuration.

each CGick Launch a Eky pair

AMI @ UbuntuServer 20.04

Instance Type

Security Groups

Key

Key pair

Edit instance type

Edit security groups

Security name

15. Select "Create key pair" and write key-pair name "vcet-te" and download this key pair and save it in respective folder.
16. Click on "Launch Instances"



Launch Status

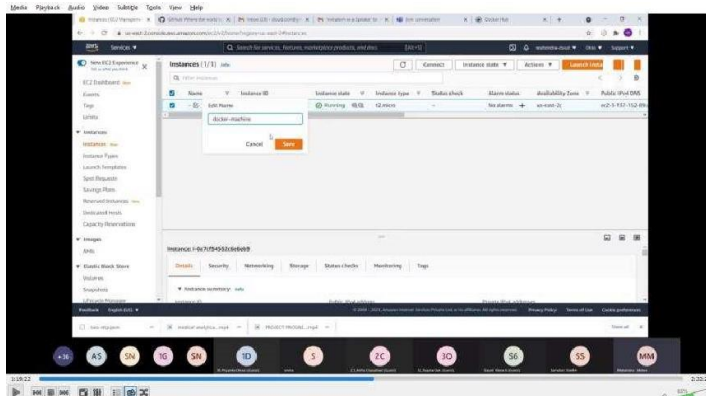


0 Get notified of estimated charges

[Create billing alerts](#) to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

Create billing alerts to

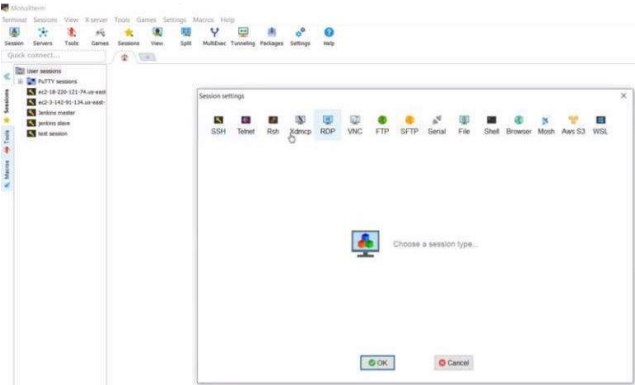
17. Write instance name as "docker-machine".



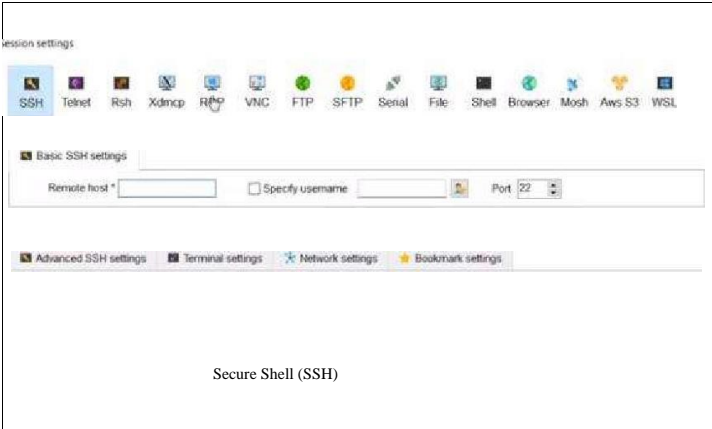
18. Your instance will be displayed on the instance monitoring window



- 19. Download "MobaXterm" for using SSH
- 20. Click on "Session" after opening the application



- 21. Click on "SSH"



Instance ID instance State v type v Status check Alarm v pub machine i07ef5*SS2c6e6eb9 @ Running QA t2.micro No alarms +

Filter instances

☒ Name

☒ docker-machine

Instance: i-Oa7cf54S52c6e6eb9 (docker-machine)

DetailsSecurityNetworking

StorageStatuschecksMonitoringTags

▼ Instance summary info

Instance ID

i-Oa7cf54S52c6e6eb9 (docker-machine)

Instance state

public

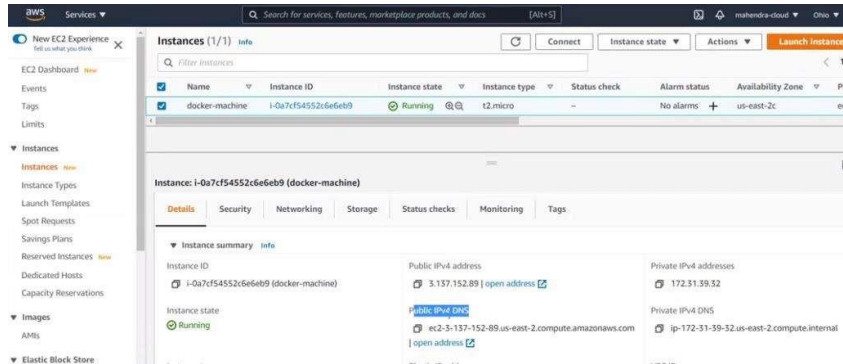
0 3137.1528910penaddressg

addresses

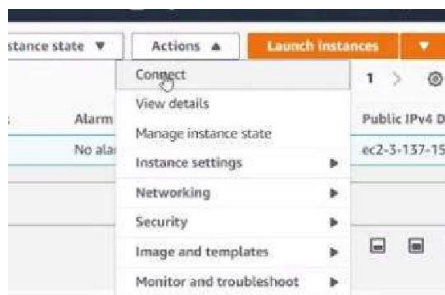
0 1723139.32

Private ONS

22. Pickup "Remote Host" from AWS Instance as "Copy IPV4 DNS address" and paste it here.



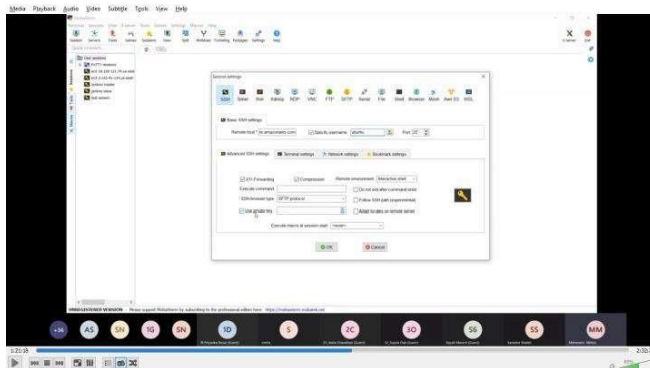
23. To get Username Click on "Action" Menu



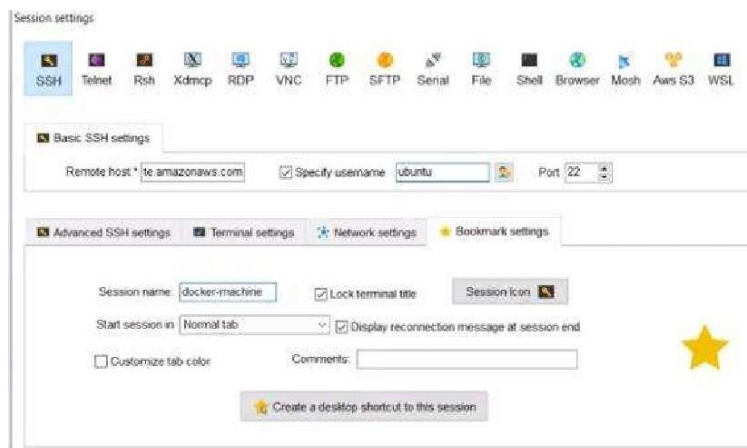
24. In "SSH Client" Tab, Copy the name as "ubuntu" and paste it in Username



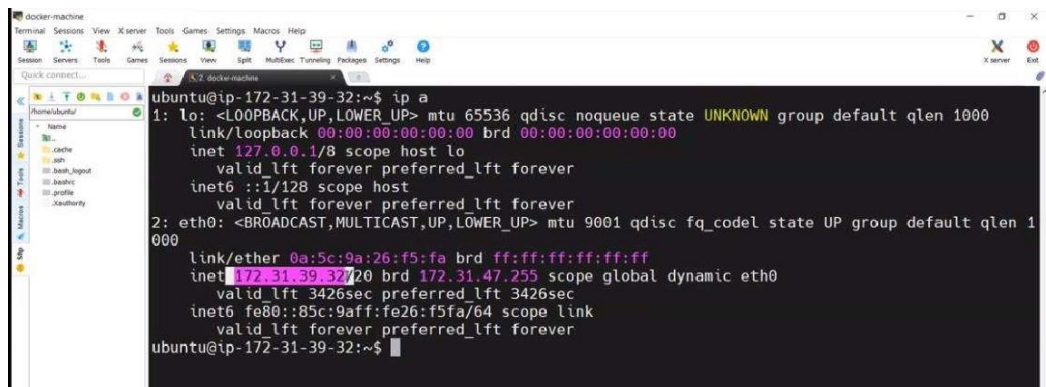
25. Click on "Advance SSH Settings". Select private key. Click on browse button and select that keypair downloaded file.



26. In bookmark settings write session name as "docker-machine"



27. It will open the following shell. Type "ip a" command to see IP address.

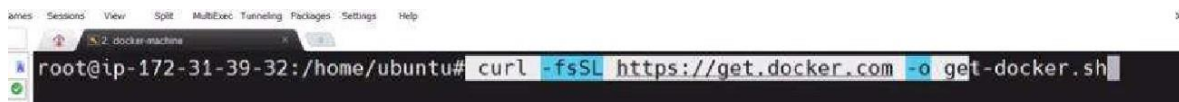


28. Type command "sudo su" to select superuser.



```
ubuntu@ip-172-31-22-76:~$ sudo su
```

29. Type command as follows to get docker shell:



```
root@ip-172-31-39-32:/home/ubuntu# curl -fsSL https://get.docker.com -o get-docker.sh
```

30. Then lastly run command "#sh get-docker.sh" to install docker in EC2.

31. Type command "#docker --version"

```
root@ip-172-31-39-32:/home/ubuntu# docker --version
Docker version 20.10.7, build f0df350 root@ip-172-31-39-32:/home/ubuntu#
```

Most Used Docker Commands

docker --version	docker ps	docker commit
docker --help	docker images	docker import
docker pull	docker stop	docker export
docker run	docker kill	docker container
docker build	docker rm	docker compose
docker login	docker rmi	docker swarm
docker push	docker exec	docker service

```
root@ip-172-31-39-32:/home/ubuntu# docker --version
Docker version 20.10.7, build f0df350
```

32. Go to https://hub.docker.com/_/mysql search for mysql image. Give following command:

#docker pull mysql

```
root@ip-172-31-39-32:/home/ubuntu# docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
b4d181a07f80: Pull complete
a462b60610f5: Pull complete
578fafb77ab8: Pull complete
524046006037: Pull complete
d0cbe54c8855: Pull complete
aa18e05cc46d: Pull complete
32ca814c833f: Pull complete
9ecc8abdb7f5: Pull complete
ad042b682e0f: Pull complete
71d327c6bb78: Pull complete
165d1d10a3fa: Pull complete
2f40c47d0626: Pull complete
Digest: sha256:52b8406e4c32b8cf0557f1b74517e14c5393aff5cf0384eff62d9e81f4985d4b
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
root@ip-172-31-39-32:/home/ubuntu# docker images
REPOSITORY    TAG        IMAGE ID      CREATED      SIZE
mysql         latest     5c62e459e087  12 days ago  556MB
```

33. Then type "#docker pull httpd"

```
root@ip-172-31-39-32:/home/ubuntu# docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
b4d181a07f80: Already exists
4b72f5187e6e: Pull complete
12b2c44d04b2: Pull complete
ef481fc2a03a: Pull complete
d483d945fcab: Pull complete
Digest: sha256:317cc1a2ded5e96225e4181323737f6d29b4fda58a6cc840a5752af6493a231f
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
root@ip-172-31-39-32:/home/ubuntu#
```

34. To make image containerize, run the docker image as follows:

#docker run -it 302 bash ('it' means interactive)

```
root@ip-172-31-39-32:/home/ubuntu# docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mysql          latest    5c62e459e087   12 days ago    556MB
httpd          latest    30287d899656   12 days ago    138MB
root@ip-172-31-39-32:/home/ubuntu# docker run -it 302 bash
root@e3d762f8f24a:/usr/local/apache2#
```

35. To display the container, give command as "#docker ps"

```
ubuntu@ip-172-31-39-32:~$ sudo su
root@ip-172-31-39-32:/home/ubuntu# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
e3d762f8f24a   302      "bash"    49 seconds ago    Up 48 seconds    80/tcp     optimistic_goldstine
root@ip-172-31-39-32:/home/ubuntu#
```

36. To check which linux is running give command as "#uname -a"

```
IS 2. docker-machine          3 dBCKermachine
root@09d92a8b0775 : /usr/local/apache2# uname -a
Linux ogdg2a8bG775 5.4.0-1G45-ews #47-Ubuntu SMP Tue Apr 13      UTC 2021 x86_64 GNU/Linux
root@09d92a8bG775 : 'usr/local/apache2#
```

This shows that you are inside the container apache2

37. To stop the container give command here "exit" and check in another tab that the container has stopped.

```
root@09d92a8b0775:/usr/local/apache2# exit
exit
root@ip-172-31-22-76:/home/ubuntu#
```

38. To check all stopped containers, give command as "#docker ps -a"

```
root@ip-172-31-39-32:/home/ubuntu# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
root@ip-172-31-39-32:/home/ubuntu# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
e3d762f8f24a   302      "bash"    4 minutes ago   Exited (1) 41 seconds ago   tic_goldstine
root@ip-172-31-39-32:/home/ubuntu#
```

39. Just press `^p^q` (To come out of container)

```
root@04e0349283b7:/usr/local/apache2# root@ip-172-31-22-76:/home/ubuntu#
root@ip-172-31-22-76:/home/ubuntu#
```

An Overview of above steps :

- 40. Give command `#docker exec -it 04e bash` (To go inside container)
- 41. To stop container give command `#docker stop 04e`
- 42. Check this with command `#docker ps`
- 43. To check previous exited containers, give command `#docker ps -a`

Conclusion : Therefore we understood the Docker Architecture and Container Life Cycle, installed Docker and successfully execute docker commands to manage images and interact with containers.