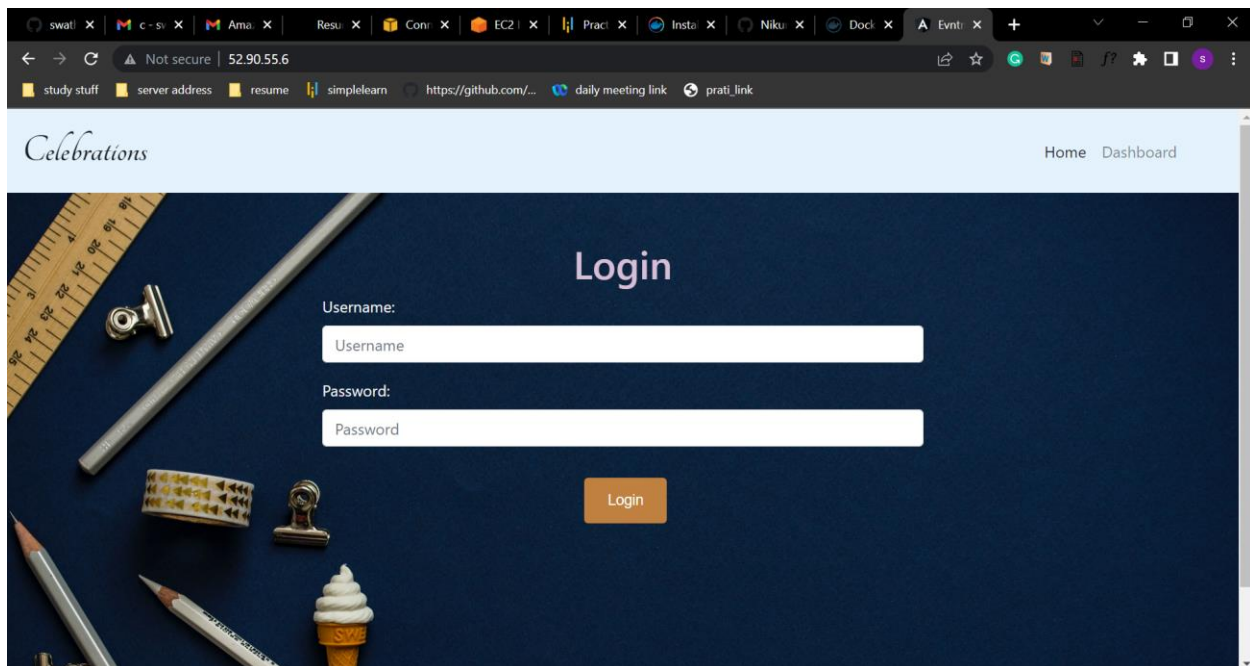


```
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-050ebe02a...
study stuff server address resume simplelearn https://github.com/... daily meeting link prati_link
aws Services Search for services, features, blogs, docs, and more [Alt+S] N. Virginia Corestack_Role/swathis6_mphasis @ 7667-8964-5146
4ad80e1f9fc13978793c): Bind for 0.0.0.0:80 failed: port is already allocated.
[ec2-user@ip-172-31-28-193 evntmgt]$ history
1 docker -v
2 git -v
3 git clone https://github.com/swathiz/angularimage.git
4 ls
5 cd angularimage/
6 cd evntmgt/
7 docker build -t eventmgtimage
8 docker build -t eventmgtimage .
9 docker container run -d --name mywebl -p 80:80 mywebsiteimage
10 history
11 docker container run -d --name mywebl -p 80:80 eventmgtimage
12 docker login
13 docker tag eventmgtimage sonal1997/e1
14 docker push sonal1997/e1
15 docker container run -d --name sswe -p 80:80 sonal1997/e1
16 history
[ec2-user@ip-172-31-28-193 evntmgt]$
```

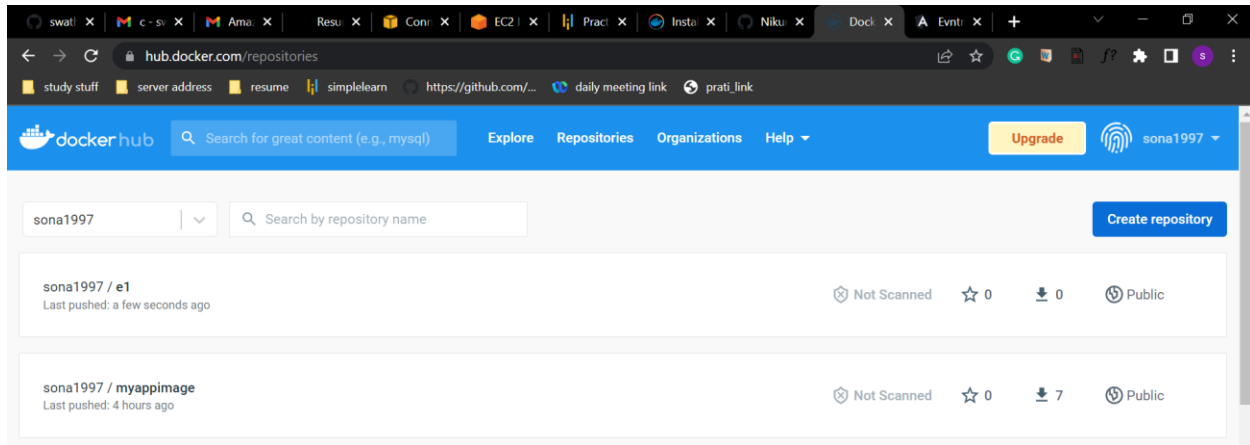
i-050ebe02a89264bd1 (dockerimg)

PublicIPs: 52.90.55.6 PrivateIPs: 172.31.28.193

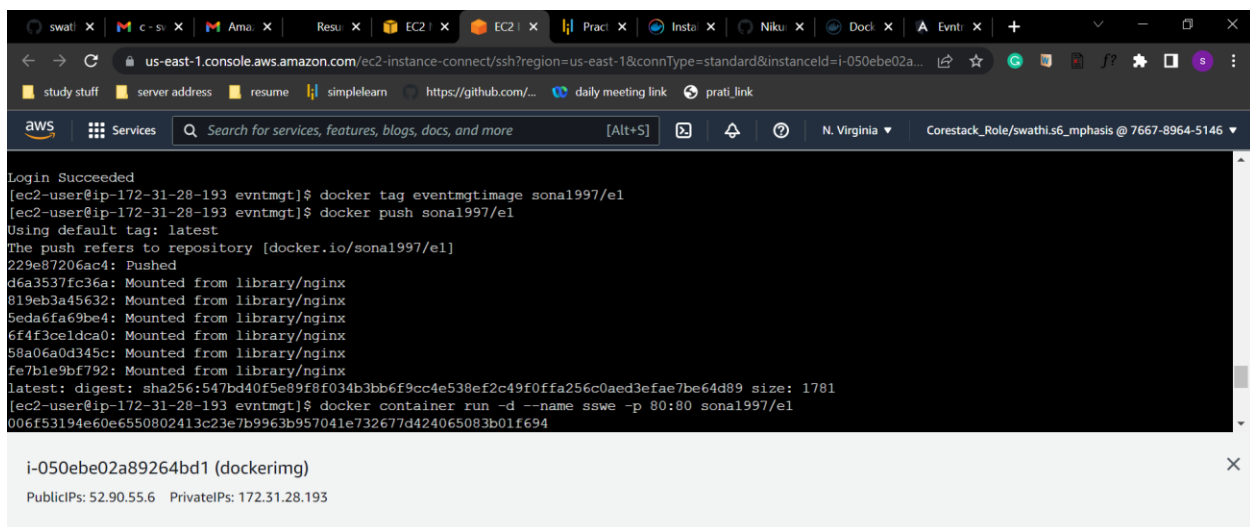
**History of commands used in setting up the docker image.**



**After running the image in the container this is the browser output.**



### Images we have added in the dockerhub.



### Output of logging into docker image running

```
swati X C - sv X Ama: X Resu X EC2 X EC2 X Pract X Insta X Niku X Dock X A Evnti X +
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-050ebe02a...
study stuff server address resume simplelearn https://github.com/... daily meeting link prati_link
aws Services Search for services, features, blogs, docs, and more [Alt+S] N. Virginia Corestack_Role/swathi.s6_mphasis @ 7667-8964-5146
b38afb8ac3513bbe0ae2523696919fdff9ec3729f2ea63a5c0d2bd8c0a5a2215
[ec2-user@ip-172-31-28-193 evntmgt]$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: sonal1997
Password:
WARNING! Your password will be stored unencrypted in /home/ec2-user/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[ec2-user@ip-172-31-28-193 evntmgt]$ docker tag eventmgtimage sonal1997/e1
[ec2-user@ip-172-31-28-193 evntmgt]$ docker push sonal1997/e1
Using default tag: latest
The push refers to repository [docker.io/sonal1997/e1]
229e87206ac4: Pushed

i-050ebe02a89264bd1 (dockerimg)
PublicIPs: 52.90.55.6 PrivateIPs: 172.31.28.193
```

## Output of docker hub login

```
swati X C - sv X Ama: X Resu X EC2 X EC2 X Pract X Insta X Niku X Dock X A Evnti X +
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-050ebe02a...
study stuff server address resume simplelearn https://github.com/... daily meeting link prati_link
aws Services Search for services, features, blogs, docs, and more [Alt+S] N. Virginia Corestack_Role/swathi.s6_mphasis @ 7667-8964-5146
latest: Pulling from library/nginx
bd159e379b3b: Pull complete
8d634ce99fb9: Pull complete
98b0bbcc0ec6: Pull complete
6ab6a6301bde: Pull complete
f5d8edcd47b1: Pull complete
fe24ce36f968: Pull complete
Digest: sha256:2f770d2fe27bc85f68fd7fe6a63900ef7076bc703022fe81b980377fe3d27b70
Status: Downloaded newer image for nginx:latest
---> 51086ed63d8c
Step 9/9 : COPY --from=build /app/dist/evntmgt/ /usr/share/nginx/html/
---> 8754a07ff7f1
Successfully built 8754a07ff7f1
Successfully tagged eventmgtimage:latest
[ec2-user@ip-172-31-28-193 evntmgt]$ docker container run -d --name myweb1 -p 80:80 mywebsiteimage
Unable to find image 'mywebsiteimage:latest' locally

i-050ebe02a89264bd1 (dockerimg)
PublicIPs: 52.90.55.6 PrivateIPs: 172.31.28.193
```

## Docker container running in

```
ec2-user@ip-172-31-17-167:~
Using username "ec2-user".
Authenticating with public key "swarkey"

  ____|_  ____|_  )
  _|_  (  _|_  /   Amazon Linux 2 AMI
  ____|_  ____|_  |

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-17-167 ~]$
[ec2-user@ip-172-31-17-167 ~]$
[ec2-user@ip-172-31-17-167 ~]$
[ec2-user@ip-172-31-17-167 ~]$ docker swarm init --advertise-addr 172.31.17.167
Swarm initialized: current node (ymx6km8ex1z07yuehac517ue9) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-198arnckpgc0k6uymooyq5bxtty84ewqkz6t2ur4nx41f26jo82-e9dw9znp2e08xv0ilje0o2srm 172.31.17.167:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

[ec2-user@ip-172-31-17-167 ~]$ ^C
[ec2-user@ip-172-31-17-167 ~]$
```

**Running the docker swarm command and provide the private IP of the master instance.**

```
ec2-user@ip-172-31-24-18:~
Using username "ec2-user".
Authenticating with public key "swarkey"

  ____|_  ____|_  )
  _|_  (  _|_  /   Amazon Linux 2 AMI
  ____|_  ____|_  |

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-24-18 ~]$ docker swarm join --token SWMTKN-1-198arnckpgc0k6uymooyq5bxtty84ewqkz6t2ur4nx41f26jo82-e9dw9znp2e08xv0ilje0o2srm 172.31.17.167:2377
This node joined a swarm as a worker.
[ec2-user@ip-172-31-24-18 ~]$
```

**Adding the docker swarm join command to the worker1 instance to join the instance to master instance.**

```
ec2-user@ip-172-31-23-101:~
Using username "ec2-user".
Authenticating with public key "swarkey"

  _ | _ | _ )
  _ | ( _ | /   Amazon Linux 2 AMI
  __| \__| __|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-23-101 ~]$ docker swarm join --token SWMTKN-1-198arnckpgc0k6uymooyq5bxtty84ewqkz6t2ur4nx41f26jo82-e9dw9znp2e08xv0ilje0o2srm 172.31.17.167:2377
This node joined a swarm as a worker.
[ec2-user@ip-172-31-23-101 ~]$
```

**Adding the docker swarm join command to the worker2 instance to join the instance to master instance.**

```
[ec2-user@ip-172-31-17-167 ~]$ docker node ls
ID                                HOSTNAME                                STATUS    AVAILABI
LITY   MANAGER STATUS  ENGINE VERSION
ymx6km8ex1z07yuehac517ue9 *   ip-172-31-17-167.ec2.internal    Ready     Active
      Leader                20.10.17
3jfxa4eloicsh2jrae9fyjh4u      ip-172-31-23-101.ec2.internal    Ready     Active
      20.10.17
rq3hvdbangnhodgxz3illesb4      ip-172-31-24-18.ec2.internal     Ready     Active
      20.10.17
[ec2-user@ip-172-31-17-167 ~]$
```

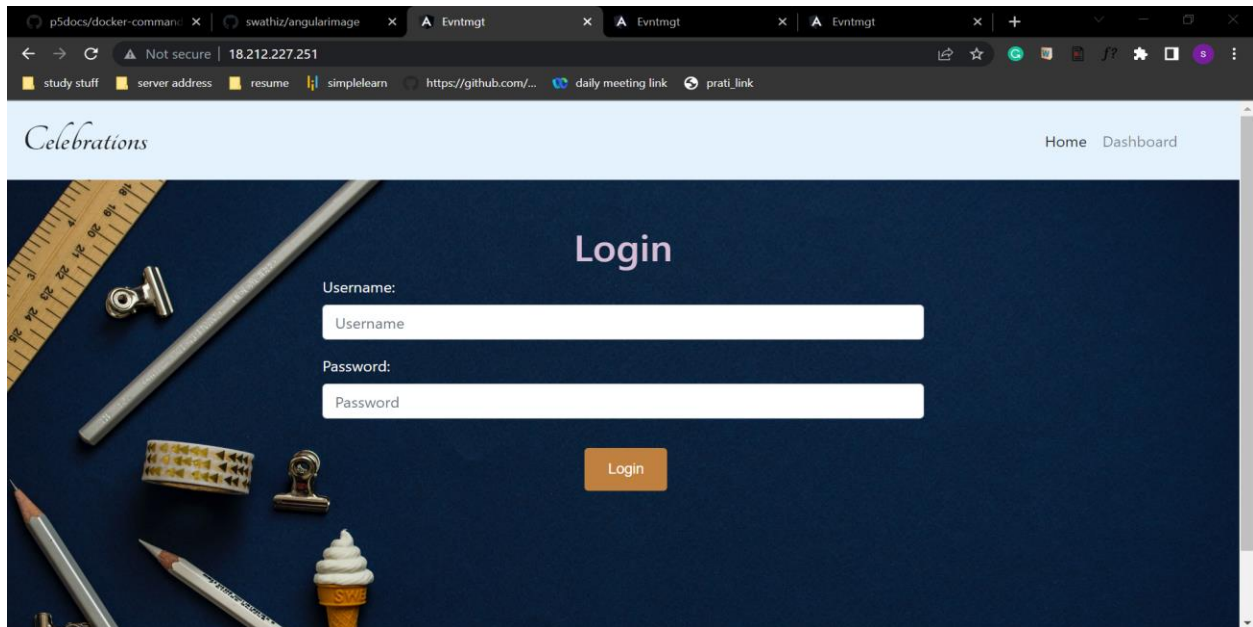
**In the master instance to check all the nodes joined to the master instance.**

```
[ec2-user@ip-172-31-17-167 ~]$ docker service create --name myappl --replicas 5 -p 80:80 sonal997/e1dovkp3xte4sxdhq4c8w9frels
overall progress: 5 out of 5 tasks
1/5: running [=====>]
2/5: running [=====>]
3/5: running [=====>]
4/5: running [=====>]
5/5: running [=====>]
verify: Service converged
[ec2-user@ip-172-31-17-167 ~]$
```

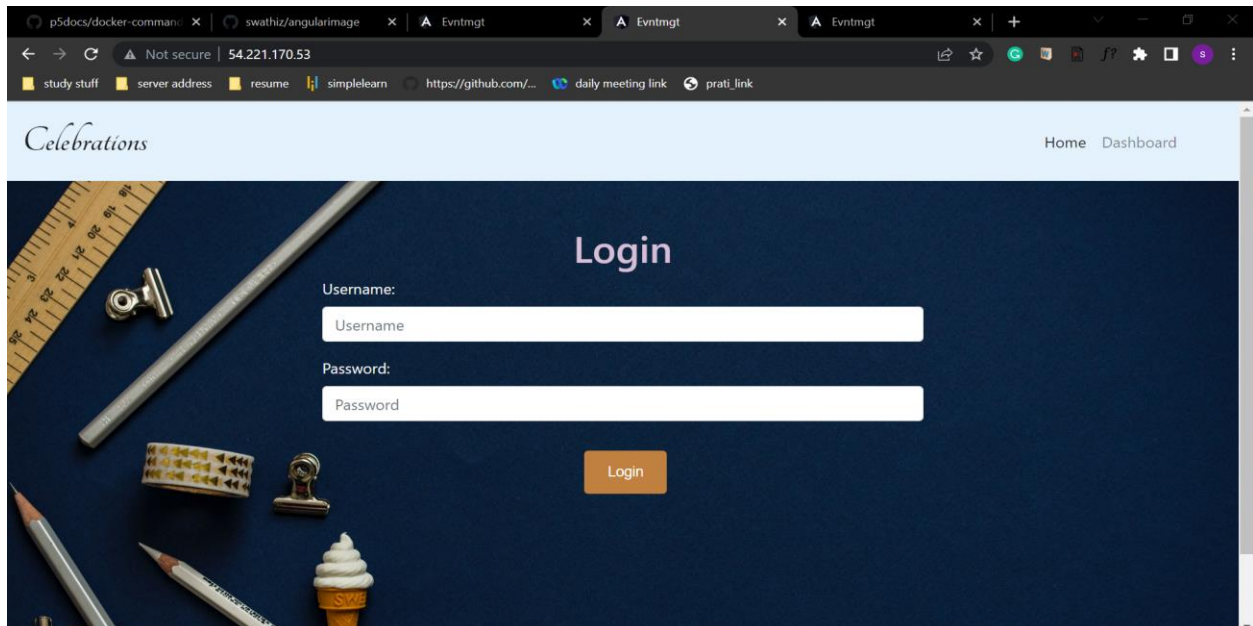
**Create a service of the e1 image of 5 replicas.**

```
[ec2-user@ip-172-31-17-167 ~]$ docker service ls
ID                NAME      MODE     REPLICAS  IMAGE              PORTS
dovkp3xte4sx     myappl    replicated 5/5        sona1997/el:latest *:80->80/tcp
[ec2-user@ip-172-31-17-167 ~]$ docker service ps myappl
ID                NAME      IMAGE              NODE                                DESIRED STATE  CURRENT STATE      ERROR      PORTS
re4yrus4duly     myappl.1  sona1997/el:latest ip-172-31-23-101.ec2.internal      Running        Running about a minute ago
9619x47uk0li     myappl.2  sona1997/el:latest ip-172-31-17-167.ec2.internal      Running        Running about a minute ago
s0pcb168lpen     myappl.3  sona1997/el:latest ip-172-31-23-101.ec2.internal      Running        Running about a minute ago
bcl76mnnitw5     myappl.4  sona1997/el:latest ip-172-31-17-167.ec2.internal      Running        Running about a minute ago
z7blpg2peblu     myappl.5  sona1997/el:latest ip-172-31-24-18.ec2.internal      Running        Running about a minute ago
[ec2-user@ip-172-31-17-167 ~]$
```

**To see the services running in the master instance .**

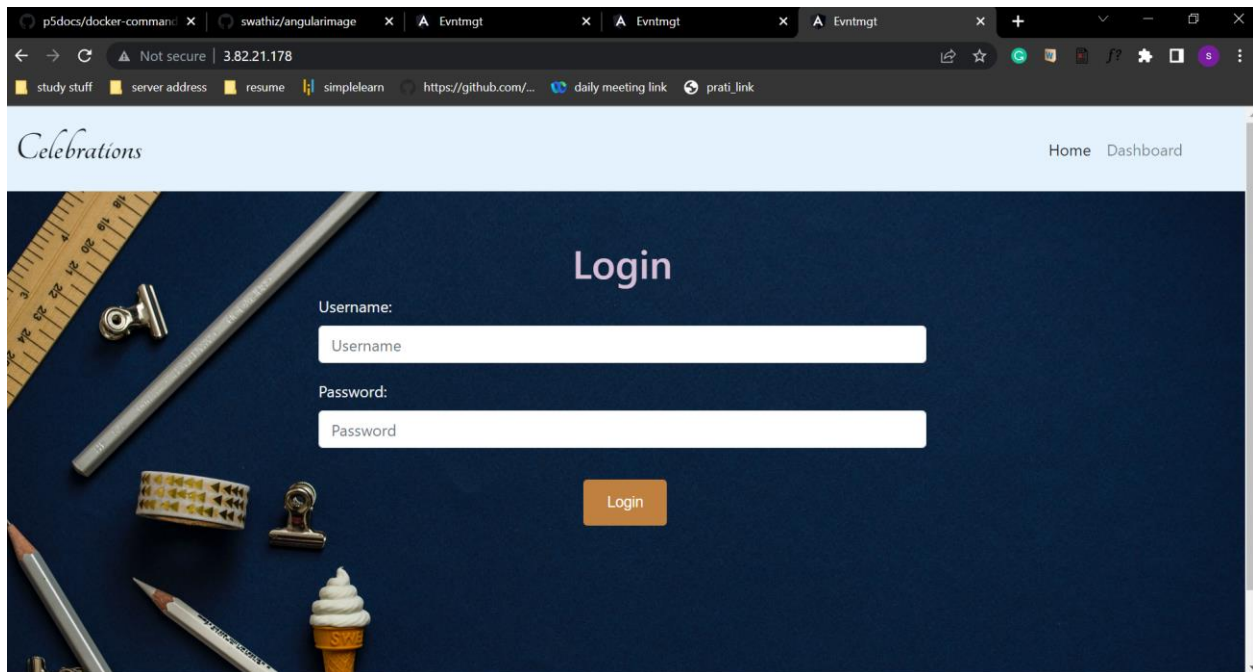


**Image Running in the master instance.**

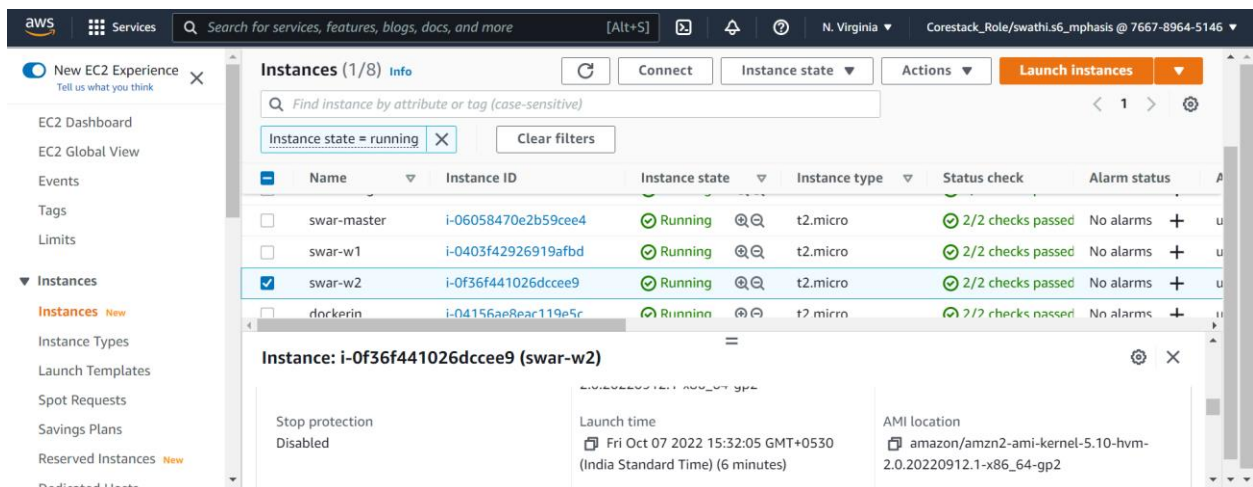


**Image running in the worker1 instnace**





## Image running in the worker2 instance



## Docker swarm worker 2 launch time

Instance: i-0f36f441026dccee9 (swar-w2)

Instance ID i-0f36f441026dccee9 (swar-w2)	Public IPv4 address 3.82.21.178   <a href="#">open address</a>	Private IPv4 addresses 172.31.24.18
--	---	--

## Docker swarm worker 2 instance details

### Instance: i-0403f42926919afbd (swar-w1)

▼ Instance summary <a href="#">Info</a>		
Instance ID i-0403f42926919afbd (swar-w1)	Public IPv4 address 54.221.170.53   <a href="#">open address</a>	Private IPv4 addresses 172.31.123.101

### Docker swarm worker 1 instance details

### Instance: i-0403f42926919afbd (swar-w1)

Linux/UNIX	amzn2-ami-kernel-5.10-hvm-2.0.20220912.1-x86_64-gp2	Disabled
Stop protection Disabled	Launch time Fri Oct 07 2022 15:32:05 GMT+0530 (India Standard Time) (5 minutes)	AMI location amazon/amzn2-ami-kernel-5.10-hvm-2.0.20220912.1-x86_64-gp2

### Docker swarm worker 1 instance launch time

### Instance: i-06058470e2b59cee4 (swar-master)

▼ Instance summary <a href="#">Info</a>		
Instance ID i-06058470e2b59cee4 (swar-master)	Public IPv4 address 18.212.227.251   <a href="#">open address</a>	Private IPv4 addresses 172.31.17.167

### Docker swarm master instance details

### Instance: i-06058470e2b59cee4 (swar-master)

Stop protection Disabled	Launch time Fri Oct 07 2022 15:32:05 GMT+0530 (India Standard Time) (7 minutes)	AMI location amazon/amzn2-ami-kernel-5.10-hvm-2.0.20220912.1-x86_64-gp2
-----------------------------	---	--

### Docker swarm master instance launch time

EC2 > Security Groups > sg-026d2debb57c1abba - launch-wizard-7 > Edit inbound rules

### Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>	
sg-09f20c16ec6da3e6f	All traffic	All	All	Custom	<input type="text" value="Q"/>	<input type="button" value="Delete"/>
sg-0ce89f770318575e8	HTTP	TCP	80	Custom	<input type="text" value="sg-026d2debb57c1abba"/>	<input type="button" value="Delete"/>
sg-01b0eb89070d89d10	SSH	TCP	22	Custom	<input type="text" value="0.0.0.0/0"/>	<input type="button" value="Delete"/>

Inbound rules added to all the instance security groups.



