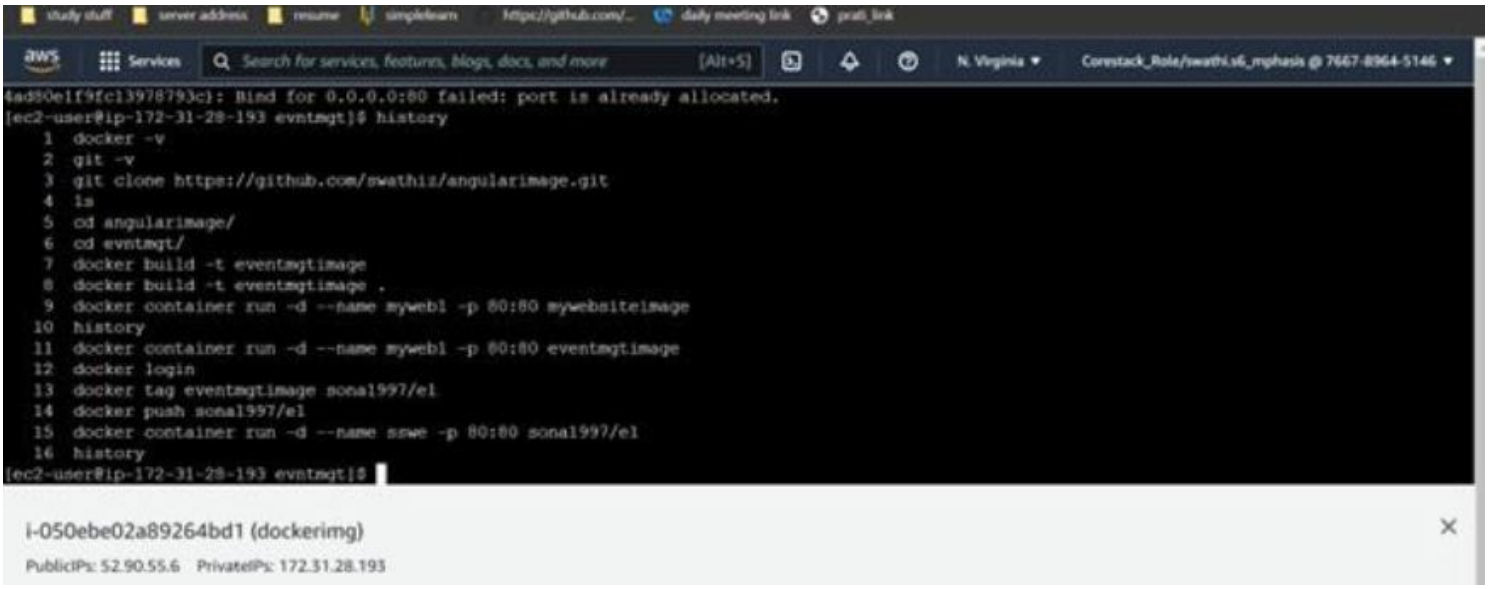
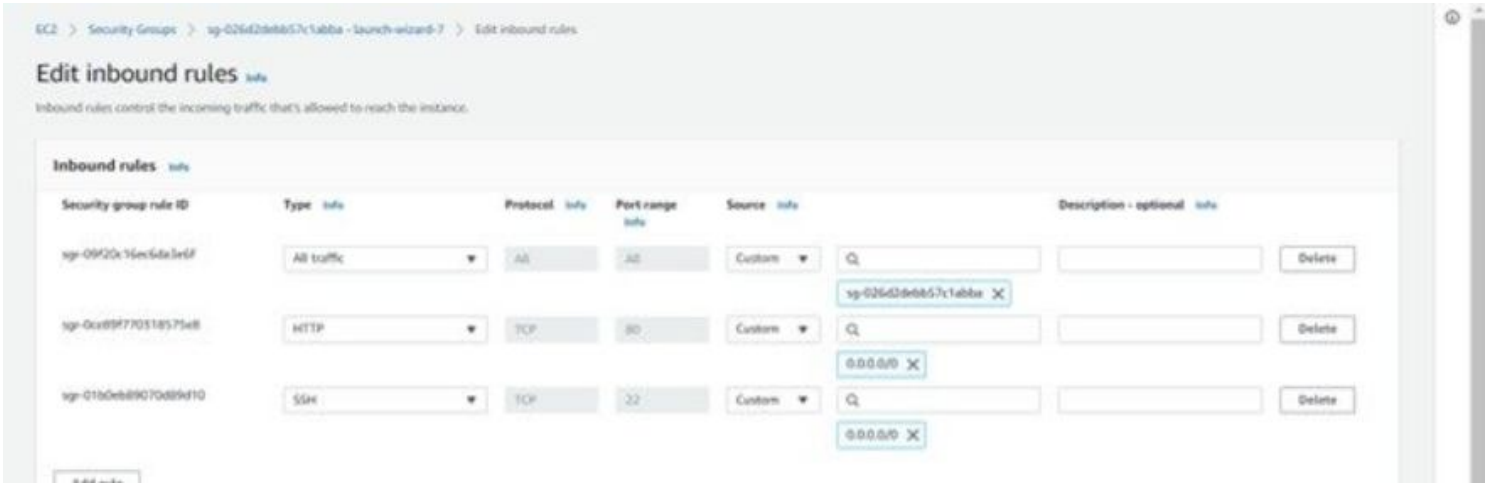
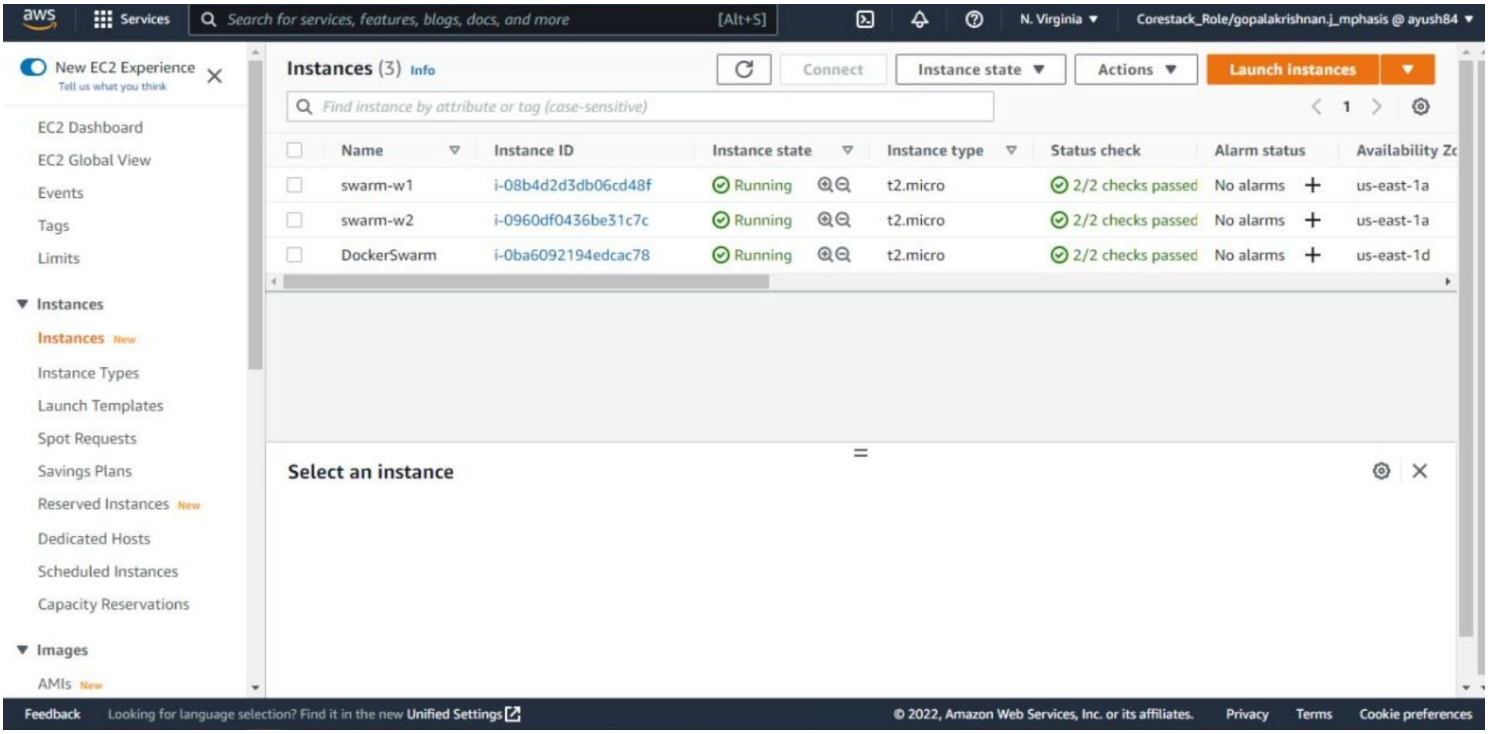
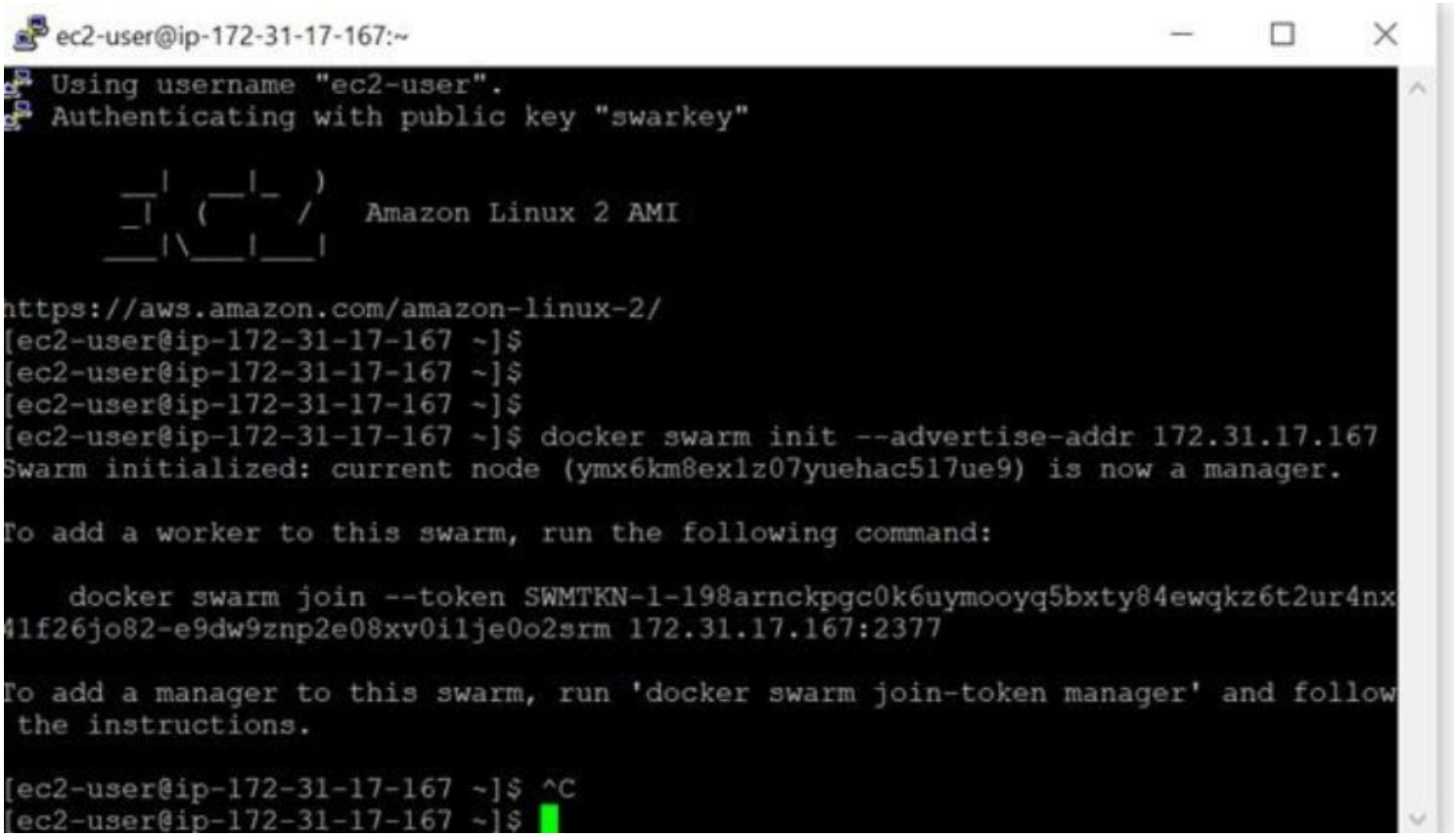

study stuff
server address
resume
simplelearn
https://github.com/...
daily meeting link
prati_link
aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
Corestack_Role/swathi.s6_mphasis @ 7667-8964-5146
4ad80e1f9fc13978793c): Bind for 0.0.0.0:80 failed: port is already 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 sonal997/el
14 docker push sonal997/el
15 docker container run -d --name sswe -p 80:80 sonal997/el
16 history
[ec2-user@ip-172-31-28-193 evntmgt]$
i-050ebe02a89264bd1 (dockerimg)
PublicIPs: 52.90.55.6 PrivateIPs: 172.31.28.193

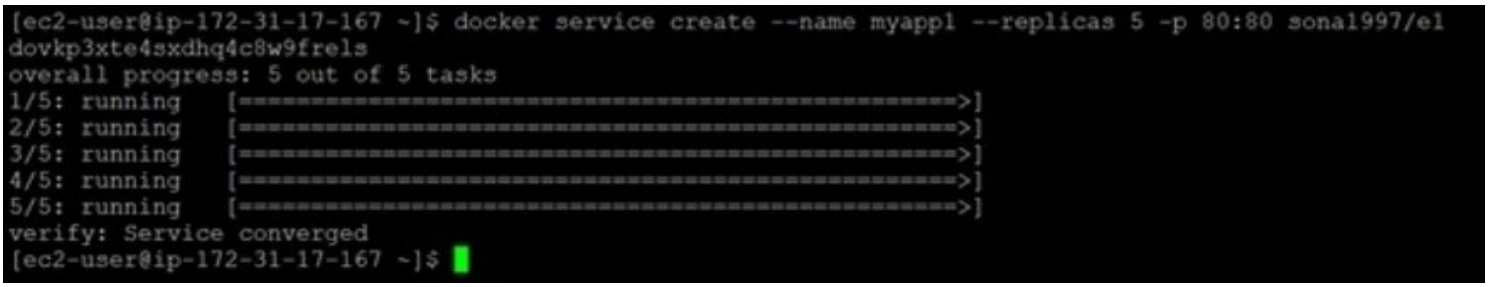
## Docker File

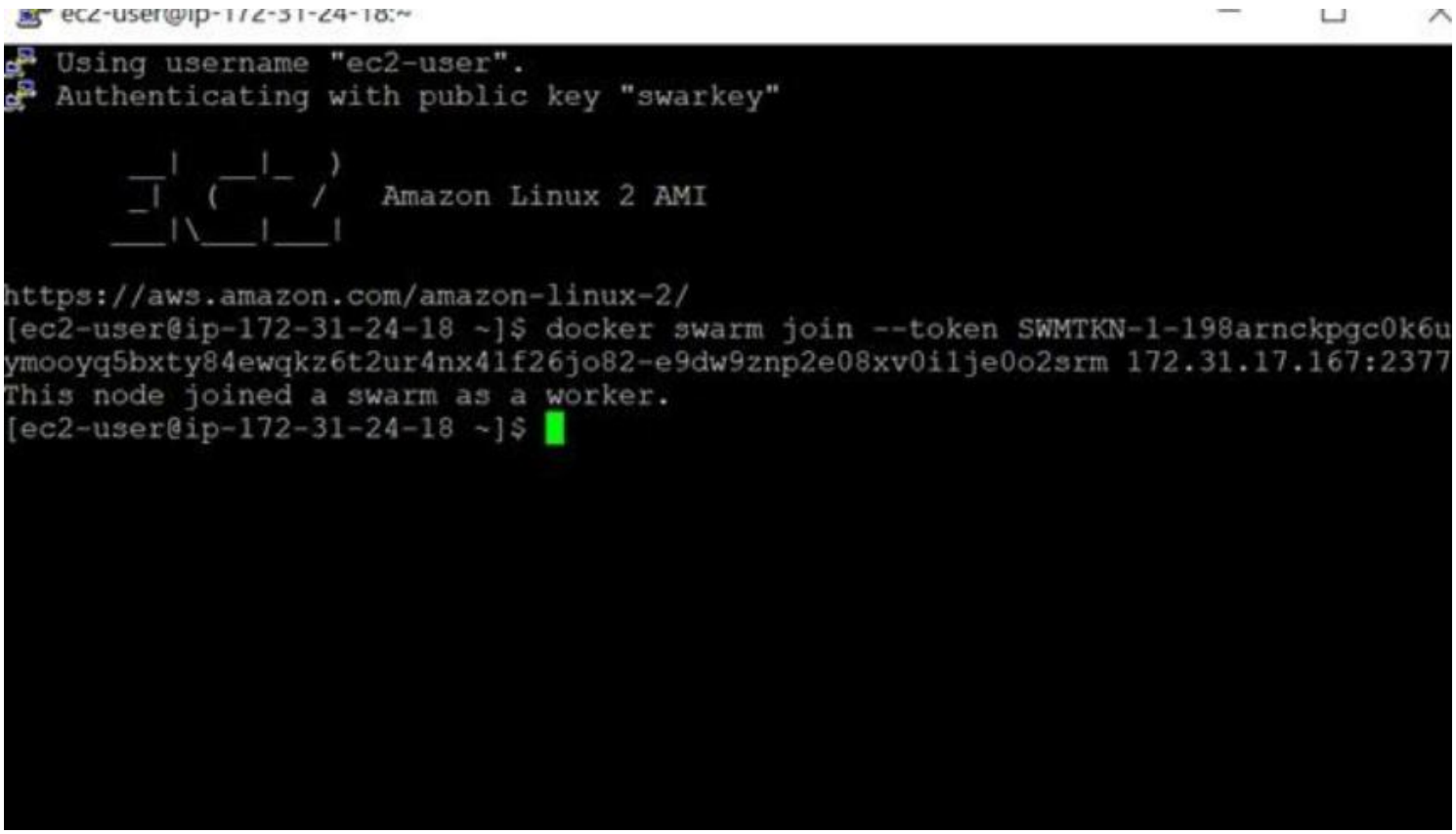
```
FROM
node:16.14.0-
alpine AS
build
                RUN npm i -g @angular/cli@13.1.0
                WORKDIR /app
                COPY package.json package-lock.json ./
                RUN npm install
                COPY . .
                RUN ng build --prod

                FROM nginx
                COPY --from=build /app/dist/evntmgt/ /usr/share/nginx/html/
```


EC2 > Security Groups > sg-026d2db6b57c1abba - launch-wizard-7 > Edit inbound rules
Edit inbound rules Info
Inbound rules control the incoming traffic that's allowed to reach the instance.
Inbound rules Info
Security group rule ID
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
sgr-09f20c16ec6da3e6f
All traffic
All
All
Custom
sg-026d2db6b57c1abba
Delete
sgr-0ce89f770518575e8
HTTP
TCP
80
Custom
0.0.0.0/0
Delete
sgr-01b0eb89070d89d10
SSH
TCP
22
Custom
0.0.0.0/0
Delete


aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia
Corestack_Role/gopalakrishnan.j_mphasis @ ayush84
New EC2 Experience
Tell us what you think
EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Instances New
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances New
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
AMIs New
Instances (3) Info
Connect
Instance state
Actions
Launch instances
Find instance by attribute or tag (case-sensitive)
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zo
swarm-w1
i-08b4d2d3db06cd48f
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
swarm-w2
i-0960df0436be31c7c
Running
t2.micro
2/2 checks passed
No alarms
us-east-1a
DockerSwarm
i-0ba6092194edcac78
Running
t2.micro
2/2 checks passed
No alarms
us-east-1d
Select an instance
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


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-1-198arnckpgc0k6uymooyq5bxty84ewqkz6t2ur4nx
41f26jo82-e9dw9znp2e08xv0i1je0o2srm 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 ~]$


[ec2-user@ip-172-31-17-167 ~]$ docker service create --name myapp1 --replicas 5 -p 80:80 sona1997/e1
dovkp3xte4sxdhq4c8w9frels
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 ~]$


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-1-198arnckpgc0k6u
ymooyq5bxty84ewqkz6t2ur4nx41f26jo82-e9dw9znp2e08xv0i1je0o2srm 172.31.17.167:2377
This node joined a swarm as a worker.
[ec2-user@ip-172-31-24-18 ~]$