D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

School of Computer Science Engineering and Application

BCA TY SEM VI

Subject Name: Container and Orchestration Practical

Assignment No. 8

Aim: Build Image with two dependencies (Flask, Redis) and create container with 5 replicas with docker stack.

Submitted By

Name: Jayesh Bhangale

PRN: 20210801024

Date: 16th April, 2024

Technology Used: Docker, AWS, Swarm, Docker Hub.

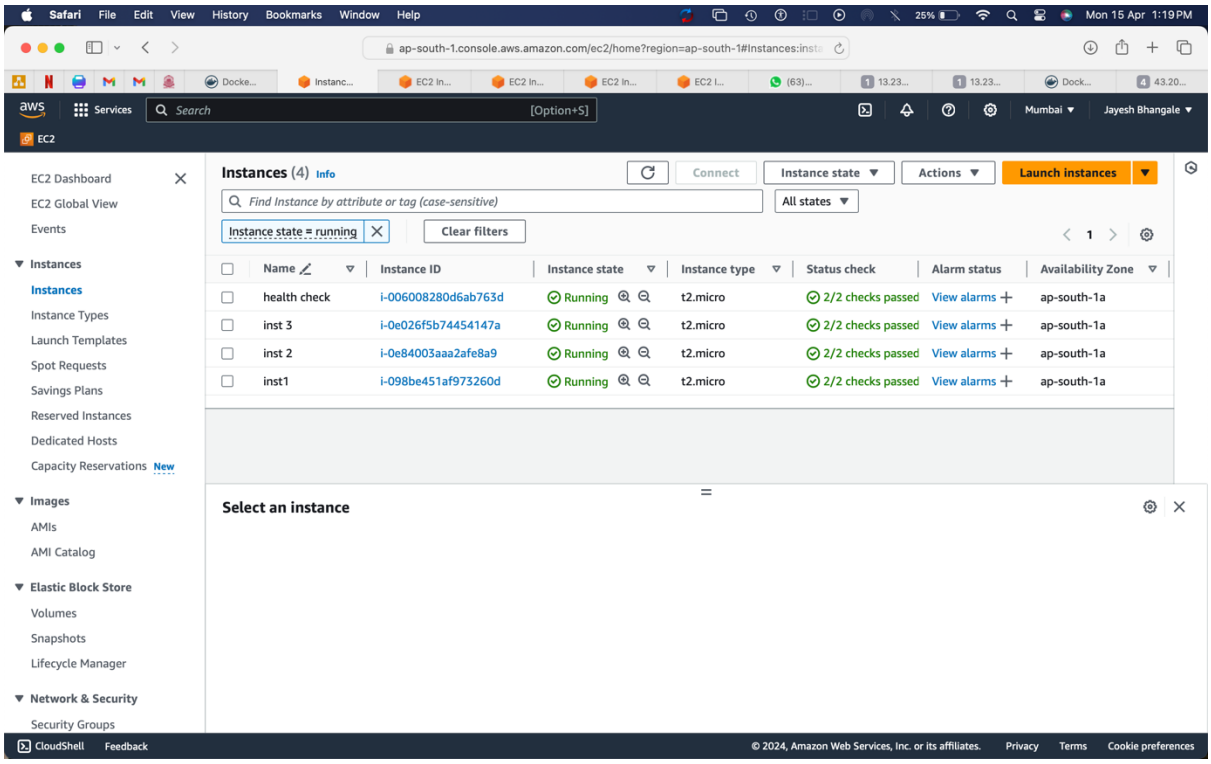
STEP -1: Create 3 instances, connect to it and install docker.

Sudo su

apt update -y

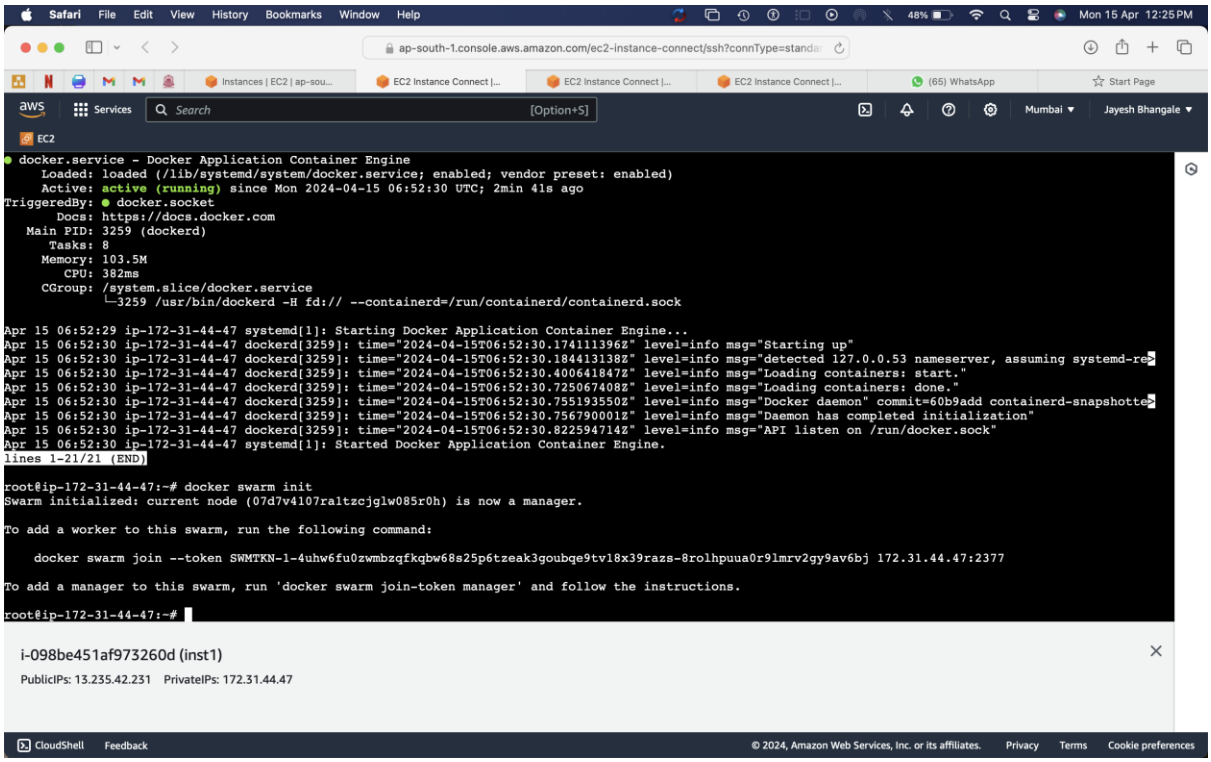
curl -fsSl https://get.docker.com -o get-docker.sh

sh get-docker.sh



STEP-2: Now Activate Docker swarm.

docker swarm init



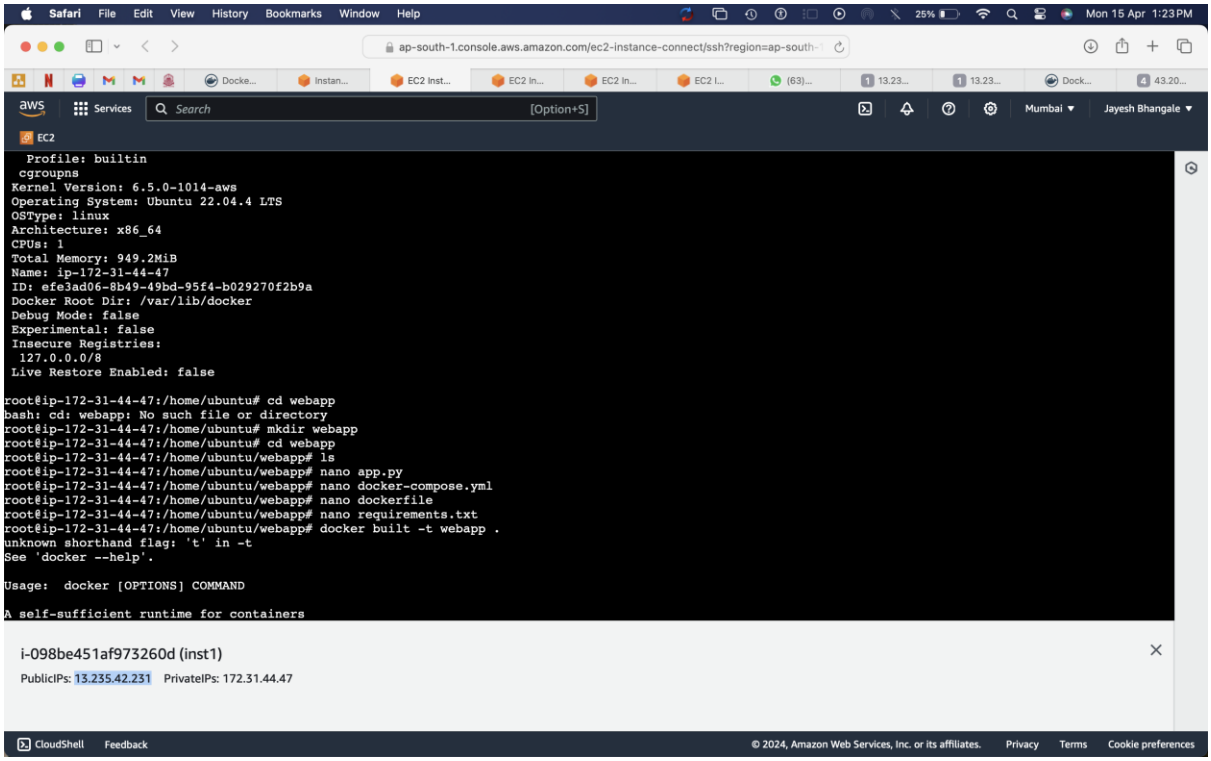
STEP-3: Now create a directory and upload the required files.

mkdir webapp

nano app.py

```
from flask import Flask
from redis import Redis, RedisError
import os
import socket
# Connect to Redis
redis = Redis(host="redis", db=0, socket_connect_timeout=2, socket_timeout=2)
app = Flask(__name__)
@app.route("/")
def hello():
    try:
        visits = redis.incr("counter")
    except RedisError:
        visits = "<i>cannot connect to Redis, counter disabled</i>"
    html = "<h3>Hello {name}!</h3>" \
           "<b>Hostname:</b> {hostname}<br/>" \
           "<b>Visits:</b> {visits}"
```

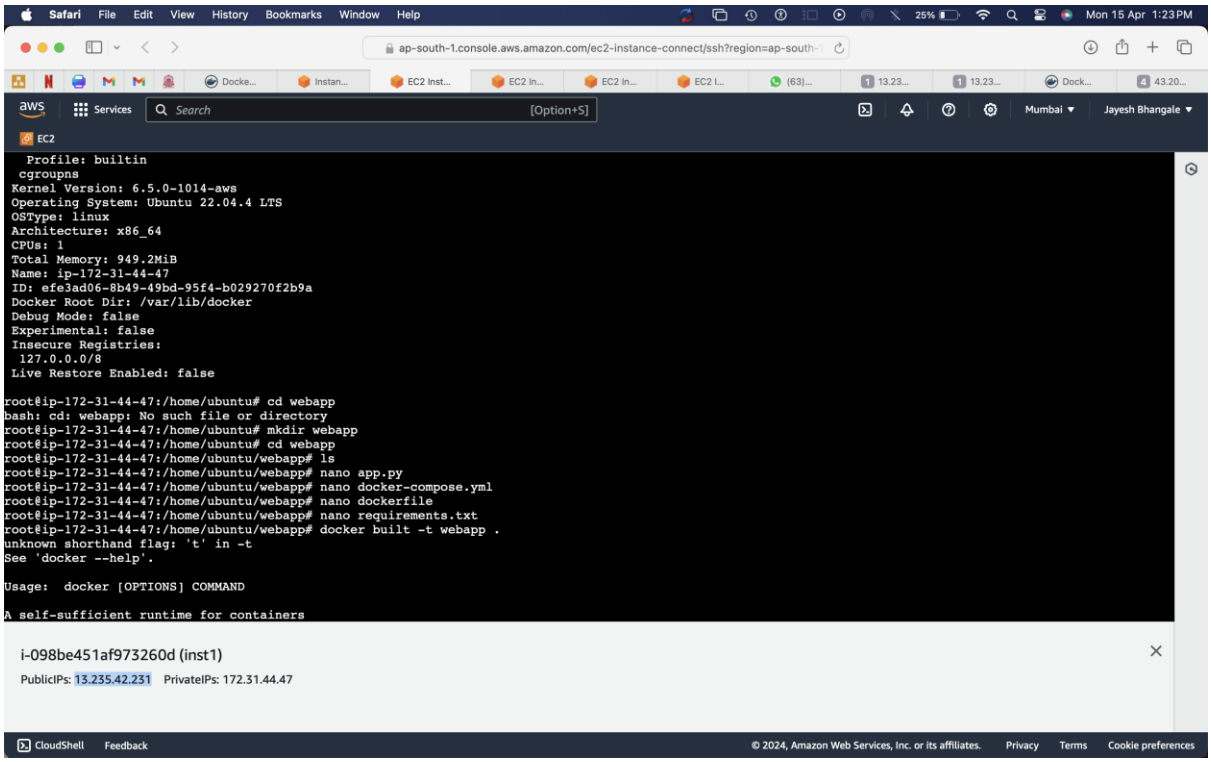
```
    return html.format(name=os.getenv("NAME", "world"),
hostname=socket.gethostname(), visits=visits)

if __name__ == "__main__":

    app.run(host='0.0.0.0', port=80)
```



nano docker-compose.yml

```
version: "3"

services:

  # Service Name Defined as web

  web:

    # Pull the Image from Repository.

    # replace username/repo:tag with your name and image details

    image: username/repo:tag

    # Command used to deploy the Service

    deploy:

      # Run 5 instances of that image as a service called web

      replicas: 5

      resources:

        # Limiting each one to use, at most, 10% of a single core of CPU time and 50MB of
RAM.
```

```
      limits:
        cpus: "0.1"
        memory: 50M
    # Immediately restart containers if one fails.
    restart_policy:
      condition: on-failure
  # Map port 4000 on the host to web's port 80.
  ports:
   - "4000:80"
  # Define default network
  networks:
   - webnet

redis:
  image: redis:latest
  ports:
   - "6379:6379"
  volumes:
   - "/app/redis_data:/data"
  deploy:
    placement:
      constraints: [node.hostname == manager2]
  command: redis-server --appendonly yes
  networks:
   - webnet

networks:
 webnet:
```



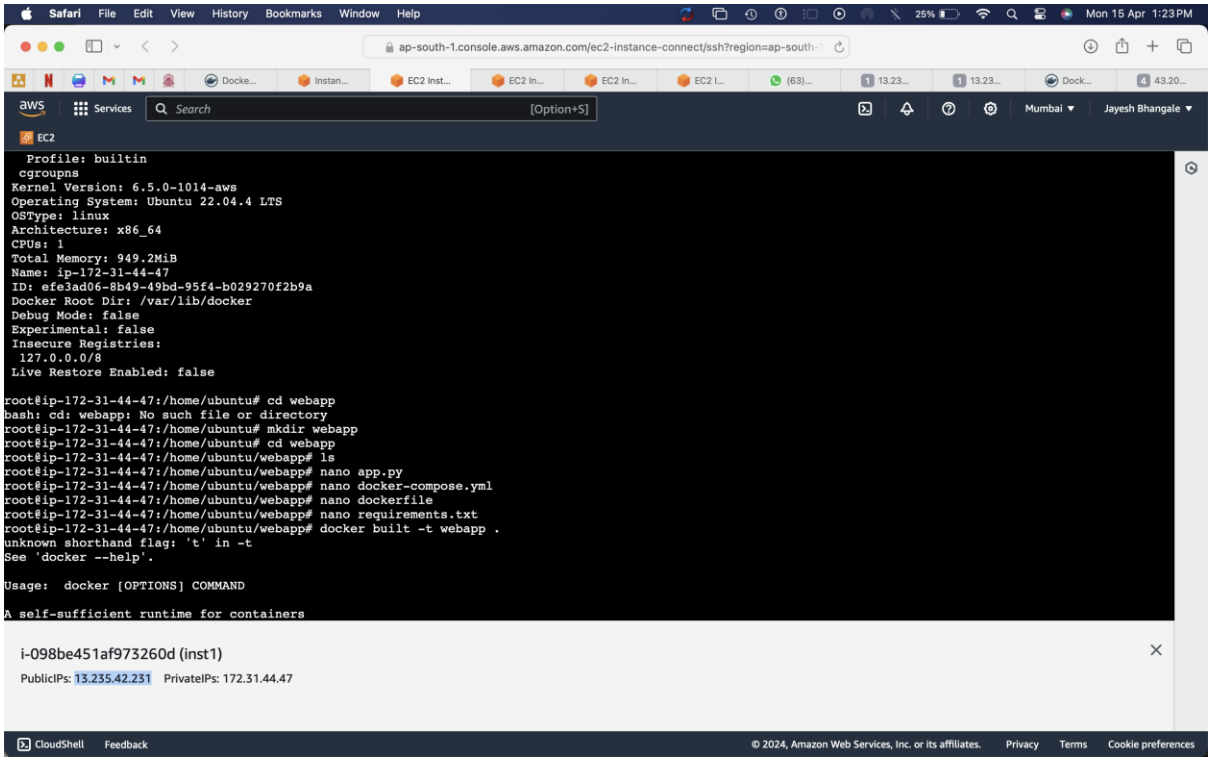
nano dockerfile

```
# Use an official Python runtime as a parent image
FROM python:3.12-slim
# Set the working directory to /app
WORKDIR /app
# Copy the current directory contents into the container at /app
COPY . /app
# Install any needed packages specified in requirements.txt
RUN pip install --trusted-host pypi.python.org -r requirements.txt
# Make port 80 available to the world outside this container
EXPOSE 80
# Define environment variable
ENV NAME World
# Run app.py when the container launches
CMD ["python", "app.py"]
```

nano requirements.txt

```
Flask
```

Redis



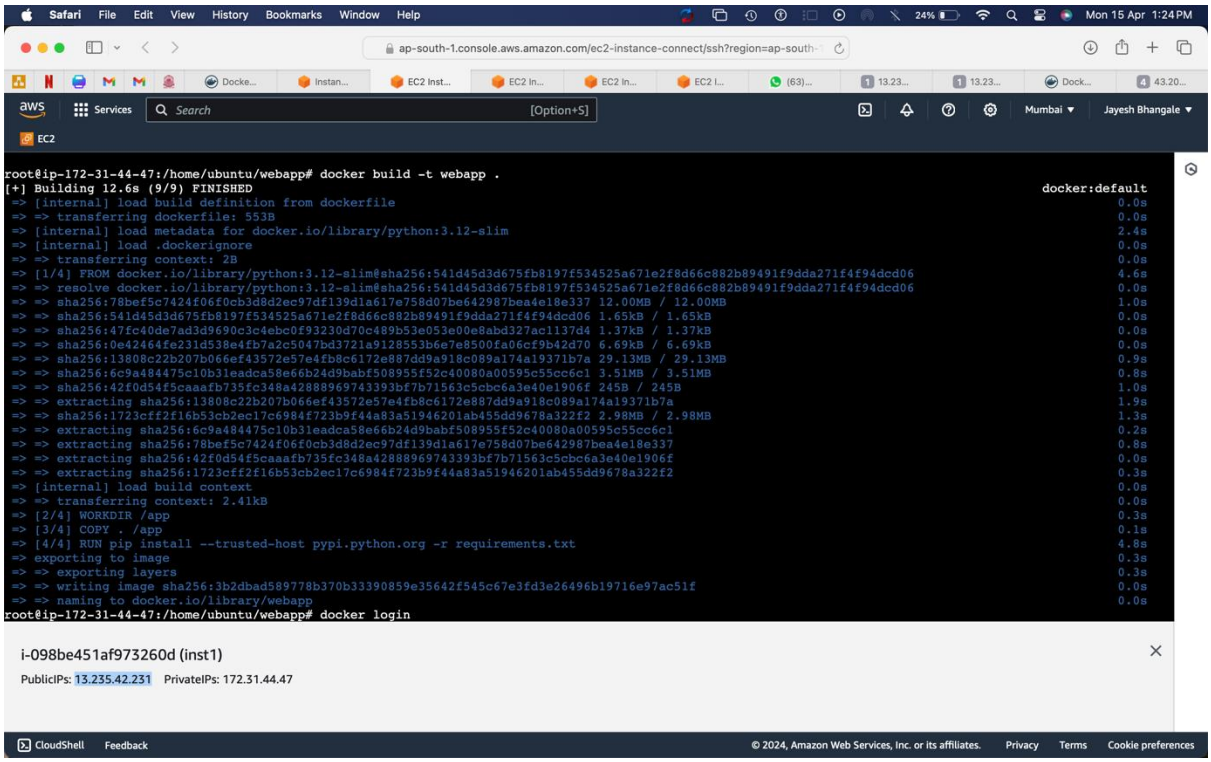
STEP-4: Now build the image and upload it to your Docker Hub.

docker build -t webapp .

docker login

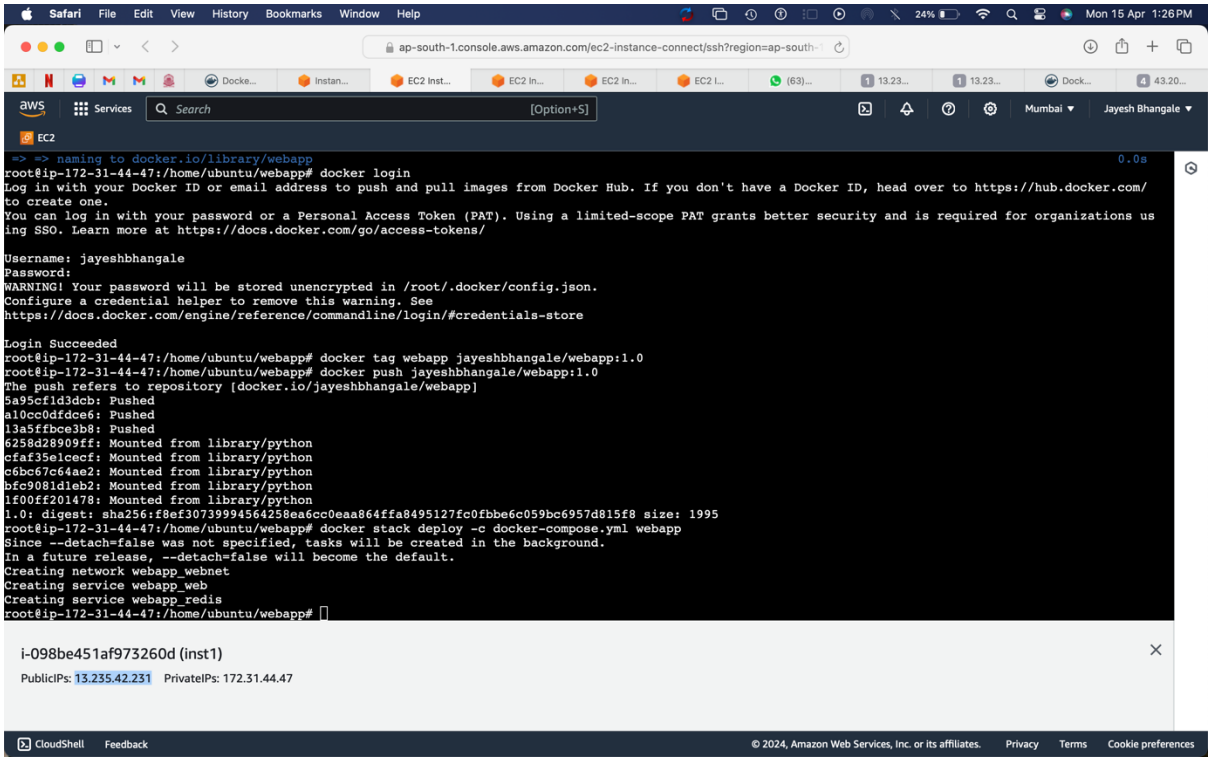
docker tag webapp aniket0724/webapp:1.0

docker push aniket0724/webapp:1.0



STEP -5 : Now deploy the Stack.

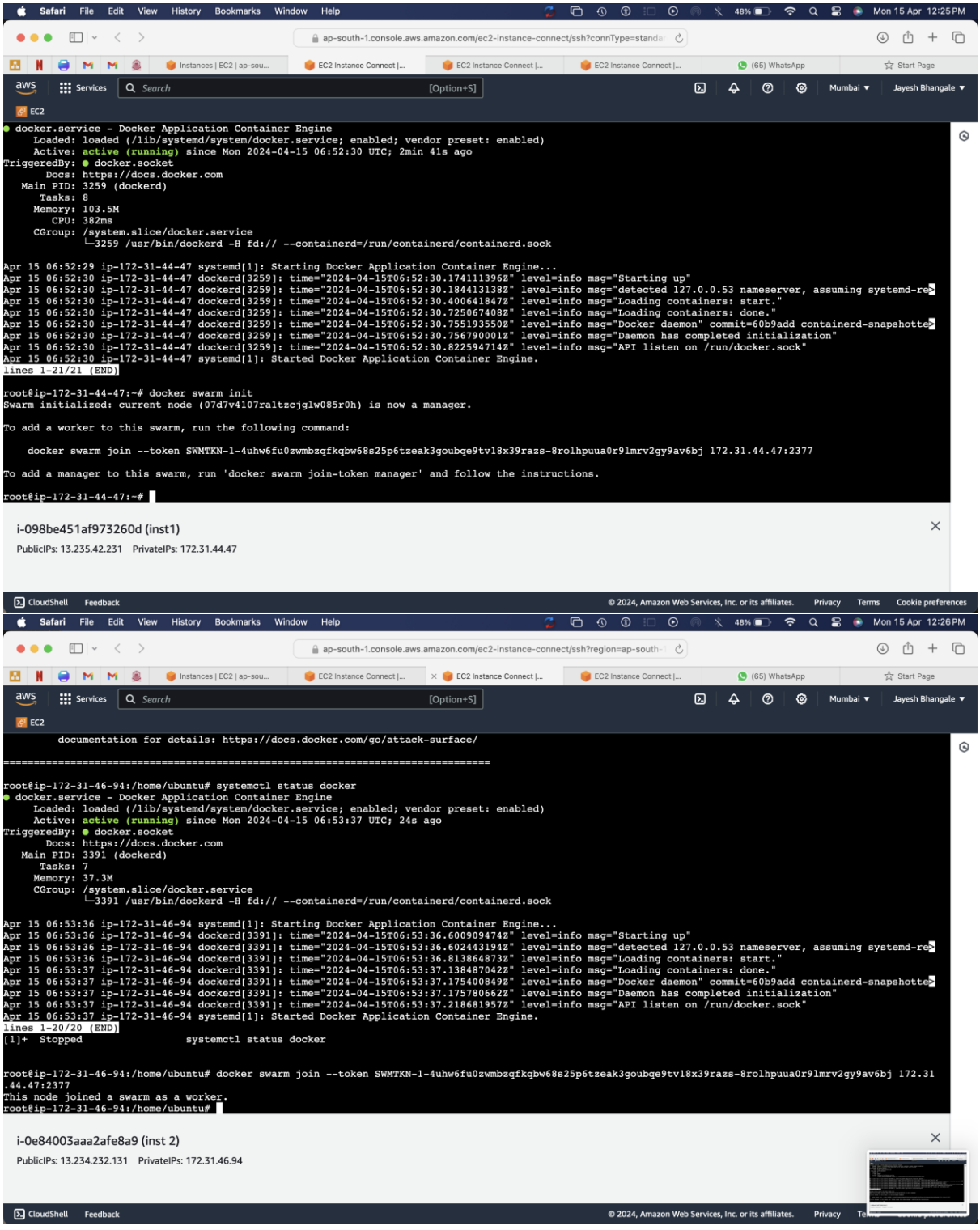
docker stack deploy -c docker-compose.yml webapp

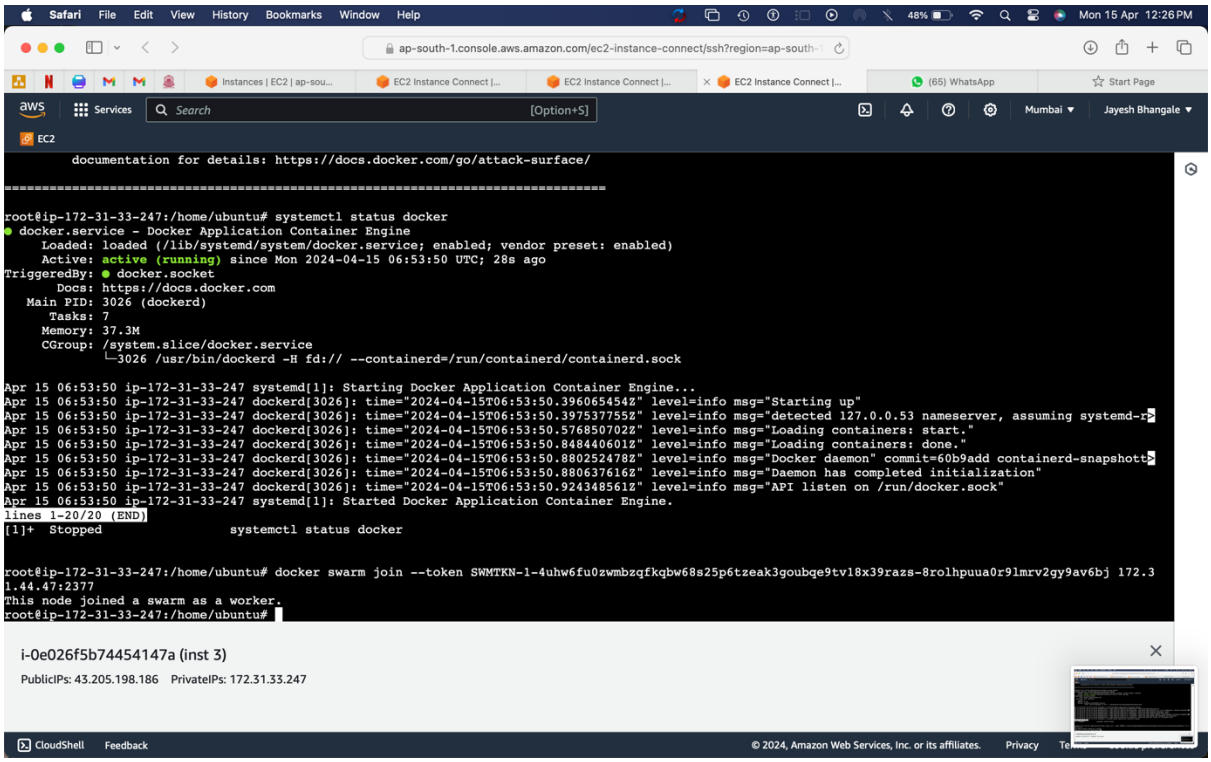


STEP-6: Now connect the other nodes to the manager node.

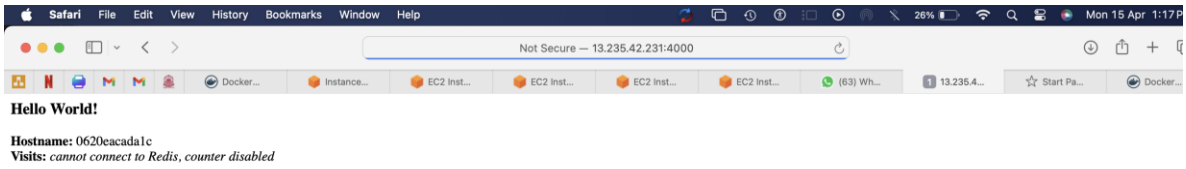
docker swarm join-token worker

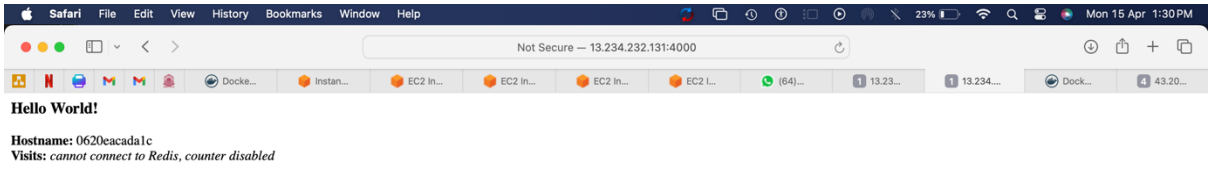
(copy the token and paste it on the worker nodes)

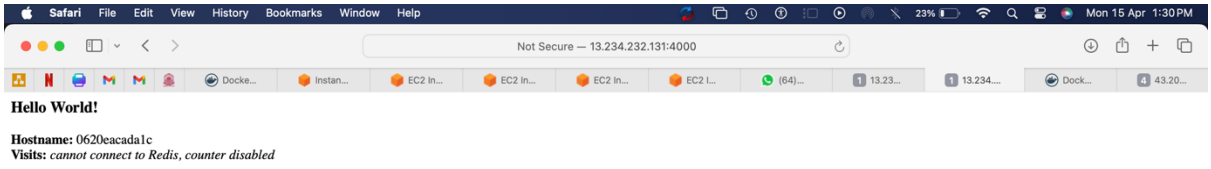

docker.service - Docker Application Container Engine
Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2024-04-15 06:52:30 UTC; 2min 41s ago
TriggeredBy: docker.socket
Docs: https://docs.docker.com
Main PID: 3259 (dockerd)
Tasks: 8
Memory: 103.5M
CPU: 382ms
CGroup: /system.slice/docker.service
└3259 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
Apr 15 06:52:29 ip-172-31-44-47 systemd[1]: Starting Docker Application Container Engine...
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.174111396Z" level=info msg="Starting up"
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.184413138Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-re
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.400641847Z" level=info msg="Loading containers: start."
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.725067408Z" level=info msg="Loading containers: done."
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.755193550Z" level=info msg="Docker daemon" commit=60b9add containerd-snapshotte
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.756790001Z" level=info msg="Daemon has completed initialization"
Apr 15 06:52:30 ip-172-31-44-47 dockerd[3259]: time="2024-04-15T06:52:30.822594714Z" level=info msg="API listen on /run/docker.sock"
Apr 15 06:52:30 ip-172-31-44-47 systemd[1]: Started Docker Application Container Engine.
lines 1-21/21 (END)
root@ip-172-31-44-47:~# docker swarm init
Swarm initialized: current node (07d7v4107ra1tzcjglw085r0h) is now a manager.
To add a worker to this swarm, run the following command:
docker swarm join --token SWMTKN-1-4uhw6fu0zwmbzqfkqbw68s25p6tzeak3goubqe9tv18x39razs-8rolhpuua0r9lmrv2gy9av6bj 172.31.44.47:2377
To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
root@ip-172-31-44-47:~#
i-098be451af973260d (inst1)
PublicIPs: 13.235.42.231 PrivateIPs: 172.31.44.47
documentation for details: https://docs.docker.com/go/attack-surface/
root@ip-172-31-46-94:/home/ubuntu# systemctl status docker
docker.service - Docker Application Container Engine
Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2024-04-15 06:53:37 UTC; 24s ago
TriggeredBy: docker.socket
Docs: https://docs.docker.com
Main PID: 3391 (dockerd)
Tasks: 7
Memory: 37.3M
CGroup: /system.slice/docker.service
└3391 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
Apr 15 06:53:36 ip-172-31-46-94 systemd[1]: Starting Docker Application Container Engine...
Apr 15 06:53:36 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:36.600909474Z" level=info msg="Starting up"
Apr 15 06:53:36 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:36.602443194Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-re
Apr 15 06:53:36 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:36.813864873Z" level=info msg="Loading containers: start."
Apr 15 06:53:37 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:37.138487042Z" level=info msg="Loading containers: done."
Apr 15 06:53:37 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:37.175400849Z" level=info msg="Docker daemon" commit=60b9add containerd-snapshotte
Apr 15 06:53:37 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:37.175780662Z" level=info msg="Daemon has completed initialization"
Apr 15 06:53:37 ip-172-31-46-94 dockerd[3391]: time="2024-04-15T06:53:37.218681957Z" level=info msg="API listen on /run/docker.sock"
Apr 15 06:53:37 ip-172-31-46-94 systemd[1]: Started Docker Application Container Engine.
lines 1-20/20 (END)
[1]+ Stopped systemctl status docker
root@ip-172-31-46-94:/home/ubuntu# docker swarm join --token SWMTKN-1-4uhw6fu0zwmbzqfkqbw68s25p6tzeak3goubqe9tv18x39razs-8rolhpuua0r9lmrv2gy9av6bj 172.31
.44.47:2377
This node joined a swarm as a worker.
root@ip-172-31-46-94:/home/ubuntu#
i-0e84003aaa2afe8a9 (inst 2)
PublicIPs: 13.234.232.131 PrivateIPs: 172.31.46.94


documentation for details: https://docs.docker.com/go/attack-surface/
root@ip-172-31-33-247:/home/ubuntu# systemctl status docker
● docker.service - Docker Application Container Engine
Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
Active: active (running) since Mon 2024-04-15 06:53:50 UTC; 28s ago
TriggeredBy: ● docker.socket
Docs: https://docs.docker.com
Main PID: 3026 (dockerd)
Tasks: 7
Memory: 37.3M
CGroup: /system.slice/docker.service
└─3026 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
Apr 15 06:53:50 ip-172-31-33-247 systemd[1]: Starting Docker Application Container Engine...
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.396065454Z" level=info msg="Starting up"
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.397537755Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-r
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.576850702Z" level=info msg="Loading containers: start."
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.848440601Z" level=info msg="Loading containers: done."
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.880252478Z" level=info msg="Docker daemon" commit=60b9add containerd-snapshott
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.880637616Z" level=info msg="Daemon has completed initialization"
Apr 15 06:53:50 ip-172-31-33-247 dockerd[3026]: time="2024-04-15T06:53:50.924348561Z" level=info msg="API listen on /run/docker.sock"
Apr 15 06:53:50 ip-172-31-33-247 systemd[1]: Started Docker Application Container Engine.
lines 1-20/20 (END)
[1]+ Stopped systemctl status docker
root@ip-172-31-33-247:/home/ubuntu# docker swarm join --token SWMTKN-1-4uhw6fu0zwmbzqfkqbw68s25p6tzeak3goubqe9tv18x39razs-8rolhpuua0r9lmrv2gy9av6bj 172.3
1.44.47:2377
This node joined a swarm as a worker.
root@ip-172-31-33-247:/home/ubuntu#
i-0e026f5b74454147a (inst 3)
PublicIPs: 43.205.198.186 PrivateIPs: 172.31.33.247
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy

Step 7: Now connect to the IP address with the port 4000 of all the nodes.







End Of Practical

Sign:

Subject In charge:

Dr. Swapnil D. Waghmare