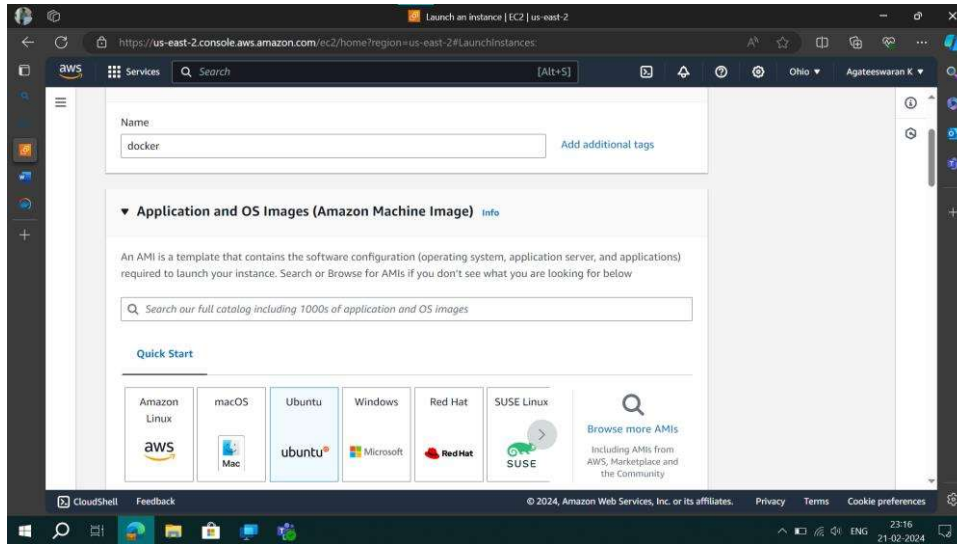


# DOCKER DEMO

Step – 1:

Create an EC2 INSTANCE for docker using AWS.

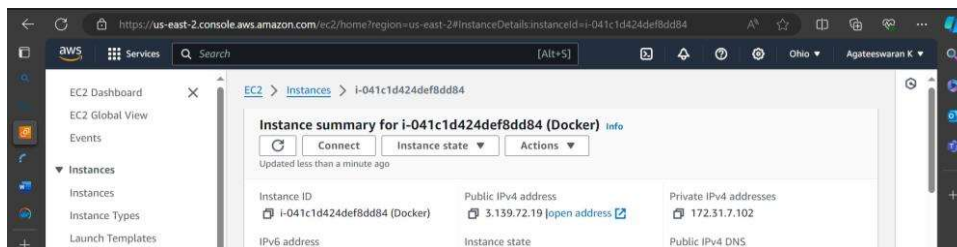


Now launch the instance.

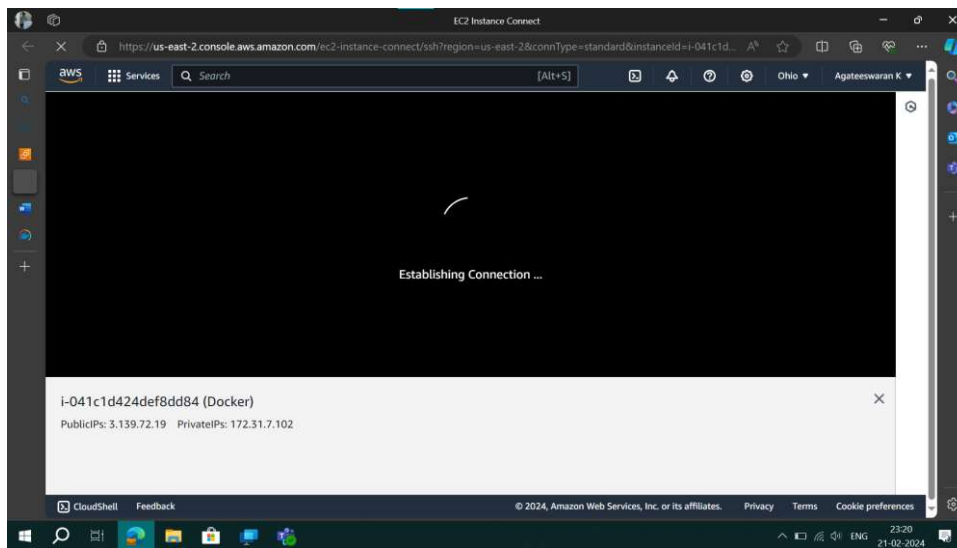


Step – 2:

Select the instance connect it using AWS connect



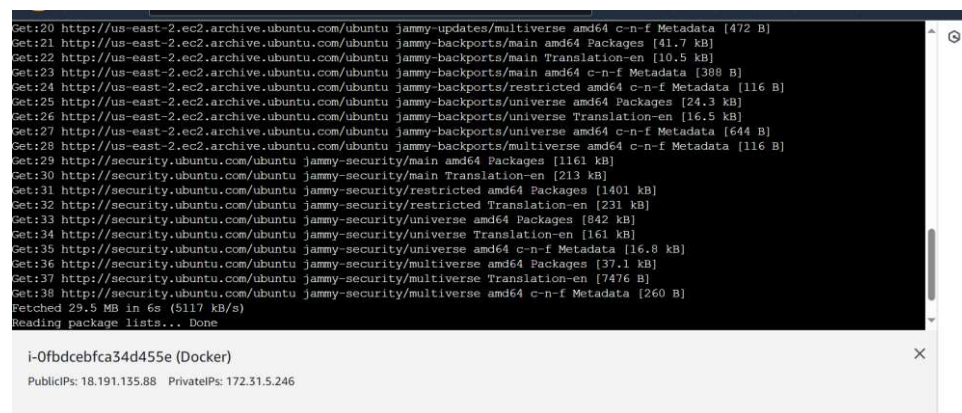
Wait for the AWS connect to establish the connection.



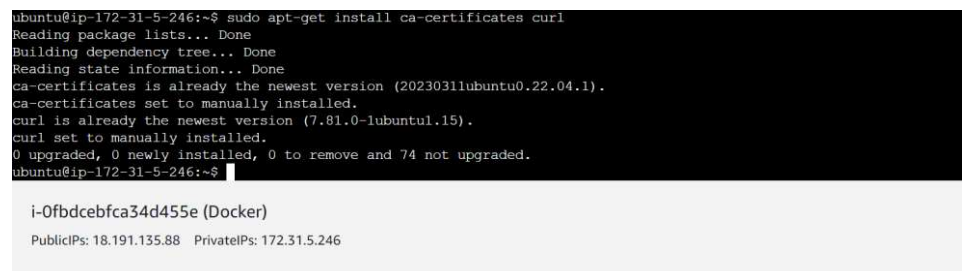
Now we can install the docker in the AWS EC2 instance

Step – 3:

*sudo apt-get update*



*sudo apt-get install ca-certificates curl*



*sudo install -m 0755 -d /etc/apt/keyrings*

*sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc*

*sudo chmod a+r /etc/apt/keyrings/docker.asc*

```
0 upgraded, 0 newly installed, 0 to remove and 74 not upgraded.
ubuntu@ip-172-31-5-246:~$ sudo install -m 0755 -d /etc/apt/keyrings
ubuntu@ip-172-31-5-246:~$ sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
ubuntu@ip-172-31-5-246:~$ sudo chmod a+r /etc/apt/keyrings/docker.asc
ubuntu@ip-172-31-5-246:~$
```

i-Ofbdcebfca34d455e (Docker)

PublicIPs: 18.191.135.88 PrivateIPs: 172.31.5.246

*echo \*  
*"deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]*  
*https://download.docker.com/linux/ubuntu \*  
*\$(. /etc/os-release && echo "\$VERSION\_CODENAME") stable" | \*  
*sudo tee /etc/apt/sources.list.d/docker.list > /dev/null*

*sudo apt-get update*

```
ubuntu@ip-172-31-5-246:~$ echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
ubuntu@ip-172-31-5-246:~$ sudo apt-get update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [26.7 kB]
Fetched 75.5 kB in 1s (93.9 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-5-246:~$
```

i-Ofbdcebfca34d455e (Docker)

PublicIPs: 18.191.135.88 PrivateIPs: 172.31.5.246

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

*sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin*

```
ubuntu@ip-172-31-5-246:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras libbtdl7 libslirp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin libbtdl7
  libslirp0 pigz slirp4netns
0 upgraded, 10 newly installed, 0 to remove and 74 not upgraded.
Need to get 117 MB of archives.
After this operation, 420 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

i-Ofbdcebfca34d455e (Docker)

PublicIPs: 18.191.135.88 PrivateIPs: 172.31.5.246

Step – 4:

*sudo docker run hello-world*

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:d000bc569937abbe195e20322a0bde6b2922d805332fd6d8a68b19f524b7d21d
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
```

i-Ofbdcebfca34d455e (Docker)

PublicIPs: 18.191.135.88 PrivateIPs: 172.31.5.246

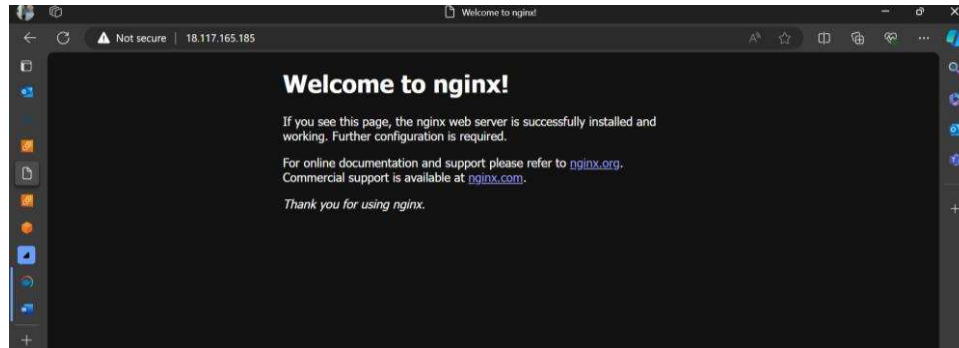
Step – 5:

*docker pull nginx*

*docker run --name some-nginx -d -p 8080:80 some-content-nginx*

```
Using default tag: latest
latest: Pulling from library/nginx
e1caac4eb9d2: Pull complete
88f6f236f401: Pull complete
c3ea3344e711: Pull complete
cc1bb4345a3a: Pull complete
da8fa4352481: Pull complete
c7f80e9edab2: Pull complete
18a869624cb6: Pull complete
Digest: sha256:c26ae7472d624balfafd296e73cecc4f93f853088e6a9c13cd52f6ca5865107
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
ubuntu@ip-172-31-9-85:~$ sudo docker run --name new-nginx -d -p 80:80 nginx
6d7778b1b4dc408016a77e8dc853bb4687a359df0d5ea7bf28550dbee6205285
ubuntu@ip-172-31-9-85:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
6d7778b1b4dc   nginx     "/docker-entrypoint...." 5 seconds ago  Up 4 seconds  0.0.0.0:80->80/tcp, :::80->80/tcp  new-nginx
ubuntu@ip-172-31-9-85:~$
```

To access the instance public IP and add the port no at back to access the docker image.



To customize the nginx server we can use the command “sudo docker exec –it containerid /bin/bash.”

```
Last login: Thu Feb 22 04:37:24 2024 from 3.16.146.3
ubuntu@ip-172-31-9-85:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
6d7778b1b4dc   nginx     "/docker-entrypoint...." 36 minutes ago  Up 36 minutes  0.0.0.0:80->80/tcp, :::80->80/tcp  new-nginx
ubuntu@ip-172-31-9-85:~$ ^C
ubuntu@ip-172-31-9-85:~$ ^C
ubuntu@ip-172-31-9-85:~$ sudo docker exec -it 6d7778b1b4dc /bin/bash
```

i-Oeaa1b505670c79c8 (Docker)

PublicIPs: 18.117.165.185 PrivateIPs: 172.31.9.85

