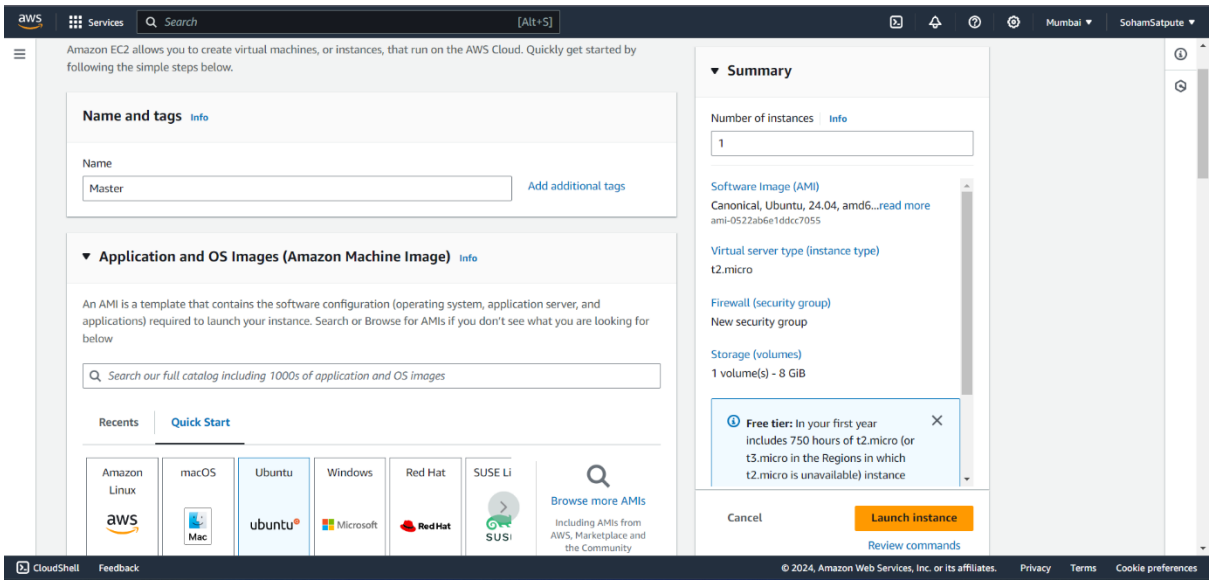
ADVANCE DEVOPS EXP-3

Name: Soham Satpute

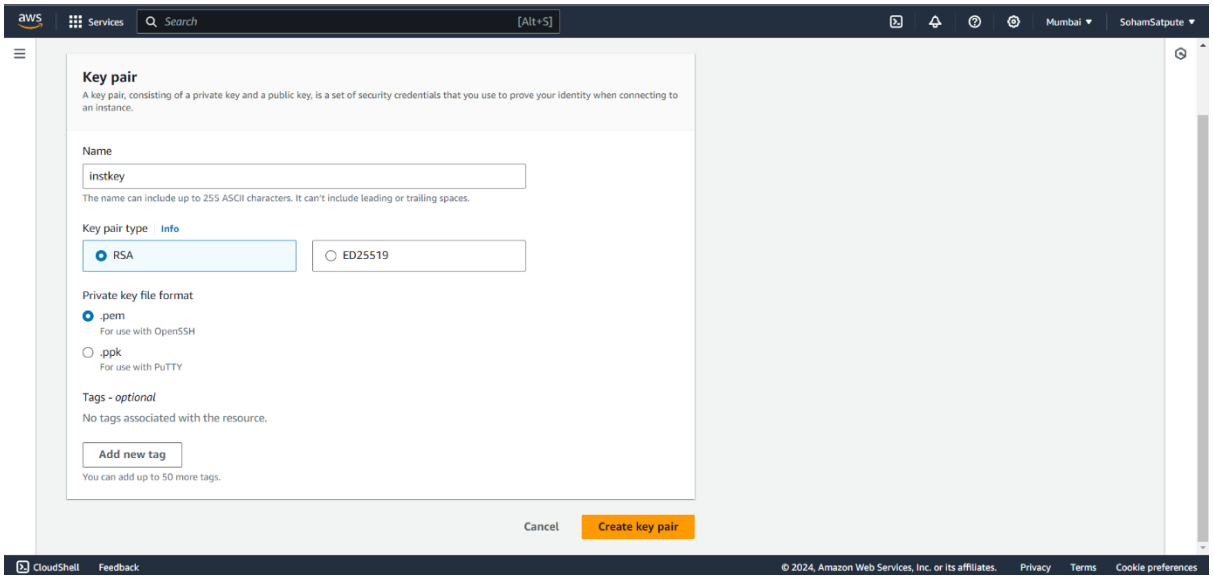
Roll No:52

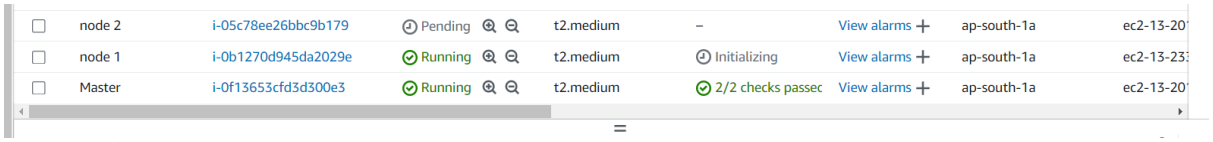
Aim: To understand the Kubernetes Cluster Architecture, install and Spin Up a Kubernetes Cluster on Linux Machines/Cloud Platforms.

Step 1: Create 2 Security Groups for Master and Nodes and add the following inbound rules in those groups:



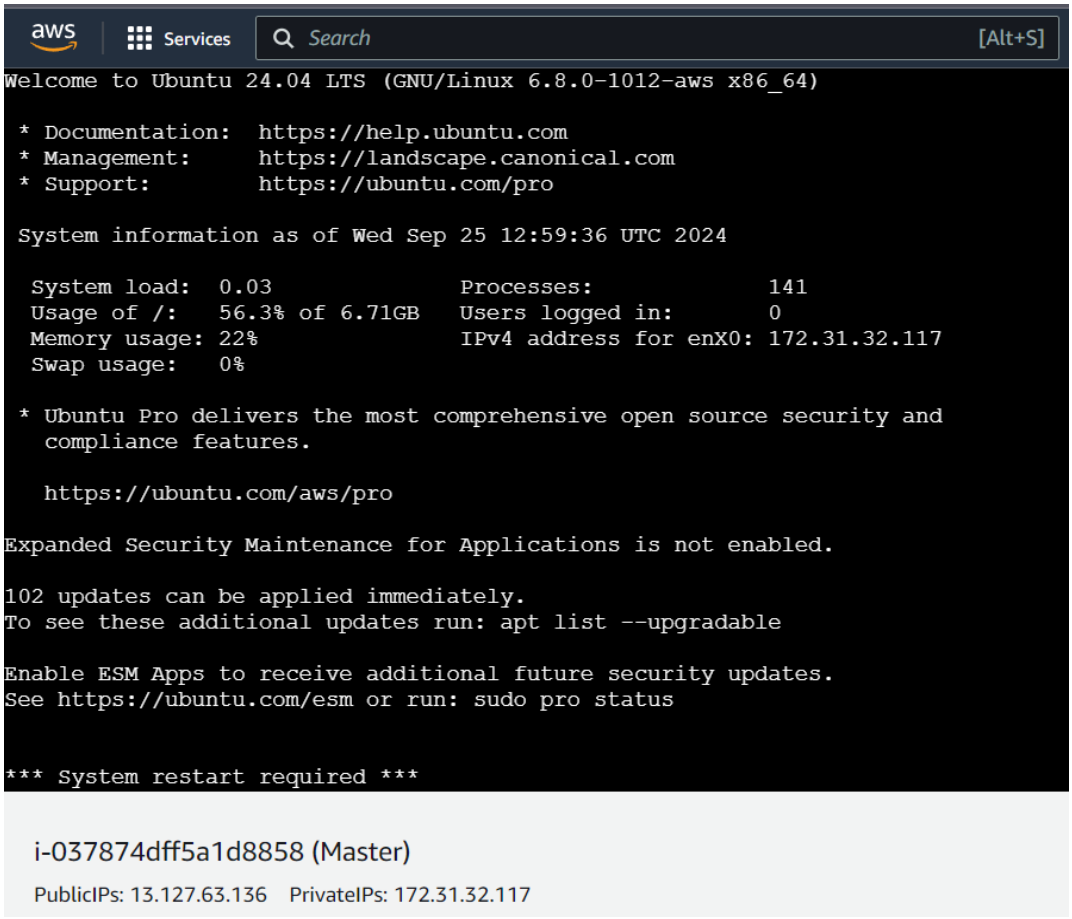
Generate a key pair for the same:

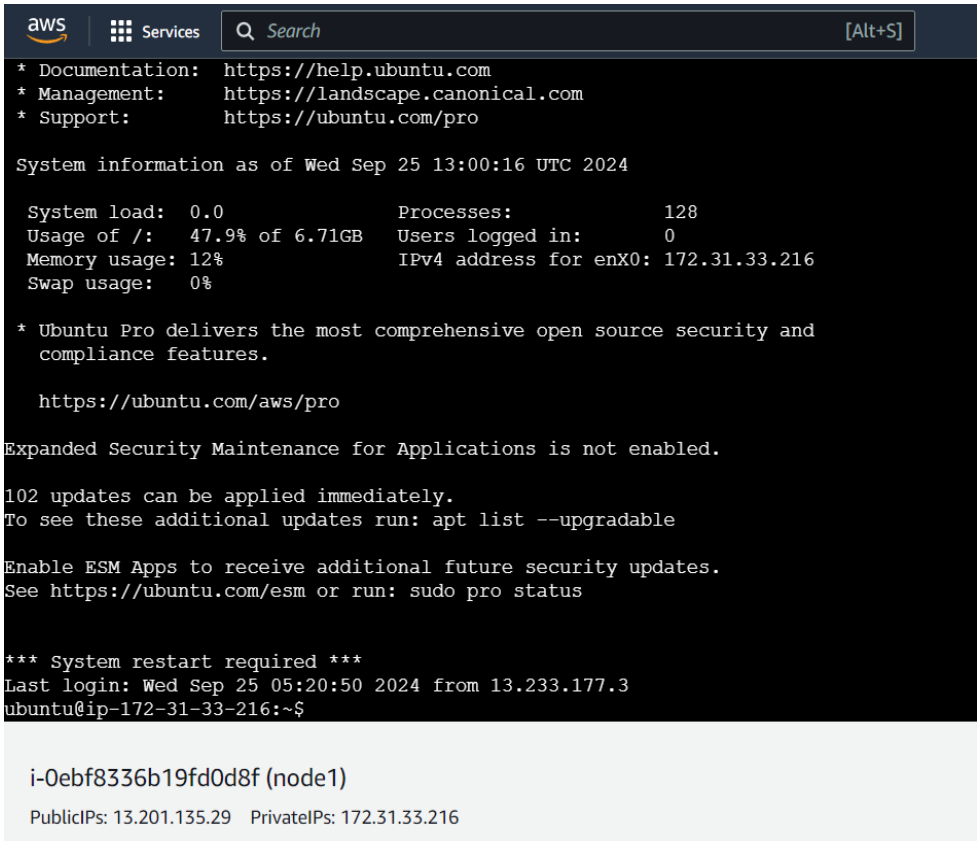




Step2:

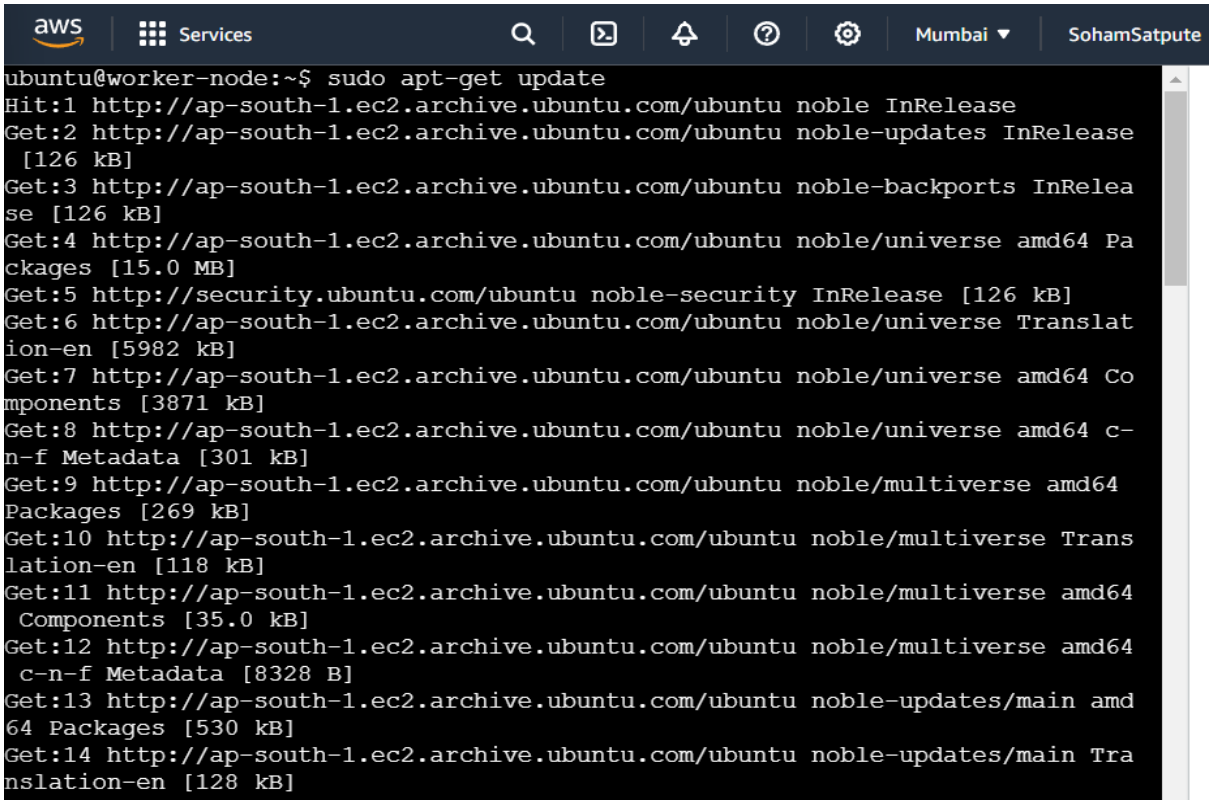
Open Master and node on EC2 terminal:

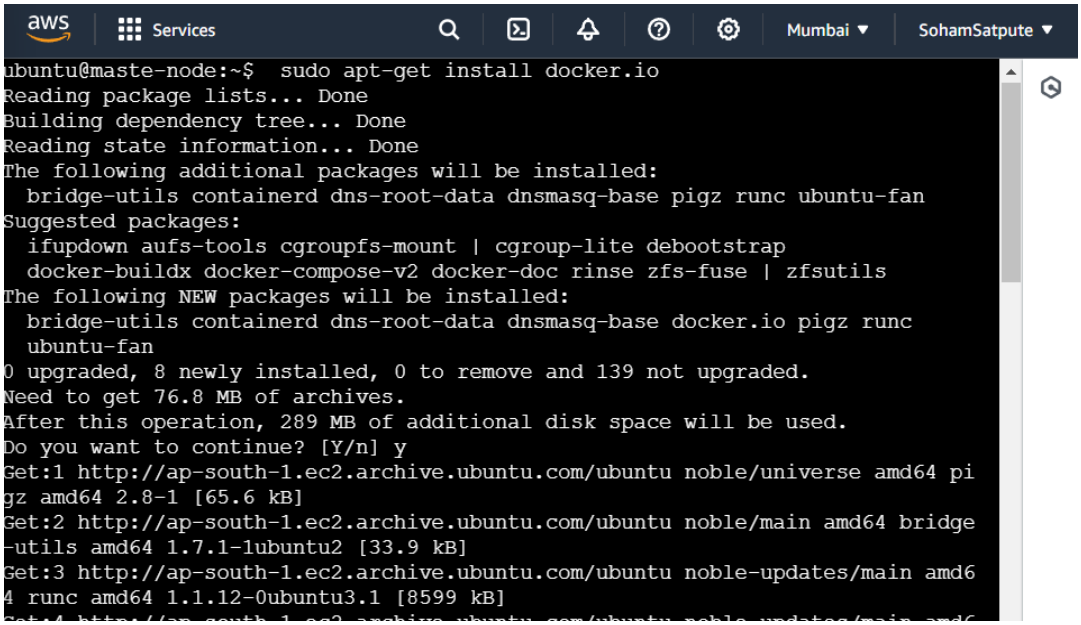


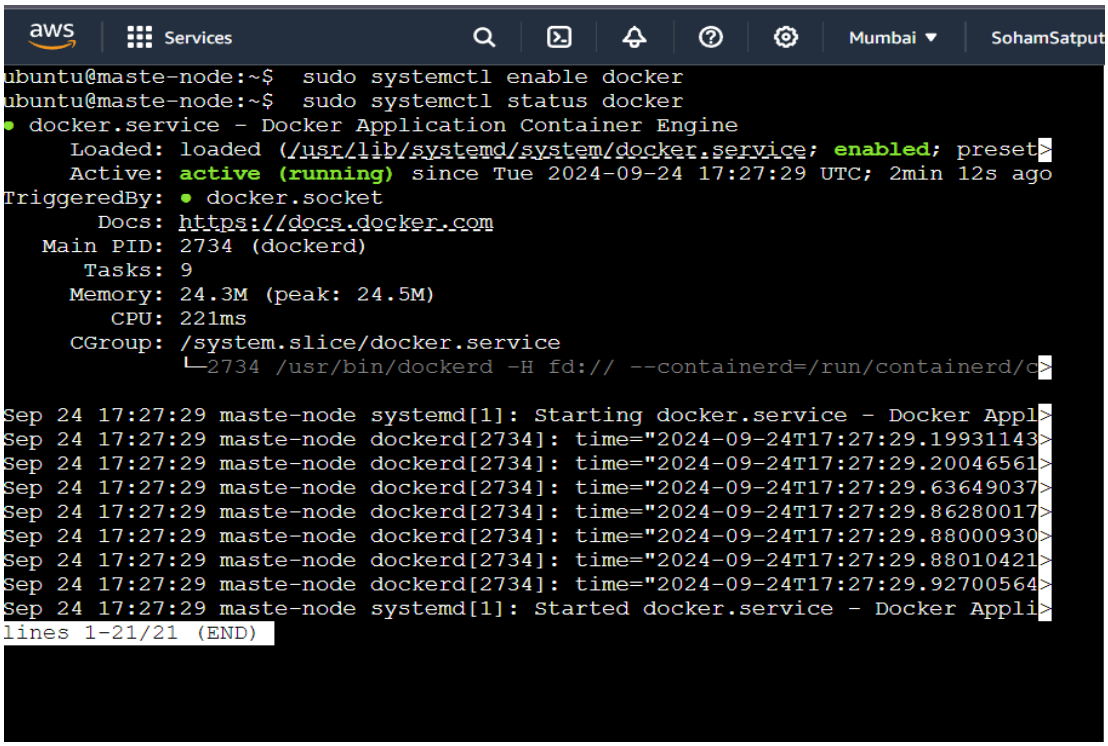


Step 3:

Install Docker

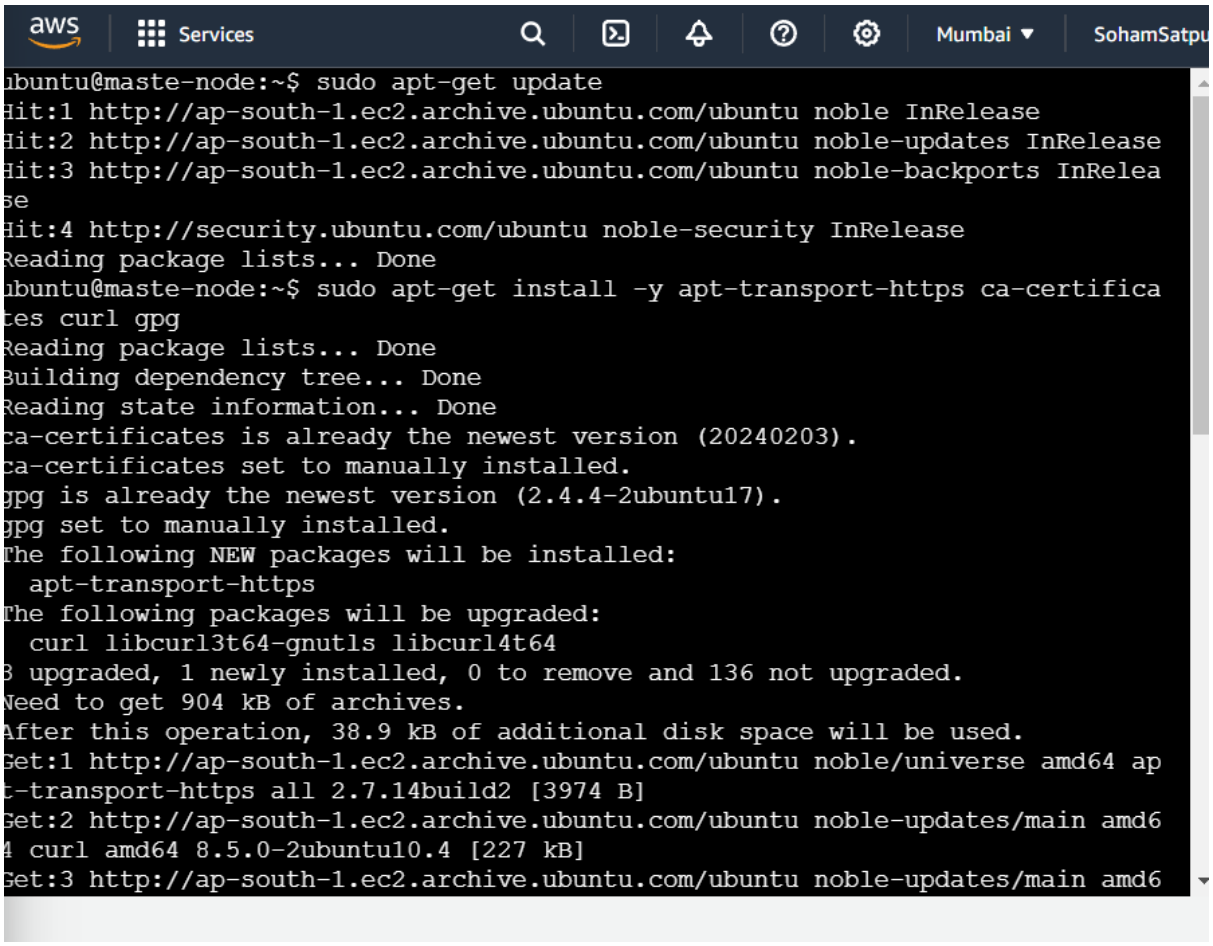


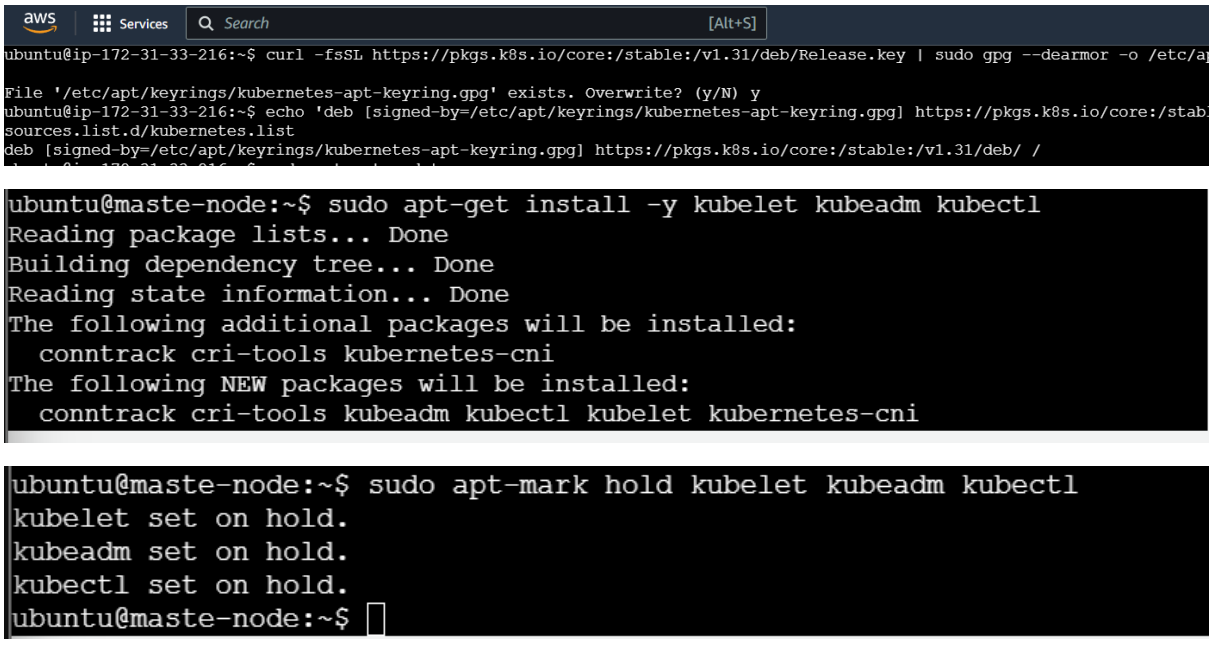

aws
Services
Mumbai
SohamSatpute
ubuntu@maste-node:~$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap
docker-buildx docker-compose-v2 docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc
ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 139 not upgraded.
Need to get 76.8 MB of archives.
After this operation, 289 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 pi
gz amd64 2.8-1 [65.6 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 bridge
-utils amd64 1.7.1-1ubuntu2 [33.9 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd6
4 runc amd64 1.1.12-0ubuntu3.1 [8599 kB]


aws
Services
Mumbai
SohamSatput
ubuntu@maste-node:~$ sudo systemctl enable docker
ubuntu@maste-node:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset
Active: active (running) since Tue 2024-09-24 17:27:29 UTC; 2min 12s ago
TriggeredBy: ● docker.socket
Docs: https://docs.docker.com
Main PID: 2734 (dockerd)
Tasks: 9
Memory: 24.3M (peak: 24.5M)
CPU: 221ms
CGroup: /system.slice/docker.service
└─2734 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/c
Sep 24 17:27:29 maste-node systemd[1]: Starting docker.service - Docker Appl
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.19931143
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.20046561
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.63649037
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.86280017
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.88000930
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.88010421
Sep 24 17:27:29 maste-node dockerd[2734]: time="2024-09-24T17:27:29.92700564
Sep 24 17:27:29 maste-node systemd[1]: Started docker.service - Docker Appli
lines 1-21/21 (END)

Step 4:

Install kubeadm, kubelet, kubectl:





Step5:

Disable Swap (Kubernetes requires swap to be off):

```
ubuntu@maste-node:~$ sudo swapoff -a
ubuntu@maste-node:~$
```

Step 6:

Initialize the Kubernetes Cluster on Master Node On the master node:

sudo kubeadm init --pod-network-cidr=10.244.0.0/16

```
aws | Services | Mumbai | SohamSatput
ubuntu@maste-node:~$  sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ig
nore-preflight-errors=all

[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
        [WARNING FileExisting-socat]: socat not found in system path
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your i
nternet connection
[preflight] You can also perform this action beforehand using 'kubeadm config
 images pull'
W0924 17:41:13.236064    4176 checks.go:846] detected that the sandbox image
"registry.k8s.io/pause:3.8" of the container runtime is inconsistent with tha
t used by kubeadm.It is recommended to use "registry.k8s.io/pause:3.10" as th
e CRI sandbox image.
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [kubernetes kubernetes
.default kubernetes.default.svc kubernetes.default.svc.cluster.local maste-no
de] and IPs [10.96.0.1 172.31.32.117]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [localhost maste-nod
e] and IPs [172.31.32.117 127.0.0.1 ::1]
```

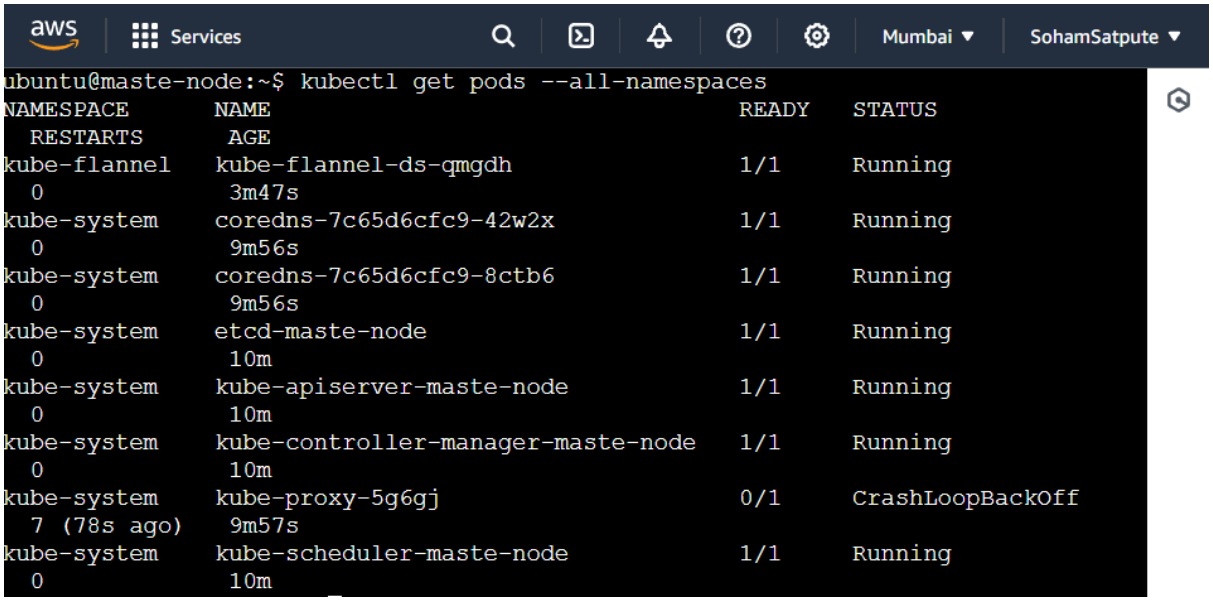
Set up kubectl on the master node:

mkdir -p $HOME/.kube

sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

sudo chown $(id -u):$(id -g) $HOME/.kube/config

```
kubeadm join 172.31.32.117:6443 --token t2jpj2.rauz0s7fimwpdo4a --discovery-token-ca-cert-hash
sha256:1fa6fa408d342aef675f4bab47ff1c02da288b9cf34dd1cff8161d84586cd50b
```



Step 7:

Join Worker Nodes to the Cluster On the worker nodes, run the command provided by the master node during initialization:

Step 8:

Verify the Cluster Once the worker node joins, check the status on the master node