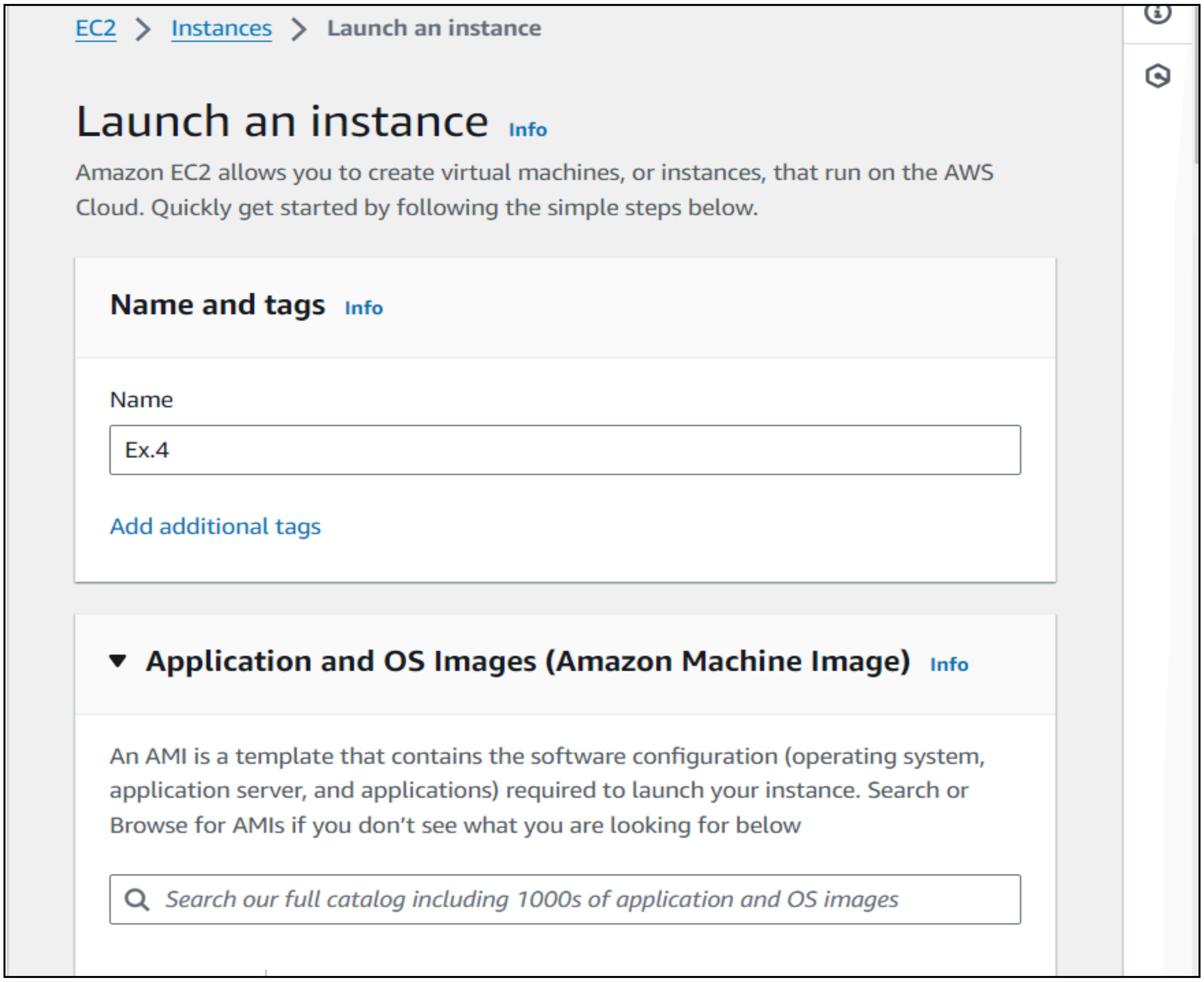
Shreya Sawant D15A -54

EXPERIMENT.4

Aim: To install Kubectl and execute Kubectl commands to manage the Kubernetes cluster and deploy Your First Kubernetes Application.

Step.1 Create new instance .



EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name

Ex.4

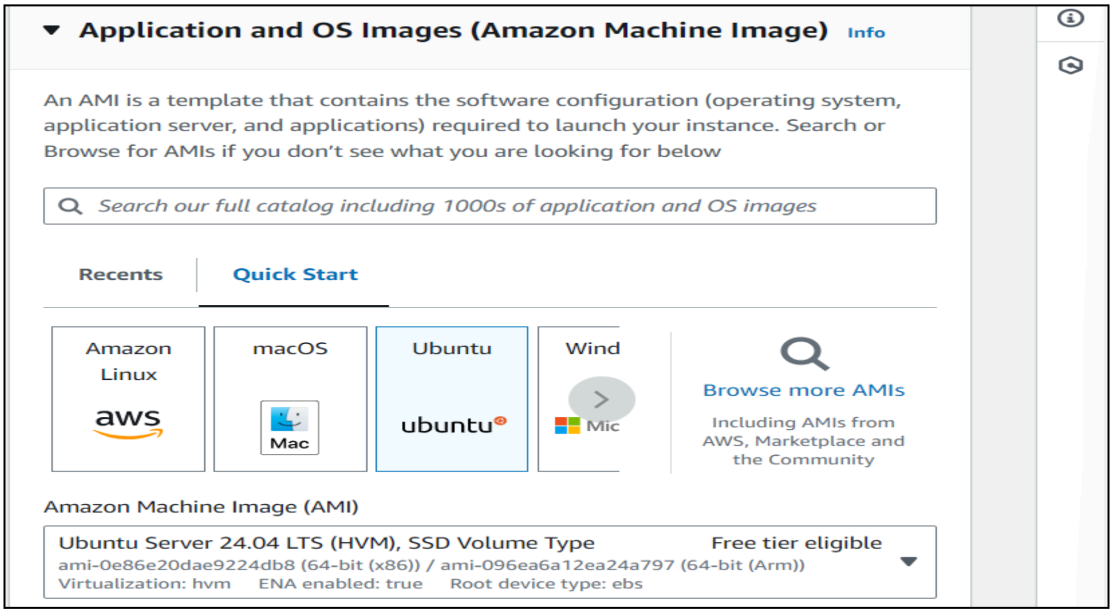
Add additional tags

▾ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Shreya Sawant D15A -54



Application and OS Images (Amazon Machine Image) Info

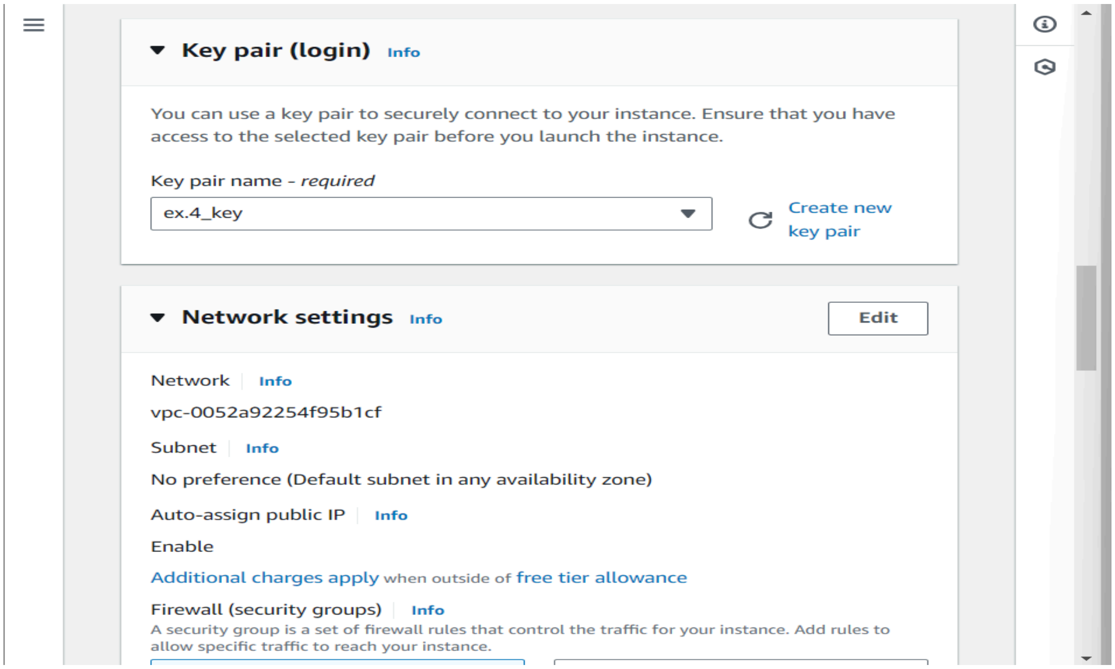
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
aws
macOS
Mac
Ubuntu
ubuntu
Wind
Mic
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type Free tier eligible
ami-0e86e20dae9224db8 (64-bit (x86)) / ami-096ea6a12ea24a797 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Create a key pair.



Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

ex.4_key

Network settings Info

Network Info
vpc-0052a92254f95b1cf

Subnet Info
No preference (Default subnet in any availability zone)

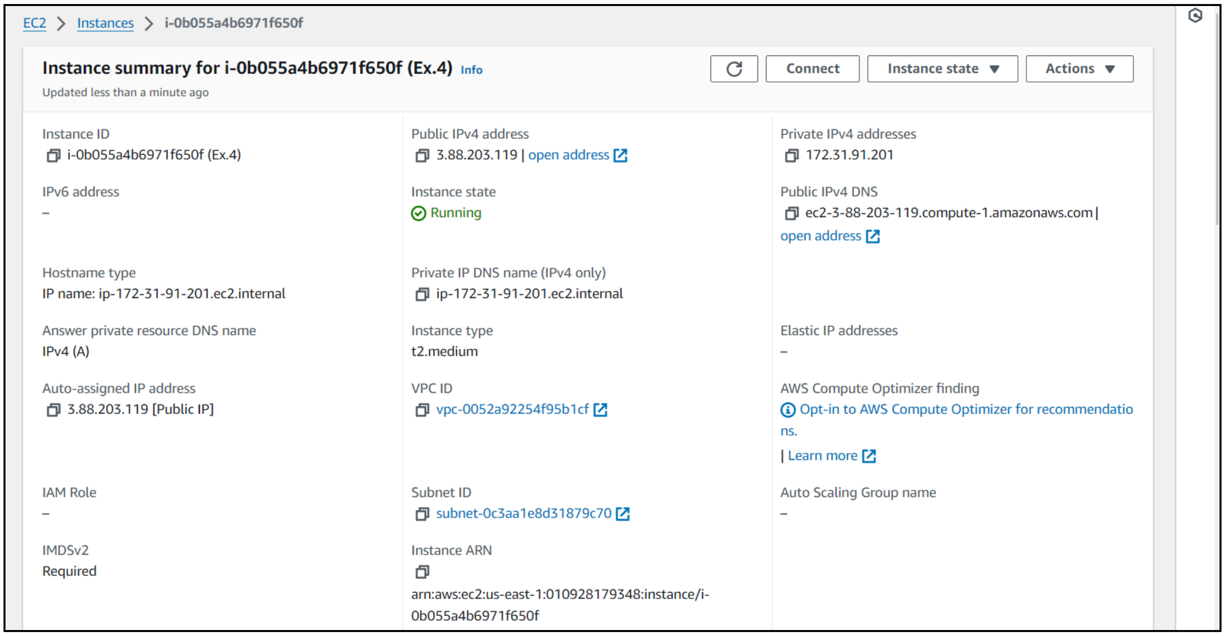
Auto-assign public IP Info
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Shreya Sawant D15A -54

Click on connect and check ssh link.

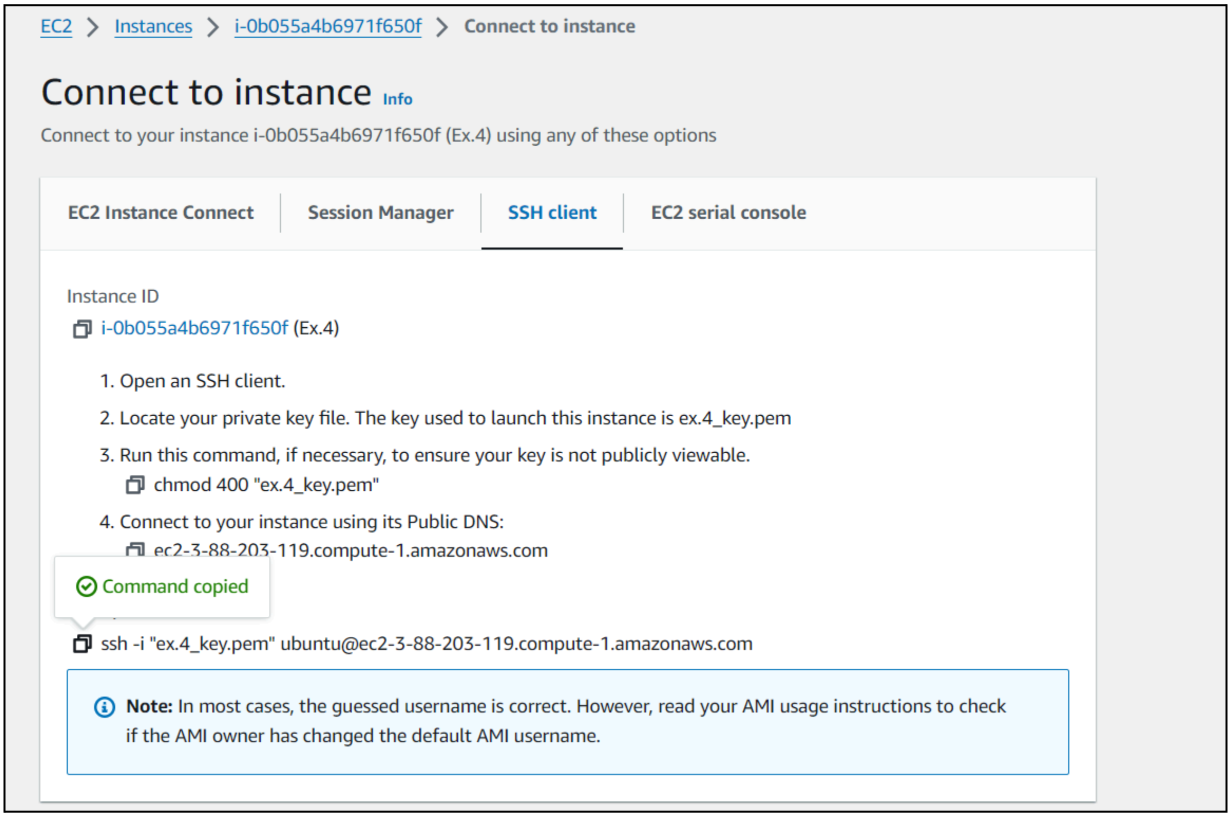


EC2 > Instances > i-0b055a4b6971f650f

Instance summary for i-0b055a4b6971f650f (Ex.4) Info

Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0b055a4b6971f650f (Ex.4)	3.88.203.119 | open address	172.31.91.201
IPv6 address	Instance state	Public IPv4 DNS
–	Running	ec2-3-88-203-119.compute-1.amazonaws.com | open address
Hostname type	Private IP DNS name (IPv4 only)	
IP name: ip-172-31-91-201.ec2.internal	ip-172-31-91-201.ec2.internal	
Answer private resource DNS name	Instance type	Elastic IP addresses
IPv4 (A)	t2.medium	–
Auto-assigned IP address	VPC ID	AWS Compute Optimizer finding
3.88.203.119 [Public IP]	vpc-0052a92254f95b1cf	Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role	Subnet ID	Auto Scaling Group name
–	subnet-0c3aa1e8d31879c70	–
IMDSv2	Instance ARN	
Required	arn:aws:ec2:us-east-1:010928179348:instance/i-0b055a4b6971f650f	



EC2 > Instances > i-0b055a4b6971f650f > Connect to instance

Connect to instance Info

Connect to your instance i-0b055a4b6971f650f (Ex.4) using any of these options

Instance ID

i-0b055a4b6971f650f (Ex.4)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is ex.4_key.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.

```
chmod 400 "ex.4_key.pem"
```

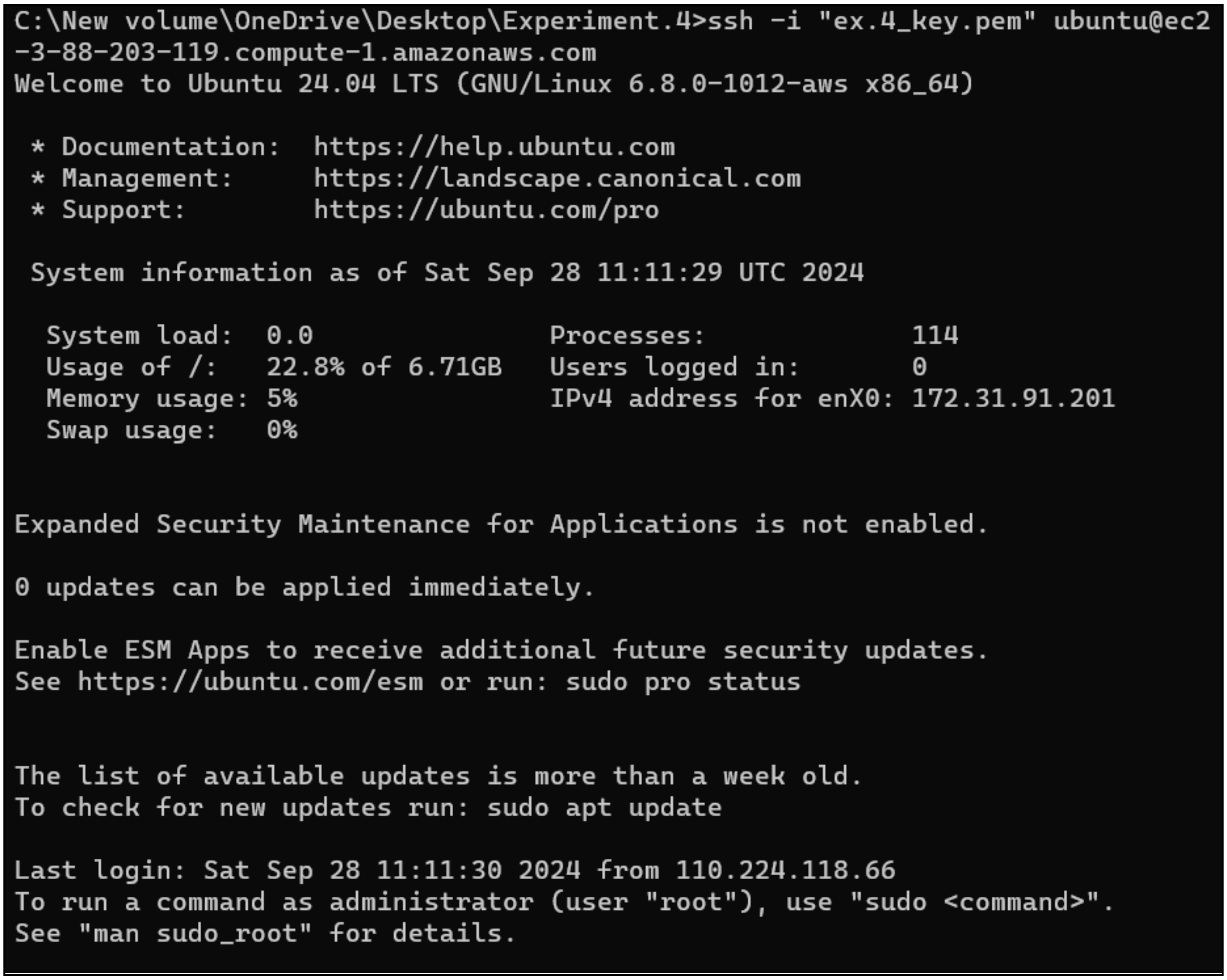
4. Connect to your instance using its Public DNS:

```
ec2-3-88-203-119.compute-1.amazonaws.com
```

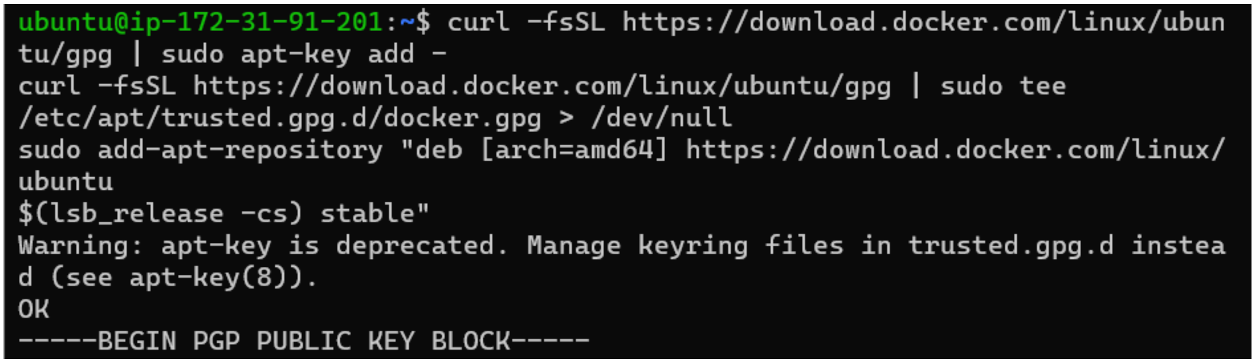
Command copied

```
ssh -i "ex.4_key.pem" ubuntu@ec2-3-88-203-119.compute-1.amazonaws.com
```

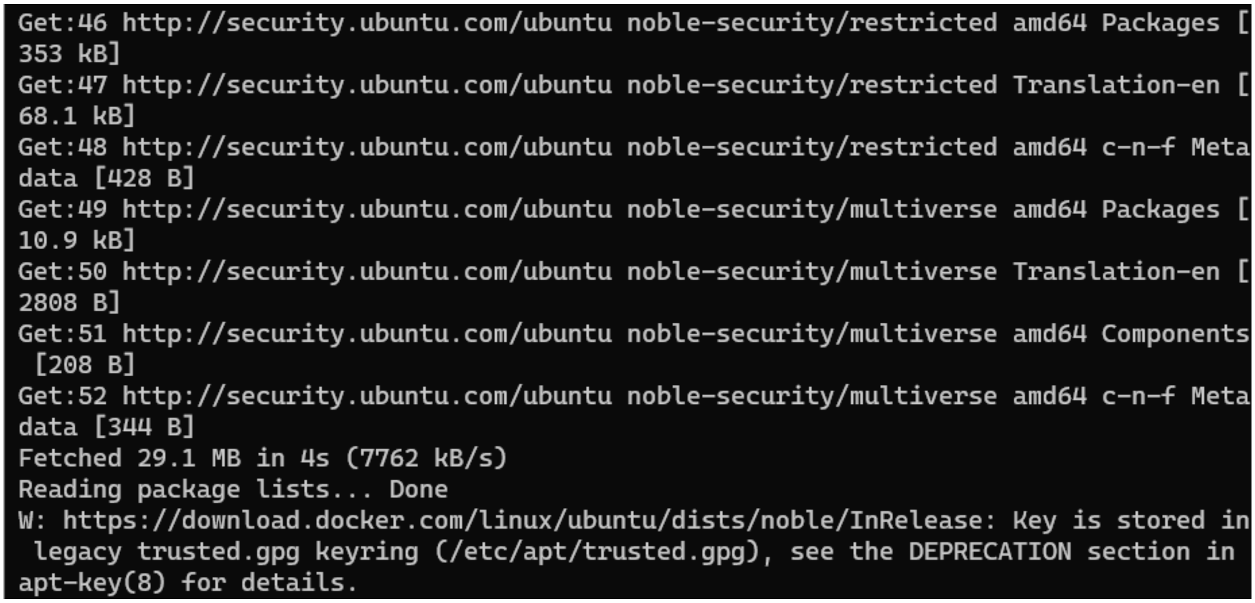
Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.



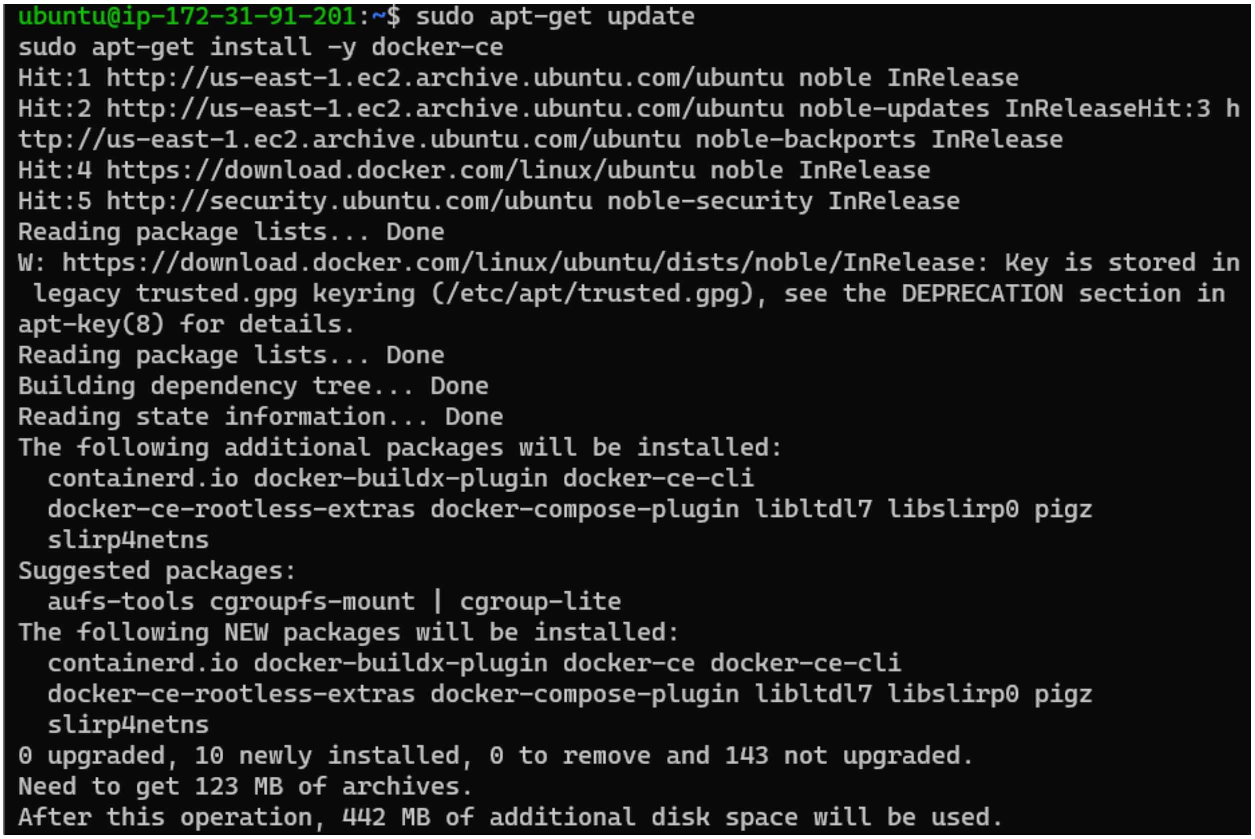
```
C:\New volume\OneDrive\Desktop\Experiment.4>ssh -i "ex.4_key.pem" ubuntu@ec2
-3-88-203-119.compute-1.amazonaws.com
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Sat Sep 28 11:11:29 UTC 2024

  System load:  0.0               Processes:             114
  Usage of /:   22.8% of 6.71GB   Users logged in:       0
  Memory usage: 5%                IPv4 address for enX0: 172.31.91.201
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sat Sep 28 11:11:30 2024 from 110.224.118.66
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```



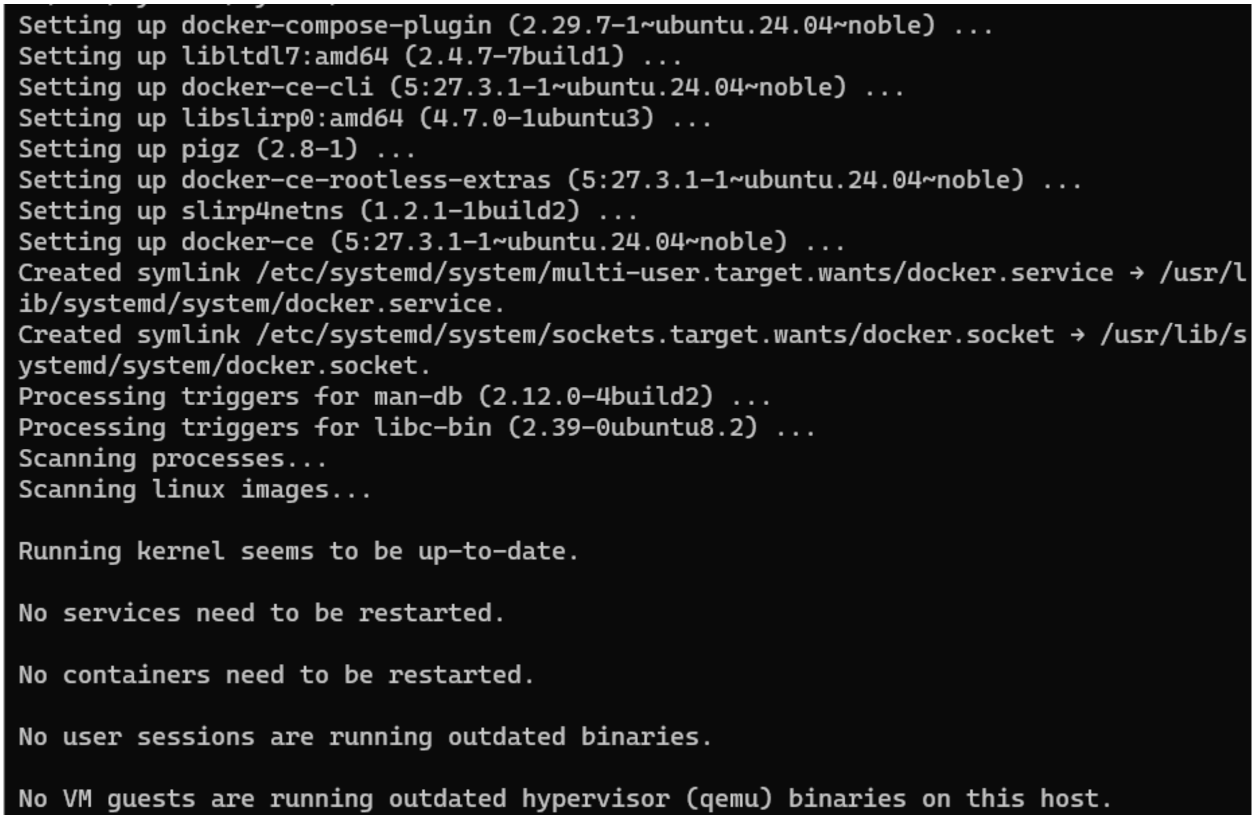
```
ubuntu@ip-172-31-91-201:~$ curl -fsSL https://download.docker.com/linux/ubun
tu/gpg | sudo apt-key add -
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo tee
/etc/apt/trusted.gpg.d/docker.gpg > /dev/null
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/
ubuntu
$(lsb_release -cs) stable"
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instea
d (see apt-key(8)).
OK
-----BEGIN PGP PUBLIC KEY BLOCK-----
```



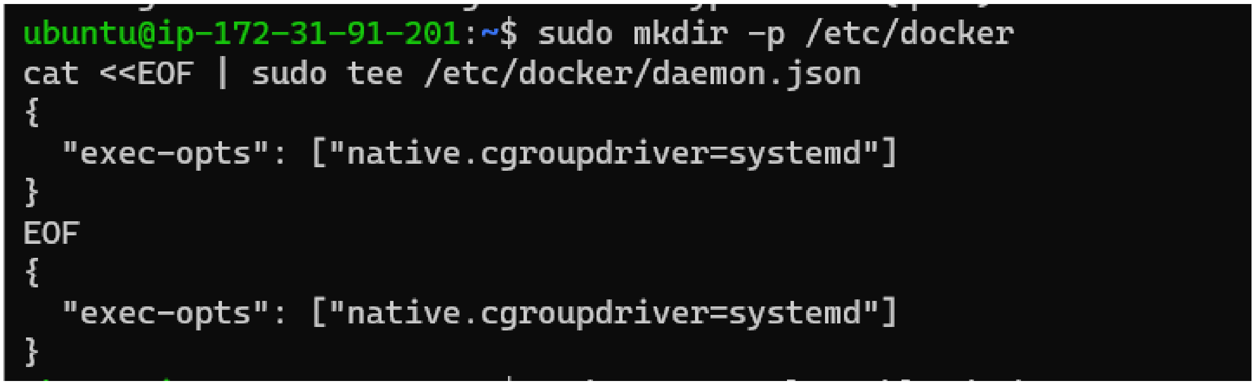
```
Get:46 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [
353 kB]
Get:47 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [
68.1 kB]
Get:48 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Meta
data [428 B]
Get:49 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [
10.9 kB]
Get:50 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [
2808 B]
Get:51 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components
 [208 B]
Get:52 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Meta
data [344 B]
Fetched 29.1 MB in 4s (7762 kB/s)
Reading package lists... Done
W: https://download.docker.com/linux/ubuntu/dists/noble/InRelease: Key is stored in
 legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in
apt-key(8) for details.
```



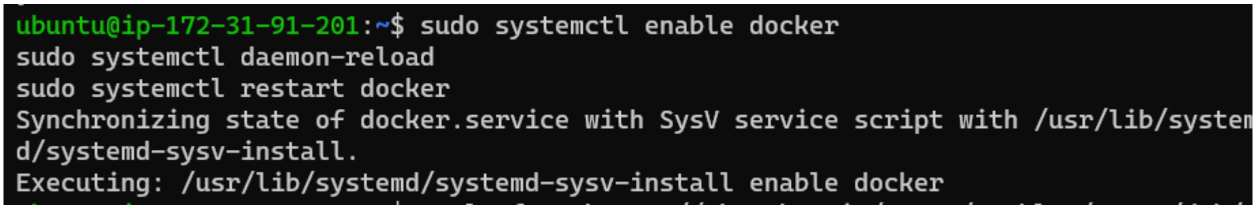
```
ubuntu@ip-172-31-91-201:~$ sudo apt-get update
sudo apt-get install -y docker-ce
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InReleaseHit:3 h
ttp://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu noble InRelease
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
W: https://download.docker.com/linux/ubuntu/dists/noble/InRelease: Key is stored in
 legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in
apt-key(8) for details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  containerd.io docker-buildx-plugin docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin libltdl7 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 libltdl7 libslirp0 pigz
  slirp4netns
0 upgraded, 10 newly installed, 0 to remove and 143 not upgraded.
Need to get 123 MB of archives.
After this operation, 442 MB of additional disk space will be used.
```



```
Setting up docker-compose-plugin (2.29.7-1~ubuntu.24.04~noble) ...
Setting up libltdl7:amd64 (2.4.7-7build1) ...
Setting up docker-ce-cli (5:27.3.1-1~ubuntu.24.04~noble) ...
Setting up libslirp0:amd64 (4.7.0-1ubuntu3) ...
Setting up pigz (2.8-1) ...
Setting up docker-ce-rootless-extras (5:27.3.1-1~ubuntu.24.04~noble) ...
Setting up slirp4netns (1.2.1-1build2) ...
Setting up docker-ce (5:27.3.1-1~ubuntu.24.04~noble) ...
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/l
ib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/s
ystemd/system/docker.socket.
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```



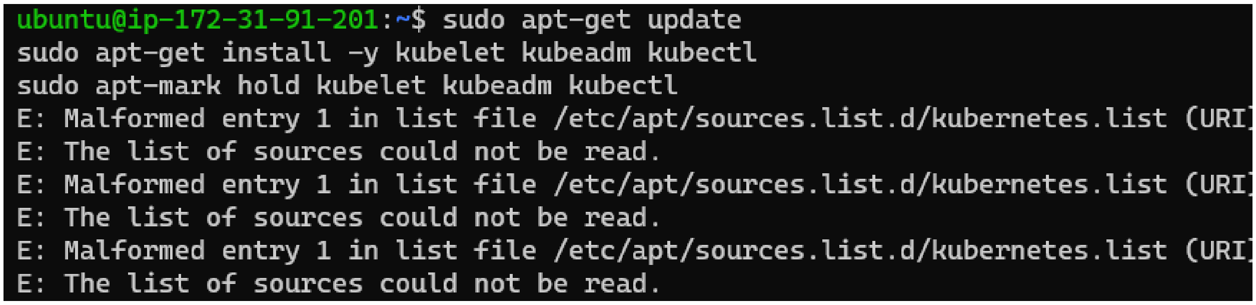
```
ubuntu@ip-172-31-91-201:~$ sudo mkdir -p /etc/docker
cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"]
}
EOF
{
  "exec-opts": ["native.cgroupdriver=systemd"]
}
```



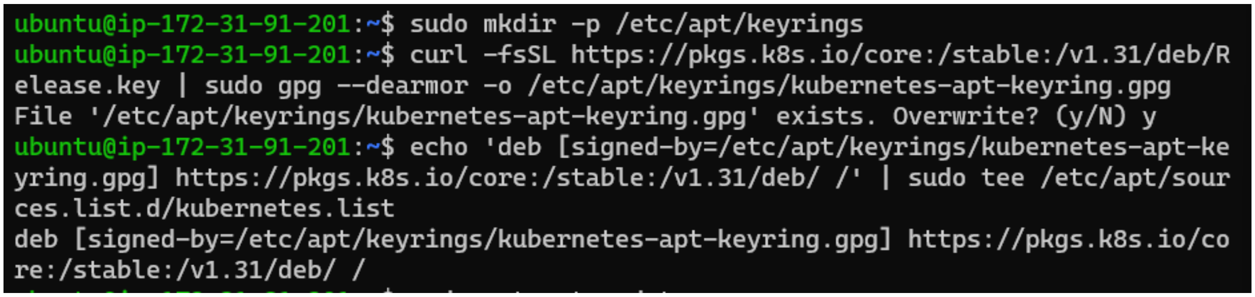
```
ubuntu@ip-172-31-91-201:~$ sudo systemctl enable docker
sudo systemctl daemon-reload
sudo systemctl restart docker
Synchronizing state of docker.service with SysV service script with /usr/lib/system
d/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable docker
```

Shreya Sawant D15A -54

Error-

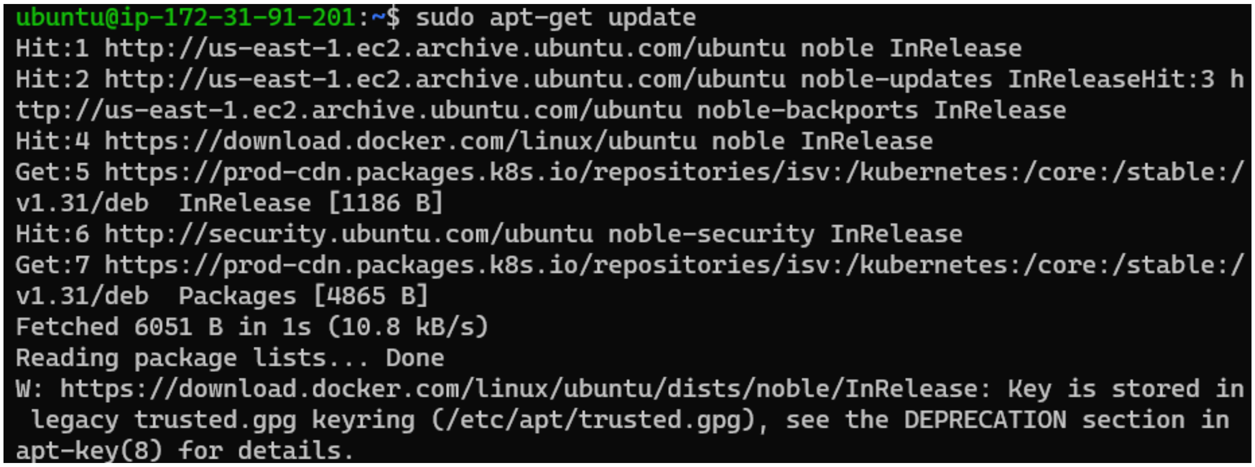


```
ubuntu@ip-172-31-91-201:~$ sudo apt-get update
sudo apt-get install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
E: Malformed entry 1 in list file /etc/apt/sources.list.d/kubernetes.list (URI]
E: The list of sources could not be read.
E: Malformed entry 1 in list file /etc/apt/sources.list.d/kubernetes.list (URI]
E: The list of sources could not be read.
E: Malformed entry 1 in list file /etc/apt/sources.list.d/kubernetes.list (URI]
E: The list of sources could not be read.
```



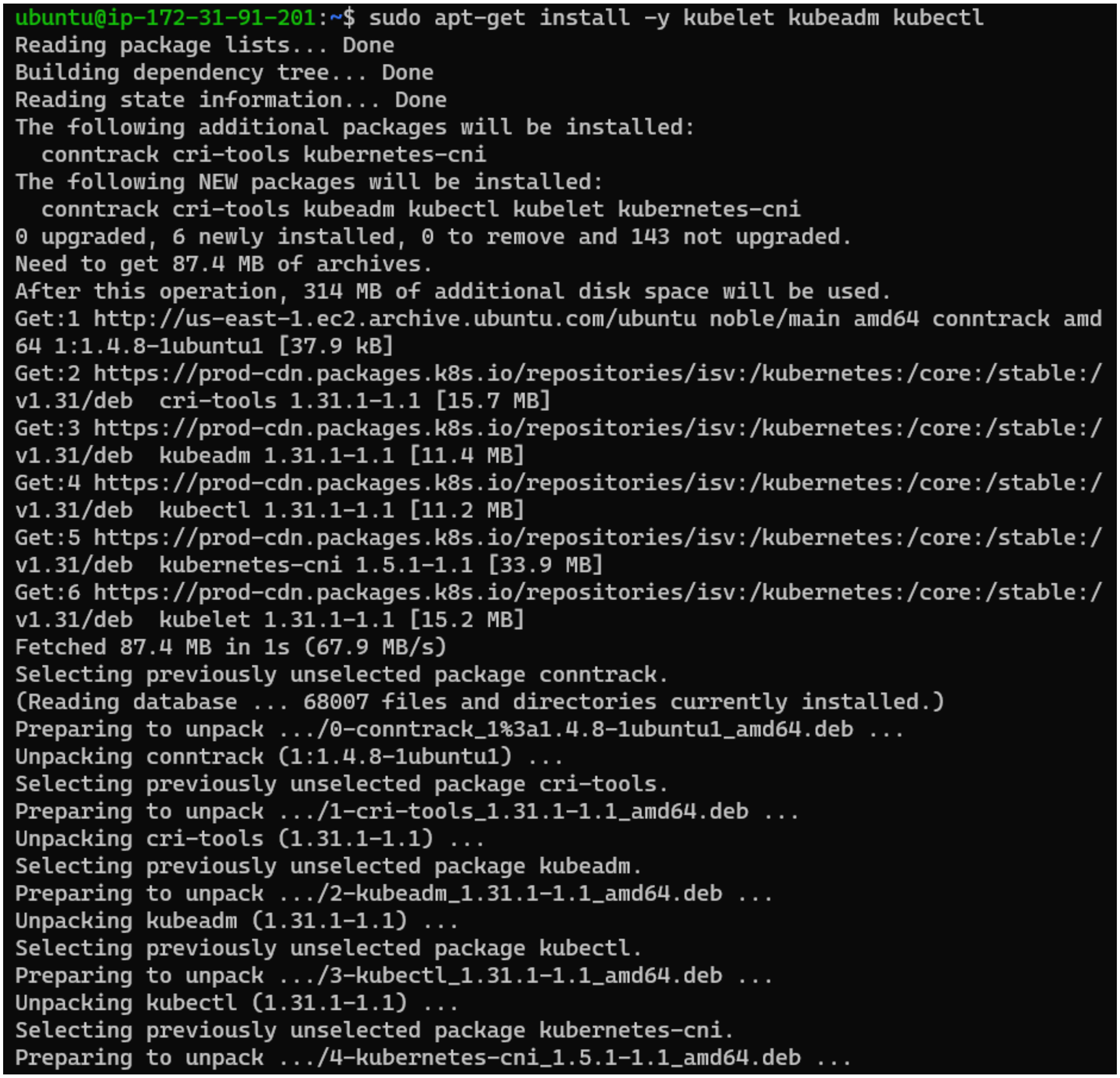
```
ubuntu@ip-172-31-91-201:~$ sudo mkdir -p /etc/apt/keyrings
ubuntu@ip-172-31-91-201:~$ curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.31/deb/R
elease.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg
File '/etc/apt/keyrings/kubernetes-apt-keyring.gpg' exists. Overwrite? (y/N) y
ubuntu@ip-172-31-91-201:~$ echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-ke
yring.gpg] https://pkgs.k8s.io/core:/stable:/v1.31/deb/ /' | sudo tee /etc/apt/sour
ces.list.d/kubernetes.list
deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/co
re:/stable:/v1.31/deb/ /
```

Error solved-

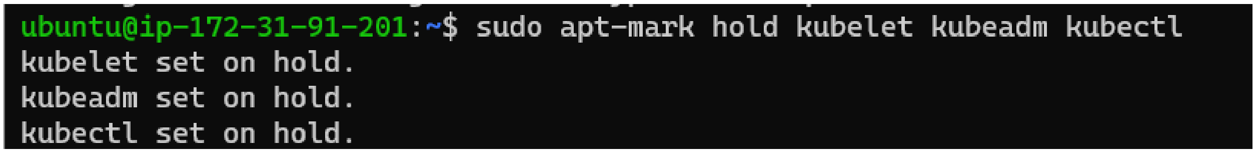


```
ubuntu@ip-172-31-91-201:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InReleaseHit:3 h
ttp://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu noble InRelease
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  InRelease [1186 B]
Hit:6 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:7 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  Packages [4865 B]
Fetched 6051 B in 1s (10.8 kB/s)
Reading package lists... Done
W: https://download.docker.com/linux/ubuntu/dists/noble/InRelease: Key is stored in
 legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in
apt-key(8) for details.
```

Shreya Sawant D15A -54

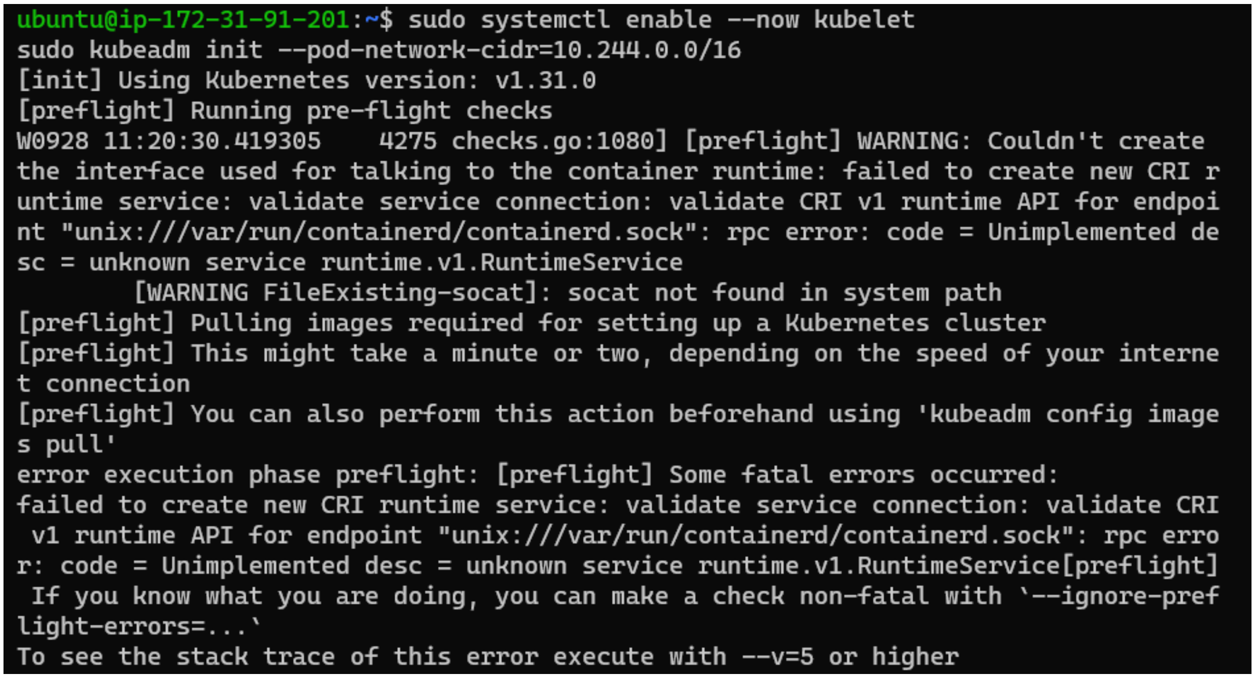


```
ubuntu@ip-172-31-91-201:~$ sudo apt-get install -y kubelet kubeadm kubectl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  conntrack cri-tools kubernetes-cni
The following NEW packages will be installed:
  conntrack cri-tools kubeadm kubectl kubelet kubernetes-cni
0 upgraded, 6 newly installed, 0 to remove and 143 not upgraded.
Need to get 87.4 MB of archives.
After this operation, 314 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 conntrack amd
64 1:1.4.8-1ubuntu1 [37.9 kB]
Get:2 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  cri-tools 1.31.1-1.1 [15.7 MB]
Get:3 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  kubeadm 1.31.1-1.1 [11.4 MB]
Get:4 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  kubectl 1.31.1-1.1 [11.2 MB]
Get:5 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  kubernetes-cni 1.5.1-1.1 [33.9 MB]
Get:6 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/
v1.31/deb  kubelet 1.31.1-1.1 [15.2 MB]
Fetched 87.4 MB in 1s (67.9 MB/s)
Selecting previously unselected package conntrack.
(Reading database ... 68007 files and directories currently installed.)
Preparing to unpack .../0-conntrack_1%3a1.4.8-1ubuntu1_amd64.deb ...
Unpacking conntrack (1:1.4.8-1ubuntu1) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.31.1-1.1_amd64.deb ...
Unpacking cri-tools (1.31.1-1.1) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../2-kubeadm_1.31.1-1.1_amd64.deb ...
Unpacking kubeadm (1.31.1-1.1) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../3-kubectl_1.31.1-1.1_amd64.deb ...
Unpacking kubectl (1.31.1-1.1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../4-kubernetes-cni_1.5.1-1.1_amd64.deb ...
```

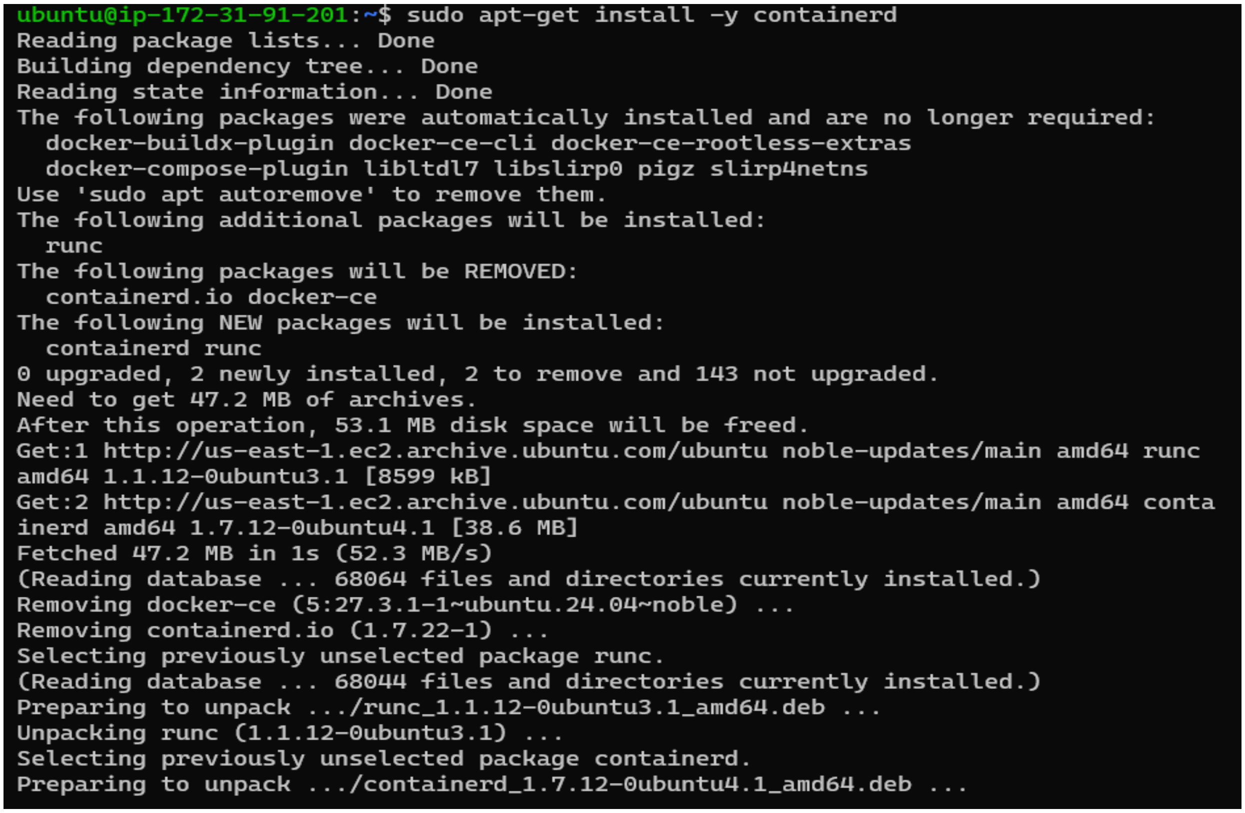


```
ubuntu@ip-172-31-91-201:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
```

Shreya Sawant D15A -54

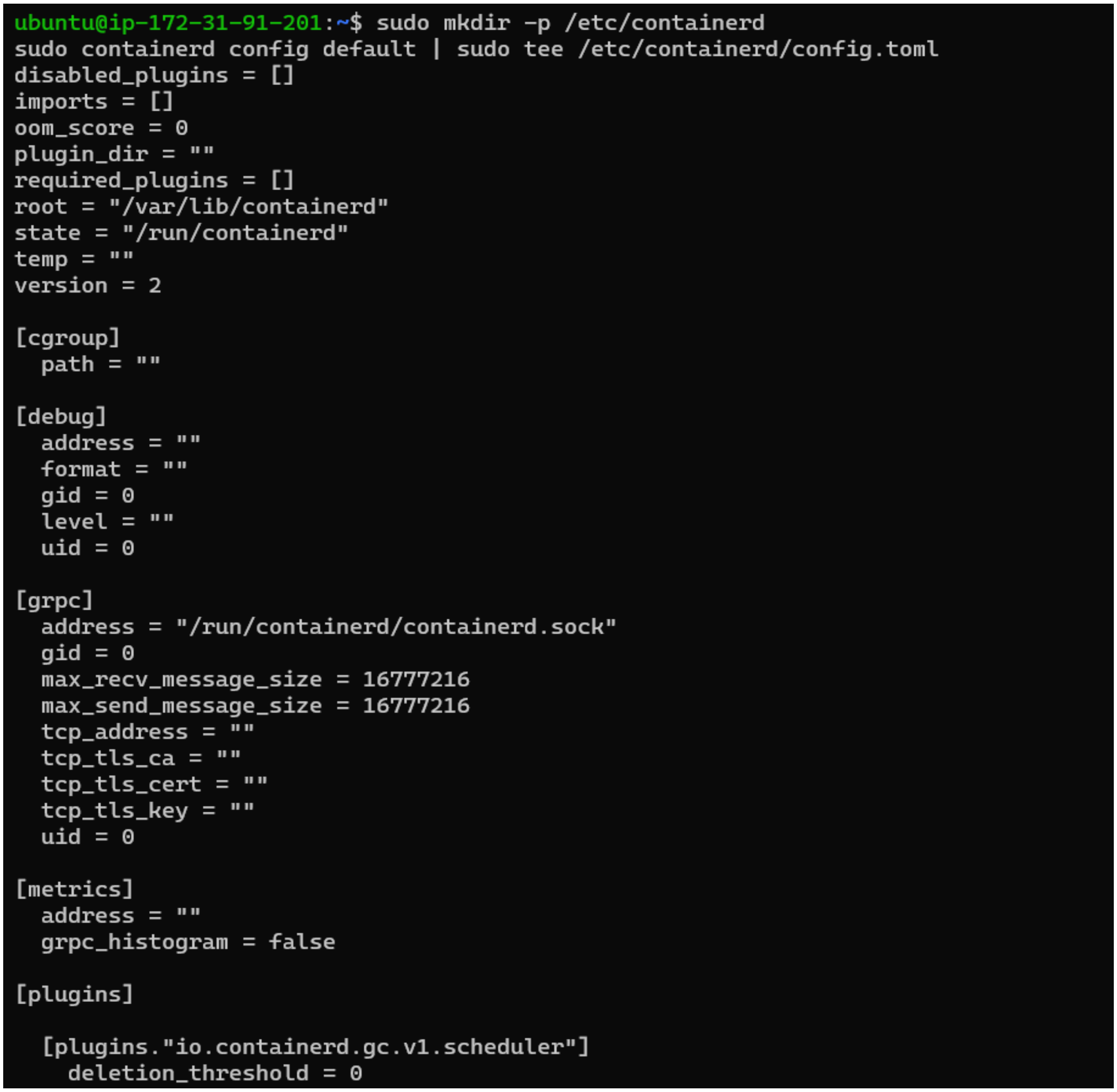


```
ubuntu@ip-172-31-91-201:~$ sudo systemctl enable --now kubelet
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
W0928 11:20:30.419305    4275 checks.go:1080] [preflight] WARNING: Couldn't create
the interface used for talking to the container runtime: failed to create new CRI r
untime service: validate service connection: validate CRI v1 runtime API for endpoi
nt "unix:///var/run/containerd/containerd.sock": rpc error: code = Unimplemented de
sc = unknown service runtime.v1.RuntimeService
        [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 interne
t connection
[preflight] You can also perform this action beforehand using 'kubeadm config image
s pull'
error execution phase preflight: [preflight] Some fatal errors occurred:
failed to create new CRI runtime service: validate service connection: validate CRI
 v1 runtime API for endpoint "unix:///var/run/containerd/containerd.sock": rpc erro
r: code = Unimplemented desc = unknown service runtime.v1.RuntimeService[preflight]
 If you know what you are doing, you can make a check non-fatal with `--ignore-pref
light-errors=...`
To see the stack trace of this error execute with --v=5 or higher
```

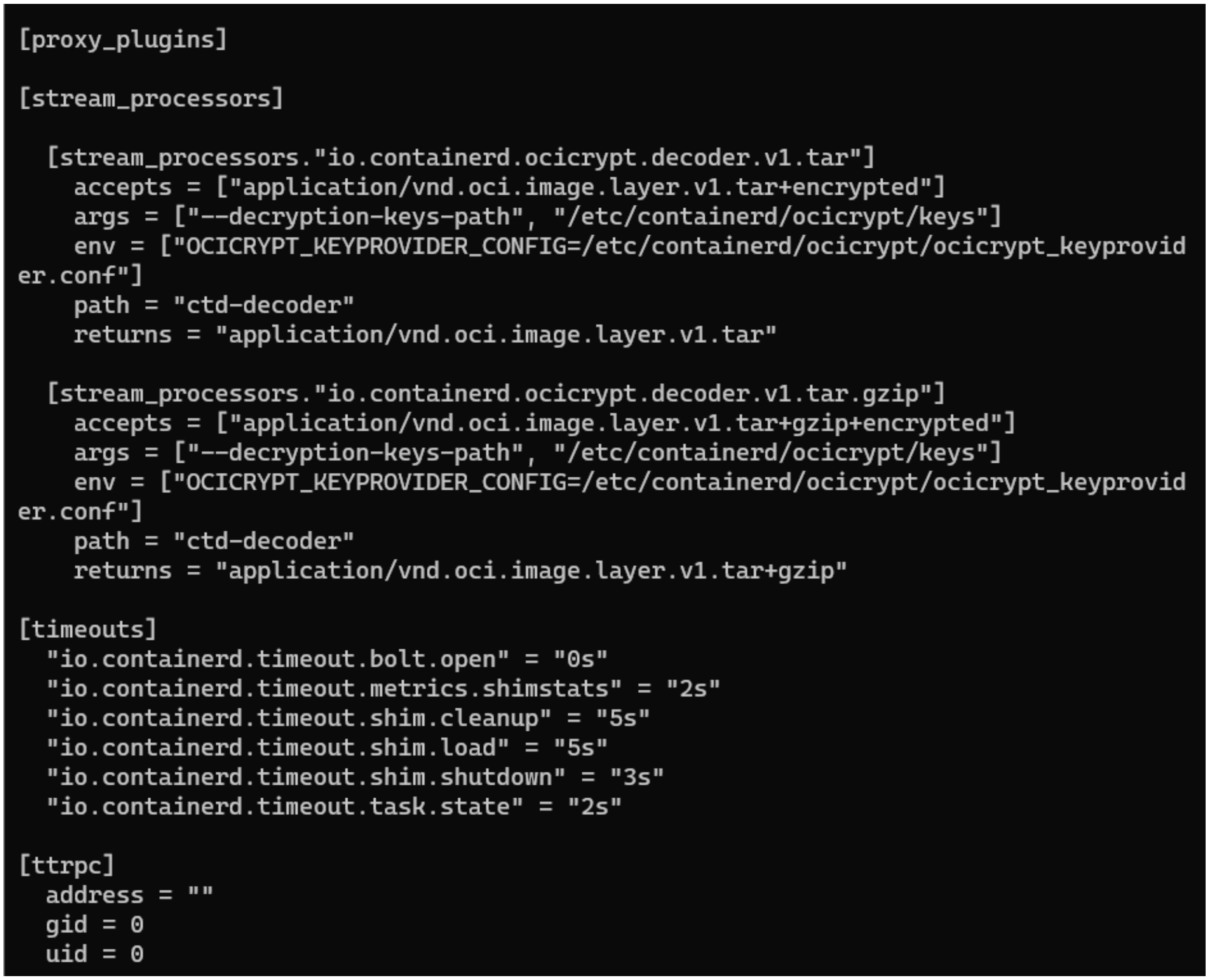


```
ubuntu@ip-172-31-91-201:~$ sudo apt-get install -y containerd
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras
  docker-compose-plugin libltdl7 libslirp0 pigz slirp4netns
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  runc
The following packages will be REMOVED:
  containerd.io docker-ce
The following NEW packages will be installed:
  containerd runc
0 upgraded, 2 newly installed, 2 to remove and 143 not upgraded.
Need to get 47.2 MB of archives.
After this operation, 53.1 MB disk space will be freed.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 runc
amd64 1.1.12-0ubuntu3.1 [8599 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 conta
inerd amd64 1.7.12-0ubuntu4.1 [38.6 MB]
Fetched 47.2 MB in 1s (52.3 MB/s)
(Reading database ... 68064 files and directories currently installed.)
Removing docker-ce (5:27.3.1-1~ubuntu.24.04~noble) ...
Removing containerd.io (1.7.22-1) ...
Selecting previously unselected package runc.
(Reading database ... 68044 files and directories currently installed.)
Preparing to unpack .../runc_1.1.12-0ubuntu3.1_amd64.deb ...
Unpacking runc (1.1.12-0ubuntu3.1) ...
Selecting previously unselected package containerd.
Preparing to unpack .../containerd_1.7.12-0ubuntu4.1_amd64.deb ...
```

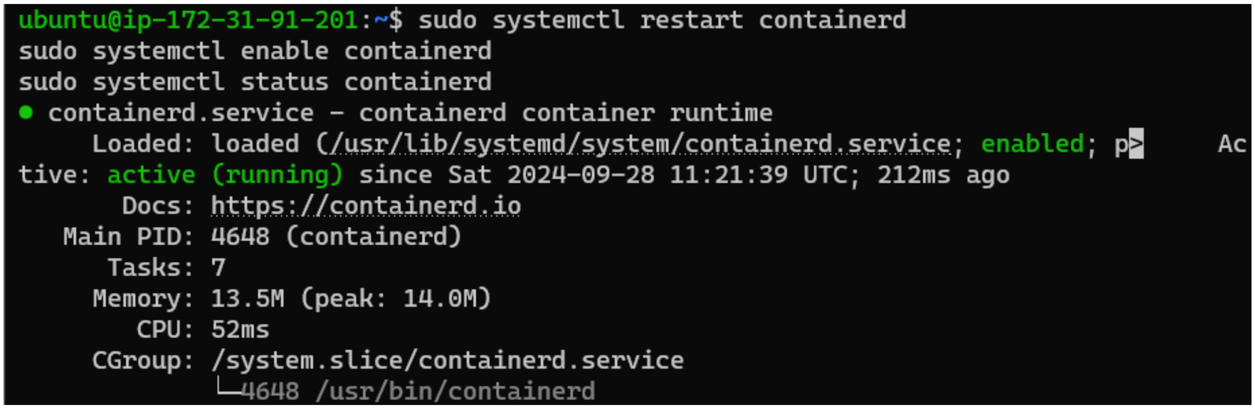
Shreya Sawant D15A -54



```
ubuntu@ip-172-31-91-201:~$ sudo mkdir -p /etc/containerd
sudo containerd config default | sudo tee /etc/containerd/config.toml
disabled_plugins = []
imports = []
oom_score = 0
plugin_dir = ""
required_plugins = []
root = "/var/lib/containerd"
state = "/run/containerd"
temp = ""
version = 2

[cgroup]
  path = ""

[debug]
  address = ""
  format = ""
  gid = 0
  level = ""
  uid = 0

[grpc]
  address = "/run/containerd/containerd.sock"
  gid = 0
  max_recv_message_size = 16777216
  max_send_message_size = 16777216
  tcp_address = ""
  tcp_tls_ca = ""
  tcp_tls_cert = ""
  tcp_tls_key = ""
  uid = 0

[metrics]
  address = ""
  grpc_histogram = false

[plugins]

  [plugins."io.containerd.gc.v1.scheduler"]
    deletion_threshold = 0
```

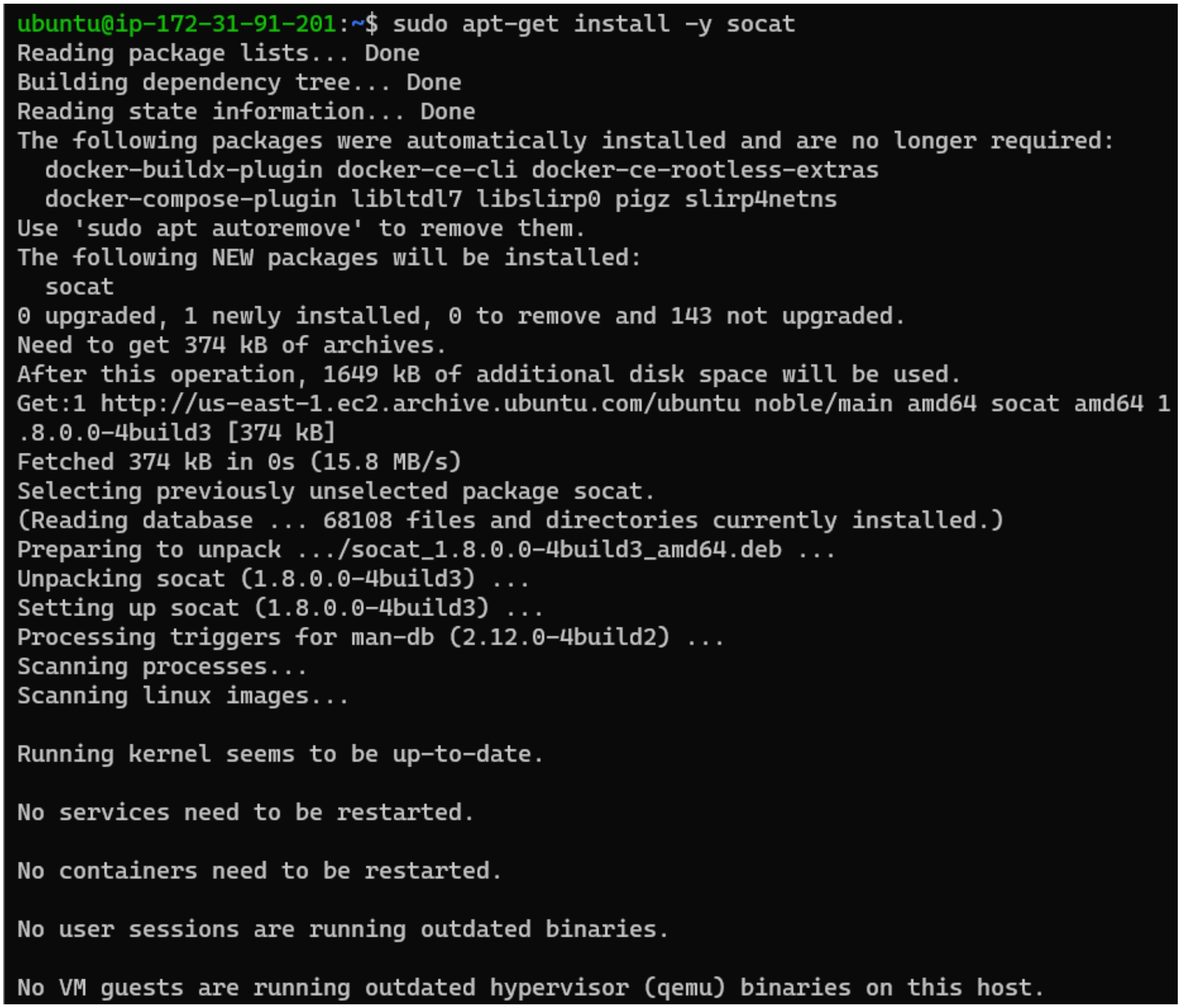


```
[proxy_plugins]

[stream_processors]

  [stream_processors."io.containerd.ocicrypt.decoder.v1.tar"]
    accepts = ["application/vnd.oci.image.layer.v1.tar+encrypted"]
    args = ["--decryption-keys-path", "/etc/containerd/ocicrypt/keys"]
    env = ["OCICRYPT_KEYPROVIDER_CONFIG=/etc/containerd/ocicrypt/ocicrypt_keyprovid
er.conf"]
    path = "ctd-decoder"
    returns = "application/vnd.oci.image.layer.v1.tar"

  [stream_processors."io.containerd.ocicrypt.decoder.v1.tar.gzip"]
    accepts = ["application/vnd.oci.image.layer.v1.tar+gzip+encrypted"]
    args = ["--decryption-keys-path", "/etc/containerd/ocicrypt/keys"]
    env = ["OCICRYPT_KEYPROVIDER_CONFIG=/etc/containerd/ocicrypt/ocicrypt_keyprovid
er.conf"]
    path = "ctd-decoder"
    returns = "application/vnd.oci.image.layer.v1.tar+gzip"

[timeouts]
  "io.containerd.timeout.bolt.open" = "0s"
  "io.containerd.timeout.metrics.shimstats" = "2s"
  "io.containerd.timeout.shim.cleanup" = "5s"
  "io.containerd.timeout.shim.load" = "5s"
  "io.containerd.timeout.shim.shutdown" = "3s"
  "io.containerd.timeout.task.state" = "2s"

[ttrpc]
  address = ""
  gid = 0
  uid = 0
```



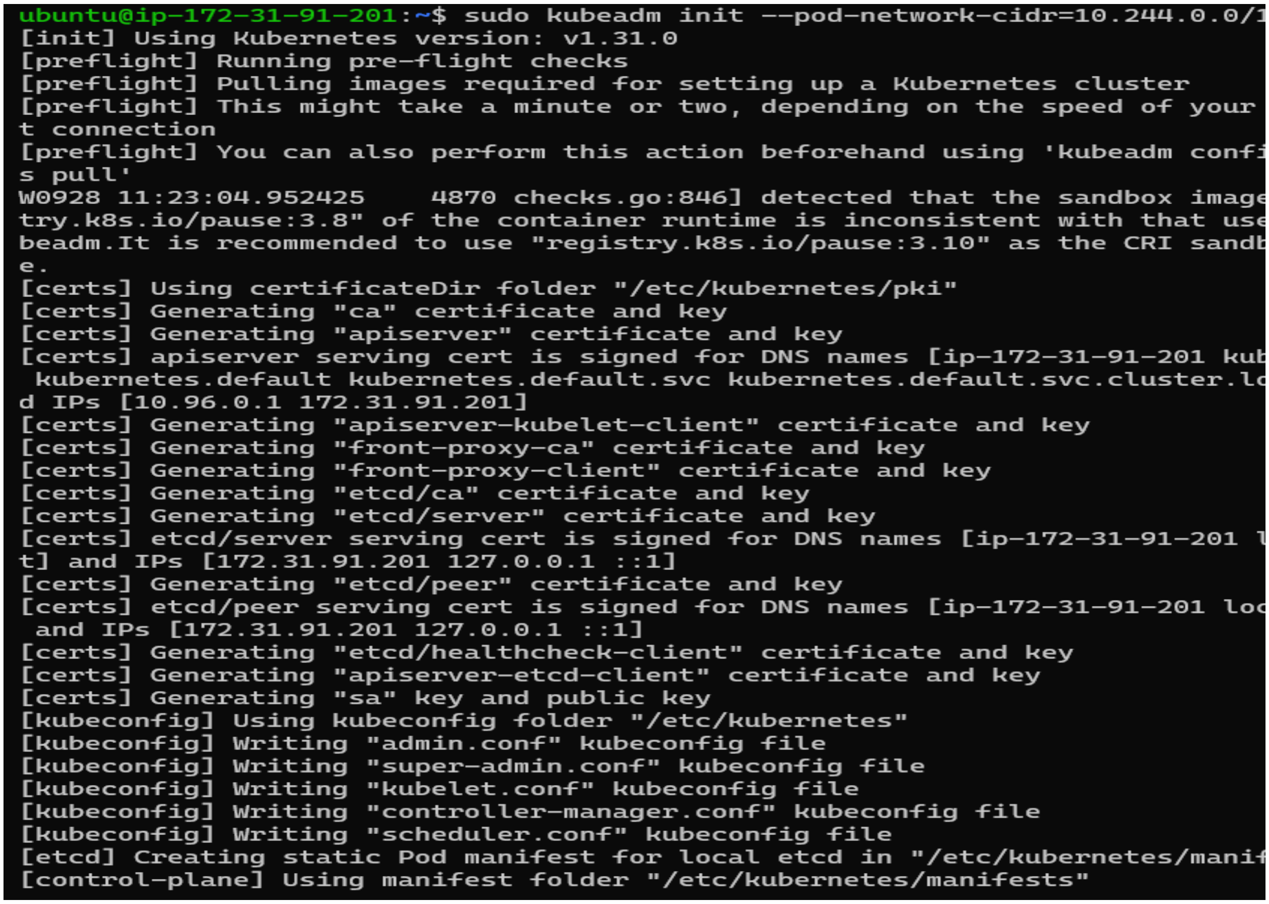
```
ubuntu@ip-172-31-91-201:~$ sudo systemctl restart containerd
sudo systemctl enable containerd
sudo systemctl status containerd
● containerd.service - containerd container runtime
     Loaded: loaded (/usr/lib/systemd/system/containerd.service; enabled; p>      Ac
tive: active (running) since Sat 2024-09-28 11:21:39 UTC; 212ms ago
       Docs: https://containerd.io
   Main PID: 4648 (containerd)
      Tasks: 7
     Memory: 13.5M (peak: 14.0M)
        CPU: 52ms
     CGroup: /system.slice/containerd.service
             └─4648 /usr/bin/containerd
```

Shreya Sawant D15A -54

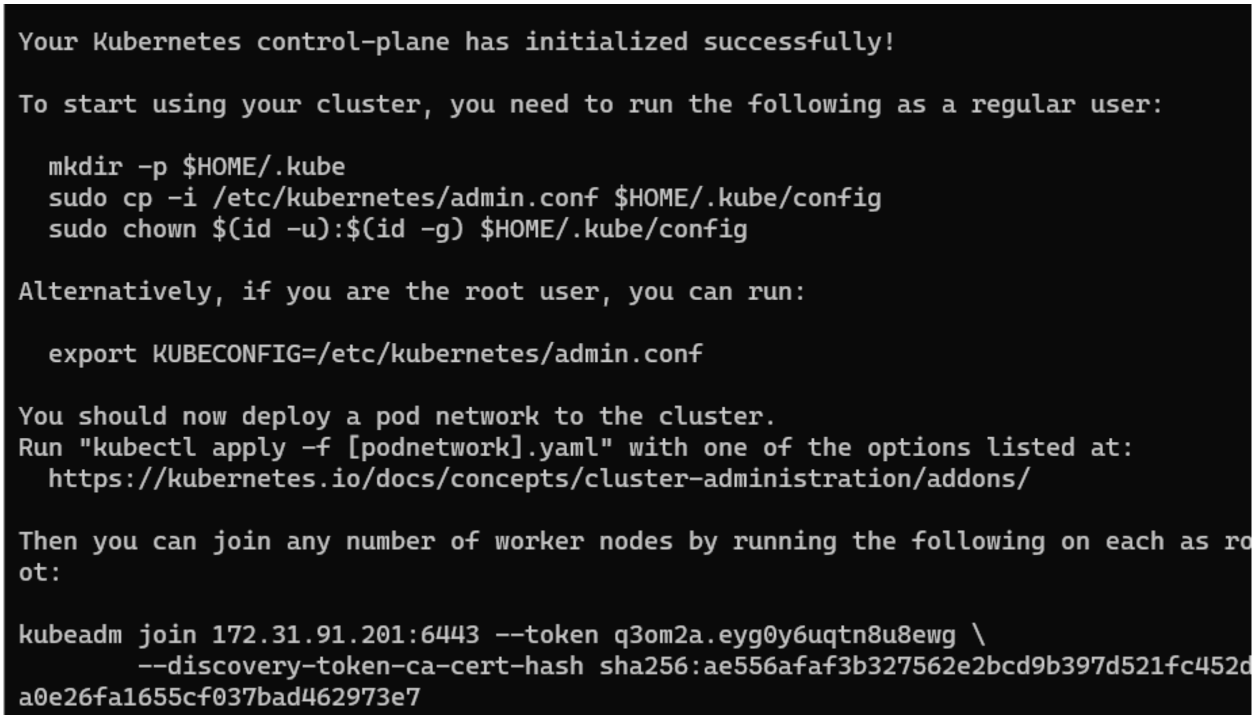


```
ubuntu@ip-172-31-91-201:~$ sudo apt-get install -y socat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras
  docker-compose-plugin libltdl7 libslirp0 pigz slirp4netns
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  socat
0 upgraded, 1 newly installed, 0 to remove and 143 not upgraded.
Need to get 374 kB of archives.
After this operation, 1649 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 socat amd64 1
.8.0.0-4build3 [374 kB]
Fetched 374 kB in 0s (15.8 MB/s)
Selecting previously unselected package socat.
(Reading database ... 68108 files and directories currently installed.)
Preparing to unpack .../socat_1.8.0.0-4build3_amd64.deb ...
Unpacking socat (1.8.0.0-4build3) ...
Setting up socat (1.8.0.0-4build3) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

Shreya Sawant D15A -54

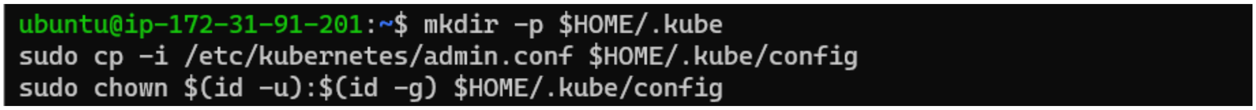


```
ubuntu@ip-172-31-91-201:~$ sudo kubeadm init --pod-network-cidr=10.244.0.0/1
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your
t connection
[preflight] You can also perform this action beforehand using 'kubeadm confi
s pull'
W0928 11:23:04.952425    4870 checks.go:846] detected that the sandbox image
try.k8s.io/pause:3.8" of the container runtime is inconsistent with that use
beadm.It is recommended to use "registry.k8s.io/pause:3.10" as the CRI sandb
e.
[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 [ip-172-31-91-201 kub
 kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.lo
d IPs [10.96.0.1 172.31.91.201]
[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 [ip-172-31-91-201 l
t] and IPs [172.31.91.201 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [ip-172-31-91-201 loc
 and IPs [172.31.91.201 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "super-admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
[kubeconfig] Writing "controller-manager.conf" kubeconfig file
[kubeconfig] Writing "scheduler.conf" kubeconfig file
[etcd] Creating static Pod manifest for local etcd in "/etc/kubernetes/manif
[control-plane] Using manifest folder "/etc/kubernetes/manifests"
```

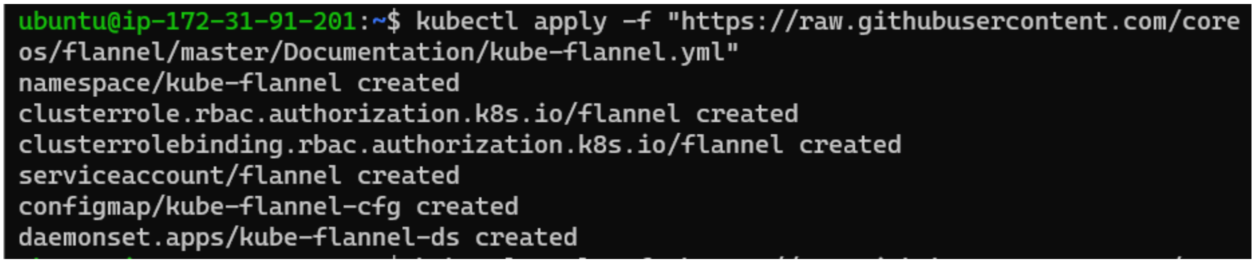


```
Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as ro
ot:

kubeadm join 172.31.91.201:6443 --token q3om2a.eyg0y6uqtn8u8ewg \
        --discovery-token-ca-cert-hash sha256:ae556afaf3b327562e2bcd9b397d521fc452d
a0e26fa1655cf037bad462973e7
```

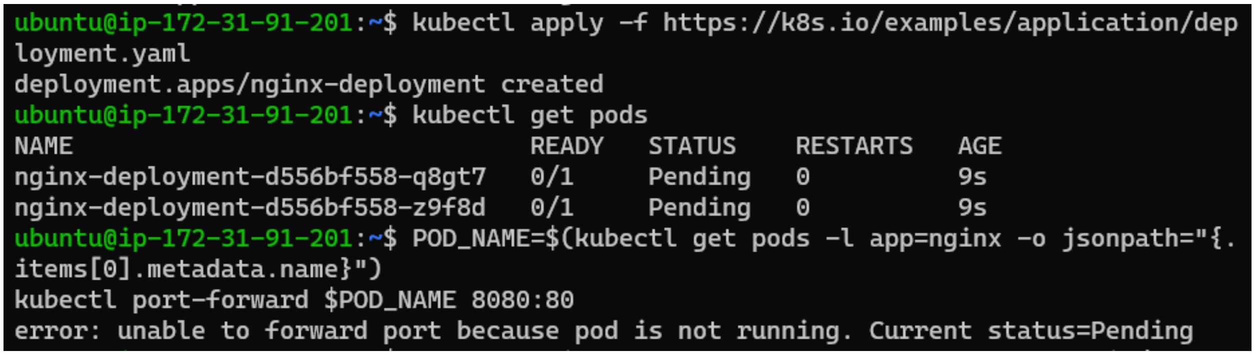
Shreya Sawant D15A -54



```
ubuntu@ip-172-31-91-201:~$ mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

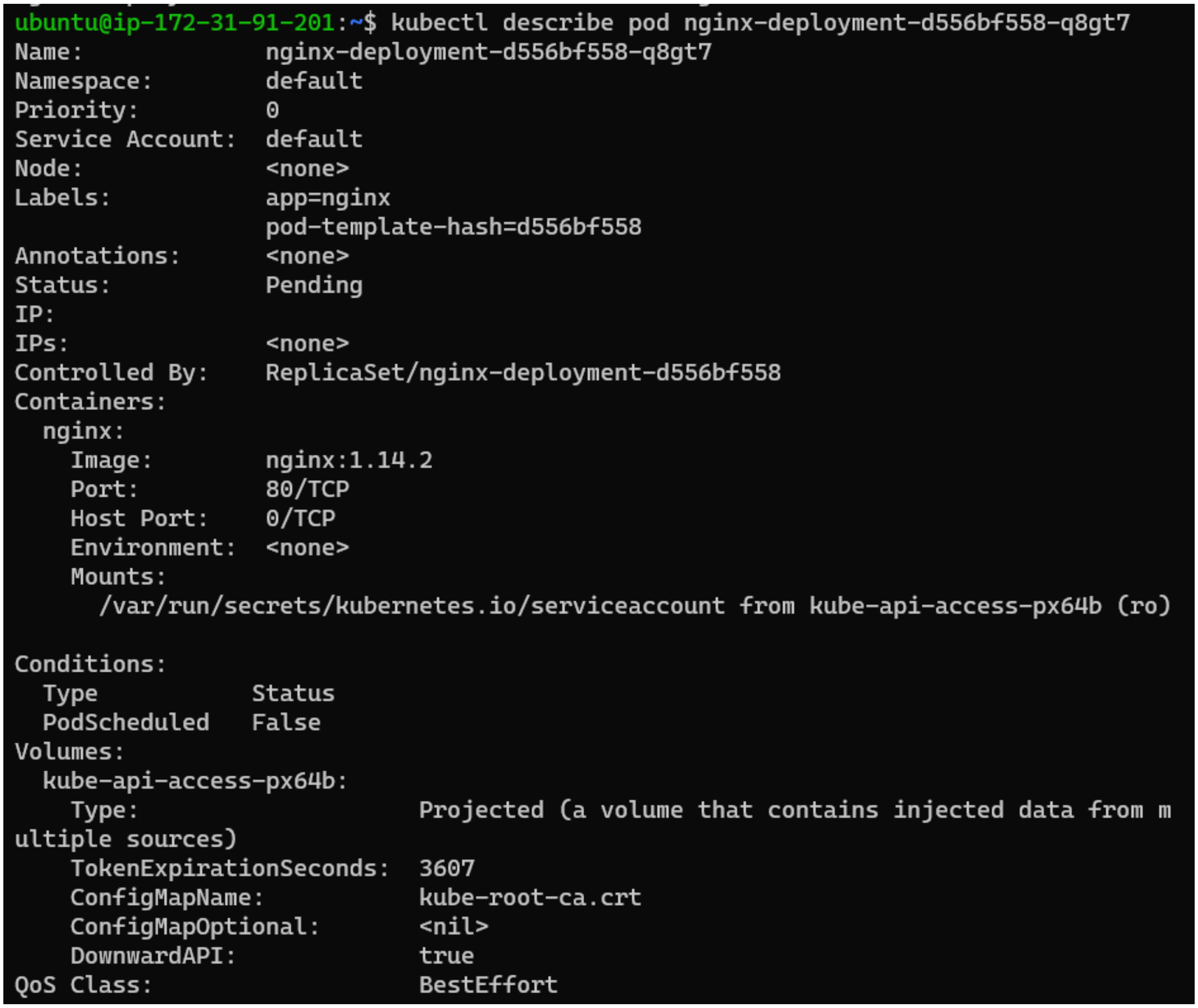


```
ubuntu@ip-172-31-91-201:~$ kubectl apply -f "https://raw.githubusercontent.com/core
os/flannel/master/Documentation/kube-flannel.yml"
namespace/kube-flannel created
clusterrole.rbac.authorization.k8s.io/flannel created
clusterrolebinding.rbac.authorization.k8s.io/flannel created
serviceaccount/flannel created
configmap/kube-flannel-cfg created
daemonset.apps/kube-flannel-ds created
```

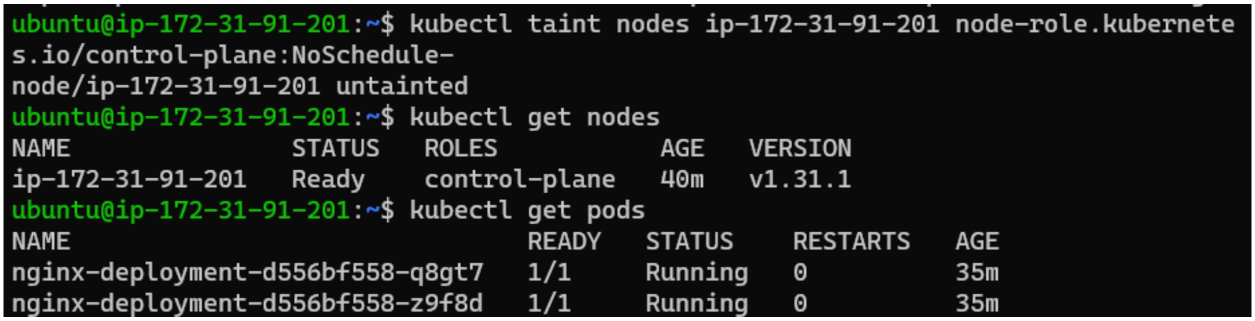


```
ubuntu@ip-172-31-91-201:~$ kubectl apply -f https://k8s.io/examples/application/dep
loyment.yaml
deployment.apps/nginx-deployment created
ubuntu@ip-172-31-91-201:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-d556bf558-q8gt7   0/1     Pending   0          9s
nginx-deployment-d556bf558-z9f8d   0/1     Pending   0          9s
ubuntu@ip-172-31-91-201:~$ POD_NAME=$(kubectl get pods -l app=nginx -o jsonpath="{.
items[0].metadata.name}")
kubectl port-forward $POD_NAME 8080:80
error: unable to forward port because pod is not running. Current status=Pending
```

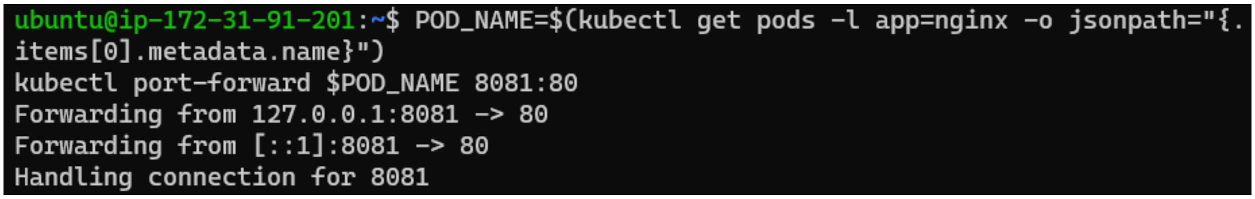
Shreya Sawant D15A -54



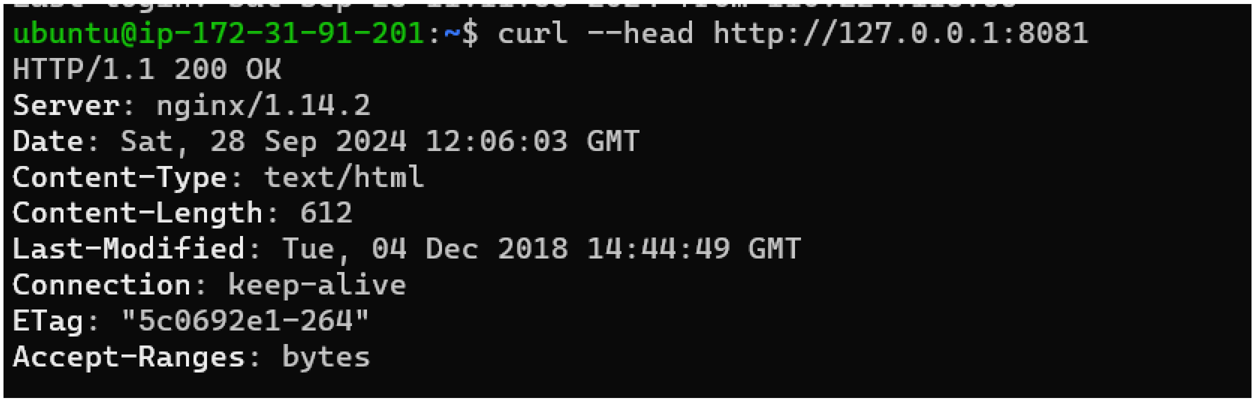
```
ubuntu@ip-172-31-91-201:~$ kubectl describe pod nginx-deployment-d556bf558-q8gt7
Name:             nginx-deployment-d556bf558-q8gt7
Namespace:        default
Priority:         0
Service Account:  default
Node:             <none>
Labels:           app=nginx
                  pod-template-hash=d556bf558
Annotations:      <none>
Status:           Pending
IP:
IPs:              <none>
Controlled By:    ReplicaSet/nginx-deployment-d556bf558
Containers:
  nginx:
    Image:        nginx:1.14.2
    Port:         80/TCP
    Host Port:    0/TCP
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-px64b (ro)

Conditions:
  Type           Status
  PodScheduled   False
Volumes:
  kube-api-access-px64b:
    Type:                    Projected (a volume that contains injected data from m
ultiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
```



```
ubuntu@ip-172-31-91-201:~$ kubectl taint nodes ip-172-31-91-201 node-role.kubernete
s.io/control-plane:NoSchedule-
node/ip-172-31-91-201 untainted
ubuntu@ip-172-31-91-201:~$ kubectl get nodes
NAME               STATUS   ROLES           AGE   VERSION
ip-172-31-91-201   Ready    control-plane   40m   v1.31.1
ubuntu@ip-172-31-91-201:~$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
nginx-deployment-d556bf558-q8gt7   1/1     Running   0          35m
nginx-deployment-d556bf558-z9f8d   1/1     Running   0          35m
```



```
ubuntu@ip-172-31-91-201:~$ POD_NAME=$(kubectl get pods -l app=nginx -o jsonpath="{.
items[0].metadata.name}")
kubectl port-forward $POD_NAME 8081:80
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80
Handling connection for 8081
```



```
ubuntu@ip-172-31-91-201:~$ curl --head http://127.0.0.1:8081
HTTP/1.1 200 OK
Server: nginx/1.14.2
Date: Sat, 28 Sep 2024 12:06:03 GMT
Content-Type: text/html
Content-Length: 612
Last-Modified: Tue, 04 Dec 2018 14:44:49 GMT
Connection: keep-alive
ETag: "5c0692e1-264"
Accept-Ranges: bytes
```