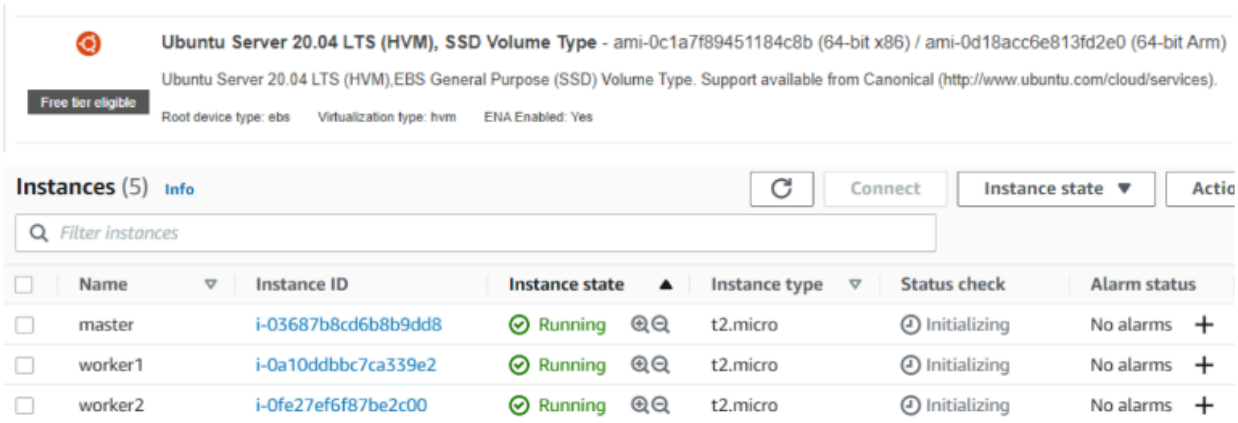
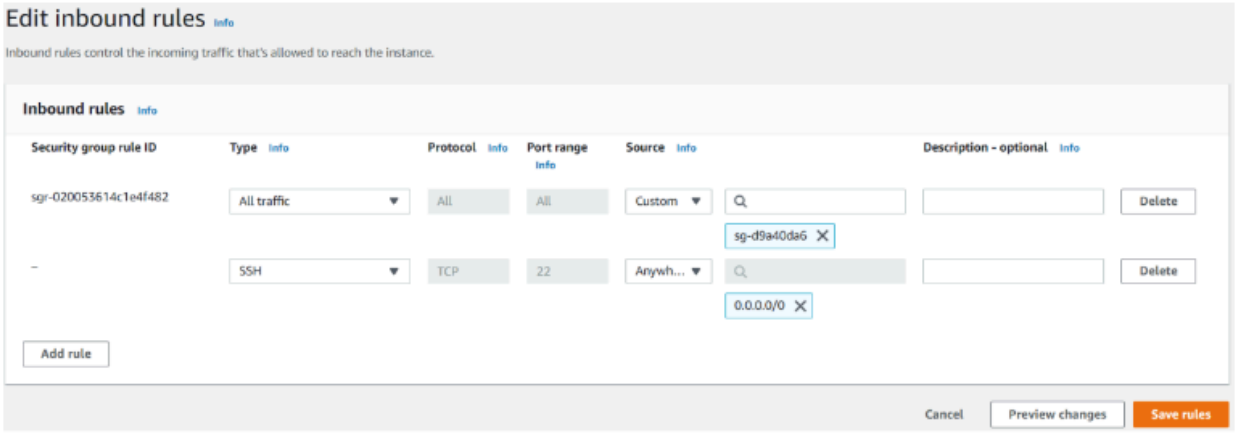
Procedure -

Setting up Kubernetes Cluster

Create 3 EC2 Ubuntu Instances on AWS. Name one as Master, the other two as worker1 and worker2.



Edit the Security Group Inbound Rules to allow SSH



Establish connection with all three machines using SSH, using the following command:

```
ssh -i <keyname>.pem ubuntu@<public_ip_address>
```

```
ubuntu@ip-172-31-10-100: ~
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vkris>cd Downloads

C:\Users\vkris\Downloads>ssh -i keypair111.pem ubuntu@13.233.119.158
The authenticity of host '13.233.119.158 (13.233.119.158)' can't be established.
ECDSA key fingerprint is SHA256:HsaFFde8w3o7UtvyTtU1SQrsJX0/oNW7Yjl13vFNAdI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.233.119.158' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

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

  System information as of Sun Oct  3 14:50:48 UTC 2021

  System load:  0.0               Processes:             100
  Usage of /:   16.4% of 7.69GB   Users logged in:       0
  Memory usage: 22%               IPv4 address for eth0: 172.31.10.100
  Swap usage:   0%
```

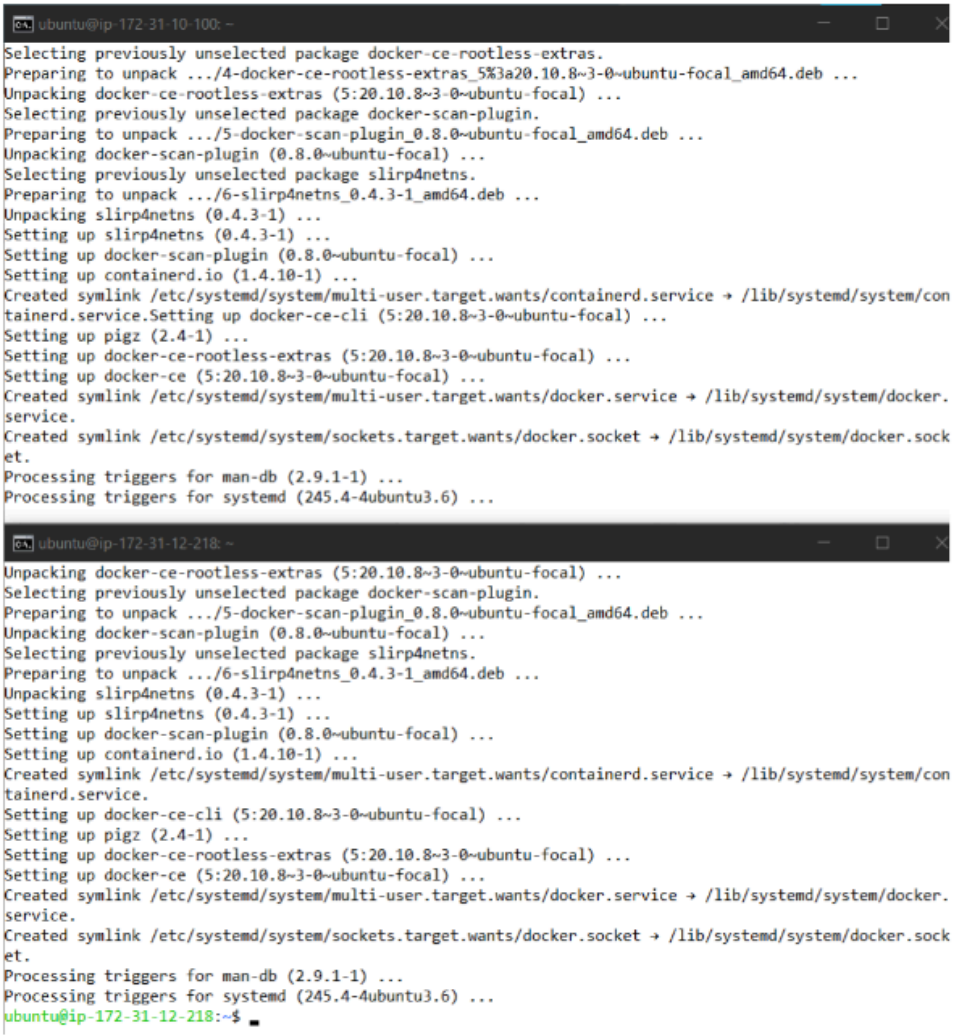
```
ubuntu@ip-172-31-12-218: ~
Microsoft Windows [Version 10.0.19043.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vkris>cd Downloads

C:\Users\vkris\Downloads>ssh -i keypair111.pem ubuntu@3.109.155.147
The authenticity of host '3.109.155.147 (3.109.155.147)' can't be established.
ECDSA key fingerprint is SHA256:ufxZirY0JTnemOTkAag7OIrvGKu2yRrpsm9ttFYHyBY.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.109.155.147' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

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

  System information as of Sun Oct  3 14:51:32 UTC 2021

  System load:  0.56              Processes:             100
  Usage of /:   16.4% of 7.69GB   Users logged in:       0
  Memory usage: 22%               IPv4 address for eth0: 172.31.12.218
  Swap usage:   0%
```

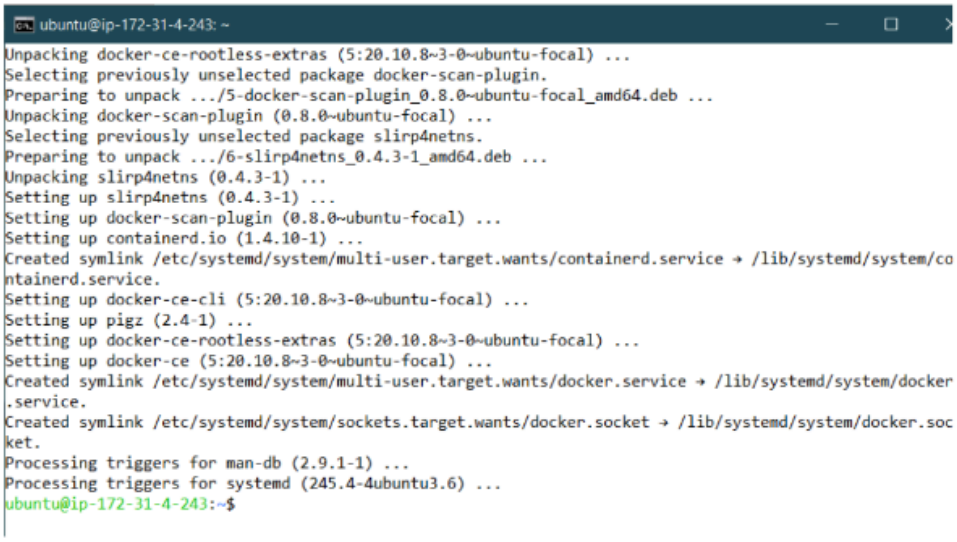
```
ubuntu@ip-172-31-4-243: ~
  Usage of /:   16.4% of 7.69GB   Users logged in:       0
  Memory usage: 22%               IPv4 address for eth0: 172.31.4.243
  Swap usage:   0%

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable


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


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-4-243:~$
```

Perform all these steps at the same time on all 3 machines, unless specified otherwise.

Install Docker:

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
sudo apt-get update
sudo apt-get install -y docker-ce
```





Then, configure cgroup in a daemon.json file.

```
cd /etc/docker
cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
sudo systemctl enable docker
sudo systemctl daemon-reload
sudo systemctl restart docker
```

Install Kubernetes on all 3 machines

```
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key
add -
cat << EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
sudo apt-get update
```

sudo apt-get install -y kubelet kubeadm kubectl

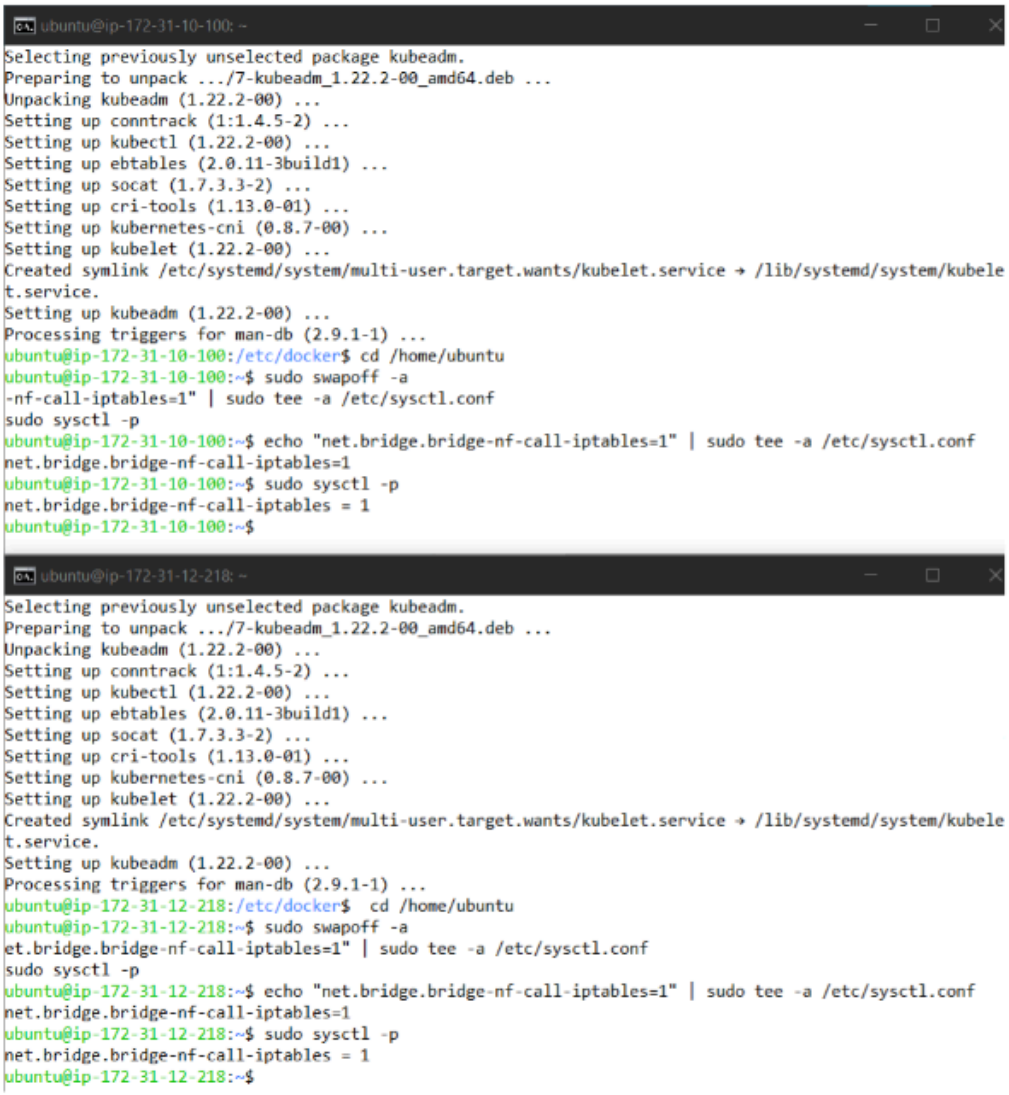
```
ubuntu@ip-172-31-10-100: /etc/docker
Preparing to unpack .../4-socat_1.7.3.3-2_amd64.deb ...
Unpacking socat (1.7.3.3-2) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.22.2-00_amd64.deb ...
Unpacking kubelet (1.22.2-00) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../6-kubectl_1.22.2-00_amd64.deb ...
Unpacking kubectl (1.22.2-00) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../7-kubeadm_1.22.2-00_amd64.deb ...
Unpacking kubeadm (1.22.2-00) ...
Setting up conntrack (1:1.4.5-2) ...
Setting up kubectl (1.22.2-00) ...
Setting up ebtables (2.0.11-3build1) ...
Setting up socat (1.7.3.3-2) ...
Setting up cri-tools (1.13.0-01) ...
Setting up kubernetes-cni (0.8.7-00) ...
Setting up kubelet (1.22.2-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubele
t.service.
Setting up kubeadm (1.22.2-00) ...
Processing triggers for man-db (2.9.1-1) ...
ubuntu@ip-172-31-10-100:/etc/docker$
```

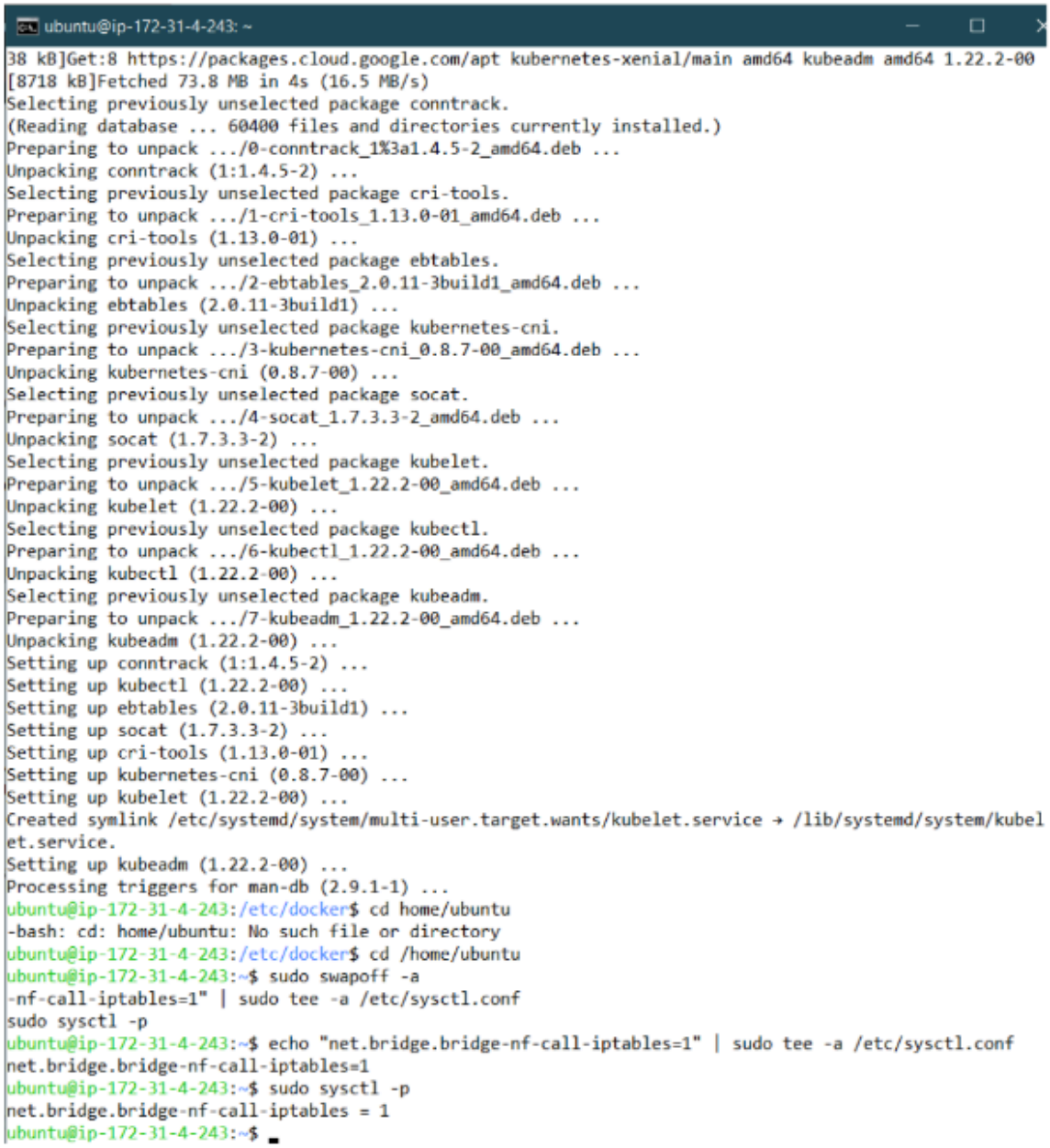
```
ubuntu@ip-172-31-12-218: /etc/docker
Preparing to unpack .../4-socat_1.7.3.3-2_amd64.deb ...
Unpacking socat (1.7.3.3-2) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.22.2-00_amd64.deb ...
Unpacking kubelet (1.22.2-00) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../6-kubectl_1.22.2-00_amd64.deb ...
Unpacking kubectl (1.22.2-00) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../7-kubeadm_1.22.2-00_amd64.deb ...
Unpacking kubeadm (1.22.2-00) ...
Setting up conntrack (1:1.4.5-2) ...
Setting up kubectl (1.22.2-00) ...
Setting up ebtables (2.0.11-3build1) ...
Setting up socat (1.7.3.3-2) ...
Setting up cri-tools (1.13.0-01) ...
Setting up kubernetes-cni (0.8.7-00) ...
Setting up kubelet (1.22.2-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubele
t.service.
Setting up kubeadm (1.22.2-00) ...
Processing triggers for man-db (2.9.1-1) ...
ubuntu@ip-172-31-12-218:/etc/docker$
```

```
ubuntu@ip-172-31-4-243: /etc/docker
MB]Get:7 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubectl amd64 1.22.2-00 [90
38 kB]Get:8 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubeadm amd64 1.22.2-00
[8718 kB]Fetched 73.8 MB in 4s (16.5 MB/s)
Selecting previously unselected package conntrack.
(Reading database ... 60400 files and directories currently installed.)
Preparing to unpack .../0-conntrack_1%3a1.4.5-2_amd64.deb ...
Unpacking conntrack (1:1.4.5-2) ...
Selecting previously unselected package cri-tools.
Preparing to unpack .../1-cri-tools_1.13.0-01_amd64.deb ...
Unpacking cri-tools (1.13.0-01) ...
Selecting previously unselected package ebtables.
Preparing to unpack .../2-ebtables_2.0.11-3build1_amd64.deb ...
Unpacking ebtables (2.0.11-3build1) ...
Selecting previously unselected package kubernetes-cni.
Preparing to unpack .../3-kubernetes-cni_0.8.7-00_amd64.deb ...
Unpacking kubernetes-cni (0.8.7-00) ...
Selecting previously unselected package socat.
Preparing to unpack .../4-socat_1.7.3.3-2_amd64.deb ...
Unpacking socat (1.7.3.3-2) ...
Selecting previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.22.2-00_amd64.deb ...
Unpacking kubelet (1.22.2-00) ...
Selecting previously unselected package kubectl.
Preparing to unpack .../6-kubectl_1.22.2-00_amd64.deb ...
Unpacking kubectl (1.22.2-00) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../7-kubeadm_1.22.2-00_amd64.deb ...
Unpacking kubeadm (1.22.2-00) ...
Setting up conntrack (1:1.4.5-2) ...
Setting up kubectl (1.22.2-00) ...
Setting up ebtables (2.0.11-3build1) ...
Setting up socat (1.7.3.3-2) ...
Setting up cri-tools (1.13.0-01) ...
Setting up kubernetes-cni (0.8.7-00) ...
Setting up kubelet (1.22.2-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubel
et.service.
Setting up kubeadm (1.22.2-00) ...
Processing triggers for man-db (2.9.1-1) ...
ubuntu@ip-172-31-4-243:/etc/docker$
```

After installing Kubernetes, we need to configure internet options to allow bridging.

```
sudo swapoff -a
echo "net.bridge.bridge-nf-call-iptables=1" | sudo tee -a /etc/sysctl
sudo sysctl -p
```





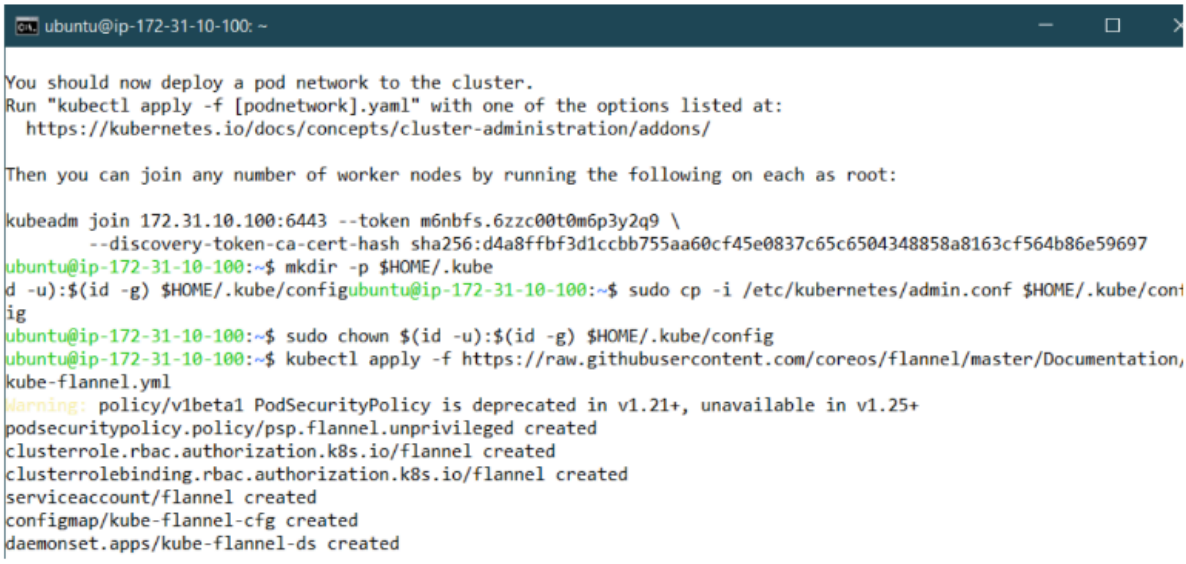
Perform this ONLY on the Master machine:

Initialize the Kubecluster

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

Copy the join command and keep it in a notepad, we'll need it later.
Copy the mkdir and chown commands from the top and execute them.
Then, add a common networking plugin called flamel file as mentioned in the code.

```
kubectl apply -f
https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml
```



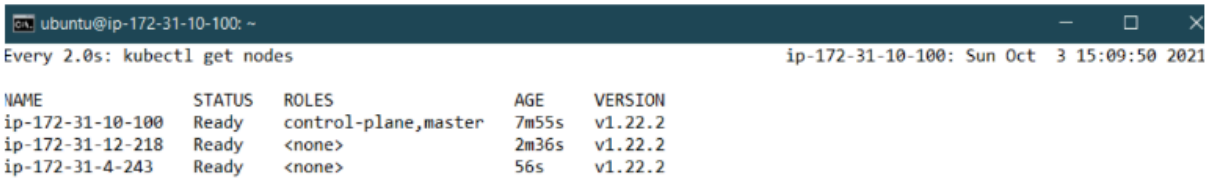
Check the created pod using this command. Now, keep a watch on all nodes using the following command:

```
watch kubectl get nodes
```

Perform this ONLY on the worker machines:

```
sudo kubeadm join <ip> --token <token> \ --discovery-token-ca-cert-hash
<hash>
```

Now, notice the changes on the master terminal



We now have a Kubernetes cluster running across 3 AWS EC2 Instances.
This cluster can be used to further deploy applications and their loads being distributed across these machines.