

How to Set-Up easy kubernetes cluster in AWS EC2(Ubuntu 24.04 LTS)

Run following commands in your server : (Mater node)

```
sudo apt update && sudo apt upgrade -y
```

```
sudo swapoff -a
```

```
sudo sed -i 's/^(\.*/\1/g' /etc/fstab
```

```
sudo tee /etc/modules-load.d/containerd.conf <<EOF
```

```
overlay
```

```
br_netfilter
```

```
EOF
```

```
sudo modprobe overlay
```

```
sudo modprobe br_netfilter
```

```
sudo tee /etc/sysctl.d/kubernetes.conf <<EOF
```

```
net.bridge.bridge-nf-call-ip6tables = 1
```

```
net.bridge.bridge-nf-call-iptables = 1
```

```
net.ipv4.ip_forward = 1
```

```
EOF
```

```
sudo sysctl -system
```

```
sudo apt install -y curl gnupg2 software-properties-common apt-transport-https ca-certificates
```

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o  
/etc/apt/trusted.gpg.d/docker.gpg
```

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -  
cs) stable"
```

```
sudo apt update
```

```
sudo apt install -y containerd.io
```

```
containerd config default | sudo tee /etc/containerd/config.toml >/dev/null 2>&1
```

```
sudo sed -i 's/SystemdCgroup \= false/SystemdCgroup \= true/g' /etc/containerd/config.toml sudo
systemctl restart containerd sudo systemctl enable containerd echo "deb [signed-
by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]
https://pkgs.k8s.io/core:/stable:/v1.30/deb/ /" | sudo tee /etc/apt/sources.list.d/kubernetes.list
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.30/deb/Release.key | sudo gpg --dearmor -o
/etc/apt/keyrings/kubernetes-apt-keyring.gpg
sudo apt update

sudo apt install -y kubelet kubeadm kubectl

sudo apt-mark hold kubelet kubeadm kubectl

sudo kubeadm init
```

Run following commands in your server : (Worker/Slave node)

```
sudo apt update && sudo apt upgrade -y

sudo swapoff -a

sudo sed -i 's/^(\s*\$)/#\1/g' /etc/fstab

sudo tee /etc/modules-load.d/containerd.conf <<EOF

overlay

br_netfilter

EOF

sudo modprobe overlay

sudo modprobe br_netfilter

sudo tee /etc/sysctl.d/kubernetes.conf <<EOF

net.bridge.bridge-nf-call-ip6tables = 1

net.bridge.bridge-nf-call-iptables = 1
```

```
net.ipv4.ip_forward = 1 EOF sudo sysctl --system sudo apt install -y curl gnupg2 software-properties-  
common apt-transport-https ca-certificates sudo curl -fsSL
```

```
https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o  
/etc/apt/trusted.gpg.d/docker.gpg
```

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -  
cs) stable"
```

```
sudo apt update
```

```
sudo apt install -y containerd.io
```

```
containerd config default | sudo tee /etc/containerd/config.toml >/dev/null 2>&1
```

```
sudo sed -i 's/SystemdCgroup \= false/SystemdCgroup \= true/g' /etc/containerd/config.toml
```

```
sudo systemctl restart containerd
```

```
sudo systemctl enable containerd
```

```
echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]  
https://pkgs.k8s.io/core:/stable:/v1.30/deb/ /" | sudo tee /etc/apt/sources.list.d/kubernetes.list  
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.30/deb/Release.key | sudo gpg --dearmor -o  
/etc/apt/keyrings/kubernetes-apt-keyring.gpg
```

```
sudo apt update
```

```
sudo apt install -y kubelet kubeadm kubectl
```

```
sudo apt-mark hold kubelet kubeadm kubectl
```

(switch into root user and enter given commands on terminal as well as token which provided by master node after kubeadm init command.)

If nodes are not ready then apply this command in master node :

```
kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml
```

Required ports : 22, 10250, 6443, 80