**Kubeadm Installation in Ubuntu 20.04**

* **sudo su**
* **apt-get update**
* **apt-get install docker.io -y**
* **service docker restart**
* **curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add -**
* **echo "deb http://apt.kubernetes.io/ kubernetes-xenial main" >/etc/apt/sources.list.d/kubernetes.list**
* **apt-get update**
* **apt install kubeadm=1.20.0-00 kubectl=1.20.0-00 kubelet=1.20.0-00 -y**
* **kubeadm init --pod-network-cidr=192.168.0.0/16**

**RUN THIS AT CLIENT PC :**

**kubeadm join 172.31.32.175:6443 --token hfrmb4.iyoveptiicf2inev \ --discovery-token-ca-cert-hash sha256:333bc1c50a26284466f632443bba81f136a23964bef07136dcbaf8fcdfa962f7**

* **mkdir -p $HOME/.kube**
* **sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config**
* **sudo chown $(id -u):$(id -g) $HOME/.kube/config**
* **kubectl apply -f** [**https://docs.projectcalico.org/manifests/calico.yaml**](https://docs.projectcalico.org/manifests/calico.yaml)
* **kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v0.49.0/deploy/static/provider/baremetal/deploy.yaml**

**Minikube installation in Ubuntu 20.04 aws ec2**

Step 1: Launch t2.medim for Ubuntu 20.04 with default SG

Step 2: Sudo su

Step 3: Apt-get update

Step 4: **Install kubectl :**

1. curl -LO [https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl](https://dl.k8s.io/release/$(curl%20-L%20-s%20https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl)

2. chmod +x ./kubectl

3. mv ./kubectl /usr/local/bin/kubectl

Step 5: apt-get update

Step 6: apt-get install docker.io –y

Step 7: curl -LO <https://storage.googleapis.com/minikube/releases/latest/minikube_latest_amd64.deb>

2.sudo dpkg -i minikube\_latest\_amd64.deb

Step 8: apt-get install conntrack -y

Step 9: git clone <https://github.com/Mirantis/cri-dockerd.git>

**Step 10: Copy ALL Below cmd at once**

wget https://storage.googleapis.com/golang/getgo/installer\_linux

chmod +x ./installer\_linux

./installer\_linux

source ~/.bash\_profile

---------------------------------------

**Step 11:copy all at once**

cd cri-dockerd

go build -o bin/cri-dockerd

mkdir -p /usr/local/bin

install -o root -g root -m 0755 bin/cri-dockerd /usr/local/bin/cri-dockerd

cp -a packaging/systemd/\* /etc/systemd/system

sed -i -e 's,/usr/bin/cri-dockerd,/usr/local/bin/cri-dockerd,' /etc/systemd/system/cri-docker.service

Step 12: systemctl daemon-reload

Step 13: systemctl enable cri-docker.service

Step 14: systemctl enable --now cri-docker.socket

------------------------------

Step 15: VERSION="v1.26.0" # check latest version in /releases page

curl -L https://github.com/kubernetes-sigs/cri-tools/releases/download/$VERSION/crictl-${VERSION}-linux-amd64.tar.gz --output crictl-${VERSION}-linux-amd64.tar.gz

Step 16: sudo tar zxvf crictl-$VERSION-linux-amd64.tar.gz -C /usr/local/bin

Step17: rm -f crictl-$VERSION-linux-amd64.tar.gz

Step 18: sudo sysctl fs.protected\_regular=0

Step 19: minikube start --vm-driver=none

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

kubectl taint nodes --all node-role.kubernetes.io/master-

37 kubectl get nodes

38 kubectl info

39 kubectl --help

40 kubectl create deploy mydeploy --image=nginx:latest

41 kubectl get pods

42 kubectl get deploy

46 kubectl get svc

48 kubectl expose deploy mydeploy --type=NodePort --port=80

49 kubectl create deploy mydeploy1 --image=nginx:latest

50 kubectl expose deploy mydeploy1 --type=NodePort --port=80

51 kubectl get svc