**Install Kubernetes Cluster using kubeadm**

Follow this documentation to set up a Kubernetes cluster on **CentOS 7**.

This documentation guides you in setting up a cluster with one master node and one worker node.

**Assumptions**

| **Role** | **FQDN** | **IP** | **OS** | **RAM** | **CPU** |
| --- | --- | --- | --- | --- | --- |
| Master | kmaster.example.com | 172.16.16.100 | CentOS 7 | 2G | 2 |
| Worker | kworker.example.com | 172.16.16.101 | CentOS 7 | 1G | 1 |

**On both Kmaster and Kworker**

Perform all the commands as root user unless otherwise specified

**Disable Firewall**

systemctl disable firewalld

systemctl stop firewalld

**Disable swap**

swapoff -a; sed -i '/swap/d' /etc/fstab

**Disable SELinux**

setenforce 0

sed -i --follow-symlinks 's/^SELINUX=enforcing/SELINUX=disabled/' /etc/sysconfig/selinux

**Update sysctl settings for Kubernetes networking**

cat >>/etc/sysctl.d/kubernetes.conf<<EOF

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

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

EOF

sysctl --system

**Install docker engine**

yum install -y yum-utils device-mapper-persistent-data lvm2

yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

yum install -y docker-ce-19.03.12

systemctl enable --now docker

**Kubernetes Setup**

**Add yum repository**

cat >>/etc/yum.repos.d/kubernetes.repo<<EOF

[kubernetes]

name=Kubernetes

baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86\_64

enabled=1

gpgcheck=1

repo\_gpgcheck=1

gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg

https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg

EOF

**Install Kubernetes components**

yum install -y kubeadm-1.18.5-0 kubelet-1.18.5-0 kubectl-1.18.5-0

**Enable and Start kubelet service**

systemctl enable --now kubelet

**On kmaster**

**Initialize Kubernetes Cluster**

kubeadm init --apiserver-advertise-address=172.16.16.100 --pod-network-cidr=192.168.0.0/16

**Deploy Calico network**

kubectl --kubeconfig=/etc/kubernetes/admin.conf create -f https://docs.projectcalico.org/v3.14/manifests/calico.yaml

**Cluster join command**

kubeadm token create --print-join-command

**To be able to run kubectl commands as non-root user**

If you want to be able to run kubectl commands as non-root user, then as a non-root user perform these

mkdir -p $HOME/.kube

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

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

**On Kworker**

**Join the cluster**

Use the output from **kubeadm token create** command in previous step from the master server and run here.

**Verifying the cluster**

**Get Nodes status**

kubectl get nodes

**Get component status**

kubectl get cs

Have Fun!!