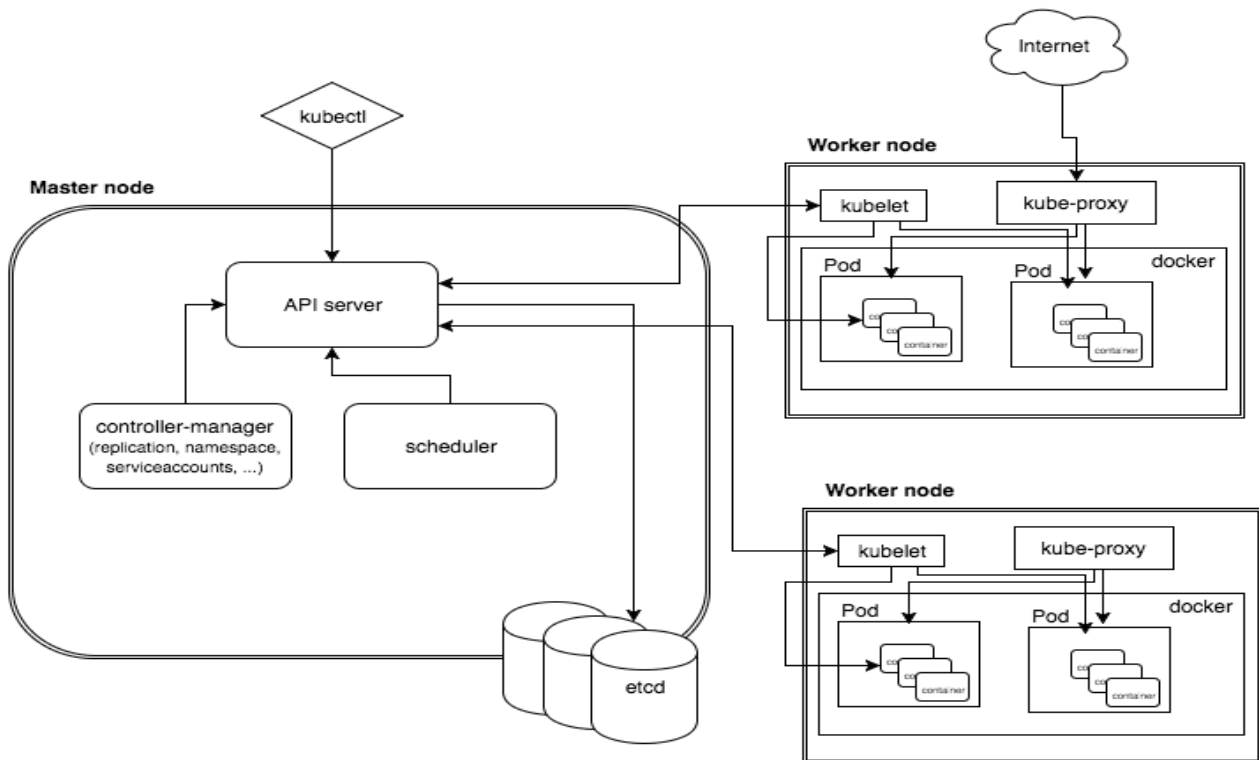


## Kubernetes System Deploy



Diagram

### Tasks to do in every node:

#### 1. Turn off swap memory

`vi /etc/fstab`

```
#/dev/mapper/centos-swap swap          swap    defaults    0 0
```

<use hash for count this line is comment line>

`swapoff -a`

`free -m`

#### 2. Turn off firewall

`systemctl stop firewalld && systemctl disable firewalld`

`systemctl status firewalld`

### 3. Update hosts

```
cat <<EOF >> /etc/hosts
```

```
192.168.1.190 master
```

```
192.168.1.191 worker
```

```
EOF
```

### 4. Load the br\_netfilter module

```
modprobe br_netfilter
```

### 5. Turn off SELINUX

```
vi /etc/selinux/config
```

```
SELINUX=disabled
```

```
reboot
```

```
getenforce
```

### 4. Remove existing version of docker (if exists)

```
sudo yum remove docker \
```

```
docker-client \
```

```
docker-client-latest \
```

```
docker-common \
```

```
docker-latest \
```

```
docker-latest-logrotate \
```

```
docker-logrotate \
```

```
docker-engine -y
```

```
yum -y remove podman
```

```
yum -y remove containers-common
```

### 5. Install docker on linux machine.

```
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 update -y && yum install -y \  
containerd.io \  
docker-ce \  
docker-ce-cli
```

```
mkdir /etc/docker
```

```
cat > /etc/docker/daemon.json <<EOF
```

```
{  
  "exec-opts": ["native.cgroupdriver=systemd"],  
  "log-driver": "json-file",  
  "log-opts": {  
    "max-size": "100m"  
  },  
  "storage-driver": "overlay2",  
  "storage-opts": [  
    "overlay2.override_kernel_check=true"  
  ]  
}  
EOF
```

```
mkdir -p /etc/systemd/system/docker.service.d
```

```
systemctl daemon-reload
```

```
systemctl restart docker && systemctl enable docker
```

## 6. Install kubernetes on linux machine.

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

```
sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/' /etc/selinux/config
```

```
yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
```

```
*kubeadm config images pull*
```

---

For specific version

```
yum install kubelet-1.19* kubeadm-1.19* kubectl-1.19*
```

---

```
systemctl enable --now kubelet
```

```
cat <<EOF > /etc/sysctl.d/k8s.conf  
  
net.bridge.bridge-nf-call-ip6tables = 1  
  
net.bridge.bridge-nf-call-iptables = 1  
  
EOF
```

```
sysctl --system
```

**Task to do in only Master node**

### 1. Initiate cluster:

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

Copy token (Example below)

```
Example: kubeadm join 192.168.10.123:6443 --token d612m1.9l18chn3o19iigmf \
--discovery-token-ca-cert-hash
sha256:c6c2a354670ef9d0c7145a497f8d67892d0c4965304d2264b81e705c2f746f11
```

```
mkdir -p $HOME/.kube
```

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

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

or

```
export KUBECONFIG=/etc/kubernetes/admin.conf
```

### 2. Configure canal:

```
# curl https://docs.projectcalico.org/manifests/canal.yaml -O
```

```
# kubectl apply -f canal.yaml
```

### 3. Untainted master:

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

#### Task to do in Worker node

worker node connect with cluster using token.

```
Example: kubeadm join 192.168.10.123:6443 --token d612m1.9l18chn3o19iigmf \
--discovery-token-ca-cert-hash
sha256:c6c2a354670ef9d0c7145a497f8d67892d0c4965304d2264b81e705c2f746f11
```

#### Task to do in only master node

### Install metrics Server:

```
# wget -O v0.3.6.tar.gz https://codeload.github.com/kubernetes-sigs/metrics-server/tar.gz/v0.3.6
```

```
# tar -xzf v0.3.6.tar.gz
```

New metrics server

=====

```
# kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml
```

>CHANGE:

Metrics server deployment:

Add below after image block...

args:

- '--cert-dir=/tmp'
- '--secure-port=4443'
- '--kubelet-preferred-address-types=InternalIP,ExternalIP,Hostname'
- '--kubelet-use-node-status-port'
- '--metric-resolution=15s'
- '--kubelet-insecure-tls'

Now system is ready to deploy