**KUBERNETES**

**1. Configure VM's hostname**

**k8s-master 192.168.239.135**

**k8s-worker-node-1 192.168.239.134**

**k8s-worker-node-2 192.168.239.136**

**exec bash**

**2. Disable swap on /etc/fstab + swapoff -a**

**3.** **Configure Firewall Rule:**

**firewall-cmd --permanent --add-port=6443/tcp**

**firewall-cmd --permanent --add-port=2379-2380/tcp**

**firewall-cmd --permanent --add-port=10250/tcp**

**firewall-cmd --permanent --add-port=10251/tcp**

**firewall-cmd --permanent --add-port=10252/tcp**

**firewall-cmd --permanent --add-port=10255/tcp**

**firewall-cmd --reload**

**modprobe br\_netfilter – update iptable**

**4. Configure Repository**

**yum-config-manager --add-repo** [**https://download.docker.com/linux/centos/docker-ce.repo**](https://download.docker.com/linux/centos/docker-ce.repo)

**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**

**5. install kubernetes + docker**

**yum install -y kubelet kubectl kubeadm docker**

**systemctl restart docker && systemctl enable docker**

**systemctl restart kubelet && systemctl enable kubelet**

**Server Part**

**1. echo '1' > /proc/sys/net/bridge/bridge-nf-call-iptables**

**modprobe br\_netfilter**

**setenforce 0**

**kubeadm init --apiserver-advertise-address=192.168.239.200**

**copy kubeadm join line to notepad**

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

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

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

**2.kubectl get nodes**

**kubectl get pods --all-namespaces**

**\*install pod network (weave):**

**kubectl apply -f** [**https://cloud.weave.works/k8s/net?k8s-version=$(kubectl version | base64 | tr -d '\n')**](https://cloud.weave.works/k8s/net?k8s-version=$(kubectl%20version%20|%20base64%20|%20tr%20-d%20'\n'))

**\*taint master node to schedule:**

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

**3.Create Dashboard**

**kubectl create -f** [**https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/recommended/kubernetes-dashboard.yaml**](https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/recommended/kubernetes-dashboard.yaml)

**kubectl create serviceaccount dashboard -n default**

**kubectl create clusterrolebinding dashboard-admin -n default --clusterrole=cluster-admin --serviceaccount=default:dashboard**

**kubectl get secret $(kubectl get serviceaccount dashboard -o jsonpath="{.secrets[0].name}") -o jsonpath="{.data.token}" | base64 –decode**

**copy the code to notepad**

**Worker Part**

**echo '1' > /proc/sys/net/bridge/bridge-nf-call-iptables**

**modprobe br\_netfilter**

**past the kubeadm join line that you copy before**

**kubectl get nodes**

**Server Part**

**kubectl proxy**

[**http://192.168.239.200:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/**](http://192.168.239.200:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/)

**past the code you copy to token option**

<https://www.youtube.com/watch?v=MgXh2HpNBtk>

**docker ps -a**

**docker start/stop "CName"**

**docker image**

**docker rm –f $(docker ps -aq)**

**boot2docker ip**

**boot2docker up**

**boot2docker socket**

**docker pull "name(centos)"**

**docker run -it "name(centos)"**

**docker attach "name(centos)"**

**docker exec -it "name(centos)" "/sbin/command"**

**docker logs "name(centos)"**

**sysctl -w net.ipv4.ip\_forward=1**