nano /home/vietdung

sudo apt-get update -y

sudo apt-get install docker-ce docker-ce-cli containerd.io -y

apt-cache madison docker-ce

sudo docker run hello-world

sudo dpkg -i /path/to/package.deb

sudo docker run hello-world

curl -fsSL https://get.docker.com -o get-docker.sh

DRY\_RUN=1 sh ./get-docker.sh

curl -fsSL https://get.docker.com -o get-docker.sh

sudo sh get-docker.sh

curl -fsSL https://test.docker.com -o test-docker.sh

sudo sh test-docker.sh

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-key add -

sudo add-apt-repository \

"deb [arch=amd64] https://download.docker.com/linux/ubuntu \

$(lsb\_release -cs) \

stable"

sudo apt-get update && apt-get install docker-ce -y

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

{

"exec-opts": ["native.cgroupdriver=systemd"],

"log-driver": "json-file",

"log-opts": {

"max-size": "100m"

},

"storage-driver": "overlay2"

}

EOF

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

systemctl daemon-reload

systemctl restart docker

sudo apt-get update && sudo apt-get install -y apt-transport-https curl

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add

chmod 777 /etc/apt/sources.list.d/

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -

echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add

cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list

deb https://apt.kubernetes.io/ kubernetes-xenial main

EOF

sudo apt-get update -y

sudo apt-get install -y kubelet kubeadm kubectl

sudo apt-mark hold kubelet kubeadm kubectl

* Lưu ý, muốn cài 1 version riêng

sudo apt-get update

sudo apt-get install -y kubelet=1.21.0-00 kubeadm=1.21.0-00 kubectl=1.21.0-00

sudo apt-mark hold kubelet kubeadm kubectl

* muốn gỡ version

kubeadm reset

apt remove kubeadm kubectl kubelet

rm -rf /root/.kube

swapoff -a

kubeadm reset

chmod +x /home/vietdung

cd /home

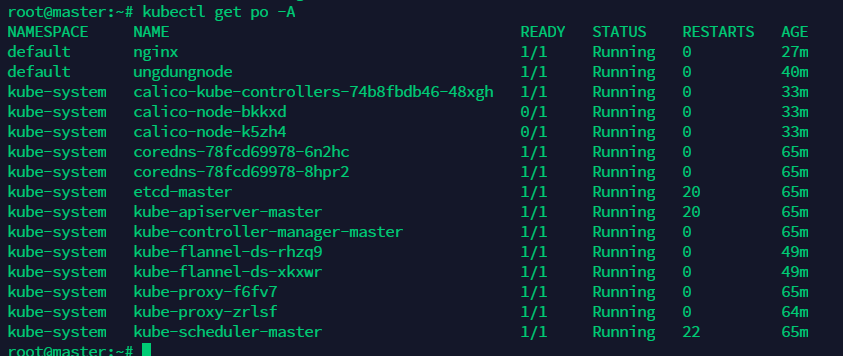
./vietdung

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

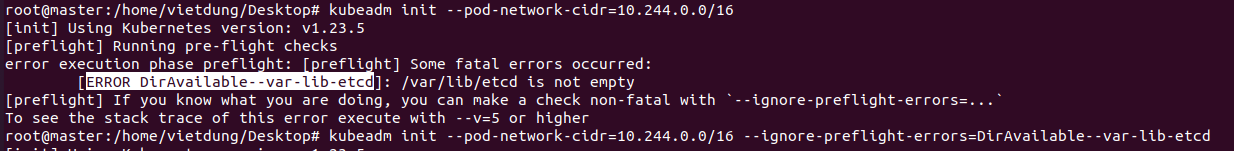
kubectl apply -f <https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml>

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

kubectl taint no master node-role.kubernetes.io/master-



Nếu khi khởi tạo báo lỗi này



Oke chỉ cần thêm câu lệnh ignore error kia

# kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=DirAvailable--var-lib-etcd

# Cài cho user

sudo rm -rf /home/vietdung/.kube

phải là folder của user

sudo swapoff -a

sudo kubeadm reset

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

kubectl apply -f <https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml>

curl https://docs.projectcalico.org/manifests/calico.yaml -O

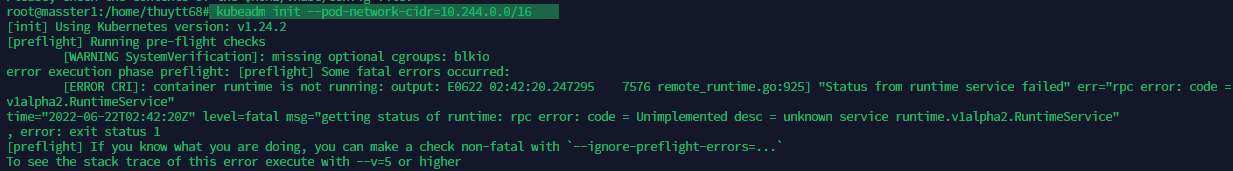
kubectl apply -f calico.yaml

kubectl taint no master node-role.kubernetes.io/master-

# echo "net.bridge.bridge-nf-call-iptables=1" | sudo tee -a /etc/sysctl.conf

sudo sysctl -p

# join node



Error

apt-get update && apt-get install -y apt-transport-https ca-certificates curl software-properties-common

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-key add -

add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"

apt-get update && apt-get install -y containerd.io

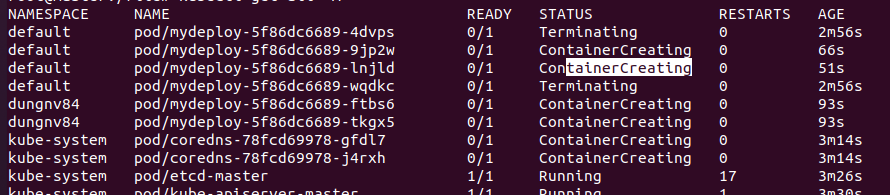
# Configure containerd

mkdir -p /etc/containerd

containerd config default > /etc/containerd/config.toml

systemctl restart containerd

Lưu ý: khi chạy deployment mà sinh ra pod, khi xóa pod quản lý bởi deployment đó thì deployment đó sẽ tự sinh ra các pod khác, chỉ khi xóa deployment đi thì pod mới bị xóa



Xóa xog n lại tự creating lại

kubectl exec -it pods/ubuntu-sleeper -- /bin/bash

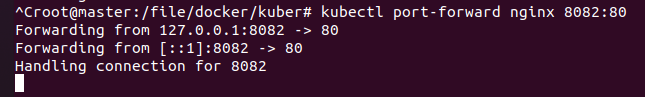
exec vào container

docker exec -it docker-vue\_app\_1 sh

**Test connection**

# kubectl port-forward nginx 8082:80

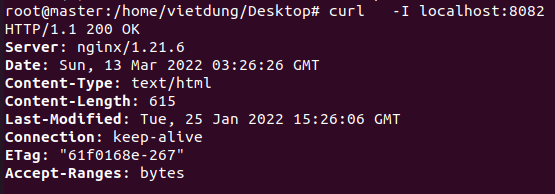
Nginx là pod, 8082 là tùy chọn, port trên local



Mở terminal khác

# curl -H "Accept-Encoding: gzip" -I localhost:8082

# curl -I localhost:8080



# Cài kubectl neat

(

*set* -x; *cd* "$(mktemp -d)" &&

*OS*="$(uname | tr '[:upper:]' '[:lower:]')" &&

*ARCH*="$(uname -m | sed -e 's/x86\_64/amd64/' -e 's/\(arm\)\(64\)\?.\*/\1\2/' -e 's/aarch64$/arm64/')" &&

*KREW*="krew-${*OS*}\_${*ARCH*}" &&

curl -fsSLO "https://github.com/kubernetes-sigs/krew/releases/latest/download/${*KREW*}.tar.gz" &&

tar zxvf "${*KREW*}.tar.gz" &&

./"${*KREW*}" install krew

)

*export* *PATH*="${*KREW\_ROOT*:-*$HOME*/.krew}/bin:*$PATH*"

kubectl krew install neat

# install kubeadm ubuntu 20.04

sudo apt install apt-transport-https curl

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add

echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" >> ~/kubernetes.list

sudo mv ~/kubernetes.list /etc/apt/sources.list.d

sudo apt update

sudo apt install kubelet -y

sudo apt install kubeadm -y

sudo apt install kubectl -y

sudo apt-get install -y kubernetes-cni

sudo apt-get install -y kubelet kubeadm kubectl kubernetes-cni

sudo swapoff –a

# sudo hostnamectl set-hostname kubernetes-master

sudo modprobe br\_netfilter

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

cat <<EOF | sudo tee /etc/docker/daemon.json

{ "exec-opts": ["native.cgroupdriver=systemd"],

"log-driver": "json-file",

"log-opts":

{ "max-size": "100m" },

"storage-driver": "overlay2"

}

EOF

sudo systemctl enable docker

sudo systemctl daemon-reload

sudo systemctl restart docker

apt install docker.io

sudo kubeadm init --ignore-preflight-errors=NumCPU,Mem --pod-network-cidr=10.244.0.0/16

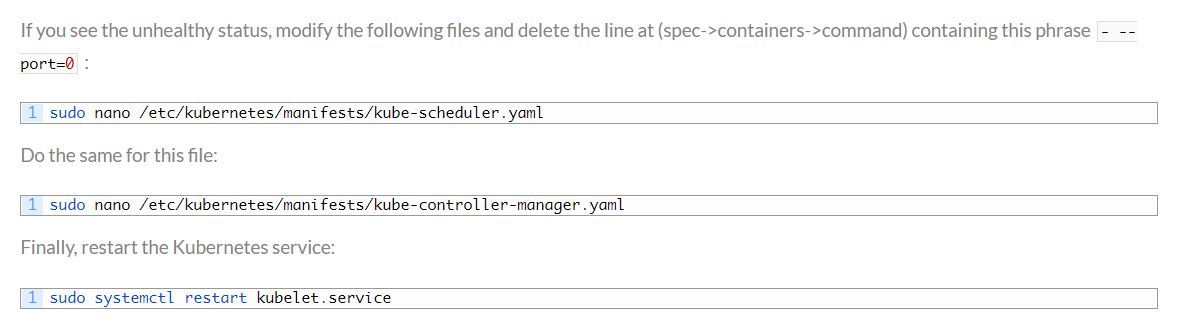
sudo ufw allow 6443

sudo ufw allow 6443/tcp

kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml

kubectl apply -f <https://raw.githubusercontent.com/coreos/flannel/master/Documentation/k8s-manifests/kube-flannel-rbac.yml>





sudo apt-get update -y

sudo apt-get install containerd –y

cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf

br\_netfilter

EOF

cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf

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

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

EOF

sudo sysctl --system

curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add

cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list

deb https://apt.kubernetes.io/ kubernetes-xenial main

EOF

sudo apt-get update -y

sudo apt-get install -y kubelet kubeadm kubectl

sudo apt-mark hold kubelet kubeadm kubectl

sudo apt-get install -y kubelet=1.20.1-00 kubeadm=1.20.1-00 kubectl=1.20.1-00