建议同步版本号，避免奇怪的问题

环境：centos 7

目的：部署k8s master，并简单部署一个tomcat的镜像，由k8s完成这个容器的调度

linux环境准备

如果多台机器，则每台都需要执行以下操作

关闭防火墙：

```
systemctl stop firewalld
systemctl disable firewalld
```

关闭selinux：

```
sed -i 's/enforcing/disabled/' /etc/selinux/config
setenforce 0
```

关闭swap：

```
swapoff -a 临时
sed -ri 's/.*swap.*/#&/' /etc/fstab 永久
free -g 验证，swap 必须为0；
```

将桥接的IPv4 流量传递到iptables 的链：

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

sysctl --system
```

docker准备

卸载系统之前的docker

```
sudo yum remove docker \
docker-client \
docker-client-latest \
docker-common \
docker-latest \
docker-latest-logrotate \
docker-logrotate \
docker-engine
```

安装Docker-CE

```
sudo yum install -y yum-utils \
device-mapper-persistent-data \
lvm2

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

sudo yum install -y docker-ce docker-ce-cli containerd.io
```

配置docker 加速

```
sudo mkdir -p /etc/docker
sudo tee /etc/docker/daemon.json <<-'EOF'
{
"registry-mirrors": ["https://82m9ar63.mirror.aliyuncs.com"]
}
EOF
sudo systemctl daemon-reload
sudo systemctl restart docker
```

启动docker & 设置docker 开机自启

```
systemctl enable docker
```

kube核心组件安装

添加阿里云yum 源

```
cat > /etc/yum.repos.d/kubernetes.repo << EOF
[kubernetes]
name=Kubernetes
baseurl=https://mirrors.aliyun.com/kubernetes/yum/repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=0
repo_gpgcheck=0
gpgkey=https://mirrors.aliyun.com/kubernetes/yum/doc/yum-key.gpg
https://mirrors.aliyun.com/kubernetes/yum/doc/rpm-package-key.gpg
EOF
```

安装kubeadm，kubelet 和kubectl

```
yum list|grep kube
yum install -y kubelet-1.17.3 kubeadm-1.17.3 kubectl-1.17.3
systemctl enable kubelet
systemctl start kubelet
```

部署k8s master

master 节点初始化

```
kubeadm init \
--apiserver-advertise-address=10.0.2.15 \
--image-repository registry.cn-hangzhou.aliyuncs.com/google_containers \
--kubernetes-version v1.17.3 \
--service-cidr=10.96.0.0/16 \
--pod-network-cidr=10.244.0.0/16
```

其中，10.0.2.15 改为你自己的master的ip

测试kubectl(主节点执行)

```
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
$ kubectl get nodes  //获取所有节点
```

目前master 状态为notready。等待网络加入完成即可。

安装Pod 网络插件（CNI）

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

这个文件需要翻墙，可以下载下来后传到主机上进行操作

等待大约3 分钟，master的状态即可变为ready

```
kubectl get pods -n kube-system 查看指定名称空间的pods
kubectl get pods –all-namespace 查看所有名称空间的pods
```

在Node 节点执行。
向集群添加新节点，执行在kubeadm init 输出的kubeadm join 命令

tomcat demo

部署一个tomcat

```
kubectl create deployment tomcat6 --image=tomcat:6.0.53-jre8

Kubectl get pods -o wide 可以获取到tomcat 信息
```

暴露nginx 访问

```
kubectl expose deployment tomcat6 --port=80 --target-port=8080 --type=NodePort
```

Pod 的80 映射容器的8080；service 会代理Pod 的80

动态扩容测试

```
kubectl get deployment
应用升级kubectl set image (--help 查看帮助)
扩容： kubectl scale --replicas=3 deployment tomcat6
```

扩容了多份，所有无论访问哪个node 的指定端口，都可以访问到tomcat6

删除

```
Kubectl get all
kubectl delete deploy/nginx
kubectl delete service/nginx-service
```

如果集群中只有master，可能会出现master无法部署pod的问题解决步骤如下

1. 查看节点名称

```
master01 ➜  k8s git:(master) kubectl get nodes
NAME       STATUS   ROLES           AGE     VERSION
master01   Ready    control-plane   2d16h   v1.24.2
master02   Ready    control-plane   47h     v1.24.2
master03   Ready    control-plane   47h     v1.24.2
worker01   Ready    <none>          47h     v1.24.2
worker02   Ready    <none>          47h     v1.24.2
```

2. 查看节点角色及是否支持schedule，例如查看master01节点

```
kubectl describe nodes master01 |grep -E '(Roles|Taints)'
```

如果出现NoSchedule，说明节点不允许被调度，需要调整node的角色

3. 允许节点部署pod

如果角色是master

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

如果角色是control-plane

```
kubectl taint nodes $node_name node-role.kubernetes.io/control-plane-
```

4. 禁止节点部署pod

```
kubectl taint nodes $node_name node-role.kubernetes.io/master=:NoSchedule
```

或者

```
kubectl taint nodes $node_name node-role.kubernetes.io/control-plane=:NoSchedule
```

最后的结果：

```
NAME                        READY   STATUS    RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     Pending   0          15m
[root@localhost ~]# kubectl describe nodes master01 | gerp -E '(Roles|Taints)'
-bash: gerp: command not found
Error from server (NotFound): nodes "master01" not found
[root@localhost ~]# kubectl describe nodes master01 | grep -E '(Roles|Taints)'
Error from server (NotFound): nodes "master01" not found
[root@localhost ~]# kubectl describe nodes localhost.localdomain | grep -E '(Roles|Taints)'
Roles:              master
Taints:             node-role.kubernetes.io/master:NoSchedule
[root@localhost ~]# kubectl taint nodes localhost.localdomain node-role.kubernetes.io/master-
node/localhost.localdomain untainted
[root@localhost ~]# kubectl describe nodes localhost.localdomain | grep -E '(Roles|Taints)'
Roles:              master
Taints:             <none>
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS              RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    0/1     ContainerCreating   0          17m
[root@localhost ~]# kubectl get pods
NAME                        READY   STATUS    RESTARTS   AGE
tomcat6-5f7ccf4cb9-mdd2n    1/1     Running   0          19m
[root@localhost ~]#
```

可以看到k8s中这个镜像正常的运行了