## 开源容器应用自动化部署技术Kubernetes

主讲：Cat 老师

北京动力节点教育科技有限公司  
2020 • 北京

动力节点•版权所有•禁止传播

## What is Kubernetes？

Kubernetes这个单词来自于希腊语，含义是 舵手 或 领航员；

Production-Grade Container Orchestration

Automated container deployment, scaling, and management

生产环境级别的容器编排

编排是什么意思？

1. 按照一定的目的依次排列；

2. 调配、安排；

Kubernetes，也称为K8S，其中8是代表中间“ubernete”的8个字符，是Google在2014年开源的一个容器编排引擎，用于自动化容器化应用程序的部署、规划、扩展和管理，它将组成应用程序的容器分组为逻辑单元，以便于管理和发现，用于管理云平台中多个主机上的容器化的应用，Kubernetes 的目标是让部署容器化的应用简单并且高效，很多细节都不需要运维人员去进行复杂的手工配置和处理；

Kubernetes拥有Google在生产环境上15年运行的经验，并结合了社区中最佳实践；

K8S是 [CNCF](https://cncf.io/) 毕业的项目，本来Kubernetes是Google的内部项目，后来开源出来，又后来为了其茁壮成长，捐给了CNCF；

CNCF全称Cloud Native Computing Foundation（云原生计算基金会）

官网：<https://kubernetes.io/>

代码：<https://github.com/kubernetes/kubernetes>

Kubernetes是采用Go语言开发的，Go语言是谷歌2009发布的一款开源编程语言；

## Kubernetes管理员认证（CKA）

CKA全称Certified Kubernetes Administrator，是Linux基金会和Cloud Native Computing Foundation（CNCF）官方推出的全球Kubernetes管理员认证，对于技术团队，CKA认证可以作为团队成员的技术能力的一个考察标准，也可以作为整个团队对Kubernetes云平台的管理能力的有力证明；

**考试难易程度**

考试只允查阅官方文档，在考试过程中你只能去

<https://kubernetes.io/>

<https://github.com>

如果去了其它的网站，按作弊处理；

**考试时长**  
考试时间为3小时；

**多少分及格**

CKA满分100分，66分及格；

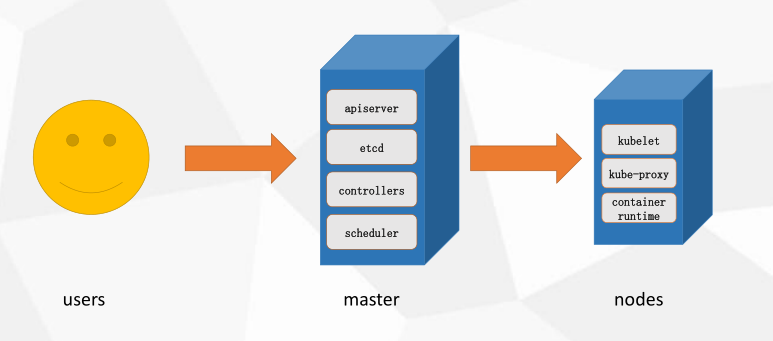
**考试费用**

美元：$300

人民币：¥2088

有一次免费重考的机会，一年后过期；

## Kubernetes整体架构



**Master**

k8s集群控制节点，对集群进行调度管理，接受集群外用户去集群操作请求；

Master Node 由 API Server、Scheduler、ClusterState Store（ETCD 数据库）和 Controller MangerServer 所组成；

**Nodes**

集群工作节点，运行用户业务应用容器；

Nodes节点也叫Worker Node，包含kubelet、kube proxy 和 Pod（Container Runtime）；

## Kubernetes环境搭建方式

部署 Kubernetes 环境（集群）主要有多种方式：

（1）minikube

minikube可以在本地运行Kubernetes的工具，minikube可以在个人计算机（包括Windows，macOS和Linux PC）上运行一个单节点Kubernetes集群，以便您可以试用Kubernetes或进行日常开发工作；

<https://kubernetes.io/docs/tutorials/hello-minikube/>

（2）kind

Kind和minikube类似的工具，让你在本地计算机上运行Kubernetes，此工具需要安装并配置Docker；

<https://kind.sigs.k8s.io/>

（3）kubeadm

Kubeadm是一个K8s部署工具，提供kubeadm init 和 kubeadm join两个操作命令，可以快速部署一个Kubernetes集群；

官方地址：

<https://kubernetes.io/docs/reference/setup-tools/kubeadm/kubeadm/>

<https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/>

（4）二进制包

从Github下载发行版的二进制包，手动部署安装每个组件，组成Kubernetes集群，步骤比较繁琐，但是能让你对各个组件有更清晰的认识；

（5）yum安装

通过yum安装Kubernetes的每个组件，组成Kubernetes集群，不过yum源里面的k8s版本已经比较老了，所以这种方式用得也比较少了；

1. 第三方工具

有一些大神封装了一些工具，利用这些工具进行k8s环境的安装；

1. 花钱购买

直接购买类似阿里云这样的公有云平台k8s，一键搞定；

## Kubeadm部署Kubernetes

kubeadm是官方社区推出的一个用于快速部署 kubernetes 集群的工具，这个工具能通过两条指令完成一个kubernetes集群的部署；

1. 创建一个Master节点：

kubeadm init

1. 将Node节点加入到Master集群中：

$ kubeadm join <Master节点的IP和端口>

## Kubernetes部署环境要求

（1）一台或多台机器，操作系统CentOS 7.x-86\_x64

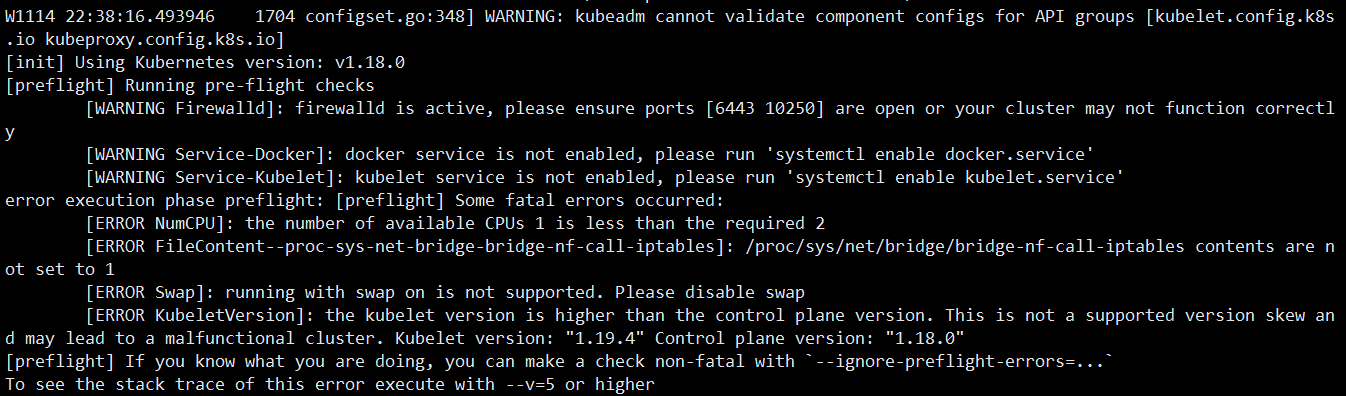
（2）硬件配置：内存2GB或2G+，CPU 2核或CPU 2核+；

（3）集群内各个机器之间能相互通信；

（4）集群内各个机器可以访问外网，需要拉取镜像；

（5）禁止swap分区；

如果环境不满足要求，会报错，比如：



## Kubernetes部署环境准备

# 关闭防火墙

systemctl stop firewalld

systemctl disable firewalld

# 关闭selinux

sed -i 's/enforcing/disabled/' /etc/selinux/config #永久

setenforce 0 #临时

# 关闭swap（k8s禁止虚拟内存以提高性能）

sed -ri 's/.\*swap.\*/#&/' /etc/fstab #永久

swapoff -a #临时

# 在master添加hosts

cat >> /etc/hosts << EOF

192.168.79.130 k8smaster

192.168.79.131 k8snode

EOF

# 设置网桥参数

cat > /etc/sysctl.d/k8s.conf << EOF

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

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

EOF

sysctl --system #生效

# 时间同步

yum install ntpdate -y

ntpdate time.windows.com

## Kubernetes安装具体步骤

所有服务器节点安装 Docker/kubeadm/kubelet/kubectl

Kubernetes 默认容器运行环境是Docker，因此首先需要安装Docker；

（1）安装 Docker

更新docker的yum源

yum install wget -y

wget https://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo -O /etc/yum.repos.d/docker-ce.repo

安装指定版本的docker：

yum install docker-ce-19.03.13 -y

*yum install docker -y （这个安装的Docker版本偏旧） 1.13.x*

配置加速器加速下载 （<https://cr.console.aliyun.com/>）

sudo mkdir -p /etc/docker

sudo tee /etc/docker/daemon.json <<-'EOF' { "registry-mirrors": ["https://stv2roat.mirror.aliyuncs.com"] } EOF

sudo systemctl daemon-reload

sudo systemctl restart docker

/etc/docker/daemon.json

{

"registry-mirrors": ["https://registry.docker-cn.com"]

}

https://gg3gwnry.mirror.aliyuncs.com

然后执行：

systemctl enable docker.service

不然会提示警告；

那么接下来需要搭建：kubeadm、kubelet、kubectl

1. 添加k8s的阿里云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

到时候下载k8s的相关组件才能找到下载源；

（2）安装 kubeadm，kubelet 和 kubectl

yum install kubelet-1.19.4 kubeadm-1.19.4 kubectl-1.19.4 -y

然后执行：

systemctl enable kubelet.service

不然会提示警告；

查看有没有安装：

yum list installed | grep kubelet

yum list installed | grep kubeadm

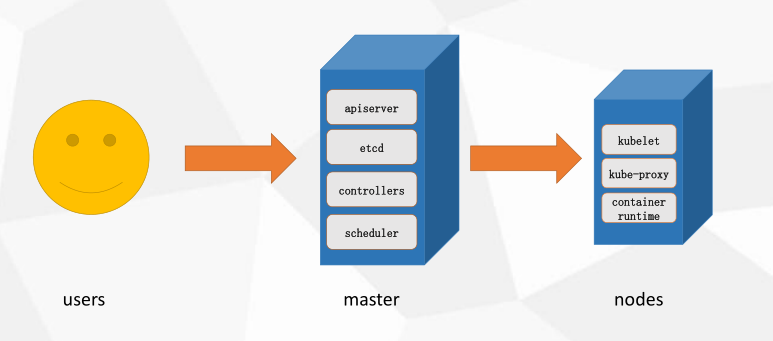
yum list installed | grep kubectl

查看安装的版本： kubelet --version

Kubelet：运行在cluster所有节点上，负责启动POD和容器；

Kubeadm：用于初始化cluster的一个工具；

Kubectl：kubectl是kubenetes命令行工具，通过kubectl可以部署和管理应用，查看各种资源，创建，删除和更新组件；



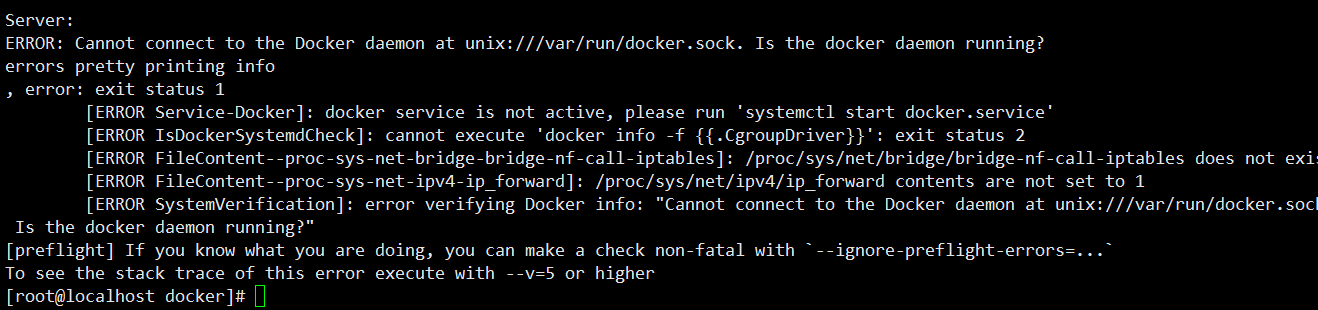
此时应该重启一下centos；

（3）部署Kubernetes Master主节点

kubeadm init --apiserver-advertise-address=192.168.79.130 --image-repository registry.aliyuncs.com/google\_containers --kubernetes-version v1.19.4 --service-cidr=10.96.0.0/12 --pod-network-cidr=10.244.0.0/16

此命令在master机器上执行；

执行报错：



解决：重启一下centos，然后再执行上面的 kubeadm init 命令；

说明：

service-cidr 的选取不能和PodCIDR及本机网络有重叠或者冲突，一般可以选择一个本机网络和PodCIDR都没有用到的私网地址段，比如PODCIDR使用10.244.0.0/16, 那么service cidr可以选择10.96.0.0/12，网络无重叠冲突即可；

接下来在master机器上执行：

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

接下来把node节点加入Kubernetes master中，在Node机器上执行；

向集群添加新节点，执行的命令就是kubeadm init最后输出的kubeadm join命令：

kubeadm join 192.168.172.132:6443 --token wa5bif.zfuvbesevdfvf4of \

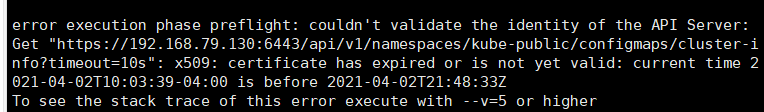
--discovery-token-ca-cert-hash sha256:87cf5828d54dd80da13c4b57c57360370ea0267a7cc3991989ca3006cf3e44d8

这是token过期，重新生成就行 并保证时间一直性。实在不行重新初始化

**需要先卸载，此步为删除节点操作。通过kubectl get nodes 已经没有node节点了**

**kubectl delete node <node name>**

**kubeadm reset**



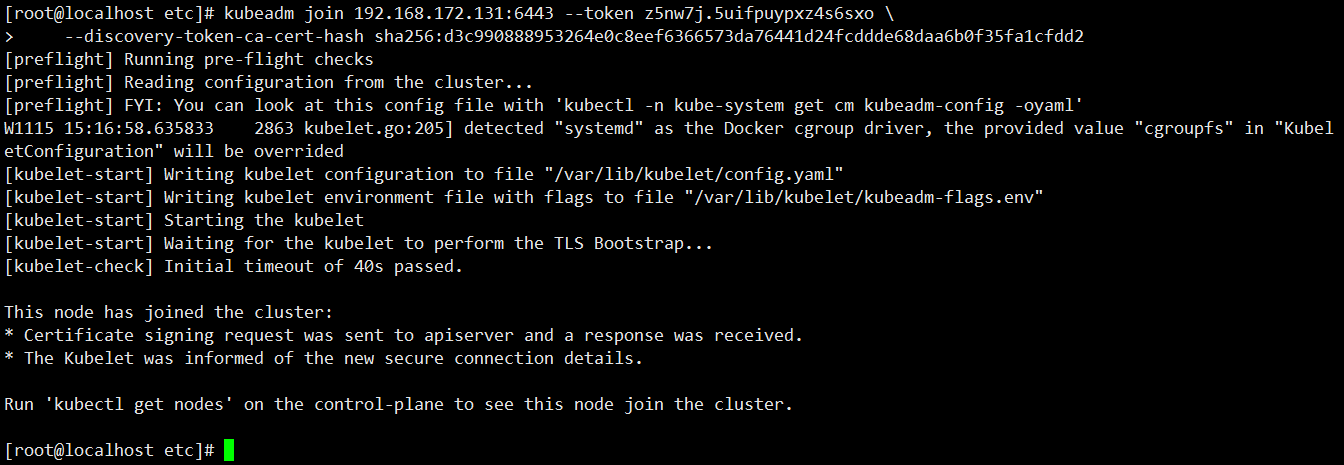
kubeadm token generate #生成token

kubeadm token create 6nzavn.xrnhpsbjcnzz474l --print-join-command --ttl=0 #根据token输出添加命令

kubeadm token create --print-join-command

kubeadm join 192.168.79.130:6443 --token lkiwjm.p1nmgvbh0w3lchpw \

--discovery-token-ca-cert-hash sha256:6fd8cf9ed88cb93614a025116df0396a83cbe39c2ad1b90e461fde2215dabd7d



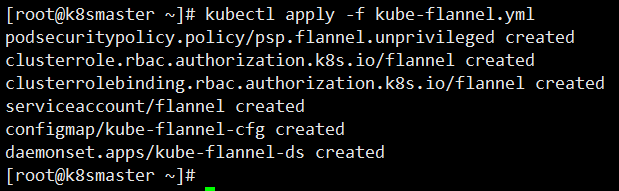
## 部署网络插件

下载kube-flannel.yml文件

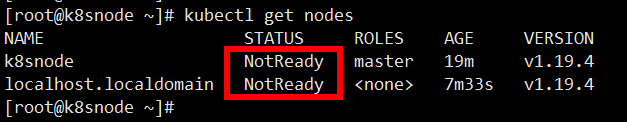
wget https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml

应用kube-flannel.yml文件得到运行时容器

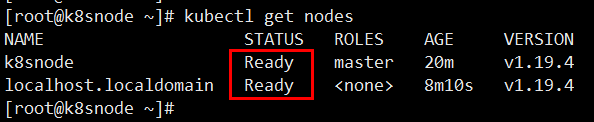
kubectl apply -f kube-flannel.yml （在master机器上执行）



然后查看节点状态： kubectl get nodes （在master机器上执行）



说明还没有就绪，需要等一会儿，然后节点就就绪了；



至此我们的k8s环境就搭建好了；

查看运行时容器pod （一个pod里面运行了多个docker容器）

kubectl get pods -n kube-system

## Kubernetes部署“容器化应用”

Docker应用-->在docker里面部署一个java程序（springboot）

**在Kubernetes集群中部署一个Nginx：**

kubectl create deployment nginx --image=nginx

kubectl expose deployment nginx --port=80 --type=NodePort

kubectl get pod,svc

kubectl get service

访问地址：<http://NodeIP:Port>

docker ps 查看容器在哪台机器上启动着

**在Kubernetes集群中部署一个Tomcat：**

kubectl create deployment tomcat --image=tomcat

kubectl expose deployment tomcat --port=8080 --type=NodePort

访问地址：<http://NodeIP:Port>

**K8s部署微服务（springboot程序）**

1. 项目打包（jar、war）-->可以采用一些工具git、maven、jenkins
2. 制作Dockerfile文件，生成镜像；
3. kubectl create deployment nginx --image= 你的镜像
4. 你的springboot就部署好了，是以docker容器的方式运行在pod里面的；