用云服务器搭建kubernetes

samctfu 20210606

这几天学习了kubernetes，并且自己试着搭建kubernetes集群，在搭建集群的过程中遇到了很多问题，在这期间我也网上很多教程根本搭不出来，有的教程干脆直接抄来抄去（这完全是在给互联网世界灌输垃圾数据），有感于此我决定记录一下我搭建kubernetes的经历，并且总结与反思遇到的一些问题。

服务器采用腾讯云服务器，在这里我注册了三台（图片很长，我裁成了两段）





这里的“可用区”就是你在创建云服务器时候选的主机地址，这里最好选择海外的主机，因为到时候下载文件会快一些。

操作系统版本我选择的是centOS7.4，64位操作系统（我用过centOS6.8搭建kubernetes，嗯失败了），ubuntu操作系统，版本号大于等于16应该也行，但是，如果你是云服务器不建议用ubuntu系统（有一个关于root权限的问题，后面再讲），如果你是自己配的服务器，用ubuntu系统没问题。

云服务器的配置，CPU核心数大于等于2，内存容量大于等于4G，硬盘容量大于等于50G，最好不要低于这个最低限度，因为在部署kubernetes的时候会检查机器配置，配置过低会导致部署失败（网上也有一些教程让kubernetes不检查机器配置，但是我按照那样的教程做并没有成功，所以还是按要求做吧，少一点骚操作）。

这里的“网络计费模式”建议选择按流量计费，因为搭建kubernetes本身并不需要太长时间（只要你掌握了方法，找到了正确的教程），选择包月计费的话，后面的时间云服务器基本闲置，嗯当然，土豪请忽略。

使用云服务器，腾讯云会让你用微信扫码验证，扫码的时候如果你的微信不是最新版，扫码验证会失败，所以还需要提前更新微信（虽然我不明白，微信是不是最新版和扫码验证有什么联系）。

然后，在你自己的电脑上安装secureCRT，这是常用的远程连接服务器的软件，安装好secureCRT之后，你会得到secureCRT和secureFX两个软件，这两个软件的基本教程可以自己百度学习一下；其他类似的软件Xshell，Xftp6，向日葵什么的都可以。什么？你说有些软件要花钱买激活码。北海，我只能告诉你要多想。

接下来就是正式部署kubernetes了，首先要以root的身份登录云服务器，centOS是没有问题的。ubuntu就不行了，因为ubuntu在创建的时候是没有root用户的，需要以userOne的身份创建root用户，再以root用户身份登录，这两步我用自己的虚拟机能实现，但是用云服务器第二步死活实现不了。

反正三台服务器都以root用户登陆了，其中一台作为master，另外两台作为slave（近年来，slave被视为political incorrect，因为这个单词会引出一大段具有争议的历史，所以有的地方会把slave换成worker）

部署kubernetes推荐参考这篇博客<https://segmentfault.com/a/1190000037682150>

二、安装docker环境（三台机器都要装）

# 安装docker所需的工具

yum install -y yum-utils device-mapper-persistent-data lvm2

# 配置阿里云的docker源，用谷歌源也可以，清华源不行，说找不到指定的文件

yum-config-manager --add-repo http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo

# 指定安装这个版本的docker-ce，docker-ce是docker的稳定版

yum install -y docker-ce-18.09.9-3.el7

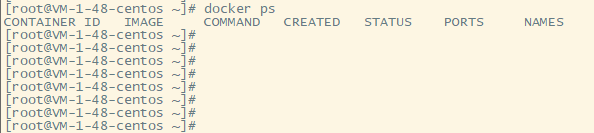
# 启动docker，其中systemctl enable docker是指设置成开机启动，systemctl start docker是指运行程序

systemctl enable docker && systemctl start docker

#验证docker安装成功（随便执行一条docker指令就可以啦）

docker ps

如果你显示的结果是这个，说明你的docker安装成功了



三、设置kubernetes环境准备条件（三台机器都要装）

# 关闭防火墙的开机启动（可以不做，下次开机再手动关闭）

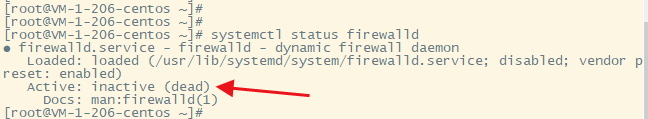
systemctl disable firewalld

# 关闭防火墙

systemctl stop firewalld

#检查防火墙的状态

systemctl status firewalld



注意语句Active: inactive (dead) 说明防火墙已经关闭

# 关闭selinux

# 临时禁用selinux

setenforce 0

# 永久关闭（可以不做，下次开机再手动关闭） 修改/etc/sysconfig/selinux文件设置

sed -i 's/SELINUX=permissive/SELINUX=disabled/' /etc/sysconfig/selinux

sed -i "s/SELINUX=enforcing/SELINUX=disabled/g" /etc/selinux/config

# 禁用交换分区（不禁用的话后面会报错，swap正在执行啊什么的）

swapoff -a

# 永久禁用，打开/etc/fstab注释掉swap那一行（用正则表达式写的）

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

# 修改内核参数，EOF是结束标识符，/etc/sysctl.d/k8s.conf是目标文件，允许路由转发，不对bridge的数据进行处理

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

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

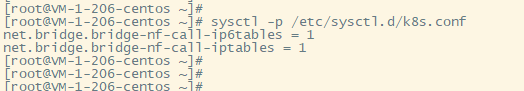
net.bridge.bridge-nf-call-iptables = 1

EOF

#生效配置文件

sysctl -p /etc/sysctl.d/k8s.conf

这时terminal会输出和你执行cat语句时类似的内容，说明配置文件已经生效了



#写入系统参数（这行语句不执行似乎也可以）

sysctl --system

四、安装k8s v1.16.0 （master节点需要这一步）

先确认步骤二安装docker-ce 18.09.9（三台机器都要装）。如果没设置k8s环境准备条件，参照本文步骤三设置k8s环境准备条件（三台机器都要装）执行。

以上两个步骤检查完毕之后，继续以下步骤。

安装kubeadm、kubelet、kubectl

# 执行配置k8s阿里云源

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

[kubernetes]

name=Kubernetes

baseurl=https://mirrors.aliyun.com/kubernetes/yum/repos/kubernetes-el7-x86\_64/

enabled=1

gpgcheck=1

repo\_gpgcheck=1

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、kubectl、kubelet

yum install -y kubectl-1.16.0-0 kubeadm-1.16.0-0 kubelet-1.16.0-0

# kubelet设置成启动并且运行kubelet服务

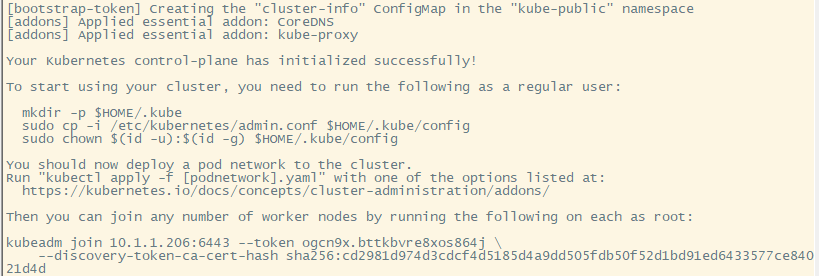
systemctl enable kubelet && systemctl start kubelet

初始化k8s 以下这个命令开始安装k8s需要用到的docker镜像，因为无法访问到国外网站，所以这条命令使用的是国内的阿里云的源(registry.aliyuncs.com/google\_containers)。另一个非常重要的是：这里的--apiserver-advertise-address使用的是master和node间能互相ping通的ip（就是master的IP），我这里是内网IP（10.1.1.206）

kubeadm init --image-repository registry.aliyuncs.com/google\_containers --kubernetes-version v1.16.0 --apiserver-advertise-address 10.1.1.206 --pod-network-cidr=10.244.0.0/16 --token-ttl 0

这条命令执行时会卡在[preflight] You can also perform this action in beforehand using ''kubeadm config images pull，大概需要2-5分钟（国内访问会慢一些），请耐心等待，千万不要ctrl + c回到命令行。几分钟之后，kubernetes节点就安装好了，如下图所示。





kubernetes还给出了服务器作为worker节点加入集群的命令，这个最好复制出来单独保存

########

kubeadm join 10.1.1.206:6443 --token ogcn9x.bttkbvre8xos864j \

--discovery-token-ca-cert-hash sha256:cd2981d974d3cdcf4d5185d4a9dd505fdb50f52d1bd91ed6433577ce84021d4d

########

# 上面安装完成后，kubernetes会提示你输入如下命令，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

六、 安装flannel（master节点需要这一步，先做第六步，再做第五步）

# 下载官方fannel配置文件 使用wget命令

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

这个地址国内访问不了，解决方案还是参考这篇博客https://segmentfault.com/a/1190000037682150

# 安装fannel

kubectl apply -f kube-flannel.yml



五. 安装k8s v1.16.0 node（worker节点需要这一步）

先确认步骤二安装docker-ce 18.09.9（三台机器都要装）。如果没设置k8s环境准备条件，参照本文步骤三设置k8s环境准备条件（三台机器都要装）执行。

# 执行配置k8s阿里云源

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

[kubernetes]

name=Kubernetes

baseurl=https://mirrors.aliyun.com/kubernetes/yum/repos/kubernetes-el7-x86\_64/

enabled=1

gpgcheck=1

repo\_gpgcheck=1

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、kubectl、kubelet

yum install -y kubeadm-1.16.0-0 kubelet-1.16.0-0

# 启动kubelet服务

systemctl enable kubelet && systemctl start kubelet

加入集群 这里加入集群的命令每个人都不一样（就是刚才说需要单独保存的命令）

kubeadm join 10.1.1.206:6443 --token ogcn9x.bttkbvre8xos864j \

--discovery-token-ca-cert-hash sha256:cd2981d974d3cdcf4d5185d4a9dd505fdb50f52d1bd91ed6433577ce84021d4d

如果加入节点卡在[preflight]迟迟不动，说明出问题了，注意看terminal窗口的提示信息。

（5.1）比如出现了[WARNING IsDockerSystemdCheck]: detected “cgroupfs“ as the Docker cgroup driver 大概意思是说不建议使用cgroupfs作为驱动，建议使用systemed作为驱动

解决方案，先回到terminal窗口，执行语句

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

{

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

}

EOF

# 重启docker

systemctl restart docker

（5.2）先不要急着重新加入集群，再检查一下还有没有其他warning，比如有的会提示说你的docker和kubectl没有设为开机自启动，这个按照提示的内容照做就好了

（5.3）处理完这些warning之后再次执行加入集群的命令，如果还是卡在了[preflight]超过三分钟，说明还是有问题，回到terminal窗口，执行语句

kubeadm join 10.1.1.206:6443 --token ogcn9x.bttkbvre8xos864j \

--discovery-token-ca-cert-hash sha256:cd2981d974d3cdcf4d5185d4a9dd505fdb50f52d1bd91ed6433577ce84021d4d --v=2

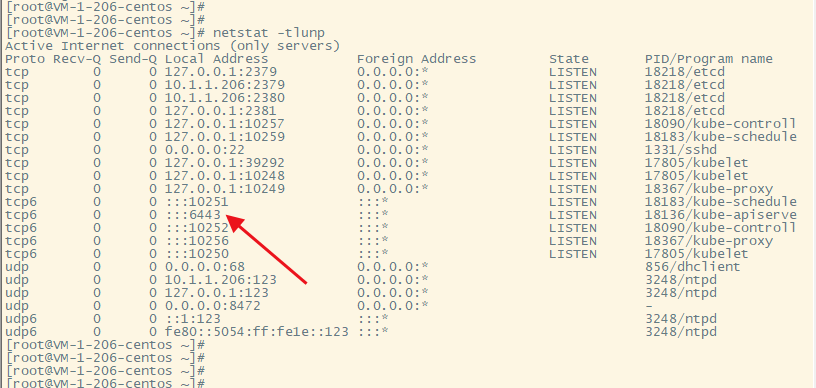
这个语句与之前的不同之处在于带了一个尾巴，它会打印出执行语句过程中的详细信息，比如我就遇到过worker节点一直在等待master节点返回建立集群的信息，由于等待时间过长，导致无法加入集群

先检查worker节点和master节点是否能ping通

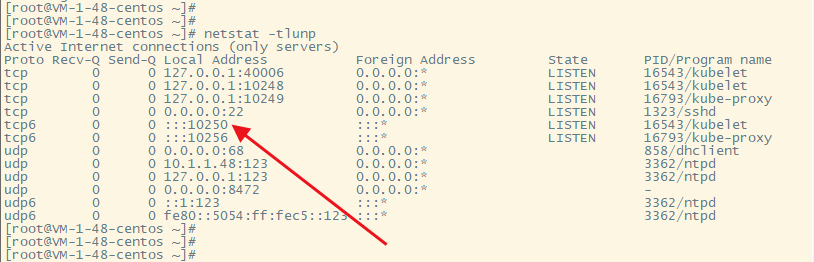
如果能ping通，这时需要查看worker节点和master节点是否开放了对应的端口

netstat -tlunp

检查master节点发现6443端口是开放的

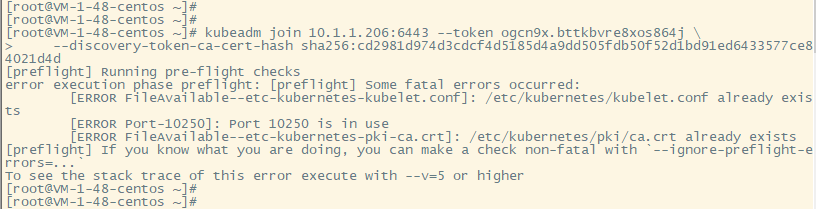


检查worker节点发现10250端口是开放的



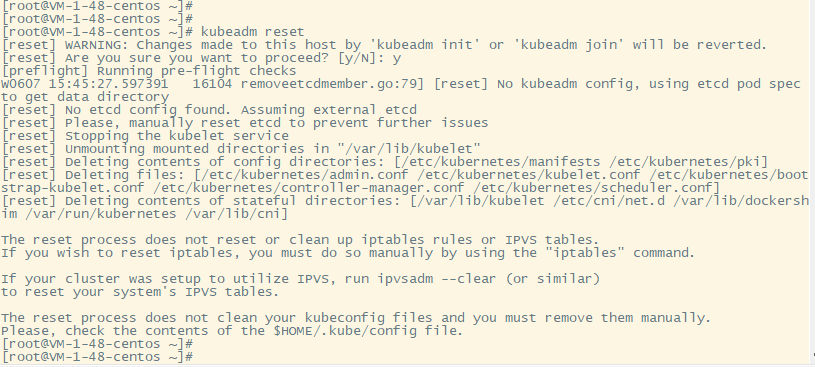
双方的端口都没有问题，这说明问题不在机器本身，在网上找了各种教程，最后才明白，我使用的是云主机，并不是普通的虚拟机，所以端口能不能访问除了上述因素控制以外，还有在云平台上的安全组控制，果然一看，云主机绑定的安全组并没有打开这个端口，在将这个端口下行权限加入到安全组之后，果然成功。

（5.4）如果执行加入集群的命令，出现如下的错误

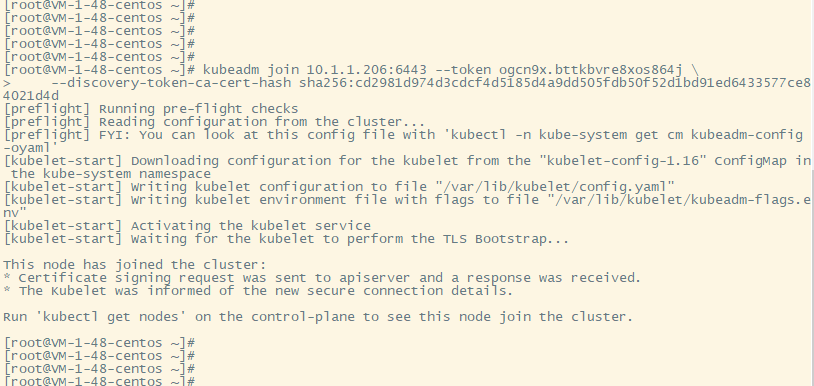


说明你的worker节点曾经加入过其他集群，重启kubeadm，kubeadm会自动帮你删除曾经加入集群的配置文件，如下图所示

kubeadm reset

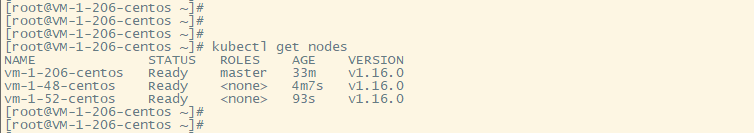


如果加入集群成功，将会看到这些信息



在master节点通过kubectl查看集群的状态

kubectl get nodes



至此，kubernetes的搭建就成功了