SOP-k8s迁移部署

一：初始环境

准备三台Ubuntu18.04至少双核

主机名	IP	用途
k8s-master	10.	k8s管理节点
k8s-node1	10.	k8s工作节点
k8s-node2	10.	k8s工作节点

二：修改主机名，更新/etc/hosts

三：安装docker

3.1安装apt依赖包，用于通过HTTPS来获取仓库

```
sudo apt-get install  apt-transport-https ca-certificates  curl  gnupg-agent  software-properties-common
```

3.2添加docker的官方GPG密钥

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

##如上面的地址不能下载，可以使用国内镜像

curl -fsSL https://mirrors.ustc.edu.cn/docker-ce/linux/ubuntu/gpg | sudo apt-key add -

3.3设置稳定版仓库

```
sudo add-apt-repository   "deb [arch=amd64] https://download.docker.com/linux/ubuntu  $(lsb_release -cs)
stable"    #ubuntu
```

```
sudo add-apt-repository "deb [arch=amd64] http://mirrors.aliyun.com/docker-ce/linux/ubuntu $(lsb_release -cs)
stable"
```

3.4安装docker-ce

```
##
sudo apt-get update
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
##apt-cache madison docker-ce
sudo apt install docker-ce=5:19.03.1~3-0~ubuntu-bionic  docker-ce-cli=5:19.03.1~3-0~ubuntu-bionic
```

3.5配置daemon.json文件，镜像加速

```
{
  "registry-mirrors": ["https://docker.mirrors.ustc.edu.cn"]
}
```

修改后重启docker

```
sudo  systemctl daemon-reload
sudo  service docker restart
```

查看版本

```
docker version      docker  -v
```

设置开机自启

```
systemctl enable docker
```

四：关闭防火墙、swap

4.1防火墙

```
sudo ufw disable
```

4.2关闭swap

```
/swap.img      none    swap    sw      0       0     #/etc/fstab
```

```
mount -a
sudo swapoff  -a
```

五：安装kubelet、kubeadm、kubectl

5.1添加阿里源 ##云平台服务器不需要添加

```
deb https://mirrors.aliyun.com/kubernetes/apt kubernetes-xenial main    #/etc/apt/sources.list
curl https://mirrors.aliyun.com/kubernetes/apt/doc/apt-key.gpg | sudo apt-key add -         #key
sudo apt-get update       #
```

5.2下载 Google Cloud 公开签名秘钥：

```
sudo curl -fsSLo /usr/share/keyrings/kubernetes-archive-keyring.gpg https://packages.cloud.google.com/apt/doc
/apt-key.gpg
```

5.3添加 Kubernetesapt仓库：

```
echo "deb [signed-by=/usr/share/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-
xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list
```

5.4开始安装

```
####  sudo apt-get install -y kubelet kubeadm kubectl
sudo apt update
sudo apt  install -y kubectl=1.17.12-00  kubeadm=1.17.12-00  kubelet=1.17.12-00
kubelet  --version
systemctl  start  kubelet
```

###以上所有主机均要操作

六：部署master节点

6.1下载

固定版本安装

```
sudo  docker pull  k8s.gcr.io/kube-proxy:v1.17.12
sudo  docker pull  k8s.gcr.io/kube-apiserver:v1.17.12
sudo  docker pull  k8s.gcr.io/kube-controller-manager:v1.17.12
sudo  docker pull  k8s.gcr.io/kube-scheduler:v1.17.12
sudo  docker pull  quay.io/coreos/flannel:v0.13.0-rc2
sudo  docker pull  k8s.gcr.io/etcd:3.4.3-0
sudo  docker pull  quay.io/prometheus/node-exporter:v0.18.1
sudo  docker pull  quay.io/coreos/kube-rbac-proxy:v0.4.1
sudo  docker pull  k8s.gcr.io/pause:3.1
```

按最新版本安装

```
####kubeadm config images list    #
```

6.2执行init，初始化k8s

```
kubeadm init  --kubernetes-version v1.17.12   --pod-network-cidr=10.244.0.0/16    --service-cidr=10.1.0.0/16 --
apiserver-advertise-address=192.168.255.123
    # joinnode
    #  --apiserver-advertise-address ipk8s
```

6.3要使非 root 用户可以运行 kubectl，请运行以下命令，它们也是kubeadm init输出的一部分

```
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

七：node节点执行join

```
kubeadm join 172.21.12.225:6443 --token ueu4y4.s3nr8nl43aqk97c3 \
    --discovery-token-ca-cert-hash sha256:bdf2dde6755475ef848eee514dae19da14ceac7f375a382eacfbb1dcf3ae0155
```

八：搭建flannel网络集群

```
curl https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml >>kube-flannel.yml
chmod 777 kube-flannel.yml
kubectl apply -f kube-flannel.yml
```

九：安装部署dashboard仪表盘

9.1部署Dashboard UI

默认情况下kunernetes是不会部署Dashboard的。可以通过以下命令部署

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.2.0/aio/deploy/recommended.yaml
```

查看是否创建成功

```
kubectl get pods --all-namespaces
```

dashboard服务的namespace是kubernetes-dashboard，但是该服务的类型是ClusterIP，不便于通过浏览器访问，因此需要改成NodePort类型。

查看现有的服务

```
kubectl get svc --all-namespaces
```

删除

```
kubectl delete service kubernetes-dashboard --namespace=kubernetes-dashboard
```

创建配置文件

```
vim dashboard-svc.yaml

#
kind: Service
apiVersion: v1
metadata:
  labels:
    k8s-app: kubernetes-dashboard
  name: kubernetes-dashboard
  namespace: kubernetes-dashboard
spec:
  type: NodePort
  ports:
    - port: 443
      targetPort: 8443
  selector:
    k8s-app: kubernetes-dashboard


#
kubectl apply -f dashboard-svc.yaml
```

再次查看服务：kubectl get svc --all-namespaces

9.2创建管理员角色

需要访问Dashboard服务，要有访问权限，创建kubernetes-dashboard管理员角色

```
vim dashboard-svc-account.yaml

#
apiVersion: v1
kind: ServiceAccount
metadata:
  name: dashboard-admin
  namespace: kube-system
---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1beta1
metadata:
  name: dashboard-admin
subjects:
  - kind: ServiceAccount
    name: dashboard-admin
    namespace: kube-system
roleRef:
  kind: ClusterRole
  name: cluster-admin
  apiGroup: rbac.authorization.k8s.io


#
kubectl apply -f dashboard-svc-account.yaml
```

9.3获取token

```
# kubectl get secret -n kube-system |grep admin|awk '{print $1}'
dashboard-admin-token-s8q69
# kubectl describe secret dashboard-admin-token-s8q69 -n kube-system|grep '^token'|awk '{print $2}'
eyJhbGciOiJSUzI1NiIsImtpZCI6IlBPMGtDVWVPWW9HN0dxT1g3RmxuY01DOEJDamk1NldldDQtcmZDLWxBOUUifQ.
eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJrdWJlLXN5
c3RlbSIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VjcmV0Lm5hbWUiOiJkYXNoYm9hcmQtYWRtaW4tdG9rZW4tczhxNjkiLCJrdWJl
cm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1lIjoiZGFzaGJvYXJkLWFkbWluIiwia3ViZXJuZXRlcy5pby9zZXJ2
aWNlYWNjb3VudC9zZXJ2aWNlLWFjY291bnQudWlkIjoiZDk3ZjQxMzAtZjM5Yy00Yzk3LWJkODktZjE5N2E4Y2I0NWJiIiwic3ViIjoic3lzdGVt
OnNlcnZpY2VhY2NvdW50Omt1YmUtc3lzdGVtOmRhc2hib2FyZC1hZG1pbiJ9.dxXxTrwndmxNppKGqwLDO7LAKEaN-NCQjJPvrFBHW15ETFmhtd-
7b0yT1uu53Wzb6jKmrFejD-
Yq56UJ1C1pYdBzh_qE1trXSiIgl4pmE4WFBANzjbiO6eQ1oE6R3ShFWBOEpI3tHee8q2qQxTpB3JFMu6IjJ52i2eyUBEpqdWNKWSAxXM0SZcZfGg
wf5MDv10GSdrJClUDATivjSBhFTyrXrceBOb5NdnCNsJ_n9cs3SLP60EECjDeyRC5Rw-
cORfMLvfoRDbNn2P_Dfh0p5lptWQDu2dStqzXcfmUmh7yirl7GSR-xB_72wTbKdz3svyEvWKO_d_jtWiLsJPfFiQ
```

配置相应的入站规则即可访问。