

**Kubernetes部署安装手册**

**（基于kubeadm）**

2019-08-20

目 录

[1 适用对象 1](#_Toc18141095)

[2 部署规划 2](#_Toc18141103)

[3 系统配置 3](#_Toc18141104)

[3.1 设置主机名 3](#_Toc18141105)

[3.2 关闭防火墙及swap分区 3](#_Toc18141106)

[3.3 设置netfilter 4](#_Toc18141107)

[3.4 免密钥登录 4](#_Toc18141108)

[4 常用命令 6](#_Toc18141109)

[4.1 Kubernetes常用命令 6](#_Toc18141110)

[4.1.1 集群命令 6](#_Toc18141111)

[4.1.2 Pod命令 6](#_Toc18141112)

[4.2 Docker常用命令 6](#_Toc18141113)

[4.2.1 容器命令 6](#_Toc18141114)

[4.2.2 镜像命令 7](#_Toc18141115)

[5 国内镜像源 8](#_Toc18141116)

[5.1 Azure镜像源 8](#_Toc18141117)

[5.2 阿里云镜像源 8](#_Toc18141118)

[5.2.1 Yum源配置 8](#_Toc18141119)

[5.2.2 Docker源配置 8](#_Toc18141120)

[6 官方文档 10](#_Toc18141121)

[6.1 Kubeadm 10](#_Toc18141122)

[6.1.1.1 10](#_Toc18141123)

[6.2 Kubesphray 10](#_Toc18141124)

[6.2.1.1 10](#_Toc18141125)

[6.3 Rancher 10](#_Toc18141126)

[6.3.1.1 官方部署文档 10](#_Toc18141127)

[6.3.1.2 其它优秀文档 10](#_Toc18141128)

[7 常见问题 11](#_Toc18141129)

# 适用对象

本手册适用于kubernetes部署、开发人员以及有一定Linux相关基础的运维人员。



# 部署规划

部署规划参考：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **角色** | **IP地址** | **主机名** | **部署组件** |
| 1 | kubespray manager | 10.110.30.41 | kmanager.k8s.pcp |  |
| 2 | master,node | 10.110.30.42 | kmaster1.k8s.pcp |  |
| 3 | master,node | 10.110.30.43 | kmaster2.k8s.pcp |  |
| 4 | node | 10.110.30.49 | knode1.k8s.pcp |  |
| 5 | node | 10.110.30.51 | knode2.k8s.pcp |  |

# 系统配置

## 设置主机名

1. 按照规划，设置主机名为：

hostnamectl set-hostname master.k8s.pcp

hostnamectl set-hostname node1.k8s.pcp

hostnamectl set-hostname node2.k8s.pcp

1. 登录master节点，修改hosts文件：

vi /etc/hosts

将集群中各节点的映射关系添加到此文件中：

10.110.30.41 master.k8s.pcp

10.110.30.42 node1.k8s.pcp

10.110.30.43 node2.k8s.pcp

1. 将修改好的配置文件分发到集群中其他节点：

scp /etc/hosts root@node1.k8s.pcp:/etc

scp /etc/hosts root@node2.k8s.pcp:/etc

## 关闭防火墙及swap分区

1. 关闭防火墙：

systemctl stop firewalld

systemctl disable firewalld

1. 关闭iptables：

systemctl disable iptables

systemctl stop iptables

1. 关闭seLinux

setenforce 0

vi /etc/selinux/config

将其中的SELinux设置为：

SELINUX=disabled

1. 关闭swap分区

swapoff -a

## 设置netfilter

1. 创建k8s.conf：

vi /etc/sysctl.d/k8s.conf

添加

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

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

net.ipv4.ip\_forward = 1

1. 使配置生效：

modprobe br\_netfilter

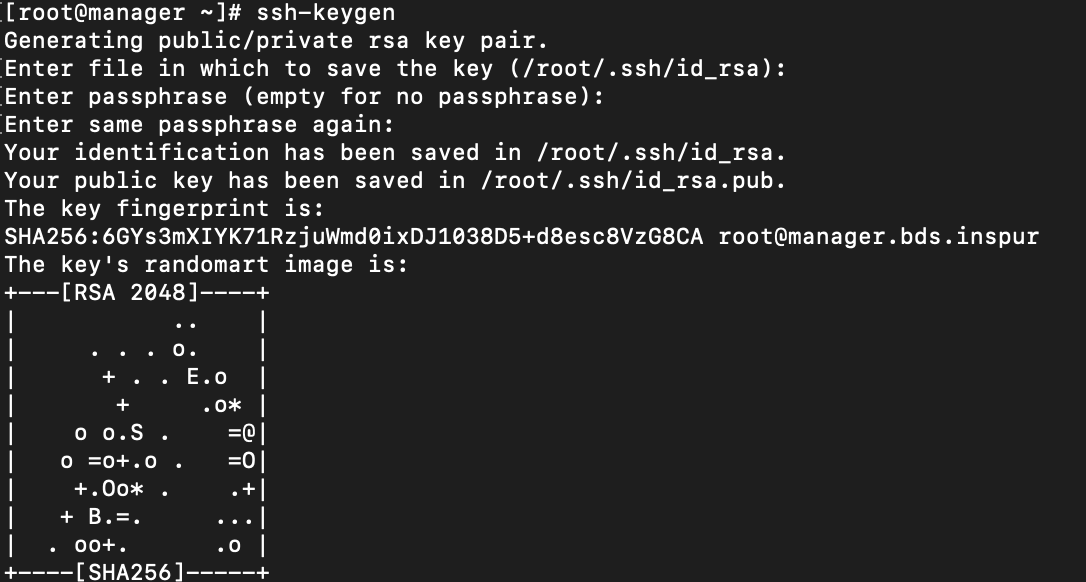
sysctl -p /etc/sysctl.d/k8s.conf

## 免密钥登录

1. 登录主节点，生成公钥、私钥对：

ssh-keygen

1. 一路回车执行，如图：



1. 将公钥复制到所有节点。注意：须往本机拷贝一份：

ssh-copy-id root@master.k8s.pcp

ssh-copy-id root@node1.k8s.pcp

ssh-copy-id root@node2.k8s.pcp

# 常用命令

## Kubernetes常用命令

### 集群命令

* **查看集群组件状态**

kubectl get componentstatuses

* **查看kubelet日志**

journalctl -xeu kubelet

journalctl -f -u kubelet

* **获取集群信息**

kubectl cluster-info

kubectl get cs

### Pod命令

#### Pod

* **查看pod列表**

#指定命名空间的pod列表

kubectl get pods -n {namespace-name} -o wide

#所有命名空间的pod列表

kubectl get pods --all-namespaces -o wide

* **查看pod日志**

kubectl logs {pod-name} -n {namespace-name}

* **查看pod详情**

kubectl describe pod {pod-name} -n {namespace-name}

* **将容器运行在pod中**

kubectl run {pod-name} --image={repository:tag} --replicas=1 --port=900

* **进入pod**

kubectl exec {pod-name} -it bash

* **查看deploy列表**

kubectl get deployments -n {namespace-name} -o wide

* **删除单个pod**

kubectl delete pods {pod-name}

#### Yaml

* **使用Yaml文件创建pod**

kubectl create -f {yaml-name}.yaml

* **在线查看Yaml语法**

kubectl explain pod

kubectl explain pod.spec | grep -i "containers" -5

* **检验Yarm语法**

kubectl create -f flask.yaml --validate

* **使用Yaml文件删除pod**

kubectl delete -f flask.yaml

## Docker常用命令

### 容器命令

* **查看容器进程**

docker ps|grep {container-name}

* **查看容器日志**

docker logs --tail=500 {container-id}

* **进入容器内部**

docker exec -it {container} /bin/bash

### 镜像命令

* **镜像列表**

docker images

* **导出镜像**

docker save -o {output-path} {repository:tag}

【说明】

output-path：导出到的本地目录

* **导入镜像**

docker load --input {image-tar}

* **修改cgroupdriver**

vi /etc/docker/daemon.json

#添加如下内容

{

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

}

# 国内镜像源

## Azure镜像源

地址统一为：{国内代理地址}/{repo-name}/{image-name}:{version}

| **国外地址** | **国内代理** | **示例** |
| --- | --- | --- |
| docker.io | [dockerhub.azk8s.cn](http://mirror.azk8s.cn/help/docker-registry-proxy-cache.html) | dockerhub.azk8s.cn/library/nginx:1.15 |
| gcr.io | [gcr.azk8s.cn](http://mirror.azk8s.cn/help/gcr-proxy-cache.html) | gcr.azk8s.cn/google\_containers/hyperkube-amd64:v1.13.5 |
| k8s.gcr.io | gcr.azk8s.cn/google-containers |  |
| quay.io | [quay.azk8s.cn](http://mirror.azk8s.cn/help/quay-proxy-cache.html) | quay.azk8s.cn/deis/go-dev:v1.10.0 |

## 阿里云镜像源

### Yum源配置

vi /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>

### Docker源配置

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

{ "registry-mirrors": ["https://xwx6wxd1.mirror.aliyuncs.com"] }

EOF

# 官方文档

## Kubeadm

#### 

## Kubesphray

#### 

## Rancher

#### 官方部署文档

<https://rancher.com/docs/rancher/v1.6/zh/installing-rancher/installing-server/>

https://hub.docker.com/r/rancher/rancher/tags/

#### 其它优秀文档

<https://www.cnblogs.com/whuanle/p/10228999.html>

<https://blog.51cto.com/13941177/2165668>

https://blog.csdn.net/aixiaoyang168/article/details/88600530

# 常见问题



1. Yaml文件格式问题

注意：Yaml文件不允许有Tab！！！只能是空格！！！按照层级，每行4个空格缩进、第二级则8个空格缩进。

1. ‘cp: can't create '/etc/cni/net.d/10-flannel.conf': Permission denied’

setenforce 0

1. Get http://localhost:8080/api?timeout=32s: dial tcp 127.0.0.1:8080: connect: connection refused

mkdir -p $HOME/.kube

sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

sudo chown $(id -u):$(id -g) $HOME/.kube/config

1. The connection to the server localhost:8080 was refused - did you specify the right host or port

scp /etc/kubernetes/admin.conf knode2:/etc/kubernetes/

echo "export KUBECONFIG=/etc/kubernetes/admin.conf" >> ~/.bash\_profile

source ~/.bash\_profile

1. Unable to update cni config: No networks found in /etc/cni/net.d

mkdir -p /etc/cni/net.d/

scp /etc/cni/net.d/10-flannel.conflist knode2:/etc/cni/net.d

1. 加入新节点卡住。原因：证书过期。
2. 生成永久证书

kubeadm token create --ttl 0

1. 生成ca证书sha256编码的hash值

openssl x509 -pubkey -in /etc/kubernetes/pki/ca.crt | openssl rsa -pubin -outform der 2>/dev/null | openssl dgst -sha256 -hex | sed 's/^.\* //'

1. 节点重新加入

kubeadm join 10.110.30.41:6443 --token akrjnq.331e3qsl5vrh9yi2 \

--discovery-token-ca-cert-hash sha256:3a66dbf5d1d4f002484a020330ecb2db00e2a1d91ed42299dace8700b3117712

# 附件

## kuburnetes

参考：https://blog.csdn.net/andriy\_dangli/article/details/79269356





