

# K8s-Install

[https://blog.csdn.net/m0\\_51720581/article/details/131153894](https://blog.csdn.net/m0_51720581/article/details/131153894)

## 环境规划

K8s 版本: 1.23.6

docker 版本: 20.10.0

容器运行时: docker

主机名	IP地址	机器参数
k8smaster	192.168.7.100	2C 2G
k8snode01	192.168.7.101	2C 2G
k8snode02	192.168.7.102	2C 2G

## 主机初始化

```
/etc/selinux/config 14L, 542C Written
[root@localhost ~]# cat /etc/selinux/config | gr
# SELINUX= can take one of these three values:
SELINUX=disabled
# SELINUXTYPE= can take one of three values:
SELINUXTYPE=targeted
[root@localhost ~]# _
```

k8smaster

开启此虚拟机

编辑虚拟机设置

▼ 设备

内存

2 GB

处理器

2

硬盘 (SCSI)

50 GB

CD/DVD (IDE)

自动检测

网络适配器

自定义 (VMnet7)

网络适配器 2

自定义 (VMnet8...

USB 控制器

存在

声卡

自动检测

打印机

存在

显示器

自动检测

▼ 描述

**Edit Connection**

Profile name System ens33  
Device ens33 (00:0C:29:37:27:88)

= **ETHERNET** <Show>

■ **IPv4 CONFIGURATION** <Manual> <Hide>

Addresses 192.168.7.100 <Remove>  
<Add...>

Gateway 192.168.7.1

DNS servers <Add...>

Search domains <Add...>

Routing (No custom routes) <Edit...>

☐ Never use this network for default route

☐ Ignore automatically obtained routes

☐ Ignore automatically obtained DNS parameters

☐ Require IPv4 addressing for this connection

= **IPv6 CONFIGURATION** <Automatic> <Show>

☒ Automatically connect

☒ Available to all users

<Cancel> <OK>

**NetworkManager TUI**

Please select an option

**Set Hostname**

Hostname k8smaster

<Cancel> <OK>

```
[root@localhost ~]#
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:37:27:88 brd ff:ff:ff:ff:ff:ff
    inet 192.168.7.100/24 brd 192.168.7.255 scope global noprefixroute ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe37:2788/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: ens36: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:37:27:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.157.149/24 brd 192.168.157.255 scope global noprefixroute dynamic ens36
        valid_lft 1525sec preferred_lft 1525sec
    inet6 fe80::bdcf:3ffc:b6f6:f1ed/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@localhost ~]# _
```

```
[root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-ens33
# Generated by parse-kickstart
IPV6INIT=yes
IPV6_AUTOCONF=yes
BOOTPROTO=none
DEVICE=ens33
ONBOOT=yes
UUID=37b2a1ab-5757-462d-a5cf-19107edf7420
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME="System ens33"
IPADDR=192.168.7.100
PREFIX=24
GATEWAY=192.168.7.1
[root@localhost ~]# _
```

## 集群初始化



```
新连接 k8smaster - root@192.168.7.100:22 k8snode01 - root@192.168.7.101:22 k8node02 - root@192.168.7.102:22 +
Ssh Sftp
Last login: Mon Sep 11 14:36:11 2023
[root@k8smaster ~]#
[root@k8smaster ~]#
[root@k8smaster ~]# getenforce
Disabled
[root@k8smaster ~]# systemctl disable --now
Too few arguments.
[root@k8smaster ~]# systemctl stop disable --now
Failed to stop disable.service: Unit disable.service not loaded.
[root@k8smaster ~]# systemctl stop firewalld
[root@k8smaster ~]# systemctl status firewalld
• firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
  Active: inactive (dead) since Mon 2023-09-11 14:56:27 CST; 6s ago
    Docs: man:firewalld(1)
  Process: 696 ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS (code=exited, status=0/SUCCESS)
 Main PID: 696 (code=exited, status=0/SUCCESS)

Sep 11 22:35:32 localhost.localdomain systemd[1]: Starting firewalld - dynamic firewall daemon...
Sep 11 22:35:32 localhost.localdomain systemd[1]: Started firewalld - dynamic firewall daemon.
Sep 11 14:56:26 k8smaster systemd[1]: Stopping firewalld - dynamic firewall daemon...
Sep 11 14:56:27 k8smaster systemd[1]: Stopped firewalld - dynamic firewall daemon.
[root@k8smaster ~]# iptables -F
[root@k8smaster ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
[root@k8smaster ~]#
```

```
Lab 画质 x +
新连接 k8smaster - root@192.168.7.100:22 k8snode01 - root@192.168.7.101:22 k8node02 - root@192.168.7.102:22 +
Ssh Sftp
Last login: Mon Sep 11 14:36:00 2023
[root@k8snode01 ~]#
[root@k8snode01 ~]#
[root@k8snode01 ~]# systemctl stop firewalld
[root@k8snode01 ~]# iptables -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
[root@k8snode01 ~]# iptables -F
[root@k8snode01 ~]# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[root@k8snode01 ~]#
```

```
Lab 画质 x +
新连接 x k8smaster - root@192.168.7.100:22 k8snode01 - root@192.168.7.101:22 k8node02 - root@192.168.7.102:22 +
Ssh Sftp
Last login: Mon Sep 11 14:36:19 2023
[root@k8snode02 ~]#
[root@k8snode02 ~]#
[root@k8snode02 ~]# systemctl stop firewalld
[root@k8snode02 ~]# systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[root@k8snode02 ~]#
```

```

Swap: 2.0G 0B 2.0G
[root@k8smaster ~]# vi /etc/fstab
[root@k8smaster ~]# free -h
              total        used        free      shared  buff/cache   available
Mem:           1.8G          134M        1.5G          9.5M        133M        1.5G
Swap:          2.0G           0B        2.0G
[root@k8smaster ~]# swapon -a
[root@k8smaster ~]# free -h
              total        used        free      shared  buff/cache   available
Mem:           1.8G          134M        1.5G          9.5M        133M        1.5G
Swap:           0B           0B         0B
[root@k8smaster ~]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Sep 11 22:19:06 2023
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/centos-root /                xfs     defaults        0 0
UUID=4bb29f43-0623-4c79-a29d-da295c1242c4 /boot  xfs     defaults        0 0
#/dev/mapper/centos-swap swap    swap    defaults        0 0
[root@k8smaster ~]#

```

```

[root@k8smaster ~]# sudo sed -e 's|^mirrorlist=|#mirrorlist=|g' \
> -e 's|^#baseurl=http://mirror.centos.org/centos|baseurl=https://mirrors.ustc.edu.cn/centos|g' \
> -i.bak \
> /etc/yum.repos.d/CentOS-Base.repo
[root@k8smaster ~]# yum makecache
Loaded plugins: fastestmirror
Determining fastest mirrors
base
extras
updates
(1/10): base/7/x86_64/group_gz
(2/10): base/7/x86_64/filelists_db
(3/10): base/7/x86_64/other_db
(4/10): base/7/x86_64/primary_db
(5/10): extras/7/x86_64/filelists_db
(6/10): extras/7/x86_64/primary_db
(7/10): extras/7/x86_64/other_db
(8/10): updates/7/x86_64/filelists_db
(9/10): updates/7/x86_64/other_db
(10/10): updates/7/x86_64/primary_db
Metadata Cache Created
[root@k8smaster ~]#

```

```

[root@k8smaster ~]#
[root@k8smaster ~]# cat /etc/host
cat: /etc/host: No such file or directory
[root@k8smaster ~]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6

# k8s
192.168.7.100 k8smaster
192.168.7.101 k8snode01
192.168.7.102 k8snode02
[root@k8smaster ~]#

```



```

[-W timeout] destination
[root@k8smaster ~]# ping -c 2 k8smaster
PING k8smaster (192.168.7.100) 56(84) bytes of data.
64 bytes from k8smaster (192.168.7.100): icmp_seq=1 ttl=64 time=0.011 ms
64 bytes from k8smaster (192.168.7.100): icmp_seq=2 ttl=64 time=0.056 ms

--- k8smaster ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.011/0.033/0.056/0.023 ms
[root@k8smaster ~]# ping -c 2 k8snode01
PING k8snode01 (192.168.7.101) 56(84) bytes of data.
64 bytes from k8snode01 (192.168.7.101): icmp_seq=1 ttl=64 time=0.112 ms
64 bytes from k8snode01 (192.168.7.101): icmp_seq=2 ttl=64 time=0.131 ms

--- k8snode01 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.112/0.121/0.131/0.014 ms
[root@k8smaster ~]# ping -c 2 k8snode02
PING k8snode02 (192.168.7.102) 56(84) bytes of data.
64 bytes from k8snode02 (192.168.7.102): icmp_seq=1 ttl=64 time=14.9 ms
64 bytes from k8snode02 (192.168.7.102): icmp_seq=2 ttl=64 time=0.182 ms

--- k8snode02 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.182/7.561/14.940/7.379 ms
[root@k8smaster ~]#

```

```

[root@k8smaster ~]# scp /etc/modules-load.d/k8s.conf root@k8snode01:/etc/mod
modprobe.d/ modules-load.d/ motd
[root@k8smaster ~]# scp /etc/modules-load.d/k8s.conf root@k8snode01:/etc/mod
modprobe.d/ modules-load.d/ motd
[root@k8smaster ~]# scp /etc/modules-load.d/k8s.conf root@k8snode01:/etc/modules-load.d/k8s.conf
The authenticity of host 'k8snode01 (192.168.7.101)' can't be established.
ECDSA key fingerprint is SHA256:PeMwAfvdyJmAcF1VwM6Y9XZ7GNx10a/mTzVyAXAX6fY.
ECDSA key fingerprint is MD5:54:63:38:be:0b:45:25:c1:72:34:a5:7c:dd:c0:1b:08.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'k8snode01,192.168.7.101' (ECDSA) to the list of known hosts.
root@k8snode01's password:
k8s.conf
[root@k8smaster ~]# scp /etc/modules-load.d/k8s.conf root@k8snode02:/etc/modules-load.d/k8s.conf
The authenticity of host 'k8snode02 (192.168.7.102)' can't be established.
ECDSA key fingerprint is SHA256:pFbuvNYuWu9pnetjVZVpBrcN0Q1/sXpI7gRgIpe3+yo.
ECDSA key fingerprint is MD5:b0:17:e2:01:f2:e2:5d:97:70:b8:d4:57:74:75:35:31.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'k8snode02,192.168.7.102' (ECDSA) to the list of known hosts.
root@k8snode02's password:
k8s.conf
[root@k8smaster ~]#

```

```

sysctl: /etc/sysctl.conf(17): invalid syntax, continuing...
[root@k8snode02 ~]# vim /etc/sysctl.conf
[root@k8snode02 ~]# sysctl -p
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_nonlocal_bind = 1
vm.swappiness = 0
[root@k8snode02 ~]#

```

## 安装docker

```

49 sudo yum install -y yum-utils device-mapper-persistent-data lvm2
50 sudo yum-config-manager --add-repo https://mirrors.aliyun.com/docker-
ce/linux/centos/docker-ce.repo
51 sudo sed -i 's+download.docker.com+mirrors.aliyun.com/docker-ce+'
/etc/yum.repos.d/docker-ce.repo
52 yum makecache fast
53 yum install -y docker-ce-20.10.0 docker-ce-cli-20.10.0 containerd.io
54 history

```

```

RETURN    all -- 0.0.0.0/0      0.0.0.0/0
[root@k8smaster ~]# vim /etc/docker/daemon.json
[root@k8smaster ~]# cat /etc/docker/daemon.json
{
    "registry-mirrors": [
        "http://hub-mirror.c.163.com",
        "https://docker.mirrors.ustc.edu.cn",
        "https://registry.docker-cn.com"
    ],
    "exec-opts": ["native.cgroupdriver=systemd"]
}
[root@k8smaster ~]# systemctl restart docker
[root@k8smaster ~]# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2023-09-11 15:42:29 CST; 4s ago
     Docs: https://docs.docker.com
    Main PID: 12094 (dockerd)
      Tasks: 9
     Memory: 42.6M
    CGroup: /system.slice/docker.service
            └─12094 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.673042854+08:00" level=info msg="ccResolverWrapper: sending new grpc request to localhost:2376"
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.673047750+08:00" level=info msg="ClientConn switc
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.678961424+08:00" level=info msg="[graphdriver] using prior storage driver: overlay2"
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.679911597+08:00" level=info msg="Loading container
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.732205969+08:00" level=info msg="Default bridge
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.754387197+08:00" level=info msg="Loading container
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.763031393+08:00" level=info msg="Docker daemon
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.763062672+08:00" level=info msg="Daemon has co
Sep 11 15:42:29 k8smaster systemd[1]: Started Docker Application Container Engine.
Sep 11 15:42:29 k8smaster dockerd[12094]: time="2023-09-11T15:42:29.775331002+08:00" level=info msg="API listen on
Hint: Some lines were ellipsized, use -l to show in full.
[root@k8smaster ~]# █

```

## K8S instll

```

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

```

```

yum install -y kubelet-1.23.6 kubeadm-1.23.6 kubectl-1.23.6

```

# 拉取镜像

```

docker pull coredns/coredns:1.8.4

```

# 将镜像改名

```

docker tag coredns/coredns:1.8.4

```

```

registry.aliyuncs.com/google_containers/coredns:v1.8.4

```

## Master 集群初始化

```
# 记得改变IP, 只要改第一行的IP地址, 一般改为master节点地址
kubeadm init \
--apiserver-advertise-address=192.168.7.100 \
--image-repository registry.aliyuncs.com/google_containers \
--service-cidr=10.1.0.0/16 \
--pod-network-cidr=10.244.0.0/16
#--pod-network-cidr=192.168.0.0/16
```

```
[root@k8smaster ~]#
[root@k8smaster ~]# kubeadm init \
> --apiserver-advertise-address=192.168.7.100 \
> --image-repository registry.aliyuncs.com/google_containers \
> --service-cidr=10.1.0.0/16 \
> --pod-network-cidr=10.244.0.0/16
I0911 16:14:56.720135 12438 version.go:255] remote version is much newer: v1.28.1; falling back to: stable-1.23
[init] Using Kubernetes version: v1.23.17
[preflight] Running pre-flight checks
[WARNING Service-Docker]: docker service is not enabled, please run 'systemctl enable docker.service'
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action in beforehand using 'kubeadm config images pull'
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [k8smaster kubernetes kubernetes.default kubernetes.default.svc k
[certs] Generating "apiserver-kubelet-client" certificate and key
```

```
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

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

Alternatively, if you are the root user, you can run:

export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 192.168.7.100:6443 --token g63wqp.lwnjze8i4hlvipcd \
--discovery-token-ca-cert-hash sha256:6cad01de848e791531d57e04a056bf41ebbb51f172b713e4a9af23bf3a034da
[root@k8smaster ~]#
```

```
--discovery-token-ca-cert-hash sha256:6cad01de848e791531d57e04a056bf41ebbb51f172b713e4a9af23bf3a034da
ot@k8smaster ~]# mkdir -p $HOME/.kube
ot@k8smaster ~]# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ot@k8smaster ~]# sudo chown $(id -u):$(id -g) $HOME/.kube/config
ot@k8smaster ~]#
```

Your Kubernetes control-plane has initialized successfully!

To **start** using your cluster, you need to run the following as a regular user:

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

Alternatively, if you are the root user, you can run:

```
export KUBECONFIG=/etc/kubernetes/admin.conf
```

You should now deploy a pod network to the cluster.

Run "**kubectl apply -f [podnetwork].yaml**" with one of the options listed at:

<https://kubernetes.io/docs/concepts/cluster-administration/addons/>

Then you can join any number of worker nodes by running the following on each as root:



```
kubeadm join 192.168.7.100:6443 --token g63wpx.lwnjze8i4hlvipcd \
--discovery-token-ca-cert-hash
sha256:6cad01de848e791531d57e04a056bf41ebbb51f172b713e4a9af23bf3a034da
```

## node 主机设置

```
coredns/coredns:1.8.4 Complete
Digest: sha256:6e5a02c21641597998b4be7cb5eb1e7b02c0d8d23cce4dd09f4682d463798890
Status: Downloaded newer image for coredns/coredns:1.8.4
docker.io/coredns/coredns:1.8.4
[root@k8snode01 ~]# docker tag coredns/coredns:1.8.4 registry.aliyuncs.com/google_containers/coredns:v1.8.4
[root@k8snode01 ~]# kubeadm join 192.168.7.100:6443 --token g63wpx.lwnjze8i4hlvipcd \
--discovery-token-ca-cert-hash sha256:6cad01de848e791531d57e04a056bf41ebbb51f172b713e4a9af23bf3a034da
[preflight] Running pre-flight checks
[WARNING Service-Docker]: docker service is not enabled, please run 'systemctl enable docker.service'
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...

This node has joined the cluster:
* Certificate signing request was sent to apiserer and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

[root@k8snode01 ~]#
```

## 完成

```
kube-flannel-ds-zz5cl 1/1 Running 0 5m2s
^C[root@k8smaster ~]# kubectl get pods -n kube-flannel --watch 3
Error from server (NotFound): pods "3" not found
[root@k8smaster ~]# kubectl get pods -n kube-flannel --watch -n3
^C[root@k8smaster ~]# kubectl get pods -n kube-flannel --watch 3
Error from server (NotFound): pods "3" not found
[root@k8smaster ~]# kubectl get pods -n kube-flannel --watch
NAME READY STATUS RESTARTS AGE
kube-flannel-ds-2bf4k 1/1 Running 0 4m11s
kube-flannel-ds-gpnpx 1/1 Running 0 4m6s
kube-flannel-ds-zz5cl 1/1 Running 0 5m21s
^C[root@k8smaster ~]# kubectl get pods -n kube-flannel --watch
NAME READY STATUS RESTARTS AGE
kube-flannel-ds-2bf4k 1/1 Running 0 4m20s
kube-flannel-ds-gpnpx 1/1 Running 0 4m15s
kube-flannel-ds-zz5cl 1/1 Running 0 5m30s
^C[root@k8smaster ~]# kubectl get node
NAME STATUS ROLES AGE VERSION
k8smaster Ready control-plane,master 11m v1.23.6
k8snode01 Ready <none> 4m31s v1.23.6
k8snode02 Ready <none> 4m26s v1.23.6
[root@k8smaster ~]#
```

```
k8snode02 Ready <none> 4m57s v1.23.6
[root@k8smaster ~]# kubectl create deployment k8s-nginx --image=nginx -r 3
deployment.apps/k8s-nginx created
[root@k8smaster ~]# kubectl get pod -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
k8s-nginx-6d779d947c-59qh8 0/1 ContainerCreating 0 16s <none> k8snode01 <none> <none>
k8s-nginx-6d779d947c-djzx7 0/1 ContainerCreating 0 16s <none> k8snode02 <none> <none>
k8s-nginx-6d779d947c-wxhx2 0/1 ContainerCreating 0 16s <none> k8snode02 <none> <none>
[root@k8smaster ~]#
```

## 其他

### 命令补全

```
yum -y install bash-completion
source /usr/share/bash-completion/bash_completion
source <(kubectl completion bash)
echo "source <(kubectl completion bash)" >> ~/.bashrc
```

## NGINX 主机设置 VPN设置

```
1 vi /etc/selinux/config
2 init 0
3 nmtui
4 exit
5 ip a
6 cd /opt/
7 ls
8 mkdir pritunl-client
9 cd pritunl-client/
10 yum install -y wget
11 wget https://note.heike07.cn/key/445bc78904f944ea8326527ed9b88424.tar
12 ls
13 sudo tee -a /etc/yum.repos.d/pritunl.repo << EOF
[pritunl]
name=Pritunl Stable Repository
baseurl=https://repo.pritunl.com/stable/yum/centos/8/
gpgcheck=1
enabled=1
EOF

14 cd /etc/yum.repos.d/
15 ls
16 ll
17 cd pritunl.repo
18 yum install -y vim
19 vi pritunl.repo
20 yum install -y vim
21 gpg --keyserver hkp://keyserver.ubuntu.com --recv-keys
7568D9BB55FF9E5287D586017AE645C0CF8E292A
22 gpg --armor --export 7568D9BB55FF9E5287D586017AE645C0CF8E292A > key.tmp;
sudo rpm --import key.tmp; rm -f key.tmp
23 yum install pritunl-client
24 yum install pritunl-client Error: Package: pritunl-openvpn-2.5.8-
1.e17.centos.x86_64 (pritunl)
25 Requires: libpkcs11-helper.so.1()(64bit)
26 You could try using --skip-broken to work around the problem
27 You could try running: rpm -Va --nofiles --nodigest
28 yum install libpkcs11-helper
29 yum install libpkcs11
30 yum install openssl
31 yum install pritunl-client
32 yum install libp11
33 yum remove openssl
34 yum install -y epel-release
35 yum install -y openssl-devel openssl11 openssl11-devel
36 yum install pritunl-client
37 cd /opt/pritunl-client/
38 ls
39 pritunl-client add 445bc78904f944ea8326527ed9b88424.tar
```

```

40 pritunl-client list
41 pritunl-client start cj9gz4mb5jxjpkwx
42 pritunl-client list
43 firewall-cmd --list-all
44 systemctl stop firewalld
45 systemctl disabled firewalld
46 systemctl disable firewalld
47 iptables -L -n
48 pritunl-client list
49 pritunl-client stop cj9gz4mb5jxjpkw
50 pritunl-client list
51 curl heike07.cn
52 curl note.heike07.cn
53 pritunl-client list
54 pritunl-client start cj9gz4mb5jxjpkwx
55 pritunl-client list
56 cd /var/log/
57 ls
58 ll
59 date
60 tail pritunl-client.log
61 tail pritunl-client.log.1
62 ls
63 ll
64 vim pritunl-client.log
65 tail pritunl-client.log
66 ss -lntp | grep openvpn
67 top
68* ss -
69 pritunl-client list
70 ls
71 ll
72 cd /opt/pritunl-client/
73 ls
74 ll
75 vim 445bc78904f944ea8326527ed9b88424.tar
76 pritunl-client list
77 rpm -qa | grep openvpn
78 rpm -qa | grep pritunl
79 yum remove pritunl-openvpn-2.5.8-1.el7.centos.x86_64
80 yum install openvpn
81 yum install pritunl-openvpn
82 yum install pritunl-client-1.3.3484
83 yum install pritunl-client
84 rpm -qa | grep pritunl
85 pritunl-client list
86 ping 192.168.239.10
87 ip a
88 history

```

## nginx steam 模块加载代理

```

142 history |grep configure
143 ./configure --with-http_v2_module --with-http_ssl_module --with-
http_sub_module --with-http_stub_status_module --with-http_gzip_static_module --
with-pcre --with-stream --with-stream=dynamic --with-stream_ssl_module --with-
stream_realip_module

```

```
[root@nginx conf]# cat nginx.conf |grep load
load_module /usr/local/nginx/modules/ngx_stream_module.so;
[root@nginx conf]#

[root@nginx sbin]# ./nginx -v
nginx version: nginx/1.24.0
built by gcc 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC)
built with OpenSSL 1.0.2k-fips 26 Jan 2017
TLS SNI support enabled
configure arguments: --with-http_v2_module --with-http_ssl_module --with-
http_sub_module --with-http_stub_status_module --with-http_gzip_static_module --
with-pcre --with-stream --with-stream=dynamic --with-stream_ssl_module --with-
stream_realip_module
[root@nginx sbin]#
```