



链滴

# Kubernetes 、 Dashboard 配置安装

作者: [450370050](#)

原文链接: <https://ld246.com/article/1537167018376>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Kubernetes集群组件:

- etcd 一个高可用的K/V键值对存储和服务发现系统
  - flannel 实现跨主机的容器网络的通信
  - kube-apiserver 提供kubernetes集群的API调用
  - kube-controller-manager 确保集群服务
  - kube-scheduler 调度容器, 分配到Node
  - kubelet 在Node节点上按照配置文件中定义的容器规格启动容器
  - kube-proxy 提供网络代理服务
- 

## 安装

### 软件更新

#### 1.机器分配

节点	ip
Master	192.168.139.148
Node1	192.168.139.149
Node2	192.168.139.150

#### 2.关闭CentOS7自带的防火墙服务

#### 3.系统初始化安装 (所有主机) -选择【最小化安装】,然后yum update,升级到最新版本

```
yum -y install epel-release  
yum update
```

#### 4.更新本地时间

```
systemctl start ntpd  
systemctl enable ntpd  
ntpdate ntp1.aliyun.com  
hwclock -w
```

## Master节点配置

### 安装软件

```
[root@master ~]#  
yum install -y etcd kubernetes-master ntp flannel
```

### 配置etcd服务器

```
[root@master ~]# vi /etc/etcd/etcd.conf
ETCD_NAME=default
ETCD_DATA_DIR="/var/lib/etcd/default.etcd"
ETCD_LISTEN_CLIENT_URLS="http://localhost:2379,http://192.168.139.148:2379"
ETCD_ADVERTISE_CLIENT_URLS="http://192.168.139.148:2379"
```

## 启动服务

```
systemctl start etcd
systemctl enable etcd
```

## 检查etcd cluster状态

```
[root@master ~]# etcdctl cluster-health
```

```
member 8e9e05c52164694d is healthy: got healthy result from http://192.168.139.148:2379
cluster is healthy
```

## 检查etcd集群成员列表，这次只配置了一台

```
[root@master ~]# etcdctl member list
```

```
8e9e05c52164694d: name=default peerURLs=http://localhost:2380 clientURLs=http://192.168.139.148:2379 isLeader=true
```

## 配置kube-apiserver

### 修改kube-apiserver bind-address

```
[root@master ~]# vi /etc/kubernetes/apiserver
KUBE_API_ADDRESS="--insecure-bind-address=0.0.0.0"
```

## 配置kube-scheduler

### 修改address

```
[root@master ~]# vi /etc/kubernetes/scheduler
KUBE_SCHEDULER_ARGS="--address=0.0.0.0"
```

## 启动服务

```
for i in kube-apiserver kube-controller-manager kube-scheduler;do systemctl restart $i; systemctl enable $i;done
```

## Node节点配置

配置node1网络，本实例采用flannel方式来配置，如需其他方式，请参考Kubernetes官网。

## 配置flannel

```
[root@node1 ~]# vi /etc/sysconfig/flanneld
FLANNEL_ETCD_ENDPOINTS="http://192.168.139.148:2379"
```

```
FLANNEL_ETCD_PREFIX="/atomic.io/network"
FLANNEL_OPTIONS=""
```

## 配置kube-proxy

修改master的apiserver地址

```
[root@node1 ~]# vi /etc/kubernetes/config
KUBE_MASTER="--master=http://192.168.139.148:8080"
```

修改kube-proxy监听

```
[root@node1 ~]# vi /etc/kubernetes/proxy
KUBE_PROXY_ARGS="--bind=address=0.0.0.0"
```

## 配置kubelet

修改hostname-override

修改api-server地址

```
[root@node1 ~]# vi /etc/kubernetes/kubelet
KUBELET_ADDRESS="--address=127.0.0.1"
KUBELET_HOSTNAME="--hostname-override=node1"
KUBELET_API_SERVER="--api-servers=http://192.168.139.148:8080"
```

## 启动node

```
for i in flanneld kube-proxy kubelet docker;do systemctl restart $i;systemctl enable $i;systemctl status $i ;done
```

## 配置node2节点

node2节点和node1配置一致，kubelet配置中hostname-override改为node2即可。

## 环境检查

### flannel信息

```
[root@master ~]# etcdctl ls /atomic.io/network/subnets
```

```
/atomic.io/network/subnets/172.16.61.0-24
/atomic.io/network/subnets/172.16.45.0-24
```

```
[root@master ~]# etcdctl get /atomic.io/network/subnets/172.16.61.0-24
{"PublicIP":"192.168.139.150"}
```

```
[root@master ~]# etcdctl get /atomic.io/network/subnets/172.16.45.0-24
{"PublicIP":"192.168.139.149"}
```

## 查看节点

```
[root@master ~]# kubectl get nodes
```

```
NAME     STATUS   AGE
node1    Ready    2h
node2    Ready    2h
```

---

## DashBoard安装

### 1.查看k8s版本

```
[root@master ~]# kubectl version
```

```
Client Version: version.Info{Major:"1", Minor:"5", GitVersion:"v1.5.2", GitCommit:"269f92821797e7126dc87e6adfa82242bfe5b1e", GitTreeState:"clean", BuildDate:"2017-07-03T15:31:10Z", oVersion:"go1.7.4", Compiler:"gc", Platform:"linux/amd64"}
Server Version: version.Info{Major:"1", Minor:"5", GitVersion:"v1.5.2", GitCommit:"269f92821757e7126dc87e6adfa82242bfe5b1e", GitTreeState:"clean", BuildDate:"2017-07-03T15:31:10Z", oVersion:"go1.7.4", Compiler:"gc", Platform:"linux/amd64"}
```

### 2.我们的k8s版本为1.5.2 为防止不兼容我们使用dashboard1.5.1的配置

```
wget https://raw.githubusercontent.com/kubernetes/dashboard/v1.5.1/src/deploy/kubernetes-dashboard.yaml
```

### 3.修改下载的dashboard配置

dashboard部署配置中使用的默认镜像需要翻墙，我们修改为阿里云的镜像。

```
image: registry.cn-hangzhou.aliyuncs.com/kube_containers/kubernetes-dashboard-amd64:v1.5.1
```

配置dashboard的api-server地址

args:

```
# Uncomment the following line to manually specify Kubernetes API server Host
# If not specified, Dashboard will attempt to auto discover the API server and connect
# to it. Uncomment only if the default does not work.
- --apiserver-host=http://192.168.139.148:8080
```

修改节点容器暴露的端口

```
- port: 80
  nodePort: 30081
  targetPort: 9090
```

### 4.部署dashboard

```
kubectl apply -f kubernetes-dashboard.yaml
```

### 5.查看pod部署状态

```
kubectl describe pods --all-namespaces
```

如果出现错误, 可以查看日志中的错误 `cat /var/log/message`

创建成功, 但是 `kubectl get pods` 没有结果

报错信息: `no API token found for service account default`

解决办法: 编辑 `/etc/kubernetes/apiserver` 去除 `KUBE_ADMISSION_CONTROL` 中的 `SecurityContextDeny,ServiceAccount`, 并重启 `kube-apiserver.service` 服务

`pod-infrastructure:latest` 镜像下载失败

报错信息: `image pull failed for registry.access.redhat.com/rhel7/pod-infrastructure:latest, this may be because there are no credentials on this request.`

解决方案:

```
wget http://mirror.centos.org/centos/7/os/x86_64/Packages/python-rhsm-certificates-1.19.10-1.el7_4.x86_64.rpm
rpm2cpio python-rhsm-certificates-1.19.10-1.el7_4.x86_64.rpm | cpio -iv --to-stdout ./etc/rhsc
/ca/redhat-uep.pem | tee /etc/rhsm/ca/redhat-uep.pem
```

## 检查服务

1.

```
[root@master k8s-yaml]# kubectl get deploy --all-namespaces
```

```
NAMESPACE  NAME                DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
kube-system  kubernetes-dashboard  2        2        2           2          1h
```

```
[root@master k8s-yaml]# kubectl get pods --all-namespaces
```

```
NAMESPACE  NAME                READY  STATUS  RESTARTS  AGE
kube-system  kubernetes-dashboard-3350179127-8scg1  1/1    Running  0         1h
kube-system  kubernetes-dashboard-3350179127-gfvg0  1/1    Running  0         1h
```

```
[root@master k8s-yaml]# kubectl get services --all-namespaces
```

```
NAMESPACE  NAME          CLUSTER-IP  EXTERNAL-IP  PORT(S)    AGE
default     kubernetes    10.254.0.1  <none>       443/TCP    3h
kube-system  kubernetes-dashboard  10.254.12.166  <nodes>     80:30081/TCP  1h
```

2. 由于k8s的的iptables外网并无法直接访问我们监听的端口, 需要允许转发

```
iptables -P FORWARD ACCEPT
```

3. 访问服务

**kubernetes** Workloads

Admin

- Namespaces
- Nodes
- Persistent Volumes

Namespace

kube-system

Workloads

- Deployments
- Replica Sets
- Replication Controllers
- Daemon Sets
- Stateful Sets
- Jobs
- Pods

Services and discovery

- Services
- Ingresses

Storage

### Deployments

Name	Labels	Pods	Age	Images
✓ <a href="#">kubernetes-dashboard</a>	app: kubernetes-dashboard	2 / 2	an hour	registry.cn-hangzhou.aliyuncs.com/kube_cont

### Replica Sets

Name	Labels	Pods	Age	Images
✓ <a href="#">kubernetes-dashboar-3350179127</a>	app: kubernetes-dashboar pod-template-hash: 3350179127	2 / 2	an hour	registry.cn-hangzhou.aliyuncs.com/kube_cont

### Pods

Name	Status	Restarts	Age
✓ <a href="#">kubernetes-dashboar-3350179127-8scg1</a>	Running	0	an hou
✓ <a href="#">kubernetes-dashboar-3350179127-gfyg0</a>	Running	0	an hou