**K8s etcd集群部署**

etcd集群及k8s相关组件的版本说明：

|  |  |
| --- | --- |
| Name | Version |
| etcd | etcd Version: 3.2.18  Git SHA: eddf599c6 |
| Etcdctl | etcdctl version: 3.2.18  API version: 2 |
| cfssl | Version: 1.2.0  Revision: dev  Runtime: go1.6 |
| calico | v3.1.0 |
| hyperkube | v1.10.13 |
| docker | 18.03.0 |

etcd集群各节点部署前，需要按照k8s集群部署要求进行检查。

1. 需下载安装初始化工具。

net-tools vim wget lrzsz git cfssl etcdctl etcd cfssljson

1. 关闭防火墙、selinx。
2. 优化内核。
3. 保证网络正常通讯。
4. 保证网卡、路由配置正确。
5. 保证时区和时间同步正常。
6. 保证操作系统版本符合要求。
7. 修改etcd集群各节点的主机名/etc/hosts。
8. 保证ssh设置免密互通（k8s集群内所有主机设置免密）。
9. 更新yum源。
10. etcd节点选择算法说明。

所采用Raft的保证分布式系统强一致性的算法。由于要做节点选举，所以节点选择单数节点，通常选择3个节点，如果要增加，每次增加两个节点。

1. 服务器的相关节点部署拓扑。

|  |  |  |  |
| --- | --- | --- | --- |
| Ip地址 | 用途 | 系统版本 | 备注 |
| 10.129.52.142 | 主节点 | Centos7.2 | 以142为例，其他节点复制142的生成证书和二进制文件 |
| 10.129.52.162 | 节点 | Centos7.2 |  |
| 10.129.52.194 | 节点 | Centos7.2 |  |
| [root@etcd2 etcd-10230202]# etcdctl --version  etcdctl version: 3.2.18  API version: 2 | | | |
| [root@etcd2 etcd-10230202]# etcd --version  etcd Version: 3.2.18  Git SHA: eddf599c6  Go Version: go1.8.7  Go OS/Arch: linux/amd64 | | | |
| [root@etcd2 etcd-10230202]# cfssl version  Version: 1.2.0  Revision: dev  Runtime: go1.6 | | | |

1. ssh主节点10.129.52.142免密配置。

expect -c "

spawn ssh-copy-id -i /root/.ssh/id\_rsa.pub root@$i

expect {

\"\*yes/no\*\" {send \"yes\r\"; exp\_continue}

\"\*password\*\" {send \"$mypass\r\"; exp\_continue}

\"\*Password\*\" {send \"$mypass\r\";}

}"

1. 时区设置。

设置所有节点的时区为Shanghai

\cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime -rf

##时区设置和基础包安装，关闭防火墙selinux，ntp时间同步。

ssh root@$i "yum install -y ntpdate net-tools vim wget lrzsz git telnet bash-completion;ntpdate -u 10.129.80.80;systemctl stop firewalld;systemctl disable firewalld;sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config;cp /usr/share/zoneinfo/Asia/Shanghai /etc/localtime -rf;reboot"

1. 主节点10.129.52.142的相关操作步骤。

|  |  |  |
| --- | --- | --- |
| etcd-ca-csr.json | /k8sInstall/etcdClusterInstall  /config/ssl/ | ca 证书请求文件 |
| etcd-config.json | /k8sInstall/etcdClusterInstall  /config/ssl/ | ca 证书文件 |
| etcd-server-csr.json | /k8sInstall/etcdClusterInstall  /config/ssl/ | etcd 证书请求 |

1. ca证书请求文件放置位置路径为：

/k8sInstall/etcdClusterInstall/config/ssl/

etcd-ca-csr.json

{

"key": {

"algo": "rsa",

"size": 4096

},

"names": [

{

"O": "etcd",

"OU": "etcd Security",

"L": "Beijing",

"ST": "Beijing",

"C": "CN"

}

],

"CN": "etcd-ca"

}

1. 依据生成证书文件。

1）依据/k8sInstall/etcdClusterInstall/config/ssl/etcd-ca-csr.json文件生成证书。

cfssl gencert --initca=true etcd-ca-csr.json | cfssljson --bare etcd-ca

生成证书地址/k8sInstall/etcdClusterInstall/config/ssl/output/如下：

etcd-ca.csr

etcd-ca.pem

etcd-ca-key.pem

1. 依据/k8sInstall/etcdClusterInstall/config/ssl/etcd-config.json、etcd-csr.json和之前生成的CA证书文件etcd-ca.pem、etcd-ca-key.pem共同生成etcd server端的证书。

cfssl gencert --ca /k8sInstall/etcdClusterInstall/config/ssl/output/etcd-ca.pem --ca-key

/k8sInstall/etcdClusterInstall/config/ssl/output/etcd-ca-key.pem --config /k8sInstall/etcdClusterInstall/config/ssl/etcd-config.json kubernetes /k8sInstall/etcdClusterInstall/config/ssl/etcd-server-csr.json | cfssljson --bare etcd-server

生成证书地址/k8sInstall/etcdClusterInstall/config/ssl/output/如下：

etcd-server.csr

etcd-server.pem

etcd-server-key.pem

1. 配置etcd.conf配置文件

根据etcd所在的服务器节点不同修改三个服务器的配置文件。

具体配置项说明：

|  |  |  |
| --- | --- | --- |
| ETCD\_NAME | 节点名称，集群中唯一 | etcd1/2/3 |
| ETCD\_DATA\_DIR | 数据目录 | /var/lib/etcd/etcd1.etcd  /var/lib/etcd/etcd2.etcd  /var/lib/etcd/etcd3.etcd |
| ETCD\_LISTEN\_PEER\_URLS | 集群通信监听地址 | https://10.129.52.142:2380 |
| ETCD\_LISTEN\_CLIENT\_URLS | 客户端访问监听地址 | https://10.129.52。142:2379,http://127.0.0.1:2379 |
| ETCD\_INITIAL\_ADVERTISE\_PEER\_URLS | 集群通告地址 | https://10.129.52.142:2380 |
| ETCD\_ADVERTISE\_CLIENT\_URLS | 客户端通告地址 | https://10.129.52.142:2379 |
| ETCD\_INITIAL\_CLUSTER | 集群节点地址 | etcd1=https://10.129.52.142:2380,etcd2=https://10.129.52.162:2380,etcd3=https://10.129.52.194:2380 |
| ETCD\_INITIAL\_CLUSTER\_TOKEN | 集群 Token | etcd-cluster |
| ETCD\_INITIAL\_CLUSTER\_STATE | 加入集群的当前状态，new 是新集群，existing 表示加入已有集群 | new |

以服务器10.129.52.142上的etcd.conf样例文件为例：

# [member]

ETCD\_NAME=etcd1

ETCD\_DATA\_DIR="/var/lib/etcd/etcd1.etcd"

ETCD\_WAL\_DIR="/var/lib/etcd/wal"

ETCD\_SNAPSHOT\_COUNT="100"

ETCD\_HEARTBEAT\_INTERVAL="100"

ETCD\_ELECTION\_TIMEOUT="1000"

ETCD\_LISTEN\_PEER\_URLS="https://10.129.52.142:2380"

ETCD\_LISTEN\_CLIENT\_URLS="https://10.129.52.142:2379,http://127.0.0.1:2379"

ETCD\_MAX\_SNAPSHOTS="5"

ETCD\_MAX\_WALS="5"

# [cluster]

ETCD\_INITIAL\_ADVERTISE\_PEER\_URLS="https://10.129.52.142:2380"

ETCD\_INITIAL\_CLUSTER="etcd1=https://10.129.52.142:2380,etcd2=https://10.129.52.162:2380,etcd3=https://10.129.52.194:2380"

ETCD\_INITIAL\_CLUSTER\_STATE="new"

ETCD\_INITIAL\_CLUSTER\_TOKEN="etcd-cluster"

ETCD\_ADVERTISE\_CLIENT\_URLS="https://10.129.52.142:2379"

# [security]

ETCD\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_CLIENT\_CERT\_AUTH="true"

ETCD\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_AUTO\_TLS="true"

ETCD\_PEER\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_PEER\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_PEER\_CLIENT\_CERT\_AUTH="true"

ETCD\_PEER\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_PEER\_AUTO\_TLS="true"

配置10.129.52.162同上：

# [member]

ETCD\_NAME=etcd2

ETCD\_DATA\_DIR="/var/lib/etcd/etcd2.etcd"

ETCD\_WAL\_DIR="/var/lib/etcd/wal"

ETCD\_SNAPSHOT\_COUNT="100"

ETCD\_HEARTBEAT\_INTERVAL="100"

ETCD\_ELECTION\_TIMEOUT="1000"

ETCD\_LISTEN\_PEER\_URLS="https://10.129.52.162:2380"

ETCD\_LISTEN\_CLIENT\_URLS="https://10.129.52.162:2379,http://127.0.0.1:2379"

ETCD\_MAX\_SNAPSHOTS="5"

ETCD\_MAX\_WALS="5"

#ETCD\_CORS=""

# [cluster]

ETCD\_INITIAL\_ADVERTISE\_PEER\_URLS="https://10.129.52.162:2380"

# if you use different ETCD\_NAME (e.g. test), set ETCD\_INITIAL\_CLUSTER value for this name, i.e. "test=http://..."

ETCD\_INITIAL\_CLUSTER="etcd2=https://10.129.52.142:2380,etcd2=https://10.129.52.162:2380,etcd3=https://10.129.52.194:2380"

ETCD\_INITIAL\_CLUSTER\_STATE="new"

ETCD\_INITIAL\_CLUSTER\_TOKEN="etcd-cluster"

ETCD\_ADVERTISE\_CLIENT\_URLS="https://10.129.52.162:2379"

# [security]

ETCD\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_CLIENT\_CERT\_AUTH="true"

ETCD\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_AUTO\_TLS="true"

ETCD\_PEER\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_PEER\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_PEER\_CLIENT\_CERT\_AUTH="true"

ETCD\_PEER\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_PEER\_AUTO\_TLS="true"

# [logging]

#ETCD\_DEBUG="false"

# examples for -log-package-levels etcdserver=WARNING,security=DEBUG

#ETCD\_LOG\_PACKAGE\_LEVELS=""

配置10.129.52.194同上：

# [member]

ETCD\_NAME=etcd3

ETCD\_DATA\_DIR="/var/lib/etcd/etcd3.etcd"

ETCD\_WAL\_DIR="/var/lib/etcd/wal"

ETCD\_SNAPSHOT\_COUNT="100"

ETCD\_HEARTBEAT\_INTERVAL="100"

ETCD\_ELECTION\_TIMEOUT="1000"

ETCD\_LISTEN\_PEER\_URLS="https://10.129.52.194:2380"

ETCD\_LISTEN\_CLIENT\_URLS="https://10.129.52.194:2379,http://127.0.0.1:2379"

ETCD\_MAX\_SNAPSHOTS="5"

ETCD\_MAX\_WALS="5"

#ETCD\_CORS=""

# [cluster]

ETCD\_INITIAL\_ADVERTISE\_PEER\_URLS="https://10.129.52.194:2380"

# if you use different ETCD\_NAME (e.g. test), set ETCD\_INITIAL\_CLUSTER value for this name, i.e. "test=http://..."

ETCD\_INITIAL\_CLUSTER="etcd1=https://10.129.52.142:2380,etcd2=https://10.129.52.162:2380,etcd3=https://10.129.52.194:2380"

ETCD\_INITIAL\_CLUSTER\_STATE="new"

ETCD\_INITIAL\_CLUSTER\_TOKEN="etcd-cluster"

ETCD\_ADVERTISE\_CLIENT\_URLS="https://10.129.52.194:2379"

# [security]

ETCD\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_CLIENT\_CERT\_AUTH="true"

ETCD\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_AUTO\_TLS="true"

ETCD\_PEER\_CERT\_FILE="/etc/etcd/ssl/etcd-ca.pem"

ETCD\_PEER\_KEY\_FILE="/etc/etcd/ssl/etcd-ca-key.pem"

ETCD\_PEER\_CLIENT\_CERT\_AUTH="true"

ETCD\_PEER\_TRUSTED\_CA\_FILE="/etc/etcd/ssl/etcd-server-key.pem"

ETCD\_PEER\_AUTO\_TLS="true"

# [logging]

#ETCD\_DEBUG="false"

# examples for -log-package-levels etcdserver=WARNING,security=DEBUG

#ETCD\_LOG\_PACKAGE\_LEVELS=""

1. 配置etcd服务启动文件

复制/k8sInstall/etcdClusterInstall/config/etcd.service 文件到/usr/lib/systemd/system/；命令如下：

cp -a /k8sInstall/etcdClusterInstall/config/etcd.service /usr/lib/systemd/system/

1. 安装etcd服务二进制脚本

复制/k8sInstall/etcdClusterInstall/software目录下etcd、etcdctl到/usr/local/bin下；命令如下：

cp -a /k8sInstall/etcdClusterInstall/software/{etcd,etcdctl} /usr/local/bin

1. 主节点创建etcd数据目录和证书目录。

mkdir -p /var/lib/etcd/ /etc/etcd/ssl/

|  |
| --- |
| 说明：以上10.129.52.142的相关操作都需要在其他节点执行一次。拷贝10.129.52.142相关的二进制文件etcd、etcdctl和/etc/etcd目录下的所有证书文件到各个etcd服务器节点。具体如下：  1）拷贝二进制文件到各个节点的操作命令：  scp -rp /k8sInstall/etcdClusterInstall/software/{etcd,etcdctl} root@"10.129.52.162":/usr/local/bin/  scp -rp /k8sInstall/etcdClusterInstall/software/{etcd,etcdctl} root@"10.129.52.194":/usr/local/bin/  2）拷贝/etc/etcd下证书文件命令：  Scp -rp /k8sInstall/etcdClusterInstall/config/ssl root@10.129.52.162:/k8sInstall/etcdClusterInstall/config/ssl  Scp -rp /k8sInstall/etcdClusterInstall/config/ssl root@10.19.52.194:/k8sInstall/etcdClusterInstall/config/ssl  3）拷贝/k8sInstall/etcdClusterInstall/config/etcd.service文件到集群各个节点的/usr/lib/systemd/system/目录下。  scp -rp /k8sInstall/etcdClusterInstall/config/etcd.service root@10.129.52.162:/usr/lib/systemd/system/etcd.servic  scp -rp /k8sInstall/etcdClusterInstall/config/etcd.service root@10.129.52.194:/usr/lib/systemd/system/etcd.servic |

1. 启动所有节点上的etcd服务（各个节点都要保证安装完毕）。

systemctl daemon-reload && systemctl enable etcd.service && systemctl start etcd.service

1. etcd集群健康状态检查，可在任意节点执行下面命令。

etcdctl --ca-file=/etc/etcd/ssl/etcd-root-ca.pem --cert-file=/etc/etcd/ssl/etcd.pem \

> --key-file=/etc/etcd/ssl/etcd-key.pem --endpoints="https://10.129.52.142:2380,etcd2=https://10.129.52.162:2380,etcd3=https://10.129.52.194:2380"" cluster-health

member f758a80efdf4be01 is healthy: got healthy result from https://10.129.52.142:2379

member fc0dd343c5856818 is healthy: got healthy result from https://10.129.52.162:2379

member f758a80efdf4be01 is healthy: got healthy result from https://10.129.52.194:2379

Etcd集群健康状态保证检查状态如上是healthy就能提供正常访问，集群安装到此完毕。

etcd常见报错举例。

1. 启动文件配置错误，etcd如果报错大部分是启动文件第3条中配置文件修改错误。

注：另外两台需要修改配置文件中的ETCD\_NAME、ETCD\_DATA\_DIR及相关IP地址，否则将无法启动，具体报错如下：

Oct 18 17:56:47 etcd2 etcd[3117]: recognized environment variable ETCD\_NAME, but unused: shadowed by corresponding flag

Oct 18 17:56:47 etcd2 etcd[3117]: recognized environment variable ETCD\_DATA\_DIR, but unused: shadowed by corresponding flag

Oct 18 17:56:47 etcd2 systemd[1]: etcd.service: main process exited, code=exited, status=1/FAILURE

Oct 18 17:56:47 etcd2 systemd[1]: Failed to start Etcd Server.