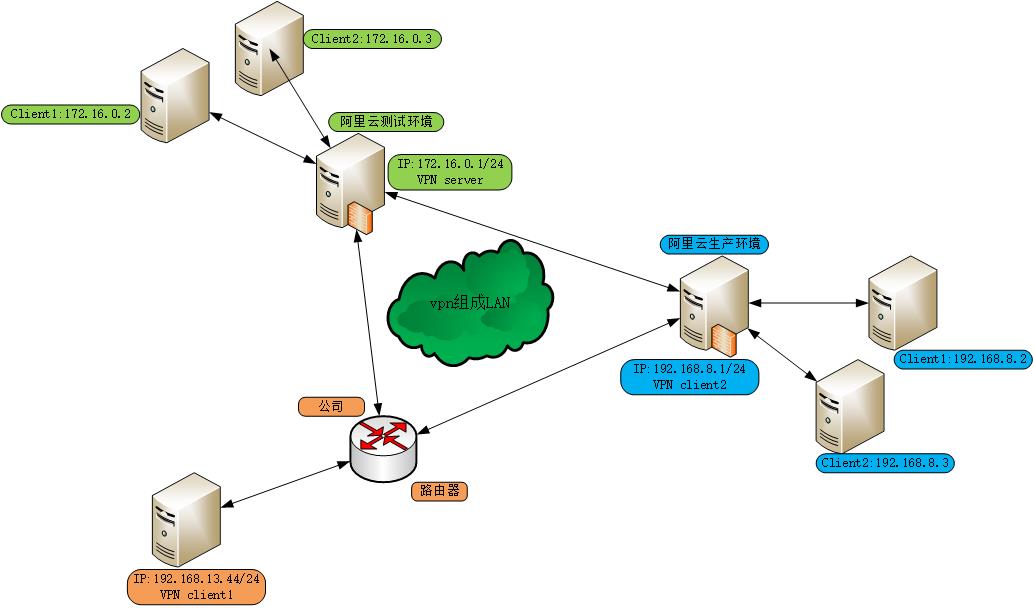
Openvpn site to site 模式的构建

Openvpn 架构图



1. **需求:**

公司内部有一台服务器，需要将公司内部的网络与阿里云组成一个内部的LAN，因为对openvpn 比较熟悉，所以搭建采用openvpn 的site to site 的模式来实现这个功能。

1. **环境准备**

|  |  |  |  |
| --- | --- | --- | --- |
| 服务器IP | 安装程序 | OS/kernel | 备注 |
| 172.16.0.1 | Openvpn easy | Centos7.2/3.10 | 服务器 |
| 192.168.13.44 | Openvpn | Centos7.2/3.10 | 客户端 |
| 192.168.8.1 | Openvpn | Centos7.2/3.10 | 客户端 |

1. **Openvpn 安装**

**3.1 Openvpn 在服务器上的安装**

Openvpn 需要安装的依赖包openssl openssl-devel lzo-devel lzo pam pam-devel automake pkgconfig libtool

yum update

yum install -y openssl openssl-devel lzo-devel lzo pam pam-devel automake pkgconfig libtool git

mkdir /opt/tool/

cd /opt/tool/

git clone <https://github.com/OpenVPN/openvpn>

cd openvpn

git branch -a

git checkout -b 2.2 remotes/origin/release/2.2

git checkout master

autoreconf -i -v -f

./configure --prefix=/usr/local/openvpn

make

make install

3.2 添加到系统变量

vim /etc/profile

export OPENVPN\_HOME=/usr/local/openvpn

export PATH=$OPENVPN\_HOME/sbin:$PATH

source /etc/profile

验证

openvpn --version

OpenVPN 2.3\_git [git:master/8cba9ffc57739eef] x86\_64-unknown-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [MH] [IPv6] built on Aug 24 2016

library versions: OpenSSL 1.0.1e-fips 11 Feb 2013, LZO 2.06

………………………………………………………………………………

* 1. 安装easy-rsa

cd /opt/tool/

git clone https://github.com/OpenVPN/easy-rsa.git

cd easy-rsa/

git branch -a

git checkout -b 2.x remotes/origin/release/2.x

cd easy-rsa/2.0/

cp vars vars.bak

vim vars

sed -i 's/export KEY\_COUNTRY="US"/export KEY\_COUNTRY="CN"/g' vars

sed -i 's/export KEY\_PROVINCE="CA"/export KEY\_PROVINCE=" BJ "/g' vars

sed -i 's/export KEY\_CITY="SanFrancisco"/export KEY\_CITY="BJ"/g' vars

sed -i 's/export KEY\_ORG="Fort-Funston"/export KEY\_ORG="LNGTOP"/g' vars

sed -i 's/export KEY\_EMAIL=mail@host.domain/ export KEY\_EMAIL=wangxingx0122@l.com.com/g' vars

sed -i 's/export KEY\_CN=changeme/export KEY\_CN=vpnserver/g' vars

sed -i 's/export KEY\_NAME=changeme/export KEY\_NAME=server/g' vars

sed -i 's/export KEY\_OU=changeme/export KEY\_OU=Network Center/g' vars

初始化参数

source ./vars

NOTE: If you run ./clean-all, I will be doing a rm -rf on /etc/openvpn/easy-rsa/easy-rsa/2.0/keys

./clean-all

##清除keys目录下所有与证书相关的文件，并生成keys 的目录，所有生成的证书都会在key目录中，每次生成证书都需要初始化一下

./build-ca

## 生成根证书ca.crt和根密钥ca.key

./build-key-server openvpn

## 为服务器端生成证书和密钥

./build-dh

## 会生成dh2048.pem 文件

现在可以生成vpn 的客户端及秘钥了

./easy-rsa/easy-rsa/2.0/build-key xing

注释： xing 是客户端的名字，每一个登录的vpn 客户端都需要一个证书，每一个客户端证书同一时刻只提供一个客户端链接，如果一个客户端需要使用同一个证书多人同时登录的话，不推荐

* 1. 修改配置文件

3.3.1 创建目录

mkdir /etc/openvpn

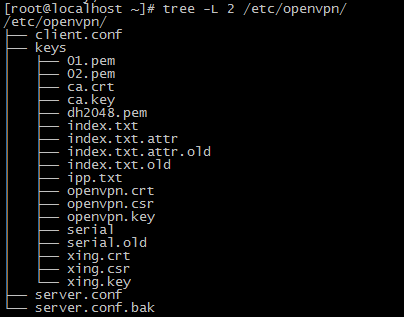
cp /opt/tool/openvpn/sample/sample-config-files/server.conf /etc/openvpn/

# openvpn 自身就带了许多配置文件，只需要拷贝配置文件进行修改即可

server.conf openvpn的主配置文件，启动的时候需要加载这个文件

为了方便管理和配置，将key 的目录也cp过来

cp /opt/tool/easy-rsa/easy-rsa/2.0/keys /etc/openvpn/



#### 3.3.2 Openvpn的配置文件

vim /etc/openvpn/server.conf

port 61199 # 端口

proto tcp #协议采用tcp，因为需要进行数据传输和下载

dev tun #这里采用的tun，openvpn 有tag 和tun模式，如果使用site to site 使用tun模式即可

ca /etc/openvpn/keys/ca.crt # openvpn 的根证书

cert /etc/openvpn/keys/vpn.crt #openvpn 服务器证书

key /etc/openvpn/keys/vpn.key # This file should be kept secret

dh /etc/openvpn/keys/dh2048.pem

server 10.8.0.0 255.255.255.0 #openvpn 的网段

push "route 172.16.0.0 255.255.255.0" # 将客户端的路由push 由服务器配置 文件push 到各个客户端

push "route 192.168.8.0 255.255.255.0"

push "route 192.168.13.0 255.255.255.0"

ifconfig-pool-persist /etc/openvpn/ipp.txt # 客户端获取的ip 指定

client-config-dir /etc/openvpn/ccd #客户端相关配置设置

route 192.168.13.0 255.255.255.0

route 192.168.8.0 255.255.255.0

client-to-client # 可以访问后端服务器，site to site模式必开

duplicate-cn # 一个证书可以多人使用登录

keepalive 10 120

comp-lzo

persist-key

persist-tun

status /var/log/openvpn-status.log

log /var/log/openvpn.log

verb 3 # 级别

#### 3.3.3 创建ccd目录

mkdir /etc/openvpn/ccd/

启动ccd 目录是以创建客户端秘钥名字一致

cat /etc/openvpn/ccd/xxxvpnclinet2

iroute 192.168.8.0 255.255.255.0

cat /etc/openvpn/ccd/xxxvpn client2

iroute 192.168.13.0 255.255.255.0

#### 3.3.4 openvpn 启动脚本

Centos 7 中的启动脚本

vim /usr/lib/systemd/system/openvpn.service

[Unit]

Description=OpenVPN vpm daemon

After=network.target

[Service]

PrivateTmp=true

Type=forking

ExecStart=/usr/local/openvpn/sbin/openvpn --daemon --cd /etc/openvpn/ --config server.conf

[Install]

WantedBy=multi-user.target

### 3.4 启动openvpn

systemctl daemon-reload

systemctl restart openvpn

### 3.5 查看路由等信息

#### 3.5.1 查看进程

ps -ef |grep openvpn

root 9802 1 0 16:32 ? 00:00:00 /usr/local/openvpn/sbin/openvpn --daemon --cd /etc/openvpn/ --config server.conf

netstat -lntp |grep openvpn

tcp 0 0 0.0.0.0:61199 0.0.0.0:\* LISTEN 9802/openvpn

ip a |grep tun

12: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UNKNOWN qlen 100

inet 10.8.0.1/24 brd 10.8.0.255 scope global tun0

openvpn 在server 端启动成功

#### 3.5.2 查看路由

route -n

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

0.0.0.0 172.16.0.253 0.0.0.0 UG 0 0 0 eth0

10.8.0.0 0.0.0.0 255.255.255.0 U 0 0 0 tun0

169.254.0.0 0.0.0.0 255.255.0.0 U 1002 0 0 eth0

172.16.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0

192.168.8.0 10.8.0.2 255.255.255.0 UG 0 0 0 tun0

192.168.13.0 10.8.0.2 255.255.255.0 UG 0 0 0 tun0

由路由表可以看出来，两个openvpn client 的路由已经在openvpn server 端生成，配置openvpn 客户端

### 3.6 开启路由转发及iptables

vim /etc/sysctl.conf

net.ipv4.ip\_forward = 1

iptables

iptables -A INPUT -p tcp -m tcp --dport 61199 -j ACCEPT

openvpn 链接之后需要上网，需要设置下面这条规则

iptables -t nat -A POSTROUTING -s 10.0.8.0/24 -o eth0 -j MASQUERADE

iptables-save > /etc/iptables.rules

添加到开启启动中

vim /etc/rc.local

iptables-restore /etc/iptables.rules

1. Openvpn 客户端

Openvpn 的安装如上面两边openvpn client 同时配置并且启动

4.1 vpn client1 设置如下

client

dev tun

proto tcp

remote 123.56.19.115 61199

resolv-retry infinite

nobind

persist-key

persist-tun

ca /etc/openvpn/keys/ca.crt

cert /etc/openvpn/keys/ljt-lan.crt

key /etc/openvpn/keys/ljt-lan.key

remote-cert-tls server

comp-lzo

verb 3

log /var/log/openvpn.log

4.2 启动openvpn 并且查看路由

route -n

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

0.0.0.0 192.168.13.1 0.0.0.0 UG 100 0 0 eno1

10.8.0.0 0.0.0.0 255.255.255.0 U 0 0 0 tun0

172.16.0.0 10.8.0.1 255.255.255.0 UG 0 0 0 tun0

172.17.0.0 0.0.0.0 255.255.0.0 U 0 0 0 docker0

192.168.8.0 10.8.0.1 255.255.255.0 UG 0 0 0 tun0

192.168.13.0 0.0.0.0 255.255.255.0 U 100 0 0 eno1

192.168.122.0 0.0.0.0 255.255.255.0 U 0 0 0 virbr0

4.3 路由器添加静态路由

因为公司有一台路由器，需要再路由器上添加静态路由：



5 测试

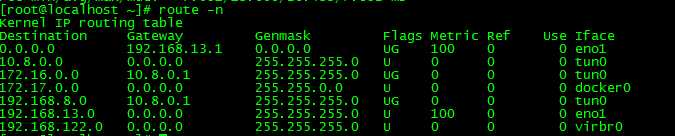
5.1 Vpn client 1 客户端测试

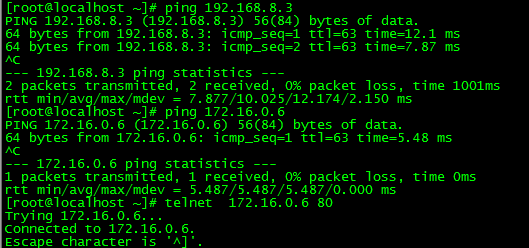
（也就是我自己的机器测试）



### 5.2 vpn client1 测试

查看路由





### 5.3 vpn client2 测试

