**CentOS6.7上安装配置OpenVPN**

1. **安装配置openvpn服务端**
2. **安装依赖**

ntpdate pool.ntp.org

yum -y install lzo lzo-devel gcc pam-devel gcc-c++

1. **安装openvpn**

mkdir tools

cd tools

wget <http://swupdate.openvpn.org/community/releases/openvpn-2.3.2.tar.gz>

tar xf openvpn-2.3.2.tar.gz

cd openvpn-2.3.2

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

make && make install && cd ..

**3、使用easy-rsa生成服务端证书：**

wget https://cloud.github.com/downloads/OpenVPN/easy-rsa/easy-rsa-2.2.0\_master.tar.gz

tar xf easy-rsa-2.2.0\_master.tar.gz

mkdir /etc/openvpn #建立openvpn的配置文件存放目录

cd easy-rsa-2.2.0\_master/easy-rsa/

cp -r \* /etc/openvpn/

cd /etc/openvpn/2.0

cp vars vars\_bak

egrep -v "^#|^$" vars

export EASY\_RSA="`pwd`"

export OPENSSL="openssl"

export PKCS11TOOL="pkcs11-tool"

export GREP="grep"

export KEY\_CONFIG=`$EASY\_RSA/whichopensslcnf $EASY\_RSA`

export KEY\_DIR="$EASY\_RSA/keys"

echo NOTE: If you run ./clean-all, I will be doing a rm -rf on $KEY\_DIR

export PKCS11\_MODULE\_PATH="dummy"

export PKCS11\_PIN="dummy"

export KEY\_SIZE=1024

export CA\_EXPIRE=3650

export KEY\_EXPIRE=3650

export KEY\_COUNTRY="US"

export KEY\_PROVINCE="CA"

export KEY\_CITY="SanFrancisco"

export KEY\_ORG="Fort-Funston"

export KEY\_EMAIL="me@myhost.mydomain"

export KEY\_EMAIL=mail@host.domain

export KEY\_CN=changeme

export KEY\_NAME=changeme

export KEY\_OU=changeme

export PKCS11\_MODULE\_PATH=changeme

export PKCS11\_PIN=1234

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

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

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

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

sed -i 's/export KEY\_EMAIL=mail@host.domain/ export KEY\_EMAIL=aqiang831214@163.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

source ./vars #初始化参数

./clean-all #清空现有的所有证书与蜜钥

./build-ca #生成CA证书

./build-key-server vpnserver #建立server key (一路回车） 有输入Y请输入

./build-dh #生成diffie hellman参数

cp keys/{ca.crt,vpnserver.crt,vpnserver.key,dh1024.pem} /etc/openvpn/

**4、创建并编辑服务器端配置文件server.conf**

cp /root/tools/openvpn-2.3.2/sample/sample-config-files/server.conf /etc/openvpn/

cp /etc/openvpn/server.conf /etc/openvpn/server.conf\_back

cat > /etc/openvpn/server.conf <<EOF

;local $localvpnip

port 52115

proto tcp

dev tun #说明连接方式是点对点的连接，如要以以太网的方式则可以将tun修改为tap

ca /etc/openvpn/ca.crt

cert /etc/openvpn/vpnserver.crt

key /etc/openvpn/vpnserver.key

dh /etc/openvpn/dh1024.pem

server 10.100.80.0 255.255.255.0

ifconfig-pool-persist ipp.txt

#以下是允许访问哪些内网段

push "route 10.0.0.254 255.255.255.0" #路由转发到内网网段

route 10.0.0.254 255.255.255.0 #路由转发到内网网段

push "dhcp-option DNS 202.100.64.68"

client-to-client #如果不加,则各个客户端之间将无法连接

duplicate-cn

keepalive 10 120

comp-lzo

persist-key

persist-tun

status openvpn-status.log

log /var/log/openvpn.log

verb 3

EOF

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

netstat -tunlp|grep openvpn

tcp 0 0 0.0.0.0:52115 0.0.0.0:\* LISTEN 14056/openvpn

**5、转发相关的配置**

sed -i '/ip\_forward/s/0/1/g' /etc/sysctl.conf

sysctl –p

iptables -I INPUT -p tcp --dport 52115 -j ACCEPT

iptables -t nat -I POSTROUTING -s 10.0.0.0/24 -j SNAT --to-source 192.168.0.200

service iptables save

**6、加入开机启动**

echo "#startup openvpn service by $dybm" >> /etc/rc.local

echo "/usr/local/openvpn/sbin/openvpn --config /etc/openvpn/server.conf --daemon " >> /etc/rc.local

1. **安装配置openvpn客户端**

<http://swupdate.openvpn.org/community/releases/openvpn-install-2.3.2-I001-i686.exe> <http://swupdate.openvpn.org/community/releases/openvpn-install-2.3.2-I001-x86_64.exe>

注意：安装客户端时最好将所有的组件全部安装

**1、生成客户端证书和密钥：**（client为客户端名字可以自定义，注意这里的客户端名字不能与上面的服务端名字相同）

cd /etc/openvpn/2.0 #服务器端openvpn配置文件目录

期间也会提示输入一些信息，直接回车默认，选择[Y/n]的都选Y。 若要生成多个客户端的证书和密钥，将client改成另外的名字重复操作即可。所有生成的证书和密钥都存放在/etc/openvpn/2.0/keys下面。

./build-key client #在服务器端生成客户端证书和密钥

1. **下载之前在服务端上生成的客户端证书及密钥**

证书和密钥都存放在/etc/openvpn/2.0/keys里，可以使用winscp链接到**服务端**上下载，将ca.crt、client.crt、client.key这三个文件下载到OpenVPN客户端程序的config文件夹里,默认为：C:\Program Files\OpenVPN\config

sz ca.crt client.crt client.key #我这里使用sz小工具下载

1. **创建客户端配置文件**

C:\Program Files\OpenVPN\sample-config\client.ovpn #客户端配置模版文件

使用记事本修改该文件，内容如下：

client #使OpenVPN作为客户端软件使用，这个client不是自定义名称 不能更改

dev tun #说明连接方式是点对点的连接，如要以以太网的方式则可以将tun修改为tap，要与前面server.conf中的配置一致。

;dev tap

proto tcp #要与前面server.conf中的配置一致。

dev-node vpnclient #设备节点名称，windows需要修改

remote 192.168.0.200 52115 #指定VPN服务器的外网IP地址与端口号，端口与前面的server.conf中配置一致。

;proto udp

;remote-random

resolv-retry infinite

nobind

persist-key

persist-tun

ca ca.crt #指定服务器CA证书文件

cert client.crt #指定客户端证书文件

key client.key #指定客户端私钥文件

ns-cert-type server

redirect-gateway

keepalive 20 60

comp-lzo

verb 3

mute 20

route-method exe

route-delay 2

注意：配置文件中的设备节点名称在windows平台下是需要修改的，并且在安装完成OpenVPN软件后，客户端计算机中会多出一块虚拟网卡，我们还需要将该网卡的名称修改为dev-node定义的设备名称。设置方法是在Windows控制面板中找到网络连接，在网络连接中，将该虚拟网卡重名为vpnclient即可。

1. **运行OpenVPN客户端**

运行OpenVPN GUI后，屏幕右下角的系统托盘区，会显示一个由两个红屏电脑组成的图标，右击它，选中菜单中的Connect，点击Connect，过一会儿，OpenVPN图标变成绿色时就表示链接成功了。

现在VPN连接成功了，让我们一起看一下服务端的日志，如下：

Sun Apr 24 02:39:01 2016 TCP connection established with [AF\_INET]192.168.0.101:49569

Sun Apr 24 02:39:01 2016 192.168.0.101:49569 TLS: Initial packet from [AF\_INET]192.168.0.101:49569, sid=c9704a4a bdf1d307

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 VERIFY OK: depth=1, C=CN, ST=GanSu, L=LanZhou, O=China Mobile, OU=Network, CN=vpnserver, name=server, emailAddress=mail@host.domain

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 VERIFY OK: depth=0, C=CN, ST=GanSu, L=LanZhou, O=China Mobile, OU=Network, CN=client, name=server, emailAddress=mail@host.domain

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 Data Channel Encrypt: Cipher 'BF-CBC' initialized with 128 bit key

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 Data Channel Encrypt: Using 160 bit message hash 'SHA1' for HMAC authentication

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 Data Channel Decrypt: Cipher 'BF-CBC' initialized with 128 bit key

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 Data Channel Decrypt: Using 160 bit message hash 'SHA1' for HMAC authentication

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 Control Channel: TLSv1, cipher TLSv1/SSLv3 DHE-RSA-AES256-SHA, 1024 bit RSA

Sun Apr 24 02:39:02 2016 192.168.0.101:49569 [client] Peer Connection Initiated with [AF\_INET]192.168.0.101:49569

Sun Apr 24 02:39:02 2016 client/192.168.0.101:49569 MULTI\_sva: pool returned IPv4=10.100.80.10, IPv6=(Not enabled)

Sun Apr 24 02:39:02 2016 client/192.168.0.101:49569 MULTI: Learn: 10.100.80.10 -> client/192.168.0.101:49569

Sun Apr 24 02:39:02 2016 client/192.168.0.101:49569 MULTI: primary virtual IP for client/192.168.0.101:49569: 10.100.80.10

Sun Apr 24 02:39:04 2016 client/192.168.0.101:49569 PUSH: Received control message: 'PUSH\_REQUEST'

Sun Apr 24 02:39:04 2016 client/192.168.0.101:49569 send\_push\_reply(): safe\_cap=940

Sun Apr 24 02:39:04 2016 client/192.168.0.101:49569 SENT CONTROL [client]: 'PUSH\_REPLY,route 10.0.0.254 255.255.255.0,dhcp-option DNS 202.100.64.68,route 10.100.80.0 255.255.255.0,topology net30,ping 10,ping-restart 120,ifconfig 10.100.80.10 10.100.80.9' (status=1)

哈哈，相信看了日志，大家就会明白的！