关于OpenVPN的入门使用

内容提纲

1.1 源代码编译安装的初步了解

1.2 安装OpenVPN

1.3 生成证书、服务器端证书、客户端证书

1.4 关于server.ovpm & client.ovpn的配置文件

1.5启动OpenVPN

**Content**

关于OpenVPN在Linux上面的安装

我们以Ubuntu为例进行安装，我们可以直接采用apt方式进行安装，但是因为软件库的问题，可能不是最新的版本，因此采用源代码的方式进行编译安装。

采用apt方式的安装

sudo apt-get install openvpn

1.1 源代码编译安装的初步了解

OpenVPN在Linux上面安装时基于源代码的编译安装，需要对这种源代码编译安装的方式进行初步的了解。

1）下载源代码解压后，

2）使用./configure 让系统检查编译源代码的所有的依赖关系存在，如果出现错误，证明缺少一些必要的依赖库，需要我们进行安装，比如说gcc g++（如果这个都没有安装，肯定是无法编译源代码的）。

3）编译源代码 make

4）安装编译后生成的执行文件到系统 make install, 这里注意使用sudo权限，因为这里是把编译生成的执行文件，复制到操作系统对应的目录中，比如/usr/bin, 还有修改系统的启动脚本命令（添加开机启动）

1.2 安装OpenVPN

1.2.1 Ubuntu安装OpenVPN

安装OpenVPN之前，你必须先确保Ubuntu上已经安装了C编译器(例如gcc)、OpenSSL、LZO(一种无损压缩算法)、PAM(一种可插入式的身份验证模块)。使用apt-get安装命令如下：

sudo apt-get install gcc libssl-dev liblzo2-dev libpam0g-dev

下载OpenVPN源代码openvpn-2.3.10.tar.gz，并且解压

1）~/openvpn-2.3.10$ ./configure # 编译检查环境和代码

2）编译: ~/openvpn-2.3.10$make

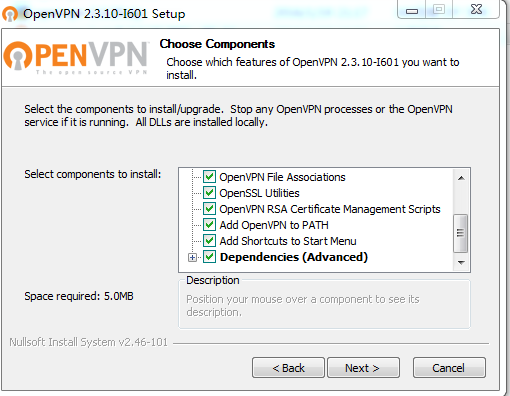
3）安装：~/openvpn-2.3.10$ sudo make install

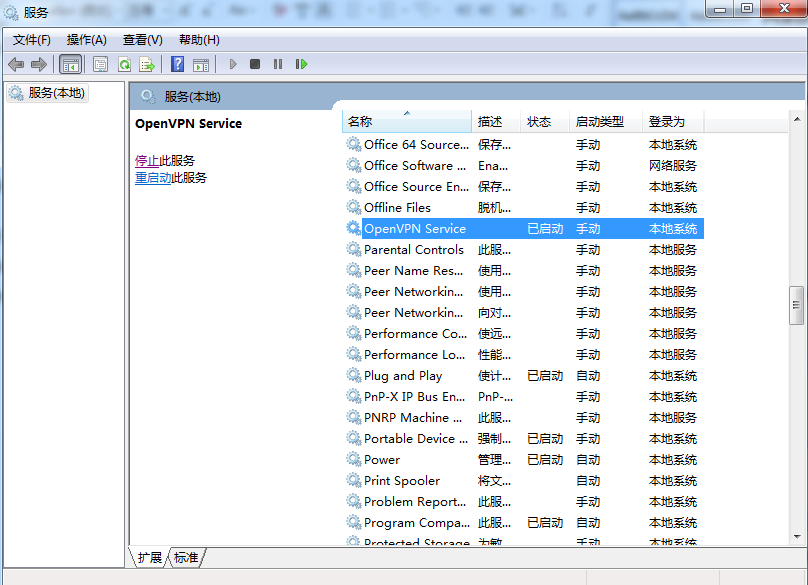
安装完成后，默认开机自启，但我们并没有配置文件，因此建议取消开机自启

1.2.2 Windows下面安装OpenVPN

Windows 7下直接安装软件即可，建议在windows7 64位操作系统中安装，安装选项里

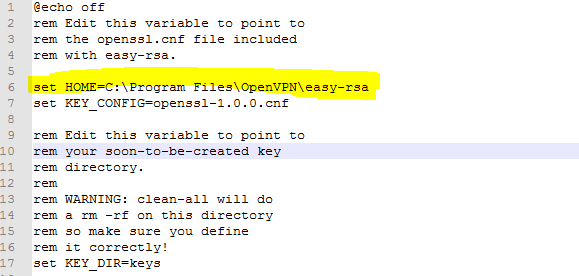
面，把所有的组件全部选上，这样软件便安装在了C:\Program Files\OpenVPN目录中，并且添加到了服务中。





1.3 生成证书、服务器端证书、客户端证书

使用easy-rsa生成OpenVPN的配置文件，在github上面，easy-rsa（<https://github.com/OpenVPN/easy-rsa> ）已经更新到了3.0 版本，但是3.0 的改动比较大，但是在OpenVPN中依旧使用的是2.0，下载地址（<http://pan.baidu.com/s/1slu20bj>），然后就可以生成对应的证书文件了。然后将客户端或者服务器的对应文件放到/etc/openvpn文件夹中。在Windows下面安装的OpenVPN已经带有组件easy-rsa组件，因此可直接在Windows下面生成证书、客户端配置文件和服务器端配置文件，首先修改easy-rsa 中的var.bat.sample



按照提示运行如下脚本命令

echo "enter the openvpn easy-rsa directory"

cd C:\Program Files\OpenVPN\easy-rsa

echo "init the env & clean all car and keys "

init-config

vars

clean-all

echo "build-ca generate the car files pay attention to setting the Common Name with 'openvpn-ca'"

#### KEY\_CN=openvpn\_ca

build-ca

echo "finished generating the car file"

echo "start buile the server key file"

echo "set the Common Name with 'server' name with 'server' , then input y & y"

bulid-key-server server

echo "build client key" #生成两个客户端证书 CommonName设置为client-a & client-b

build-key client-a

build-key client-b

cd C:\Program Files\OpenVPN\bin

#生成关于防止DDOS攻击的安全文件，生成TLS-auth密钥

openvpn --genkey --secret ../easy-rsa/keys/ta.key

#生成迪菲•赫尔曼交换密钥，这是一种安全的加密协议，用于加密数据进行传输

build-dh

查看生成文件的结果：

ca.crt # CA证书

ca.key # CA密钥 这两个文件是使用build-ca生成的

server.crt

server.key #服务端证书和密钥 使用build-key-server server

client-a.crt

client-a.key #客户端A的证书和密钥

client-b.crt

client-b.key #客户端A的证书和密钥

ta.key #TLS-auth密钥

dh1024.pem #迪菲•赫尔曼交换密钥

服务器端的文件（7个）：ca.crt ca.key server.crt server.key dh1024.pem ta.key server.ovpn

客户端端的文件（5个）：ca.crt client-a.crt client-a.key ta.key client.ovpn

关于server.ovpn 和 client.ovpn文件是OpenVPN启动的配置文件，后面讲解如何配置。

1.4 关于server.ovpm & client.ovpn的配置文件

上面两个配置文件是OpenVPN启动的关键配置文件，在sample-config有关于server.ovpn和client.ovpn的样例文件：

server.ovpn

|  |
| --- |
| # Which local IP address should OpenVPN  # listen on? (optional)  local 192.168.1.101  port 1194  proto tcp  ;proto udp  ;dev tap  dev tun  ca ca.crt  cert server.crt  key server.key # This file should be kept secret  dh dh1024.pem  server 10.8.0.0 255.255.255.0  ifconfig-pool-persist ipp.txt  client-to-client  keepalive 10 120  tls-auth ta.key 0 # This file is secret  comp-lzo  persist-key  persist-tun  status openvpn-status.log  verb 3 |

client.ovpn

|  |
| --- |
| client #指定当前VPN是客户端  dev tun #必须与服务器端的保持一致  proto udp #必须与服务器端的保持一致  remote 192.168.1.101 1194 #指定连接的远程服务器的实际IP地址和端口号  resolv-retry infinite #断线自动重新连接，在网络不稳定的情况下(例如：笔记本电脑无线网络)非常有用。  nobind #不绑定特定的本地端口号  persist-key  persist-tun  ca ca.crt #指定CA证书的文件路径  cert client1.crt #指定当前客户端的证书文件路径  key client1.key #指定当前客户端的私钥文件路径  ns-cert-type server #指定采用服务器校验方式  tls-auth ta.key 1 #如果服务器设置了防御DoS等攻击的ta.key，则必须每个客户端开启；如果未设置，则注释掉这一行；  comp-lzo #与服务器保持一致  verb 3 #指定日志文件的记录详细级别，可选0-9，等级越高日志内容越详细 |

1.5启动OpenVPN

1.5.1 Windows OpenVPN Server

把服务器端的配置文件放在OpenVPN安装目录中的config下面，通过OpenVPN服务启动该服务，则OpenVPN Server已经开启。

1.5.2 Windows OpenClient

把客户端的所有配置文件放在OpenVPN安装目录中的config下面，通过OpenVPN客户端双击启动即可。如果连接成功在右下角图标编程绿色。

1.5.3 Linux（Ubuntu）启动OpenVPN

启动脚本 start-vpn.sh

#!/bin/bash

openvpnPid="vpn-pid.txt"

if [ ! -f "$openvpnPid" ];then

sudo nohup openvpn /etc/openvpn/client.ovpn & echo $! > vpn-pid.txt

else

echo "please stop the openvpn first by the stop-vpn.sh"

fi

关闭脚本 stop-vpn.sh

#!/bin/bash

openvpnPid="vpn-pid.txt"

if [ -f "$openvpnPid" ];then

sudo kill `cat vpn-pid.txt`

rm vpn-pid.txt

else

echo "the openvpn service is not running"

fi

配置NAT端口映射脚本 fo.sh

其中10.8.0.6是VPN客户端的VPN-IP 192.168.1.64是IPC摄像机的IP，这里涉及到了Linux数据包转发、防火墙ufw和iptables的知识，之后在专门的进行讲解

iptables -t nat -F

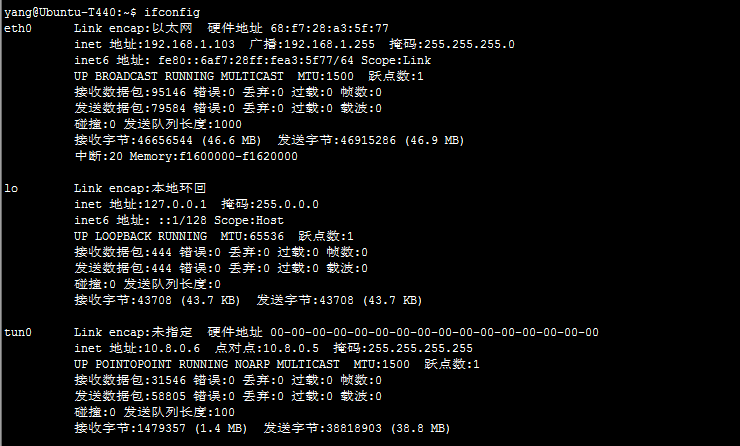
iptables -P INPUT ACCEPT

iptables -P FORWARD ACCEPT

iptables -t nat -A PREROUTING -d 10.8.0.6 -p tcp --dport 8080 -j DNAT --to-destination 192.168.1.64:80

iptables -t nat -A PREROUTING -d 10.8.0.6 -p tcp --dport 8000 -j DNAT --to-destination 192.168.1.64:8000

iptables -t nat -A POSTROUTING -s 192.168.1.64 -p tcp -j SNAT --to-source 10.8.0.6



XXXXXXX:~$ ping 10.8.0.1

PING 10.8.0.1 (10.8.0.1) 56(84) bytes of data.

64 bytes from 10.8.0.1: icmp\_seq=1 ttl=64 time=1.42 ms

64 bytes from 10.8.0.1: icmp\_seq=2 ttl=64 time=1.64 ms

64 bytes from 10.8.0.1: icmp\_seq=3 ttl=64 time=1.48 ms

Success

软件下载：

1. OpenVPN源代码（Linux安装）：<http://pan.baidu.com/s/1skSIkMD> 或者是去github下载

2. OpenVPN Windows安装包：<http://pan.baidu.com/s/1c1Mxbo8>

3. linux下面生成证书工具easy-rsa: <http://pan.baidu.com/s/1slu20bj>