

# OpenVPN服务搭建与管理

## 引言：

本文利用OpenVPN搭建VPN服务，并利用pam\_sqlite3插件实现用户认证；通过openvpn\_web进行用户管理与日志系统。

## 一、安装OpenVPN服务

### 基础环境：

服务端：CentOS 7.6

客户端：Windows 7

OpenVPN: openvpn-2.4.7 (<https://github.com/OpenVPN/openvpn>)

easy-rsa: easy-rsa 3.0.6 (<https://github.com/OpenVPN/easy-rsa>)

OpenVPN GUI: openvpn gui (<https://gitee.com/lang13002/openvpn-portable>)

### 服务器：

外网IP: 1.10.10.123 ens33

内网IP: 192.168.100.100 ens34

### 1.1 安装openvpn

#### 安装依赖包

```
# yum install -y gcc gcc-c++ libtool automake lz4-devel lzo-devel pam-devel  
openssl-devel systemd-devel sqlite-devel
```

从github上下载openvpn源代码包并解压

```
# wget https://github.com/OpenVPN/openvpn/archive/v2.4.7.tar.gz  
# tar -xvf v2.4.7.tar.gz
```

#### 编译openvpn并安装

```
# cd openvpn-2.4.7  
# autoreconf -i -v -f  
# ./configure --prefix=/usr/local/openvpn --enable-lzo --enable-lz4 --enable-  
crypto --enable-server --enable-plugins --enable-port-share --enable-iproute2 --  
enable-pf --enable-plugin-auth-pam --enable-pam-dlopen --enable-systemd  
# make && make install
```

#### 配置系统服务

安装openvpn-server@.service系统服务

```
# cp /usr/local/openvpn/lib/systemd/system/openvpn-server@.service
/usr/lib/systemd/system/
```

设置openvpn-server@.service启动

```
# ln -s /usr/lib/systemd/system/openvpn-server\@.service
/etc/systemd/system/multi-user.target.wants/openvpn-server@server.service
# systemctl daemon-reload
# systemctl enable openvpn-server@server
```

## 1.2 生成证书

下载easy-rsa3并解压

```
# wget https://github.com/OpenVPN/easy-rsa/archive/v3.0.6.tar.gz
# tar -xvf v3.0.6.tar.gz
```

根据easy-rsa-3.0.6/easyrsa3/vars.example文件生成全局配置文件vars

```
# cd easy-rsa-3.0.6/easyrsa3/
# cp vars.samples vars
```

修改vars文件，根据需要去掉注释，并修改对应值

```
set_var EASYRSA_REQ_COUNTRY      "CN"
set_var EASYRSA_REQ_PROVINCE     "HUBEI"
set_var EASYRSA_REQ_CITY         "WUHAN"
set_var EASYRSA_REQ_ORG          "ZJ"
set_var EASYRSA_REQ_EMAIL        "zj@test.com"
set_var EASYRSA_REQ_OU           "ZJ"
set_var EASYRSA_KEY_SIZE         2048
set_var EASYRSA_ALGO             rsa
set_var EASYRSA_CA_EXPIRE        3650
```

生成服务端证书

```
# ./easyrsa init-pki      # 初始化，生成一系列文件与目录
# ./easyrsa build-ca      # 生成根证书，记住ca密码
# ./easyrsa build-server-full server nopass # 生成服务端证书，nopass参数生成一个无密码的证书
# ./easyrsa gen-dh        # 生成Diffie-Hellman
```

生成客户端证书

```
# ./easy-rsa build-client-full client1 nopass
注：可生成client1，client2，client3或对应姓名的客户端证书
```

为了提高安全性，生成ta.key

```
# openvpn --genkey --secret ta.key
```

整理服务端证书

```
# cp pki/ca.crt /etc/openssl/server/
# cp pki/private/server.key /etc/openssl/server/
# cp pki/issued/server.crt /etc/openssl/server/
# cp pki/dh.pem /etc/openssl/server/
# cp ta.key /etc/openssl/server/
```

### 1.3 添加SQLite认证

下载pam\_sqlite3并安装

```
# git clone https://gitee.com/lang13002/pam_sqlite3.git
# cd pam_sqlite3
# make && make install
```

添加pam认证文件

```
# vim /etc/pam.d/openvpn
auth      required    pam_sqlite3.so db=/etc/openvpn/openvpn.db table=t_user
user=username passwd=password expire=expire crypt=1
account   required    pam_sqlite3.so db=/etc/openvpn/openvpn.db table=t_user
user=username passwd=password expire=expire crypt=1
```

导入sqlite3数据库文件

```
# sqlite3 /etc/openvpn/openvpn.db
sqlite> .read openvpn_web/model/openvpn.sql
```

### 1.4 创建服务端配置文件

参照sample/sample-config-files/server.conf文件

```
# vim /etc/openvpn/server/server.conf
port 1194
proto tcp-server
;proto udp
dev tun
topology subnet

ca /etc/openvpn/server/ca.crt
cert /etc/openvpn/server/server.crt
key /etc/openvpn/server/server.key
dh /etc/openvpn/server/dh.pem

cipher AES-256-CBC
auth SHA512
tls-version-min 1.2
tls-cipher TLS-DHE-RSA-WITH-AES-256-GCM-SHA384:TLS-DHE-RSA-WITH-AES-128-GCM-SHA256:TLS-DHE-RSA-WITH-AES-256-CBC-SHA:TLS-DHE-RSA-WITH-CAMELLIA-256-CBC-SHA:TLS-DHE-RSA-WITH-AES-128-CBC-SHA:TLS-DHE-RSA-WITH-CAMELLIA-128-CBC-SHA

tls-auth /etc/openvpn/server/ta.key 0
;tls-crypt /etc/openvpn/server/ta.key ;客户端将ta.key嵌入到配置文件

user openvpn
group openvpn
```

```
server 10.8.0.0 255.255.255.0
push "dhcp-option DNS 114.114.114.114"
push "route 192.168.100.0 255.255.255.0"
push "route-gateway 192.168.100.1"

verify-client-cert require
username-as-common-name
plugin /usr/local/openvpn/lib/openvpn/plugins/openvpn-plugin-auth-pam.so openvpn

keepalive 10 120
comp-lzo
compress "lz4"
persist-key
persist-tun
status /var/log/openvpn-status.log
log /var/log/openvpn.log
verb 3
```

## 1.5 开启路由转发功能

```
# 路由转发
# vim /etc/sysctl.conf
net.ipv4.ip_forward = 1

# 临时启用
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

## 1.6 启动openvpn服务

```
# systemctl start openvpn-server@server
```

# 二、客户端配置

## 2.1 下载客户端程序：

从<https://gitee.com/lang13002/openvpn-portable/repository/archive/v1.0>下载程序，并安装网卡驱动；

## 2.2 安装驱动：

运行openvpn-portable/tap-windows.exe

## 2.3 设置客户端证书

将上面生成的ca.crt, client1.crt, client1.key放到openvpn-portable的data/config下，并修改客户端配置

```
client
dev tun
proto tcp-client
remote vpnserver.com 1194

allow-recursive-routing
```

```

resolv-retry infinite
nobind
persist-key
persist-tun

remote-cert-tls server
auth-user-pass
auth-nocache
ca ca.crt
cert client1.crt
key client1.key

cipher AES-256-CBC
auth SHA512
tls-version-min 1.2
tls-cipher TLS-DHE-RSA-WITH-AES-256-GCM-SHA384:TLS-DHE-RSA-WITH-AES-128-GCM-SHA256:TLS-DHE-RSA-WITH-AES-256-CBC-SHA:TLS-DHE-RSA-WITH-CAMELLIA-256-CBC-SHA:TLS-DHE-RSA-WITH-AES-128-CBC-SHA:TLS-DHE-RSA-WITH-CAMELLIA-128-CBC-SHA

tls-auth ta.key 1

comp-lzo
compress lz4
verb 3
mute 20

```

注：当有多个客户端时，有多个文件(ca.crt, client1.crt, client1.key, client.ovpn)需要分发给客户，势必会很麻烦；可以将证书嵌入到客户端配置文件中：

```

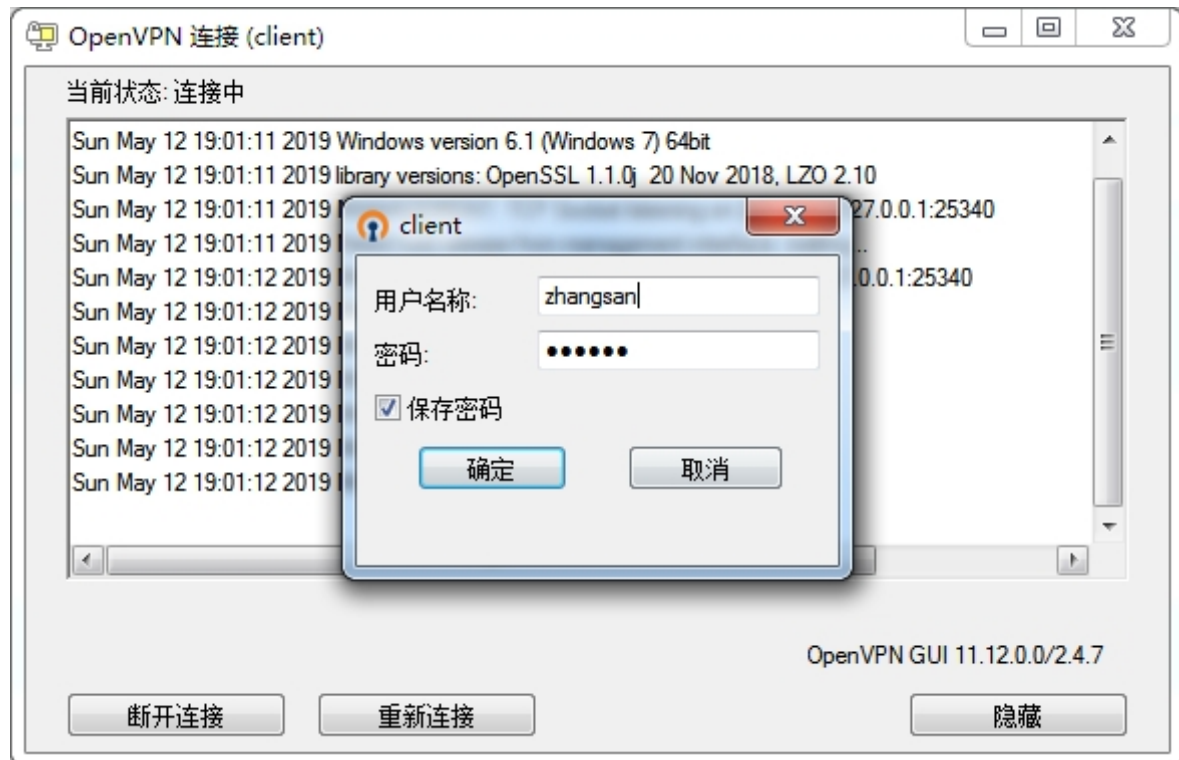
;ca ca.crt          // 将这行注释掉
;cert client.crt    // 将这行注释掉
;key client.key     // 将这行注释掉
;tls-auth ta.key 1 // 将这行注释掉
<ca>
-----BEGIN CERTIFICATE-----
MIIDGDCCAgCgAwIBAgIJAi9Ld4P1KEiOMA0GCSqGSIb3DQEBCwUAMA0xCzAJBgNV
....
OCeTQvQ4whyIvVgURV3ITcAKYFKUQ1sPbpjuZg==
-----END CERTIFICATE---
</ca>
<cert>
-----BEGIN CERTIFICATE-----
MIIDODCCAiCgAwIBAgIRAIzoeQ5PvHDS9xpTLMp3RqMwDQYJKoZIhvcNAQELBQAw
.....
nCpzc3l8svezxk2r
-----END CERTIFICATE-----
</cert>
<key>
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKggggSkAgEAAoIBAQDw1iq3HBe1otCU
.....
u1laNc6mu3N/wTPZoQhDOKAO
-----END PRIVATE KEY-----
</key>
<tls-crypt>
#
# 2048 bit OpenVPN static key

```

```
#
-----BEGIN OpenVPN Static key v1-----
376ff00121bc6cd39fe1382c44be1433
.....
-----END OpenVPN Static key v1-----
</tls-crypt>
```

## 2.4 连接VPN

启动openvpn-porable



## 三、OpenVPN用户管理与日志

### 3.1 安装依赖

```
# pip install peewee tornado pycryptodome apscheduler
```

### 3.2 下载openvpn-web

```
# git clone https://gitee.com/lang13002/openvpn_web.git
```

### 3.3 添加日志脚本

服务端配置添加运行脚本

```
script-security 2
client-connect /etc/openvpn/server/connect.py
client-disconnect /etc/openvpn/server/disconnect.py
```

connect.py

```
#!/usr/bin/python

import os
import time
import sqlite3

username = os.environ['common_name']
trusted_ip = os.environ['trusted_ip']
trusted_port = os.environ['trusted_port']
local = os.environ['ifconfig_local']
remote = os.environ['ifconfig_pool_remote_ip']
timeunix= os.environ['time_unix']

logintime = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime(time.time()))

conn = sqlite3.connect("/etc/openvpn/openvpn.db")
cursor = conn.cursor()
query = "insert into t_logs(username, timeunix, trusted_ip, trusted_port, local,
remote, logintime) values('%s', '%s', '%s', '%s', '%s', '%s', '%s')" %
(username, timeunix, trusted_ip, trusted_port, local, remote, logintime)
cursor.execute(query)
conn.commit()
conn.close()
```

disconnect.py

```
#!/usr/bin/python

import os
import time
import sqlite3

username = os.environ['common_name']
trusted_ip = os.environ['trusted_ip']
received = os.environ['bytes_received']
sent = os.environ['bytes_sent']

logouttime = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime(time.time()))

conn = sqlite3.connect("/etc/openvpn/openvpn.db")
cursor = conn.cursor()
query = "update t_logs set logouttime='%s', received='%s', sent= '%s' where
username = '%s' and trusted_ip = '%s'" % (logouttime, received, sent, username,
trusted_ip)
cursor.execute(query)
conn.commit()
conn.close()
```

### 3.4 初始化配置

使用OpenSSL生成密钥对

```
# openssl genrsa -out private.pem 1024
# openssl rsa -in private.pem -pubout -out public.pem
# cat private.pem
# cat public.pem
```

修改config.py文件，指定sqlite3 数据文件位置及用于用户密码加密与解密的密钥对

```
dbfile="E:\\Projects\\openvpn_web\\handler\\openvpn.db"
private_key=b'-----BEGIN RSA PRIVATE KEY-----
\\nMIIIEowIBAAKCAQEAAQIff+G/cpiP582c2JGKA6cb8eIrzuUq9eF1swwhVRjKcEMaQz\\nQZH01pxb4GJT
PPNl64YK2uyZWkdWvrwhnas2v4GzpXNL1tkv4YYlT3rSvCFF3ouW\\nTICsOvZaaZso7w7W2NzvvF0vGO
wv7o8ad+hBXlczUvAXISaodQW8+aMNs0GMVwmG\\nXEVtGsjg+LuoDDiRnkq3B0lClAroa1tzMw2yAu+g
urYzEaZ1rt4JOzo4RMwu+CqI\\ninqbIPGLZZ7lCWhd1A6AKKzTOjfgEnzQDkNrZwPSAPIK6GaIWOSRbw
8JswHuBVOI\\nNGsjc5lIg5U8hOWKZGIWqWYC58uhnDW3b8BdNWIDAQABAOIBABC0X2gboi2Lg/J8\\nxD
gnbfXRXbu3gdKvDVPJ3vLW1YwzSbnQrtwq147wh9byxpnMi90/kDS1664Ehpv\\nhsKFvTJQB4Gd9ltb
9x2/v470QspgaVcs3wpyDjaCc/E7mDYSS1/vHlwfaLdb6TzI\\nbYMTPrPzkvGmo6hC8ZlWJH2D7AHMEbb
qd4SuYdilEmejOu7Ec1rngFFuVD1xy0LeQ\\nqlEwXwwijSqi3E5o/XyRZGvkft4d7seMxmswFVFXxlK7
JX/hPV0qqk9AjaBQJK2u\\nlwZUqVCra1rZWGa4pk7VaqabC3qtFLPL0uLaJ2e1w8dMhtjwHts50fqpi0
5P6FWJ\\nzMXCxCNECgYEAvxZAnAju1cFRQE4mx0OmMq/wdYXq9SX6KSfHgSci8yMRi25Jovhg\\njHxf4P
adbhYCWfvg/G5/XDJJzLH+Q2xhMCCMi1KG9hC8xja+wtwniJdWAYUUhcm\\nwvHKNG4dTDekY93EOglw
s1bPSRJ1+uE9+jemRdqfOpKkC01zVcg+DpECgYEA4/wD\\n/bjPgWlYpiTTCGMBsXzpp618amXYbuh3CR
AORuXiethfq2dw4ogP12RbCyqC1AX2\\nPOLOTEORVDKuzEFWDdKOFskI7A1xtiPwccDCq5RCPwWSZxAT
5DAY5Eccl/5/P9E\\np+VEfMtPRDt6YDsLwbcCHegLF+7IYpnP+RkEo0cCgYBdJHKf3CoLSSzaxH1gfP
bb\\nESTr1DhgAH/82ZgEm1gm3dYqebrJBm3jG8ecd3lrdKz2wbD8Orw365P77fL7WHPT\\nKrp2nh2Ncc
GKEJrpjaQTKz/wA7dqWRRoFh2zCs2b5crWJuQDf000Jt6b+FYrymku\\nc0kbr6IXwe0JCWkiGLYVEQKB
gQC9UJmkbGzOyEkTmwtrrJ2sZDu8GHT4ok5i08s3\\nyJDCyonzUZzqQXFDwpGIPjzqIgzylzIjF2b0I
vyMoE+eohYBGQigisZvXQ629gE\\n8Hv7ek4nnp3+ZR6/ZD5X3t1Rc2mudeTztPDRPxt+ZBL2tjLGVxE3
+wyzfIgiCwro\\nkaHTYwKBgEt8wiU1zjnXfDBWFBQ9ldqkZ4YgouS6yifzy8aPwR3faTEG68aGS9lC\\n
whdatQmT4vprCASCAELGNJD7DP8/nxtTF6oVBg69Lssxv2gowduw1u1sg56ytMOC\\nuRTbukdbladS65
LDIbz7c5nVEJ+ZGUL88bbPxZ+SWXEwFRZGZNj+\\n-----END RSA PRIVATE KEY-----'
public_key=b'-----BEGIN PUBLIC KEY-----
\\nMIIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAAQIff+G/cpiP582c2JGKA6\\ncb8eIrzuUq9e
F1swwhVRjKcEMaQzQZH01pxb4GJTTPNl64YK2uyZWkdWvrwhnas2\\nv4GzpXNL1tkv4YYlT3rSvCFF3o
uwTICsOvZaaZso7w7W2NzvvF0vGOWv7o8ad+hB\\nxlczUvAXISaodQW8+aMNs0GMVwmGXEVtGsjg+Luo
DDiRnkq3B0lClAroa1tzMw2y\\nAu+gurYzEaZ1rt4JOzo4RMwu+CqI\\ninqbIPGLZZ7lCWhd1A6AKKzTOj
fgEnzQDkNr\\nzWPSAPIK6GaIWOSRbw8JswHuBVOINGSjc5lIg5U8hOWKZGIWqWYC58uhnDW3b8Bd\\nNW
IDAQAB\\n-----END PUBLIC KEY-----'
```

## 3.5 启动服务

```
# python myapp.py
```

## 3.6 管理界面





