开发记录

卷一

Dolphin

EDITED BY XIAOQIANG JIANG

HTTP://JIANGXIAOQIANG.GITHUB.COM/

Version 23:30, March 17, 2018

Contents

这里记录的是一些比较杂乱的笔记，绝大多数文字皆来源于网络，不是自己的原创，这里没有高深的算法，没有宏伟的技术及系统架构，只是一些平时工作中遇到的一些问题，和解决问题的思路以及所采用的方案。由于平时工作时还没有遇到前人没有遇到过的问题需要自己发明方去解决 (其实真的遇到估计也是没辙)，所以绝大部分内容是为了避免再遇到同样的问题时，又需要到处去搜寻，索性将之记录下来，以便于下次可以快刀斩乱麻，迅速解决问题。

I Tool

0.1 Tool Set

0.1.1 ECS(Elastic Compute Service)

配置内网 ECS 端口映射规则, 在云服务器 ECS–> 网络和安全–> 安全组–> 配置规则.

0.1.2 shadowsocks

启动 ss 服务器:

```
#可以查看启动日志
ssserver -c /home/ec2-user/shadowsocks.json
#后台启动
ssserver -c /home/ec2-user/shadowsocks.json -d start
```

Shadowsocks 客户端操作:

```
sudo apt-get install python-pip
sudo apt-get install python-setuptools m2crypto
#安装 Shadowsocks(Ubuntu/Fedora)
pip install shadowsocks
#前台启动
#可以看到实时的日志输出
#关闭终端后代理断开
sslocal -c /etc/shadowsocks/shadowsocks.json
#后台启动
sslocal -c /etc/shadowsocks/shadowsocks.json -d start
```

0.1.3 youtube-dl

YouTube 视频一般是不能下载的, 但是是国内访问 YouTube 比较慢, 经常卡顿, 所以可以使用 youtube-dl 工具下载 YouTube 视频:

```
#下载默认的视频格式
youtube-dl https://www.youtube.com/watch?v=
        SnHxKQiXrFU
#查看所有视频格式
youtube-dl -F https://www.youtube.com/watch?v=
        SnHxKQiXrFU
```

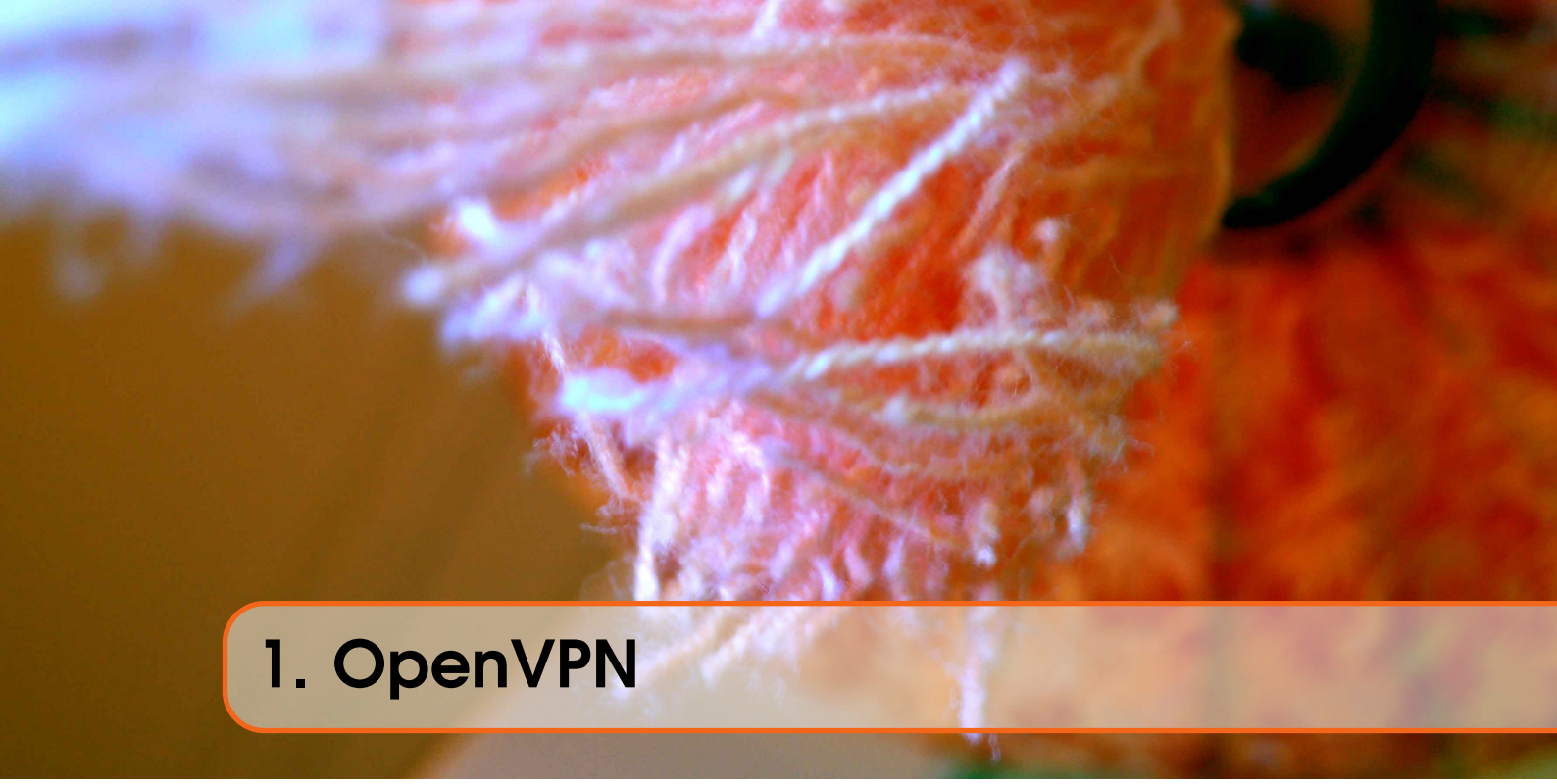
```
#下载指定清晰度的视频
#137 为指定视频格式的编码 format code
youtube-dl -f 137 https://www.youtube.com/watch?v=
        SnHxKQiXrFU
```

查看 YouTube 所有格式视频输出效果如图所示:

```
[youtube] eq3KiAH4IBI: Downloading webpage
[youtube] eq3KiAH4IBI: Downloading video info webpage
[youtube] eq3KiAH4IBI: Extracting video information
[info] Available formats for eq3KiAH4IBI:
format code  extension  resolution note
249          webm       audio only DASH audio   55k , opus @ 50k, 21.82MiB
171          webm       audio only DASH audio   80k , vorbis@128k, 31.52MiB
250          webm       audio only DASH audio   90k , opus @ 70k, 30.13MiB
140          m4a        audio only DASH audio   99k , m4a_dash container, mp4a.40.2@128k, 40.92MiB
251          webm       audio only DASH audio  126k , opus @160k, 44.80MiB
278          webm       256x144    144p   64k , webm container, vp9, 13fps, video only, 16.01MiB
242          webm       426x240    240p  114k , vp9, 25fps, video only, 18.26MiB
160          mp4        256x144    144p  126k , avc1.42c00c, 13fps, video only, 47.61MiB
243          webm       640x360    360p  218k , vp9, 25fps, video only, 36.78MiB
134          mp4        640x360    360p  248k , avc1.4d401e, 25fps, video only, 49.41MiB
```

Figure 1: 查看所有格式视频

0.1.4 Jenkins



1. OpenVPN

OpenVPN 从 2001 年开始开发，使用的是 C 语言。此处使用的 OpenVPN 版本是 2.4.1。如果使用 Mac 下的 brew 工具安装，则 OpenVPN 目录在：/usr/local/Cellar/openvpn/2.4.1，OpenVPN 的配置文件在：/usr/local/etc/openvpn。目前 OpenVPN 能在 Solaris、Linux、OpenBSD、FreeBSD、NetBSD、Mac OS X 与 Microsoft Windows 以及 Android 和 iOS 上运行，并包含了许多安全性的功能。此处的服务器使用的是 CentOS 7.3，客户端包含 Fedora 24、Ubuntu 14.04、Ubuntu 16.04、Window 7、Windows 10。在 OpenVPN 网络中查看存活的主机：

```
nmap -A -T4 10.0.0.*
```

1.0.1 安装 (Install)

安装基础包：

```
sudo yum -y install openssl openssl-devel lzo openvpn
         easy-rsa --allowerasing
#手动安装
wget -c https://swupdate.openvpn.org/community/
        releases/openvpn-2.4.1.tar.gz
tar -zxvf openvpn-2.4.1.tar.gz
./configure
```

```
make && make install
```

LZO 是致力于解压速度的一种数据压缩算法，LZO 是 Lempel-Ziv-Oberhumer 的缩写。这个算法是无损算法，参考实现程序是线程安全的。实现它的一个自由软件工具是 lzop。最初的库是用 ANSI C 编写、并且遵从 GNU 通用公共许可证发布的。现在 LZO 有用于 Perl、Python 以及 Java 的各种版本。代码版权的所有者是 Markus F. X. J. Oberhumer。

1.0.2 生成客户端证书

生成客户端证书端步骤如下：

```
#建立 openvpn 配置文件存放目录
mkdir -p /etc/openvpn
#如果没有安装 easy-rsa 工具
yum install easy-rsa
#建立一个空的 pki 结构，生成一系列的文件和目录
./easyrsa init-pki
```

PKI：Public Key Infrastructure 公钥基础设施。生成请求：

```
./easyrsa gen-req dolphinfedora
```

输入 PEM 验证码。PEM - Privacy Enhanced Mail, 打开看文本格式, 以"——BEGIN..." 开头, "——END..." 结尾, 内容是 BASE64 编码。Apache 和 *NIX 服务器偏向于使用这种编码格式. 签约：

```
#切换到服务端生成 rsa 的目录
#导入 req
./easyrsa import-req ~/client/easyrsa/easy-rsa-master
        /easyrsa3/pki/reqs/dolphinfedora.req
        dolphinfedora
#用户签约，根据提示输入服务端的 ca 密码
./easyrsa sign client dolphinfedora
```

PKI：Public Key Infrastructure 公钥基础设施。输入 PEM 验证码。PEM - Privacy Enhanced Mail, 打开看文本格式, 以"——BEGIN..." 开头, "——END..." 结尾, 内容是 BASE64 编码. 查看 PEM 格式证书的信息:openssl x509 -in certificate.pem -text -noout。Apache 和 *NIX 服务器偏向于使用这种编码格式. 服务端生成的文件有：

文件名称	说明 (Purpose)	位置
ca.crt	根证书 (Root CA certificate) 件	Server+All Clients
reqs/server.req		
reqs/dolphin.req		
private/ca.key	根证书私钥 (Root CA key)	key signing machine only
private/server.key		
issued/server.crt	服务器证书 Server Certificate	server only
issued/dolphin.crt		
dh.pem	Diffie Hellman parameters	server only

客户端生成的文件有：

序号	名称
private/dolphinclient.key	
reqs/sdolphinclient.req	

拷贝出客户端证书文件：

```
cp easyrsa/easy-rsa-master/easyrsa3/pki/ca.crt ~/
        dolphinfedora/
cp easyrsa/easy-rsa-master/easyrsa3/pki/issued/
        dolphinfedora.crt ~/dolphinfedora/
cp ~/client/easyrsa/easy-rsa-master/easyrsa3/pki/
        private/dolphinfedora.key ~/dolphinfedora/
```

启动 OpenVPN：

```
sudo openvpn server.conf
# Mac 下启动 OpenVPN
sudo /usr/local/Cellar/openvpn/2.4.1/sbin/openvpn /
        usr/local/etc/openvpn/client.conf
# 需要以后台交互方式启动时
screen sudo openvpn client.conf
```

客户端端配置如下：

```
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
#指定连接的远程服务器的实际 IP 地址和端口号
remote 192.168.1.106 1194
#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
resolv-retry infinite
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
#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；
tls-auth ta.key 1
comp-lzo                #与服务器保持一致
#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
verb 3
```

配置 ns-cert-type(Netscape Cert Type) 指定为 server 主要是防止中间人攻击(Man-in-the-Middle Attack)。在服务端做如下配置：

```
nsCertType server
```

Bibliography

Books

Articles