Ubuntu 下如何使用 SSH Tunneling Proxy ?

SSH Tunneling 介绍

什么是 SSH Tunnel

SSH Tunnel 属于 port forwarding, 它建立在 ssh 连接上的一个加密的通道，利用它可以传输没经加密的数据从而达到安全的目的。创建了 SSH Tunnel 之后，你不用直接去访问远程的服务(假使你是连接某个远程服务)，相反你访问你本机的某个端口，而 SSH Client 就会把你的数据通过已建立的加密通道转发到远程主机的目的端口去。

需要的工具

SSH 帐户一枚（老 T SSH 提供高速美国&日本 SSH 账户），SSH Tunnel 连接创建工具，如 SSH 命令，Plink 及 AutoSSH 等。

建立 SSH Tunneling Proxy

在使用客户端连接 SSH Tunneling Proxy 之前，我们必需先通过相应的工具建立 SSH 隧道，常用的工具有原生的 SSH 命令、 AutoSSH 及 Plink 等。

原生 SSH

在终端下输入如下命令

ssh -N -v username@hostip -D 127.0.0.1:7070

把其中的 username ， hostip 替换成你自己的内容。

第一次运行此命令需要输入 yes 来接受证书，最后输入 SSH 密码。如果你不想每次都输入密码的话，可以采用证书认证方式。

终端下运行 ssh-keygen 命令来生成证书，直接按三次回车无需输入任何内容。

ssh-keygen

进入 SSH 目录

cd ~/.ssh

打开 id_rsa.pub 文件，复制里面的所有内容。

然后通过 ssh 连接到远程 ssh 主机，进入 ~/.ssh 目录，打开 authoried_keys 文件，把刚才复制于 id_rsa.pub 的内容粘贴进去。

这样下次再建议 ssh tunneling 的时候就无需输入密码了。

最后你可以建立一个 shell 脚本文件，这样不必每次输入命令了，方便使用。

AutoSSH

AutoSSH 的使用方法和 SSH 类似，只是它提供了断线自动连接功能，这样就不必每次重新输入命令了。

安装

sudo apt-get install autossh

使用

autossh -M 2000 -N -v username@hostip -D 127.0.0.1:7070

Plink

Plink 最大的好处在于可以指定密码，不必采用证书方式就可以不输入密码建立链接了。

安装

sudo apt-get install putty-tools

使用

plink -N -v username@hostip -D 127.0.0.1:7070 -pw password

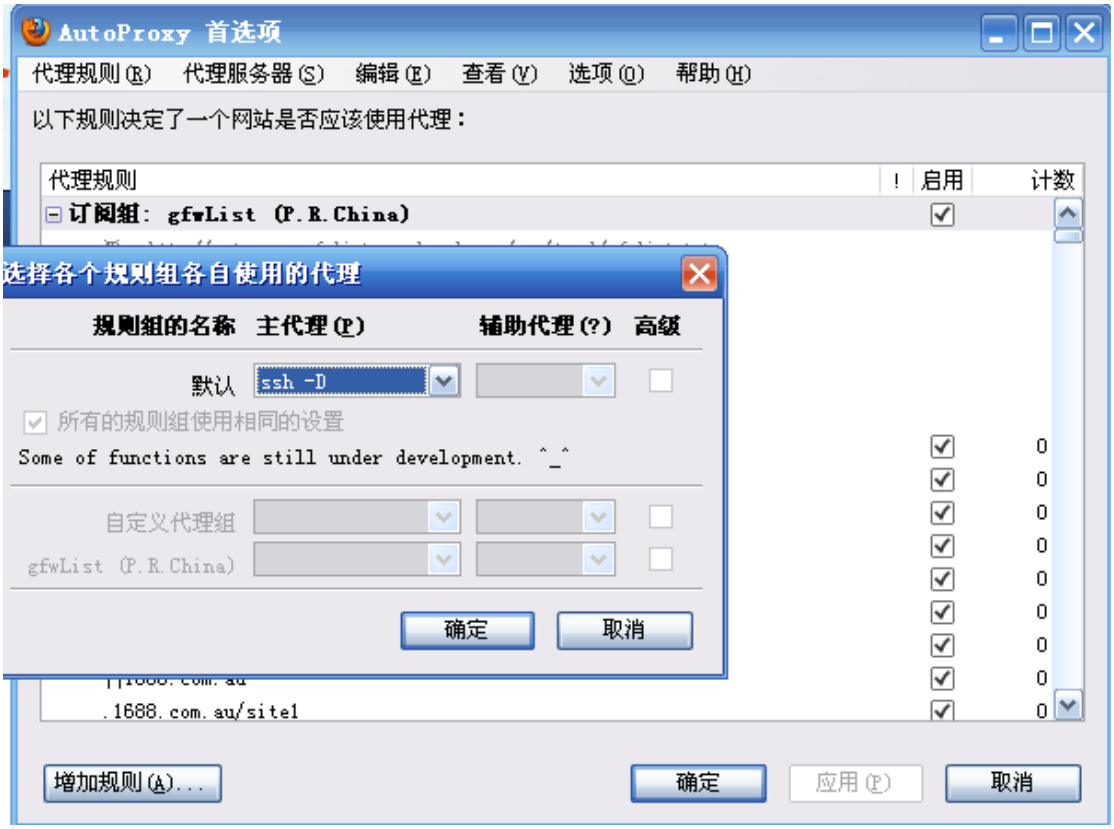
把其中的 username ， hostip ， password 替换成你自己的内容。

gSTM

gSTM 是图形化工具，见此文：gSTM：Ubuntu 上的 SSH Tunneling 图形界面管理工具
客户端配置介绍

FireFox + AutoProxy

下载安装 AutoProxy 插件，地址： https://addons.mozilla.org/zh-CN/firefox/addon/11009 。
使用 AutoProxy 的好处在于，只有访问被墙网站的情况下 FireFox 才通过 SSH Tunneling Proxy 访问网站，所以对于国内网站还是采用本地线路。
安装完插件后，只要默认启用 SSH -D 代理可以了，Autoproxy 的默认端口配置就是 7070 ，如果你之前采用和我一样的 7070 端口的话就不用作任何修改，如图。

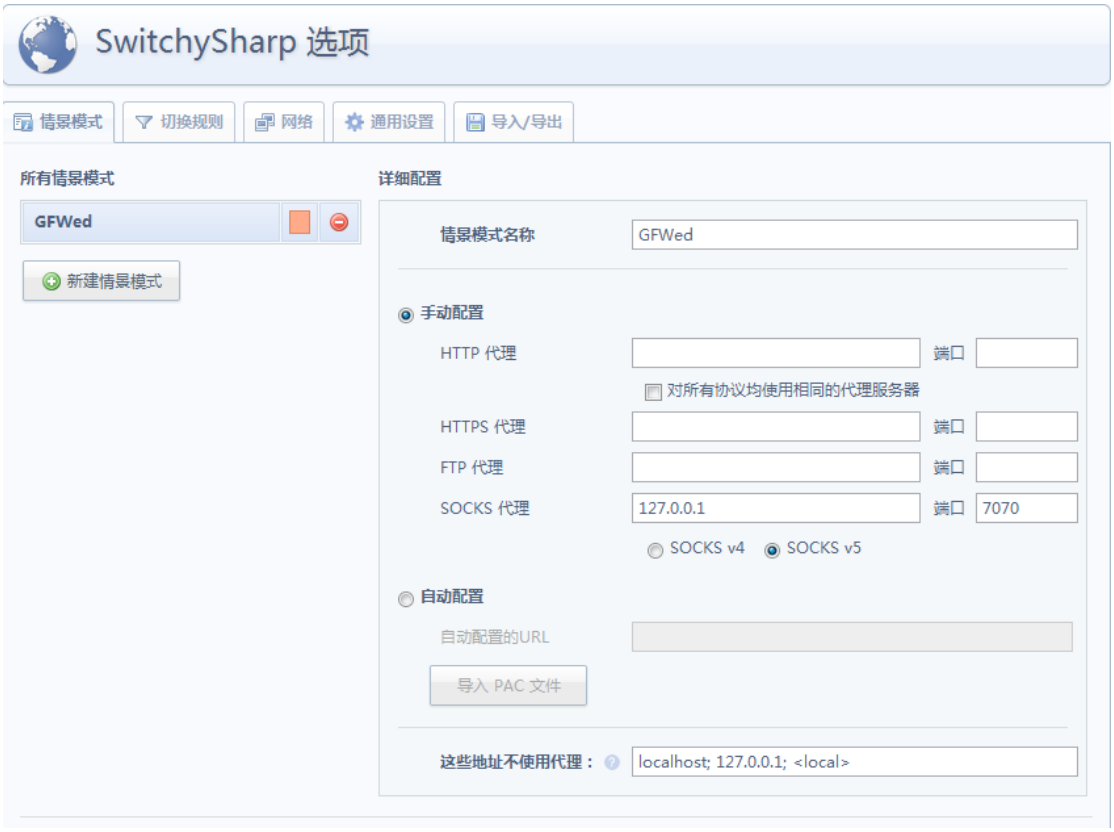


Chrome + AutoProxy
安装 Chrome
运行下面的命令，添加证书
sudo wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt-key add -
把下面的源添加到 /etc/apt/sources.list 文件中
deb http://dl.google.com/linux/deb/ stable non-free main
随后更新数据库
sudo apt-get update
安装 Chrome
sudo apt-get install google-chrome-unstable
在 Chrome 下可以利用 Switchy 中实现类似 Autoproxy 的自动规则功能。

关于 Switchy 的使用，更详细的介绍可以看此文： https://autoproxy.org/zh-CN/node/73

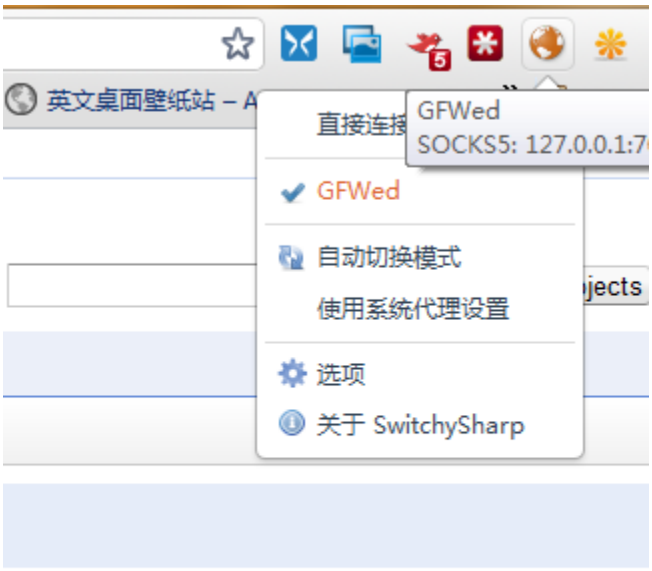
1 下载 Chrome Switchy 插件 http://code.google.com/p/switchysharp/

2 根据下图设置 Switchy 选项



3.然后在线恢复备份,备份文件地址:http://switchysharp.googlecode.com/files/X_GFWList.bak

4 最后启用 Auto Switch 模式，如下图



ProxyChains 及 Tsocks

现在我们可以用 FireFox 及 Chrome 通过 SSH Tunneling 来访问之前无法访问的网站了，但有些程序本身不支持 Socket 5 代理，那么我们可以通过 ProxyChains 及 Tsocks 这两个工具来实现。

ProxyChains

ProxyChains 是一个类似于 windows 下 freecap 的软件，可结合 ssh tunneling 功能来实现翻墙。

安装

```
sudo apt-get install proxychains
```

配置,修改 /etc/proxychains.conf

```
# proxychains.conf    VER 2.0
#
#        HTTP, SOCKS4, SOCKS5 tunneling proxifier.
#

# The option below identifies how the ProxyList is treated.
# only one option should be uncommented at time,
# otherwise the last appearing option will be accepted
#
# Dynamic - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
# at least one proxy must be online to play in chain
# (dead proxies are skipped)
# otherwise EINTR is returned to the app
#
# Strict - Each connection will be done via chained proxies
# all proxies chained in the order as they appear in the list
```

```
# all proxies must be online to play in chain
# otherwise EINTR is returned to the app
#
# Random - Each connection will be done via random proxy
# (or proxy chain, see  chain_len) from the list
# this option is good for scans

dynamic_chain
#strict_chain
#random_chain

# Make sense only if random_chain
chain_len = 2

# Quiet mode (no output)
#quiet_mode

# Write stats about good proxies to proxychains.stats
#write_stats

#Some timeouts in milliseconds
#
tcp_read_time_out 15000
tcp_connect_time_out 10000

[ProxyList]
# ProxyList format
#       type  host  port [user pass]
#       (values separated by 'tab' or 'blank')
#
#
#        Examples:
#
#               socks5  192.168.67.78 1080  lamer  secret
#       http 192.168.89.3  8080  justu  hidden
#       socks4  192.168.1.49  1080
#            http 192.168.39.93 8080
#
#
#       proxy types: http, socks4, socks5
#        ( auth types supported: "basic"-http  "user/pass"-socks )
#
#http  10.0.0.5 3128
socks5 127.0.0.1 7070
```

注意选 dynamic_chain
如何使用？
比如我们可以通过 proxychains 来运行 Twitter 客户端 pino ，如下。
proxychains pino &
就这么简单。
tsocks
安装 tsocks
sudo apt-get install tsocks
配置 tsocks，让它使用我们上面建立的 socks5 代理，用 root 权限编辑/etc/tsocks.conf，修改以下几行：
server = 127.0.0.1
# Server type defaults to 4 so we need to specify it as 5 for this one
server_type = 5
The port defaults to 1080 but I've stated it here for clarity
server_port = 7070
tsocks 的使用方法和 proxychains 差不多，也是把 tsocks 放在软件名称之前就可以了。
tsocks pino &
另外通过 proxychains 和 tsocks 来安装来自于 PPA 的软件可以起到增速的作用。
sudo tsocks apt-get install software-name