<https://github.com/shadowsocks/shadowsocks/tree/master>

<https://github.com/shadowsocks/shadowsocks-windows>

<https://shadowsocks.org/en/index.html>

<http://www.cnblogs.com/Gbeniot/p/5567187.html>

<https://xuri.me/2014/08/14/shadowsocks-setup-guide.html>

Require: python 2.7 (2.7自带pip)

Python 2.6 升级到 2.7

下载

# https://www.python.org/ftp/python/2.7.12/Python-2.7.12.tgz

# tar -zxvf Python-2.7.6.tgz

# cd

# ./configure

# make all

# make install

# make clean

# make distclean

查看版本信息

# /usr/local/bin/python2.7 -V

建立软连接，使系统默认的 python指向 python2.7 （可省略）

# mv /usr/bin/python /usr/bin/python2.6.6

# ln -s /usr/local/bin/python2.7 /usr/bin/python

解决系统 Python 软链接指向 Python2.7 版本后，因为yum是不兼容 Python 2.7的，所以yum不能正常工作，我们需要指定 yum 的Python版本

将 /usr/bin/yum 改成 python2.6

#!/usr/bin/python2.6.6

安装SS服务端：

yum update

yum install python-setuptools && easy\_install pip

pip install shadowsocks

（最好还是编译安装pip吧）

（出现No module named pkg\_resources，就pip install distribute）

# pip2.7 install shadowsocks

新建ss配置文件

vi /etc/shadowsocks.json

单用户添加以下内容：

{

"server":"0.0.0.0",

"server\_port":8848,

"local\_address":"127.0.0.1",

"local\_port":1080,

"password":"your password",

"timeout":300,

"method":"aes-256-cfb",

"fast\_open":false

}

多用户添加以下内容：

{

"server":"0.0.0.0",

"port\_password":{

"8381":"password1",

"8382":"password2",

"8383":"password3",

"8384":"password4"

},

"local\_address":"127.0.0.1",

"local\_port":1080,

"timeout":300,

"method":"aes-256-cfb",

"fast\_open":false

}

启动

ssserver -c /etc/shadowsocks.json -d start

停止

ssserver -d stop

ssserver -c /etc/shadowsocks.json -d restart

**客户端安装**

<https://www.linuxbabe.com/linux-server/setup-your-own-shadowsocks-server-on-debian-ubuntu-centos>

<http://www.linuxdiyf.com/linux/17144.html>

Kali安装 GUI Client

<http://www.jianshu.com/p/aa48a0a5b30e>

先编译安装 libqtshadowsocks-dev

在编译安装 shadowsocks-qt5

运行 ss-qt5

Firefox使用AutoProxy插件

其实shadowsocks安装时是不分客户端还是服务器端的，只不过安装后有两个脚本一个是sslocal代表以客户端模式工作，一个是ssserver代表以服务器端模式工作。

安装shadowsocks。然后执行下述命令后台启动：

nohup sslocal -s your\_server\_ip -p your\_server\_port -l 1080 -k your\_server\_passwd -t 600 -m rc4-md5 > /dev/null 2>&1 &

注意

1、使用的是sslocal这个命令，表示shadowsocks以客户端模式工作

2、将上述命令里的your\_server\_ip,your\_server\_port,your\_server\_passwd换成自己的，这三个分别代表服务器ip，服务器上shadowsocks的端口以及密码.后面的rc4-md5加密方式也要换成跟server端一致。

3、前面的nohub表示后台执行，否则将会阻塞shell端口.

为了更方便，建议新建一个.json的文件，将上述信息放里面,如新建/etc/shadowsocks.json文件，内容为：

{

"server":"your\_server\_ip", #ss服务器IP

"server\_port":your\_server\_port, #端口

"local\_address": "127.0.0.1", #本地ip

"local\_port":1080, #本地端口

"password":"your\_server\_passwd",#连接ss密码

"timeout":300, #等待超时

"method":"rc4-md5", #加密方式

"fast\_open": false, # true 或 false。如果你的服务器 Linux 内核在3.7+，可以开启 fast\_open 以降低延迟。开启方法： echo 3 > /proc/sys/net/ipv4/tcp\_fastopen 开启之后，将 fast\_open 的配置设置为 true 即可

"workers": 1 # 工作线程数

}

然后运行

nohup sslocal -c /etc/shadowsocks.json /dev/null 2>&1 &

启动shadowsocks。

**安装Privoxy**

上述安好了shadowsocks，但它是socks5代理，我们在shell里执行的命令，发起的网络请求现在还不支持socks5代理，只支持http／https代理。因此需要安装privoxy代理，它能把电脑上所有http请求转发给shadowsocks。

访问官网http://www.privoxy.org/获得Privoxy的最新源码:privoxy-3.0.24-stable-src.tar.gz,执行tar -zxvf privoxy-3.0.24-stable-src.tar.gz解压，然后cd privoxy-3.0.24-stable进去。

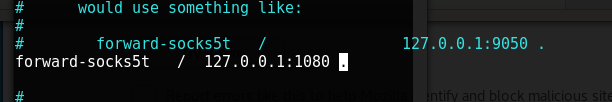
安装前需要执行useradd privoxy创建一个用户privoxy，然后依次执行如下三条命令:

autoheader && autoconf

./configure

make && make install

查看vim /usr/local/etc/privoxy/config文件，先搜索关键字:listen-address找到listen-address 127.0.0.1:8118这一句，保证这一句没有注释，8118就是将来http代理要输入的端口。然后搜索forward-socks5t,将forward-socks5t / 127.0.0.1:1080 .此句的注释去掉。



执行如下命令启动privoxy，参考官网,不同的平台对应不同的方法:

privoxy --user privoxy /usr/local/etc/privoxy/config

# apt-get install privoxy

安装好后进行配置，Privoxy的配置文件在/etc/privoxy/config

# /etc/init.d/privoxy restart

配置/etc/profile

执行vim /etc/profile,添加如下三句:

export http\_proxy=http://127.0.0.1:8118

export https\_proxy=http://127.0.0.1:8118

export ftp\_proxy=http://127.0.0.1:8118

第三句ftp的代理根据需要，不需要的话可以不添加.然后source /etc/profile，执行curl www.google.com或wget [www.google.com](http://www.google.com) 判断是否成功访问。

如果不需要用代理了，记得把bash里的配置注释上，免得把流量跑完了。