

三、部署 Shadowsocks

Shadowsocks 需要同时具备客户端和服务端，所以它的部署也需要分两步。

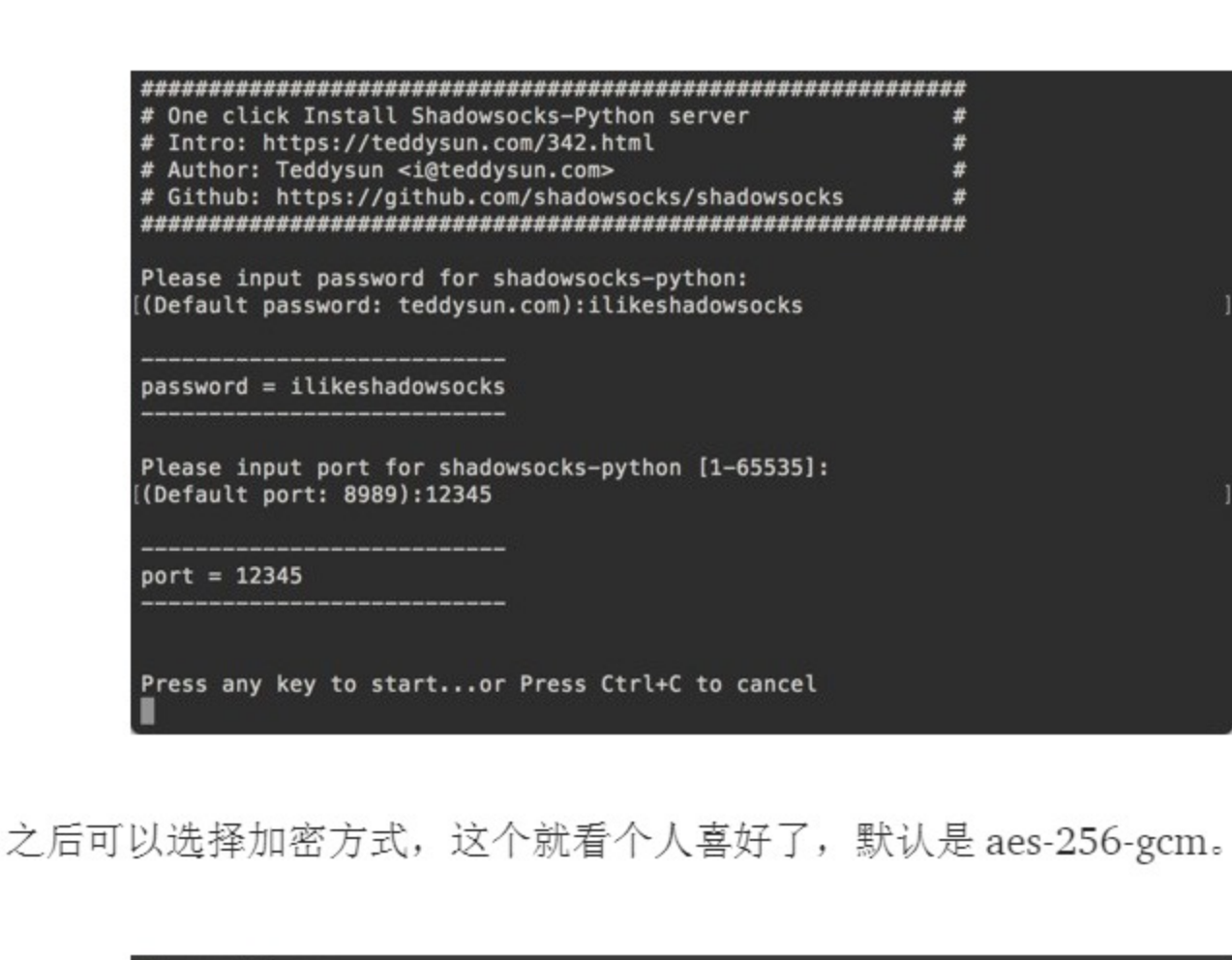
3.1部署 Shadowsocks 服务器端

这里使用 [teddysun](#) 的一键安装脚本。

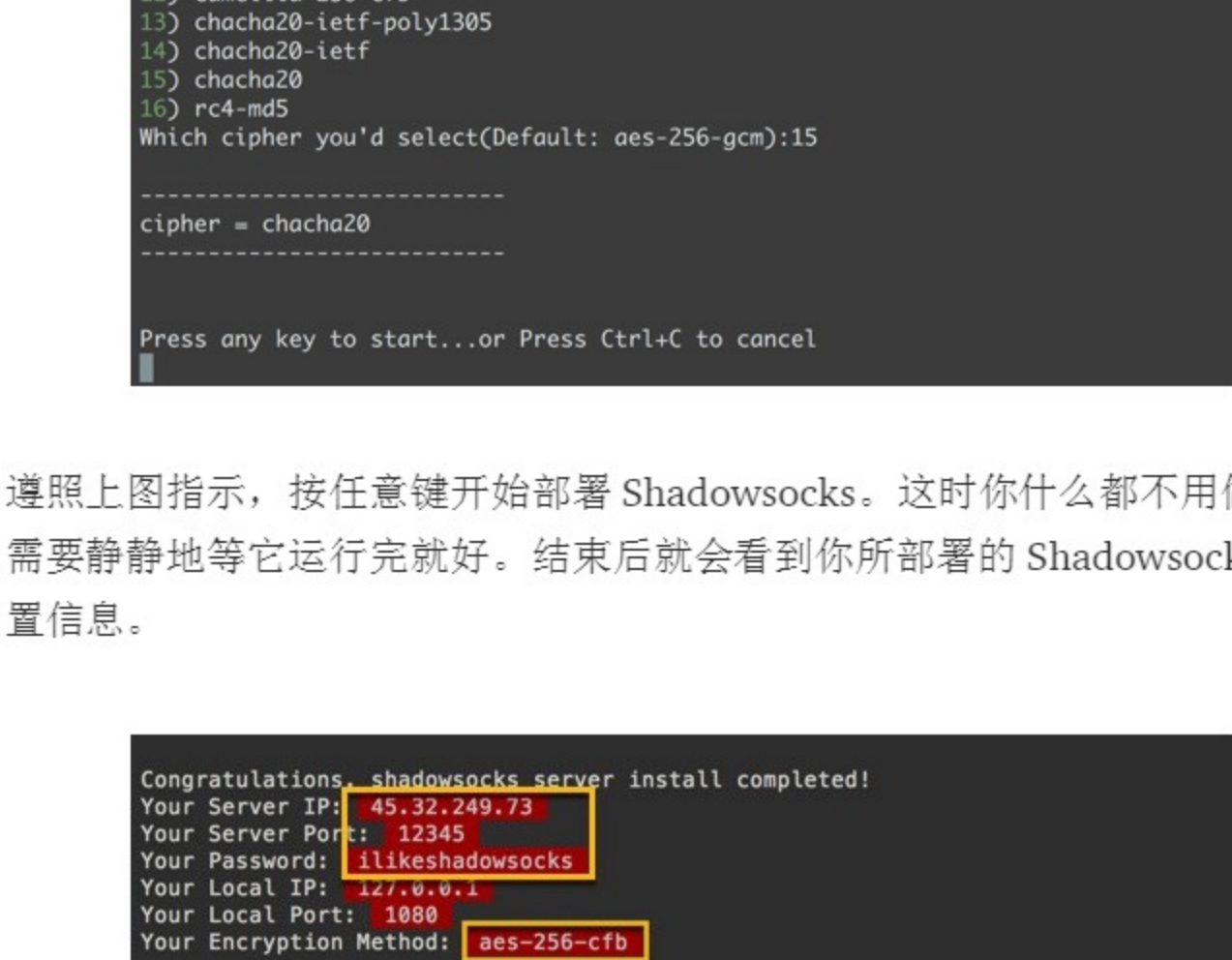
以下是3条命令，每次输入一行、回车，等待屏幕上的操作完成后再输入下一条。

```
wget --no-check-certificate
https://raw.githubusercontent.com/teddysun/shadowsocks_install/master/shadowsocks.sh
chmod +x shadowsocks.sh
./shadowsocks.sh 2>&1 | tee shadowsocks.log
```

最后一步输完，你应该会看到下图内容——是要你为 Shadowsocks 服务设置一个个人密码。



输好回车后会让你选择一个端口，输入 1-65535 间的数字都行。



之后可以选择加密方式，这个就看个人喜好了，默认是 aes-256-gcm。



遵照上图指示，按任意键开始部署 Shadowsocks。这时你什么都不用做，只需要静静地等它运行完就好。结束后就会看到你所部署的 Shadowsocks 的配置信息。



记住其中黄框中的内容，也就是服务器 IP、服务器端口、你设的密码和加密方式。

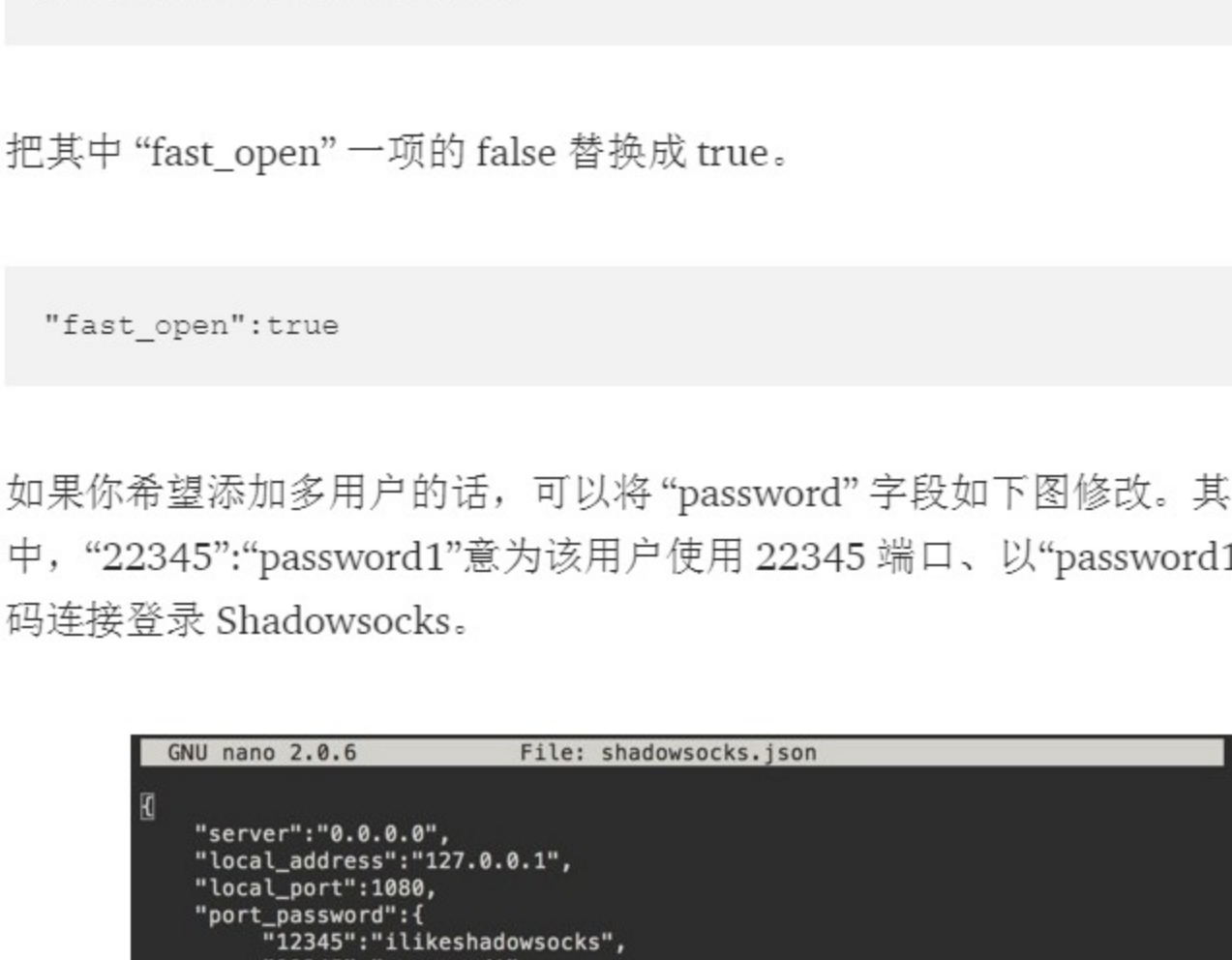
3.2 TCP Fast Open

实际上只要具备上述四个信息，你就可以在自己的任意设备上进行登录使用了。但是为了更好的连接速度，你还需要多做几步。

首先是打开 TCP Fast Open，输入以下命令，意为用 nano 这个编辑器打开一个文件。

```
nano /etc/rc.local
```

你的「终端」会刷新一下，出现下图。



别慌张，它就是个文本编辑器。用方向键把光标移到最末端，粘贴下面这一行内容，然后按 Ctrl + X 退出。

```
echo 3 > /proc/sys/net/ipv4/tcp_fastopen
```

输入“Y”并回车确认退出。



然后依法炮制，输入：

```
nano /etc/sysctl.conf
```

在文末加上下面的内容，保存退出。

```
net.ipv4.tcp_fastopen = 3
```

再打开一个 Shadowsocks 配置文件。

```
nano /etc/shadowsocks.json
```

把其中“fast_open”一项的 false 替换成 true。

```
"fast_open":true
```

如果你希望添加多用户的话，可以将“password”字段如下图修改。其中，“22345”、“password1”意为该用户使用 22345 端口、以“password1”为密码连接登录 Shadowsocks。



如果添加了多用户，需要更改防火墙规则，开放刚刚新增的端口。centOS 6 系统开放办法如下。保存退出后，输入以下命令，注意一次输入一行，并把 <newport> 替换为刚添加的端口：

```
iptables -I INPUT -m state -- state NEW -m tcp -p tcp -dport <newport> -j ACCEPT
iptables -I INPUT -m state -- state NEW -m udp -p udp -dport <newport> -j ACCEPT
/etc/init.d/iptables save
/etc/init.d/iptables restart
```

保存退出。最后，输入以下命令重启 Shadowsocks。

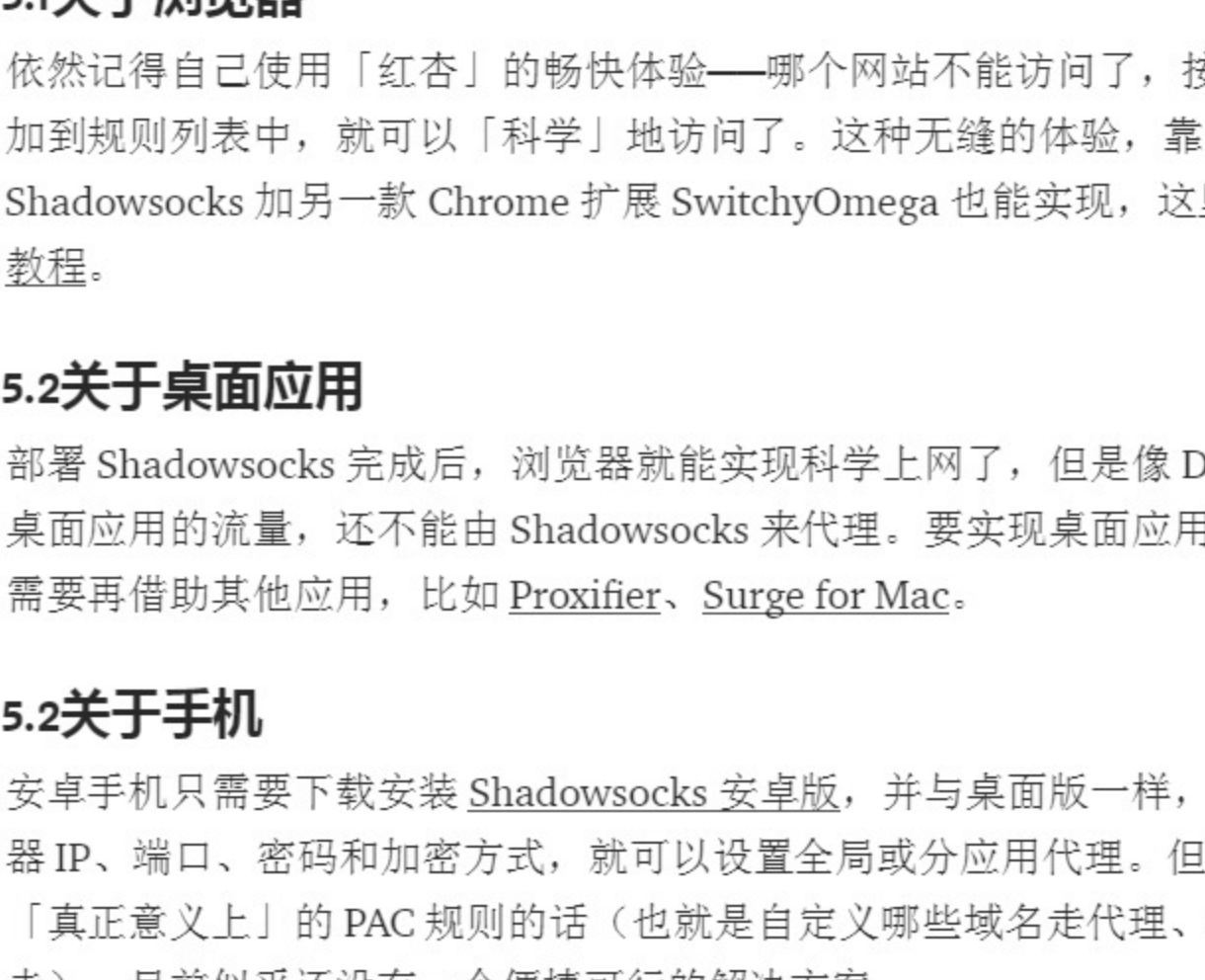
```
/etc/init.d/shadowsocks restart
```

3.3安装 Shadowsocks 客户端

相比服务器端的安装，客户端的安装就简单了许多。首先，根据操作系统下载相应的客户端。

- [Mac 版客户端下载](#)
- [Win 版客户端下载](#)

打开客户端，在「服务器设定」里新增服务器。然后依次填入服务器 IP、服务器端口、你设的密码和加密方式。



然后启用代理，就可以实现科学上网了。

四、开启 BBR

完成上述步骤后，使用过程中可能会发现连接速度有时不太令人满意。这就是加速工具们发挥功能的时候了。常用的加速工具有「锐速」、TCP BBR、

ktcpun，等等。原理虽各不相同，但都是通过一些技术手段，来实现连接和传输速度的提升。

4.1什么是 BBR

TCP BBR 是 Google 于 2016 年发布的，一种避免网络拥塞的算法。目的是要尽量跑满带宽，并且尽可能避免排队的情况。

实际使用下来，BBR 的速度提升效果并不比「锐速」（据说已倒闭）、ktcpun（多倍发包耗流量）等差。而且 BBR 已植入 Linux 4.9+ 版本的内核中，服务器开启后就能有显著的效果提升。

4.2安装 BBR

和 2.3 中的步骤一样，首先需要用 SSH 登录 VPS。

```
ssh root@<host>
```

这里依然使用 @teddysun 的一键安装脚本，输入以下命令。

```
wget --no-check-certificate
https://github.com/teddysun/across/raw/master/bbr.sh && chmod +x
bbr.sh && ./bbr.sh
```

按任意键开始安装，安装需要一段时间，等待一会即可。

安装完成后，脚本会提示需要重启 VPS，输入 **y** 并回车后重启。

重新使用 SSH 登录 VPS，这时 BBR 应该已经开启了。你可以使用以下两行命令验证一下。

```
uname -r
lsmod | grep bbr
```

如果输出的内核版本为 4.9 以上版本，且返回值有 `tcp_bbr` 模块的话，说明 bbr 已启动。

至此，整个搭建过程就大功告成了！接下来，尽情地享受自由的网络吧☺

五、一些也许对你有用的教程

5.1关于浏览器

依然记得自己使用「红杏」的畅快体验——哪个网站不能访问了，按下图标添加到规则列表中，就可以「科学」地访问了。这种无缝的体验，靠

Shadowsocks 加另一款 Chrome 扩展 SwitchyOmega 也能实现，这里是使用教程。

5.2关于桌面应用

部署 Shadowsocks 完成后，浏览器就能实现科学上网了，但是像 Dropbox 等桌面应用的流量，还不能由 Shadowsocks 来代理。要实现桌面应用的代理，

需要再借助其他应用，比如 [Proxyfier](#)、[Surge for Mac](#)。

5.2关于手机

安卓手机只需要下载安装 [Shadowsocks](#) 安卓版，并与桌面版一样，填入服务器 IP、端口、密码和加密方式，就可以设置全局或分应用代理。但是要实现

「真正意义上」的 PAC 规则的话（也就是自定义哪些域名走代理、哪些不走），目前似乎还没有一个便捷可行的解决方案。

iOS 可以通过类 [Surge](#) 的 App 实现 PAC 规则下的自动翻墙。

- [详尽版 Surge 教程](#)
- [简化版 Surge 教程](#)

Surge 目前定价 328 元，另有 Mac 版（另外收费）。作为一款兼具「科学上网」功能的网络开发调试利器，它完全值这个价。觉得太贵的话，可以使用

相对便宜的 [Wingy](#) 或者 [Potatso](#)，不过二者都没有前者来得稳定好用。