

1 综述

在饱受蓝灯挂掉之苦半个月后，终于决定自己搭建 VPS 服务器 FQ 了。虽然网上 VPS 服务器教程很多，但是我按照那些教程弄好 VPS 服务器总是不稳定，用着用着就不能用了。这应该是这次 GFW 升级带来的后遗症，所以以前的教程大多过时了。本文总结了自己在搭建 VPS 过程中遇到的各种坑，以及最后的解决方案，以供大家参考。

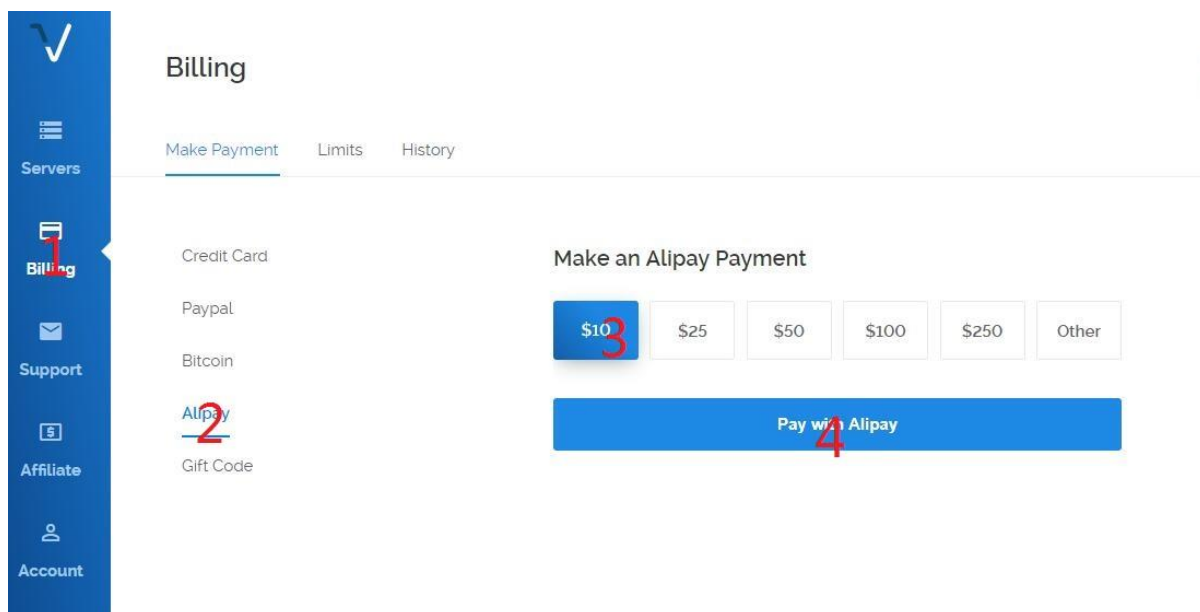
2 VPS 服务器购买

2.1 注册 vultr

我使用的服务器是 vultr。主要因为其按小时计费，随时可以停用服务器。这一点很重要，现在 GFW 相当高，很容易被墙。一旦被墙月结的损失比较大。其次，vultr 本身没有被墙，这一点很重要，不然就是先有鸡还是先有蛋的问题了。当然，可以支付宝付款也是一个加分点。vultr 注册地址：vultr.com。点击链接跳转到 vultr，网页右上角有个 Create Account 就是注册的地方了，接下来就是注册并激活账号了。

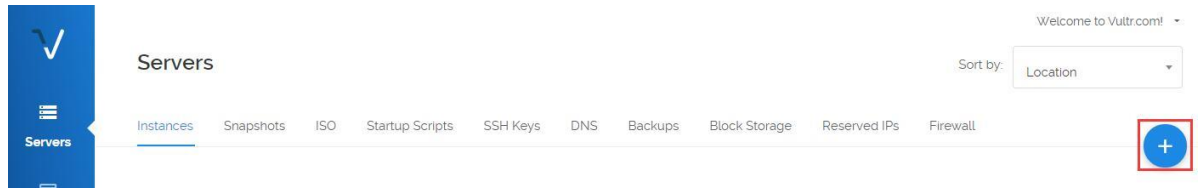
2.2 充值

在购买服务器之前需要先充值。登陆 vultr 之后如图 Billing->Alipay，选好要充值的金额数，然后 Pay with Alipay 即可跳转到支付宝扫码支付页面。



2.3 购买服务器

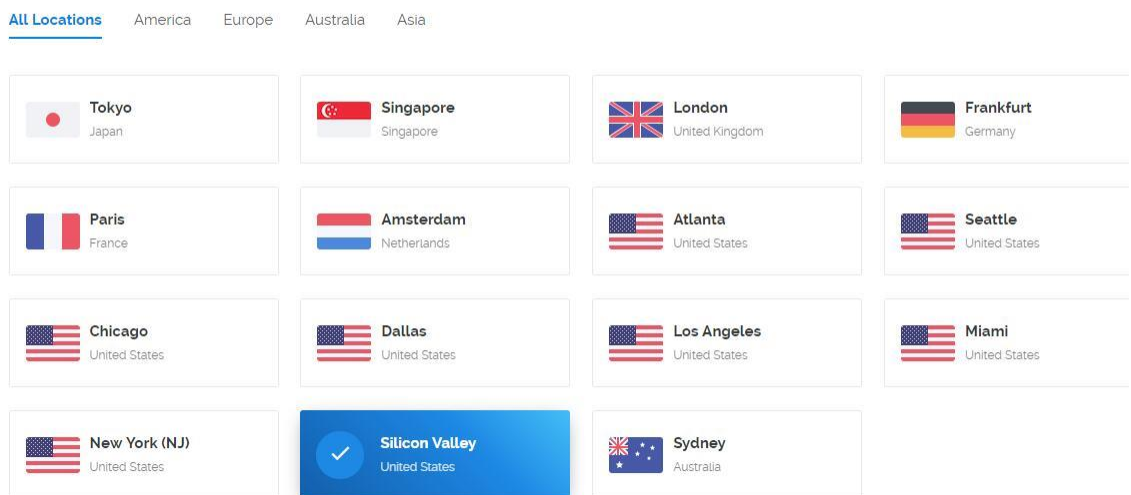
点击 vultr 网页右上角的蓝色+图标即可为账户添加服务器。



注意，向下滚动有很多选项，不要直接点击 Deploy Now，这样会使用默认参数，但不是我们想要的！

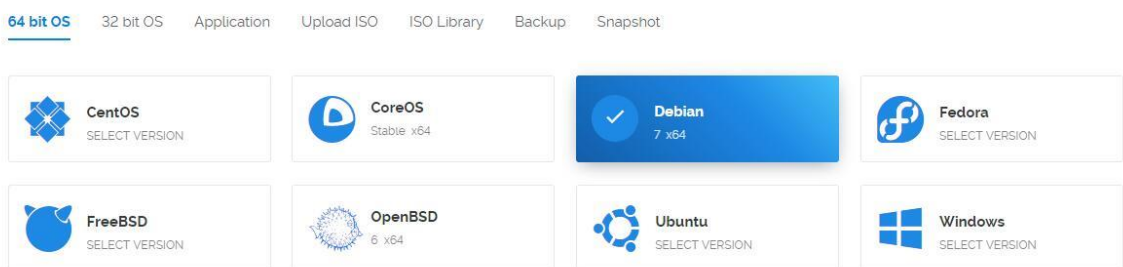
第一个选项为选择服务器地址（Server Location），建议选择硅谷（Silicon Valley，离 google 最近），而不要选择日本（用的人多，大部分 IP 已经被墙）；

1 Server Location



第二个选项为选择服务器类型（Server Type），即选择服务器操作系统。建议选择 Debian 7 x64，默认没有防火墙，会减少很多麻烦。

2 Server Type



第三个选项为选择服务器大小（Server Size），即选择服务器硬件性能。因为我们使用 VPS 服务器 FQ，所以性能瓶颈不在配置，而在网络带宽之类的。由于 vultr 的服务器带宽都是 100M 的，所以服务器当然是往便宜了选。当然，\$2.5/mo 的一般都是缺货的。选择 \$5/mo 的就好，一个月三十几块钱，要是有好几个人一起分担的话就相当实惠了。

3 Server Size

<div>Temporarily Sold Out</div> <div>20 GB SSD</div> <div>\$2.50/mo</div> <div>\$0.004/h</div> <div>1 CPU</div> <div>512MB Memory</div> <div>500GB Bandwidth</div>	<div>25 GB SSD</div> <div>\$5/mo</div> <div>\$0.007/h</div> <div>1 CPU</div> <div>1024MB Memory</div> <div>1000GB Bandwidth</div>	<div>40 GB SSD</div> <div>\$10/mo</div> <div>\$0.015/h</div> <div>1 CPU</div> <div>2048MB Memory</div> <div>2000GB Bandwidth</div>	<div>60 GB SSD</div> <div>\$20/mo</div> <div>\$0.03/h</div> <div>2 CPU</div> <div>4096MB Memory</div> <div>3000GB Bandwidth</div>
<div>100 GB SSD</div> <div>\$40/mo</div> <div>\$0.06/h</div> <div>4 CPU</div> <div>8192MB Memory</div> <div>4000GB Bandwidth</div>	<div>200 GB SSD</div> <div>\$80/mo</div> <div>\$0.119/h</div> <div>6 CPU</div> <div>16384MB Memory</div> <div>5000GB Bandwidth</div>		

第四、五、六三个选项不懂就可以不用管它。

第七条要求为即将部署的服务器命名并给一个标签，随意就好。

此时可以点击右下角的现在展开（Deploy Now），vultr 即开始分配资源，安装系统。此时 Servers 页面可以看到服务器信息，状态显示 installing。当状态切换为 Running 的时候，服务器就可以使用了。

7 Server Hostname & Label

Enter server hostname test	Enter server label test
-------------------------------	----------------------------

Servers Qty:
- 1 +

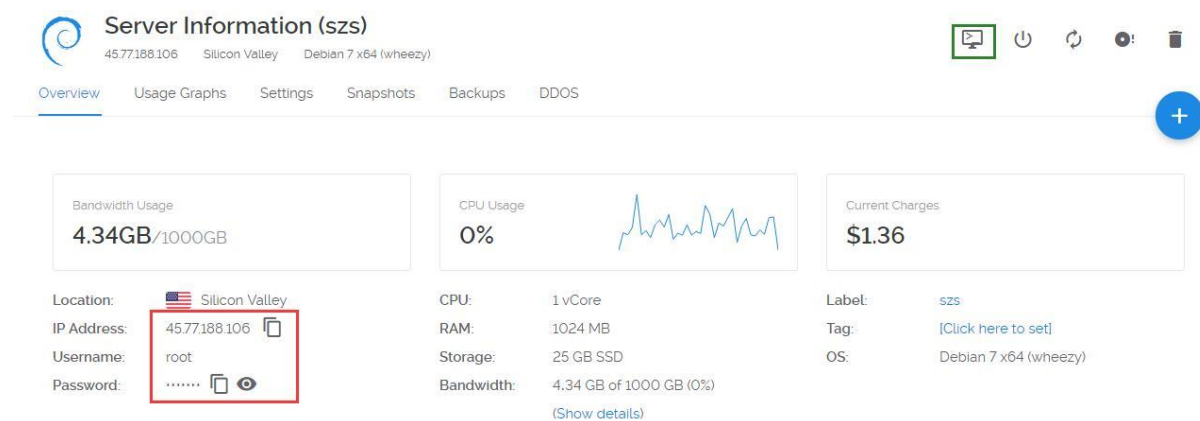
Summary:
\$5.00/mo (\$0.007/hr)

Deploy Now

3 VPS 服务器部署

3.1 服务器信息查看

服务器安装结束之后在 Servers 页面点击刚才购买的服务器名字即可跳转到服务器详情页。如图右上角有一个 View console 可以直接通过网页连接到服务器，但是由于它是老外弄的，所以不支持中文，不能使用，我们需要使用第三方工具连接到服务器。所以需要知道 IP、用户名和密码，如图左下角。

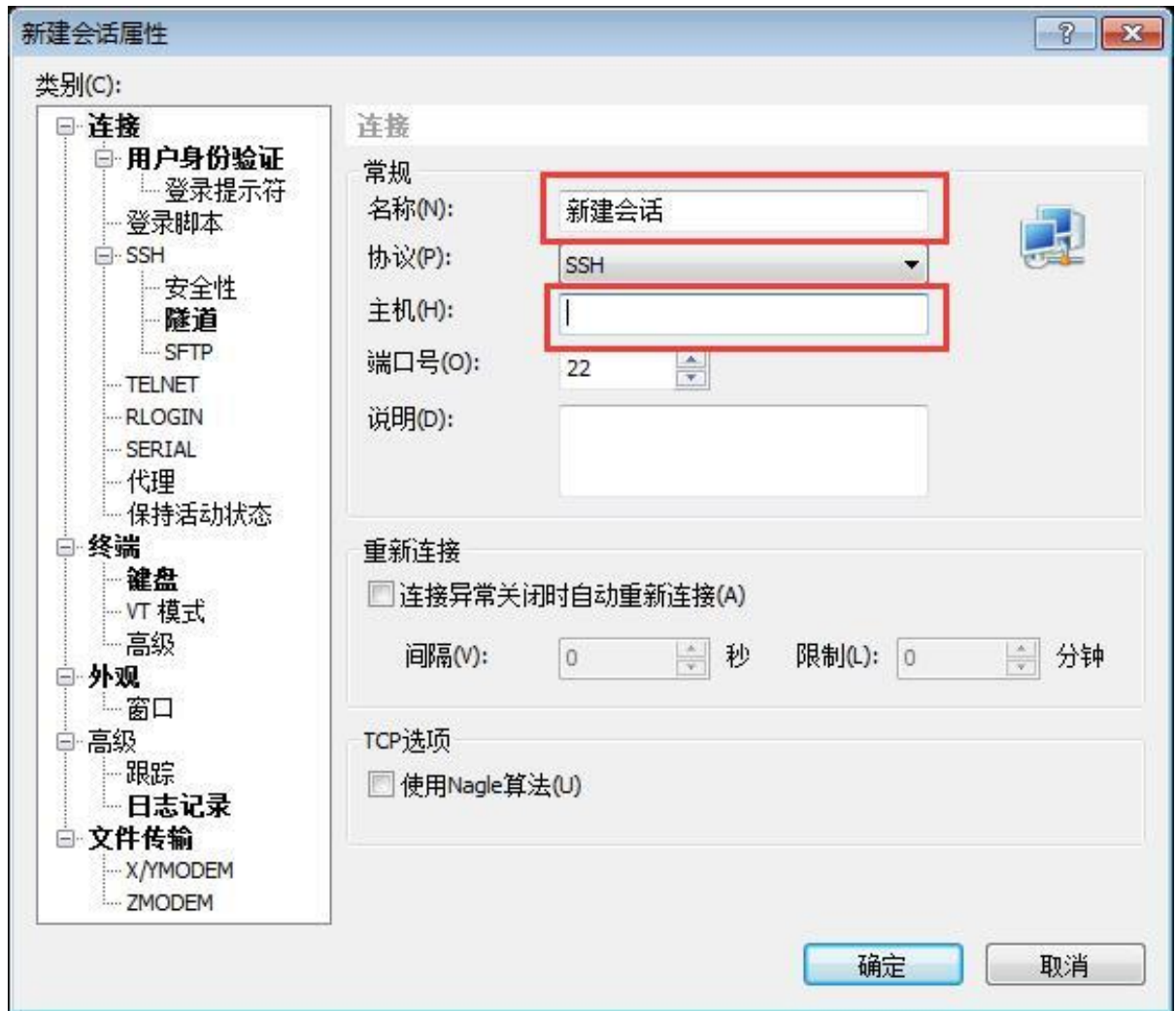


3.2 远程工具安装

这个第三方远程管理工具叫做 Xshell，现在是第五版。自己随便找个连接下载就好了，各个软件管家里面都有的。而且 Xshell 学生和家庭版是免费的，可以随便使用而不需要破解。安装好 Xshell 5 之后即可开始连接服务器进行部署了。

3.3 连接服务器

启动 Xshell，从菜单栏的文件->新建打开新建会话窗口如下图。会话名称随便取一个都可以，主机填写刚才记下来的服务器 IP 地址。点击确定立即尝试连接服务器，如果能连接上服务器会提示输入用户名和密码。用户名为 root，密码可以从 vultr 服务器详情页拷贝过来。可以记住用户名和密码，这样下次连接就不用再输入那个复杂的密码了（当然密码是可以修改的，但是没必要。系统自动生成的密码强度相当高，可以避免暴力破解。自己设的密码就不好说了，而且还容易忘记）。



连接成功之后会出现命令提示符：root@username:~#。接下来就可以开始安装 SSR 了。如果 Xshell 一直连不上服务器，那么多半是被墙了，可以尝试着 ping 一下服务器看看。如果 ping 不通的话就只能换掉服务器了（血的教训：先购买新服务器，再去 vultr 服务器页面 Destroy。反过来的话间隔太小，估计又买到原来的 ip 了）。

3.4 安装 ShadowsocksR 一键管理脚本

接下来要通过一个一键管理脚本安装 SSR，作者是在 Debian 7 x64 上开发的该脚本，这也是为什么推荐这个系统的原因之一。

首先需要安装 wget（Web Get），使得可以在命令行下载互联网文件。命令如下：

```
yum -y install wget
```

拷贝安装一键管理脚本的命令到控制台运行。以下两个命令分别为脚本命令以及备用命令（其实就是地址不一样），运行任意一个即可。然后等待脚本安装完成，期间如果有提示输入 Y 记得输入一下。注意，命令的参数包括命令后面的网址，要一起拷贝，虽然显示出来可能是两行，但其实是没换行的。

```
wget -N --no-check-certificate https://softs.fun/Bash/ssr.sh && chmod +x ssr.sh && bash ssr.sh
```

```
wget -N --no-check-certificate https://raw.githubusercontent.com/ToyoDAdoubi/doubi/master/ssr.sh && chmod +x ssr.sh && bash ssr.sh
```

上述命令运行后会提示你输入数字来选择要做什么。输入 1，就会开始安装 ShadowsocksR 服务端，并且会提示你输入 ShadowsocksR 的 **端口/密码/加密方式/ 协议/混淆**（混淆和协议是通过输入数字选择的）等参数。虽然没有必要，但是可以考虑重启一下服务器（命令：reboot）。然后输入运行脚本的命令即可开始配置 SSR 了。运行管理脚本的命令为：

```
bash ssr.sh
```

3.5 安装 ShadowsocksR

运行脚本：

```
bash ssr.sh
```

输入命令对应的数字来执行命令。

ShadowsocksR 一键管理脚本脚本 [v2.0.35]

1. 安装 ShadowsocksR
2. 更新 ShadowsocksR
3. 卸载 ShadowsocksR
4. 安装 libsodium(chacha20)

-
5. 查看 账号信息
 6. 显示 连接信息
 7. 设置 用户配置
 8. 手动 修改配置
 9. 切换 端口模式

-
10. 启动 ShadowsocksR
 11. 停止 ShadowsocksR
 12. 重启 ShadowsocksR
 13. 查看 ShadowsocksR 日志

-
14. 其他功能

15. 升级脚本

当前状态：已安装 并 已启动

当前模式：单端口

请输入数字(1-15)：

建议配置为：

端口：443

加密：chacha20

协议：auth_sha1_v4

混淆：http_simple

所以按照如下步骤配置即可（下面步骤输入的数字可能会随着版本的变化而有细微变化，以实际为准。内容太多，懒得截图了）：

1. 运行：bash SSR.sh;
2. 输入 4：安装 libsodium，chacha20 加密协议的依赖包，等待安装完成，期间如果有提示输入 Y 记得输入一下；
3. 运行：bash SSR.sh;
4. 运行安装 SSR 的命令：1. 安装 ShadowsocksR;
5. 输入端口号：443。443 端口为 HTTPS 协议的默认端口，经过测试是最稳定的，不建议使用其他端口或者使用多端口模式；
6. 输入密码：强度越大越好，反正不需要记忆；
7. 输入 15：使用 chacha20 加密方式；
8. 输入 2：使用 authsha1_v4 协议插件，输入 2 并确认后会提示是否兼容原版，原版指的是 Shadowsocks，没有必要选择兼容，填 n 后回车即可。
9. 输入 2：使用 http_simple 混淆，延时小，可靠性高，同样也不需要兼容原版；
10. 然后就是一些限制，一般默认几次回车就好了。

上面的步骤无误之后一键管理脚本就开始安装 SSR 了，过一会儿会提示下载文件，输入 y 后回车即可，然后继续等待。

安装完成之后会显示服务器配置信息。包括 ip、端口、密码、加密、协议、混淆等。当然这些都没必要记下来（如果 Linux 客户端应该是要手动输入的），配置客户端的时候只需要下面的 SSR 连接即可。

3.6 安装 BBR 和封禁垃圾端口

运行 bash SSR.sh 命令后输入 14 即可进入安装 BBR 和封禁垃圾端口的界面。这两个都按提示部署一下即可。BBR 是 google 开发的拥塞控制算法，亲测有效，安装之后 google 秒开，不安装延时相当大。

4 windows 客户端部署

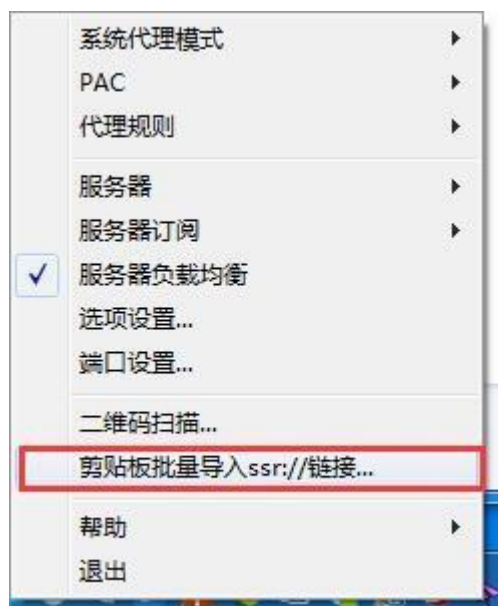
几点说明，首先 SSR 维护者也已经被请喝茶了，所以 SSR 项目已经从 GitHub 移除了，所以 SSR 的 PCA 模式已经不能使用了。其次，不要使用 360 浏览器，国内浏览器厂商和 GFW 有 py 交易的。所以，推荐使用 SSR+chrome 插件 SwitchyOmega 实现只能 chrome 翻墙的目的。这样 SSR 使用直连模式，由 chrome 插件决定当前流量是否走代理。如果你有别的用途可以单独在某个软件内部指定代理地址为 SSR 的地址：127.0.0.1:1080，如果有需要可以留言，我再看看要不要补充吧。

4.1 配置 ShadowsocksR

配置 SSR 之前请将所有安全软件以及国产浏览器关闭，否则 IP 很快就会被封掉！

ShadowsocksR 客户端下载地址 [SSRforWindows](#)。SSR 不需要安装就可以直接运行，但是会在当前目录生成许多运行时文件，所以建议拷贝 ShadowsocksR-4.7.0-win 文件夹到电脑里面，然后根据自己系统安装的 .net 情况运行 ShadowsocksR-dotnet4.0.exe 或者 ShadowsocksR-dotnet2.0.exe。为了方便自己建立一个快捷方式到桌面或者设置开机自启即可。运行 SSR 之后可以在托盘区看到一个小飞机。

复制服务器链接到剪贴板，然后右键小飞机剪贴板批量导入 ssr://链接...，如图所示：



初始的时候软件里会有一台免费的服务器，由于 SSR 作者被请喝茶，已经没有维护了，是不能用的，需要将其删除掉，然后确定。如图所示：



此时已经可以访问 google 啦！由于服务器是美国的服务器，访问 google 的时候请使用 google.com。其它后缀会慢很多。但是不要急着去玩，后面两步更加重要！！

4.2 安装 SwitchyOmega_Chromium 插件

在 Chrome 地址栏输入 chrome://extensions 打开扩展程序，点击底部获取更多拓展程序。搜索 proxy 即可找到 SwitchyOmega 插件安装之。

4.3 配置 SwitchyOmega 插件（以下每一步配置完之后记得应用选项）

在 chrome 右上角打开 SwitchyOmega 配置界面，如图所示：



情景模式中的 auto switch 配置如图所示：注意规则列表格式为 AutoProxy，其网址为：<https://raw.githubusercontent.com/gfwlist/gfwlist/master/gfwlist.txt>。填写完成之后需要先更新情景模式使其生效，如图所示：

SwitchyOmega

情景模式：auto switch

发布规则列表 导出PAC 更改名称 删除

设置

界面 通用 导入/导出 情景模式 proxy auto switch 新建情景模式... ACTIONS 应用选项 撤销更改

切换规则 编辑源代码

排序	条件类型	条件设置	情景模式	操作
↑↓	域名通配符	*.example.com	proxy	删除 置顶 重置
↑↓	域名通配符	internal.example.com	[直接连接]	删除 置顶 重置
+ 添加条件				
☰	规则列表规则	(按照规则列表匹配请求)	proxy	删除
默认情景模式			[直接连接]	↑

规则列表设置

规则列表格式 Switchy AutoProxy

规则列表网址 <https://raw.githubusercontent.com/gfwlist/gfwlist/master/gfwlist.txt>

应用将从此网址下载规则列表。如果网址留空，则以下文本会被直接处理后作为规则列表使用。

立即更新情景模式

规则列表正文

规则列表最后更新时间：2017年10月30日 下午6:30:14

情景模式中的 proxy 配置如下图所示，其中 127.0.0.1:1080 为 SSR 代理地址，只有流向这个地址的流量 SSR 才会处理。

SwitchyOmega

情景模式：proxy

导出PAC

设置

界面 通用 导入/导出 情景模式 proxy auto switch 新建情景模式... ACTIONS 应用选项 撤销更改

代理服务器

网址协议	代理协议	代理服务器	代理端口
(默认)	SOCKS5	127.0.0.1	1080

显示高级设置

不代理的地址列表

不经过代理连接的主机列表: (每行一个主机)
(可使用通配符等匹配规则...)

127.0.0.1
::1
localhost

设定中的初始情景模式选择 auto switch，如图所示。这样当检测到目标地址在 GFW 列表中的时候 chrome 会让其走 SSR 代理。否则直连。



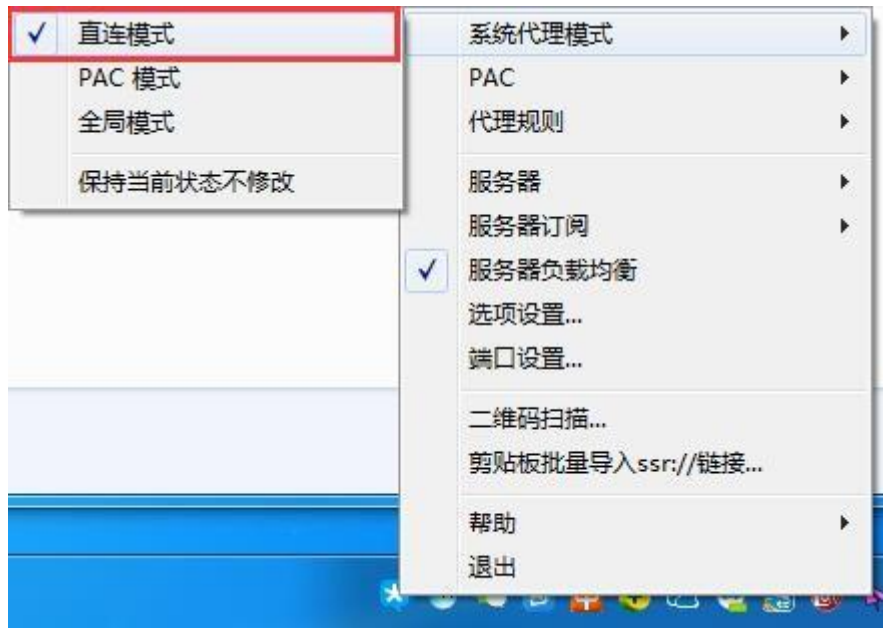
然后重启 chrome，保证 SwitchyOmega 运行在 auto switch 模式，否则可能无法正常使用，常见现象有无法访问国外网站、导致访问国内网站速度变慢或无法访问内网等。如图所示：



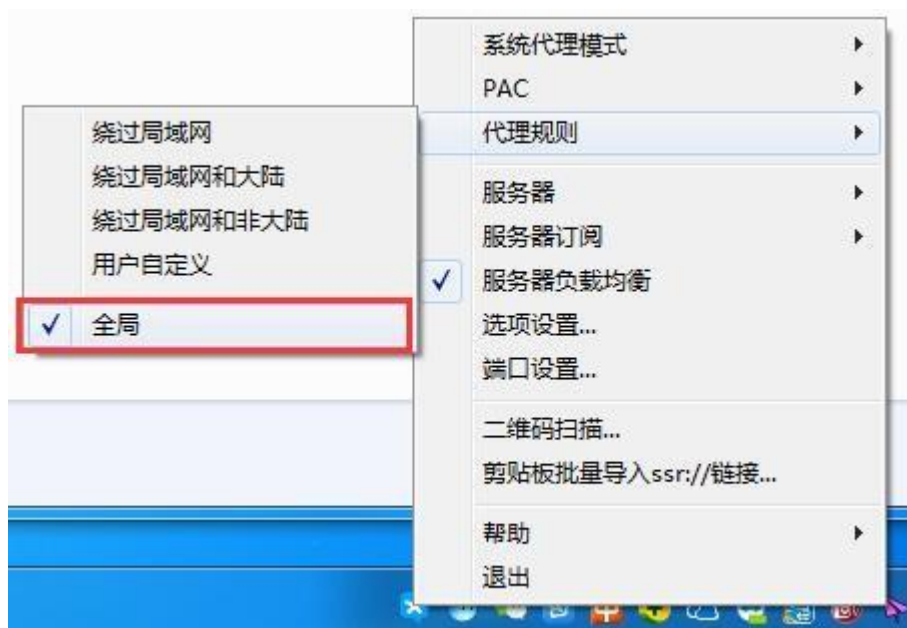
4.4 修改 SSR 参数

刚才没有在配置好 SSR 之后修改参数的原因在于如果改了参数就访问不了 google 了，这样安装不了 SwitchyOmega 插件（其实可以去 GitHub 下载安装）。所以刚配置好 SSR 的时候使用了全局模式，保证能访问 google，但是这样流量会被国内软件检测到，所以让关闭所有国内软件了。现在要将 SSR 参数修改为直连模式，等待被 chrome 调用即可。步骤如下：

右键小飞机系统代理模式改为“直连模式”，如图所示：



右键小飞机代理规则改为“全局”，如图所示：



至此 SSRwindows 客户端已经配置完成，尽情享受吧！！

可以打开安全软件和杀毒软件了！！

5 其他客户端部署指南

Linux 客户端一键安装配置脚本：[SSRforLinux](#)；

Linux 客户端 GUI 版本：[SSRforLinuxGUI](#)；

MAC 客户端：[SSRforMAC](#)；

Android 客户端: [SSRforAndroid](#)。