

由于Ubuntu Server是不带用户界面的，所以要为Server配置Shadowsocks还是稍显麻烦。本文就是我为配置Shadowsocks的一些经验，以待参考。

## 安装shadowsocks

由于shadowsocks是基于python开发的，所以必须安装python：

```
sudo apt-get install python
```

接着安装python的包管理器pip：

```
sudo apt-get install python-pip
```

安装完毕之后，通过pip直接安装shadowsocks：

```
sudo pip install shadowsocks
```

## 配置shadowsocks

新建一个配置文件 `shadowsocks.json`，然后配置相应的参数：

```
{
  "server": "your_server",
  "server_port": your_port,
  "local_port": 1080,
  "password": "your_password",
  "timeout": 300,
  "method": "aes-256-cfb"
}
```

上面的参数需要你的shadowsocks服务提供商为你提供，当然你也可以[自己搭建一个](#)。

配置完成后就可以启动shadowsocks服务：

```
sslocal -c shadowsocks.json -d start
```

## 配置全局代理

启动shadowsocks服务后，发现并不能翻墙上网，这是因为shadowsocks是socks 5代理，需要客户端配合才能翻墙。

为了让整个系统都走shadowsocks通道，需要配置全局代理，可以通过polipo实现。

首先是安装polipo：

```
sudo apt-get install polipo
```

接着修改polipo的配置文件 `/etc/polipo/config`:

```
logSyslog = true
logFile = /var/log/polipo/polipo.log

proxyAddress = "0.0.0.0"

socksParentProxy = "127.0.0.1:1080"
socksProxyType = socks5

chunkHighMark = 50331648
objectHighMark = 16384

serverMaxSlots = 64
serverSlots = 16
serverSlots1 = 32
```

重启polipo服务:

```
sudo /etc/init.d/polipo restart
```

为终端配置http代理:

```
export http_proxy="http://127.0.0.1:8123/"
```

接着测试下能否翻墙:

```
curl www.google.com
```

如果有响应, 则全局代理配置成功。

## 注意事项

服务器重启后, 下面两句需要重新执行:

```
sudo sslocal -c shawdowssocks.json -d start
export http_proxy="http://127.0.0.1:8123/"
```