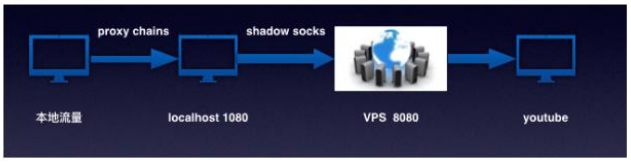
# Linux　配置　SSR

好多人都用过代理，但可能大部分却都不太了解代理是如何工作的，我在这里给大家简单介绍一下，不涉及底层知识，相信很容易理解！




```
首先通过proxychains 将本地流量发往localhost 的 1080 端口
localhost 的1080 端口和VPS 的8080 端口通过shadowsocksr建立ssr连接
本地流量 -> localhost 1080 -> VPS 8080 -> 目标站点
```

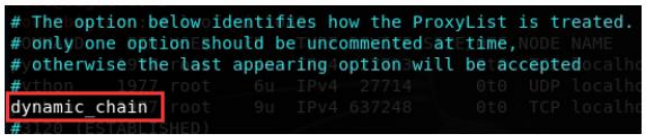
以上就是ss 代理的流程，是不是感觉很简单，下面我给大家演示一下如何配置

# 第一步

## * 配置 proxychains

```
vim /etc/proxychains.conf
```

将dynamic_chain前面的注释去掉，再将[ProxyList]下的socks4 改为socks5，并且127.0.0.1后面的端口改为1080





# 第二步

## * 安装SSR

之前使用的ss-qt5感觉老出问题，比较麻烦，而且配置比较复杂，这个ssr配置简单

下载SSR脚本:

```
wget https://onlyless.github.io/ssr
sudo mv ssr /usr/local/bin
sudo chmod 766 /usr/local/bin/ssr
```

安装SSR:

```
ssr install
```

配置SSR:

```
ssr config
```

```
{
    "server": "0.0.0.0",    //服务器IP地址
    "server_ipv6": "::",
    "server_port": 2333,   //端口
    "local_address": "127.0.0.1",
    "local_port": 1080,

    "password": "password",   //密码
    "method": "aes-256-cfb", //加密方式
    "protocol": "auth_aes128_md5", //协议
    "protocol_param": "",
    "obfs": "plain",   //混淆方式
    "obfs_param": "",
    "speed_limit_per_con": 0,
    "speed_limit_per_user": 0,

    "additional_ports" : {},
    "additional_ports_only" : false,
    "timeout": 120,
    "udp_timeout": 60,
    "dns_ipv6": false,
    "connect_verbose_info": 0,
    "redirect": "",
    "fast_open": false
}
```

# 第三步

配置完之后保存，ssr就会自动启动，查看本地1080端口：

```
root@tony:~# lsof -i:1080
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
python  1763 root    4u  IPv4  29387      0t0  TCP localhost:socks (LISTEN)
python  1763 root    6u  IPv4  29388      0t0  UDP localhost:socks
root@tony:~#
```

设置浏览器或系统代理


# SSR 的启动和关闭方式

ssr的启动和关闭方式为：

```
ssr start
ssr stop
```

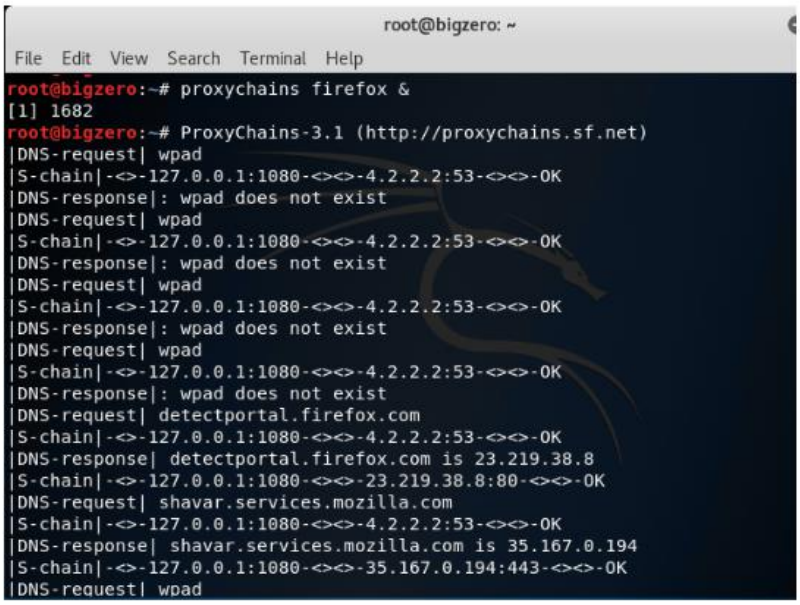
使用浏览器测试

然后执行proxyresolv www.google.com

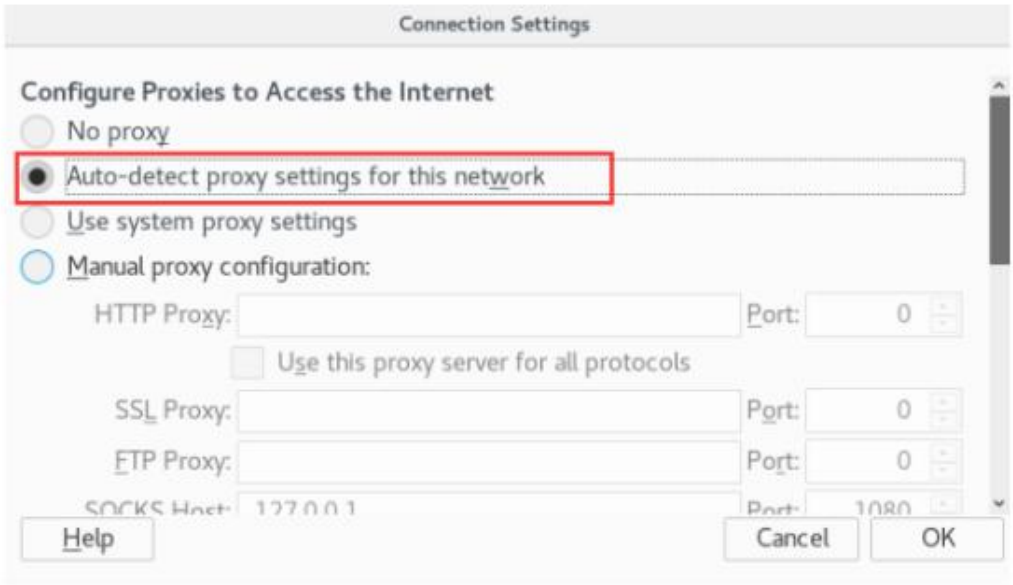
```
root@bigzero:~# proxyresolv www.google.com
|S-chain|-<>-127.0.0.1:1080-<><>-4.2.2.2:53-<><>-OK
172.217.11.68
```

说明没问题，使用proxychain firefox命令启动火狐浏览器：

```
root@bigzero:~# proxychains firefox
```



火狐浏览器设置代理：




然后就可以打开 google 测试一下了