Payload Types in the Metasploit Framework

Expanding on Payload Types in Metasploit

We briefly covered the three main payload types: **singles, stagers and stages**. Metasploit contains many different types of payloads, each serving a unique role within the framework. Let's take a brief look at the various types of payloads available and get an idea of when each type should be used.

Inline (Non Staged)

- A single payload containing the exploit and full shell code for the selected task. Inline payloads are by design more stable than their counterparts because they contain everything all in one. However some exploits wont support the resulting size of these payloads.

Stager

- Stager payloads work in conjunction with stage payloads in order to perform a specific task. A stager establishes a communication channel between the attacker and the victim and reads in a stage payload to execute on the remote host.

Meterpreter

- Meterpreter, the short form of Meta-Interpreter is an advanced, multi-faceted payload that operates via dll injection. The Meterpreter resides completely in the memory of the remote host and leaves no traces on the hard drive, making it very difficult to detect with conventional forensic techniques. Scripts and plugins can be loaded and unloaded dynamically as required and Meterpreter development is very strong and constantly evolving.

PassiveX

- PassiveX is a payload that can help in circumventing restrictive outbound firewalls. It does this by using an ActiveX control to create a hidden instance of Internet Explorer. Using the new ActiveX control, it communicates with the attacker via HTTP requests and responses.

NoNX

- The NX (No eXecute) bit is a feature built into some CPUs to prevent code from executing in certain areas of memory. In Windows, NX is implemented as Data Execution Prevention (DEP). The Metasploit NoNX payloads are designed to circumvent DEP.

Ord

- Ordinal payloads are Windows stager based payloads that have distinct advantages and disadvantages. The advantages being it works on every flavour and language of Windows dating back to Windows 9x without the explicit definition of a return address. They are also extremely tiny. However two very specific disadvantages make them not the default choice. The first being that it relies on the fact that **ws2_32.dll** is loaded in the process being exploited before exploitation. The second being that it's a bit less stable than the other stagers.

IPv6

- The Metasploit IPv6 payloads, as the name indicates, are built to function over IPv6 networks.

Reflective DLL injection

- Reflective DLL Injection is a technique whereby a stage payload is injected into a compromised host process running in memory, never touching the host hard drive. The VNC and Meterpreter payloads both make use of reflective DLL injection. You can read more about this from Stephen Fewer, the creator of the reflective DLL injection method. [Note: This site no longer exists, and is linked to for historical purposes]

Now that we have an understanding of what a payload is, payload types, and when to use them, let's generate some payloads.