Project 0 - Starting with the Basics

Nipun Shrivastava 2011cs50288

SIL765 - NETWORK & SYSTEM SECURITY

For this assignment I created two virtual machines with Kali Linux running on one of it and the other one with Metasploitable, which is virtual machine based on Linux that contains several intentional vulnerabilities to exploit.

For this assignment I used the Metasploit Framework which is an auditing tool. It has a wide array of commercial grade exploits and an extensive exploit development environment, all the way to network information gathering tools and web vulnerability plugins. Nessus, which is a tool used to scan for vulnerabilities in an operating system and wireshark, a free and open-source packet analyser were also used as per the project requirement.

Besides the above mentioned tools, I also tried my hands on tools like Nexpose. But due to my system limitations I wasn't able to use them successfully.

Upon running a scan using Nessus with the Metasploitable virtual machine as the target on a host only network, I was able to find 110 vulnerabilities, out of which 6 had a critical threat level, 2 had high and 6 more had medium threat level. The critical vulnerabilities can be and were used to hack the Metasploitable operating system. Some of the exploits that I used are given below:

Vsftpd Smiley Face Backdoor:

Using ***exploit/unix/ftp/vsftpd_234_backdoor***

The version of vsftpd running on the remote host has been compiled with a backdoor. Attempting to login with a username containing ☺ (a smiley face) triggers the backdoor, which results in a shell listening on TCP port 6200. The shell stops listening after a client connects to and disconnects from it. This vulnerability was used to perform an unauthenticated remote attack. Once successful the attacker can execute arbitrary code as **root**.

Packet Analysis using WireShark:

After going through the packets that were exchanged during the attack happened, I realised that my Metasploit framework connected to the vulnerable Metasploitable over a FTP connection and then sent the string Ea2☺ as the user name and some string as the password. Judging from the multiple exploit attempts, I was able to deduce that the password string was a random string sent by the Metasploit framework. The ☺ in the username as mentioned above triggered a backdoor entry and game me the root access in the vulnerable OS.

Some other exploits that I tried but didn't analyse in detail were:

1. **Samba NDR MS-RPC Heap-Based Remote Buffer Overflow:**
   exploit/unix/misc/distcc_exec: Samba is a widely used open-source implementation of Server Message Block (SMB)/Common Internet File System (CIFS). Network Data Representation

(NDR) is the scheme to encode MS-RPC data for transport. Samba fails to properly validate MS-RPC packets. Specifically, Samba's NDR functions do not properly validate arguments supplied to memory allocation routines. This results in a buffer of insufficient size being allocated. When data is copied to this buffer, a heap-based buffer overflow may occur and hence, a remote attacker may be able to execute arbitrary code.

2. ***exploit/multi/samba/usermap_script*** Samba contains a flaw that may allow a malicious user to execute arbitrary shell commands. The issue is triggered due to MS-RPC does not properly check user-supplied input when passing RPC messages from external scripts to '/bin/sh'. It is possible that the flaw may allow code execution resulting in a loss of integrity. The root cause is passing unfiltered user input provided via MS-RPC calls to /bin/sh when invoking externals scripts defined in smb.conf. By specifying a username containing shell meta characters, attackers can execute arbitrary commands.

3. **UnrealIRCd Backdoor Detection**

   exploit/unix/irs/unreal_ircd_3281_backdoor: Checks if an IRC server is backdoored by running a time-based command (ping) and checking how long it takes to respond. The irc-unrealircd-backdoor.command script argument can be used to run an arbitrary command on the remote system. Because of the nature of this vulnerability (the output is never returned) we have no way of getting the output of the command. It can, however, be used to start a netcat listener. In addition to running arbitrary commands, the irc-unrealircd-backdoor.kill script argument can be passed, which simply kills the UnrealIRCd process.