|  |  |  |
| --- | --- | --- |
| Course # | James Riley Dorough | Target Network System Prep |
| CSIS 462 | **8MAR22** | **Semester Week 9, Lab 5** |

**Title**

* Demonstrate all progress in penetration testing attempts and progress up to this point

**Introduction**

There have been multiple penetration testing objectives to meet through the course. Of these, some focuses are information gathering and exfiltration, persistence improvement, covert operations, and obfuscation. Each of these topics should help support the larger focus of maintaining persistence within blue team networks. Continuous work has been made to utilize new TTPs, tools, and vulnerabilities. This document outlines the progress made in red team efforts through effectively the first half of this semester.

**Initial Progress**

The initial step in preparing for penetration testing was to inspect our (red team’s) clone of the blue team’s individual networks. From here I can begin staging intrusion against blue team systems. I have been partially responsible for the Windows 10, Windows 7, and Manjaro boxes. However, due to available time I have only been able to operate against one box at a time. I have focused all my efforts on Windows 10 to learn PowerShell, Windows architecture, and Windows exploits & vulnerabilities. As my focus is centered on a single OS I can design scripts with higher complexity than I would should I have to stretch my limited red teaming time across three separate boxes.

**Red Team Operations**

**Early Exploitation**

My first malicious attempt was generating obfuscated code to ensure that all local users were available as administrators. Noted below is my first attempt at my simple script which was converted to a binary using the ps2exe tool and inserted into blue team Windows 10 machines.

Graphical user interface, text

Description automatically generated

Converting to a binary and running from an active account would show a display of outcome. However, running in the background as a system account would keep the operations hidden.

A screenshot of a computer

Description automatically generated with medium confidence

To stage all my future exploits and malware as separate from my box, I utilized our second firing station box as its external IPs are NATed. To do this I established a constant ssh tunnel through it to return all blue team traffic through a different IPv4 address than my Kali box.

Text

Description automatically generated

My first intrusion attempts are through Metasploit’s psexec module for Windows. This was a highly beneficial utility toward the beginning of the semester. However, I currently must research what allows this exploit to work and use other methods of entry to re-enable this exploit.

A picture containing graphical user interface

Description automatically generated

Running this exploit with valid credentials will return a live meterpreter session which can be migrated and hidden among currently running processes.

Text

Description automatically generated with medium confidence

Text

Description automatically generated

Text

Description automatically generated

This upside to using meterpreter through Metasploit console is that you can activate the multi handler and run and swap between multiple meterpreter sessions.

Graphical user interface

Description automatically generated with medium confidence

Additionally, you can use the “impacket-psexec” command. However, I believe that the target system requires the same services to exploit as Metasploit’s psexec payload.

Text

Description automatically generated

Early within my attempts I also had access to nearly all hidden shares on target systems. This was helpful to quickly transport malicious files between systems. After blue teams began securing their hidden shares or turning them off, I began focusing on using curl to retrieve files from a simple HTTP server running on my kali box hosting a hidden share on my development Windows 10 box. This meant that if I used a malicious command and control (C2) implant or the console backdoor, I could easily transfer my malware to their systems.

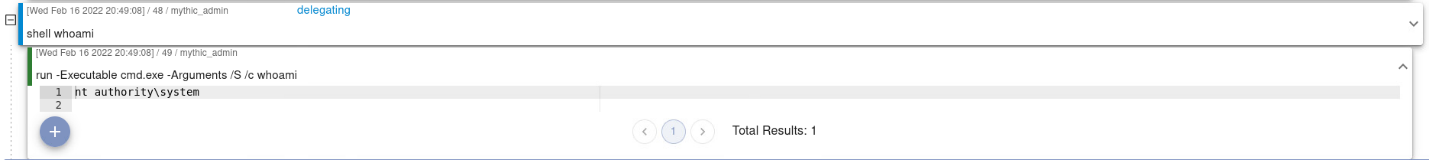
**Command & Control**

To maintain persistence, the use of (C2) servers were used. We have two operating C2 servers – Firing Station one and Firing Station two. Firing Station one is used for high-end persistence and reaming covert. Firing Station two provide the NATed IP addresses for all visible traffic. Thusly, Firing Station two is used to take all focus from other persistence methods. Using the C2 servers we can install implants onto target boxes to issue commands from our servers. I used a service manager to run my implant application on the blue team networks.

Graphical user interface, application

Description automatically generated

These implants run as the authority SYSTEM user, allowing full access to the target boxes.



**Enumeration**

To improve the Red team’s effectiveness we need actionable information from blue team systems. I had focused on gathering local user information from blue Windows 10 boxes. I used custom PowerShell scripts to gather user account information from active local user accounts and provide a list of any of these users that have blank passwords set.

Text

Description automatically generated

Application, table

Description automatically generated

Application, table

Description automatically generated

In this endeavor, I had attempted to obfuscate the intent of the script by encoding the CSV file. However, I could not figure out how to encode the newline characters to base 64. The rest of the program works well. I could syntax check the file and manually inserts newlines after recovering the output file.

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

With this focus on access to users, I have also added my own user to all Windows 10 boxes with administrator rights.

Text

Description automatically generated with medium confidence

I have avoided using this account so that it can hopefully remain hidden until we begin focusing on more destructive and noisy activity. To ensure that local users remain easily abusable with administrator privileges, I have created a binary that downloads a task scheduler XML file to import into the target system’s task scheduler and download the binary that the task is to execute. It should run every day at 0300 and the second binary “DefenderSvcCheck” sets all local accounts to administrators.

Graphical user interface, text, email

Description automatically generated

As a test, the first binary that creates the custom task also runs the target executable once to show that it works. From the perspective of the blue team, they won’t see this output.

Graphical user interface, text

Description automatically generated

Here you can see the live task waiting to run at 0300 the next morning.

Text

Description automatically generated

**Conclusion**

I have consistently learned new topics, capabilities, vulnerabilities, OS architectures, and methods of remaining covert every day I practice on the blue team systems. With as many different teams using different TTPs of defense I have gained a better understanding of what attack methods are effective given various levels of competence in defense. I anticipate that some of the more effective blue teams will notice my activity sooner than others or will notice some of my blunt, noisier attacks. I plan to continue to develop new scripts and tools to implant on their systems for me to exploit and them to find (eventually).