|  |  |  |
| --- | --- | --- |
| Course # | James Riley Dorough | Covert Persistence |
| CSIS 462 | **24MAR22** | **Semester Week 12, Lab 7** |

Title

* Detail additional covert persistence operations not listed in previous reports

Introduction

In this lab, I have outlined some additional actions I am taking to install better covert persistence on blue team networks. I have utilized the new C2 payloads available on our FiringStation. These payloads, merlin specifically, should utilize about 0-1%~ CPU usage instead of around 60%~ such as Apollo. I also attempt to manipulate my binaries to mask themselves as legit Windows applications and services.

Method

* First, the C2 payload binary needs to be created

Graphical user interface, text, application, email

Description automatically generated

* Next, you need to strip a copy of NSSM of its metadata and replace the metadata with a legit windows binary’s metadata

Graphical user interface, text, application

Description automatically generated

* After the service manager is configured, strip your C2 payload’s metadata and replace it with another legit Windows application’s data
  + It is recommended that you find a running service that has a child executable as well so that you can either create a visually similar copy or use similar naming conventions

A screenshot of a computer

Description automatically generated with medium confidence

* Once both the service manager and C2 payload are configured, you need to utilize the CLI to operate the service manager’s functionality as the altered version is now very buggy and does not have a GUI
* This can be accomplished by writing a PowerShell script to manage the full installation process in a blue team system

Text

Description automatically generated

* Even though you receive error messages for 95% of your actions, most of the commands still function fine
* After running the script, you should have a running service managing the C2 payload that looks like a legit Windows service

Graphical user interface

Description automatically generated

Results

This is not a perfect solution to ensure that your malware is fully undetectable. However, it does make it much harder to notice unless you know the Windows OS very well, understand that this service shouldn’t be running twice, or catch that the task manager lists the parent process as a blank named service.

Graphical user interface, table

Description automatically generated

It should remain covert long enough to complete the rest of the semester’s penetration testing. Once all of the configuration and coding have been completed. You simply need to transfer the installation PowerShell script to the target device and run it. Every additional step should be completed by the script, and you should now have a new callback from that device.

Conclusion

In this lab, we made attempts to bolster our persistence, possibly for the last time this semester. This should net us access to the blue team networks long enough to finish any other operations we are working on before moving on to final reporting and divulging of information. It is possible that some teams may discover this as they scan their systems for their final incident response reports. I intend on saving an explanation of this operation for our second presentation to the blue team.