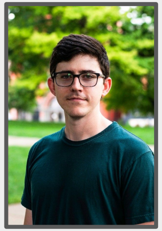
Malicious Honeypot

Team Members



Nick Marquardt - CS

John Velten - CS

John Franco - Faculty Advisor

Challenges

-Logging system: We needed to create a system to record server sessions without the user knowing. This involved using an **SFTP** connection to read from server logs.

-Linux Scripting: We knew very little about **Linux** commands beyond the basics, let alone writing scripts to do jobs. This required heavy research on **Linux** scripting.

-Orchestrating an attack: We have to consider how a malicious hacker would think when attempting to break into a system. Doing something too easy would set them off to the **Honeypot**. This required research into the common methods that hackers use.

Purpose

Honeypots are a useful tool to learn about how attackers are trying to gain access to systems that they should not have access to. We wanted to implement the basic functionality **Honeypot** with malicious characteristics. Our ultimate goal will be to gain access to the attacker's system by using the data recorded from their server session. This is probably illegal, but this is purely for learning purposes and to practice ethical hacking.

Design Decisions

-Types of Honeypots: There are several types of **Honeypots** with different levels of intractability. We decided to make an **SSH Honeypot** with a **medium** level of intractability. The intractability level refers to the privileges the attacker will have when they gain access to the system.

-Language: We decided on **Python** as our primary language to use because it is versatile and has a large number of external libraries that might come in handy. It also has the OS Module which is perfect for what we need to do.

-Front End: Technically we could have done without a GUI and just looked at the logs and records ourselves, but it was nicer to have an Admin app that could pull logs, and show live server sessions all in one place. The GUI was inspired by the Minecraft Server GUI.

-Back End: The **Honeypot** was set up on a **Raspberry Pi** computer with a **Linux** based operating system (**Raspbian**). We added some extra scripts to some of the existing processes on the system to help log sessions.

Accomplishments

-Created **GUI** to pull all server logs to view in one location
-Created **notification system** to let admin know that there is an unauthorized user on the server
-Learned about cyber security and **Linux** scripting

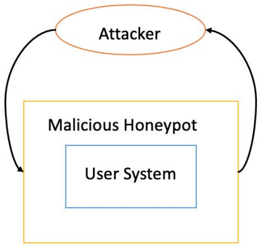
Future Development

-Automate information collection and counterattack
-Refine system security to minimize loopholes for attacker
-Hide background activity from the attacker

Technologies

SSH

Design Diagram


Attacker
Malicious Honeypot
User System