CS 395

Ransomware Lab Handout

JT Reagor

4/19/2023

Introduction

You are a free-lance cybersecurity contractor, renowned for your ability to triage ransomware attacks with ease. A piece of ransomware is a type of malware that encrypts important files on a file system. Attackers will encrypt a file system then hold the private decryption keys for ransom. When the victim has paid up, only then will they hand over the decryption keys.

Because of your famous skillset, you have been contacted by TIC Corp, a recent ransomware victim. Below is the memo you received.

> Dear Mr. Cybersecurity Professional,
>
> We have been hacked by malicious programmers. They have encrypted all of our important files. Without these files, our business may never recover! We think it was some kind of ransomware, and we have heard that you are the leading expert in the field. Fortunately, some of our internal cybersecurity experts discovered that the hackers were a bit sloppy in covering their tracks. This is some of the information that our experts collected:
>
> - Ransomware driver source code
> - Ransomware executable
>   - Appears to be homegrown encryption algorithm (crackable?)
> - Keyseed file (WOW!)
> - Encrypted txt file (TIC property)
>
> We tried running the encryption executable again on the encrypted files to see if it is its own inverse, but that did nothing but encrypt the files further. Will you reverse engineer the ransomware executable for us to develop a decryption algorithm?
>
> You are our only hope,
>
> CEO

Work has been slow for you recently, so you could really use the commission. Your task, as outlined above, is to use the recovered artifacts to reverse engineer the encryption algorithm and create a program that TIC corp can use to decrypt the rest of their files. Good luck!

Step 1: Get the Artifacts

Your instructor will deliver to you a .tar.gz file containing everything you need to complete this assignment. Once you have received this file, extract the contents with the command “tar -xvf <projname>.tar.gz”. You should now have a folder containing the four artifacts outlined in the memo from TIC Corp:

- malware.c - source code for the malware driver
- malware - executable compiled from malware.c and included files
- keyseed - unique key used to encrypt TIC system files (different for each student)
- supersecretfile.txt - encrypted file corresponding to your unique keyseed

Step 2: Reverse Engineer the Executable

Your job for this project is to reverse engineer the malware executable. You can use any tool to do this, but it is recommended that you use GDB in your UK VM. The executable takes 1 argument, the name of a file in the current working directory. This file will be encrypted in place.

To reverse engineer the executable, you will need to be able to step through the assembly instructions of the encrypt() function referenced in malware.c. To do this, you should set breakpoints in smart locations in GDB. You will also need to learn how to inspect registers and the stack in GDB.

Submission

To receive credit for this assignment, you must submit a working decryption program that takes a filename as an argument. Your program should completely decrypt the file in place using your assigned keyseed. Submit your original keyseed file along with your decryption program. You can write this in Python, C, or C++.