Reverse Write Up

170D WOBC: Module K Exam II-A

CW2 Kyle Spicer

Due Date: 21 October 2022

1 Project Summary

Scenario: We've been given various .pcap files that contain sensitive information regarding alien activity. The .pcap files contain either a message or movement coordinates, among various other pieces of information.

We are tasked to write a command line utility (reverse) that will decrypt each message in the .pcap, reverse the direction commands, re-encrypt, and write the result back out to out.pcap. If the payload contains a message we need to print the message to STDOUT without the padding. We will then prepend the message with "Don't" and reinsert into the out.pcap conversation.

2 Challenges

2.1 .pcap files

This is my first time encountering .pcap files and it took me several days of research to understand what a .pcap is and how to interpret the data. I felt unprepared to begin this assignment, because several of the first days were spent on researching new topics that have not been covered in the course up to this point.

2.2 binary files and manipulation

While we covered binary data in the course, there weren't many exercises or a heavy emphasis on the topic. Although two weeks seems like adequate time to complete a project, when foreign topics are introduced along with bit manipulation and networking-type activities, it becomes overwhelming.

2.3 robust unit testing

Unit testing is essential to verify programs/functions act accordingly upon valid/invalid input. Creating meaningful unit tests proved difficult for me. I had to create mock structures and figure out how to create a legitimate test with the proper result. Perhaps

in the future, I could write unit tests while I am creating the functions to save time and verify functionality.

2.4 research, research, research...

This project had so many new topics that only a good understanding of C programming wouldn't be enough to complete. This added to stress and derailed my timeline a few times. Being put on a timeline with foreign topics added stress, but managing time and knowing how to properly research and understand these foreign topics helped to complete this project on time.

3 Successes

3.1 creating and packing structures

When we first received this project, while thinking about my design plan, I envisioned using structures to capture all of the data within the packet. Although the test document wasn't very clear about how many bytes each section of information was, I was able to create the structs (some with bit fields) and pack them appropriately. Incorporating the pragma pack(1) was extremely useful as well.

3.2 documentation and organization

I felt that my organization between files and program flow was better on this project. While this project was large and lengthy, I felt my time management and ability to keep concepts and code organized helped me achieve success.

3.3 design and time management

By taking the time to design and take a systematic approach to completing this assignment, helped me stay on track to completion.

4 lessons learned

4.1 frustration

Frustration took a toll on me for this project. Not only was the scope lengthy, but being introduced to multiple new topics on a timeline only raised questions and added stress. Maybe it was by design, however, I learned that some things are out of my control and that not every project will be created with ease. This project was a marathon rather than a sprint.

4.2 triple check

Every section of the program I checked for accuracy, valgrind errors, and verified incoming/outgoing data via byte streams. Prior to this assignment, I had never worked with byte streams, but I was able to overcome the irritation and get into the weeds to make my code work how I needed.

4.3 relax, it'll be alright

I became overwhelmed rather quickly at the thought of not being able to complete this project in time. Everyday, it seemed, I was struggling with a new concept and fighting my emotions to be able to adequately solve the problem at hand. If I could tell myself one thing prior to this practical, it would be relax, you're going to be alright.