CS 3843 Computer Organization I

Project Report

Dec 1, 2021 11:59 p.m.

Team\_10

Group Members

Jude Nelson

Austin Thrash

Nick Sevilla

By signing this report I affirm that I know and agree with the contents.

Signatures: (Just type the Names)

Name #1: Jude Nelson

Name #2: Austin Thrash

Name #3: \_\_\_\_\_\_\_\_\_\_\_

**Objectives:**

For each objective, briefly describe at least 3 specific things you learned by doing the project. This is a group answer so you do not need to list everything for everyone in the group.

1. **Learn to read/write x86 assembly code**
   1. Better understood the flow of an assembly program and the important role of jumps
   2. Better understanding of how certain instructions affect certain registers and flags.
   3. How registers can interact with local and global variables.
2. **Learn about cryptography**
   1. Learned about encryption methods and how to reverse them for decryption
   2. Better understood how encryption is utilized in the real world
   3. The amount of encryption and decryption that can be used on a file.
3. **Learn to follow directions and implement a specific algorithm**
   1. How to convert pseudo code to assembly language.
   2. The many different ways to implement the same algorithm in assembly.
   3. How some complex functions can be implemented in less lines in assembly language.
4. **Learn to work in a small group**
   1. Learning to communicate with everyone and help others.
   2. Learned to distribute certain tasks to verify and make sure the project goes smoothly
   3. If a member doesn’t contribute, being able to work past it and complete the task on time.

**Algorithm Milestone #1**

Everybody’s algorithm for milestone #1 should be pretty similar as the only thing we are doing is looping through the data and xoring every byte with the gkey. However, everyone's output encryption file is different due to the crypto order being used in the sha-256 hash of the password. The decryption is the exact same algorithm as there is not any other manipulation done to the data on the first milestone.

**Algorithm Milestone #2**

The algorithm is milestone #2 is similar to the first, however in this one, after xoring the byte, we then implement the crypto order to the byte. For our team, we start with a code table swap. Both the encode table and decode table correlate so you get the right value back when swapping. We then rotate 3 bits left, which is accomplished with the ROL instruction. Then we reverse the bit order by rotating it out to the right and rotating in to the left on another register. Our fourth instruction is a simple invert of bits using the xor instruction with decimal 98. Finally we nibble rotate left 1, this is done by isolating the nibble and shifting out a bit. That bit is then examined and determines what is added to the nibble, then the nibbles are combined. Once the crypto order is complete the byte is moved into data and the process loops for the length of data.

**Algorithm Milestone #3:**

The final algorithm implements the random hop portion of the encryption. The hop count depends on the password as it is used when creating the password into a randomized array using the SHA-256. The hop count is added to the index and provides another level of security when getting the gKey[index] to xor data.

**Technical Issues:**

The first issue we encountered was implementing rounds on the first milestone. In the first milestone, rounds equals zero, so attempting to implement it threw off the index of gPasswordHash. Another issue was encountered when attempting to nibble rotate left. There was trouble when trying to isolate and test the nibbles but this was fixed by isolating the nibbles with the AND instruction and using the SHL or SHR and testing that shifted bit. Our final issue occurred on the decryption of milestone #3. Instead of decrypting the file in the reverse round order, we were still doing it in the normal order (i.e. we needed rounds to be 3,2,1,0 instead of 0,1,2,3).

**Group Member Contributions:**

Austin Thrash: Majority of code writing, commenting out code, report writeup.

Jude Nelson: Testing, validation, report writeup, turn-ins, and little code writing

Nick Sevilla: None

**Group Issues and Resolutions:**

One of our group members, Nick Sevilla, did not contribute.

**Conclusion:**

The project was fun overall! It’s great when a project demonstrates how something works in the real-world even if it isn’t something you’ll likely code directly in the future. The knowledge of how encryption functions can be applied to a wide variety of other projects even if not directly related.