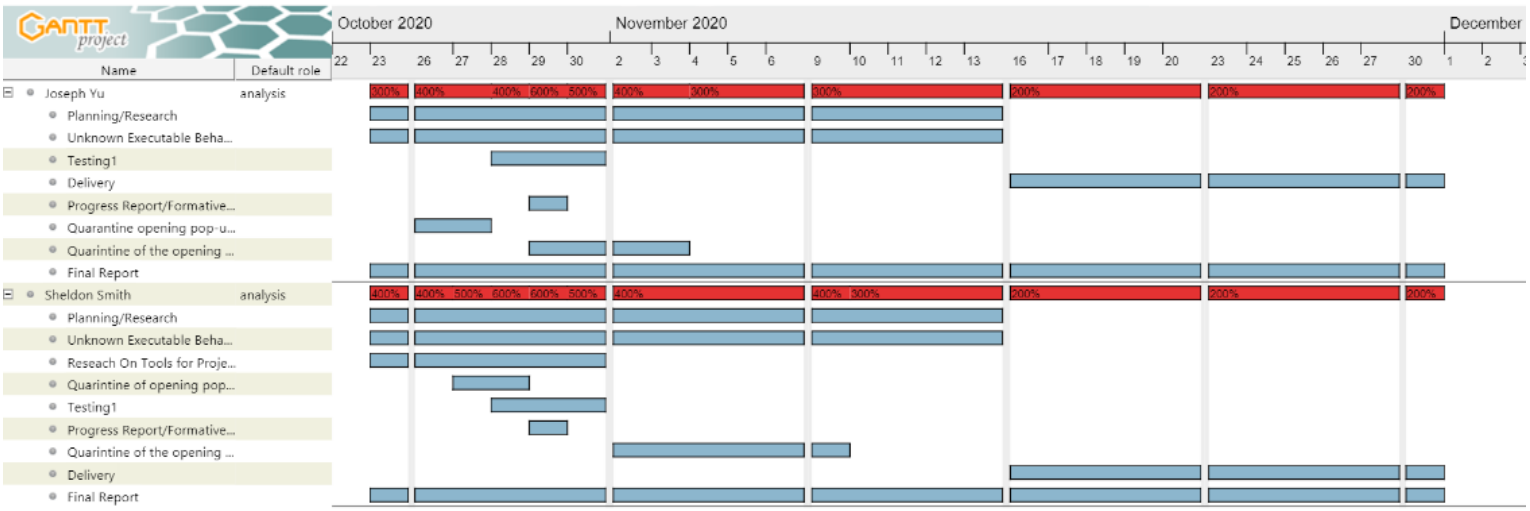
Progress Report

11-5-2020

Team Name: Windows cannot find "Paint"
Members: Sheldon Smith, Joseph Yu

Roles / Responsibilities



Statement of Work

The purpose of this project is to identify what the provided unknown executable ECS.exe does when opened. To aid in our investigation, we will find and use online reverse engineering tools, including disassemblers and decompilers. After using them, we will compile a list with the tools used, in which we describe where to find them, their ease of use, advantages, disadvantages, and lessons learned using the program. Our initial investigation has led us to believe that this executable is a mix of a simple calculator function intermixed with malicious code. We will make the add, subtract, multiply, divide, power, natural log, compound interest, factorial, combination, permutation, and the guessing game work correctly. To do this, we will isolate the dangerous code in the file. We will modify the guessing game to allow the user to automatically win when a secret key is entered. To complete this project, we will meet both online as well as in person. To safety analyze the executable, we will use a Windows 10 virtual machine so that we do not risk infecting our own systems. We will start the project on October 21, 2020 and will finish by December 1, 2020.

We submitted a Progress Report to our client on October 29, 2020, documenting our initial findings and project plan. We will submit our final executable, final report, and present our findings, on December 1, 2020.

Reverse Engineering Tools

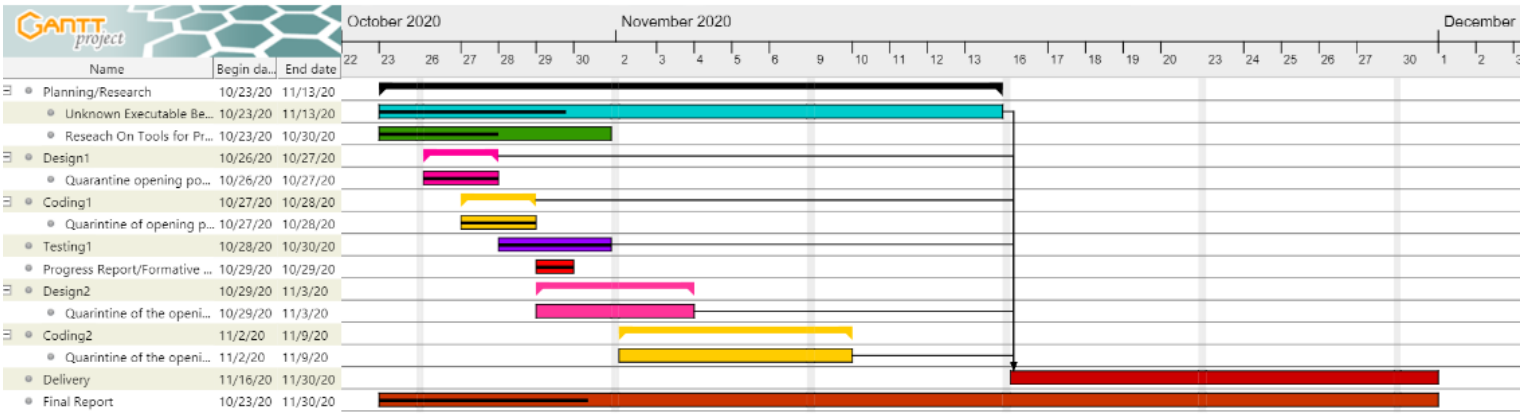
Tool Name	Type of Tool	Where to Find Tool	Ease of Use	Advantages	Disadvantages	Lessons Learned
Ghidra	Disassembler/decomplier	https://ghidra-sre.org/	This program is pretty easy to use. After playing around with the functions and seeing how it works, there is nothing difficult about it.	Ghidra creates a sample of what the code might look like in a high-level language. This makes it easier for the user to understand that overall function of the program without having to understand the assembly language or machine code.	This program splits up the code into multiple sections, such at .data, .text., etc. While this can be helpful, we've noticed that it can be difficult to find the correct section to find a specific piece of code.	
Relyze	Disassembler/decomplier	https://www.relyze.com/	This program is relativly easy to use. Jumping from the flowchart to the assembly code can be a bit confusing at first, but can quickly be understood	This program creates a flow chart of the code which allows the user to easily see how each function interacts with each other.	Relyze tries to create high-level pseudocode from the given file, but the results are very confusing and hard to read. The program is not great at extrapolating the high-level code just from the given file.	
HexEd.it	Hex editor	https://hexed.it/	This program is easy to use. It's simple layout does not confuse the user of how the editor operates.	This program highlight the specific modification that the user made to the original program. This allows for easy recognition of the step to get back to the original design.	The editor has a search function that you can input a series of hex codes and the program finds where that string is found, but from our testing, this function does not work consistantly and can be frustrating.	

Project Plan

Tasks

Name	Begin date	End date
Planning/Research	10/23/20	11/13/20
Unknown Executable Behavior Research	10/23/20	11/13/20
Reseach On Tools for Project	10/23/20	10/30/20
Design1	10/26/20	10/27/20
Quarantine opening pop-up YT videos	10/26/20	10/27/20
Coding1	10/27/20	10/28/20
Quarintine of opening pop-up YT videos	10/27/20	10/28/20
Testing1	10/28/20	10/30/20
Progress Report/Formative Assessments	10/29/20	10/29/20
Design2	10/29/20	11/3/20
Quarintine of the opening of applications and html page	10/29/20	11/3/20
Coding2	11/2/20	11/9/20
Quarintine of the opening of applications and html page	11/2/20	11/9/20
Delivery	11/16/20	11/30/20
Final Report	10/23/20	11/30/20

Gantt Chart



Preliminary Description of Executable's Behavior

This unknown executable appears to be a calculator with multiple options to select from that also has malicious code intermixed with the main code. There are multiple stages of the malicious code. Upon execution in a Windows 10 VM, the program first starts a repetitive cycle of opening a YouTube video 19 times on both Microsoft Edge and Google Chrome. The program then shows the menu options, prompting the user for input (example: "1.", "2." Etc.) The calculator section contains 10 mathematical functions options and one guessing game mode for the user to select. Upon entering the valid instructions the program will then print out "wrong answer" 500 times. The program then reprints the menu options, and then automatically inputs "-1" while printing "choice = -1". The program again displays "wrong answer" 500 times. The process repeats until in the fourth iteration, the program prints "don't do it again" 1,000 times to the screen after reprinting the menu. The program automatically inputs "-1" once more while printing "choice = -1", and again outputs "wrong answer" 500 times. On the next iteration, after printing the menu, the words "I warned you" are output 1000 times. The executable then starts to open the programs Outlook, Paint, Photos, Notepad, along with html page of corndogs floating on Microsoft Edge and Google Chrome on the user's computer that loops for 1000 times.

Github Repository

CSI-2334-Group-Project

https://github.com/legomansps/CSI-2334-Group-Project