Timothy Rolich

Obfuscator write up

Analysis:

A close up of a logo

Description automatically generatedA screenshot of a cell phone

Description automatically generatedThrough multiple attempts to change the outputfile.txt file I have not been successful. I have tried altering all different values in ways that I thought would be undetectable but was foiled at every attempt. The first set of pictures seen here are from an empty run, showing the base values stored after a fresh play. As you can see in the outputfile.txt image Somethings are fairly easy to identify, such as the words Quincy, dagger and helmet, as well as the numbers 100, and 1.

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedAs you can see in the next set of images, I attempted to change the name from Tim to Tnj. By moving one letter up by 1 and one letter down by 1 I had hoped to trick the redundancy check which simply adds all the characters together and mods that value by 29 but it did not seem to work since the program used the default values

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedAnother attempt I made can be seen here at the right. The image below shows two files side by side. The file on the left is the original written by the program and the file on the right is my attempt at manipulating a couple of the numbers by shifting the digit to a different location. On the right you can see the program run with the altered text file and you’ll notice that this was another failure.

A screenshot of a cell phone

Description automatically generatedA screen shot of a social media post

Description automatically generatedA screenshot of a cell phone

Description automatically generatedAs you can see on the left this was a more in depth run that allowed the player to defeat a few enemies and die a few times. The text document below is the output of this run. The capture below that is a replay showing the functionality of reading the text document correctly.

A screenshot of a social media post

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedThis page contains multiple outputs from the IdaPro application. As you can see in the strings window it contains all of the default strings as well as all possible weapons and armor in the game. You can also see that it contains some of the output that is displayed during gameplay such as “You attack!” but it does not contain any information on the encoding process or any of the encoded strings. Beyond this there are the import and functions windows which display similar information. The information in these windows paints a picture of some of the things that the program is doing, such as using random numbers, finding the length of a string, allocating new memory and more.

This final page contains the entropy information. As you can see by the diagram there are no places within the code that would call attention to a cyber-security person.

A screenshot of a computer

Description automatically generated