Caleb Panoch

COT 6427-002: Secret Sharing Protocols

December 3rd, 2024

Project Dependencies

My project was made with Python 3.12.2 and the libraries Tkinter and PyCrypto. Tkinter was used to make the graphical user interface and PyCrypto was used for generating the prime modulus.

Project Description

This project was created to have two programs each read a separate list of names. Then each program/player will compare the names on the list with the other names on the other list. They will do so without compromising the actual names they are comparing by using secure Multi-Party Computation.

Project Design

The program would start by encoding all the input names, whether it be from the textbox or by file. Each name would be encoded with a-z being 1-26 respectively. The space character is encoded as 27. The max character input size is 20, so the largest encoded number is 2727272727272727272727272727272727272727. The communication between the two players is started and the first communication is each other's name list size. Then each player creates two loops so that each name is cross-examined. Within these loops, player 2 chooses the prime by obtaining the bit length of the largest encoded number times two, and then adding one bit. Then player 2 uses this value and inputs it into the PyCrypto prime generator function. Player 2 then sends the prime and its own function's player 1 share to player 1. Player 1 then sends its own function's player 2 share and its own MPC share to player 2. Player 2 then sends its own MPC share to player 1. Each player then uses Lagrange reconstruction to reconstruct the addition of the two encoded names. If the result is twice the original encoded name, then the two names are equal. If they are equal, then the name is added to a list in the program. At the end of the cross-examination, this list will be output in the terminal.