David Morfe
Professor Qiu
CS351 - Intro. to Cybersecurity
13 December 2020

Final Report December 13th

My project's back end is written in Python's Flask framework and front end written in JavaScript, HTML & CSS. I also utilize the Cryptico JavaScript library. I have already created my database with 2 tables; Users & Messages. This end to end secure messaging system will be a web app with the back end server handling all the client interactions and message communication.

The first module, the user resgitration and authentication compenent, is composed on the index web page. With a generic username password login form and registration button. Users will login with their credentials they registered with and the data will be sent as a post request to the back end server. Password will be hashed and stored that way in the database with randomized salt.

The second module, server authentication and user account managment incorporate two different database tables; Users & Messages. Users will have basic user information and salt for preventing password breach. To be sure users are talking to the right server each client will be put in a session once logged in. In order to access the server the user must login and be redirected to each of the webpages. Session data and cookies will contain stored user data and a pass phrase for generating the RSA key.

As for the third module, for secure user communication, it will be handled with secure channel sessions and the *friends* will be linked in the Messages database table. Once a chat is initiated, the chat channel's randomized ID will be stored in each client's session to be sure only the clients on both ends can use said channel. Both clients need to have the browser window open on the chat channel page in order to view each other's messages live. This is because with RSA encryption they're able to communicate in a secure manner by sharing their public keys and decrypting with associated private keys. However this prevents users from being able to view old messages in the chat log since they don't have a shared key to decrypt their friend's message. The Messages database table will keep track of all the channels, and the users that are senders and recipients within a particular channel. RSA encryption key pairs are generated on the fly upon each login with a random pass phrase associated with the username.