Weekly Project Update Report

(Manigandan Ramadasan)

As I approach the end of the internship, I started wrapping up the project by developing client and server code using Flask. The server code consists of all the models and prediction code. The client-side code is responsible for sniffing the packets using tcpdump which outputs a PCAP file. This PCAP file is automatically fed into CICFlowMeter. CICFlowMeter extracts the statistical features and outputs a CSV file which is imported further for data cleaning and preprocessing. The selected 30 features and kept and the other features are dropped. The context for the homomorphic encryption is generation and it is made public by dropping the secret key. The context and the encrypted data are serialized, and it is encoded into base64 format which is sent to the server for prediction. The server decodes the data and performs computations on the encrypted data and the results are sent back to client and the results are decrypted and sigmoid operation is applied at the client side to obtain final results.

The comparison operation on homomorphically encrypted data started working but with less efficiency as it takes a long time to evaluate one comparison operation. I also developed decision tree model. I extracted all the rules of the decision tree forming if-else statements. The working of decision tree will be different from other models: The rules along with the data will be encrypted and will be sent to the server where the comparison operation is p0erformed in batches. The comparison result is sent back to the client, and it is fed into the if-else statements extracted from the decision trees and the final results will be obtained. This is done because the comparison results must be decrypted at the client side again. If we keep the model evaluation at server side, the comparison operation will be performed inn server and the results will be sent to the client which will be decrypted, and he results has to be sent to server again to evaluate the decision tree. This can also be done but we have to send data to the server twice which can be avoided if we get comparison results, and the if-else rules are evaluated at client side instead of server side.

The report is about to be completed as I must add some graphs and testing results of decision tree and other algorithms.