Homework 3

Submission format: submit one .pdf file as the report for all tasks (like the solution I provided) and submit it separately on Canvas (**2 points bonus**). For the code files, you can upload them separately or pack them in .zip file to upload.

1. Homomorphic Encryption (50 points)

Alice holds a private matrix A (nonnegative integer entries) with size 5×8 while Bob holds a private matrix B (nonnegative integer entries) with size 8×4. Design and implement a two-party protocol to securely compute the product $A \times B$. Hint: Homomorphic Encryption (e.g., Paillier Cryptosystem which is asymmetric) can be used to design the protocol.

- Paillier in Python:

  https://python-paillier.readthedocs.io/en/develop/

  https://github.com/mikeivanov/paillier

- Paillier in Java:

  https://www.csee.umbc.edu/ kunliu1/research/Paillier.html

Tasks:

(a) Alice generates random nonnegative integer entries for A while Bob generates random nonnegative integer entries for B. (**5 points**)

(b) Design the cryptographic protocol between Alice and Bob to perform secure computation. (**20 points**)

(c) Write the programs for Alice and Bob: computation and communication. Note that communication should be established to exchange encrypted messages, e.g., using Socket programming. (**10 points**)

- Socket Programming in Python: https://realpython.com/python-sockets/
- Socket Programming in Java: https://www.tutorialspoint.com/java/java_networking.htm

(d) Report the input matrices, the last ciphertext (right before the decryption) and the decrypted product $A \times B$ using two different key sizes 512-bit and 1024-bit. (**5 points**)

(e) Discuss the following two cases (ideally theoretically justify the conditions for two cases) and verify them using the source code: (1) multiplicative homomorphic property of Paillier holds, and (2) multiplicative homomorphic property of Paillier does not hold. Submit the screenshot of results and discuss your conclusion. (**10 points**)

2. Secure Multiparty Computation (50 points)

Alice holds a private Boolean vector $\vec{A}$ with 10 Boolean entries ($\{0,1\}^{10}$) while Bob holds another private Boolean vector $\vec{B}$ with another 10 Boolean entries ($\{0,1\}^{10}$). Design and implement a protocol using the *Fairplay* to securely compute the scalar product $\vec{A} \cdot \vec{B}$ without sharing their inputs to each other.

- The scalar product computation should be converted to garbled circuits using SFDL.
- *Fairplay* secure function evaluation: https://www.cs.huji.ac.il/project/Fairplay/.
- Readme file for running *Fairplay* SFE:

  https://www.cs.huji.ac.il/project/Fairplay/Fairplay/Readme.txt

Tasks:

(a) Alice generates random Boolean entries for $\vec{A}$ while Bob generates random Boolean entries for $\vec{B}$. (**5 points**)

(b) Write the SFDL program for Alice and Bob. (**20 points**)

(c) Compile it for Alice and Bob, and run the protocol (communication is integrated in *Fairplay*). (**15 points**)

(d) Report the input Boolean vectors, the SFDL program, SHDL circuit, and output results $\vec{A} \cdot \vec{B}$ (for two parties). (**10 points**)