CS523 Project 1 Report

Pavliv Valentyna, Lion Clement

***Abstract*—During this project, we had to design a protocol that could compute secure-multiparty communications[1]. Our programming language is Lattigo, for it offers the opportunity to create goroutines. These are very appropriate to simulate multiple parties trying to communicate.**
During our project, we used the library Lattigo[2]. It offers homomorphic encryption which is particularly useful for the exchange of Beaver triplets, an invaluable element to do secure communications.

Please report your design, implementation details, findings of the first project in this report.
You can add references if necessary [1][2].
THE REPORT SHOULD NOT EXCEED 3 PAGES.

I. Introduction

With the surge of internet

Give a brief introduction about the aim of the project, and your road-map about the design/implementation.

II. Part I

A. Threat model

Give the corresponding threat model for the first part of the project that you implemented.

B. Implementation details

- Give your implementation details
- Detail the circuit you created at the end of the first part

III. Part II

A. Threat model

Give the corresponding threat model for the second part of the project that you implemented.

B. Implementation details

Give implementation details.

IV. Evaluation

- Give a comprehensive comparison and evaluation about Part1 and Part2 of the project including performance results. Feel free to use charts, tables, plots...

- What affects the efficiency of the executions? Be specific, which types of operations/circuits are directly linked to performance?
- Is there any difference in terms of performance between Part I and Part II? Why?

V. Discussion

- Comment on your findings, discuss different outcomes for each part.
- Discuss outcomes from different circuits including your own circuit.
- In your opinion, which model is appropriate to use under which conditions/threat model? Why? Discuss.
- Come up with a scenario for each part of the implementation, discuss why it makes sense to use homomorphic encryption based generation of Beaver triplets.

VI. Conclusion

- Assess your learning outcomes for this project.
- What did you do? What did you learn? Any interesting design ideas?

References

[1] O. Goldreich, S. Micali, and A. Wigderson, “A completeness theorem for protocols with honest majority,” *Conference Proceedings of the Annual ACM Symposium on Theory of Computing*, 01 1987.

[2] “Lattigo 1.3.1,” feb 2020, ePFL-LDS. [Online]. Available: http://github.com/ldsec/lattigo