Secure Multiparty Computation Project

Relja Eskic, Galen Metzger, Van Lund

Project Goals:

- Us as a group to learn about an interesting and useful Computer Security topic
- Group have a sufficient understanding of the topic to explain it to people not familiar with it
- Design a lecture introducing SMPC, its use cases, and walking through an example protocol
- Design a demo implementing a demo protocol in code

Project Design:

- Slideshow presentation explains the problem that SMPC solves
- Lecture over the slides introduces the topic, explains why it is important, shows an example problem where SMPC can be a solution, and also steps through the simple math/logic of an entry-level SMPC protocol
- Demo simulates two parties exchanging information between each other and doing the math to derive the result of the multi-party computation

Project Implementation:

- Biweekly internal meetings discussing research we had completed about the topic
- Demo implementation written in C++
    - C++ provides faster speed when using cryptographic functions when compared to other languages such as Python
    - C++ has available libraries to efficiently do math with arbitrarily large numbers
    - We already had available C++ code implementation of RSA written for a previous class
- Demo simulation

    - Two parties pass information back and forth via variables in the scope of the main function
- Slides/Lecture/Protocol Walkthrough developed based on both research conducted as well as lessons learned through implementing the code demo

Project Results:

- Group members gained an understanding and appreciation for SMPC, what problems it can be used to solve, and some insights as to why it hasn't seen widespread adoption yet
- C++ demo is functional, produces the correct outputs, and displays information being passed between the parties
- Lecture introduces the topic, the abstract problem that it solves, lists multiple possible use cases, and walks through the toy demo ("Yao's Millionare Problem") both logically and mathematically
    - We believe that the lecture was broadly successful in introducing our peers to the topic and shedding some light on how the concept works and/or "why it is possible". After our presentation during class participation time, we did receive a few interesting follow-up questions on the topic which showed us that the lecture was able to get people understanding and thinking about the field.

Project Evaluation:

- We believe the project was all-around successful given the time constraints and our unfamiliarity with the topic
- Things we would iterate on:
    - Understanding and explaining the process by which you can take *any* mathematical function with a single output and turn it into an SMPC function

- In-depth mathematical explanation including the reasoning behind using modulus on the data being received/modified
- Networked demo connecting two independent clients to an untrusted middle-man server that forwards data as well as displays all data it receives to show that you don't have to trust the middle-man server to achieve privacy

Resources used throughout the project:

- MPC: A Complete Guide https://www.partisia.com/tech/multi-party-computation
- What is Secure Multiparty Computation? https://www.geeksforgeeks.org/blogs/what-is-secure-multiparty-computation/
- Protocols for Secure Computation (extended abstract), Andrew Yao 1982 https://research.cs.wisc.edu/areas/sec/yao1982-ocr.pdf
- ChatGPT used in early stages of research to understand high-level overview of the field