Secure Multi-Party Computation as a Tool for Sensitive Data Analysis

Samuel Havron
havron@virginia.edu
University of Virginia

Abstract— **Poster: A *Secure multi-party computation* (MPC) is a protocol which allows for two or more parties to compute a function on sensitive input data provided by each party, without revealing anything about the inputs (other than what can be inferred from the revealed output result). Most current implementations of MPC work by executing instructions in a *data-oblivious* manner, where the control flow of the program is independent of the inputs provided by each party and the program executes without any knowledge of the cleartext data it is operating on. Many researchers and social scientists analyze datasets that are sensitive, and they often need to rely on closed-source software to execute MPC protocols.**

OblivC is a new programming language which is designed to make it simple for anyone to write code which provides secure protocols for analyzing data and suiting individual needs for a particular project. Obliv-C implements secure MPC through optimizations of Yao's garbled circuit protocol for use with semi-honest adversaries. This effectively creates a black box, such that semi-honest adversaries cannot gain addition insight about the source data, receiving only the computated results. The languages library includes numerous Application Programming Interface (API) protocols, which allow an application programmer to quickly implement MPC programs.

Without needing to be an expert in cryptography, a researcher can write OblivC code using the language's API for secure protocols and TCP/IP connections, as well as link their OblivC code to standard C code (where C code could be a means of writing all code that does not need to be under a secure protocol). In the ongoing development of applications that exemplify the power of OblivC, a linear regression analysis program was developed with considerations to scalability, dataset matchup, and qualitative demonstration of using OblivC API and linking OblivC code and C code together.

The scalability and speed of OblivC as a tool for implementing MPC programs was tested using c4.large *Elastic Compute Cloud* (EC2) nodes from *Amazon Web Services* (AWS) of Amazon.com®. Two c4.large instances were launched and connected through OblivC's API for TCP/IP connections. One node instance provided independent (x) data points, while the other provided dependent (y) data points; data points used were 32-bit integers, using fixed-point mathematics to convert raw data values into scaled integers (OblivC does not currently support floating point numbers). The time needed to execute the MPC between instances appears to scale linearly with the size of the data input, with 100K data points finishing execution in 12.7 minutes on average, 500K completing in 64 minutes, and 1 million data points finishing execution in just over 120 minutes on average. Synthetic data was generated for testing the scalability of input size.

To provide a clear example of the utility of OblivC for analyzing sensitive datasets, additional data for computation was obtained from the public New York State Department of Health dataset of Hospital Inpatient Discharges from 2011. Comparisons between fields such as "Length of Days stayed" and "Total Costs" over approximately 2.6 million data points are currently being tested. This dataset is particularly amenable to analysis, as all data is already matched properly and each data value is a comparable number. However, using OblivC's API for private set intersection and ORAM access with a unique identifier (such as SSN) is being investigated so as to provide an application which can match all datapoints from two separate datasets which are tied to a common identifier.