SRD-Final Project “Privacy-Preserving Graph Algorithms in the Semi-honest Model”

By Aviad Gilboa,Yaakov Khodorkovski , Dolev Dublon ,and Daniel Zaken

1. Introduction:

The paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" by Justin Brickell and Vitaly Shmatikov presents new algorithms for computing all pairs shortest distance (APSD) and single source shortest distance (SSSD) in a privacy-preserving manner. These algorithms have potential applications in scenarios such as Internet providers considering a merger, transportation companies determining shipping capacity between cities, and social networking websites calculating aggregate statistics without compromising user privacy.

The purpose of this SRD (System Requirements Document) is to specify the requirements and design for an implementation of the privacy-preserving graph algorithms presented in the paper. The scope of this SRD includes the functional and non-functional requirements for the implementation, as well as the high-level design and planned implementation approach.

This SRD assumes a basic understanding of the secure multi-party computation paradigm and the semi-honest model, which are discussed in the paper.

2. Background:

The paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" presents new algorithms for computing all pairs shortest distance (APSD) and single source shortest distance (SSSD) in a privacy-preserving manner. These algorithms have potential applications in scenarios where two parties have a graph each and want to compute some algorithm on the combined graph, but do not want to reveal any information about their individual graphs beyond what is revealed by the output of the algorithm.

Examples of such scenarios include Internet providers considering a merger, transportation companies determining shipping capacity between cities, and social networking websites calculating aggregate statistics without compromising user privacy. In these cases, the parties may want to compute some graph algorithm on the combined graph, but do not want to reveal any sensitive information about their individual graphs.

The privacy-preserving graph algorithms presented in the paper are designed to work in a standard secure multi-party computation paradigm, where two or more parties wish to compute some function on their private inputs without revealing those inputs to each other. The algorithms are proven to be secure in the semi-honest model, where the parties are assumed to be "honest but curious" and correctly follow the protocol, but there is no way to verify the authenticity of their inputs.

3. Requirements:

The implementation of the privacy-preserving graph algorithms presented in the paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" should meet the following requirements:

Functional Requirements:

- The system should be able to compute all pairs shortest distance (APSD) on two input graphs in a privacy-preserving manner.
- The system should be able to compute single source shortest distance (SSSD) on two input graphs in a privacy-preserving manner.
- The system should be able to compute privacy-preserving set union on two input graphs.

Non-Functional Requirements:

- The system should be efficient, able to handle large graphs in a reasonable amount of time.
- The privacy of the input graphs should not be compromised by the algorithms.
- The system should be secure, following the secure multi-party computation paradigm and the semi-honest model.

Input/Output Formats:

- The input to the algorithms should be two graphs in a standard format such as adjacency lists.
- The output of the APSD and SSSD algorithms should be a dictionary of dictionaries containing the shortest distances between pairs of nodes or a single source and all other nodes, respectively.
- The output of the privacy-preserving set union algorithm should be a graph representing the union of the two input graphs.

4. Design & Implementation:

The design of the system for implementing the privacy-preserving graph algorithms presented in the paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" should meet the following requirements:

- The system should consist of a set of functions or methods for computing APSD, SSSD, and privacy-preserving set union on two input graphs.
- The APSD and SSSD algorithms should be based on the algorithms presented in the paper, which are more efficient than generic constructions such as Yao's garbled circuits.

- The privacy-preserving set union algorithm should be based on the algorithms presented in the paper, which use canonical orderings on graph edges to achieve efficiency.

Trade-offs or Decisions:

- The choice of programming language and libraries will depend on the desired performance, ease of implementation, and any other relevant considerations.
- The design of the algorithms may involve trade-offs between efficiency and the level of privacy preserved.
- The design should also aim for ease of use and accessibility, as it is intended to be a familiar and used open-source code.

The implementation will involve creating a set of functions or methods for computing APSD, SSSD, and privacy-preserving set union on two input graphs. These algorithms should be based on the algorithms presented in the paper, which are more efficient than generic constructions such as Yao's garbled circuits and use canonical orderings on graph edges to achieve efficiency.

Challenges or Limitations:
Implementing the algorithms in a manner that is both efficient as possible and secure may present challenges.
Ensuring that the implemented algorithms are correct and meet the requirements may also be a challenge.

5. Testing:

The implementation of the privacy-preserving graph algorithms presented in the paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" should be thoroughly tested to ensure that it meets the requirements specified in the "Requirements" section of this SRD.

Testing Strategy:

- The testing strategy should involve a combination of unit tests, integration tests, and system limits tests to ensure that the implemented algorithms are correct and efficient.
- Test that will demonstrate the situation of an adversary who wants to get information about the second user graph.
- The tests should cover a range of sample graphs to ensure that the algorithms work correctly on different inputs.
- The tests should also ensure that the privacy of the input graphs is not compromised by the algorithms.

Types of Tests:

- Unit tests: These tests will verify the correctness of individual functions or methods within the implementation.

- Integration tests: These tests will verify that the different components of the implementation work correctly together.
- System limit tests: These tests will use large graphs to determine the maximum size that the program can handle.
- Other relevant tests: These could include tests to ensure that the algorithms are efficient, or tests to verify the security of the implementation.

6. Deployment:

The implementation of the privacy-preserving graph algorithms presented in the paper "Privacy-Preserving Graph Algorithms in the Semi-honest Model" should be deployed in a manner that meets the requirements and constraints of the target environment.

Deployment Strategy and Security Considerations

- The implementation will be released as an open-source project on GitHub for public use.
- The deployment should ensure that it is deployed in a secure manner and that any sensitive data is handled appropriately.