An Anonymous Reputation System Resistant against Tracking and Intersection Attacks

CPSC 490 Project Proposal

Tony Jiang and Ian Zhou
Advisor: Ennan Zhai
Submitted: February 8, 2018

AnonRep is an anonymous reputation system that enables users to anonymously post new messages to a global message board and give feedback on previously posted messages [1]. Unlike traditional reputation systems, AnonRep prevents attackers from linking identities between messages by providing stronger guarantees on privacy and anonymity. However, AnonRep is prone to long-term intersection attacks. For example, if a system contains one user with a significantly higher reputation score than all other users, an attacker may be able to link posts of high reputation to the same person. We solve this issue by incorporating blockchain into AnonRep; using blockchain, users can control the amount of reputation they attach to posts that they make. In this project, I will implement the core AnonRep program in Python and perform analyses on the efficiency of our blockchain-based solution.

1 Background

AnonRep operates in a series of message-and-feedback rounds. In practice, each round may last anywhere from a few minutes to an entire day, depending on the application scenario. Each round is split into three phases: announcement, message posting, and feedback. At the beginning of each round, each server maintains a table of all of the clients’ long-term pseudonyms and their corresponding encrypted reputation scores. In the announcement phase, the servers collectively create a table of encrypted one-time pseudonyms (from the clients’ long-term pseudonyms) and their respective decrypted
reputation scores. In the message posting phase, clients post messages using these one-time pseudonyms to a global message board maintained by all of the servers. Finally, in the feedback phase, clients anonymously provide feedback to the posted messages. Long-term reputation scores are updated using a reverse-scheduling protocol before the next round begins. More detailed descriptions of the workflow are described below.

1.1 Client Registration

A new client starts by generating both a private and a public key and sending the public key (long-term pseudonym) to a server. The server generates an initial reputation score for the client. Each server then sequentially encrypts the reputation score using its own public key. Once all servers have finished encryption, the long-term pseudonym and respective encrypted reputation score is broadcast to all servers.

1.2 Announcement Phase

At the beginning of the announcement phase, each server maintains a table of the clients’ long-term pseudonyms and their corresponding encrypted reputation scores. The table is passed around sequentially among all of the servers. Each server performs the following series of events.

- Verifiable shuffle of the entries in the table.
- Encryption of list of pseudonyms using an ephemeral secret.
- Decryption of list of reputation scores using its private key.

Once all servers have finished this procedure, the table of one-time pseudonyms and their respective (decrypted) reputation scores are broadcast to all servers and clients.
1.3 Message Posting Phase
In the message posting phase, clients post messages using their one-time pseudonyms and attach signatures to their messages made using their private keys. A server verifies that the message and one-time pseudonym match their corresponding signature. If they do, the message is posted to a global message board.

1.4 Feedback Phase
In the feedback phase, clients provide feedback (either positive or negative) and sign their feedback using a linkable ring signature. A linkable ring signature hides a voter’s pseudonym and prevents duplicate votes from clients. Afterwards, the announcement process is performed in reverse to update the initial reputation table.

1.5 Blockchain and Tokenization
A shortcoming of AnonRep is that it is prone to intersection attacks. A user with a unique reputation score, for example, one that is much higher than all the other users, has little anonymity. This is because other users will be able to link any message posted by a user with that high of a reputation to her.

Our proposed solution to this vulnerability is to tokenize reputation into a “reputation coin” on the blockchain. Users will have their reputation in different “wallets” that they use to post messages. If another user votes up a message, one of the wallets used to post the message will gain a reputation coin. If the message is downvoted, one reputation coin will be removed from one of the wallets, unless there is no more reputation left in the wallets. We will use CoinShuffle to make transactions on the blockchain anonymous.

2 Project Outline
A brief outline of our project is provided below.

1. Read existing papers on Anonymous Reputation and understand various cryptographic primitives (e.g. verifiable shuffle, linkable ring signature, and coin shuffle).

2. Find and test related cryptography libraries for Python (or if needed, implement the cryptographic methods ourselves).

3. Implement the existing version of Anonymous Reputation in Python.

4. Extend our implementation to a blockchain-based program to prevent against tracking and intersection attacks.

5. Run analyses on the efficiency of our blockchain-based program as the number of clients and servers increases. Compare its efficiency to that of the existing version of Anonymous Reputation.
6. If time permits, research other solutions that may improve upon a blockchain-based approach.

3 Deliverables

The main deliverables involve code making up the system we build and a report with our learnings, results, and future directions. I will focus more on the coding and data analysis aspect of the project. More specifically, my deliverables will include:

- A Python implementation of the original AnonRep system.
- A Python implementation of the leader election protocol.
- Analyses of the scalability of our system as the number of clients and servers increases.
- Tests that show correctness of our system.

References