ZheroTag: Problem Statement

Yiğit Kılıçoğlu*, Lucas Huang†

Last updated: September 2022

About: In this document we describe a game, currently called ZheroTag, that uses advanced cryptographical concepts like zero-knowledge proofs, homomorphic encryption and multi-party computation. The game is relatively simple, but the challenge is to 1) come up with a P2P protocol that can be used to drive the game, 2) find libraries that can be used to implement the game. We are looking for help for both. This document is intended to make the problem statement and our current progress clearer to those who are trying to help us.

*yigit.kilicoglu@yale.edu.
†lucas.huang@yale.edu.

Game Description

The game is played on a finite square grid with two players. Each player has a single piece on the grid. The initial coordinates of the pieces are predetermined and are relatively far from each other. The pieces move like how kings move in chess. The players can only see the squares that are the Moore neighbors of the positions of their pieces (i.e. the 8 surrounding squares if the piece is not at a border of the grid). The game is played in alternating turns. At each turn, one player moves their piece to one of the Moore neighbors. The goal is to capture the opponent's piece, which can be done only when the opponent is on a Moore neighbor.

Note 1: The game is somewhat similar to Dark Chess, a chess variant. It can be played on chess.com. A tutorial can be found here.
Note 2: The game described above is the simplest version of this game that still requires advanced cryptography. Once we solve the cryptography part, the game can be made more complex.

Implementation

With a trusted third party

The implementation is very simple if there is a trusted party. The players share each of their moves with the trusted party, which in turn updates the board and tells each player what they can see on the board.

Without a trusted third party

This is what we are trying to do.

Progress and Problem Statement

The challenge with implementing without a trusted third party is to keep the locations of the pieces private while updating the views of the players. Zero-knowledge proofs can be used to prove that a move was valid without actually revealing the move. However, this information is not enough for the next player to know which moves they can make. After each move, each player needs to update their board in such a way that the opponent does not learn anything about the location of their piece. If the last move brought a piece to a square around the other piece, then both players should learn about this fact.

Questions:

- A protocol that we came up with for ZheroTag is the following: Player A makes a move. Now both players need to update their boards. It is apparently possible for private set intersection (PSI) protocols to reveal the intersection to only one party. For the first PSI, the first set is the Moore neighbors of player A's new position. The second set is player B's position. Player A sends a zero-knowledge proof to prove that its move is valid. Then, they do the PSI and player A learns whether player B is next to player A. For the second PSI, the first set is player B's Moore neighbors and the second is player A's new position. So player B also learns whether A is next to them. They will also send each other zero-knowledge proofs to ensure that the elements in the sets are what they are supposed to be. Would a protocol like this work?
- We read that private set intersection can be implemented using homomorphic encryption. Are there any libraries that can be used to implement this game? How about Zama's Concrete?

Thanks for your help!