Ro-Sham-Bo

HW6 - CNS Sapienza

Giulia Muscarà 1743261

5 December, 2019

1 Objectives

The goal was to design a protocol suitable to implement the game of rock, paper, scissors, allowing two players to play a fair match in real time, without the intervention of third parties. To do that, it was fundamental to guarantee data integrity through a robust protocol and cryptographic tools.

2 Rules

A match is a made of N games, with N being an odd number. To win the match each player has to win at least more than the half of games in a match. There are only three possible moves to make: "Paper" ,"Rock" and "Scissors". A player who decides to play rock will beat another player who has chosen scissors, but will lose to one who has played paper and a play of paper will lose to a play of scissors. If both players make the same move, the game is tied and it is replayed. We can assume that users cannot play one in front of the other but remotely and that both respect the game protocol. If there is the possibility to elude a weak protocol, players will, which, indeed, cannot be considered cheating.

3 Design

To start a game, a player first sends an invitation to their desired opponent, attaching a nonce. Upon receiving a request, the other player can either send back a rejection message not to open the connection or an acceptance message. In the last case the connection is established between the players and the game can start. In each round, each player commits to their move,

that can be either Rock, Paper, or Scissors, and only after the other party commits to a move as well, it reveals the commitment to the other player.

4 The protocol

The proposed protocol only uses nonces, and a cryptographic hash function. The following variables will be used in the description of the protocol.

- H = cryptographically secure hashing function
- nonce = nonce generated by Alice at the beginning of a match
- C_A = move of Alice
- C_B = move of Bob
- W_A = number of games won by Alice
- W_B = number of games won by Bob

To start with, let's suppose that there are two players: Alice and Bob. After establishing a connection, Alice selects a move C_A and Bob chooses C_B. Then A sends to Bob $H_A = H(nonce||C_A)$. After receiving H_A, Bob computes $H_B = H(H_A||C_B)$ and sends it to A. After receiving H_B, A sends the nonce back to Bob. At this point, Alice can compute the three hashes $H(H_A||C)$, with C being a move among rock, paper and scissors: if one of these hashes equals to H_B, she came to know the move made by B, otherwise B made an invalid move and the game should stop. In the first case, if she turns out to be the winner of this game she can increment her local register W_A by 1. A is the winner if and only if $C_A > C_B$ and viceversa, assuming that $Paper > Rock, Rock > Scissors$ and $Scissors > Paper$. Bob as well, after receving the nonce, can compute the three hashes $H(nonce||C)$, with C being rock, paper or scissors and either find out the move made by A or stop the game. In the first case, also Bob can increment his own local register W_B only if he is the winner of the game.
At the end of each round, Alice and Bob exchange $(W_A||W_B)$ to verify that the local register were updated correctly and that neither of them cheated. If the message received by the honest party is different from the one he sent, the opponent cheated and the game should stop. Otherwise, the game can proceed with the next game, repeating the procedure. As soon as one party's W reaches N/2 +1 games won, the match can terminate with the winning of that party and the connection can be shut down. One party can check if the opponent won by looking at the number of won games.

5 Security

The nonces used for each round are needed to prevent replay attacks. Indeed, if Alice made a choice $C_A \in \{Paper, Scissors, Rock\}$, thanks to the round nonce, in the next game the same choice would lead to a different hash, preventing Bob from knowing in advance what is Alice's move.
Also, nonces make it impossible to pre-compute the move of the opponent, as one party receives the nonce only after having sent his move.
Finally, the local registers for counting points guarantee that once the moves are sent hashed together with the private nonce, the two players can compute the hash and update the register basing on the move originally sent by the opponent. If one of the two tries to change his choice, the opponent will notice the unfair behavior and quit the game.
Moreover, there is no need for encryption as confidentiality is not a requirement.