DEAL: A trustless cardgame on blockchain

Hari

September 2, 2020

Motivation

- Playing a card game that involves *shuffling* on the blockchain is difficult.
- *Why?* Requires trusting someone to shuffle randomly and without bias.
- *Centralised solution*: A Casino generates a random deck, sends cards to two people.

Mental Poker

- *Project*: implement a simple protocol that allows two people to shuffle a deck in a *trustless* way.
- The protocol is based on a article by Shamir, Rivest and Adleman called "Mental Poker."

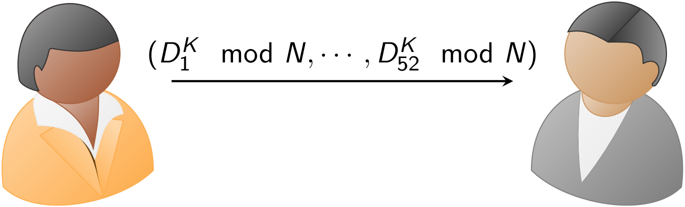
Protocol

- Similar to a state machine.
- *Game*: Whoever picks the largest card wins.
- Each player has three calls in a certain order.
- A call to commit.
- A call to play/shuffle.
- A call to reveal the secret.
- Finally anyone can verify if the ‘game’ was played fairly.

Encoding, Encryption and Decryption

- Fix numbers $D_1, \cdots, D_{52}$ to denote a deck of cards.
- In the contract, decimal digits of the Golden Ratio was chosen.
- A number $N = 2 \cdot 3 \cdot 5 \cdots 193$ is fixed.
- Front-end: Secret numbers K and L.
- Encrypting x by $x^K \mod N$.
- Decrypting y by $y^L \mod N$.

Commit: Alice



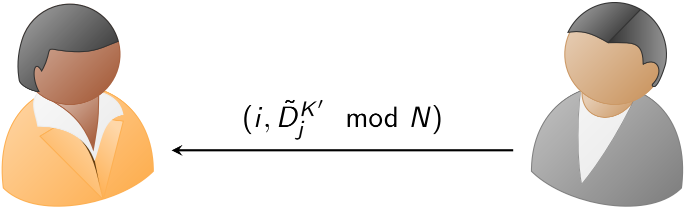
$$(D_1^K \mod N, \cdots, D_{52}^K \mod N)$$

Alice: K, L

Bob: K', L'

- The numbers are sent shuffled.

Commit and Play Bob



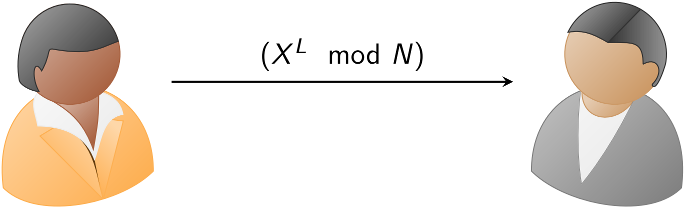
$(i, \tilde{D}_j^{K'} \mod N)$

Alice: K, L

Bob: K', L'

- i corresponds to Alice's index and j corresponds to Bob's index.

Play: Alice



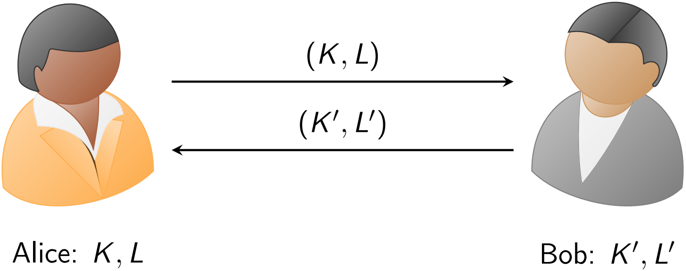
$(X^L \mod N)$

Alice: K, L

Bob: K', L'

- $X = \tilde{D}_j^{K'} \mod N$ is the Bob's card encrypted by Bob.

Reveal: Alice and Bob



(K, L)

(K', L')

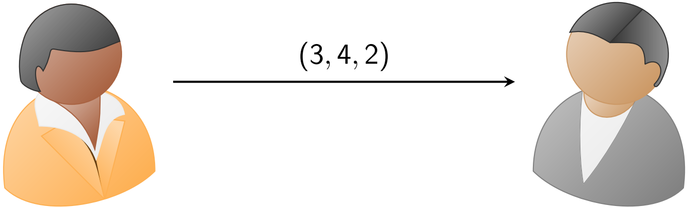
Alice: K, L

Bob: K', L'

Example

- Deck of only three cards: $2, 3, 4$.
- N: 5.
- Alice's secrets: $(K, L) = (1, 1)$.
- Bob's secrets: $(K', L') = (3, 3)$.

Example: Commit Alice



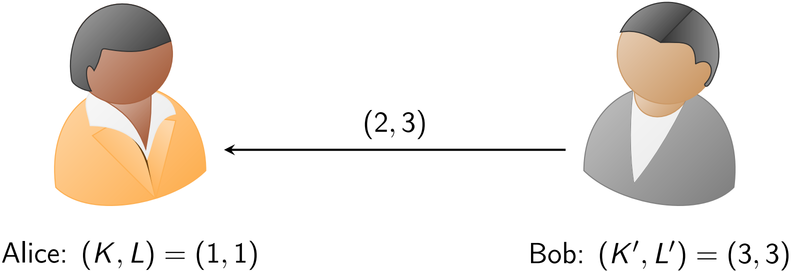
$(3, 4, 2)$

Alice: $(K, L) = (1, 1)$

Bob: $(K', L') = (3, 3)$

- The encrypted deck $(2^1, 3^1, 4^1)$ is sent shuffled.

Example: Commit and Play Bob



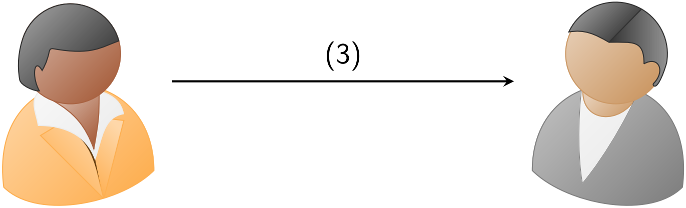
$(2, 3)$

Alice: $(K, L) = (1, 1)$

Bob: $(K', L') = (3, 3)$

- 2 corresponds to Alice's index and 3 corresponds to Bob's encrypted card.
- Bob chooses the last card. Bob's card encrypted: $2^3 = 3 \mod 5$

Example: Play Alice



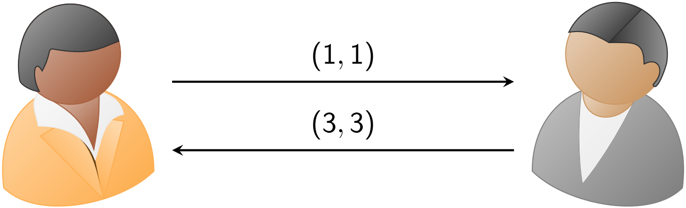
(3)

Alice: $(K, L) = (1, 1)$

Bob: $(K', L') = (3, 3)$

- Bob's card encrypted by Bob is 3. Alice decrypts it by $3^L = 3^1 = 3$. Bob decrypts by $3^{L'} = 3^3 = 2 \mod 5$

Example: Reveal Alice and Bob



$(1,1)$

$(3,3)$

Alice: $(K, L) = (1,1)$

Bob: $(K', L') = (3,3)$

- Alice's card: $4^L = 4$. And Bob's card is 2.

Frameworks

- Used Truffle framework for testing and deploying.
- Front-end in react.js and web3.js.