A Smart Card-Based Mental Poker System

Jordi Castellà-Roca, Josep Domingo-Ferrer and Francesc Sebé

Rovira i Virgili University of Tarragona
Dept. of Computer Engineering and Maths
Av. Països Catalans 26
E-43007 Tarragona, Catalonia
{jordi.castella,josep.domingo,francesc.sebe}@urv.net

Abstract. On-line casinos have experienced a great expansion since the generalized use of Internet started. There exist in the literature several proposals of systems allowing secure remote gaming. Nevertheless, the security requirements of some game families lead to the use of complex and costly cryptographic protocols. A particularly challenging game family is mental poker. In this paper we present a smart card-based e-gaming system for mental poker with a low computational cost.
Keywords: Smart cards and applications in the Internet, Cryptographic protocols for smart cards, E-gambling, Mental poker.

1 Introduction

Computer networks, and especially Internet, allow a lot of usual activities to be carried out in a time- and space-independent way. Leisure is a sector that has quickly grasped and exploited the possibilities of the network as a new business outlet. On-line casinos are a particularly visible form of on-line leisure. Increasing sales figures of on-line gambling companies are a clear indicator of the positive evolution in this sector. According to Merryll Lynch the on-line gambling business is expected to grow to $48 billion by 2010 and $177 billion by 2015. This booming turnover must be accompanied by enough security guarantees for on-line players; unfortunately, this is not always the case.

In an on-line casino, players usually go through the following steps:

Registration: Prior to accessing the on-line casino, players must register themselves. In the registration step, players give their personal information. This information is used by the on-line casino to create an account for the player. Players will access the on-line casino via their account.

Authentication: After registration, players possess the necessary information (typically a username and a password) to authenticate themselves to the casino and log in their accounts.

Increase credit: On-line casinos tend to use pre-payment methods. Thus, players must make a payment to the casino before starting to play. The amount of cash that has been paid by a player receives the name of credit, and it is transferred to the account created in the registration step. When a player makes a bet, the on-line casino verifies that the player has got enough credit. If the player loses/wins her bet, the on-line casino substracts/adds the bet amount from/to the player's credit.

Withdraw credit: The player transfers her game earnings from her casino account to her bank account.

Bet: At least one bet is made in every casino game. The game rules specify how many bets are possible and when players can bet.

Game: The rules of each game drive its operation. Based on those rules, players obtain one or several random events during the game. The game result, *i.e.* who wins and who loses, is based to some extent on the obtained events.

We can assert that a gaming system is secure if each of the above steps can be done in a secure manner.

1.1 Contribution and plan of this paper

We present in this paper a gambling system that allows poker to be played remotely, while offering security for the different steps players need to go through.

In the proposed system, each player owns a smart card that runs the security-critical parts of the aforementioned steps. We assume that smart cards are issued by the public authority that regulates on-line gambling. This authority ensures that: i) player registration is made properly; ii) the software inside the smart card is fair.

This paper is organized as follows. A state of the art is given in Section 2. Section 3 justifies the security requirements that will be considered. The architecture of the proposed systems is described in Section 4. The relevant protocols of our system are specified in Section 5. Section 6 is a security analysis. Finally, conclusions are summarized in Section 7.

2 State of the art

Hall *et al.* propose a remote gambling system ([7]). Each player has a key pair of a public-key cryptosystem. Players use their private key to

authenticate themselves to the on-line casino, and also to sign each message they send. The paper does not describe how key pairs are generated and distributed; this is a relevant issue, *e.g.* because any minor under the legal age for gambling should be unable to register and get a key pair. The random events used in the game are computed jointly by all players using a cryptographic protocol. The protocol ensures that no player is in a privileged position to influence the outcome of the random event.

In [9] a remote gambling system is described. The system has the same security properties as [7]. Nevertheless, its implementation uses multicast, so the proposed system is more efficient as far as communication is concerned.

Proposals [7] and [9] do not present any protocol to play poker without a trusted third party (TTP). Their authors argue that fulfilling the security properties enumerated in [3] without a TTP is too costly. As an example, they quote the work by Edwards in [6], where an implementation of the protocol [4] on three Sparc workstations is reported to have taken eight hours to shuffle a deck.

Recent proposals, like [2] and [10], improve on [4] from the efficiency point of view. Nonetheless, they use zero-knowledge proofs to satisfy all security requirements enumerated in [3]. Their computational and communication costs preclude their commercial use.

Zhao *et al.* present in [11] a payment method for on-line casinos. The payment protocol uses an optimistic TTP. Each bet includes the payment information in encrypted form. The TTP verifies that payment information is correct. If a player loses a bet and refuses to pay, the TTP reveals the payment information to the winner. Again, zero-knowledge proofs are used, which degrade the performance of the protocol.

Aiello *et al.* propose in [1] a gambling system, where players have an electronic device. The device allows players to play off-line. It is based on a smart card that manages the player's credit and ensures game fairness. Our proposal below is based on the same principle to design an efficient and secure mental poker protocol. The difference is that players are on-line and the smart card does all security-critical operations.

3 Security requirements

In Section 1, we have enumerated the steps done by players in an on-line casino. Now we define the security properties that must be guaranteed at each step:

Registration: Registration must collect accurate and truthful information about people wishing to play. This is necessary to detect, *e.g.*, minors under the legal age for gambling, known dishonest players and people with mental diseases related to gambling.

Authentication: The authentication method used by players must be a strong one. It must be resistant against common attacks, for instance birthday and replay attacks.

Credit: Players increase their credit when they make a payment to the on-line casino and decrese ther credit when they make a withdrawal. Consequently, the action to increase or decrease the player's credit must satisfy the same security requirements as an electronic payment:

- *Confidentiality.* the payment information is a private business between the payment issuer (player) and the payment receiver (the on-line casino or the bank).
- *Integrity.* Once the payment has been sent out, no party must be able to modify the payment information.
- *Authentication.* Each message must include a non-malleable and verifiable proof of who is the message originator.
- *Non-repudiation.* Once the payer has sent her payment, she must be unable to repudiate it. Moreover, the payer must obtain a receipt of the payment so that the receiver cannot later deny having been paid.

Bet: When a player places a bet, the following properties must be satisfied:

- *Integrity.* The bet cannot be modified once it has been sent to the on-line casino, neither the player nor the casino can alter the bet.
- *Authentication.* All messages exchanged in a bet are public to all players and the on-line casino. In this way, any game participant can verify the origin of any message.
- *Non-repudiation.* A player cannot repudiate her bet and the on-line casino cannot repudiate a previously accepted bet.

A bet must have at least the following information:

- Bet amount;
- An identifier of the game;
- The concept of the bet, *e.g.* what condition is being betted on.

Game Poker over a network is one of the most complex games from the security point of view. Crépeau [3] enumerated a list of requirements and properties that must be met by a mental poker protocol:

- *Uniqueness of cards.* Traditional decks of cards can be verified before the game starts, and players can be assured that there are

not duplicate cards. In a mental poker protocol players should be able to verify that each card appears once and only once.
- *Uniform random distribution of cards.* In a traditional hand of poker, one player shuffles the deck and the rest of players can see it. Cards are uniform randomly distributed, because the shuffling player cannot influence the result of shuffling. A way to guarantee uniform random distribution in mental poker is for the hand of each player to depend on decisions made by all players.
- *Cheating detection with a very high probability.* A mental poker protocol must detect any attempt to cheat, *e.g* seeing a face-down card, changing a face-up card, etc.
- *Complete confidentiality of cards.* If the deck is face-down then no partial or total information about any card from the deck ought to be disclosed. Also when a player draws a card, the rest of players should not be able to get information on that card.
- *Minimal effect of coalitions.* A secret communication channel between the players of a coalition is possible in mental poker, *e.g.* one player can ring another player to tell her her cards. A mental poker protocol should reduce the effect of coalitions, so that if a player is not cheating then nobody can learn more about her hand, or about the cards in the deck, than what they can infer from the cards in their coalition.
- *Complete confidentiality of strategy.* It is strategically very important in the game of poker that the losing players may keep their cards secret at the end of a hand. The whole concept of bluffing is based on this fact.

4 Architecture

TTP-based mental poker proposals share the common feature that the on-line casino performs most of the above steps: the on-line casino registers players, authenticates them, and manages bets, the credit of players and the entire game. Note that it is the casino who generates the game events (card shuffling, etc.) and controls the game rules.

Allowing the casino to act as a TTP places it in a privileged position: the casino controls the game and at the same time takes part in it. Thus, security in the TTP-based paradigm completely depends on the on-line casino. If the casino security is compromised by an external or an internal attacker, then the result of the game can be manipulated against honest players.

Thus, it is desirable to prevent the casino from being critical to security. To that end, we propose a new gambling system where security is distributed among the following parties: regulator, on-line casino and players.

Each player has a smart card. The regulator (public authority, government, etc.) certifies the smart card and the software in it. The certification is a guarantee on the fairness of the gambling system. Thus, trust as far as the smart card is concerned rests on the public regulator. This should give more guarantees than relying on the on-line casino, which is often located off-shore or in some tax paradise. We next describe each party in our architecture:

Regulator: In a vast majority of countries, on-line gambling is not regulated. This legal void results in a lack of protection for players, and in some cases for the on-line casinos too [5]. In our proposal the game regulator is the government or a public authority. The regulator watches over the rights of the players and on-line casinos. Moreover, the regulator facilitates to players and on-line casinos the fulfillment of their duties when they must declare their earnings. The game regulator issues the smart cards used by players. Every smart card contains a player's key-pair and a sofware application to play on-line. The software allows the following actions: authenticate to players, increase credit, place a bet and play.

On-line casino: The on-line casino authenticates players in a secure way and puts them in touch so that they can start playing with each other. The on-line casino manages the players' accounts (increase credit, decrease credit, place a bet, pay a bet). For each of the above actions, we propose a cryptographic protocol in this paper where the TTP is "distributed" between the regulator and the smart cards. To the extent that they use no centralized TTP, our protocols are TTP-free, albeit in a weak form.

Players: We use the term "players" to denote the set of players plus the software and hardware in the smart cards they use to play remotely.

Protocols: A protocol is described for each of the steps required in the game.

5 The protocols

The following notation is used in order to describe the protocols and procedures presented.

- P_{entity}, S_{entity}: Asymmetric key pair of *entity*, where P_{entity} is the public key and S_{entity} is the private key.
- $S_{entity}[m]$: Digital signature of message m by *entity*, where digital signature means computing the hash value of message m using a collision-free one-way hash function and encrypting this hash value under the private key of *entity*.
- $E_{entity}(m)$: Encryption of message m under the public key of *entity*.
- $H(m)$: Hash value of message m using a collision-free one-way hash function.
- $m_1||m_2$: Concatenation of messages m_1 and m_2.

5.1 Player registration

A player $\mathcal{P}_i$ can play only if she is registered. In the registration process, the player provides her information. This information must be strongly verified, in order to ensure that registered players are legally allowed to gamble.

Carrying out such a verification over the network is a complex problem. However, governments in several countries are promoting the distribution of smart card-based electronic IDs. Basically, such IDs are smart cards containing a key pair certified by the government. The private key never leaves the smart card, so that a high standard of security is achieved. In addition, those smart cards are able to run application software.

We propose to use these electronic IDs in our e-gambling system. The government issuing the IDs (or a governmental authority) is assumed to regulate e-gambling in its territory. This is no extravagant assumption, since most governments have traditionally been involved in gambling or at least gaming (lotteries, etc.). In this way, we can assume that the relevant application software for e-gambling comes already installed in the electronic IDs. Note that including application software in the IDs can be a way to involve the private sector in co-funding electronic ID manufacturing and distribution.

At least, the smart card stores the following data on the player:

$I_{\mathcal{P}_i}$**:** Player identifier. In our protocols, we will use as identifier the hash value of the player's public key certificate.
$Cert_i$**:** Digital certificate of $\mathcal{P}_i$'s public key.
$P_{\mathcal{P}_i}, S_{\mathcal{P}_i}$**:** Public and private keys of player $\mathcal{P}_i$.
$C_{\mathcal{P}_i}$**:** Credit of $\mathcal{P}_i$, initially set to 0.
B**:** Credit card data for $\mathcal{P}_i$.

5.2 Increase/decrease credit

Player $\mathcal{P}_i$ wishes to deposit money in her casino account in order to be able to play. Alternatively, she may be interested in withdrawing money. Let G denote the on-line casino and V denote the amount to be deposited or withdrawn (depending on whether it is a positive or negative value). Credit increase/decrease is performed with Protocol 1.

Protocol 1

1. *$\mathcal{P}_i$ runs Procedure 1 with parameters $Cert_G$ and V in the smart card to increase/decrease her credit and obtain $E_G(A)$ and $S_{\mathcal{P}_i}[E_G(A)]$.*
2. *$\mathcal{P}_i$ sends $E_G(A)$ and $S_{\mathcal{P}_i}[E_G(A)]$ to G.*
3. *G does:*
   (a) *Verify the signature $S_{\mathcal{P}_i}[E_G(A)]$.*
   (b) *Decrypt $E_G(A)$ using the casino's private key S_G to get V and B.*
   (c) *Verify the deposit/withdrawal data V and B.*
   (d) *Update the credit of player $\mathcal{P}_i$ as $C'_{\mathcal{P}_i} := C_{\mathcal{P}_i} + V$.*
   (e) *Compute a receipt R_C for the new credit as $R_C = S_G[I_{\mathcal{P}_i}||C'_{\mathcal{P}_i}]$.*
   (f) *Encrypt R_C and $C'_{\mathcal{P}_i}$ with the public key of $\mathcal{P}_i$ to get $E_{\mathcal{P}_i}(C'_{\mathcal{P}_i}, R_C)$.*
   (g) *Send $E_{\mathcal{P}_i}(C'_{\mathcal{P}_i}, R_C)$ to $\mathcal{P}_i$.*
4. *$\mathcal{P}_i$ checks that her credit has been updated by running Procedure 2 in the smart card.*

Procedure 1 *$[Cert_G, V]$*

1. *Randomly obtain a value r.*
2. *Fetch the player's credit card data B (stored in the card).*
3. *Compute the identifier of the credit update operation $A = r||V||B$.*
4. *Encrypt A using G's public key (extracted from $Cert_G$) to get $E_G(A)$.*
5. *Sign $E_G(A)$ with the player's private key $S_{\mathcal{P}_i}$ to get $S_{\mathcal{P}_i}[E_G(A)]$.*
6. *Return $E_G(A)$ and $S_{\mathcal{P}_i}[E_G(A)]$.*

Procedure 2 *$[E_{\mathcal{P}_i}(C'_{\mathcal{P}_i}, R_C)]$*

1. *Decrypt $E_{\mathcal{P}_i}(C'_{\mathcal{P}_i}, R_C)$ using the player's private key $S_{\mathcal{P}_i}$ to obtain $C'_{\mathcal{P}_i}$ and R_C.*
2. *Verify the digital signature in the receipt R_C.*
3. *Check against the receipt that the credit amount $C'_{\mathcal{P}_i}$ is correct.*

5.3 Start a game

Once a player is registered, he can start a game. To start playing, the on-line casino G and players use Protocol 2.

Protocol 2

1. *G computes a game identifier I_P with Procedure 3.*
2. *G reveals I_P and $S_G[I_P]$ to all players.*
3. *If a player $\mathcal{P}_i$ wishes to enter game I_P, she must go through the following steps:*
   (a) *Create a request to enter game I_P using Procedure 4, which is run in the smart card and yields as output $\rho_i = S_{\mathcal{P}_i}[S_G[I_P], I_{\mathcal{P}_i}]$ and $Cert_i$.*
   (b) *Send $S_{\mathcal{P}_i}[S_G[I_P], I_{\mathcal{P}_i}]$ and $Cert_i$ to G.*
4. *Let us assume that n players have requested their participation in the game. G generates a certificate for participants in game I_P by the following steps:*
   (a) *Sign all requests to participate in the game, that is, $S_G[\rho_1, \ldots, \rho_n]$*
   (b) *Send $S_G[\rho_1, \ldots, \rho_n]$, $\{\rho_1, \ldots, \rho_n\}$ and $\{Cert_1, \ldots, Cert_n\}$ to players who asked to participate.*
5. *Each player who asked to participate verifies $S_G[\rho_1, \ldots, \rho_n]$, $\{\rho_1, \ldots, \rho_n\}$ and $\{Cert_1, \ldots, Cert_n\}$ using Procedure 5 which is run in the smart card.*

Procedure 3

1. *Generate a random r.*
2. *Obtain the current time T.*
3. *Obtain the number of past games N.*
4. *Compute $I_P = r||T||N + 1$.*
5. *Increase N by one unit.*
6. *Sign I_P using the casino's private key to get $S_G[I_P]$.*
7. *Return $S_G[I_P]$ and $Cert_i$.*

Procedure 4

1. *Verify the signature $S_G[I_P]$.*
2. *Create a request to participate in the game: $S_{\mathcal{P}_i}[S_G[I_P], I_{\mathcal{P}_i}]$.*
3. *Return $S_{\mathcal{P}_i}[S_G[I_P], I_{\mathcal{P}_i}]$.*

Procedure 5 *$[S_G[\rho_1, \ldots, \rho_n]$, $\{\rho_1, \ldots, \rho_n\}$, $\{Cert_1, \ldots, Cert_n\}]$*

1. *For $i = 1$ to n do:*

   (a) *Verify whether* $Cert_i$ *has been issued by the regulator's CA.*
   (b) *Verify* ρ_i *with* $Cert_i$.
2. *Verify* $S_G[\rho_1, \ldots, \rho_n]$.
3. *Store* I_P *and certificates* $\{Cert_1, \ldots, Cert_n\}$ *if all verifications are correct.*
4. *Return the verification result (OK or NOT OK).*

5.4 Bet placing

A player $\mathcal{P}_i$ places a bet in a game I_P using the following protocol:

Protocol 3 *$[I_P]$*

1. *$\mathcal{P}_i$ requests to place a bet by running Procedure 6 in the smart card and gets* (I_A, I_A^*).
2. *$\mathcal{P}_i$ sends* (I_A, I_A^*) *to* G.
3. *The on-line casino* G *performs the following steps:*
   (a) *Verify the digital signature* I_A^* *using the public key of* $\mathcal{P}_i$.
   (b) *Verify the bet data: game identifier* I_P*, bet amount* V*, bet concept* K *(what is being betted on).*
   (c) *Verify that* $\mathcal{P}_i$ *has got enough credit, that is, check that* $C_{\mathcal{P}_i} - V \geq 0$*, where* $C_{\mathcal{P}_i}$ *is the player credit.*
   (d) *If the player has got enough credit:*
      i. *Update the player's credit as* $C'_{\mathcal{P}_i} = C_{\mathcal{P}_i} - V$.
      ii. *Compute the receipt* R_A *for the bet* I_A *as* $R_A = S_G[I_A^*]$.
      iii. *Compute the receipt* R_C *for the remaining credit as* $R_C = S_G[I_{\mathcal{P}_i} || C'_{\mathcal{P}_i}]$.
      iv. *Send* $C'_{\mathcal{P}_i}$, R_A *and* R_C *to* $\mathcal{P}_i$.

   Otherwise (the player hasn't got enough credit) the bet is not accepted.
4. *$\mathcal{P}_i$ runs Procedure 7 in the smart card to verify that the on-line casino has updated her credit.*

Procedure 6 *$[I_P, V, K]$*

1. *Obtain a random value* r.
2. *Compute the bet identifier* $I_A = \{I_P || r || V || K\}$*, that is the concatenation of the game identifier,* r*, the bet amount and the bet concept.*
3. *Sign* I_A *with the player's private key* $S_{\mathcal{P}_i}$ *to get* $I_A^* = S_{\mathcal{P}_i}[I_A]$
4. *Return* (I_A, I_A^*).

Procedure 7 *$[R_A, R_C, C_{\mathcal{P}_i}]$*

1. *Verify the digital signature in* R_A.
2. *Verify the digital signature in* R_C.
3. *Check that the credit* $C'_{\mathcal{P}_i}$ *is correct.*

At the end of a game, the casino pays her earnings to player $\mathcal{P}_i$ using the following protocol:

Protocol 4 *[*I_A, I_A^*, R_A*]*

1. *G does:*
   (a) *Verify the signatures on the bet receipt* R_A *and the bet* I_A^*.
   (b) *Compute the earnings* g *of* $\mathcal{P}_i$ *in game* I_P *with bet* I_A.
   (c) *Update the player's credit as* $C'_{\mathcal{P}_i} = C_{\mathcal{P}_i} + g$.
   (d) *Compute the receipt of the available credit* R_C *as* $R_C = S_G[I_{\mathcal{P}_i}||C'_{\mathcal{P}_i}]$.
   (e) *Send* R_C *to* $\mathcal{P}_i$.
2. $\mathcal{P}_i$ *verifies that she got paid by running Procedure 8 in the smart card.*

Procedure 8 *[*$R_C, C_{\mathcal{P}_i}$*]*

1. *Verify the signature on* R_C.
2. *Verify that the new credit for* $C'_{\mathcal{P}_i}$ *is correct.*

5.5 Deck shuffling

Once the game has started with Protocol 2, the smart card of each player contains the certificates of the rest of players. Based on the key identifier field within the players' certificates, an order between players is established: the first player is the one with the lowest identifier. The first player has a singular role. The smart card of the first player (not the player herself) creates a permutation of 52 values, that is, the smart card shuffles the deck; then, following the prescribed player ordering, the smart card of the first player computes the cards for each player. For each of the remaining players, the first player's smart card computes a digital envelope containing the cards of that player. This digital envelope can only be opened by the corresponding player's smart card (player's cards are managed by the player's smart card).

The method for shuffling the deck is described in Protocol 5.

Protocol 5

1. *Let us assume that players* $\{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$ *and the casino* G *start a game using Protocol 2.*

2. *Based on her certificate, each $\mathcal{P}_i$ derives her order in the player ordering.*
3. *$\mathcal{P}_1$ does:*
   (a) *Run Procedure 9 in the smart card and obtain a shuffled deck.*
   (b) *For $i = 2$ to n do:*
      i. *Run Procedure 10 in the smart card to obtain the cards for $\mathcal{P}_i$ encrypted under $\mathcal{P}_i$'s public key and signed under $\mathcal{P}_1$'s private key. Denote the output of the smart card by ξ and $S_{\mathcal{P}_1}[\xi_i]$, where ξ_i are the encrypted cards for $\mathcal{P}_i$.*
      ii. *Send ξ and $S_{\mathcal{P}_1}[\xi_i]$ to player $\mathcal{P}_i$;*
4. *Each player $\mathcal{P}_i$ for $i \in \{2, \ldots, n\}$ recovers her cleartext cards by running Procedure 11 inside her smart card.*

Procedure 9 is used by player $\mathcal{P}_1$ to generate a shuffled deck and compute the cards corresponding to each player.

Procedure 9

1. *Generate a permutation π of 52 elements;*
2. *For $i = 2$ to n do:*
   (a) *Compute the cards for $\mathcal{P}_i$ as $D_i = \{d_{i,1}, \ldots, d_{i,10}\}$, where $d_{i,j} = \pi(5 * (j-1) + i)$ and $j \in \{1, \ldots, 10\}$;*
3. *Initialize the counter k of requested cards and the counter l of discarded cards to $k = 0$ and $l = 0$, respectively.*

The following procedure encrypts player $\mathcal{P}_i$'s cards under that player's public key and signs the result under player $\mathcal{P}_1$'s private key.

Procedure 10 *[i]*

1. *Generate a random R.*
2. *Encrypt D_i, I_P and R under $P_{\mathcal{P}_i}$'s public key to get $\xi = E_{P_{\mathcal{P}_i}}(I_P, D_i, R)$.*
3. *Sign ξ to get $S_{\mathcal{P}_1}[\xi]$.*
4. *Return ξ and $S_{\mathcal{P}_1}[\xi]$.*

Players decrypt their cards by running Procedure 11 in their smart cards.

Procedure 11 *[ξ,$S_{\mathcal{P}_1}[\xi]$]*

1. *Verify the signature $S_{\mathcal{P}_1}[\xi]$ on ξ using the certificate $Cert_1$.*
2. *Decrypt ξ with the player's private key $S_{\mathcal{P}_i}$ and obtain D_i, I_P and R.*
3. *Check I_P is the current game identifier.*
4. *Store D_i in the smart card.*
5. *Initialize the counter k of requested cards and the counter l of discarded cards to $k = 0$ and $l = 0$, respectively.*

5.6 Card draw

A player's smart card keeps track of how many cards it has given to the player, the set τ of cards that are in the hand of the player and the set of cards that have been discarded. When the player wants to draw a card, her smart card checks that she is allowed to do so, *i.e.* that she has got less than five cards in her hand. If yes, the next stored card is given to the player and added to the set τ.

Procedure 12
If $k - l < 5$ *then*

1. *Retrieve the next card* $\tau_{k+1} = d_{k+1}$*, where* $d_{k+1} \in D_i$*.*
2. *Let* $k := k + 1$*.*
3. *Add* τ_{k+1} *to the set* τ*.*
4. *Return* τ_{k+1}*.*

Otherwise return error (player not allowed to draw).

5.7 Card discarding

In the following Procedure 13, if a user discards a card τ_j, the counter l is incremented and τ_j is eliminated from τ.

Procedure 13 *[*τ_j*]*

1. *If* $\tau_j \in \tau$ *then do:*
   (a) *Let* $l := l + 1$*.*
   (b) *Eliminate* τ_j *from* τ*.*
2. *If* $\tau_j \notin \tau$ *then return error.*

5.8 Card opening

If a player wants to show the cards in her hand, she runs the following Procedure 14 in her smart card.

Procedure 14

1. *Sign* τ *to get* $S_{\mathcal{P}_i}[I_P||\tau]$*.*
2. *Return* $S_{\mathcal{P}_i}[I_P||\tau]$ *and* τ*.*

6 Security analysis

Security in our mental poker system depends on whether all steps performed by players in the on-line casino are secure. We will examine whether each protocol or procedure described above fulfills the properties enumerated in Section 3.

Registration: In Section 5.1, we propose that registration be handled by the public authority issuing electronic IDs. Thus, registration is performed in a controlled environment and offers whatever security is provided to register for an electronic ID.

Start a game: In Protocol 2 presented in Section 5.3, the on-line casino acts as a central node that puts players in touch with each other. All actions (game creation, request to participate) done by the parties are signed. Thus message authentication and integrity can be verifies by any player or external party. Also, message non-repudiation is guaranteed.

Credit increase: Protocol 1 described in Section 5.2 encrypts and signs all messages between the player's smart card and the on-line casino, so that confidentiality, authentication, integrity and non-repudiation are ensured.

Bet placing: In Protocol 3 of Section 5.4 messages between the player and the on-line casino are signed. The digital signature ensures message authentication, integrity and non-repudiation. Non-repudiation is especially important, as it prevents the player from repudiating a lost bet and it also prevents the on-line casino from repudiating an accepted bet.

Deck shuffling: The most complex shuffling operations are performed by the smart card. Let us check that Protocol 5 of Section 5.5 meets the security requirements enumerated in Section 3.

- *Uniqueness of cards.* The smart card of $\mathcal{P}_1$ follows Procedure 9 to create a permutation of 52 elements that corresponds to the deck. The permutation ensures that there are no duplicated cards. Cards are distributed to each player so that each card belongs only to a player.
- *Uniform random distribution of cards.* The smart card uses its random generator to obtain a shuffling permutation. We assume that the generator is good enough to ensure uniform random distribution of shuffled cards.
- *Cheating detection with a very high probability.* Thanks to its exclusive knowledge of the player's private key (we assume the smart

card is tamper-resistant enough for its contents to be safely held), the smart card cannot be bypassed by a cheating player. Thus, any cheater will be unable to sign messages and will be detected.
- *Complete confidentiality of cards.* $\mathcal{P}_1$ creates the deck of cards by running Procedure 9 *within the smart card*. Then cards are distributed using Procedure 10: cards exit the smart card encrypted under the public key of the player who requested them. In order to recover a cleartext card, an intruder should be able to decrypt the digital envelope containing the cards; but this cannot be done without the requesting player's private key, which is securely held by that player's smart card.
- *Minimal effect of coalitions.* Cards are initially in the smart card of player $\mathcal{P}_1$ and are subsequently sent to the rest of players in encrypted form. There are two possible attacks for a coalition of players to obtain cards which are not theirs: i) extract the cards from player $\mathcal{P}_1$'s smart card, which is deemed infeasible because of the tamper-resistance of $\mathcal{P}_1$'s smart card; ii) decrypt the cards which are sent in encrypted form, which is deemed infeasible because the private keys needed for decryption are safely held by the smart cards of players having legitimately requested the cards.
- *Complete confidentiality of the strategy.* Revealing players' strategies is not needed to verify the game fairness at the end of the game. The control exerted by the player's and casino's smart cards is deemed sufficient to ensure game fairness and correctness.

Card discarding: Procedure 13 is run inside the smart card. If the discarded card is in the player's hand, the smart card removes it and allows the player to request a new card. The information on the discarded card does not leave the smart card.

Card opening: $\mathcal{P}_i$ can show her cards using Procedure 14. The digital signature on the game identifier I_P and the player's set of cards τ can only be computed using the private key that is held by the smart card. This private key never leaves the smart card, so that the latter cannot be bypassed.

7 Conclusions

We have presented a system whereby players can play poker over a network with a high degree of security. The different parties (players and casino) must use their tamper-resistant smart cards to take part in the game, which leads to secure and simple protocols. The same approach can be extended to other games over a network.

Note. A patent application covering the essentials of the proposed system is in process.

Acknowledgments

The authors are partly supported by the Catalan government under grant 2002 SGR 00170, and by the Spanish Ministry of Science and Education through project SEG2004-04352-C04-01 "PROPRIETAS".

References

1. W. A. Aiello, A. D. Rubin, and M. J. Strauss. Using smartcards to secure a personalized gambling device. In *CCS '99: Proceedings of the 6th ACM conference on Computer and communications security*, pages 128–137, New York, NY, USA, 1999. ACM Press.
2. A. Barnett and N. Smart. Mental poker revisited. In *Proc. Cryptography and Coding*, volume 2898 of *Lecture Notes in Computer Science*, pages 370–383. Springer-Verlag, December 2003.
3. C. Crépeau. A secure poker protocol that minimizes the effect of player coalitions. In Hugh C. Williams, editor, *Advances in Cryptology - Crypto '85*, volume 218 of *Lecture Notes in Computer Science*, pages 73–86, Berlin, 1985. Springer-Verlag.
4. C. Crépeau. A zero-knowledge poker protocol that achieves confidentiality of the players' strategy or how to achieve an electronic poker face. In A. M. Odlyzko, editor, *Advances in Cryptology - Crypto '86*, volume 263, pages 239–250, Berlin, 1986. Springer-Verlag. Lecture Notes in Computer Science.
5. Department for Culture Media and Sport of Great Britain. Gambling review body. http://www.culture.gov.uk/role/gambling_review.html, July 17 2001. chapter 13, page 167.
6. J. Edwards. Implementating electronic poker: A practical exercise in zero-knowledge interactive proofs. Masters thesis, Department of Computer Science, University of Kentucky, May 1994.
7. C. Hall and B. Schneier. Remote electronic gambling. In *13th Annual Computer Security Applications Conference*, pages 227–230. ACM, December 1997.
8. R. M. Needham and M. D. Schroeder. Authentication revisited. *ACM Operating Systems Review*, 21(1), 1987.
9. R. Oppliger and J.L. Nottaris. Online casinos. In *Kommunikation in verteilten Systemen*, pages 2–16, 1997.
10. W.H. Soo, A. Samsudin and A. Goh. Efficient mental card shuffling via optimised arbitrary-sized benes permutation network. In *Information Security Conference*, volume 2433 of *Lecture Notes in Computer Science*, pages 446–458. Springer-Verlag, 2002.
11. W. Zhao, V. Varadharajan and Y. Mu. Fair on-line gambling. In *16th Annual Computer Security Applications Conference (ACSAC'00)*, pages 394–400, New Orleans, Louisiana, December 2000. IEEE.