

# Mental Poker in practice

An extended implementation of  
Schindelhauer's *Toolbox for Mental Card Games*

Heiko Stamer

University of Kassel  
Department of Mathematics/Computer Science  
Heinrich-Plett-Straße 40, D-34132 Kassel  
stamer@theory.informatik.uni-kassel.de  
76F7 3011 329D 27DB 8D7C 3F97 4F58 4EB8 FB2B E14F

2. Krypto-Tag, March 2005

U N I K A S S E L  
V E R S I T Ä T

1 Introduction

2 Toolbox for Mental Card Games

3 Mental Poker Revisited

4 Implementation: `libTMCG`

- A. SHAMIR, R.L. RIVEST, L.M. ADLEMAN: *Mental Poker*  
Technical Report MIT-LCS-TM-125, MIT, February 1979  
Reprinted in *The Mathematical Gardner*, pp. 37–43, 1981
  - Impossibility proof (information-theoretic sense)
  - Protocol for dealing cards (commutative encryption)
  - Basically a two player solution (Alice and Bob)
- R. LIPTON: *How to Cheat at Mental Poker*  
AMS Short Course in Cryptography, AMS, January 1981
  - Attack on [SRA79] with the “quadratic residue” trick
- D. COPPERSMITH: *Cheating at Mental Poker*  
Advances in Cryptology: Proceedings of CRYPTO '85
  - Generalization of the “quadratic residue” trick
  - Attack on [SRA79], if “random padding” is used

- M. BLUM: *Coin Flipping by Telephone: A protocol for solving impossible problems*, Proceedings of CRYPTO '81
  - Security relies on the Quadratic Residuosity Assumption (QRA)
- S. GOLDWASSER, S. MICALI: *Probabilistic Encryption & How to play Mental Poker keeping secret all partial information* 14th ACM Symposium on Theory of Computing (STOC), 1982
  - Unbroken two player solution (QRA), but with serious drawbacks:
    - Exhaustive use of prime numbers
    - After the game all cards have to be publicized, i.e. the players' strategy will be revealed (e.g. unsatisfactory for poker)
  - Led to the important definition of *Semantic Security*.
- I. BANARY, Z. FÜREDI: *Mental Poker with Three or More Players*, Information and Control 59, pp. 84–93 , 1983
- S. FORTUNE, M. MERRIT: *Poker protocols*, Proc. CRYPTO '84

- C. CRÉPEAU: *A secure poker protocol that minimizes the effects of player coalitions*, Proceedings of CRYPTO '85
- C. CRÉPEAU: *A zero-knowledge poker protocol that achieves confidentiality of the players' strategy or how to achieve an electronic poker face*, Proceedings of CRYPTO '86
  - First complete solution to the Mental Poker problem:
    - Uniqueness of cards
    - Uniform random distribution of cards
    - Absence of trusted third party (TTP-free)
    - Cheating detection with a very high probability
    - Complete confidentiality of cards
    - Minimal effect of coalitions
    - Complete confidentiality of strategy
  - All-or-Nothing disclosure of secrets (ANDOS)
  - Drawbacks: uniqueness of cards necessary, no “exotic” card operations available, strict sequential ordering of some protocols

CHRISTIAN SCHINDELHAUER: *A Toolbox for Mental Card Games*,  
Technical Report A-98-14, University of Lübeck, 1998

38. Workshop über Komplexitätstheorie, Datenstrukturen und Effiziente Algorithmen, 1999

- Simple data structure for electronic playing cards
- Several operations on cards and stacks (“toolbox”):
  - Creation of open or covered cards
  - Mask, pickup and public open of (covered) cards
  - Mask, shuffle, pickup and public open a (covered) stack
  - “Just in time” rule control (set properties of stacks)
  - Exotic operations: secretly insert a card into a stack
- Security relies on the *Quadratic Residuosity Assumption*

Players can show the correctness of their performed operations by **interactive Zero-Knowledge Proofs** with max. error probability of  $1/2$ .

- Sequential repeating  $t$  times reduces the probability to  $p \leq 2^{-t}$

Key generation for every player  $i = 1, \dots, k$ :

**Secret key:**  $(p_i, q_i)$   $p_i, q_i \in \mathbb{P}$

**Public key:**  $(m_i, y_i)$   $m_i = p_i \cdot q_i, \ell_m = |m_i|, y_i \in \text{NQR}_{m_i}^\circ, (1 \in \mathbb{QR}_{m_i})$

**Proof of correctness:** (NI)ZK proof (e.g. Gennaro, Micciancio, Rabin)

$$m_i = p^\nu \cdot q^\eta, \nu, \eta \geq 1, p, q \in \mathbb{P} \wedge y_i \in \text{NQR}_{m_i}^\circ$$

Let  $k$  be the number of players and  $M$  the number of different cards:

$$\mathcal{Z} = \begin{pmatrix} z_{1,1} & \cdots & z_{1, \lceil \log_2 M \rceil} \\ \vdots & \ddots & \vdots \\ z_{k,1} & \cdots & z_{k, \lceil \log_2 M \rceil} \end{pmatrix} \quad z_{i,j} \in \mathbb{Z}_{m_i}^\circ$$

Computing the type  $\tau \in [0, M - 1]$  of the card  $\mathcal{Z}$ :

$$b_{i,j} = \begin{cases} 0 & z_{i,j} \in \mathbb{QR}_{m_i} \\ 1 & \text{else} \end{cases} \quad \tau = \sum_{j=1}^{\lceil \log_2 M \rceil} 2^{j-1} \cdot \bigoplus_{i=1}^k b_{i,j}$$



$$b = 00010$$

Binary representation

$\Rightarrow$

$$a_1 = 11001$$

$\oplus$

$$a_2 = 01110$$

$\oplus$

$$a_3 = 10101$$

$=$

$$b = 00010$$

“Secret sharing”

$z_{i,j} \in \mathbb{Q}_{m_i}$   
else

$\parallel a_{i,j} = \begin{cases} 0 \\ 1 \end{cases}$

$\mathcal{Z} =$

$$\begin{pmatrix} z_{1,1} & \dots & z_{1,5} \\ z_{2,1} & \dots & z_{2,5} \\ z_{3,1} & \dots & z_{3,5} \end{pmatrix}$$

$$z_{i,j} \in \mathbb{Z}_{m_i}^{\circ}$$

Encoded card

Mask/Encryption operation  $\mathcal{Z} \mapsto \mathcal{Z}'$

$$z'_{i,j} = z_{i,j} \cdot r_{i,j}^2 \cdot y_i^{b_{i,j}} \pmod{m_i}$$

- Masking secret of the card  $\mathcal{Z}'$ :  $r_{i,j} \in_R \mathbb{Z}_{m_i}^*$ ,  $b_{2,j} \in_R \{0, 1\}, \dots, b_{k,j} \in_R \{0, 1\}$
- Keeping the former type of the card  $\mathcal{Z}$ :  $b_{1,j} = \bigoplus_{i=2}^k b_{i,j}$
- $\mapsto$  is a **equivalence relation**  $\rightsquigarrow$  Thus it is easy to prove the correctness in ZN through  
P:  $\mathcal{Z}' \mapsto \mathcal{Z}''$ , V:  $b \in_R \{0, 1\}$ , P: reveals the secret of  $\mathcal{Z} \mapsto \mathcal{Z}''$  or  $\mathcal{Z}' \mapsto \mathcal{Z}''$ , V: checks

ADAM BARNETT, NIGEL P. SMART: *Mental Poker Revisited*,  
9th IMA International Conference, Cirencester, 2003

- Verifiable  $k$ -out-of- $k$  Threshold Masking Function (VTMF)
  - Protocol framework with semantically secure encryption function
  - Key Generation Protocol, Verifiable Masking Protocol, Verifiable Re-masking Protocol, Verifiable Decryption Protocol
  - Creation of an open card, Masking a card, Creation of a private card, Opening a card, Mask-shuffling the deck, Splitting the deck, Drawing a card from the deck, Rule control, . . .
  - Correctness of the operations is shown by (non-interactive) **honest-verifier zero-knowledge Proofs of Knowledge**
- Main advantage: Reduced size of the card encoding
  - The size of each card is independent of the number of players and (for the most games) independent of the number of different cards.
- Two possible instantiations of a VTMF:
  - Discrete Logarithm based variant (ElGamal encryption)
  - Factoring based variant (Paillier's encryption)

Key Generation: The  $k$  parties agree on ...

- a finite abelian group  $G$  in which the Decision Diffie-Hellman (DDH) problem is believed to be hard,
- a generator  $g \in G$  of sufficient order  $q$ , and
- the sets  $\mathcal{M} = G$  (types),  $\mathcal{R} = \mathbb{Z}_q$  (secrets) and  $\mathcal{C} = G \times G$  (cards).

Secret key:  $x_i$

$$x_i \in \mathbb{Z}_q$$

Public key:  $h_i, h$

$$h_i = g^{x_i}, h = \prod_{i=1}^k h_i$$

Verifiable Masking Protocol  $m \in \mathcal{M} \mapsto \mathcal{Z}$

$$z_1 = g^r, z_2 = m \cdot h^r$$

Verifiable Re-masking Protocol  $\mathcal{Z} \mapsto \mathcal{Z}'$

$$z'_1 = z_1 \cdot g^r, z'_2 = z_2 \cdot h^r$$

- Masking secret of the cards  $\mathcal{Z}$  resp.  $\mathcal{Z}'$ :  $r \in_R \mathbb{Z}_q$ ,  $\ell_r = |r|$  (size of the secret exponent)
- **Proof of Knowledge of Equality of Discrete Logarithms** (Chaum, Pedersen) shows the correctness of the operation, i.e.  $z_1 = g^\alpha \wedge z_2/m = h^\alpha$  resp.  $z'_1/z_1 = g^\alpha \wedge z'_2/z_2 = h^\alpha$

Cre87

J. EDWARDS: *Implementing electronic poker: A practical exercise in zero-knowledge interactive proofs*  
Master's thesis, University of Kentucky, 1994

Cre87

M. PINNA: *A Secure Card Game*, BA-Thesis, Gonville & Caius College, University of Cambridge, May 2002

Sch98

R. HANNA, A. RIDEOUT, D. ZIEGLER: *Secure Poker: Secure Peer-to-Peer Texas Hold'em*, Midterm project/Course 6.857, Massachusetts Institute of Technology, Fall 2003

- The authors “translated” some parts of an early version of the file `SchindelhauerTMCg.cc` to `Player.java` without referencing the original source but claiming the “first known implementation”.
- ★★★ RANT ★★★ SCO-style proof-of-code-copying:
  - Identifier names of temporary variables: `foo`, `bar`, `lej`
  - Similar comments: e.g. `m in SF' (square free integer)`
  - Suspicious function nomenclature: e.g. `glueStacks` vs. `TMCg_GlueStackSecret` (Has in effect nothing to do with glue!)
  - Superfluous code: e.g. restricting the  $m_i$ 's for SAEP
- Soundness of (their) key generation proof is questionable!

## H. STAMER: *The free (like in speech and beer) C++ library libTMCG*

<http://savannah.nongnu.org/projects/libtmcg>

- $\approx 6700$  lines of code + the GNU libraries GMP and GCRYPT
- VTMF: DDH-hard group  $G = \mathbb{QR}_p$  with  $p = 2q + 1$  and  $p, q \in \mathbb{P}$
- TMCG:  $p_i, q_i \equiv 3 \pmod{4}$ ,  $p_i \not\equiv 1 \pmod{8}$ ,  $p_i \not\equiv q_i \pmod{8}$
- Three straight forward optimizations:
  - 1 VTMF: Use the particular generator  $g = 2$ , i.e.  $p \equiv 7 \pmod{8}$
  - 2 VTMF: Use shortened exponents, e.g.  $\ell_r = 160$  bit, to improve the computational and communication efficiency  
*Discrete Logarithm with short exponent assumption (DLSE)*  
 T. KOSHIBA, K. KUROSAWA: *Short Exponent Diffie-Hellman Problems*, PKC 2004 (Short, Full)-DDH  $\equiv_{\mathcal{P}} \text{DDH}_G + \text{DLSE}_G$
  - 3 TMCG, VTMF: Use shortened commitments in the ZN proofs
- ROM: Are Random Oracles still practical?
- **Patent issues?!** (Schnorr Signatures, Elliptic Curve VTMF-DLOG variant)
- TODO: Missing operations, documentation, examples, ...

<code>mulm</code>	modular multiplication, inversion, etc.	$O((\log_2 n)^2)$
<code>powm</code>	modular exponentiation	$O((\log_2 n)^3)$
<code>spowm</code>	“blinded” modular exponentiation	$= 2\text{powm} + 3\text{mulm}$

	TMCG $p_i, q_i \equiv 3 \pmod{4}$ [Sch98]	VTMF Discrete Logarithm [BSm03]
Encryption of a card	$= 3k \lceil \log_2 M \rceil \text{mulm}$	$= 2\text{spowm} \langle \ell_r \rangle + 2\text{mulm}$
Prover	$= t \cdot 5k \lceil \log_2 M \rceil \text{mulm}$	$= 2\text{spowm} \langle \ell_q \rangle + 5\text{mulm}$
Verifier	$= t \cdot 3k \lceil \log_2 M \rceil \text{mulm}$	$= 4\text{powm} \langle \ell_r, \ell_q \rangle + 6\text{mulm}$
Decryption of a card	$= 2 \lceil \log_2 M \rceil \text{mulm}$	$= 1\text{spowm} \langle \ell_q \rangle + 2\text{mulm}$
Prover	$\leq \lceil \log_2 M \rceil ((4t + 5)\text{mulm} + 2\text{powm} \langle \ell_m/2 \rangle)$	$= 3\text{spowm} \langle \ell_q \rangle + 1\text{mulm}$
Verifier	$\leq \lceil \log_2 M \rceil (2t + 2)\text{mulm}$	$= 4\text{powm} \langle \ell_q \rangle + 3\text{mulm}$
Shuffle of a stack $\mathcal{S}$	$=  \mathcal{S}  \cdot 3k \lceil \log_2 M \rceil \text{mulm}$	$=  \mathcal{S}  \cdot 2\text{spowm} \langle \ell_r \rangle + 2\text{mulm}$
Prover	$\approx t \cdot  \mathcal{S}  \cdot 3k \lceil \log_2 M \rceil \text{mulm}$	$\approx t \cdot  \mathcal{S}  \cdot 2\text{spowm} \langle \ell_r \rangle + 2\text{mulm}$
Verifier	$\approx t \cdot  \mathcal{S}  \cdot 3k \lceil \log_2 M \rceil \text{mulm}$	$\approx t \cdot  \mathcal{S}  \cdot 2\text{powm} \langle \ell_r \rangle + 2\text{mulm}$

The german card game **Skat** (<http://www.ispaworld.org/>):

- The deck consists of  $M = 32$  different cards. It is shuffled (and sometimes cutted; not considered here) at begin of each game.
- Each of the three players take 10 hidden cards. The *Skat* (remaining two cards) can be revealed by the successful bidder.
- Each player opens one of his hidden cards in each round (trick).

$t$	Cheating probability	Compressed network traffic for each game and each player	
	$p \leq 2^{-t}$	OpenSkat $\leq 1.9$	SecureSkat [2]
		$\ell_m = 1024$ bit, QRA [Sch98]	$\ell_p = 1024$ bit, $\ell_r = 160$ bit, DDH [BSm03] + DLSE
2	$\leq 0.25$	$\approx 3$ MB	$\approx 0.32$ MB (0.32 MB)
4	$\leq 0.0625$	$\approx 5$ MB	$\approx 0.36$ MB (0.44 MB)
8	$\leq 0.00390625$	$\approx 10$ MB	$\approx 0.4$ MB (0.69 MB)
16	$\leq 0.00001526$	$\approx 20$ MB	$\approx 0.51$ MB (1.16 MB)
32	$\leq 2.3284 \cdot 10^{-10}$	$\approx 40$ MB	$\approx 0.71$ MB (2.12 MB)
64	$\leq 5.4211 \cdot 10^{-20}$	$\approx 80$ MB	$\approx 1.09$ MB (4.04 MB)

- 1 E-Gambling: Realize other solitary, board or casino games
  - C. CRÉPEAU, J. KILIAN: *Discreet Solitary Games*, Crypto '93
- 2 Secret Voting
  - Simple idea (implemented in SecureSkat):
    - (1) Each participant create a private card (vote).
    - (2) The cards are stacked and this stack is shuffled by each participant.
    - (3) The cards of the stack are disclosed to all participants.
- 3 Secure Multiparty Computation
  - B. DEN BOER: *More efficient match-making and satisfiability: The five card trick*, Eurocrypt '89
  - V. NIEMI, A. RENVALL: *Secure multiparty computations without computers*, Theoretical Computer Science 191, 1998
  - A. STIGLIC: *Computations with a deck of cards*, Theoretical Computer Science 259, 2001
    - Improved AND-protocol (Las Vegas algorithm)
    - Implemented (`StiglicMPC.cc`), but not yet used

- [Sch98] Christian Schindelhauer  
A Toolbox for Mental Card Games  
Technical Report A-98-14, University of Lübeck, 1998  
<http://citeseer.ist.psu.edu/schindelhauer98toolbox.html>
- [BSm03] Adam Barnett, Nigel P. Smart  
Mental Poker Revisited  
9th IMA International Conference, LNCS 2898, pp. 370–383, 2003
- [St04] Heiko Stamer  
Kryptographische Skatrunde  
Offene Systeme 4 (2004), pp. 10–30, 2004
- [1] LibTMCG: <http://savannah.nongnu.org/projects/libtmcg>
- [2] SecureSkat: <http://savannah.nongnu.org/projects/secureskat>