A New Approach to Practical Active-Secure Two-Party Computation

Jesper Nielsen, Peter Nordholdt
Claudio Orlandi, Sai Sheshank

http://eprint.iacr.org/2011/091

Online Poker

2♠, 5♠ ,2♥,5 ♥,J♦

Q♠,Q♣,7♣,3♥,2♦

10 ♠,9♣, 8♣,7♦,6♦

3♠, 4♠,7♥,Q ♦,10♦

Poker With Pirates

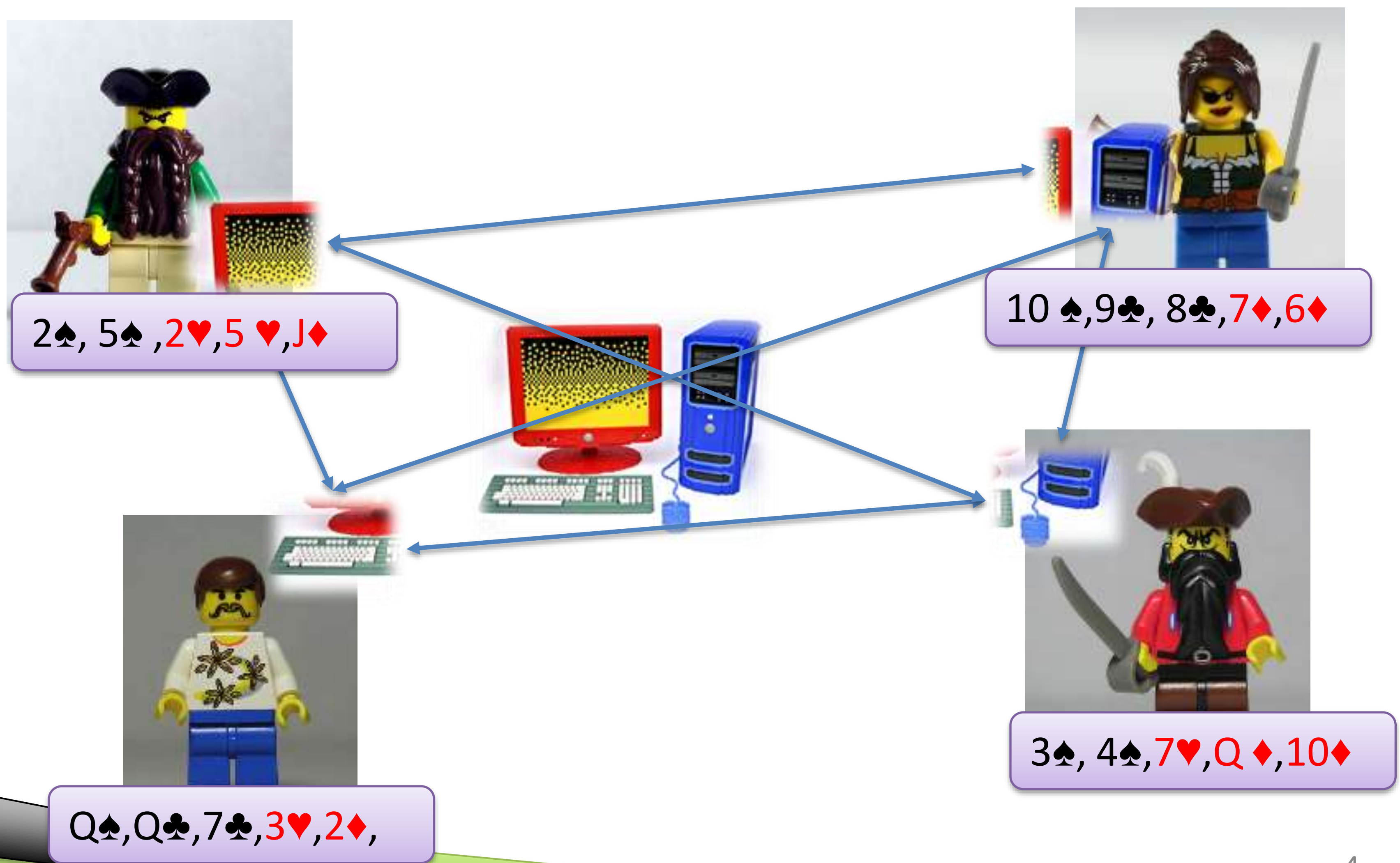
2♠, 5♠ ,2♥,5 ♥,J♦

Q♠,Q♣,7♣,3♥,2♦,

10 ♠,9♣, 8♣,7♦,6♦

A♠,A♣,A♥,A♦,K♦

Secure Computation



PRACTICAL MPC?
YES, WE CARE!

Hospitals and Insurances

Syge mister millioner af kroner

Af CHARLOTTE BEDER
Offentliggjort 19.02.09 kl. 08:39

Danskerne går årligt glip af 80 mio. kr., fordi de ikke aner, at de er forsikret ved kritisk sygdom.

Brystundersøgelse. Foto: Colourbox

Relaterede artikler

- Nyt system sikrer syge 80 mio. kr.
- Forsikringsklager i bund
- Boom i sundhedsforsikringer

Hundredvis af alvorligt syge danskere går hvert år glip af millioner af kroner, fordi de ikke har overblik over deres forsikringsdækning.

Derfor kontakter de ikke deres pensions- eller forsikringsselskab, når de bliver ramt af kræft, blodpropper eller anden kritisk sygdom. Og så får de aldrig den check på typisk mellem 50.000 og 200.000 kr., som de har ret til, lyder det fra forsikrings- og pensionsbranchen.

»Forudsætningen er, at systemet skrues sammen på en måde, så selskaberne ikke får andre oplysninger om kunderne, end de bør få. For det enkelte individ må ikke miste kontrollen over egne helbredsoplysninger,« siger jurist Lars Kofod.

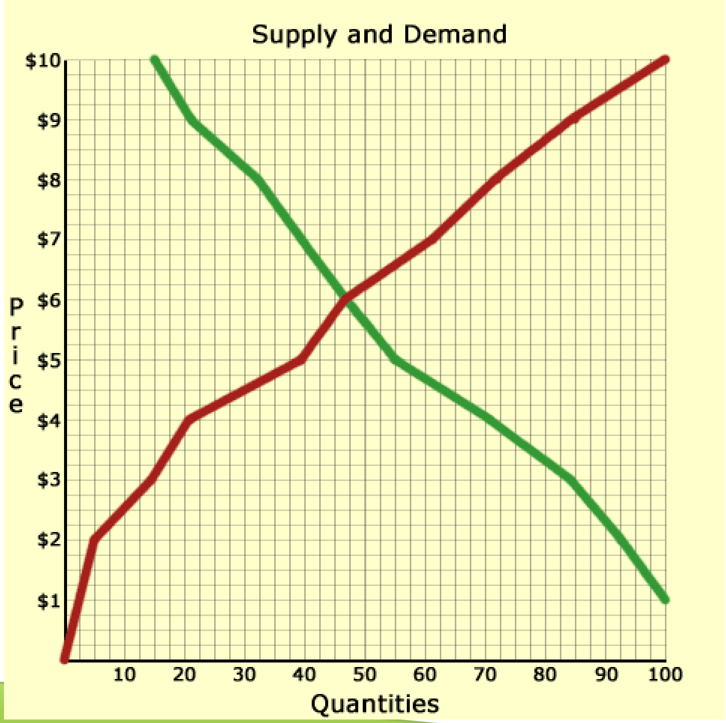
- **Problem:** Sick people forget to claim their insurance money
- **Solution:** Insurances and hospitals could periodically compare their data to find and help these people
- **Privacy Issue**: insurance and medical records are sensitive data! No other information than what is strictly necessary must be disclosed!

Sugarbeets Farmers

Bogetoft et al.

“Multiparty Computation Goes Live”

- January 2008
- **Problem**: determine market price of sugar beets contracts
- 1200 farmers
- Computation: 30 minutes
- *Weak security* ☹
  - *Passive adversary*
  - *Honest majority*



ICT Benchmarking

sharemind

- Estonian ICT companies want to **benchmark** their performances
- Statistics on **private** information (salaries, profit, …)
- Use MPC for **privacy** and **correctness**
- *Weak security* ☹

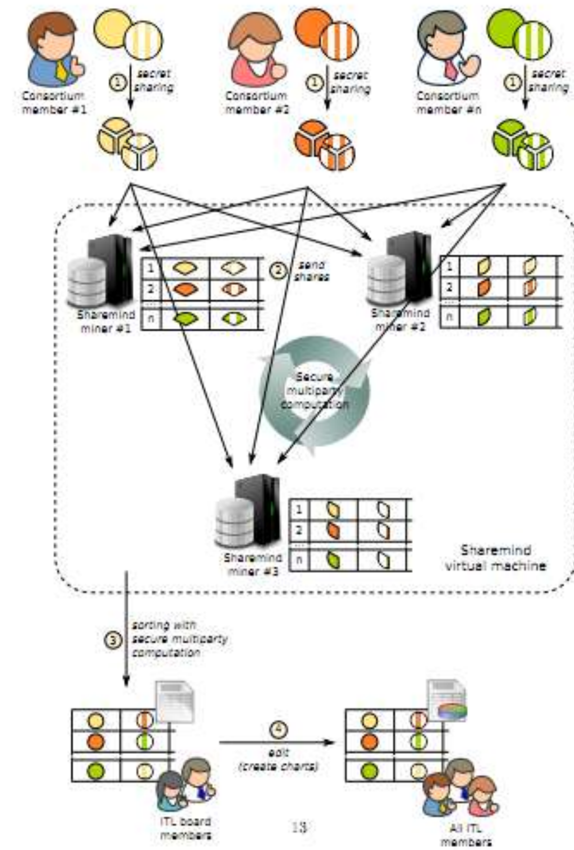
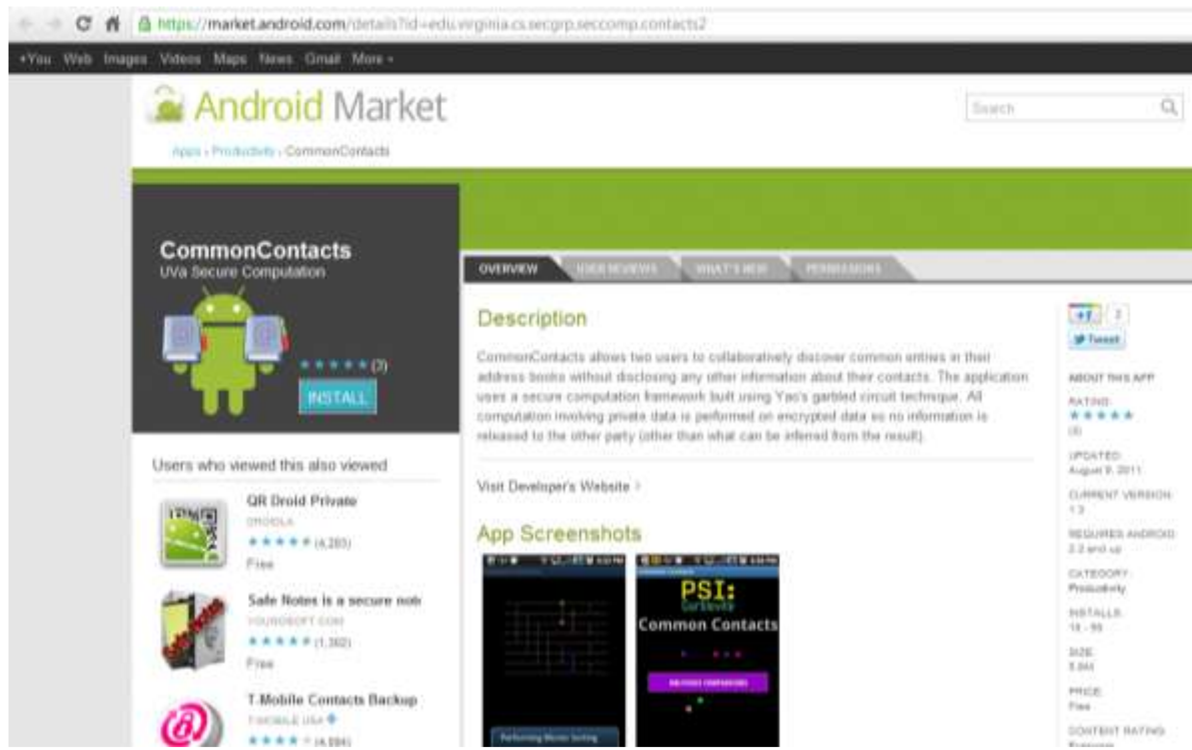


Figure 3: Data flow and visibility in the improved solution using the SHAREMIND framework.

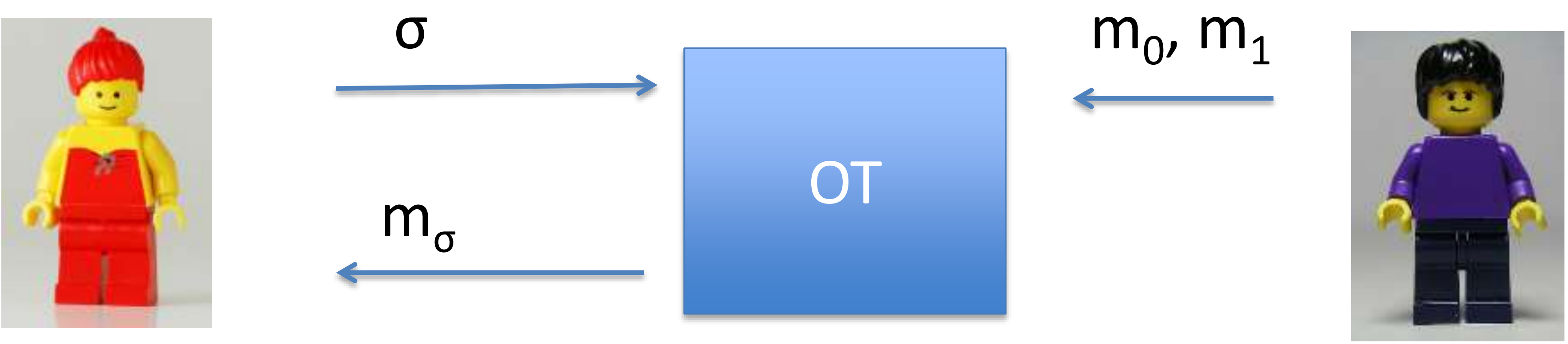
Private Contact Discovery

- Private Set Intersection of Address Books between two smartphones!
- *mightbeevil.org*
- *Weak security* ☹



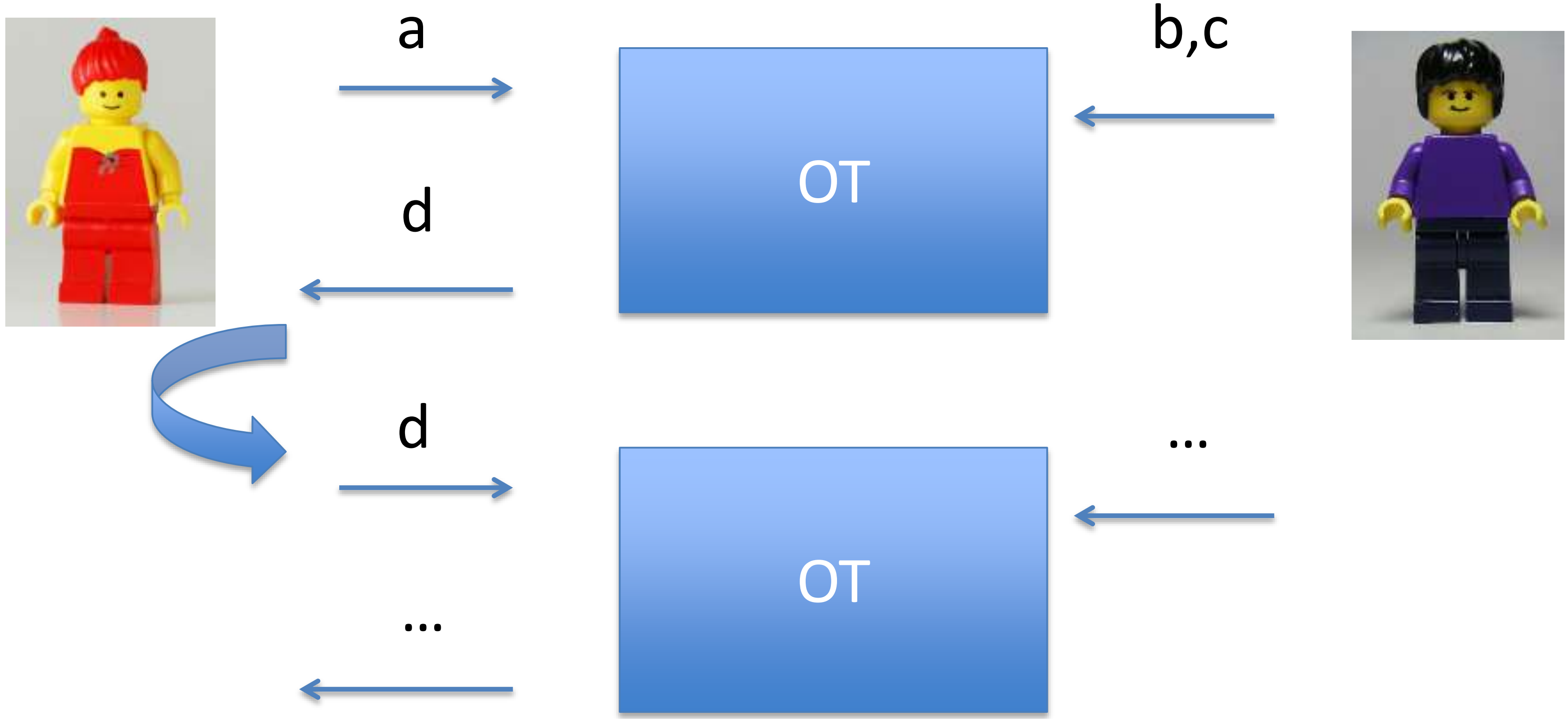
THIS TALK IN 5 MINUTES

Oblivious Transfer

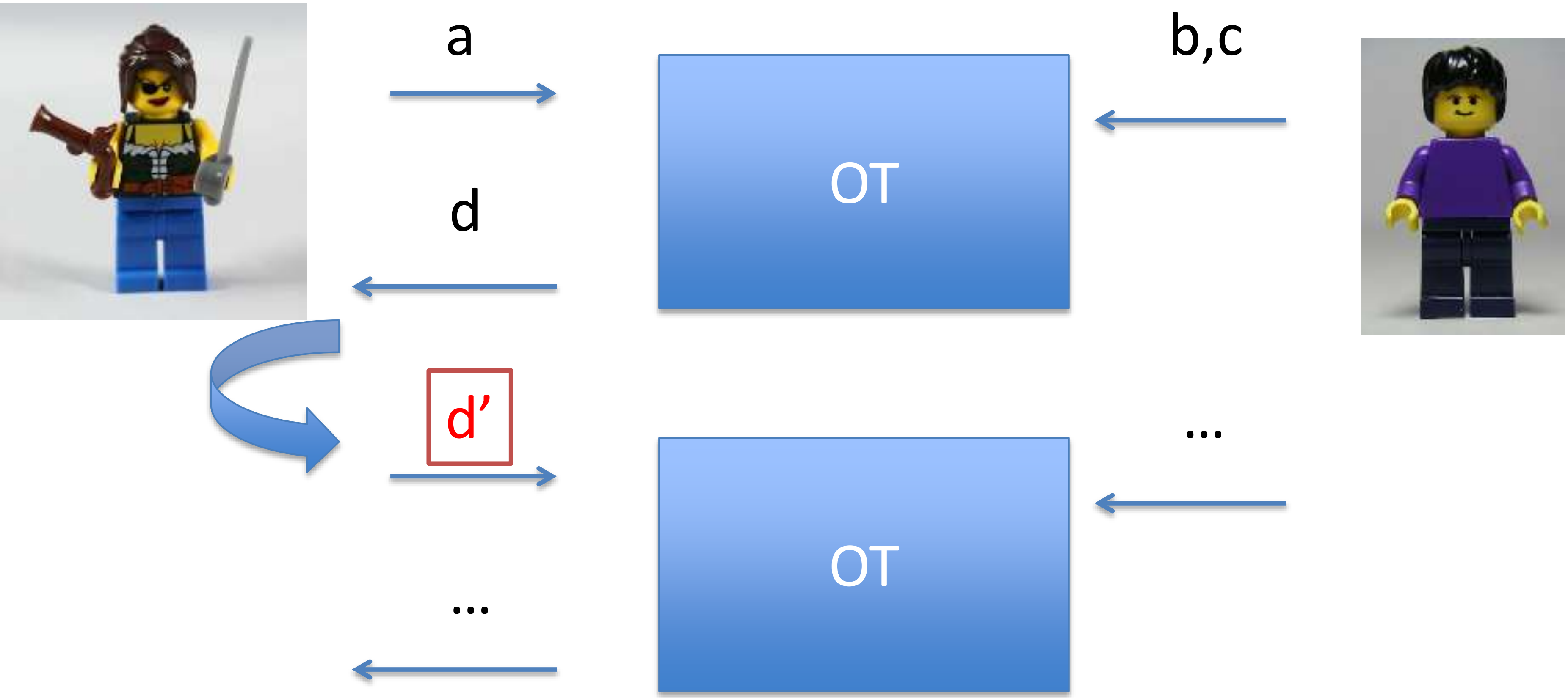


OT Based Computation

OT Based Computation



OT Based Computation



Authenticated Oblivious Transfer

a, …

b,c,K, …

aOT

$M = MAC_K(d)$

d,M

d’,M’

…, K

aOT

If $M' \neq MAC_K(d')$
abort

…

Protocol Structure

Two Phases	Goal	Requires	Time
Preprocessing	Create many aOTs	Public Key Crypto	90-99%
Online Phase	Use aOTs to compute $f(x_1, \dots, x_n)$	No Crypto	1-10%

2PC Techniques

- Boolean Constant Rounds:
  - Yao + Zero Knowledge: JS‘07
  - Yao + Cut-And-Choose: LP‘07, NO‘09, LP‘11
- Boolean Non-constant Rounds:
  - MPC in the head: IPS‘08, LOP‘11
  - GMW + MACs: this work
- Arithmetic Computation:
  - CDN’02, IPS‘09, DO‘10, BDOZ’11

2PC Performances

- Securely computing Boolean circuits against active adversary.

	Rounds	Gates	Time	Gates/sec
Pinkas, Schneider, Smart, Williams (ASIACRYPT'09)	O(1)	~30000	20 min	30
Shelat, Shen (EUROCRYPT'11)	O(1)	~30000	3 min	190
This Work	O(d)	~900000	45 sec	20000
Huang, Evans, Katz, Malka (USENIX'11) [passive]	O(d)	~30000	0.2 sec	170000

OT BASED 2PC

GMW - Two-Party AND

- Input: Alice $x_A, \mathrm{y_A}$ Bob x_B, y_B
- Output: z_A, z_B s.t.

$$z_A \oplus z_B = (x_A \oplus x_B)(y_A \oplus y_B)$$

$$x_A y_A \oplus x_A y_B \oplus x_B y_A \oplus x_B y_B$$

Need interaction

Can be computed locally

Two-Party AND

- Input: Alice $x_A, \mathrm{y_A}$ Bob x_B, y_B
- Output: z_A, z_B s.t. $\mathrm{z}_A \oplus z_B = (x_A \oplus x_B)(y_A \oplus yB)$
- OT:
  - Alice picks random r_A and transfers $\begin{cases} m_0 = r_A \\ m_1 = r_A \oplus x_A \end{cases}$
  - Bob input choice bit y_B, to the OT
  - Bob retrieves $s_B \stackrel{\text{def}}{=} m_{y_B} = r_A \oplus x_A y_B$
  - Bob computes $z_B = s_B \oplus r_B \oplus x_B y_B$
  - Symmetric for $z_A = s_A \oplus r_A \oplus x_A y_A$

Two-Party AND

- Input: Alice $x_A, \mathrm{y_A}$ Bob x_B, y_B
- Output: z_A, z_B s.t. $\mathrm{z}_A \oplus z_B = (x_A \oplus x_B)(y_A \oplus yB)$
- OT:
  - Alice picks random r_A and transfers $\begin{cases} m_0 = r_A \\ m_1 = r_A \oplus x_A \end{cases}$
  - Bob input choice bit y_B, to the OT
  - Bob retrieves $s_B \stackrel{\text{def}}{=} m_{y_B} = r_A \oplus x_A y_B$
  - Bob computes $z_B = s_B \oplus r_B \oplus x_B y_B$
  - Symmetric for $z_A = s_A \oplus r_A \oplus x_A y_A$

NEW TOOLS: AUTHENTICATED BIT, AND, OT

Authenticated Bit

- We say Alice has an authenticated bit x if:
  - Alice holds x, M
  - Bob holds K, Δ
  - S.t.

$$M = K \oplus x\,\Delta$$

- Where if S is a string:
  - $0S = 0^{|S|}$ and $1S = S$

Authenticated Bit

- Dealer:
  - Gives Alice $x, M = K \oplus x\Delta$
  - Gives Bob K, Δ
- Reveal:
  - Alice sends Bob (x', M')
  - Bob accepts x' if $M' = K \oplus x'\Delta$
- "Reveal $x' \neq x$" $\equiv$ "Guess Δ",
  - $(K \oplus x\,\Delta) \oplus (K \oplus x'\Delta) = \Delta$

Authenticated Bit(s)

- To authenticate $x_1, x_2, \ldots$
  - Alice holds $x_1, x_2, M_1, M_2, \ldots$
  - Bob holds $K_1, K_2, \ldots, \Delta$ *(Δ is the "global key")*
  - S.t. $M_i = K_i \oplus x_i\Delta$
- If $x_3 = x_1 \oplus x_2$
  - $M_3 := M_1 \oplus M_2$
  - $K_3 := K_1 \oplus K_2$
- Then $M_3 = K_3 \oplus x_3\Delta$

Authenticated Bit

- From now on: $[x_i]_A$ to say
  - A holds x_i, M_i
  - B holds K_i, Δ
- $[x_3]_A = [x_2]_A \oplus [x_1]_A$ means:

  $x_3 = x_1 \oplus x_2,$

  $K_3 = K_1 \oplus K_2,$

  $M_3 = M_1 \oplus M_2$
- *Symmetric for Bob* $[y]_B$

Authentication of a Constant

- "Authenticate" a constant bit $b \rightarrow [b]_A$
  - Alice sets $M = 0^n$
  - Bob sets $K = b\Delta$
  - Then

$$M = K \oplus b\Delta = b\Delta \oplus b\Delta = 0^n$$

ONLINE PHASE:
EFFICIENT 2PC WITH ABIT,AAND,AOT

For now: assume a trusted dealer

- Provides random Authenticated Bits:
  - $[x_1]_A, \ldots, [x_n]_A \ldots$, with $x_i \in_R \{0,1\}$
- Random Authenticated (local) ANDs
  - $[a]_A, [b]_A, [c]_A,$
  - With $a, b \in_R \{0,1\}, c = ab$
- Random Authenticated OTs
  - $[c]_A, [y]_A, [x_0]_B, [x_1]_B$
  - With $y = x_c = c(x_0 \oplus x_1) \oplus x_0$
- *And symmetric ones for Bob's side*

Using Random Bits

- To authenticate an secret bit x

  (using a random authenticated bit $[r]_A$)

  - Alice announces: d = r $\bigoplus$ x,
  - Alice and Bob: $[x]_A = [r]_A \bigoplus d$

Using Random (local) ANDs

- To compute the AND of $[x]_A, [y]_A$

  (using a random AND $[a]_A, [b]_A, [c]_A$)

  – Compute and reveal: $[d]_A = [a]_A \oplus [x]_A$

  – Compute and reveal: $[e]_A = [b]_A \oplus [y]_A$

  – Compute: $[xy]_A = [c]_A \oplus e[x]_A \oplus d[y]_A \oplus ed$

$0[x]_A = [0]_A$ and $1[x]_A = [x]_A$

Using Random OTs

- OT with choice bit $[c]_A$ and messages $[x_0]_B, [x_1]_B$
- *(Using a random OT* $[b]_A, [z]_A, [y_0]_B, [y_1]_B$ *,* $z = y_b$*)*
  - Compute and open: $[d]_A = [b]_A \oplus [c]_A$
  - Compute and open: $[f_0]_B = [y_d]_A \oplus [x_0]_A$
  - Compute and open: $[f_1]_B = \left[y_{1\oplus d}\right]_A \oplus [x_1]_A$
  - Compute: $[w]_A = [z]_A \oplus [c]_A(f_0 \oplus f_1) \oplus f_0$
- At the end: $w = x_c$

Evaluate 2-Party Gate (AND)

- Input: Alice $[\mathrm{x_A}]_\mathrm{A}$, $[y_A]_A$, Bob $[x_B]_B, [y_B]_B$
- Output: $[z_A]_A, [z_B]_B$
- Alice picks random $[r_A]$ and transfers

$$\begin{cases} [m_0]_A = [r_A]_A \\ [m_1]_A = [r_A]_A \oplus [x_A]_A \end{cases}$$

- Bob input choice bit $[y_B]_B$, to the OT
- Bob retrieves

$$[s_B]_\mathrm{B} \stackrel{\text{def}}{=} \left[m_{y_B}\right]_B = [r_A \oplus x_A y_B]_B$$

- Compute

$$[z_B\,]_B = [s_B \;\oplus\; r_B \oplus x_B y_B\,]_B$$

- Symmetric for $[z_A\,]_A = [s_A \;\oplus\; r_A \oplus x_A y_A\;]_A$

Deferred MAC Checks

- Before any output is opened, cheating is useless.
- “Deferred” MAC checks:
  - compute hash tree of all MACs thet should have been sent/received
  - compare only at the end before output reconstruction.
- Communication Complexity:
  - From $O(k|C|)$ to $O(|C|+k)$, essentially same as passive security!

PREPROCESSING PHASE: CONSTRUCTING AUTHENTICATED BITS

First Attempt

- Obs: Authenticated Bit = OT with constant diff.
- Alice and Bob run OTs
  - Alice chooses x_i
  - Bob transfers $(K_i, K_i \oplus \Delta)$
  - Alice receives $M_i = K_i \oplus x_i \Delta$
- *If Bob is corrupted, can input different Δ's in different OTs!*

"Cut and Choose"

- Alice and Bob run n OTs
  - Alice chooses x_i
  - Bob transfers $(K_i, K_i \oplus \Delta_i)$ *(If Bob is honest $\Delta_i = \Delta$ for all i)*
  - Alice receives $M_i = K_i \oplus x_i \Delta_i$
- Alice randomly pairs the instances
- For all pairs (j,k) in parallel:
  - Alice reveals $d = x_j \oplus x_k$
  - Bob commits to $U = K_j \oplus K_k \oplus d\,\Delta$
  - Alice reveals $V = M_j \oplus M_k$ *(If A,B honest U=V)*
  - Bob opens. If $U \neq V$ abort.

Analysis

- If Bob is dishonest ($\Delta_j \neq \Delta_k$) computing U = V is equivalent to guessing (x_j, x_k)
  - $V = M_j \oplus M_k = K_j \oplus K_k \oplus (x_j\Delta_j \oplus x_k \Delta_k)$
  - Bob knows just $d = x_j \oplus x_k$, he wins with prob. ½
  - If too many $\Delta_j \neq \Delta$, Alice detects with overwhelming probability
  - *Result: "Leaky" Authenticated Bits*

Another look at the result

$$M_1 \; M_2 \; \ldots \; M_n = K_1 \; K_2 \; \ldots \; K_n \oplus x_1\Delta \; x_2\Delta \; \ldots \; x_n\Delta$$

Turn your head

n auth. bits for Alice with m-long bit key

=

m auth. bits for Bob with n-long bit key

Leakage of k Alice bits $\{x_i\}$ ➔

Leakage of k bits of the new "global key" X

N_n L_n d_nX

$X = (x_1 || x_2 || \ldots || x_n)$ $\Delta = (d_1 || d_2 || \ldots || d_m)$

Result

- Obs. 1: OT messages can be stretched with PRG
- Obs. 2: leakage of X can be removed with universal hashing

- Start with O(n) n-bit OTs
- Then:
  - From n-bit OTs to poly(n) OTs (via PRG)
  - Random pairing, “Cut-and-Choose”
  - Turn your head: leakage of bits ➔ leakage of the key
  - Universal hashing: auth. bits with shorter, full entropy key.
- End with poly(n) authenticated bits

Construct aOT, aAND

- We then combine few aBits to construct aOT and aAND
  - Use cut-and-choose like *LEGO (TCC'09)*
  - Replication factor only

$$\ell = O(s/ln|C|),$$

  - *e.g., C="AES", then*

$$|C| \approx 35000, \ell = 4 \Rightarrow \text{security} \approx 1 - 2^{-50}$$

IMPLEMENTATION

Implementation

- Proof of concept implementation in Java
- m “Oblivious AES” evaluation in parallel:
  - Circuit optimized for “XOR for free” (thanks to Pinkas, Schneider, Smart and Williams)
  - Circuit depth: 40 (is non-constant round really an issue?)
- Aarhus University’s intranet
- Intel Xeon 2.40GHz duo-core

Timings

- Computational security parameter 120
  - i.e. the adversary gets hash of a 120 bit string
- Statistical security parameter s
  - I.e. the adversary wins the cut-and-choose w.p. $< 2^{-s}$

m	Gates	s	T_pre	T_onl	T_tot/m	Gates/T_tot
1	$3.4 \cdot 10^4$	55	38	4	42	822
27	$9.2 \cdot 10^5$	55	38	5	1.6	21545
54	$1.8 \cdot 10^6$	58	79	6	1.6	21623
81	$2.7 \cdot 10^6$	60	126	10	1.7	20405
…						
16384	$5.5 \cdot 10^8$	83	46584	517	2.9	11874

Conclusions

- New technique for active-secure 2PC
  - GMW + MACs and OT extension
  - Security in the OT-hybrid model + Random Oracle
  - Fastest protocol up to date, 20000 gates/second
  - (Only a factor 10 from passive security)
- How to further reduce the gap?

2^{nd} Bar-Ilan Winter School "Lattice-Based Cryptography"

- February 19-22, 2012 @BIU, Tel Aviv, Israel
- Speakers:
  - Craig Gentry, Chris Peikert,
    Vadim Lyubashevsky, Oded Regev
- The school is free (some stipends for travel/hotel available)

http://crypto.biu.ac.il/winterschool2012