Elektronische Wahlen nach dem Schachmatt

Martin Hirt

ETH Zurich

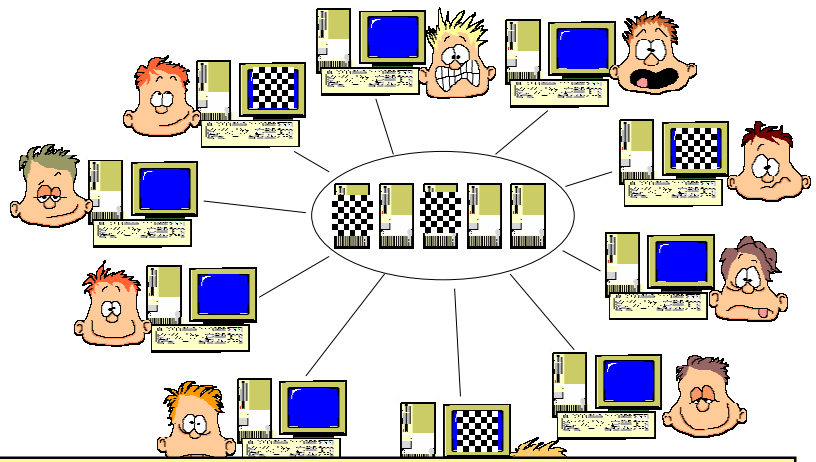
E-Voting Workshop, 5. Juni 2009

Outline

Talk Outline

- Motivation
- Yes/No Voting Protocol
- K-out-of-L Voting Protocol
- Receipt-free Yes/No Voting Protocol
- Receipt-free K-out-of-L Voting Protocol
- Conclusions

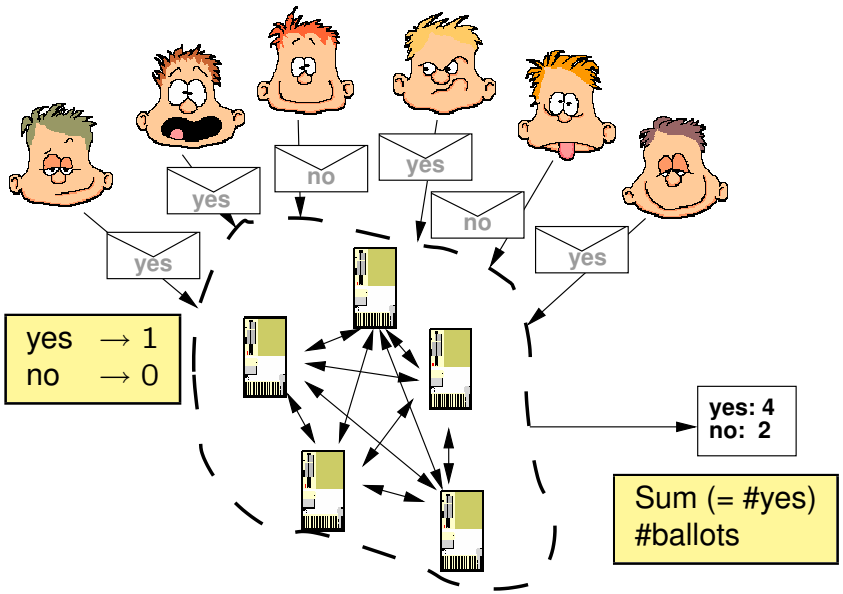
Electronic Voting: Motivation



Summary

- voter must trust his own computer (can control it)
- voter must trust some of the servers

Yes/No Voting Protocol



Security Requirements

Correctness

- validity of ballots (in {yes,no}, entitled voter, $\leq$ 1 ballots)
- tallying (correct and *verifiable* sum)
- verifiability (anyone(?) can verify tally)

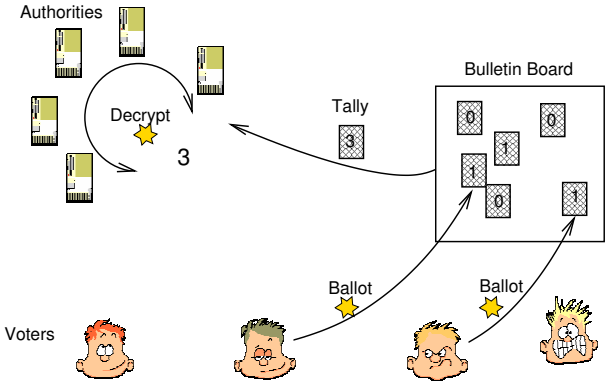
Privacy

- secrecy (cannot determine voter's vote)
- anonymity (who casts a vote?)
- independence (no partial results)

Availability

- accessibility (physical & logical)
- robustness (cannot abort)

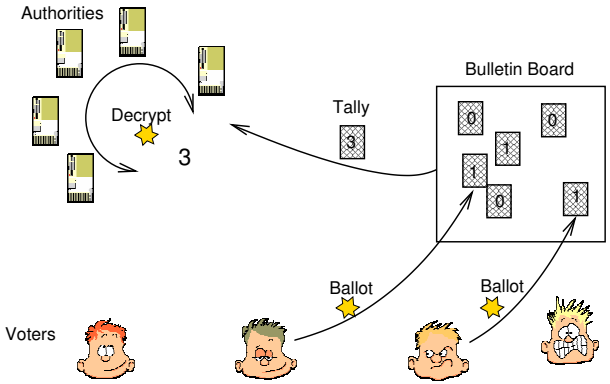
Voting Schemes based on Homomorphic Encryption



Basic Ideas

- ballot = encrypted vote
- abstraction: Bulletin Board
- encryption is homomorphic $\rightarrow$ anyone can add encryptions
- protocol for threshold decryption

Voting Schemes based on Homomorphic Encryption



Protocol Sketch

1. authorities generate SK/PK, SK is shared
2. voters send encrypted vote, validity proof, signature onto BB
3. anybody can compute encrypted tally (due to homomorphism)
4. authorities jointly decrypt and prove tally

Model

Entities

- N authorities
- M voters

Communication

- bulletin board (public channels)

PKI

- each authority A_i has a secret key; public keys are known
- each voter has a signing key; verification keys are known

Generality

- L valid votes $\mathcal{V} = \{v_1, \ldots, v_L\}$, e.g., $\mathcal{V} = \{0, 1\}$

Security

- correctness $\Leftarrow$ at least t honest authorities tally
- privacy $\Leftarrow$ less than t authorities are curious

Homomorphic Encryption Function

Encryption function: $(v, \alpha) \mapsto E(v, \alpha)$

Requirements

- semantically secure (w.r.t. v)
- homomorphic: $E(v_1, \alpha_1) \otimes E(v_2, \alpha_2) = E(v_1 + v_2, \alpha_1 + \alpha_2)$
- distributed set-up (threshold security)
- verifiable decryption (threshold security)
- q-invertible: $D_q(e) = (v, \alpha)$ s.t. $E(v, \alpha) = qe$.

Instances

- [CGS97]: variant of [ElGamal84], with [Pedersen91] setup
- [DJ00], [FPS00]: threshold setup for [Paillier99]

Encryption Function [ElGamal84, CGS97]

Setup [Ped91, CGS97]

- cyclic group $G = \langle g \rangle$
- shared SK z, PK $Z = g^z$

Encryption

- $E(v, \alpha) = (g^\alpha, g^v Z^\alpha)$

Homomorphism

- $(x_1, y_1) \otimes (x_2, y_2) \stackrel{\text{def}}{=} (x_1 x_2, y_1 y_2)$
  $\Rightarrow E(v_1, \alpha_1) \otimes E(v_2, \alpha_2) = E(v_1 + v_2, \alpha_1 + \alpha_2)$

Decryption

- $E(T) = (x, y) \rightarrow \frac{y}{x^z} = \frac{g^T Z^\alpha}{(g^\alpha)^z} = \frac{g^T (g^z)^\alpha}{(g^\alpha)^z} = g^T$
- $g^T \rightarrow T$, with cost $O(T)$

Σ-Proofs

q-One-Way-Group-Homomorphism (q-OWGH)

- $f : (G, \oplus) \rightarrow (H, \otimes)$
- homomorphic: $f(x \oplus x') = f(x) \otimes f(x')$
- q-invertible: $\hat{f}_q(y) = x_q$ s.t. $f(x_q) = y^q$

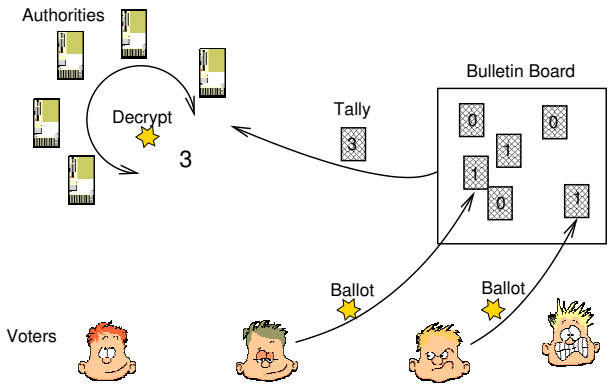
Σ-Proofs

- interactive proof of knowledge
- honest-verifier zero-knowledge
- non-interactive proof via Fiat-Shamir heuristics

Σ-Proofs for q-OWGH

- given y, prove knowledge of x with $f(x) = y$
- given $y_1, \ldots, y_\ell$, prove knowledge of x, i with $f(x) = y_i$

Voting Schemes based on Homomorphic Encryption



Protocol Sketch

1. authorities generate SK/PK, SK is shared
2. voters send encrypted vote, validity proof, signature onto BB
3. anybody can compute encrypted tally (due to homomorphism)
4. authorities jointly decrypt and prove tally

Validity Proof I

Given: encryption $e = E(v, \alpha)$, valid votes $\mathcal{V} = \{v_1, \ldots, v_L\}$

Prove: know α s.t.

$$e = E(v_1, \alpha)$$
or $e = E(v_2, \alpha)$
or $\ldots$
or $e = E(v_L, \alpha)$

Alternative: know α s.t.

$$E(0, \alpha) = e \oslash E(v_1, 0) \quad (y_1)$$
or $E(0, \alpha) = e \oslash E(v_2, 0) \quad (y_2)$
or $\ldots$
or $E(0, \alpha) = e \oslash E(v_L, 0) \quad (y_L)$

Technically: knows pre-image α of either

$$y_1 = e \oslash E(v_1, 0), \quad y_2 = e \oslash E(v_2, 0), \quad \ldots, \quad y_L = e \oslash E(v_L, 0),$$

w.r.t. to q-OWGH: $f : \alpha \mapsto E(0, \alpha)$.

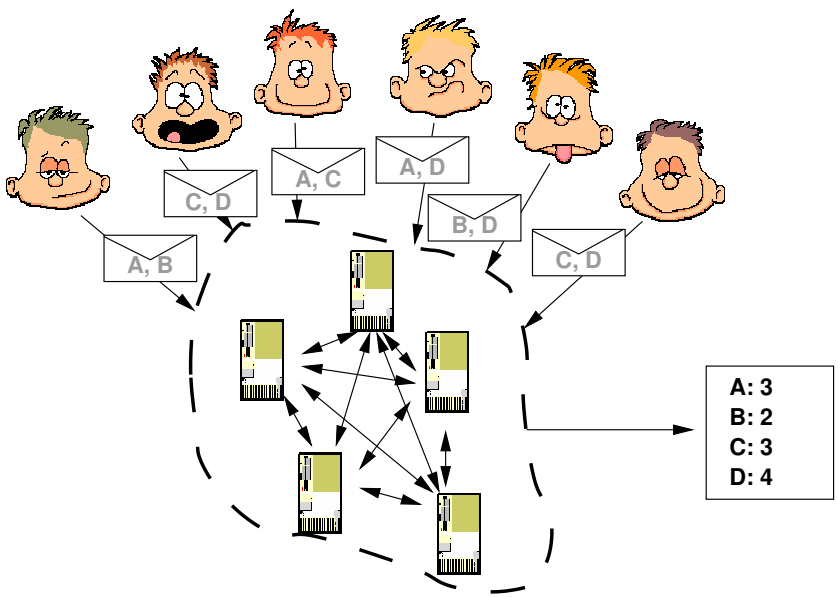
$\Rightarrow$ non-interactive validity proof.

Validity Proof II

Group homomorphism $f : \mathbb{Z}_q \to G^2, \alpha \mapsto E(0, \alpha)$

Prover		Verifier
$r_i \in_R \mathbb{Z}_q, t_i = E(0, r_i)$		
For $j = 1, \ldots, L, j \neq i$:		
$c_j, s_j \in_R \mathbb{Z}_q$		
$t_j = E(0, s_j) \oslash (e \oslash E(v_j, 0))^{c_j}$	$\xrightarrow{t_1, \ldots, t_L}$	
	$\xleftarrow{c}$	$c \in_R \mathbb{Z}_q$
$c_i = c - \sum_{j=1, j\neq i}^{L} c_j$		
$s_i = r_i + c_i \alpha$	$\xrightarrow{s_1, .., s_L, c_1, .., c_L}$	$c \stackrel{?}{=} \sum_{j=1}^{L} c_j$
		For $j = 1, \ldots, L$: $E(0, s_j) \stackrel{?}{=} t_j \otimes (e \oslash E(v_j, 0))^{c_j}$

K-out-of-L Voting Protocol



K-out-of-L Voting

K-out-of-L Vote

- L candidates/options, vote for K of them ($K < L$)
- ballot: | 0 | 1 | 1 | 0 | 0 | (L-vector, K ones)
- result: #votes per candidate

L parallel 0/1-Votes ...

- L-vector of mini-ballots: | v_1 | v_2 | v_3 | v_4 | v_5 |
- encrypt: | e_1 | e_2 | e_3 | e_4 | e_5 |
- validity proof for each i (i.e., e_i "$\in$" $\{0, 1\}$)

... Plus

- implicit vote $v_\Sigma = \sum v_i$ (should be K)
- implicit encrypted sum: $e_\Sigma = \bigotimes e_i$
- validity proof for $\mathcal{V} = \{K\}$

Efficiency

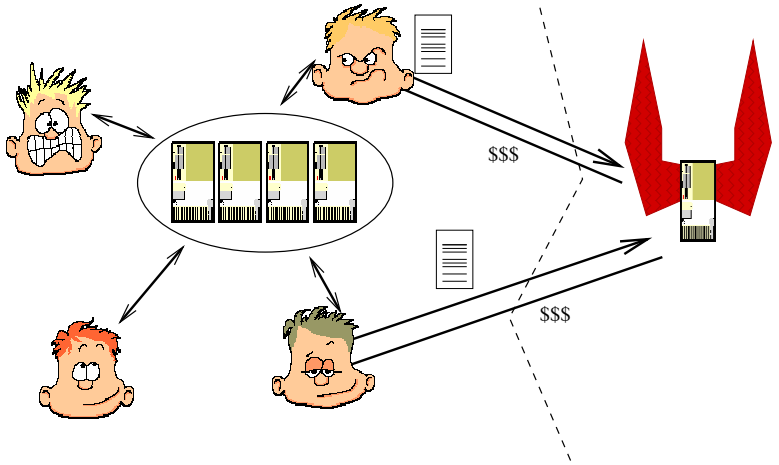
Proposed Scheme

- ballot size: $2L$ field elements
- validity proof size: $4L + 2$ field elements
- voter's signature: 2 field elements
- **total on bulletin board:** $6L + 4$ field elements

Cramer/Gennaro/Schoenmakers

- ballot size: 1 field element
- validity proof: $4L^{K-1}$ field elements
- voter's signature: 2 field elements
- **total on bulletin board:** $4L^{K-1} + 3$ field elements
- with ElGamal: exponential computation in L
- with ElGamal and Paillier: exponential communication in K

Vote-Buying



Receipt-Freeness

New Requirement

- secrecy: voter can keep vote secret
- receipt-freeness: voter must keep vote secret

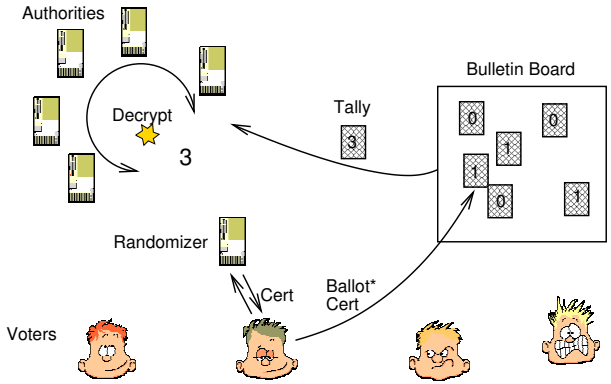
Remarks

- captures both vote-buying and coercion
- impossible for write-ins
- impossible in the standard model

New Assumptions

- voting booth
- untappable channels (many flavors)
- erasures (voter partially honest)
- Others?

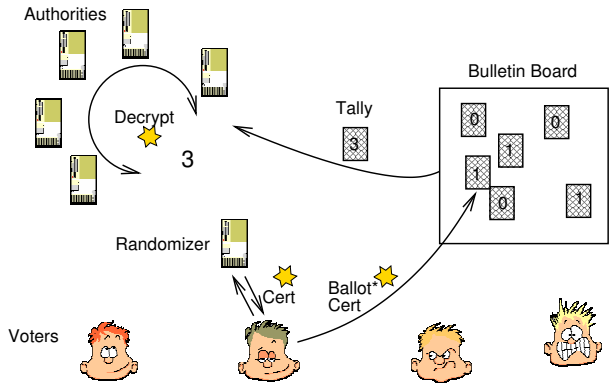
Receipt-Free Voting Scheme with Randomizers



Basic Ideas

- randomizer changes randomness in ballot
- voter does not know new randomness!
- randomizer should not learn vote
- randomizer is authority or hardware device

Receipt-Free Voting Scheme with Randomizers



Protocol Sketch

1. voter sends encrypted ballot $e = E(v, \alpha)$ to randomizer
2. randomizer sends $e^* = e \otimes E(0, \xi)$ to voter
3. randomizer gives randomization certificate (for e^*) to voter
4. proofs: randomization proof and validity proof

Receipt-Free Voting Scheme with Randomizers: Techniques

Randomization

- voter sends encrypted ballot $e = E(v, \alpha)$ to randomizer
- randomizer computes $e^* = e \otimes E(0, \xi)$.
- randomizer sends e^* and signature on e^* to voter

Randomization Proof

- randomizer proves to voter that $e^* \cong e$
- voter must not give away this proof!
- $\Rightarrow$ designated-verifier proof

Validity Proof

- randomizer and Voter together generate validity proof for e^*
- $\Rightarrow$ diverted proof

Randomization Proof

Given: randomizer knows ξ s.t. $e^* = e \otimes E(0, \xi)$

Idea: $\boxed{f : \mathbb{Z}_q \to G \times G, r \mapsto E(0, r),}$
prove knowledge of pre-image ζ s.t. $f(\zeta) = e^* \oslash e$

Problem: voter can give proof to vote-buyer

Designated-Verifier Proof

- randomizer proves knowledge of either ξ or voter's SK z_v
- OR-proof
  - Σ-proof of knowledge of pre-image of $e^* \oslash e$
  - Σ-proof of knowledge of SK corresponding PK Z_v
- non-interactive with Fiat-Shamir heuristics
- resulting proof is non-transferable

Randomization Proof II (With Schnorr Identification)

Randomizer		Voter
knows Z_v, α s.t. $E(0, \alpha) = e^* \oslash e$		knows $e, e^*, z_v, Z_v = g^{z_v}$
$r_1 \in_R \mathbb{Z}_q, t_1 = E(0, r_1)$		
$c_2, s_2 \in_R \mathbb{Z}_q, t_2 = g^{s_2} / Z_v^{c_2}$	$\xrightarrow{t_1, t_2}$	
	$\xleftarrow{c}$	$c \in_R \mathbb{Z}_q$
$c_1 = c - c_2 \pmod q$		
$s_1 = r_1 + c_1 \alpha \pmod q$	$\xrightarrow{s_1, s_2, c_1, c_2}$	$c_1 + c_2 \stackrel{?}{=} c \pmod q$
		$E(0, s_1) \stackrel{?}{=} t_1 \otimes (e^* \oslash e)^{c_1}$
		$g^{s_2} \stackrel{?}{=} t_2 \cdot Z_v^{c_2}$

NI: (s_1, s_2, c_1, c_2) s.t. $c_1 + c_2 = H\left(E(0, s_1) \oslash (e^* \oslash e)^{c_1} \parallel g^{s_2} / Z_v^{c_2}\right)$.

Diverted Validity Proof

Problem

- voter knows v, α s.t. $e = E(v, \alpha)$
- randomizer knows ξ s.t. $e^* = e \otimes E(0, \xi)$
- **who** proves knowledge of i, α such that $E(0, \alpha) = e^* \oslash E(v_i, 0)$?

Linear Σ-Proofs

- def: Σ-proof is linear when sum of accepting transcripts is accepting
- note: all used Σ-proofs are linear

Solution

- voter proves validity of e to randomizer
- randomizer generates random accepting transcript (using simulator)
- sum of transcripts is a random transcript for validity of e
- can be adjusted for e^*

Diverted Validity Proof II

Voter **Randomizer**

$t_1, \ldots, t_L \rightarrow$

$\leftarrow c$ $\quad c = \ldots$

$s_1, \ldots, s_L, c_1, \ldots, c_L \rightarrow$ $\quad c \stackrel{?}{=} \sum_{j=1}^{L} c_j$

$\forall j \in \{1, \ldots, L\}:$

$E(0, s_j) \stackrel{?}{=} t_j \otimes (e \oslash E(v_j, 0))^{c_j}$

Diversion:

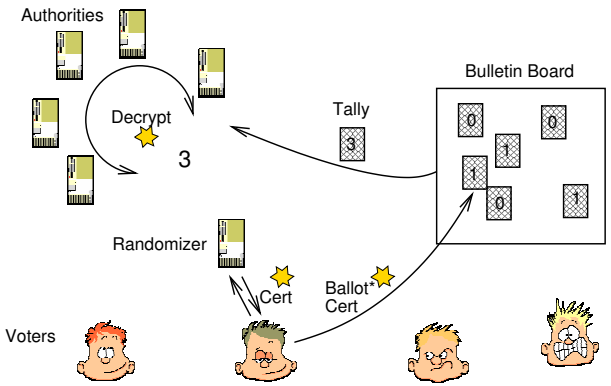
- $s_j \rightarrow s_j + s'_j$
- $c_j \rightarrow c_j + c'_j$

$\Rightarrow \quad t_j \rightarrow t_j \otimes E(c'_j v_j, s'_j) \oslash e^{c'_j}$

Adjust:

- $e \rightarrow e + E(0, \alpha) \quad \Rightarrow \quad s_j \rightarrow s_j + \alpha c_j$

Receipt-Free Voting Scheme with Randomizers



Protocol Sketch

1. voter sends encrypted ballot $e = E(v, \alpha)$ to randomizer
2. randomizer sends $e^* = e \otimes E(0, \xi)$ to voter
3. randomizer gives randomization certificate (for e^*) to voter
4. proofs: randomization proof and validity proof

K-out-of-L Voting

L parallel 0/1-Votes ...

- L-vector of mini-ballots:

v_1	v_2	v_3	v_4	v_5

- encrypt:

e_1	e_2	e_3	e_4	e_5

- randomizer:

e_1^*	e_2^*	e_3^*	e_4^*	e_5^*

- randomization proof for each i (i.e., $e_i^* \cong e_i$)
- diverted validity proof for each i (i.e., e_i^* "$\in$" $\{0, 1\}$)

... Plus

- implicit vote $v_\Sigma = \sum v_i$ (should be K)
- implicit encrypted sum: $e_\Sigma = \bigotimes e_i$
- implicit randomized sum: $e_\Sigma^* = \bigotimes e_i^*$
- diverted validity proof for $\mathcal{V} = \{K\}$

Efficiency

Proposed Scheme

- ballot size: $2L$ field elements
- diverted validity proof size: $4L + 2$ field elements
- voter's & randomizer's signature: 4 field elements
- **total on bulletin board:** $6L + 6$ field elements

Cramer/Gennaro/Schoenmakers (not receipt-free)

- ballot size: 1 field element
- validity proof: $4L^{K-1}$ field elements
- voter's signature: 2 field elements
- **total on bulletin board:** $4L^{K-1} + 3$ field elements
- with ElGamal: Exponential computation in L
- with ElGamal and Paillier: Exponential communication in K

Conclusions & Open Problems

Electronic Voting is . . .

- . . . more secure than paper-ballot voting
- . . . flexible enough in most cases
- . . . efficient enough for real world
- . . . appealing

Open Issues

- auditability
- legal system
- people