A Practical Voting Scheme with Receipts

Mirosław Kutyłowski, Marek Klonowski, Anna Lauks,
Filip Zagórski

Wrocław University of Technology

International Security Conference
Singapore, 23 IX 2005

Flaws of traditional voting

- high costs
- lack of verifiability
- vote selling

What is e-voting?

- e-voting $=$ voting with use of electronic devices

What is e-voting?

- e-voting $=$ voting with use of electronic devices
- e-voting $\neq$ internet voting

What is e-voting?

- e-voting $=$ voting with use of electronic devices
- e-voting $\neq$ internet voting
- voting machine
  - dedicated solutions (expensive)
  - voting at a voting station: computer + printer (use of existing public infrastructure)
  - voting at home: computer + other equipment (i. e. biometrics)

Existing internet voting solutions (?)

- easy to sell votes (all systems - just install special software)

Existing internet voting solutions (?)

- easy to sell votes (all systems - just install special software)
- problems with anonymity (i. e. Estonia)

Existing internet voting solutions (?)

- easy to sell votes (all systems - just install special software)
- problems with anonymity (i. e. Estonia)
- problems with verifiability

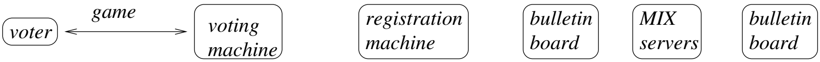
A typical voting system:

- Many systems consist in:

voter — *voting machine* — *registration machine* — *bulletin board* — *MIX servers* — *bulletin board*

Typical voting system:

- Voter and a voting machine play a game:

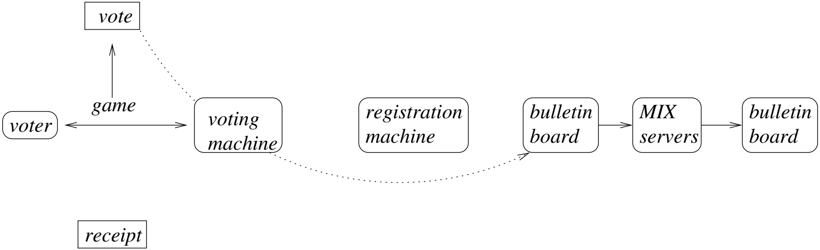


Typical voting system:

- As a result of that game a vote is being created

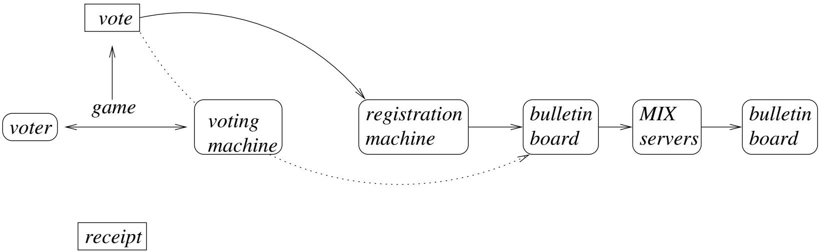
Typical voting system:

- In many systems ballots are being sent directly from a voting machine. A machine is connected to the internet, so i. e. there are possible hackers attacks (on voting machines)



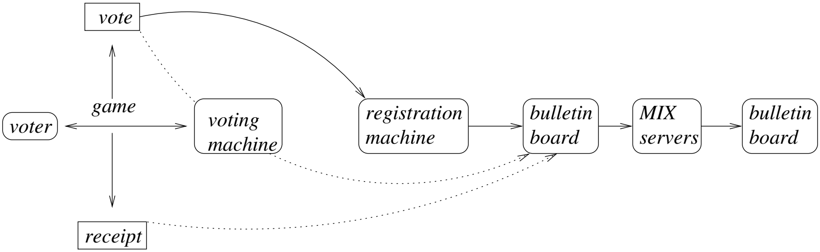
Typical voting system:

- When a voting machine only prepares a vote and another machine is responsible for vote casting then such solutions are much more secure



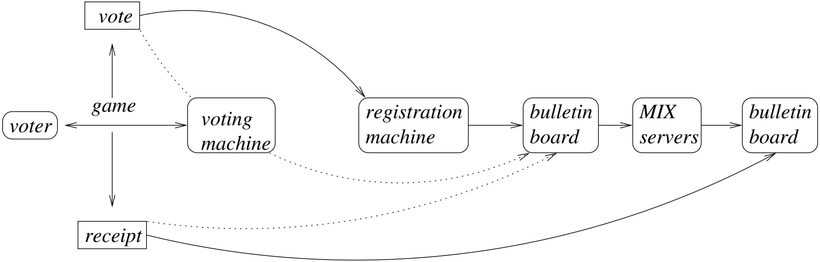
Typical voting system:

- In some systems, a voter gets a receipt (good solution), but in most cases he can verify only if it appears on first bulletin board. Two stage verification required - checking receipt and a proof of correct mixing



Typical voting system:

- When a receipt allows to verify final tally (last bulletin board) then we have double verification (existing solutions only: Chaum's, ours)



Previous solutions:

- D. Chaum’s [IEEE S&P’04] Visual Voting - expensive (special printers needed), high number of mixing servers (due to RPC)

Previous solutions:

- D. Chaum's [IEEE S&P'04] Visual Voting - expensive (special printers needed), high number of mixing servers (due to RPC)
- P. Ryan, D. Chaum, S. Schneider [ESORICS last week] - easy vote selling, problems with ballots distribution, problems with privacy (trusted authority knows voter's choice!)

Previous solutions:

- D. Chaum's [IEEE S&P'04] Visual Voting - expensive (special printers needed), high number of mixing servers (due to RPC)
- P. Ryan, D. Chaum, S. Schneider [ESORICS last week] - easy vote selling, problems with ballots distribution, problems with privacy (trusted authority knows voter's choice!)
- Zero Knowledge Proof based schemes (i. e. A. Neff [ACM'01], D. Reynolds [FEE last week]) unacceptable computational and communicational overhead

Previous solutions:

- D. Chaum's [IEEE S&P'04] Visual Voting - expensive (special printers needed), high number of mixing servers (due to RPC)
- P. Ryan, D. Chaum, S. Schneider [ESORICS last week] - easy vote selling, problems with ballots distribution, problems with privacy (trusted authority knows voter's choice!)
- Zero Knowledge Proof based schemes (i. e. A. Neff [ACM'01], D. Reynolds [FEE last week]) unacceptable computational and communicational overhead
- in many others - lack of verifiability

Verifiability

- **local verifiability** – the voter should be convinced that his vote was counted)

Verifiability

- **local verifiability** – the voter should be convinced that his vote was counted)
- **global verifiability** – mechanism that allows everyone to check whp that every vote was properly counted (not every voter will check his receipt)

$\Rightarrow$ receipts used

Anonymity

- nobody, in particular the authorities, cannot derive the choice of the voter

Anonymity

- nobody, in particular the authorities, cannot derive the choice of the voter
- voter should not be able to sell a vote and prove how he voted,

Election results

- a vote cannot be modified or erased from the tally,

Election results

- a vote cannot be modified or erased from the tally,
- an election commission cannot cast additional ballots,

Detecting cheaters

- misbehavior can be detected whp
- a proof of misbehaviour can be presented

Advantages

- standard, low cost equipment
- receipts for the voters printed on paper as bar codes (or 2D codes)
- the number of commissions decoding the ballots can be low

Advantages

- standard, low cost equipment
- receipts for the voters printed on paper as bar codes (or 2D codes)
- the number of commissions decoding the ballots can be low
- nobody in the system is assumed to be honest
- ballots prepared by voting machines (and not by external trusted parties)

Advantages

- standard, low cost equipment
- receipts for the voters printed on paper as bar codes (or 2D codes)
- the number of commissions decoding the ballots can be low
- nobody in the system is assumed to be honest
- ballots prepared by voting machines (and not by external trusted parties)
- full anonymity
- no way to sell a vote (thanks to special receipt)

Advantages

- standard, low cost equipment
- receipts for the voters printed on paper as bar codes (or 2D codes)
- the number of commissions decoding the ballots can be low
- nobody in the system is assumed to be honest
- ballots prepared by voting machines (and not by external trusted parties)
- full anonymity
- no way to sell a vote (thanks to special receipt)
- no way to cheat without being detected

Other features

- each ballot is linked with a voting machine
- a voting machine knows the preferences of a voter :(
  hard to avoid ...

Components of the system

Servers:

- at the polling station:
  - voting machines with printers, disconnected from the net
  - registration machines
- tallying machines (i. e. controlled by different parties)
- plus control servers (independent, watch dog organisations)

Authorities:

- local election committee
- tallying authorities

A voter in a voting booth - step 1/5

- a local commission committee checks identity of a voter
- a voter goes to a voting booth
- a machine prepares a *virtual ballot* at random,

A voter in a voting booth - step 1/5

- a local commission committee checks identity of a voter
- a voter goes to a voting booth
- a machine prepares a *virtual ballot* at random,
- a voter gets a printout of a *hash ballot* – a commitment to the virtual ballot
  - the machine cannot change the virtual ballot
  - the voter cannot recover the virtual ballot

A voter in the voting booth - step 2/5

- a voter sees depiction of the virtual ballot: 2 lists with the names of the candidates and a random ID
- green squares correspond to „cryptographic material”

3 Anna Lauks	1 Marek Klonowski
2 Miroslaw Kutylowski	276529020911123445 6
4 Filip Zagorski	3 Anna Lauks
1 Marek Klonowski	2 Miroslaw Kutylowski
276529020911123445 6	4 Filip Zagorski

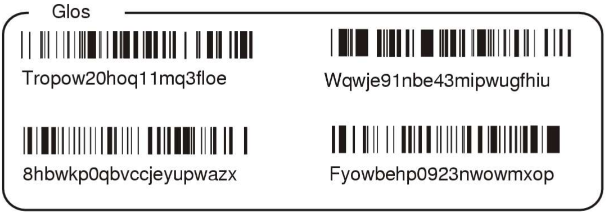
A voter in the voting booth - step 3/5

- a voter makes his choice by choosing a candidate and either the left or the right side:

3 Anna Lauks	1 Marek Klonowski
2 Miroslaw Kutylowski	2765290209111234456
4 Filip Zagorski	3 Anna Lauks
1 Marek Klonowski	2 Miroslaw Kutylowski
2765290209111234456	4 Filip Zagorski

A voter in the voting booth - step 4/5

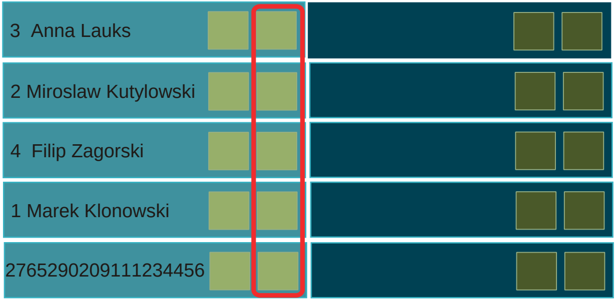
- a voter gets the split votes and spit identifiers (encrypted, randomly permuted):



- the values printed can be later compared with appropriate hashes from the hash ballot,

A voter in the voting booth - step 5/5

- preparation of a *control ballot*:



A voter in the voting booth - step 5/5

- a voter gets the control ballot: identifier, some keys, chosen onions:

Karta do weryfikacji

3 Anna Lauks	Reiopenl92nhdvampsddkffp
2 Miroslaw Kutylowski	Wqwje91nbe43mipwugfhiu
4 Filip Zagorski	Jobwjb821q10nxnimmajwax
1 Marek Klonowski	Pquioeiuwdf7218nzoiqpjq11
27652902091112344 56	Oqihiqhrt098hk2229kPiuo

Scanning of encoded votes

- (under supervision) a voter presents his voting ballot for scanning by a scanning machine
- the ballot gets stamped -
  - for possible investigations, and
  - for preventing to scan the same ballot twice

Vote counting

the parts of votes get separated!

Vote counting

the parts of votes get separated!

Scanned votes are:

- recoded (partial decoding + re-encryption)
- randomly mixed

by different Tallying Authorities

- anonymity is guaranteed

Vote counting

- The last counting commission decrypts (proof of correctness needed) and publishes:
  - identifiers
  - votes

Vote counting

- The last counting commission decrypts (proof of correctness needed) and publishes:
  - identifiers
  - votes
- A voter can check if his pair of identifiers is on the list.
- Everybody can
  - check if identifiers are paired,
  - check if votes are paired,
  - compute the result of elections.

RE-onions

- y_i - public key, x_i corresponding private key, $y_i = g^{x_i}$, g is generator of a group G with hard discrete logarithm problem. The order of G is a prime.
- ElGamal ciphertexts: for k_1 chosen uniformly at random $(\alpha, \beta) := (m \cdot (y_1 \cdot \ldots \cdot y_\lambda)^{k_1}, g^{k_1})$
- ith server gets: $(\alpha_i, \beta_i) := (m \cdot (y_i \cdot \ldots \cdot y_\lambda)^{k_i}, g^{k_i})$
  and outputs: $(\alpha_{i+1}, \beta_{i+1}) := (\alpha_i / \beta_i^{x_i} \cdot (y_{i+1} \cdot \ldots \cdot y_\lambda)^{r_i}, \beta_i g^{r_i})$
  r_i randomly chosen, $k_{i+1} = k_i + r_i$

Initialization of *voting machines*

- The public keys (y_i) of the tallying authorities are loaded to the voting machines

Initialization of *voting machines*

- The public keys (y_i) of the tallying authorities are loaded to the voting machines
- A private key K' generated: for signing votes and identifiers with a signature scheme sig'

Initialization of *voting machines*

- The public keys (y_i) of the tallying authorities are loaded to the voting machines
- A private key K' generated: for signing votes and identifiers with a signature scheme sig'
- A private key K generated: for creating seeds for constructing RE-onions (deterministic signature scheme)

Initialization of *voting machines*

- The public keys (y_i) of the tallying authorities are loaded to the voting machines
- A private key K' generated: for signing votes and identifiers with a signature scheme sig'
- A private key K generated: for creating seeds for constructing RE-onions (deterministic signature scheme)
- The corresponding public keys are delivered to the local registration machine and to the final tallying authority

A voter in a voting booth - step 1/5

- a machine prepares a *virtual ballot*, which consists in the following data:
  - r - ballot identifier, a random string
  - q - an auxiliary string used for constructing RE-onions
  - r_L, r_R - random strings chosen separately for each side
  - $$\begin{array}{ccc|ccc} B & B_1^L & B_2^L & I & I_1^R & I_2^R \\ I & I_1^L & I_2^L & Y & Y_1^R & Y_2^R \\ Y & Y_1^L & Y_2^L & B & B_1^R & B_2^R \end{array}$$

    where I, B, Y are labels

Content of the onions

- For constructing the RE-onion Z_i^X the voting machine creates signatures $sig_K(q,i,X,Z)$
- Using random bit generator $\mathcal{R}$, the machine computes: $k_1 = \mathcal{R}(sig_K(q,i,X,Z))$
- After full decoding of the onions we get, for $X = L,R$, $i = 1,2$:
  - $(B, r_X, ser_V, sig'_{K'}(B, r_X, i))$ from B_i^X
  - $(Y, r_X, ser_V, sig'_{K'}(Y, r_X, i))$ from Y_i^X
  - $(r, ser_V, sig'_{K'}(r, X, i))$ from I_i^X

A voter in the voting booth - step 2/5

- a voting machine creates and prints a *hash ballot*, which is a commitment to the *virtual ballot*, it contains:
  - r - ballot identifier
  - h_0 - the root of a Merkle tree of hashes. Its leaves are hashes of r, q, r_L, r_R, and of the RE-onions (without labels)

A voter in the voting booth - step 3/5

- Once the hash ballot is printed, a visualization of the virtual ballot appears on the screen
- The voter chooses a side c and an icon P of the party for which he votes,
- the *voting ballot* is created and printed. It consists in RE-onions of the party selected and the identifier.

Example:

$$\begin{matrix} P_1^c & I_2^c \\ I_1^c & P_2^c \end{matrix}$$

A voter in the voting booth - step 4/5

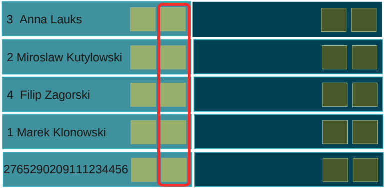
- a *control ballot* is created for the column chosen by the voter
- it contains
  - the onions from the side and column chosen
  - the data $q, sig_K(q, i, X, Z)$ that enable to reveal $k_1 = \mathcal{R}(sig_K(q, i, X, Z))$
- k_1 can be used to *open an onion* –decipher the ElGamal ciphertext– without the private keys

(trick borrowed from D.Chaum's paper)

Why a voting machine cannot cheat?

Each voting machine:

- a *virtual ballot* once created cannot be changed (thanks to the *hash ballot*,
- a voter can verify the contents of an unused half of vote – verification procedure for *control ballot*



Impossibility of manipulations

- encoded votes and identifiers are indistinguishable
- one cannot link together the corresponding onions

Impossibility of manipulations

- encoded votes and identifiers are indistinguishable
- one cannot link together the corresponding onions

Cheating attempt:

- if one part of an identifier or a vote is removed, then the cheating attempt will be discovered,

Impossibility of manipulations

- encoded votes and identifiers are indistinguishable
- one cannot link together the corresponding onions

Cheating attempt:

- if one part of an identifier or a vote is removed, then the cheating attempt will be discovered,
- if a pair of identifiers are removed, then a voter will see that his identifier is missing.

Impossibility of manipulations

- encoded votes and identifiers are indistinguishable
- one cannot link together the corresponding onions

Cheating attempt:

- if one part of an identifier or a vote is removed, then the cheating attempt will be discovered,
- if a pair of identifiers are removed, then a voter will see that his identifier is missing.
- vote removal - successful if one can find both pairs

  for 1000 voters probability of successful removal of a single vote is $\approx 1/8000$, two votes - 1.5e-8, 3 votes - 1.9e-12

Impossibility of manipulations

- encoded votes and identifiers are indistinguishable
- one cannot link together the corresponding onions

Cheating attempt:

- if one part of an identifier or a vote is removed, then the cheating attempt will be discovered,
- if a pair of identifiers are removed, then a voter will see that his identifier is missing.
- vote removal - successful if one can find both pairs

  for 1000 voters probability of successful removal of a single vote is $\approx 1/8000$, two votes - 1.5e-8, 3 votes - 1.9e-12
- an investigation procedure finds a tallying authority which made an attempt to cheat.

Why a voter cannot cheat?

- on verification cards there are only halves of votes/identifiers
  they cannot be used for casting an additional vote

Investigations

- partially reveal the permutation used
- reveal some re-encryption exponents
- zero knowledge proof of correctness of partial decoding

Implementation

test implementation prepared by students of Wrocław University of Technology

`http://e-voting.im.pwr.wroc.pl`

Still to be done

Apply schemes yielding short

- signatures,
- ciphertexts (of asymmetric schemes),
- and enabling fast re-encryption and fast partial decoding (for the China case)

Here, security level for signatures can be lower than in the case of digital signatures used for signing contracts ... that must remain secure for years.

Thanks for your attention

Details and contact: `http://e-voting.im.pwr.wroc.pl`