Automatic Verification of Remote Electronic Voting Protocols

Cătălin Hrițcu
Saarland University, Saarbrücken, Germany
Joint work with: Michael Backes and Matteo Maffei
Did you know that...

- ... in Germany, in the latest parliamentary elections **18.7%** of the votes were cast by post?
Did you know that ...

- ... in Germany, in the latest parliamentary elections 18.7% of the votes were cast by post?
- this is a form of remote voting
Remote voting (by post)

• More convenient than supervised voting
  ▸ This should increase voter participation
Remote voting (by post)

- More convenient than supervised voting
  - This should increase voter participation

- Voting by post raises many **security concerns**
  - An autograph signature does not authenticate the voter
  - An envelope does not guarantee secrecy or integrity
  - The post is not always a secure channel
  - Extremely easy to sell your vote
  - You can coerce voters to vote as you like
Remote voting (by post)

• More convenient than supervised voting
  ‣ This should increase voter participation

• Voting by post raises many **security concerns**
  ‣ An autograph signature does not authenticate the voter
  ‣ An envelope does not guarantee secrecy or integrity
  ‣ The post is not always a secure channel
  ‣ Extremely easy to sell your vote
  ‣ You can coerce voters to vote as you like

• Still, this has been used in Germany for 50+ years
Remote **electronic** voting

- Seems even cheaper and even more convenient
- Promises better security (than voting by post at least)
  - the security properties can be cryptographically enforced
Remote **electronic** voting

- Seems even cheaper and even more convenient
- Promises better security (than voting by post at least)
  - the security properties can be cryptographically enforced

Different security risks
- Easier to launch large-scale attacks and erase evidence
- Clients are the weakest link: e.g. remotely exploitable software flaws, viruses, Internet worms, trojans, lack of physical security, social engineering attacks, etc.
- Network also vulnerable: e.g. voter demographic-based DDOS, cache poisoning DNS attacks, etc.
desired properties

accuracy eligibility democracy
inalterability non-reusability fault tolerance
completeness correctness scalability
fairness

universal verifiability receipt-freeness

individual verifiability

coercion-resistance
vote-privacy

no forced-abstention attacks

robustness
desired properties

- eligibility
- non-reusability
- inalterability
- completeness
- correctness
- fairness
- universal verifiability
- individual verifiability
- receipt-freeness
- coercion-resistance
- scalability
- availability
- vote-privacy
- fault tolerance
- robustness
- scalability
- accuracy
- democracy
- fault tolerance
- robustness
- availability
- vote-privacy
- no forced-abstention attacks

• Careful formalization and automatic verification of these properties important before widespread adoption.
eligibility

inalterability  non-reusability

vote-privacy

no forced-abstention attacks

receipt-freeness

coercion-resistance

• Careful formalization and automatic verification of these properties important before widespread adoption
soundness

eligibility

inalterability  non-reusability

vote-privacy
no forced-abstention attacks
receipt-freeness
coercion-resistance

• Careful formalization and automatic verification of these properties important before widespread adoption
• Careful formalization and automatic verification of these properties important before widespread adoption
What we did

• General technique for **modeling** remote electronic voting protocols (in the applied pi-calculus) and automatically verifying their security

• New **formal definitions of**
  ‣ soundness - trace property
  ‣ coercion-resistance - observational equivalence
  ‣ both definitions amenable to automation (e.g. ProVerif)

• Automatically verified the security of the JCJ protocol
What we did

• General technique for modeling remote electronic voting protocols (in the applied pi-calculus) and automatically verifying their security

• New formal definitions of
  ‣ soundness - trace property
  ‣ coercion-resistance - observational equivalence
  ‣ both definitions amenable to automation (e.g. ProVerif)

• Automatically verified the security of the JCJ protocol

• For all details see [Backes, Hriţcu & Maffei, CSF 2008]
The Big Picture
Hi, I'm Alice

Soundness (eligibility, non-reusability, inalterability)
Hi, I'm Alice

eligible(Alice)
Soundness (eligibility, non-reusability, inalterability)

Hi, I'm Alice

eligible(Alice)
Soundness (eligibility, non-reusability, inalterability)

Hi, I'm Alice

vote(Alice, pink)

eligible(Alice)
Soundness (eligibility, non-reusability, inalterability)

Hi, I'm Alice

eligible(Alice)

vote(Alice, pink)
Soundness (eligibility, non-reusability, inalterability)

Hi, I’m Alice

vote(Alice, pink)

eligible(Alice)
Hi, I’m Alice

eligible(Alice)

vote(Alice, pink)
Hi, I'm Alice

eligible(Alice)
vote(Alice, pink)
Soundness (eligibility, non-reusability, inalterability)

Hi, I'm Alice

eligible(Alice)

vote(Alice, pink)

tally(pink)
Soundness (eligibility, non-reusability, inalterability)

Hi, I'm Alice

Trace: $t_1$ eligible(Alice) $t_2$ vote(Alice, pink) $t_3$ tally(pink)
Soundness (eligibility, non-reusability, inalterability)

Hi, I’m Alice

Trace: \( t_1 \) eligible(Alice) \( t_2 \) vote(Alice, pink) \( t_3 \) tally(pink)
Soundness (eligibility, non-reusability, inalterability)

Trace: $t_1$ $\text{eligible}(\text{Alice})$ $t_2$ $\text{vote}(\text{Alice, pink})$ $t_3$ $\text{tally}(\text{pink})$

and the trace $t_1$ $t_2$ $t_3$ is also sound (injective matching)
Vote-privacy

Voters

Alice
Bob
Charlie
Vote-privacy

Voters
Alice
Bob
Charlie

Results
pink party |
blue party ||
Vote-privacy

Voters
Alice
Bob
Charlie

Results
pink party
blue party

“Detailed” results
Alice ............ pink party
Bob .............. blue party
Charlie ........ blue party
Vote-privacy

Voters
Alice
Bob
Charlie

Results
pink party |
blue party ||

“Detailed” results
Alice ......... pink party
Bob .............. blue party
Charlie ......... blue party

The votes are kept private, and only the total results for each party are disclosed.
Definition of vote-privacy

[Delaune, Kremer & Ryan; CSF ’06]
Definition of vote-privacy

∀ [ S[ ] ] indistinguishable from [ S[ ] ]

[Delaune, Kremer & Ryan; CSF ’06]
Definition of vote-privacy

[Delaune, Kremer & Ryan; CSF ’06]

\[ S[ \begin{array}{c} \text{pink} \\ \text{blue} \end{array} ] \approx S[ \begin{array}{c} \text{pink} \\ \text{blue} \end{array} ] \]
Definition of vote-privacy

[Delaune, Kremer & Ryan; CSF ’06]
Definition of vote-privacy

[Delaune, Kremer & Ryan; CSF ’06]
Immunity to forced-abstention

\[ S[ ] \approx S[ ] \]
Receipt-freeness

- Cryptographic setting [Benaloh & Tuinstra; STOC '94]
Receipt-freeness

- Cryptographic setting [Benaloh & Tuinstra; STOC '94]

- We adapted definition by [Delaune, Kremer & Ryan; CSF '06] to remote voting
Coercion-resistance

- Cryptographic setting [Juels, Catalano & Jakobsson; WPES 2005]
Coercion-resistance

- Cryptographic setting [Juels, Catalano & Jakobsson; WPES 2005]
Coercion-resistance

- Cryptographic setting [Juels, Catalano & Jakobsson; WPES 2005]

\[ \Rightarrow \text{receipt-freeness (up to abstraction)} \]

- Proved: coercion-resistance \( \Rightarrow \) no forced-abstention \( \Rightarrow \) vote-privacy
# Definitions of coercion-resistance

<table>
<thead>
<tr>
<th></th>
<th>JCJ-WPES’05</th>
<th>DKR-CSF’06</th>
<th>DKR-TR’08</th>
<th>current</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>setting</strong></td>
<td>remote voting</td>
<td>supervised voting</td>
<td>supervised voting</td>
<td>remote voting</td>
</tr>
<tr>
<td><strong>automation</strong></td>
<td>no (crypto)</td>
<td>no (adaptive simulation)</td>
<td>no ($\forall C. P \approx Q$)</td>
<td>yes ($\approx$)</td>
</tr>
<tr>
<td><strong>vote-privacy</strong></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td><strong>no simulation attacks</strong></td>
<td>yes</td>
<td>n/a</td>
<td>n/a</td>
<td>yes</td>
</tr>
<tr>
<td><strong>no forced-abstention</strong></td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td><strong>no randomization attacks (?)</strong></td>
<td>yes (claimed not proved)</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td><strong>receipt-freeness</strong></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes (up to abstraction)</td>
</tr>
</tbody>
</table>
Analysis of JCJ

- first coercion-resistant protocol for remote voting [Juels, Catalano & Jakobsson; WPES ’05]
- forms the basis of many recent protocols (e.g. Civitas [Clarkson, Chong & Myers; S&P ’08])
- Analysis performed with ProVerif [Blanchet et. al.]
  - automatic protocol analyzer using Horn-clause resolution
  - we use our symbolic abstraction of zero-knowledge [Backes, Maffei & Unruh; S&P ’08]
  - analyzing observational equivalence required (re)writing the specification in the shape of a biprocess
  - verification of JCJ succeeds, which yields security guarantees for unbounded number of voters, sessions, etc.
Current and Future work

- Currently analyzing a model of Civitas
Current and Future work

• Currently analyzing a model of Civitas
• Defining and analyzing other properties
  ‣ Individual verifiability (trace property)
  ‣ Immunity to randomization attacks (privacy property)
Current and Future work

• Currently analyzing a model of Civitas
• Defining and analyzing other properties
  ‣ Individual verifiability (trace property)
  ‣ Immunity to randomization attacks (privacy property)
• Using type systems for trace properties
  ‣ e.g. type system for ZK [CCS ’08] [Fournet et. al., CSF ’07]
Current and Future work

• Currently analyzing a model of Civitas
• Defining and analyzing other properties
  ‣ Individual verifiability (trace property)
  ‣ Immunity to randomization attacks (privacy property)
• Using type systems for trace properties
  ‣ e.g. type system for ZK [CCS ’08] [Fournet et. al., CSF ’07]
• Different techniques for observational equivalence
  ‣ for instance using symbolic bisimulation [DKR, SecCo ’07]
Current and Future work

• Currently analyzing a model of Civitas

• Defining and analyzing other properties
  ‣ Individual verifiability (trace property)
  ‣ Immunity to randomization attacks (privacy property)

• Using type systems for trace properties
  ‣ e.g. type system for ZK [CCS ’08] [Fournet et. al., CSF ’07]

• Different techniques for observational equivalence
  ‣ for instance using symbolic bisimulation [DKR, SecCo ’07]

• More accurate protocol models
  ‣ The ultimate goal is to analyze implementations
Backup slides
Simplified JCJ protocol
Simplified JCJ protocol

Hi, I’m Alice

cred

(Private channel)

\{ cred, r_1 \}_{pk(kT)}
Hi, I'm Alice

cred

(/private channel)

{cred, r₁}pk(kT)

{cred, r₂}pk(kT), {pink}pk(kT), ZK
Hi, I’m Alice

cred

(Private channel)

\{ cred, r_1 \}_{pk(kT)}

\{ cred, r_2 \}_{pk(kT)}

\{ pink \}_{pk(kT)}

ZK
Hi, I’m Alice

cred

(pk(kT))

Simplified JCJ protocol

cred, r₁

(pk(kT))

cred, r₂

(pk(kT)), pink

(pk(kT)), ZK
Hi, I'm Alice

cred

(Private channel)

cred, r_1 \{ pk(kT) \}

cred, r_2 \{ pk(kT) \}

\{ pink \} \{ pk(kT) \}, ZK
Hi, I’m Alice

cred

(Private channel)

\{cred, r_1\}_{pk(kT)}

\{cred, r_2\}_{pk(kT)}, \{pink\}_{pk(kT)}, ZK

pink

Simplified JCJ protocol