Challenges and Advances in E-voting Systems
Technical and Socio-technical Aspects

Peter Y A Ryan
Lorenzo Strigini

Outline

• The problem.
• Voter-verifiability.
• Overview of “Prêt à Voter”.
• Resilience and socio-technical aspects
• Conclusions.
• Future work (in ReSIST)
The Problem

- Highly adversarial: system trying to cheat voters, voters trying to cheat the system, coercers trying to influence voters, voters trying to fool coercers etc.
- The Ancient Greeks experimented with primitive technological solutions to try to shift the trust from people (officials) to mechanical devices.
- In the US technological devices for voting have been used for over a century: e.g., lever machines since 1887, punch cards, optical scans, touch screen etc. prompted by high instance of fraud with paper ballots!
- All have problems, see “Steal this Vote” Andrew Gumbel.

“The Computer Ate my Vote”

- In the 2004 US presidential election, ~30% of the electorate used DRE, touch screen devices.
- Aside from the “thank you for your vote for Kerry, have a nice day” what assurance do they have that their vote will be accurately counted?
- What do you do if the vote recording and counting process is called into question?
- Need to trust the (proprietary) software.
- Voter Verifiable Paper Audit Trail (VVPAT) and “Mercuri method” have been proposed. But paper trails are not infallible either.
- Nedap machines in the Netherlands etc.
The challenge

- Digital voting technologies hold out promise of accessible and efficient democracy.
- Want high assurance that all votes are accurately recorded and counted-whilst maintaining ballot secrecy.
- The challenge is to reconcile these two conflicting requirements whilst minimising, ideally eliminating, dependence on the components (devices, tellers, software, hardware, officials etc.) of the scheme.
- Needs to be usable and sufficiently understandable to be widely trusted.
Technical Requirements

• Elections should be “free and fair”.
• Typical, key requirements:
  – (unconditional) integrity: count accurately reflects votes cast.
  – Ballot secrecy: the way a voter cast their vote should only be known to the voter.
  – Voter verifiability: the voter should be able to confirm that their vote is accurately included in the count and prove to a 3rd party if it is not (without having to revealing their vote).
  – Universal verifiability: anyone should be able to verify the count.
  – Availability: all eligible voters should be able to cast their vote without let or hindrance throughout the voting period.
  – Ease of use, public understanding and trust, cost effective, scalable etc. etc.....

Assumptions

• For the purposes of the talk we will make many sweeping assumptions, e.g.:
  – An accurate electoral register is maintained and available.
  – Mechanisms are in place to ensure that voters can be properly authenticated.
  – Crypto algorithms are sufficiently secure.
  – Etc.
Voter-verifiability in a nutshell

- Voters can confirm that their vote is accurately but not prove to a third party how they voted.
- Voters are provided with an encrypted “receipt”.
- Copies of the receipts are posted to a secure web bulletin board. Voters can verify that their (encrypted) receipt is correctly posted.
- A (universally) verifiable, anonymising tabulation is performed on the posted receipts.
- Checks (random audits) are performed at each stage to detect any attempt to corrupt the encryption and the decryption or the receipts.
- The guarantees of integrity are not dependent on correct behaviour of software, hardware, officials etc.
Prêt à Voter

• The key innovation of Prêt à Voter is to encode the vote by randomising the candidate order.
  – Voter experience simple and familiar.
  – Votes are not directly encrypted, just the frame of reference in which votes encoded. Hence:
    • The vote recording device doesn’t get to learn the vote.
    • No need for ZK proofs of correct encryption of votes—but onus of proof shifts to showing the well-formedness of the ballot forms.
    • Avoids subliminal, kleptographic and side channels.
• Prior work: Chaum, Benaloh, Neff,…

Typical Ballot Sheet

<table>
<thead>
<tr>
<th>Obelix</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Asterix</td>
<td></td>
</tr>
<tr>
<td>Idefix</td>
<td></td>
</tr>
<tr>
<td>Panoramix</td>
<td></td>
</tr>
<tr>
<td>Geriatrix</td>
<td></td>
</tr>
<tr>
<td>$rJ9*mn4R&amp;8</td>
<td></td>
</tr>
</tbody>
</table>
Voter marks their choice

<table>
<thead>
<tr>
<th>Obelix</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Asterix</td>
<td>✗</td>
</tr>
<tr>
<td>Idefix</td>
<td></td>
</tr>
<tr>
<td>Panoramix</td>
<td></td>
</tr>
<tr>
<td>Geriatrix</td>
<td></td>
</tr>
<tr>
<td>$rJ9*mn4R&amp;8</td>
<td></td>
</tr>
</tbody>
</table>

Voter’s Ballot Receipt

Cast - valid

$\text{rJ9}\ast\text{mn4R&8}$
449034729948
After the voting phase

– Once the election is closed, digital copies of the receipts are posted to the Web Bulletin Board (WBB).
– The voters can visit the WBB and confirm that their receipt appears correctly.
– Additionally, checks could be performed by independent entities between the (encrypted) paper audit trail and posted receipts.
– A verifiable, anonymising tabulation is performed with all intermediate stages posted to the WBB.
Auditing the tellers

Enhancements

- Vulnerability analysis.
- Randomising encryption and re-encryption mixes.
- Distributed generation of encrypted ballots.
- On-demand decryption and printing of ballot forms.
- (A variant of) Adida/Rivest off-line audit mechanism.
- Coercion-resistant remote variants (with Cornell).
- Crypto-free, scratch card version.
Resilience aspects

- cryptography-supported voter-verifiability promises much
  - *more* integrity and privacy than paper systems
  - run-time monitoring reduces need for special, heavily verified machinery
- but there is more to a voting system
  - error/attack detection does not make error/attack tolerance
  - ... recovery delegated to human part of system

ICT fault tolerance in the election system

Adversaries → Attacks

Ballots from voting booths → Ballot processing system → Vote count

Outputs from error checks → Triggers to external recovery/compensation mechanisms (e.g., recounts, prosecutions, re-run of election)
Effects of strong error detection

• election corruption is made more difficult
• but detected errors are expensive, so:
  – error recovery (automated and human) is important
  – better coverage may shift attackers’ preference, e.g. from attempting *undetected* vote corruption to simply sinking the election
  – good integrity and privacy; *availability* issues
    • e.g. DDoS attacks on bulleting boards?
    • increased requirements for ICT support to be robust/resilient

Wider socio-technical aspects

• attacker’s target might become simply the *reputation* of the election system
• implications cross the boundary between what can be designed (hardware, procedures) and political management
• so, a range of issues
  – from user-friendliness, HCI of voting machines
  – to choice of algorithms that public will be able to trust
  – to ensuring enough parties do perform the checks that anyone *may* perform
  – to ensuring *correct* perception of trustworthiness of each specific election
Conclusions

• we have presented: a technical problem, some solutions
  – Maximal transparency (consistent with ballot secrecy).
  – Accuracy independent of software, hardware, etc.
  – High assurance of detection of corruption.
  – Verify the election not the system!

• And open issues

Conclusions cont.

• E-voting is a ReSIST problem par excellence..
  – large distributed system, complex dependability requirements, evolving threats
  – “must work well the first time around”, every time - implying need for resilience
  – ICT entwined with users and their reactions
Future work

• Further enhancements (simplifications!?)
• Further analysis of the resilience of the system
• Investigate recovery mechanisms and strategies
• Investigate socio-technical aspects
• Investigate public understanding and trust
• Basis for a ReSIST case study