Online verifiable elections with Helios

Stéphane Glondu

LORIA (Inria, CNRS, Université de Lorraine)

Libre Software Meeting, Geneva
July 10, 2012
Outline

Overview

Building a verifiable voting system

Security considerations

Conclusion
Electronic voting

Elections are a security-sensitive process which is the cornerstone of modern democracy.

Electronic voting promises

- convenient, efficient and secure facility for recording and tallying votes
- for a variety of types of elections:
  - from small committees or on-line communities...
  - to public office (political) elections

Already used e.g. in Switzerland, France, USA...
Two main families of e-voting

Voting machines
- voters have to attend a polling station
- external authentication system (e.g. ID card)

Internet voting
- voters vote from home
- using their own computer
A trust issue

In many systems in use today...

- the whole procedure is secret
  - secret specification
  - closed source software and/or proprietary hardware
  - audit restricted to (some) (supposedly honest) experts
  - ...

i.e. **blind trust**

- open source software/hardware is not enough!
  - the result should be verifiable independently
  - software should not matter

- people claim it’s needed for security
  (security through obscurity)
A trust issue

1. Vendor
2. Voting Machine
3. Polling Location
4. Alice
5. Ballot Box Collection
6. Results
Properties

- **Fairness**: the result corresponds to the votes
- **Eligibility**: only legitimate voters can vote, and only once
- **Individual verifiability**: a voter can verify that her vote was really counted
- **Universal verifiability**: everyone can verify that the published outcome really is the sum of all votes
Public ballots

Alice: Obama
Bob: McCain
Carol: Obama

Tally
Obama: 2
McCain: 1

7/26
Properties

- **Fairness**: the result corresponds to the votes
- **Eligibility**: only legitimate voters can vote, and only once
- **Individual verifiability**: a voter can verify that her vote was really counted
- **Universal verifiability**: everyone can verify that the published outcome really is the sum of all votes
- **Privacy**: the fact that someone voted in a particular way is not revealed to anyone else
Anonymized public ballots

Alice verifies her vote

Everyone verifies the tally

Obama...2
McCa in...1
Encrypted public ballots

Alice:

Bob:

Carol:

Alice verifies her vote

Everyone verifies the tally

Obama...2
McCain...1

Alice verifies her vote

Ev­ery one verifies the tally

Alice

10/26
Democratizing audits

- each voter is responsible for checking her receipt
- anyone (individual or organization) can audit the tally and verify the list of cast ballots

Verifiable elections
End-to-end verification

Polling Location

Voting Machine

Ballot Box / Bulletin Board

Vendor

Alice

Receipt

1

2

Results

12/26
Public key encryption

Public key: $\text{pk}(A)$
Encryption: $\{m\}_{\text{pk}(A)}$

Encryption with the public key and decryption with the private key.
Randomized encryption

"Obama" $\xrightarrow{\text{Enc}_{pk}}$ 8b5637

"McCain" $\xrightarrow{\text{Enc}_{pk}}$ c5de34

"Obama" $\xrightarrow{\text{Enc}_{pk}}$ a4b395
Homomorphic encryption

- allows computations on encrypted messages without decrypting them

\[
\{m_1\}_{pk} \times \{m_2\}_{pk} = \{m_1 + m_2\}_{pk}
\]

- for example: use the property

\[
g^{m_1} \times g^{m_2} = g^{m_1+m_2}
\]
A concrete voting system

Phase 1: voting

<table>
<thead>
<tr>
<th>Bulletin Board</th>
<th>( {v_A}_{\text{pk}(S)} )</th>
<th>( v_A = 0 ) or ( 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>( {v_B}_{\text{pk}(S)} )</td>
<td>( v_B = 0 ) or ( 1 )</td>
</tr>
</tbody>
</table>
| Bob           | ...                             | ...

Phase 2: tallying using homomorphic encryption

\[
\prod_{i=1}^{n} \{v_i\}_{\text{pk}(S)} = \{\sum_{i=1}^{n} v_i\}_{\text{pk}(S)}
\]

Phase 3: decrypt the final result

*Only the final result needs to be decrypted!*

\( \text{pk}(S) \): public key of the election
Cheating voters

- a malicious voter can cheat:
  \[
  \{v_A + v_B + v_C + v_D + \cdots\}_{pk(S)} \quad \text{Result:}
  \{v_A + v_B + v_C + 100 + \cdots\}_{pk(S)} \quad \text{Result:}
  \{v_A + v_B + v_C + v_D + \cdots\}_{pk(S)}
  \]

- hence, each voter must prove that her vote is 0 or 1 without revealing it
- it is possible with zero-knowledge proofs
Cheating authorities

- malicious election authorities can cheat:

  \[ \{v_A + v_B + v_C + v_D + \cdots\}_{pk(S)} \]

<table>
<thead>
<tr>
<th>Bulletin Board</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>({v_A}_{pk(S)})</td>
</tr>
<tr>
<td>Bob</td>
<td>({v_B}_{pk(S)})</td>
</tr>
<tr>
<td>Chris</td>
<td>({v_C}_{pk(S)})</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- can be mitigated by use of threshold decryption
Threshold decryption

\[ \text{Dec}_{\text{sk}_1}(8b5637) \rightarrow b739cb \]
\[ \text{Dec}_{\text{sk}_2}(8b5637) \rightarrow 261ad7 \]
\[ \text{Dec}_{\text{sk}_3}(8b5637) \rightarrow 7231bc \]
\[ \text{Dec}_{\text{sk}_4}(8b5637) \rightarrow 8239ba \]

"Obama"
Helios

http://vote.heliosvoting.org/

- developed by B. Adida et al
- used for:
  - university elections (Louvain, Princeton)
  - IACR board election

- libre version:
  - https://github.com/{benadida,glondu}/helios-server
  - better thought as an open specification for electronic voting
  - actively studied by the scientific community
Disclaimer

The security of Helios relies on the assumption that the voter’s computer can be trusted.

- Not suitable for political elections
  A corrupted machine may:
  - leak the choice of the voter
  - vote for a different candidate

The same applies to systems currently deployed for political elections!
- concrete attack by Laurent Grégoire on the system used by the French abroad

- Suitable for medium issue elections:
  - professional elections
  - scientific councils, students representatives, etc.

- To be compared with remote voting:
  - better guarantees than vote by mail
Guaranteed properties

- **Fairness**: the result corresponds to the votes
- **Eligibility (partial)**: voters vote only once
- **Individual verifiability**: a voter can verify that her vote was really counted
- **Universal verifiability**: everyone can verify that the published outcome really is the sum of all votes
- **Privacy**: the fact that someone voted in a particular way is not revealed to anyone else
Mitigation for questionable properties

- LiveCD with minimal software and certificates
  - and documentation on how to build it by oneself
- Voter-initiated audit before casting
  - using third-party software and/or hardware
  - possibly home-made
- Honeypots
Room for improvement

- resistance to ballot stuffing
- coercion resistance, ticket freeness
- everlasting privacy
- mixnets
- elliptic curve cryptography
Electronic voting is possible without *blind* trust.

... but it is not ready to replace “traditional” voting.
Questions?

Contact:

- helios-voting@googlegroups.com
- steph@glondu.net

Slides under CC-BY-SA 3.0. Acknowledgements:

- Ben Adida, http://ben.adida.net/
- Véronique Cortier