PRIVACY
in electronic voting

Michael Clarkson
Cornell University

Workshop on Foundations of Security and Privacy
July 15, 2010
Secret Ballot
Florida 2000:
Bush v. Gore
Security FAIL
Analysis of an electronic voting system. [Kohno et al. 2003, 2004]

- DRE trusts smartcards
- Hardcoded keys and initialization vectors
- Weak message integrity
- Cryptographically insecure random number generator
- ...
California top-to-bottom reviews
[Bishop, Wagner, et al. 2007]

• “Virtually every important software security mechanism is vulnerable to circumvention.”

• “An attacker could subvert a single polling place device...then reprogram every polling place device in the county.”

• “We could not find a single instance of correctly used cryptography that successfully accomplished the security purposes for which it was apparently intended.”
Why is this so hard?
PRIVACY

INTEGRITY
Cryptography
Can cryptography be defended?
Low-tech crypto?
Simple Voting Protocol

1. $V \rightarrow BB$: $\text{sign}(\text{enc}(\text{vote}); k_V)$
2. Talliers: check signatures
3. Talliers: decrypt votes, tally
Simple Voting Protocol

1. V → BB: sign(enc(vote); kV)
2. Talliers: check signatures
3. Talliers: decrypt votes, tally

How to build secure, scalable BB?
PRIVACY via cryptography

- Blind signatures
- Mix networks
- Homomorphic encryption
PRIVACY via cryptography

- Blind signatures
- Mix networks
- Homomorphic encryption

Why these three? What others?
When is Vote Anonymized?

Before submission  After submission
Blind Signatures

[Chaum 1983]
unblind(sign(blind(m); k)) = sign(m; k)
V → BB: \text{sign}(\text{enc}(\text{vote}); k_V)
V → BB: sign(enc(vote); k_A)
Simple Blind Signature Election Protocol

1. $V \rightarrow \text{Auth}: V, \text{sign}(\text{blind}(\text{enc}(\text{vote})); k_V)$
Simple Blind Signature Election Protocol

1. $V \rightarrow Auth: V, \text{sign}(\text{blind}(\text{enc}(\text{vote})); k_V)$

2. $Auth \rightarrow V: \text{sign}(\text{blind}(\text{enc}(\text{vote})); k_A)$
Simple Blind Signature Election Protocol

1. $V \rightarrow Auth$: $V, \text{sign(}\text{blind(}\text{enc(vote)})\text{)}; k_V$

2. $Auth \rightarrow V$: $\text{sign(}\text{blind(}\text{enc(vote)})\text{)}; k_A$

3. $V \rightarrow BB [\text{anon.}]: \text{sign(}\text{enc(vote)}; k_A)$
Simple Blind Signature Election Protocol

1. $V \rightarrow \text{Auth}: V, \text{sign}(\text{blind}(\text{enc}(\text{vote})); k_V)$
2. $\text{Auth} \rightarrow V: \text{sign}(\text{blind}(\text{enc}(\text{vote})); k_A)$
3. $V \rightarrow \text{BB [anon.]}: \text{sign}(\text{enc}(\text{vote}); k_A)$
4. Talliers: check signatures, decrypt votes, tally
Blind Signature
Voting Protocols


How to achieve high integrity?
When is Vote Anonymized?

- Before submission
- After submission
- Before tallying
Mix Networks

[Chaum 1981]
Decryption Mix

\[
\text{enc}( \text{enc}( \text{enc}( m ; K_3 ) ; K_2 ) ; K_1 )
\]
Reencryoption Mix

\[ \text{enc}(m; K) \quad \text{reenc}(m; K) \quad \text{reenc}(m; K) \]

[Park et al. 1994]
Simple Mix Network Election Protocol

1. $V \rightarrow BB$: $\text{sign(\text{enc(vote)}; k_v)}$

2. Talliers: check signatures

3. Mixers: remove signatures, mix votes

4. Talliers: decrypt votes, tally
Mix Network
Election Protocols

When is Vote Anonymized?

- Before submission
- After submission
  - Before tallying
  - During tallying
Homomorphic Encryption

[Rivest, Adleman, Dertouzos 1978]

Fully homomorphic?
\[ \text{enc}(v) \times \text{enc}(v') = \text{enc}(v+v') \]
1. $V \rightarrow BB: \text{sign(enc(vote)); } k_V$

2. Talliers:
   a. check signatures
   b. compute $T = \prod_i \text{enc(vote}_i)$, which is $\text{enc}(\sum_i \text{vote}_i)$
   c. compute $\text{dec}(T)$

Simple Homomorphic Encryption Election Protocol
Homomorphic Encryption
Election Protocols

Formal Definitions of Privacy Integrity?
PRIVACY

- Vote privacy
- Receipt freeness
- Coercion resistance
Vote Privacy

Nothing about map from voters to votes revealed

(assuming everyone is honest)
Vote Privacy

Y Y Y Y Y N N N N N
Vote Privacy

Y Y Y Y Y N N N N N N N N

≈

N N N N Y Y Y Y Y Y Y Y
Vote Privacy

\[ V(x) \approx V(y) \]
Vote Privacy

\[ V(y) \cdot V(x) \cong \cdot V(y) \cdot V(x) \]
Vote Privacy
Formal Definitions

Computational:  Cohen (Benaloh) & Fisher 1985

Symbolic:  Delaune, Kremer & Ryan 2006
Receipt Freeness

Voters do not obtain information (a receipt) that proves how they voted.
Receipt Freeness

\[ V(x) \]

\[ x \]

\[ \text{Adv} \]
Receipt Freeness

\[ V(x) \rightarrow^x \text{Adv} \]  
\[ V(y) \rightarrow^x \text{Adv} \]
Receipt Freeness

\[ V(x) \xrightarrow{x} \text{Adv} \approx V(y) \xrightarrow{x} \text{Adv} \]
Receipt Freeness

\[ V(y) \land V(x) \Downarrow^x \text{Adv} \approx V(y) \land V(x) \Downarrow^x \text{Adv} \]
Receipt Freeness Requirements

- Private/untappable channel from authorities to voter [Benaloh 1994, Sako & Killian 1995]
- Trusted voter hardware [Lee et al. 2004]

What is minimal requirement?
Receipt Freeness
Formal Definitions

**Computational:** Benaloh & Tuinstra 1994 (there called uncoercible), Okamoto 1997

**Symbolic:** Delaune, Kremer & Ryan 2006, Jonker & de Vink 2006, Backes et al. 2008

**Logical:** Jonker & Pieters 2006
Fails to defend against:

- Randomization attacks
- Forced abstention attacks
- Simulation attacks

[Schoenmakers 2000, Juels et al. 2005]
Coercion Resistance

Voters cannot prove how they voted, even by fully cooperating with the adversary.
Coercion Resistance

\[ V(x) \]

\[ x \]

\[ \text{Adv} \]
Coercion Resistance

\[ V(x) \quad \text{Adv} \quad x \quad \text{Adv} \quad V(y) \]

\[ \uparrow x \quad \downarrow \]

\[ \text{Adv} \quad \text{Adv} \]
Coercion Resistance

\[ V(x) \xrightarrow{x} \text{Adv} \approx V(y) \xrightarrow{x} \text{Adv} \]
Coercion Resistance

\[ V(y), V(x) \approx \]

\[ \text{Adv} \]

\[ \approx \]

\[ \text{Adv} \]
Coercion Resistance
Formal Definitions

**Computational:** Juels et al. 2005, Moran & Naor 2006 (there called *receipt freeness*)

**Symbolic:** Delaune, Kremer & Ryan 2006, Backes et al. 2008
Coercion resistance

Receipt freeness

Vote privacy

[Delaune, Kremer & Ryan 2006]
Civitas
Secure Remote Voting

[Clarkson, Chong & Myers 2008]
based on [Juels, Catalano & Jakobsson 2005]
1. V → BB: sign(enc(vote); k_V)
2. Talliers: check signatures
3. Mixers: remove signatures, mix votes
4. Talliers: decrypt votes, tally
Voter Credentials

- Registrar → V: cred
- Registrar → BB: enc(cred)  [electoral roll]
- V → BB: enc(cred), enc(vote)
J CJ
Voter Credentials

- Registrar $\rightarrow$ V: cred
- Registrar $\rightarrow$ BB: $\text{enc(cred)}$ [electoral roll]
- V $\rightarrow$ BB: $\text{enc(cred)}, \text{enc(vote)}$
Voter Credentials

• Registrar $\rightarrow$ V [untap.]: $\text{cred, zkpf}_1$

• Registrar $\rightarrow$ BB: $\text{enc(cred)}$ [electoral roll]

• V $\rightarrow$ BB [anon.]: $\text{enc(cred)}, \text{enc(vote)}, \text{zkpf}_2$
JCJ
Tallying Protocol

Talliers:
Tallying Protocol

Talliers:

1. Retrieve votes from BB, check proofs
Tallying Protocol

Talliers:

1. Retrieve votes from BB, check proofs
2. Eliminate unauthorized credentials (requires mixes, zkpfss)
Tallying Protocol

Talliers:

1. Retrieve votes from BB, check proofs
2. Eliminate unauthorized credentials (requires mixes, zkpfds)
3. Decrypt votes, tally
Removing Unauthorized Credentials

Electoral roll, mixed

Submitted votes, mixed

PETs

\[ \text{enc(cred)}, \text{enc(vote)} \]
JCJ Credentials

• Verifiable
• Unsalable
• Anonymous
• Unforgeable
Coercion resistant: voters use fake (unauthorized) credentials to comply with coercer
Civitas Architecture

JCJ: single trusted registrar
Civitas: distributed trust
...improved privacy and integrity

registration teller

voter client

ballot box

tabulation teller

bulletin board
Civitas Architecture

JCJ: single trusted registrar
Civitas: distributed trust
...improved privacy and integrity

registration

teller

JCJ: no ballot boxes
Civitas: distributed storage
...improved availability

voter

client

ballot box

tabulation teller

bulletin board

tabulation teller

tabulation teller
Civitas
Architecture

**JCJ:** single trusted registrar
Civitas: distributed trust
...improved privacy and integrity

**JCJ:** no ballot boxes
Civitas: distributed storage
...improved availability

**JCJ:** $O(V^2)$
Civitas: $O(B^2)$, $B \ll V$
...improved scalability
Civitas Architecture

JCJ: single trusted registrar
Civitas: distributed trust
...improved privacy and integrity

JCJ: no ballot boxes
Civitas: distributed storage
...improved availability

JCJ: $O(V^2)$
Civitas: $O(B^2)$, $B \ll V$
...improved scalability

Civitas: concrete implementation, 21K LoC
Civitas

Security:

• Coercion resistance & universal verifiability
• Distributed trust

Assurance:

• Security proofs & security-typed implementation

Also: Ranked voting
Civitas

- High integrity voter client?
- Eliminate untappable channel in registration?
- Credential management?
- Application-level DoS?
www.cs.cornell.edu/projects/civitas

or google “civitas voting”
PRIVACY in electronic voting

- History
- Cryptographic techniques
- Formal definitions
- Civitas
PRIVACY
in electronic voting

Michael Clarkson
Cornell University

Workshop on Foundations of Security and Privacy
July 15, 2010