A Brief History of Secure Voting

Ben Adida
ben@eecs.harvard.edu
27 September 2006
2 Weeks Ago: Princeton Report

- Diebold touch-screen runs executable code loaded from memory card
- All audit logs modified to be consistent
- Can spread virally by memory card.

[VOTE STEALING CONTROL PANEL]

Select the race and candidate to fix:

- President of the United States

<table>
<thead>
<tr>
<th>Candidate Name</th>
<th>Votes So Far</th>
</tr>
</thead>
<tbody>
<tr>
<td>George Washington</td>
<td>9 (90%)</td>
</tr>
<tr>
<td>Benedict Arnold</td>
<td>1 (10%)</td>
</tr>
</tbody>
</table>

Set the final outcome: Percent for "Benedict Arnold" 75%

[OK] [Cancel]

[FHF2006]
Why is Voting so Hard?
1892 - Australian Ballot

Democratic:

FOR MAYOR, AUGUST LEUZ, JR.
CORNER BURLINGTON AND JOHNSON STREETS.

FOR TREASURER, GEORGE W. KOONTZ
NO. 620 EAST BURLINGTON STREET.

FOR CITY SOLICITOR, FRANK J. HORAK
NO. 120 DODGE STREET.

FOR ASSESSOR, F. A. HEINSIUS
NO. 948 EAST MARKET STREET.

FOR TRUSTEE, JOHN U. MILLER
EAST MARKET STREET.

Republican:

FOR MAYOR, CHAS. LEWIS
NO. 227 NORTH CLINTON STREET.

FOR TREASURER, L. H. FULLER
NO. 472 SOUTH DUBUQUE STREET.

FOR SOLICITOR, H. W. LATHROP
NO. 518 IOWA AVENUE.

FOR ASSESSOR, J. C. LEASURE
COR. VAN BUREN ST. AND IOWA AVENUE.

Majorities:

221

848

101

198

http://www.cs.uiowa.edu/~jones/voting/pictures/
Secret Ballot vs. Verifiability

**Protection from Undue Influence:**
Alice cannot prove how she voted.

**Verifiability:**
Alice gains confidence that her vote was properly recorded.
The Next Harvard Pres!

SOURCES: HARVARD WANTS CONDOLEEZZA RICE OR BILL CLINTON FOR NEXT PRES...

The Ballot Handoff

Alice the Voter

Clinton
The Ballot Handoff

Clinton

Alice the Voter
The Ballot Handoff

Alice the Voter
The Ballot Handoff

Alice the Voter
The Ballot Handoff

Alice the Voter

Clinton

Rice
Chain of Custody
Chain of Custody

Vendor

/*
 * source
 * code
 */
if (...
Chain of Custody

Vendor

Voting Machine

/*
 * source
 * code
 */

if (...
Chain of Custody

Polling Location → Voting Machine → Vendor

Vendor

/*
 * source
 * code
 */

if (...
Chain of Custody

1. Vendor
   /*
    * source
    * code
    */
   if (...)

2. Voting Machine

3. Polling Location

4. Alice
   (Voting Machine)
   (Polling Location)
   (Vendor)
   (Ballots)

Polling Location

Vendor
Chain of Custody

1. Vendor

2. Voting Machine

3. Polling Location

4. Alice
Chain of Custody

1. Vendor
   /*
    * source
    * code
    */
   if (...)

2. Voting Machine

3. Polling Location

4. Alice

5. Ballot Box Collection

Polling Location

Ballot Box Collection
Chain of Custody

Vendor

/*
 * source
 * code
 */
if (...

Polling Location

Alice

Ballot Box Collection

Results

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

Vendor
/*
 * source
 * code
 */
if (...)
Polling Location

Alice

Voting Machine

Ballot Box Collection

Results
The Cost of Secrecy
The Cost of Secrecy

Scavenged ballot box lids haunt S.F. elections

Erin McCormick, Chronicle Staff Writer

Monday, January 7, 2002
The Cost of Secrecy

Scavenged ballot box lids haunt S.F. elections

Erin McC: Helicopter Crash Delays Afghan Vote Count

Helicopter Sent to Pick Up Afghan Ballots in Remote Province Crash-Lands, Delaying Vote Count
The Cost of Secrecy

Scavenged ballot box lids haunt S.F. elections

Erin McClelland

Monday, 1

Helicopter Crash Delays Afghan Vote Count

Absentee ballots 'lost' in Florida

October 28, 2004 09:28 IST

Nearly 58,000 absentee ballots for the US presidential election may never have reached Florida's Broward County voters, who had requested them more than two weeks ago, election officials said.
The Cost of Secrecy

Scavenged ballot box lids haunt S.F. elections

Erin McCutcheon

Helicopter Crash Delays Afghan Vote Count

Absentee ballots 'lost' in Florida

October 28, 2004 09:28 IST

Nearly 58,000 absentee ballots for the US presidential election may never have reached Florida's Broward County, election officials said.

Mexico Presidential Election Ballots Found in Dump

RAW STORY

Published: Thursday July 6, 2006
Maybe the Secret Ballot isn’t so Important?

“Secret ballots and transparency in government are mutually exclusive concepts.”

Lynn Landes - Nov. 2005

http://votebymailproject.org

Thursday, July 27, 2006

San Diego task force recommends by mail voting

The City of San Diego is considering all mail balloting for special city elections, the Union Tribune reports.
Secret Ballot implemented in Chile in 1958.

“the secrecy of the ballot [...] has first-order implications for resource allocation, political outcomes, and social efficiency.”

[BalandRobinson 2004]
Cryptographic Voting achieves verifiability and secrecy without depending on equipment correctness.
End-to-End Verification
End-to-End Verification

Voting Machine

Polling Location

Vendor

/*
 * source
 * code
 */

if (...
End-to-End Verification

Voting Machine

Polling Location

Ballot Box / Bulletin Board

Vendor

Alice

/*
 * source
 * code
 */

if (...
End-to-End Verification

Voting Machine

Polling Location

Ballot Box / Bulletin Board

Vendor

Results

Alice
End-to-End Verification

Polling Location

Voting Machine

Ballot Box / Bulletin Board

Vendor

Receipt

Alice

Results

/*
 * source
 * code
 */

if (....)
End-to-End Verification

[SR8C1]
An Encrypted Bulletin Board!

Bulletin Board

Alice:

Bob:

Charlie:

Tally

Clinton: 1
Rice: 2
An Encrypted Bulletin Board!

Ballot Casting Assurance

Tally
Clinton: 1
Rice: 2
An Encrypted Bulletin Board!

Bulletin Board

Alice: ✗✗✗✗✗
Bob: ✗✗✗✗✗
Charlie: ✗✗✗✗✗

Ballot Casting Assurance
Universal Verifiability

Tally
Clinton: 1
Rice: 2
Crypto Voting Schemes

Verification

Ballot Data Flow
Crypto Voting Schemes

Ballot Data Flow:
- Encrypted Votes
- Encryption

Verification:
- 

Alice
Adrienne
Crypto Voting Schemes

Verification
Ballot Data Flow

Encrypted
Votes
Alice
Adrienne
encryption
Crypto Voting Schemes

Verification

Ballot Data Flow

Encrypted Votes

anonymization

Alice

Adrienne

encryption
Crypto Voting Schemes

Verification

Ballot Data Flow

Encrypted Votes

anonymization

decryption

Alice

Adrienne

encryption
Crypto Voting Schemes

Results
Tally

Ballot Data Flow

Verification

Encryption

Encrypted
Votes

Anonymization

decryption

Tally

Results

-- Crypto Voting Schemes --

Alice

Adrienne

Verification

Ballot Data Flow
Crypto Voting Schemes

Registration Database

Encrypted Votes

anonymization

decryption

Tally

Results

Verification

Ballot Data Flow
Anonymous Tallying

Anonymization

Decryption

Results

Encrypted Votes

Tally

Results
Homomorphic Schemes

- Encrypted Votes
- Encrypted Tallying
- Encrypted Aggregate Tally (decryption)
- Results
Homomorphic Encryption

\[ \text{Enc}(m_1) \times \text{Enc}(m_2) = \text{Enc}(m_1 + m_2) \]
Homomorphic Encryption

\[ \text{Enc}(m_1) \times \text{Enc}(m_2) = \text{Enc}(m_1 + m_2) \]

\[ \text{Enc}_{pk}(m) = y^m r^s \mod n \]

\[ \text{Enc}_{pk}(m) = (g^r, g^m y^r) \mod p \]

- Residuosity encryption
- Exponential El-Gamal
Binary (Approval) Voting

Yes = Enc(1)
No = Enc(0)

[CohenFischer85, Benaloh86]
Binary (Approval) Voting

Yes = Enc(1)
No = Enc(0)

[Bulletin Board]

Alice: Enc(m_a)
Bob: Enc(m_b)
Charlie: Enc(m_c)

[CohenFischer85, Benaloh86]
Binary (Approval) Voting

Yes = Enc(1)
No = Enc(0)

EncryptedTally = Enc(m_a) × Enc(m_b) × Enc(m_c)
= Enc(m_a + m_b + m_c)

[B] CohenFischer85, Benaloh86 [B]
Needs Proofs!

[CohenFischer85, Benaloh86]
Needs Proofs!

\textbf{Eve:} \texttt{Enc(1000)}

Cannot trust that voters submit honest votes.

[CohenFischer85, Benaloh86]
Needs Proofs!

Eve: $\text{Enc}(1000)$

Cannot trust that voters submit honest votes.

ZK proof that each vote is $\text{Enc}(0)$ or $\text{Enc}(1)$

[CohenFischer85, Benaloh86]
Multi-Candidate Elections

Vote for Clinton: 2^{20} 

Vote for Rice: 2^{10} 

Vote for None: 2^0 

Tally: 0004 0001 0008

[P99, BFPSP2001]
Write-Ins?
Preserving Ballots?
Anonymous Tallying

Encrypted Votes → anonymization → decryption → Results

Tally
Mixnets

Republicans  Democrats  Independents
Mixnets

Mix servers operated by mutually suspicious organizations.
Chaumian Mixnet (Onions)

\[ c_j = \text{Enc}_{p k_1} (\text{Enc}_{p k_2} (\text{Enc}_{p k_3} (m_j))) \]

[Chaum81]
Chaumian Mixnet (Onions)

Each mix server “unwraps” a layer of this encryption onion.

\[ c_j = Enc_{p_{k_1}}(Enc_{p_{k_2}}(Enc_{p_{k_3}}(m_j))) \]

[Chaum81]
El Gamal Reencryption
El Gamal Reencryption

\[ sk = x \mod q \quad \text{and} \quad pk = y = g^x \mod p \]
El Gamal Re-encryption

\[ sk = x \mod q \quad pk = y = g^x \mod p \]

\[
\text{Enc}_{pk}(m; r) = (\alpha, \beta) = (g^r, m \cdot y^r)
\]

\[
\text{Dec}_{sk}(c) = \frac{\beta}{\alpha^x}
\]
El Gamal Reencryption

\[ sk = x \mod q \quad pk = y = g^x \mod p \]

\[ \text{Enc}_{pk}(m; r) = (\alpha, \beta) = (g^r, m \cdot y^r) \]

\[ \text{Dec}_{sk}(c) = \frac{\beta}{\alpha^x} \]

\[ \text{Reenc}_{pk}(c; r') = c \cdot \text{Enc}_{pk}(1, r') \]

\[ = (g^{r+r'}, m \cdot y^{r+r'}) \]
Reencryption Mixnet

\[ c_{\pi(j)} = \text{Reenc}(c_j; r_j) \]
First Proof of Mixnet

\[ \pi, \{ r_j \} \]

Primary Mix

\[ \pi', \{ r'_j \} \]

Secondary Mix

[SK94]
First Proof of Mixnet

On 1-bit challenge, reveal secondary mix, or secondary-to-primary difference.

[SK94]
Verifying a Mixnet (II)

• Neff’s Proof is fastest to date 8N modular exponentiations. 2000 ballots mixed in a few minutes.

• **Intuition**: dot product of input exponents with a random vector = dot product of output exponents with permuted random vector.

• Implemented in VoteHere technology.

[Neff2001]
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet
Verifying any Mixnet
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet
Verifying any Mixnet

[JJR2002]
Verifying any Mixnet

Tricks to ensure no complete path is revealed.

[JJR2002]
Anonymous Tallying

Private

Public
Anonymous Tallying

Homomorphic

Private

Public
Anonymous Tallying

Homomorphic

Encrypted
Identified
Ballots

Private
Public
Anonymous Tallying

Homomorphic

Encrypted Identified Ballots → Encrypted Tally

Private →
Public →
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Private → Public
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet
Anonymous Tallying

Homomorphic

Encrypted Identified Ballots → Encrypted Tally → Decrypted Tally

Mixnet

Encrypted Identified Ballots
Anonymous Tallying

**Homomorphic**
- Encrypted Identified Ballots → Encrypted Tally → Decrypted Tally

**Mixnet**
- Encrypted Identified Ballots → Encrypted De-Identified Ballots

Private → Public
Anonymous Tallying

Homomorphic

1. Encrypted Identified Ballots ➔ Encrypted Tally ➔ Decrypted Tally

Mixnet

1. Encrypted Identified Ballots ➔ Encrypted De-Identified Ballots ➔ Decrypted De-Identified Ballots
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally
Anonymous Tallying

Homomorphic

Encrypted Identified Ballots → Encrypted Tally → Decrypted Tally

Mixnet

Encrypted Identified Ballots → Encrypted De-Identified Ballots → Decrypted De-Identified Ballots → Decrypted Tally

Private → Public
Best of Both Worlds?

<table>
<thead>
<tr>
<th>Private</th>
<th>Public</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$</td>
<td>$c'_1$</td>
</tr>
<tr>
<td>$c_2$</td>
<td>$c'_2$</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
<tr>
<td>$c_n$</td>
<td>$c'_n$</td>
</tr>
</tbody>
</table>

[AW2006]
Best of Both Worlds?

\[ \pi, \{r'_i\} \]

Private

\[
\begin{align*}
C_1 \\
C_2 \\
\vdots \\
C_n
\end{align*}
\]

Public

\[
\begin{align*}
C'_1 \\
C'_2 \\
\vdots \\
C'_n
\end{align*}
\]

[AW2006]
Best of Both Worlds?

what if we could replace the private mixnet with a public program?

[AW2006]
Best of Both Worlds?

Private

$\pi, \{r'_i\}$

Public

$c_1, c'_1$
$c_2, c'_2$
$\vdots$
$c_n, c'_n$

what if we could replace the **private** mixnet with a **public** program?

[AW2006]
Best of Both Worlds?

what if we could replace the private mixnet with a public program?

Private

\[ \pi, \{ r'_i \} \]

Public

\[ c_1 \]
\[ c_2 \]
\[ \vdots \]
\[ c_n \]

\[ c'_1 \]
\[ c'_2 \]
\[ \vdots \]
\[ c'_n \]

what if we could replace the **private** mixnet with a **public** program?

[AW2006]
Best of Both Worlds?

what if we could replace the private mixnet with a public program?

[AW2006]

π, {r′ᵢ} → P

C₁, C₂, ..., Cₙ → P → C₁', C₂', ..., Cₙ'
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally

(Public Mixing)
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally

(Public Mixing)

- Encrypted Identified Ballots

Anonymous Tallying

Homomorphic

Encrypted Identified Ballots → Encrypted Tally → Decrypted Tally

Mixnet

Encrypted Identified Ballots → Encrypted De-Identified Ballots → Decrypted De-Identified Ballots → Decrypted Tally

(Public Mixing)

Encrypted Identified Ballots → Encrypted De-Identified Ballots

Private → Public
Anonymous Tallying

Homomorphic

1. **Encrypted Identified Ballots** → **Encrypted Tally** → **Decrypted Tally**

Mixnet

1. **Encrypted Identified Ballots** → **Encrypted De-Identified Ballots** → **Decrypted De-Identified Ballots** → **Decrypted Tally**

(Public Mixing)

1. **Encrypted Identified Ballots** → **Encrypted De-Identified Ballots** → **Decrypted De-Identified Ballots** → **Decrypted Tally**
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally

(Public Mixing)

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally
Anonymous Tallying

Homomorphic

- Encrypted Identified Ballots
- Encrypted Tally
- Decrypted Tally

Mixnet

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally

(Public Mixing)

- Encrypted Identified Ballots
- Encrypted De-Identified Ballots
- Decrypted De-Identified Ballots
- Decrypted Tally
Incoercibility
Incoercibility

Encrypted Votes

Alice

Adrienne
Incoercibility

- How do votes get to the bulletin board in the first place?
Incoercibility

- How do votes get to the bulletin board in the first place?

- Alice can’t perform the encryption herself, or she would be able to sell her vote.
Incoercibility

- How do votes get to the bulletin board in the first place?
- Alice can’t perform the encryption herself, or she would be able to sell her vote.
- How can Alice be certain that the encryption was performed correctly, yet still not be able to sell her vote?
Receipt-Freeness
Receipt-Freeness

- A Voting Machine prepares the encrypted ballot for Alice.

- The Machine provides Alice with a ZK proof that her vote was correctly encrypted.
Receipt-Freeness

- A Voting Machine prepares the encrypted ballot for Alice.
- The Machine provides Alice with a ZK proof that her vote was correctly encrypted.

[BT94]
Receipt-Freeness

- A Voting Machine prepares the encrypted ballot for Alice.
- The Machine provides Alice with a ZK proof that her vote was correctly encrypted.
- Requirement: private voting booth where crucial elements of this proof occur.

[BT94]
MarkPledge
MarkPledge

- Challenge Ticket

[Neff2004]
MarkPledge

- Challenge Ticket
- Fill out ballot normally
MarkPledge

- Challenge Ticket
- Fill out ballot normally
- Get a printed receipt and Screen Confirmation Code

Alice

[Rice: dhjq]

[Receipt]

[ab54]

[Neff2004]
MarkPledge

- Challenge Ticket
- Fill out ballot normally
- Get a printed receipt and Screen Confirmation Code
- Scan Challenge Ticket

[Neff2004]
MarkPledge

- Challenge Ticket
- Fill out ballot normally
- Get a printed receipt and Screen Confirmation Code
- Scan Challenge Ticket
- Receive complete receipt

[Neff2004]
MarkPledge

- Challenge Ticket
- Fill out ballot normally
- Get a printed receipt and Screen Confirmation Code
- Scan Challenge Ticket
- Receive complete receipt
- Verify that codes match

[Neff2004]
MarkPledge (II)

- Receipt contains encrypted ballot that can be checked against bulletin board by voter or a helper.

- Codes are random: Alice cannot convince anyone that dhjq was the “real” code.

Receipt:

Ticket: ab54
Clinton: 34c7
Rice: dhjq
None: 8489

[Neff2004]
Paper-Based
Cryptographic
Voting Schemes
Secret-Ballot Receipts

• First scheme to use the physical properties of paper (and overlays) to achieve encrypted receipts.

• Uses of visual cryptography: two overlays achieves a visual XOR

• Central inspiration for simpler paper-based schemes.

[Chaum2004]
<table>
<thead>
<tr>
<th></th>
<th>Clinton</th>
<th>None</th>
<th>Rice</th>
</tr>
</thead>
<tbody>
<tr>
<td>8c3sw</td>
<td>_______</td>
<td>_______</td>
<td>_______</td>
</tr>
</tbody>
</table>
Prêt-à-Voter

<table>
<thead>
<tr>
<th>Clinton</th>
<th>None</th>
<th>Rice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[ChaumRyanSchneider2004]
Prêt-à-Voter

[ChaumRyanSchneider2004]
Prêt-à-Voter

[ChaumRyanSchneider2004]
PaV: Onion Decryption
PaV: Onion Decryption

Onion = Enc_{pk_1}(r_1; Enc_{pk_2}(r_2; Enc_{pk_3}(r_3)))
PaV: Onion Decryption

Onion = Enc_{pk_1}(r_1; Enc_{pk_2}(r_2; Enc_{pk_3}(r_3)))
PaV: Onion Decryption

Onion = Enc_{pk_1}(r_1; Enc_{pk_2}(r_2; Enc_{pk_3}(r_3)))

- Rice
- Clinton
- None
PaV: Onion Decryption

Onion = Enc_{p k_1}(r_1; Enc_{p k_2}(r_2; Enc_{p k_3}(r_3)))
PaV: Onion Decryption

Onion = Enc_{pk_1}(r_1; Enc_{pk_2}(r_2; Enc_{pk_3}(r_3)))
PaV: Onion Decryption

Onion = \text{Enc}_{p_{k_1}}(r_1; \text{Enc}_{p_{k_2}}(r_2; \text{Enc}_{p_{k_3}}(r_3)))
PaV: Onion Decryption

Onion = $\text{Enc}_{pk_1}(r_1; \text{Enc}_{pk_2}(r_2; \text{Enc}_{pk_3}(r_3)))$

$r_1$:
- None
- Rice
- Clinton

$r_2$:
- Rice
- None
- Clinton

$r_3$:
- Rice
- Clinton
- None
PaV: Onion Decryption

Onion = Enc_{pk_1} (r_1; Enc_{pk_2} (r_2; Enc_{pk_3} (r_3)))
PaV: Onion Decryption

Onion = $\text{Enc}_{p_{k_1}}(r_1; \text{Enc}_{p_{k_2}}(r_2; \text{Enc}_{p_{k_3}}(r_3)))$

<table>
<thead>
<tr>
<th>Clinton</th>
<th>Rice</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
PaV: Onion Decryption

Onion = Enc_{pk_1}(r_1; Enc_{pk_2}(r_2; Enc_{pk_3}(r_3)))
PaV: Onion Decryption

Onion = $\text{Enc}_{pk_1}(r_1; \text{Enc}_{pk_2}(r_2; \text{Enc}_{pk_3}(r_3)))$
PaV: Onion Decryption

Onion = $\text{Enc}_{pk_1}(r_1; \text{Enc}_{pk_2}(r_2; \text{Enc}_{pk_3}(r_3)))$

<table>
<thead>
<tr>
<th>Clinton</th>
<th>None</th>
<th>Rice</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔️</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$r_1$ $r_2$ $r_3$
PaV: Verification

- Before Election, half of the ballots are audited: the onions are opened up and the order of the candidates confirmed.

- Audited Ballots are spoiled: they no longer protect privacy
Punchscan

[Chaum2005]
Punchscan

[Chaum2005]
Punchscan

Rice - x
Clinton - m
None - r

8c3sw

[Chaum2005]
Punchscan

Rice - x
Clinton - m
None - r

8c3sw

Rice - x
Clinton - m
None - r

8c3sw

Rice - x
Clinton - m
None - r

8c3sw

[Chaum2005]
Punchscan

[Chaum2005]
Punchscan Verification

• After election day, officials reveal what the kept half should have looked like.

• Voters’ 50-50 choice of which sheet to keep provides 50% soundness (per voter)

• Voter is involved directly!
Scratch & Vote

PARAMETERS
#1 - Rice
#2 - Clinton
#3 - None

M=10, Public Key = pk

Clinton
None
Rice

E_{pk}(2^{10}; r_1)
E_{pk}(2^{20}; r_2)
E_{pk}(2^0; r_3)

r_1 r_2 r_3

[AR2006]
S&V Verification

• Alice takes two ballots, one for audit, one for voting. She selects which one to audit at random.

• Auditing: scratch off the surface (voiding the ballot), verify that the ciphertexts encoded in the ballot are correct given randomness.

• Vote with remaining ballot: 50% soundness and immediate ability to detect and complain while at the polling place.
e2e Verification

Voting Machine

Vendor

Polling Location

Ballot Box / Bulletin Board

1

Receipt

Alice

2

Results

/*
 * source
 * code
 */

if (....)
Is Crypto Voting Too Difficult to Explain/Use?
Is Crypto Voting Too Difficult to Explain/Use?

- Sure, it’s complicated
Is Crypto Voting Too Difficult to Explain/Use?

• Sure, it’s complicated

• But so is formal modeling of programs to ensure that voting machines are properly implemented (only we let officials worry about that.)
Is Crypto Voting Too Difficult to Explain/Use?

- Sure, it’s complicated
- But so is formal modeling of programs to ensure that voting machines are properly implemented (only we let officials worry about that.)
- The voter experience is almost as simple as non-crypto schemes.
Is Crypto Voting Too Difficult to Explain/Use?

- Sure, it’s complicated
- But so is formal modeling of programs to ensure that voting machines are properly implemented (only we let officials worry about that.)
- The voter experience is almost as simple as non-crypto schemes.
- **Anyone** can verify!
Questions?