



Electronic Voting

Ronald L. Rivest

MIT CSAIL

NSA June 3, 2004

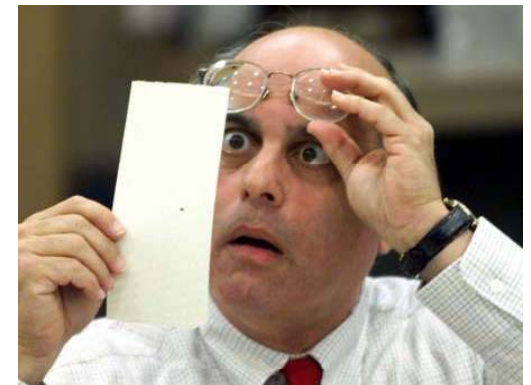
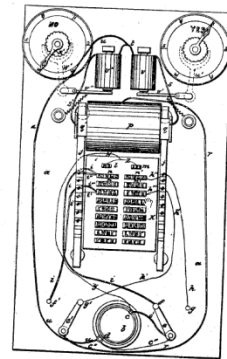


Outline

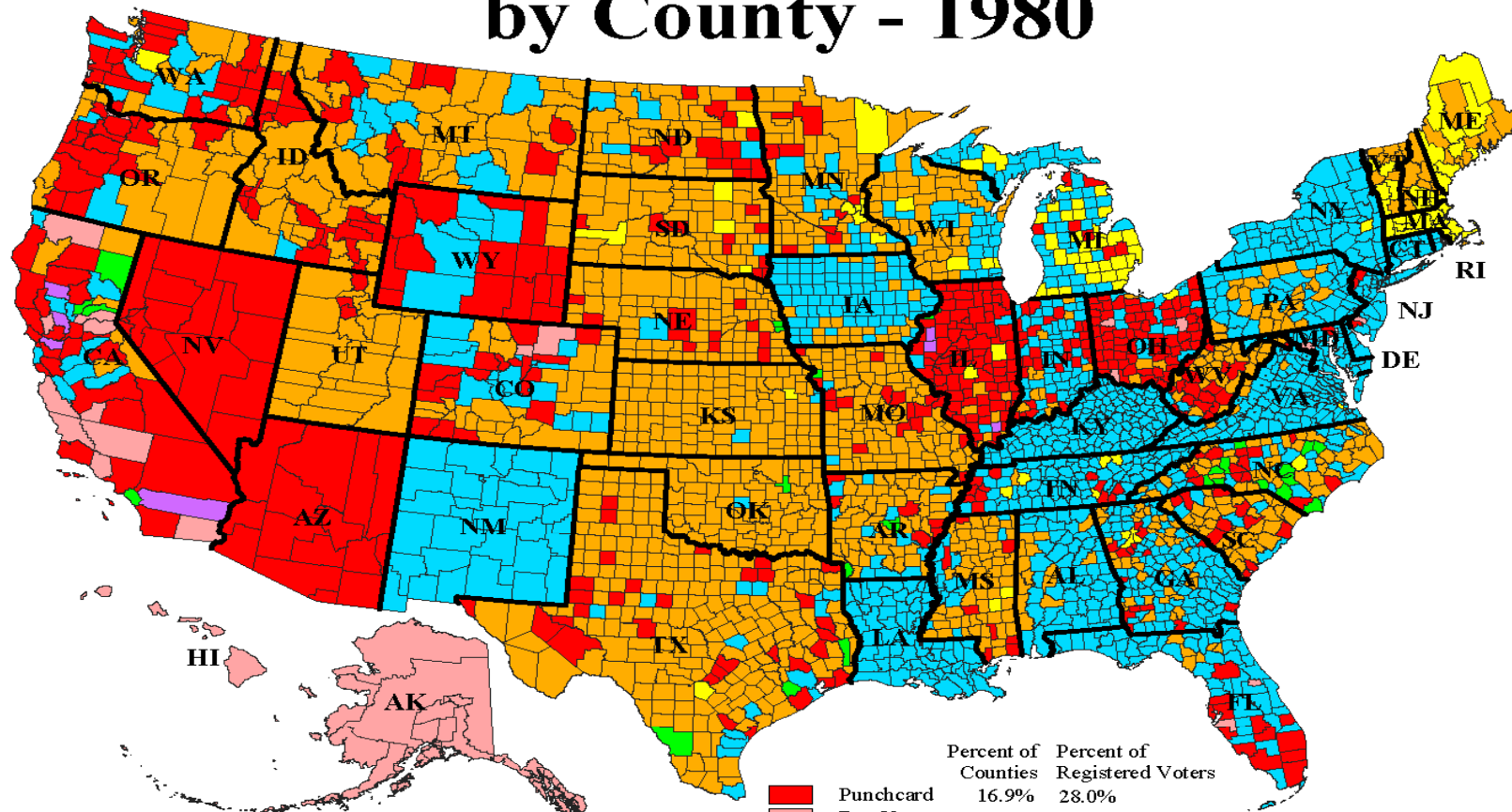
- ◆ Introduction / Voting
- ◆ Voting using mix-nets
- ◆ Randomized Partial Checking
(Jakobsson/Juels/Rivest USENIX '02)
- ◆ Pedagogic variant of Chaum's proposal

Voting tech is in transition...

- ◆ Voting tech follows technology:
Stones → Paper → Levers →
Punch cards → Op-scan →
Computers(??)
- ◆ Punch cards "out" after Nov. '00
- ◆ DRE's (touch-screen) require
VVPAT (voter-verified paper audit
trail) in Cal.
- ◆ Is technology ready for
electronic (paperless) voting?



Type of Voting Equipment by County - 1980



Alaska does not have counties.
DataVote systems is used statewide
except for a few paper ballot precincts.

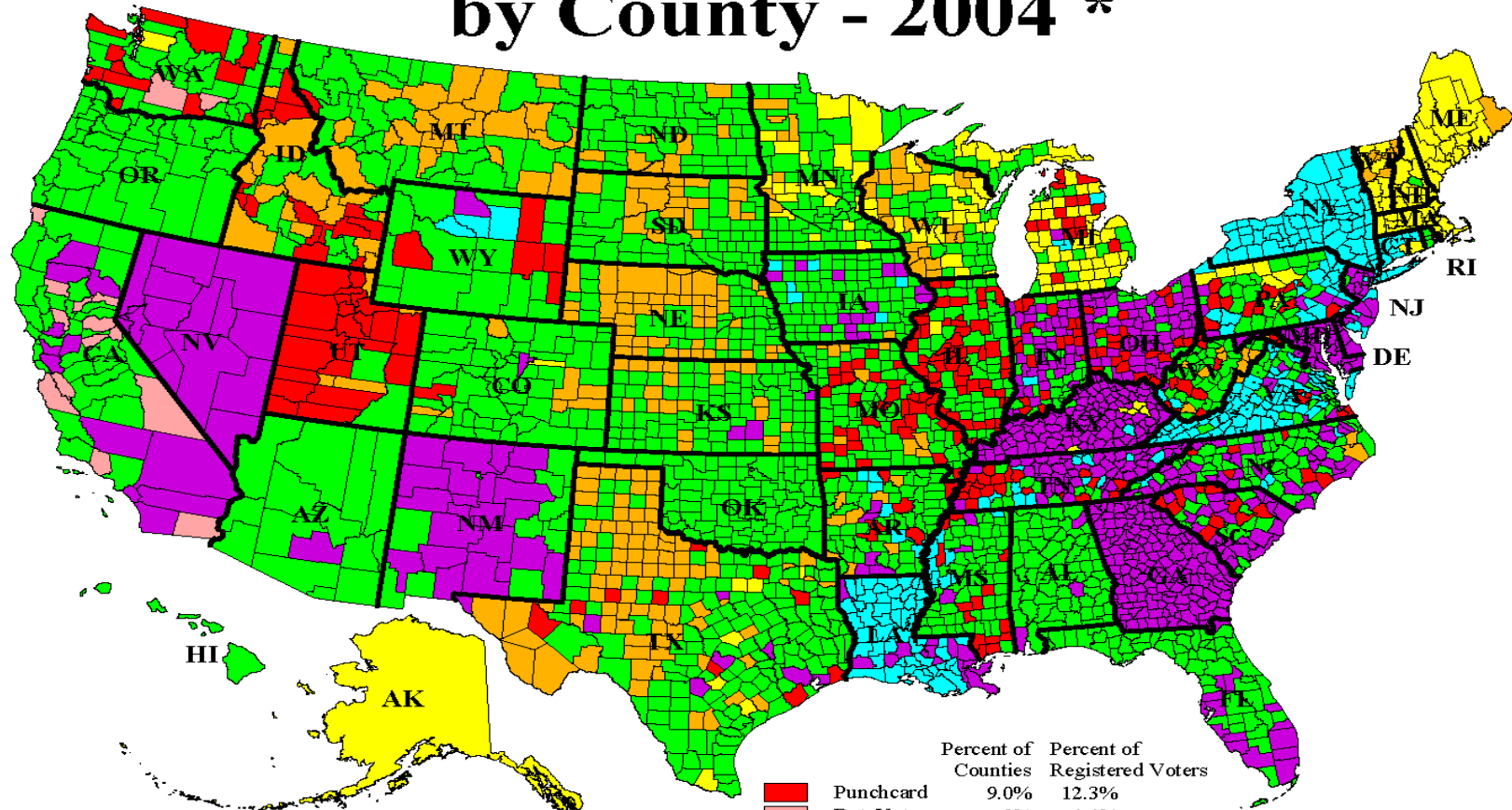
	Percent of Counties	Percent of Registered Voters
■ Punchcard	16.9%	28.0%
■ DataVote	1.1%	3.0%
■ Lever	36.9%	42.9%
■ Paper	41.0%	10.8%
■ Optical	.8%	2.1%
■ Electronic	.2%	.7%
■ Mixed Systems	3.0%	12.5%

Equipment used in the November 1980
election as reported by state election officials.
The map shows equipment used at polling
places, not necessarily absentee balloting.

Copyright © 2004 Election Data Services, Inc.
Revised 5/4/2004.

Election Data Services
 (202) 789-2004 
 1401 K Street NW, Suite 500
 Washington, DC 20005-3417
www.ElectionDataServices.com

Type of Voting Equipment by County - 2004 *



Alaska does not have counties.
AccessVote system is used statewide
except for a few paper ballot precincts.

	Percent of Counties	Percent of Registered Voters
■ Punchcard	9.0%	12.3%
■ DataVote	.8%	1.4%
■ Lever	8.6%	13.9%
■ Paper	9.6%	.7%
■ Optical	45.4%	33.7%
■ Electronic	21.7%	30.8%
■ Mixed Systems	4.8%	7.2%

* Equipment expected to be used
in the November 2004 election as
reported by state election officials.
The map shows equipment used at polling
places, not necessarily absentee balloting.

Copyright © 2004 Election Data Services, Inc.
*Printed with information available as of 5/3/2004.

Election Data Services
(202) 789-2004
1401 K Street NW, Suite 500
Washington, DC 20005-3417
www.ElectionDataServices.com

Voting is a hard problem

- ◆ **Voter Registration** - each eligible voter votes at most once
- ◆ **Voter Privacy** - no one can tell how any voter voted, even if voter wants it; no "receipt" for voter
- ◆ **Integrity** - votes can't be changed, added, or deleted; tally is accurate.
- ◆ **Availability** - voting system is available for use when needed
- ◆ **Ease of Use** - esp. for disabled

Voting is important

- ◆ Cornerstone of our (any!) democracy
- ◆ *Voting security is clearly an aspect of national security.*
- ◆ "Those who vote determine nothing; those who count the votes determine everything." -- *Joseph Stalin*

Are DRE's trustworthy?

- ◆ Diebold fiascoes..??
- ◆ Intrinsic difficulty of designing and securing complex systems
- ◆ Many units (100,000's) in field, used occasionally, and managed by the semi-trained
- ◆ Certification process is "riddled with problems" (NYT editorial 5/30/04)



Voter-Verified Paper Audit Trails?

- ◆ Rebecca Mercuri: Voting machine should produce "paper audit trail" that voter can inspect and approve.
- ◆ VVPAT is "official ballot" in case of dispute or recounts.
- ◆ David Dill (Stanford CS Prof.) initiated on-line petition that ultimately resulted in California requiring VVPAT's on many DRE's.

VVPAT's controversial...

- ◆ Still need to guard printed ballots.
- ◆ Two-step voting procedure may be awkward for some voters (e.g. disabled).
- ◆ Doesn't catch all problems (e.g. candidate missing from slate)
- ◆ Malicious voters can cause DOS by casting suspicion on voting machine
- ◆ Not "*end-to-end*" security:
 - Helps ensure votes "cast as intended"
 - Doesn't help ensure votes "counted as cast".

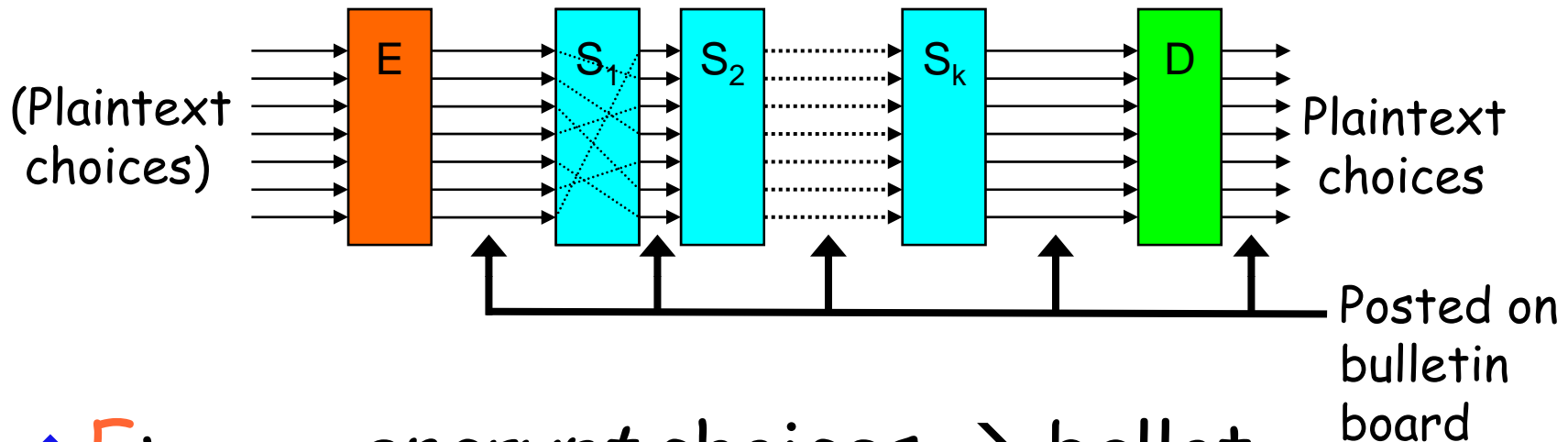
Outline

- ◆ Introduction / Voting
- ◆ Voting using mix-nets
- ◆ Randomized Partial Checking
(Jakobsson/Juels/Rivest USENIX '02)
- ◆ Pedagogic variant of Chaum's proposal

Can *cryptology* help?

- ◆ Yes - using "mix-nets" (Chaum) and "voter-verified secret ballots" (Chaum; Neff)
- ◆ Official ballot is *electronic* not paper.
- ◆ Ballot is *encrypted* version of choices.
- ◆ Ballots posted on public *bulletin board*.
- ◆ Voter gets paper "receipt" so she can:
 - Ensure that her ballot is properly posted
 - Detect voting machine error or fraud

Voting using mix-nets



- ◆ **E**: *encrypt* choices → ballot
(done at each voting machine)
- ◆ $S_1 \dots S_k$: *mix-servers* provide anonymity
(secretly permute and re-encrypt)
- ◆ **D**: *decrypt* ballots
(trustees threshold decrypt)

Voter needs evidence

- ◆ That her vote is "cast as intended":
- ◆ That her ballot is indeed encryption of her choices, and what her ballot is.
 - ▶ This is extremely challenging, since
 - ▶ She can't compute much herself
 - ▶ She can't take away anything that would allow her to prove how she voted
- ◆ So: she takes away *evidence* that allows her (as she exits polling site) to detect whether cheating occurred, and *receipt* to prove what her ballot is.

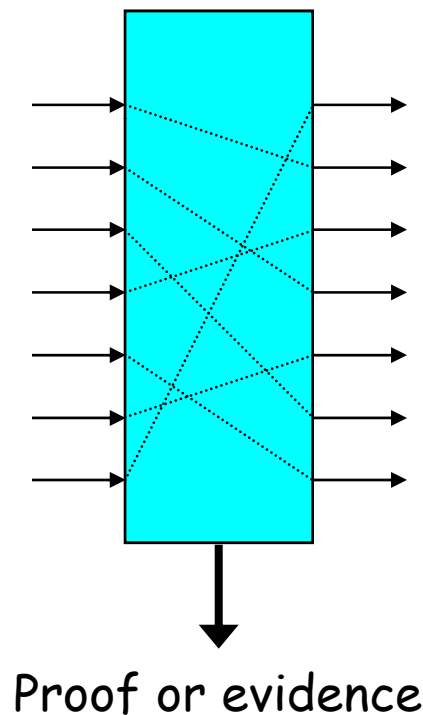
Everyone needs evidence

- ◆ That votes are "counted as cast":
- ◆ That mix-servers ("mixes") properly permute and re-encrypt ballots.
 - This is challenging, since
 - Mixes can not reveal the permutation they applied to ballots
- ◆ That trustees properly decrypt the permuted ballots
 - This is relatively straightforward, using known techniques.

Outline

- ◆ Introduction / Voting
- ◆ Voting using mix-nets
- ◆ Randomized Partial Checking
(Jakobsson/Juels/Rivest USENIX '02)
- ◆ Pedagogic variant of Chaum's proposal

Robust mixes



- ◆ Provide *proof* (or at least *strong evidence*) of their correct operation.
- ◆ *Anyone* can check proof.
- ◆ Even if all mixes are corrupt and collude, it is infeasible for them to produce such proof (*universally verifiable*).
- ◆ Proof does not reveal input / output correspondence!

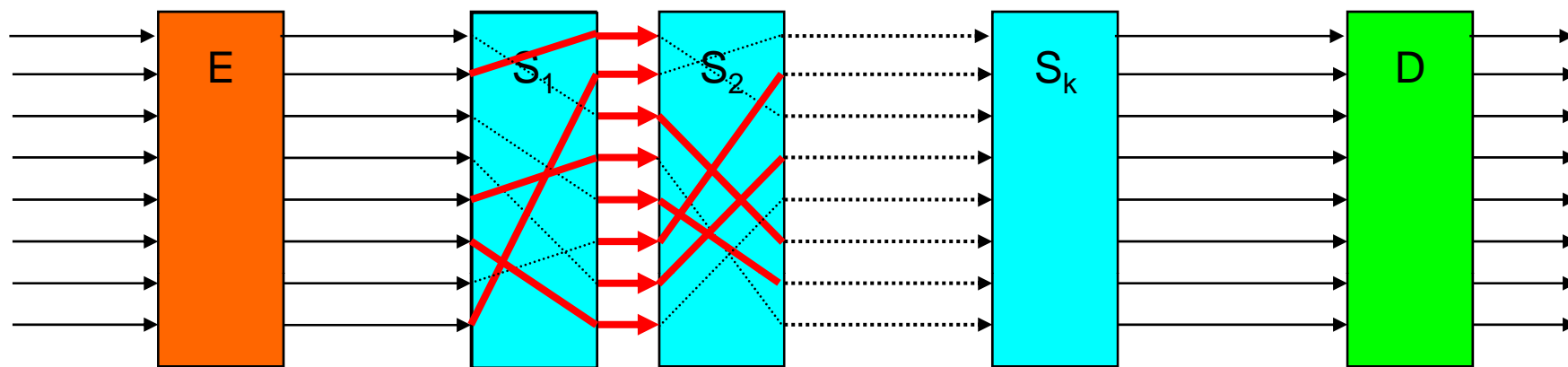
Practical Robust Mixes

- ◆ Jakobsson "Flash Mix" (PODC '99)
- ◆ Mitomo and Kurosawa (Asiacrypt '00)
- ◆ Desmedt and Kurosawa (EC '00)
- ◆ Neff (ACM CCS '01)
- ◆ Furukawa-Sako (Crypto '01)
- ◆ Golle (ACM CCS '02)
- ◆ Golle, Zhong, Boneh, Jakobsson, Juels (Asiacrypt '02)
- ◆ ...

"Randomized Partial Checking Mix

- ◆ Conceptually very simple
- ◆ Very efficient
- ◆ Works with any cryptosystem
- ◆ Aimed at voting
- ◆ Force each mix to reveal and prove *half* of its input-output correspondences
- ◆ No complete path from input to output revealed; voter's anonymity preserved within set of at least $\frac{1}{2}$ the voters.

RPC illustrated



- ◆ Mixes are *paired* (S_1, S_2), (S_3, S_4), etc.
- ◆ For each ballot B between elements of a pair (e.g. (S_1, S_2)), produce "challenge bit" b from hash of all bulletin board contents
- ◆ If $b = 0$, first server must reveal where B came from and prove it by revealing keys/randomness.
- ◆ If $b = 1$, second server must reveal where B goes and prove it by revealing keys/randomness.

Security theorem

- ◆ An adversary who queries random oracle (\approx hash function) at most q times will have a chance of at most $q 2^{-t}$ of producing a bulletin board transcript that passes public verification yet where the vote count has been altered by t votes.

Outline

- ◆ Introduction / Voting
- ◆ Voting using mix-nets
- ◆ Randomized Partial Checking
(Jakobsson/Juels/Rivest USENIX '02)
- ◆ Pedagogic variant of Chaum's proposal

A pedagogical variant of Chaum's voting proposal

- ◆ Used in my class this spring as introductory example, before going into details of Chaum's and Neff's schemes.
- ◆ Captures many significant features, but not all; some problems/concerns not well handled.
- ◆ Intended to be simpler to explain and understand than full versions.
- ◆ Related to Jakobsson/Juels/Rivest RPC mix-net scheme.
- ◆ Main ideas (e.g. cut and choose) already present in Chaum's scheme.

Pedagogical variant (overview)

- ◆ Voting machine produces ballot that is *encryption* of voter's choices.
- ◆ Ballot is posted on bulletin board as "official cast ballot" (electronic).
- ◆ Voter given *receipt copy* of ballot.
- ◆ Voter given *evidence* that ballot correctly encodes his intended choices.
- ◆ Ciphertexts "mixed" for anonymity.
- ◆ Ciphertexts decrypted and counted (threshold decryption by trustees).

Pedagogical variant (details)

- ◆ Voter V_i prepares choices B_i
- ◆ Machine prints and signs B_i, C_i, D_i, r_i, s_i and gives them to voter.
 C_i is encryption of B_i (randomization r_i)
 D_i is re-encryption of C_i (randomization s_i)
- ◆ If voter doesn't like B_i , she starts over.
- ◆ Voter destroys either r_i or s_i , and keeps the other information as *evidence* (paper).
- ◆ Voting machine signs and posts $(V_i, D_i, \text{"final"})$, and gives (paper) *receipt copy* to voter.
- ◆ Final D_i 's mixed up (mixnet), decrypted, and counted.

Pedagogical variant (details)

$$B_i \xrightarrow{r_i} C_i \xrightarrow{s_i} D_i$$

- ◆ El-Gamal encryption and re-encryption:
 $C_i = (g^{r_i}, B_i * y^{r_i}), D_i = (g^{r_i+s_i}, B_i * y^{r_i+s_i})$
- ◆ Voter keeps only one link as evidence (similar to Jakobsson/Juels/Rivest, or Chaum)
- ◆ Any attempt by voting machine to cheat will be detected with probability $\frac{1}{2}$.
- ◆ Voter can check evidence on exit.
- ◆ Signed B_i 's are easy to get...

Pedagogical variant (details)

$$B_i \xrightarrow{r_i} C_i \quad D_i$$

- ◆ El-Gamal encryption and re-encryption:
 $C_i = (g^{r_i}, B_i * y^{r_i}), D_i = (g^{r_i+s_i}, B_i * y^{r_i+s_i})$
- ◆ Voter keeps only one link as evidence (similar to Jakobsson/Juels/Rivest, or Chaum)
- ◆ Any attempt by voting machine to cheat will be detected with probability $\frac{1}{2}$.
- ◆ Voter can check evidence on exit.
- ◆ Signed B_i 's are easy to get...

Pedagogical variant (details)

$$B_i \quad C_i \xrightarrow{S_i} D_i$$

- ◆ El-Gamal encryption and re-encryption:
 $C_i = (g^{r_i}, B_i * y^{r_i}), D_i = (g^{r_i+s_i}, B_i * y^{r_i+s_i})$
- ◆ Voter keeps only one link as evidence (similar to Jakobsson/Juels/Rivest, or Chaum)
- ◆ Any attempt by voting machine to cheat will be detected with probability $\frac{1}{2}$.
- ◆ Voter can check evidence on exit.
- ◆ Signed B_i 's are easy to get...

Variant with "visual crypto"

- ◆ Naor/Shamir: can do "xor" visually:

$$\begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array} + \begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array} = \begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array} \quad 0 + 0 = 0$$

$$\begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array} + \begin{array}{|c|c|} \hline \square & \blacksquare \\ \hline \blacksquare & \square \\ \hline \end{array} = \begin{array}{|c|c|} \hline \blacksquare & \blacksquare \\ \hline \blacksquare & \blacksquare \\ \hline \end{array} \quad 0 + 1 = 1$$

$$\begin{array}{|c|c|} \hline \square & \blacksquare \\ \hline \blacksquare & \square \\ \hline \end{array} + \begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array} = \begin{array}{|c|c|} \hline \blacksquare & \blacksquare \\ \hline \blacksquare & \blacksquare \\ \hline \end{array} \quad 1 + 0 = 1$$

$$\begin{array}{|c|c|} \hline \square & \blacksquare \\ \hline \blacksquare & \square \\ \hline \end{array} + \begin{array}{|c|c|} \hline \square & \blacksquare \\ \hline \blacksquare & \square \\ \hline \end{array} = \begin{array}{|c|c|} \hline \square & \blacksquare \\ \hline \blacksquare & \square \\ \hline \end{array} \quad 1 + 1 = 0$$

Variant with visual crypto

$$\begin{array}{ccc} B'_i & \xrightarrow{r'_i} & D'_i \\ + & & \\ B''_i & \xrightarrow{r''_i} & D''_i \end{array}$$

$$B_i$$

- ◆ Print B'_i and B''_i on transparencies
- ◆ Visually verify $B'_i + B''_i = B_i$
- ◆ Keeps D'_i , D''_i , and either (B'_i, r'_i) or (B''_i, r''_i)

Variant with visual crypto

$$B'_i \xrightarrow{r'_i} D'_i$$
$$D''_i$$

- ◆ Print B'_i and B''_i on transparencies
- ◆ Visually verify $B'_i + B''_i = B_i$
- ◆ Keeps D'_i , D''_i , and either (B'_i, r'_i) or (B''_i, r''_i)

Variant with visual crypto

$$B''_i \xrightarrow{r''_i} D''_i$$

D'_i

- ◆ Print B'_i and B''_i on transparencies
- ◆ Visually verify $B'_i + B''_i = B_i$
- ◆ Keeps D'_i , D''_i , and either (B'_i, r'_i) or (B''_i, r''_i)

Variant with visual crypto

- ◆ Any attempt by voting machine to cheat will result in detection with probability $\frac{1}{2}$.

Pedagogical variant (summary)

- ◆ Schemes such as these (Chaum / Neff) provide an interesting degree of "*end-to-end*" security: from *voter's intentions* to *final tally*.
- ◆ Paper is used, but not to record official ballots or for recounts, but as *commitments* so fraud and error can be detected.

Conclusions

- ◆ Voting technology is in a state of transition to electronics.
- ◆ It seems possible to have electronic voting without:
 - trusting machines for integrity
 - using paper ballots for recounts
 - revealing how any voter votes
- ◆ How can we do all of this *well*?

(The End)
