

eTegrity, eScantegrity and ePunchScan

David Chaum, **Stefan Popoveniuc**, Poorvi Vora
End-to-End Voting Systems Workshop
October 13-14, 2009

Previous work

- Most end-to-end verifiable voting systems use paper
 - Pret a Voter
 - PunchScan
 - Scratch&Vote
 - Scantegrity
- Paper serves as a write-once media
- Problems with paper:
 - Un-accessible
 - Elections officials don't want to handle paper
- Helios (Benaloh challenge) does not use paper

Trusted Computing Device (TCD)

- Paper replaced by TCD
- TCD and voting system do not collude
- The TCD is trusted to follow the protocol
 - not trusted for integrity
 - not trusted for privacy
- TCD can be programmed by
 - the voter
 - a helper organization
 - a coercer
- TCD knows crypto, but does not have a private key

Model

- A coercer can examine TCD's memory, but not program it
- A publically verifiable back-end for counting the votes
 - Distributed among non-colluding entities
- Properties obtained
 - Privacy
 - Incoercibility
 - integrity

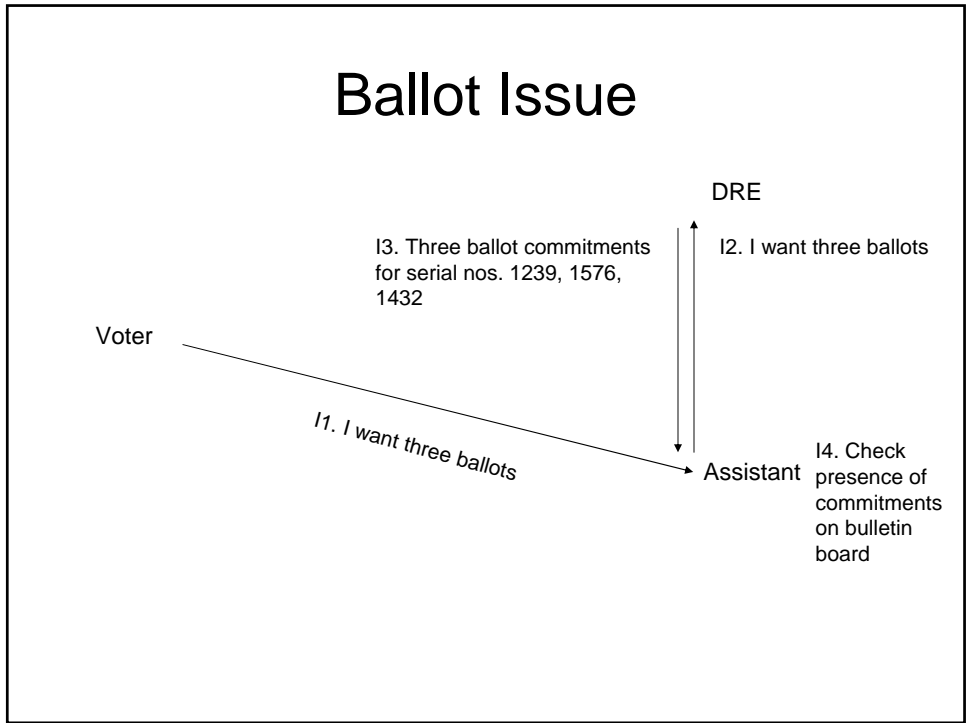
Front-end

- The three new protocols only address “the voting ceremony” – how the clear text vote is “encrypted” and cast
- To count the votes, use any back-end
 - Onion mixnets
 - PunchScanian mixnets
 - Pointer-based mixnets (Scantegrity style)
 - Homomorphic tallying
- Polling place voting

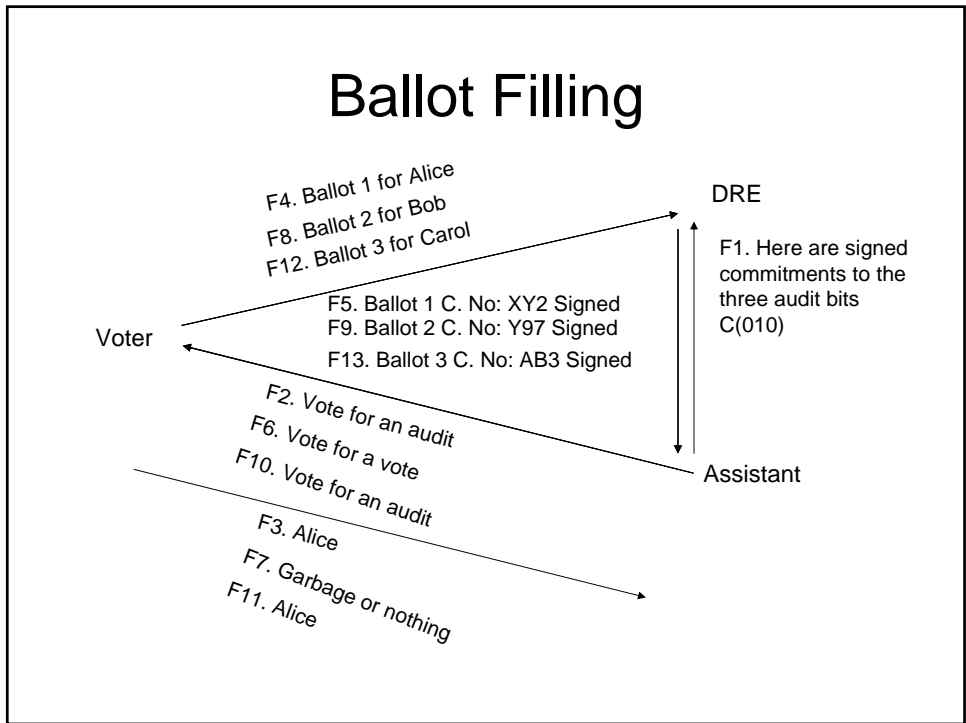
eTegrity

- A generalization of the Benaloh challenge
- The voter fills-in many ballots
 - Casts one ballot
 - Audits the other ballots

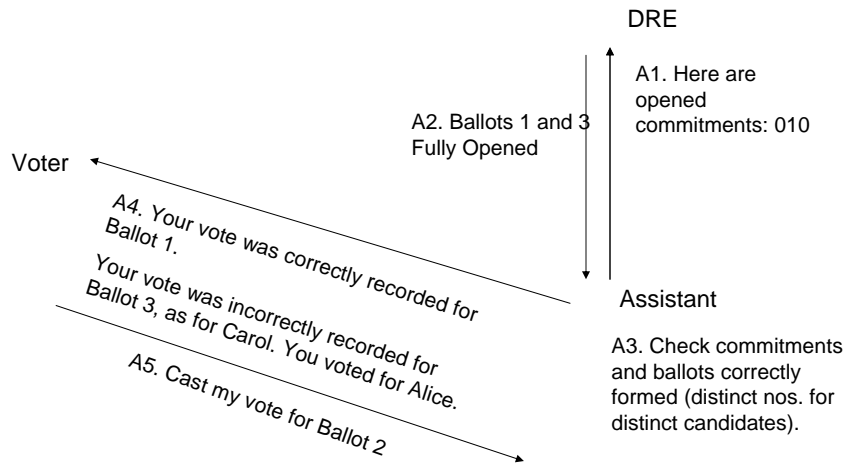
Ballot Issue



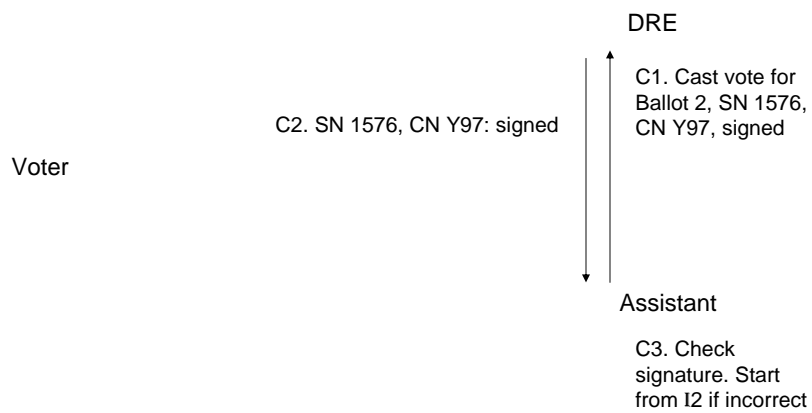
Ballot Filling



Ballot Audit



Ballot Casting



Brief security analysis

Pros:

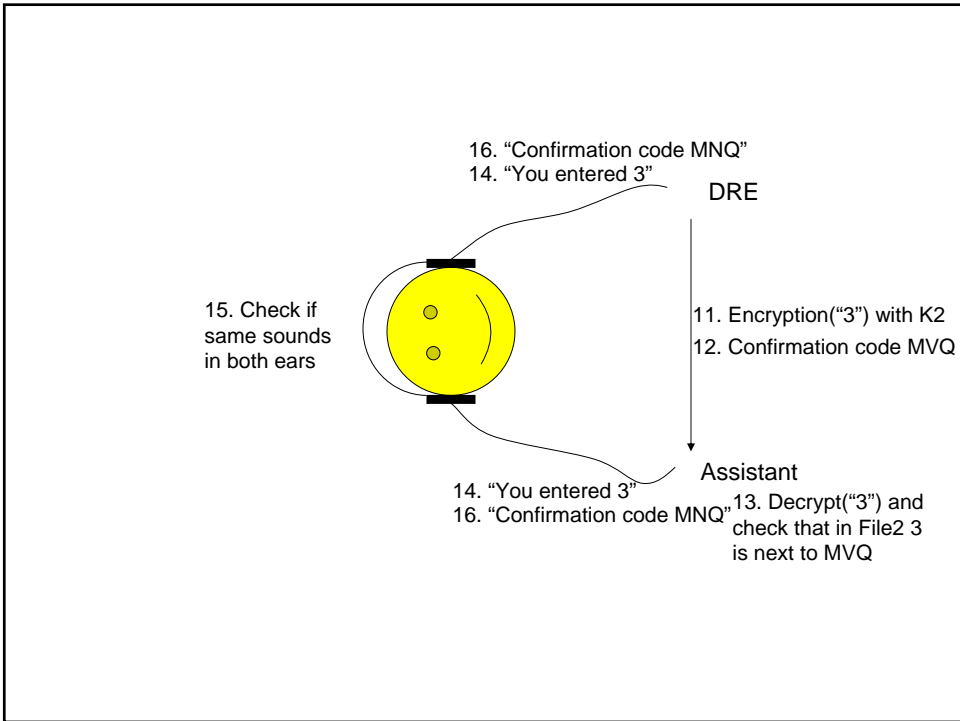
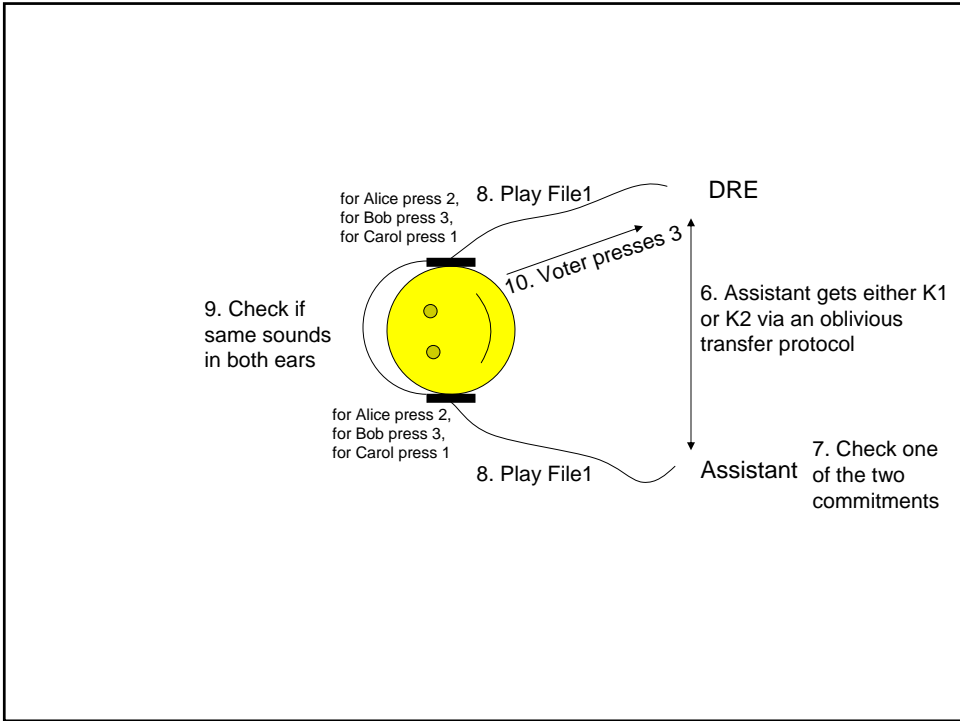
- Assistant does not learn the vote cast
- 1/n probability of cheating by DRE

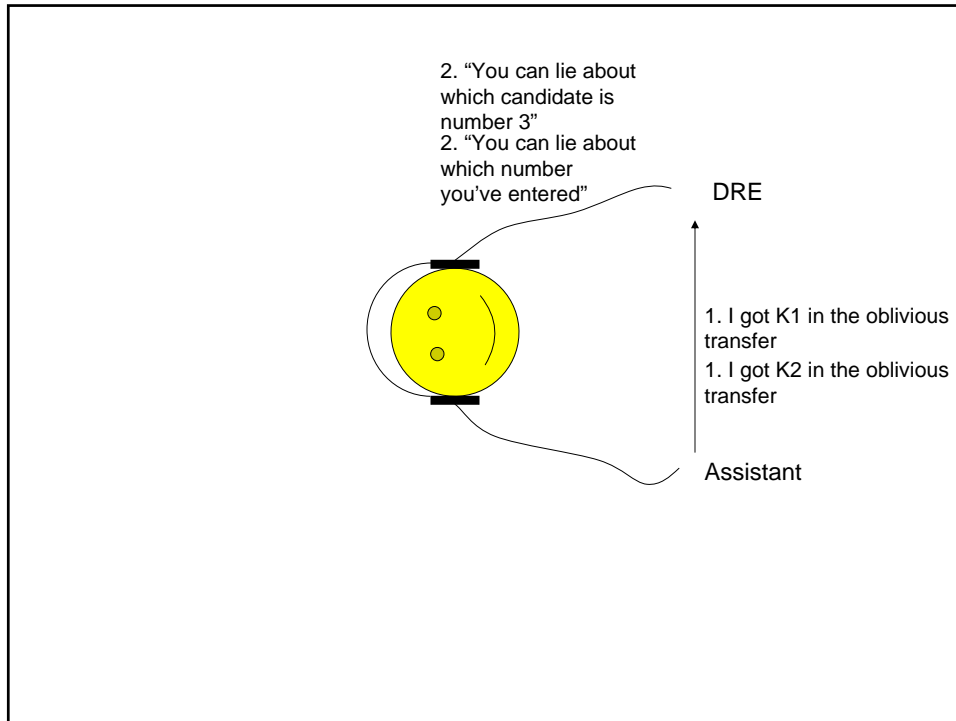
Cons:

- No proof of cheating if DRE ignores voter's input
- Voter has to tell DRE many votes and remember them
- If a coercer programs Assistant not to follow the protocol, then the voter can be coerced

eScantegrity

1. DRE computes File1:
 - a. for Alice press 2,
 - b. for Bob press 3,
 - c. for Carol press 1
2. DRE computes File2:
 1. confirmation code for 1 is B7K
 2. confirmation code for 2 is X8T
 3. confirmation code for 3 is MWQ
3. DRE publishes commitments to File1 and File2
4. DRE generates two secret keys K1 and K2
5. DRE sends Assistant File1 encrypted with K1 and File2 encrypted with K2





Brief security analysis

Pros:

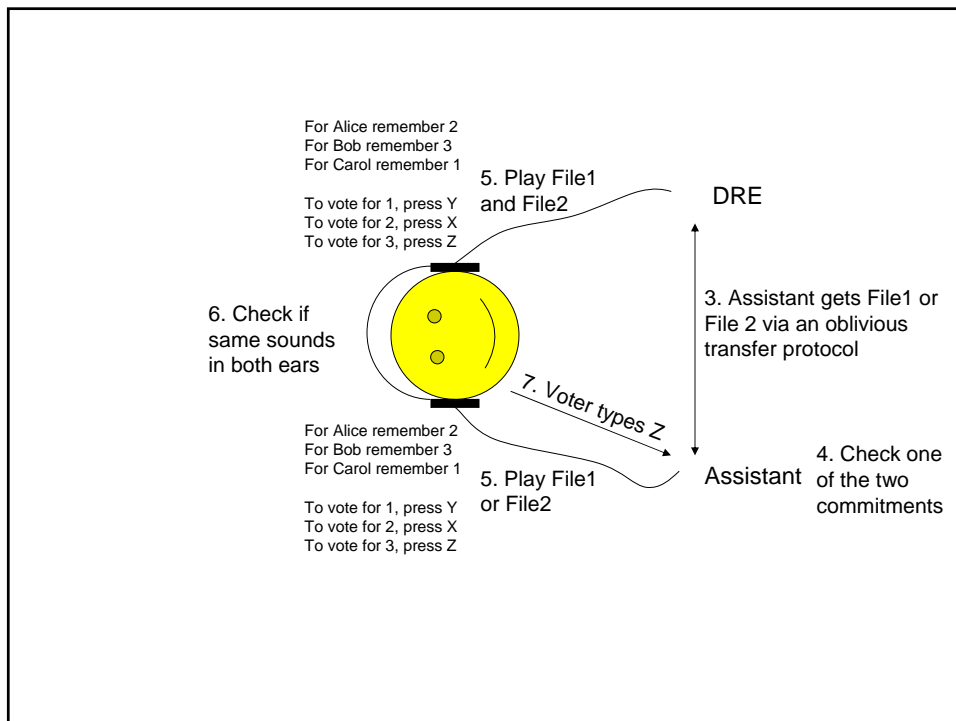
- Assistant does not learn the vote cast
- 1/2 probability of cheating by DRE

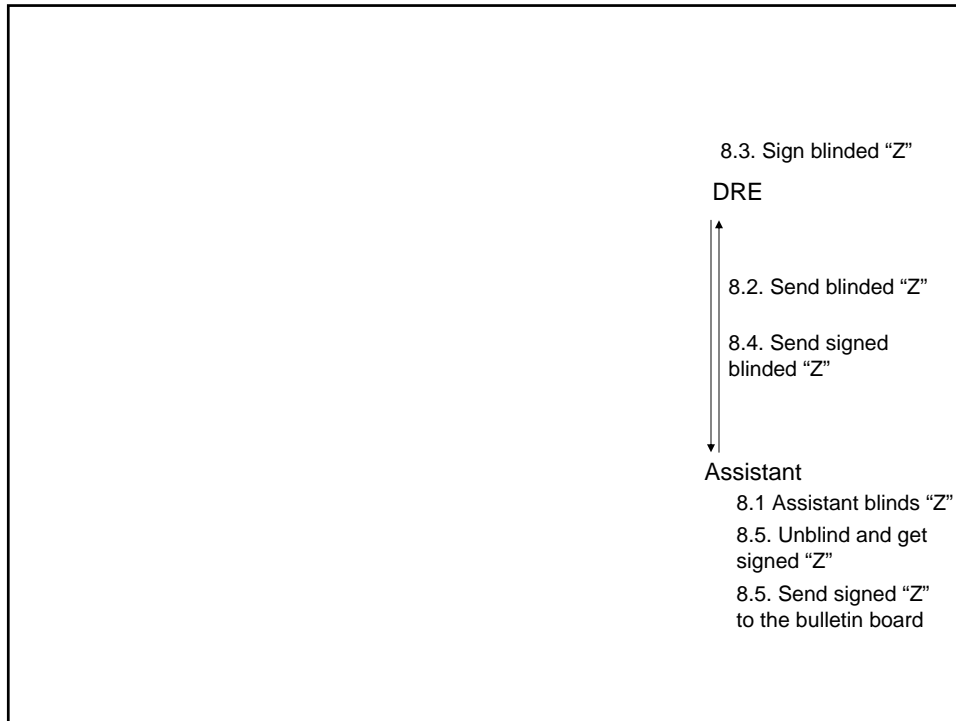
Cons:

- No proof of cheating if DRE ignores voter's input
- No proof of cheating if voter hears different things in different ears.
- If a coercer programs Assistant not to follow the protocol, then the voter can be coerced

ePunchScan

1. DRE computes File1:
 - a. for Alice remember 2
 - b. for Bob remember 3
 - c. for Carol remember 1and DRE computes File2:
 - To vote for 1, press Y
 - To vote for 2, press X
 - To vote for 3, press Z
2. DRE publishes commitments to File1 and File2





Brief security analysis

Pros:

- Assistant does not learn the vote cast
- 1/2 probability of cheating by DRE
- DRE cannot ignore voter's input

Cons:

- No proof of cheating if voter hears different things in different ears
- Ballot indirection
- If a coercer programs Assistant not to follow the protocol, then the voter can be coerced

Conclusions

- Three new front-ends for polling place E2E systems
 - Do not use paper
- Originally invented with accessibility in mind
 - Can be used by all voters