# Pretty Good Democracy

Peter Y A Ryan

University of Luxembourg

Vanessa Teague

University of Melbourne

# Outline

- The challenge
- Pretty Good Democracy
- Threats
- Enhancements
- Conclusions

# Where is my Vote?

# “The Computer Ate my Vote”

- In the 2004 US presidential election, ~30% of the electorate used DRE, touch screen devices.
- Aside from the “thank you for your vote for Kerry, have a nice day” what assurance do they have that their vote will be accurately counted?
- What do you do if the vote recording and counting process is called into question?
- Voter Verifiable Paper Audit Trail (VVPAT) and “Mercuri method”. But paper trails are not infallible either.

# Remote vs Supervised

- Important to draw a clear distinction between supervised and remote voting.
- In the former the voter casts their vote in enforced isolation, e.g., in a booth in a polling station.
- Remote voting, e.g., internet, postal etc. such isolation cannot be enforced.
- Hence dangers of coercion.

# Code Voting

- Distribute code sheets to voters using another, secure channel, e.g. conventional post.
- Code sheets have random voting codes and acknowledgement codes for each candidate.
- In effect each voter is provided with a personal code book to communicate with the Vote Server.
- Sidesteps many of the insecurities of the web, client devices etc.

# Code sheet

| Candidate | Voting code | Acknowledgment code |
|---|---|---|
| Asterix | 4098 | 1385 |
| Idefix | 3990 | 3682 |
| Obelix | 6994 | 2904 |
| Panoramix | 2569 | 7453 |
| Serial number | 49950284926 | |

# Voting

- Voters logs onto the Vote Server, provides her code sheet id and the vote code for her candidate.
- VS responds with the correct ack code.
- Authenticates the VS and confirms receipt of the code.
- Sidesteps many insecurities of the internet and clients but doesn't provide end-to-end verifiability.

UNIVERSITÉ DU LUXEMBOURG

# Pretty Good Democracy

- Key ideas:
  - Access to the codes are shared amongst a set of Trustees.
  - Each code sheet carries just a single ack code.
- Thus, the Server has to pass on the correct vote code to a threshold set of the Trustees in order to return the correct ack code.
- Compatible with Prêt à Voter.

# Security properties

- Receiving the correct acknowledgement code gives assurance that the vote is correctly registered on the WBB (and hence will be correctly tabulated).
- Tabulation much as in Prêt à Voter.
- Do need trust assumptions: violation of secrecy of codes can violate accuracy.
- Receipt free due to single ack code per code sheet.
- Simple voter experience: vote, check, go….

# PGD Code sheet

| Candidate | Voting code |
|---|---|
| Asterix | 4098 |
| Idefix | 3990 |
| Obelix | 6994 |
| Panoramix | 2569 |
| Serial number | 49950284926 |
| Acknowledgement code | 4482094 |

# Cryptographic setup

- The Voting Authority generates a table in which each row contains the voting codes for one ballot, encrypted under the Trustees threshold key $PK_t$ .
- Table includes the ack codes encrypted under $PK_t$ .
- For each row, the encrypted vote codes are permuted with respect to the order shown on the code sheet.
- The permutations are encoded in Prêt à Voter style onions .

# The Voting Protocol

  - Voter → Server: i, VC_ij
  - Server → WBB: i, $\{VC\_ij\}_{PKt}$, ZKP(VC_ij)
- Trustees check the ZKP and perform a threshold PET of $\{VC\_ij\}_{PKt}$ against the terms of the appropriate row.
- If a term matches it is flagged and the trustees decrypt the ack code.
- The Vote Server can then return the ack code to the voter.

# Registering the vote

- PKZ and PETs posted to the WBB.
- Serves to counter attempts to alter votes or ballot stuffing etc.

# Distributed construction of code sheets

- A VA generates a set of $\lambda n(c+1)$ distinct codes.
- Where n is the size of the electorate  the and c number of candidates.
- $\lambda>1$ multiplier to allow for random audits.
- These are encrypted under the Trustees PK.
- Put through re-encryption mixes
- Assembled into a $\lambda n$ by c+1 table-P table.
- Note: generic construction.

# The P table

- The k-th row of the P table:
- $k \, , \{VC_{i1}\}_{PKT}, \{VC_{i2}\}_{PKT}, ........, \{VCi_{c}\}_{PKT}, \{Ack_{i}\}_{PKT}$

# Printing the code sheets

- Each row of the P table corresponds to a code sheet, the c+1 column is the ack code.
- A threshold set of trustees decrypt the rows and print the code sheets.
- This stage is critical.
- The Registrar distributes one code sheet to each eligible voter

# The Q Table

- An initial Clerk takes the P table and, for each row performs a re-encryption and shuffle of the first c entries.
- Information defining the shuffle in encrypted under the Tellers threshold key in an onion:

# Row permutations

$$K, \{VC_{i1}\}_{PKTr}, \{VC_{i2}\}_{PKTr}, \ldots\ldots\ldots, \{VC_{ic}\}_{PKTr}, \{Ack_i\}_{PKTr}$$

$$\rightarrow$$

$$K, \{VC_{i\pi i1(1)}\}_{PKTr}, \ldots\ldots\ldots, \{VCi_{\pi i1(c)}\}_{PKTr}, \{Acki\}_{PKTr, ,}\ \theta_{i1}$$

Where $\theta_{i1} = \{\pi_{i1}\}_{PKTe}$

# The Q Table

- Further k-1 shuffles performed:
- $\{VC_{i\pi ik(1)}\}_{PKTr},.........,\{VCi_{\pi ik(c)}\}_{PKTr}, \{Acki\}_{PKTr}, \theta_{ik}$
- The Q table in now posted to the WBB.
- Audits are performed on a randomly selected subset of the code sheets.
- Check for consistency with the corresponding rows of the Q table.

# Threats

- Leaking codes: threatens accuracy but also integrity.
- VS guessing codes.
- VS submits re-encryption of posted terms.
- Voters submitting fake codes.

# Recovery mechanisms

- Incorrect ack code.
- Voters should report and use alternate VS.
- Finalisation codes?

# Online distribution

- Dual channel distribution.
- Visual crypto.
- Add long term secret values.
- Decryption keys via snail mail-but the crypto constructs are tricky.
- Oblivious transfer style protocol.
- Spooky voting at a distance.

# Coercion resistance

- PGD not as it stands coercion resistant.
- Could add JCJ style tokens, but still tricky to see how best to update the WBB.

# Discussion

- Have the voter's client perform the encryptions of the ballot index and VC.
- But then need to trust the client, to some extent.
- Almost certainly not suitable for binding political elections.
- Perhaps ok for student elections, professional bodies, e.g. The IACR.

# Conclusions

- Fiendishly hard problem.
- Perhaps impossible without some residual trust.
- Not clear how to really solve the coercion problem.
- Need to figure out effective recovery mechanisms.
- Plenty of open questions.