Election Verifiability or Ballot Privacy Do We Need to Choose?

Edouard Cuvelier – Thomas Peters – Olivier Pereira

Université catholique de Louvain
ICTEAM – Crypto Group

SecVote 2012

Privacy and Verifiability

Privacy and Verifiability

19th century:

- increasing concerns about bribery and coercion
- secret ballots become mandatory in most countries
- and there are the troubles for correctness

Privacy and Verifiability

Abstract

Verifiable Secret-Ballot Elections

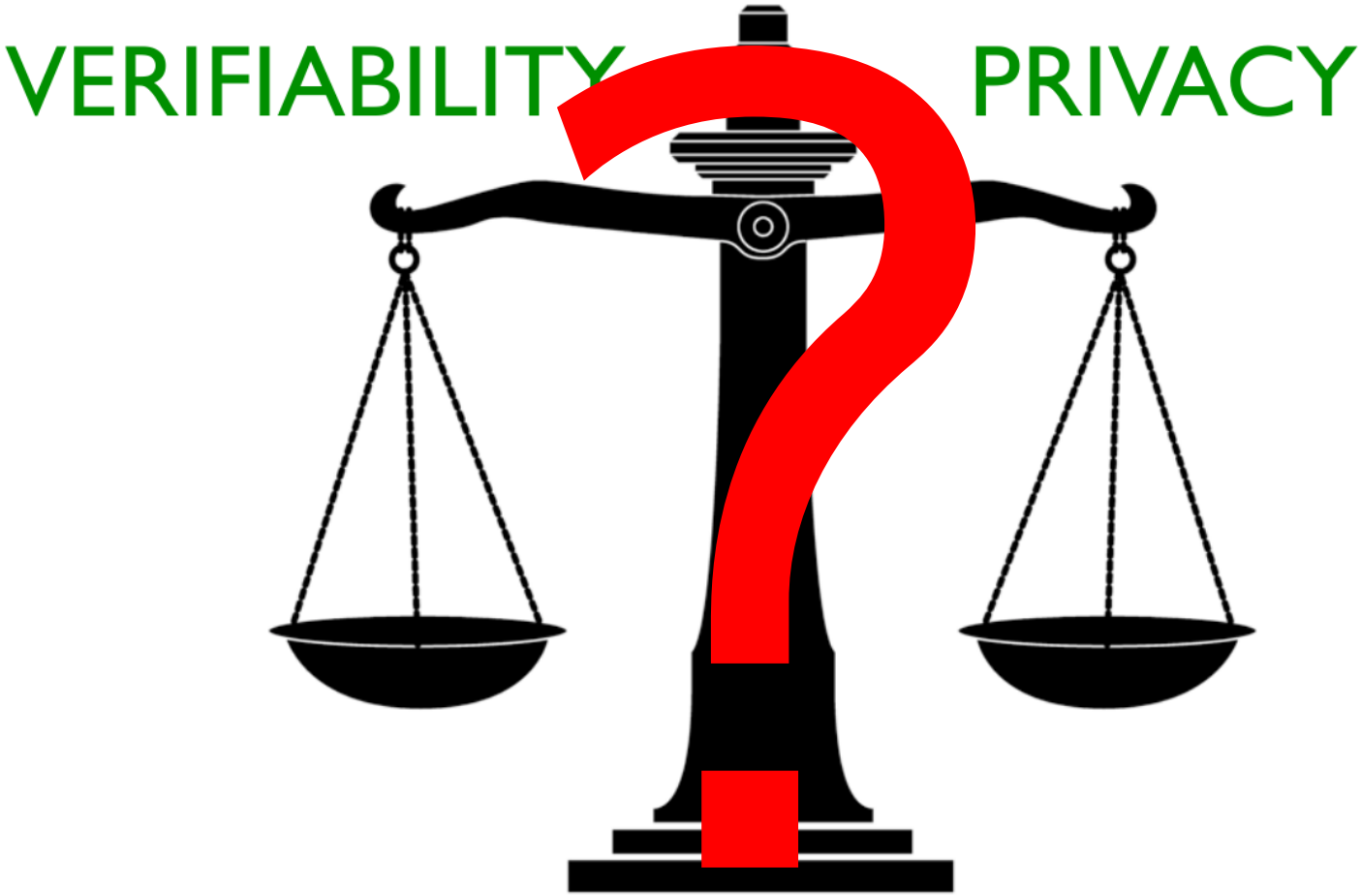
Josh Daniel Cohen Benaloh

Yale University

1996

Privacy in secret-ballot elections has traditionally been attained by using a ballot box or voting booth to disassociate voters from ballots. Although such a system might achieve privacy, there is often little confidence in the accuracy of the announced tally. This thesis describes a practical scheme for conducting secret-ballot elections in which the outcome of an election is verifiable by all participants and even by non-participating observers. All communications are

Privacy and Verifiability



Setting and Goals

- Large scale elections: single asynchronous pass by the voters
- Confidentiality rests on a set of trustees who perform the tally
- Offer verifiability without impacting privacy
- Solutions for both homomorphic and mixnet-based tallying
- Preserve optimal efficiency [CGS97]:

  (workload taken as $\|\text{modexp}\|$)
  - workload by voters independent of number of trustees
  - workload by voters logarithmic in number of choices
  - workload by trustees linear in number of ballots
  - ballot size linear in number of choices
  - workload independent of security parameter

Voting with Perfectly Private Audit Trail

Consider:

1. A private bulletin board
   - Used by authorities
   - Corresponds to the view in the non-verifiable system
   - Should offer usual computational privacy [BCPSW11]
2. A public bulletin board
   - Used for universal verifiability
   - Should offer perfect/statistical privacy
   - [BCPSW11] privacy with unbounded adversary

A New Primitive

Commitment Consistent (CC) Encryption:

- Regular (threshold) encryption
- \+ Extract_C that extracts a commitment from and on encrypted message (could formally just be the identity)
- \+ Extract_E that extracts an encryption of the opening of that commitment

"Naive" way of building this:

- Take Enc and Com schemes
- Gen uses Gen_E twice and Gen_C to get keys from these two schemes
- $\text{Enc}^{CC}(m)$ computes $(c, a) = \text{Com}_{ck}(m)$, $c_1 = \text{Enc}_{pk_1}(m)$ and $c_2 = \text{Enc}_{pk_2}(a)$ and outputs (c, c_1, c_2).

Application: have c perfectly hiding and use it for verifiability

A New Primitive

CC Encryption with **Validity Augmentation** (CCVA):

- For privacy:
  Augmentation that makes the scheme NM-CPA
- For accountability:
  Augmentation that convinces the trustees that the output of Extract_E really makes it possible to open Extract_C

Summing Up the Process

CCEnc2Vote(Π) works as follows from CCVA scheme Π

- Generate public key of Π and publish it
- Voters submit $e_i = \mathsf{Enc}_\Pi(v_i)$
- Authorities verify the augmentations and publish $c_i = \mathsf{Extract}_C(e_i)$

For homomorphic tallying:

- Authorities publish an opening of $\prod c_i$
- Verifiability follows from the binding property of Com

For mixnet-based tallying:

- Authorities publish openings of verifiably shuffled c_i (using a statistical ZK proof)
- Verifiability follows from the binding property of Com

Privacy and Verifiability

Privacy:

- The BB contains perfectly hiding commitments
  this satisfies an IT version of ballot privacy definition
- The BB contains opening of the election outcome
  an unbounded adversary can derive this opening from the outcome
- The BB may contain extra proofs
  this does not give more as long as they are statistical ZK

Universal Verifiability:

- Offered by computational binding property of commitments
- And soundness of ZK proofs

How to make this work?

Based on ElGamal and Pedersen?

- Commitment $g^v h^r$ and ciphertext $(g^s, h^r y^s)$?
  But r is full size, so we cannot extract DL
- Commitment $g^v h^r$ and ciphertext $(g^s, "r" y^s)$?
  But not additively homomorphic and seems to require cut-and-choose validity proofs

Based on Paillier and Pedersen?

- Commitment $g^v h^r$ and ciphertext $(1+N)^r s^N$? [MN07]
  But :
  - Paillier distributed key generation extremely challenging (needs $N = pq$ with unknown primes p and q)
  - Paillier works $\bmod N^2$ which can be too expensive
- Still, we proved that it is secure for our generic construction

CC encryption for Homomorphic Tallying

Use EC groups with asymmetric pairing $e : G_1 \times G_2 \to G_T$ with DDH assumption on G_1 and G_2 (e.g., BN or BLS curves)

The PPAT1 scheme:

- Public key: random g, g_1 generating G_1, h, h_1 generating G_2
  Private key: $x_1 : g_1 = g^{x_1}$
- $\mathrm{Enc}(v) := (c_0, c_1, c_2) = (g^s, g^r g_1^s, h^r h_1^v)$
- $\mathrm{Extract}_C(c_0, c_1, c_2) := c_2$
- $\mathrm{Dec}(c_0, c_1, c_2) :=$ DL of $e(c_0^{x_1} c_1^{-1}, h) \cdot e(g, c_2)$ in basis $e(g, h_1)$
- The opening of c_2 is g^r – verification: $e(g^r, h) \stackrel{?}{=} e(g, c_2/h_1^v)$

Observations:

- This scheme is homomorphic and IND-CPA under DDH
- VA can be made from usual sigma protocols
- Looks like Pedersen, but actually quite different

CC encryption for Mixnet-based Tallying

PPAT1 scheme requires DL extraction in decryption
Mixnets only require reencryption possibility

The PPAT2 scheme:

- Public key: random g, g_1, g_2 generating G_1, h, h_1 generating G_2
  Private key: $x_1 : g_1 = g^{x_1}$ and $x_2 : g_2 = g^{x_2}$
- $\mathrm{Enc}(v) := (a_1, a_2, b, c_1, c_2) = (g^{r_1}, g^{r_2}, g_1^r g_2^{r_2}, v g_1^{r_1}, h^r h_1^{r_1})$
- $\mathrm{Extract}_C(a_1, a_1, b, c_1, c_2) := (c_1, c_2)$
- $\mathrm{Dec}(c_0, c_1, c_2) := c_1 / a_1^{x_1}$
- The opening of (c_1, c_2) is g_1^r $(e(g, c_2) \stackrel{?}{=} e(g_1^r, h) e(c_1/v, h_1))$

Observations:

- Same remarks for IND-CPA and VA
- Homomorphic for EC point addition (but we do not care)
- Looks like Pedersen/PPAT1, but again fairly different

Efficiency Comparisons

Assuming:

- 256 bit multiplication costs 1
- multiplication has quadratic complexity
- exponentiation/point multiplication by square and multiply

Cost of 1 encryption ($+$ 0/1 proof for PPAT1)

Scheme	$\mathbb{Z}_p^*$	$\mathbb{Z}_{N^2}^*$	$\mathbb{G}_1$	$\mathbb{G}_2$	Total Cost
Pedersen/Paillier	4	10	0	0	8.650.752
PPAT1	0	0	6	6	115.200
PPAT2	0	0	9	4	96.000

Implementation estimates for JavaScript implementation:

- Standard techniques provide a PPAT2 ciphertext in $<$ 1s
- Ongoing implementation expected to improve this by ≈ 20

Conclusions

We provide a model and tools for building universally verifiable voting systems with a perfectly private audit trail:

- Our CCVA schemes make it possible to get a perfectly private audit trail efficiently
- Can be plugged into most voting systems based on homomorphic encryption, inherit the properties of those systems + PPAT
- Standard "sigma" ZK protocols can be used for validity proofs and mixing