Ballot privacy in elections: new metrics and constructions.

Olivier Pereira – Université catholique de Louvain

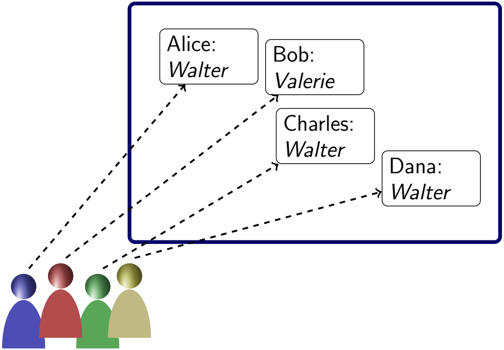
Based on joint works with:
D. Bernhard, V. Cortier, E. Cuvelier,
T. Peters and B. Warinschi

March 2015

Open Voting

Open Voting

Open Voting

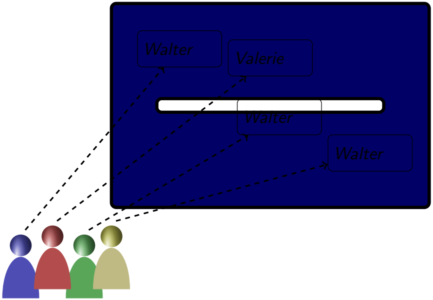


- Every voter can verify that nobody tampered with her/his vote
- Every voter can compute the tally
- No privacy, no coercion-resistance, no fairness, ...

Secret Ballot

- Liberal motivation: "My vote is my own business, elections are a tool for aggregating private opinions"
- Practical motivation: Prevent coercion and bribery

A traditional paper approach



- With voting booth: privacy, coercion-resistance, fairness, ...
- **If** a voter keeps an eye on the urn and tally all day long, he can be convinced that:
  - his vote is untampered
  - the tally is based on valid votes and correct
- A minute of inattention is enough to break this

Privacy vs Verifiability – Two Extremes

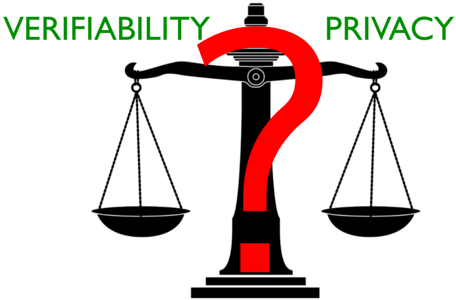
Hand raising vote

Verifiability 100%
Privacy 0%

Uncontrolled ballot box

Verifiablility 0%
Privacy 100%

Privacy and Verifiability



Defining Vote Privacy

Not an absolute notion:

- Usually accepted that there is no privacy when all voters support the same candidate

Elections as Secure Function Evaluation [Yao82]:

- "The voting system should not leak more than the outcome"
- But we would like to know how much the outcome leaks!

Game-style definition [KTV11]:

- Privacy measured as max probability to distinguish whether I voted in one way or another
- Often too strong: that probability is ≈ 1 when:

$$\#\text{different ballots} \gg \#\text{voters}$$

Defining Vote Privacy

What do we want to measure?

1. With what probability can $\mathcal{A}$ guess my vote?
   Sounds like min-entropy!
2. In how many ways can I pretend that I voted?
   Sounds like Hartley entropy!

Notations

Let:

- $\mathcal{D}$ be the distribution of honest votes (if known)
- $T : \sup(\mathcal{D}) \mapsto \{0,1\}^*$ be a target function
  - $T(v_1, \ldots, v_n) := v_i$
  - $T(v_1, \ldots, v_n) := (v_i \stackrel{?}{=} v_j)$
- $\rho(v_1, \ldots, v_n)$ be the official outcome of the election
- $\mathrm{view}_{\mathcal{A}}(\mathcal{D}, \pi)$ be the view of $\mathcal{A}$ participating to voting protocol π in which honest voters vote according to $\mathcal{D}$

Measure(s) for privacy

$$\mathsf{M}_x(T, \mathcal{D}, \pi) := \inf_{\mathcal{A}} \mathsf{F}_x(T(\mathcal{D}) | \mathsf{view}_{\mathcal{A}}(\mathcal{D}, \pi), \rho(\mathcal{D}, v_{\mathcal{A}}))$$

where:

- $\mathsf{F}_x(A|B)$ is some x-Réniy entropy measure on A given B

Choices for $\mathsf{F}_x(A|B)$

$$\mathsf{M}_x(T,\mathcal{D},\pi) := \inf_{\mathcal{A}} \mathsf{F}_x(T(\mathcal{D})|\mathsf{view}_{\mathcal{A}}(\mathcal{D},\pi),\rho(\mathcal{D},v_{\mathcal{A}}))$$

Choices for $\mathsf{F}_x(A|B)$:

$\tilde{\mathsf{H}}_\infty$ Average min-entropy: $-\log\left(\mathop{\mathbb{E}}_{b\in B}\left[2^{-\mathsf{H}_\infty(A|B=b)}\right]\right)$ [DORS08]
Measures the probability that $\mathcal{A}$ guesses the target

$\mathsf{H}_\infty^\perp$ Min-min-entropy: $\min_{b\in B} \mathsf{H}_\infty(A|B=b)$
Same as before, but for the worst possible b

$\mathsf{H}_0^\perp$ Min-Hartley-entropy: $\min_{b\in B} \mathsf{H}_0(A|B=b)$
Measures the number of values that the target can take for the worst b – No probabilities involved!

An example…

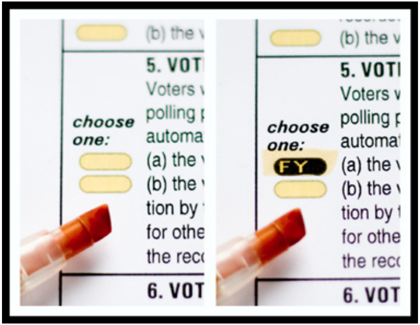
Consider:

- An approval (yes/no) election with 1 question
- 3 voters voting uniformly at random
- target is the first voter

	$\tilde{H}_\infty$	$H^\perp_\infty$	$H^\perp_0$
$\rho_1 := \perp$	1	1	1
$\rho_2 := \lvert\vec{v}\rvert_{\text{yes}} > \lvert\vec{v}\rvert_{\text{no}}$	.4	.4	1
$\rho_3 := (\lvert\vec{v}\rvert_{\text{yes}}, \lvert\vec{v}\rvert_{\text{no}})$	.4	0	0
$\rho_4 := \vec{v}$	0	0	0

$(.4 \approx -\log \frac{3}{4})$

Scantegrity Audit Data



- Official outcome: number of votes received by each candidate
- Scantegrity audit trail exposes all ballots (codes removed)
- Scantegrity take-home receipt shows how many bullets you filled

Scantegrity Audit Data

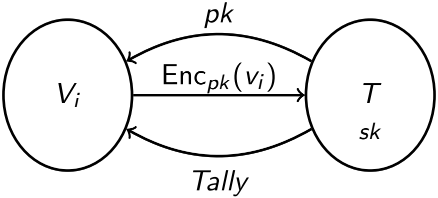
From the 2009 Takoma Park municipal election data :

Ward	1		5		6	
#Ballots	470		85		198	
Question	A	B	A	B	A	B
$H_0^{\perp}$ from official outcome	6	3.17	6	3.17	6	6
$H_0^{\perp}$ with receipts	1.58	1.58	0	1	2	1.58

- 6/3.17 bits is a question with 3/2 candidates to rank (including incorrect rankings)
- In most cases, rankings of a certain length are uncommon
- In Ward 5, a voter looses his/her privacy completely on Question A if he/she shows his/her receipt!

Single-Pass Cryptographic Voting

A common approach ([CGS97], [DJ01], Helios, . . .):



1. Trustees create an election public key pk
2. Voters publish an encryption of their vote v_i
3. Trustees compute and publish the tally, using the secret key sk
4. Everyone can verify that the tally is consistent with the encrypted votes

Cryptographic Voting

Problem with entropic measures of privacy:

$$\mathsf{H}(v_i|\mathsf{Enc}_{pk}(v_i), pk) = 0$$

Solution: use a computational analog of entropy :

- $\mathsf{F}^c_x(A|B) \geq r \Leftrightarrow \exists B' \approx^c B$ and $\mathsf{F}_x(A|B') \geq r$

In particular,

$$\mathsf{H}^c(v_i|\mathsf{Enc}_{pk}(v_i), pk) \geq r \quad \text{if} \quad \mathsf{H}(v_i|\mathsf{Enc}_{pk}(0), pk) \geq r$$

Computational Measure(s) for privacy

$$\mathsf{M}^c_x(T, \mathcal{D}, \pi) := \inf_{\mathcal{A}} \mathsf{F}^c_x(T(\mathcal{D}) | \mathsf{view}_{\mathcal{A}}(\mathcal{D}, \pi), \rho(\mathcal{D}, v_{\mathcal{A}}))$$

where:

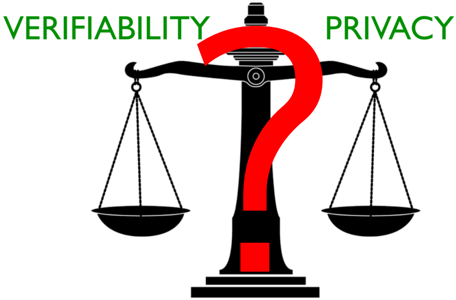
- $\mathsf{F}^c_x(A|B)$ is a x-Réniy computational entropy metric on A given B

Definition (informal): A voting scheme π with tallying function ρ offers *ballot privacy* if, for all T, $\mathcal{D}$:

$$\mathsf{M}^c_x(T, \mathcal{D}, \pi) = \inf_{\mathcal{A}} \mathsf{F}^c_x(T(\mathcal{D}) | \rho(\mathcal{D}, v_{\mathcal{A}}))$$

Privacy and Verifiability

Do we *need* to move to computational entropies?



- Publish encrypted votes, but what if encryption gets broken?
  - because time passes and computing speed increases
  - because decryption keys are lost/stolen
  - because there is an algorithmic breakthrough

Voting with a Perfectly Private Audit Trail

Can we offer verifiability without impacting privacy?

More precisely:
Can we take a non-verifiable voting scheme and add verifiability without impacting privacy?

Goal:

- Have a new kind of audit data
- Audit data must perfectly hide the votes
- Usability must be preserved:
  1. Practical distributed key generation
  2. No substantial increase of the cost of ballot preparation
  3. Be compatible with efficient proof systems

Commitments Can Enable Perfect Privacy

- A commitment is *perfectly hiding* if d is independent of m
- A commitment is *computationally binding* if it is *infeasible* to produce $d, (m, a), (m', a')$ such that d can be opened on both (m, a) and (m', a') $(m \neq m')$

Example:

- Let g_0, g_1 be random generators of a cyclic group $\mathbb{G}$
- Set $d = g_0^a g_1^m$ as a commitment on m with random opening a
- Finding a different (m, a) pair consistent with d is as hard as computing the discrete log of g_1 in base g_0

A New Primitive : Commitment Consistent Encryption

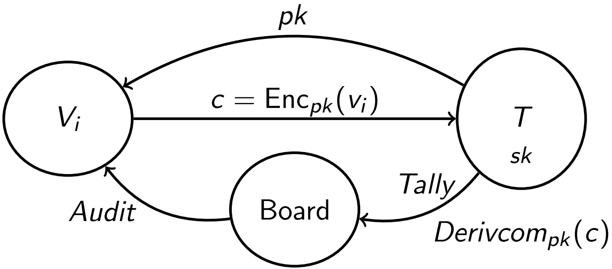
Commitment Consistent Encryption (CCE) scheme
$\Pi = (\mathit{Gen}, \mathit{Enc}, \mathit{Dec}, \mathit{DerivCom}, \mathit{Open}, \mathit{Verify})$

$(\mathit{Gen}, \mathit{Enc}, \mathit{Dec})$ is a classic encryption scheme
$c = \mathit{Enc}_{pk}(m)$

$\mathit{DerivCom}_{pk}(c)$	from the ciphertext, derives a commitment d
$\mathit{Open}_{sk}(c)$	outputs an opening value a from c using sk
$\mathit{Verify}_{pk}(d, a, m)$	checks that d is a commitment on m w.r.t. a

Single-Pass Cryptographic Voting

Voting with a CCE scheme:



1. Trustees create an election public key pk
2. Voters submit an encryption of their vote v_i to Trustees
3. Trustees publish commitments extracted from encrypted votes
4. Trustees publish the tally, as well a proofs of correctness

Voting with a Perfectly Private Audit Trail

If:

- Commitments are perfectly hiding
- Proofs are perfect/statistical zero-knowledge

Then:

- the audit trail is independent of the votes

$$\Rightarrow \mathbf{H_x(votes \mid audit\ trail + tally) = H_x(votes \mid tally)}$$

If cryptographic assumptions are broken:

- Someone might be able to "prove" a wrong result

But:

- Proof needs to be produced fast enough to be compelling
- Only people who believe in crypto assumption will trust the proof

Building CC Encryption Schemes

Group setup:

$\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ different groups of same prime order

A bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$

$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_T$
g	h	$e(g, h)$
g^a	h	$e(g^a, h) = e(g, h)^a$
g	h^b	$e(g, h^b) = e(g, h)^b$

DDH problem expected to be hard in $\mathbb{G}_1$ and $\mathbb{G}_2$

The PPATS Scheme

Additively homomorphic scheme for small message $m \in \mathbb{Z}_q$

$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_T$
$g, g_1 = g^{x_1}$	h, h_1	
$c_1 = g^s$	$d = h^r h_1^m$	
$c_2 = g^r g_1^s$		$Dec_{sk}(c)$: *DLog* of $e(c_1^{x_1}/c_2, h) \cdot e(g, d)$ $= e(g, h_1)^m$
$Open_{sk}(c)$: $a = c_2/c_1^{x_1}$		
		$Verif_{pk}(d, m, a)$: $e(a, h) \stackrel{?}{=} e(g, d/h_1^m)$

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)

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
PPATS	0	0	6	6	115.200

+ PPATS has considerably simpler threshold variants, thanks to the public order groups

Conclusions: Privacy and Verifiability

Two apparently conflicting requirements on votes:

Hiding for privacy $\leftrightarrow$ Showing for verifiability

Commitment-consistent encryption can reconcile these goals!

Experiences and metrics are useful: the outcome of an election can, in itself, give more information than expected, as voters vote highly non uniformly!