

Receipt-Freeness in Electronic Voting

Martin Hirt

ETH Zurich

Security Requirements

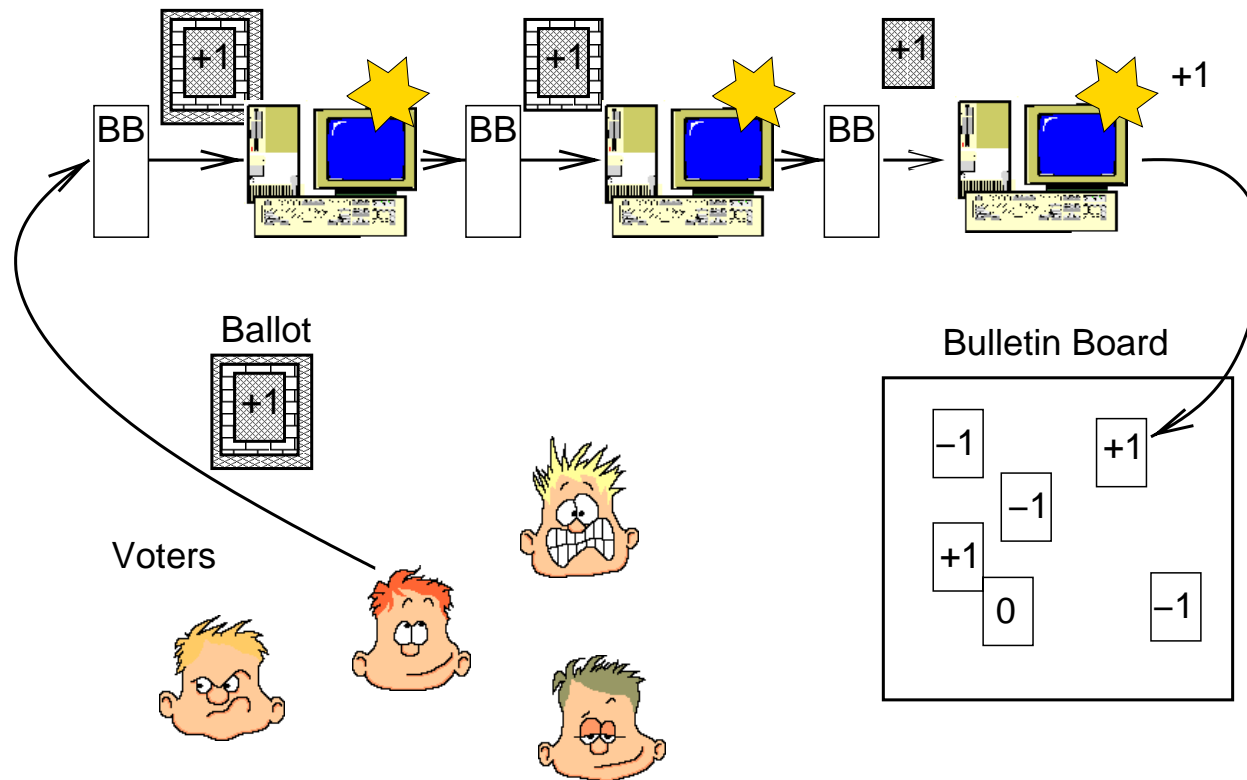
Correctness

- **Validity of ballots** (in {yes/no}, entitled voter, ≤ 1 ballots)
- **Tallying** (correct and verifiable sum)

Privacy

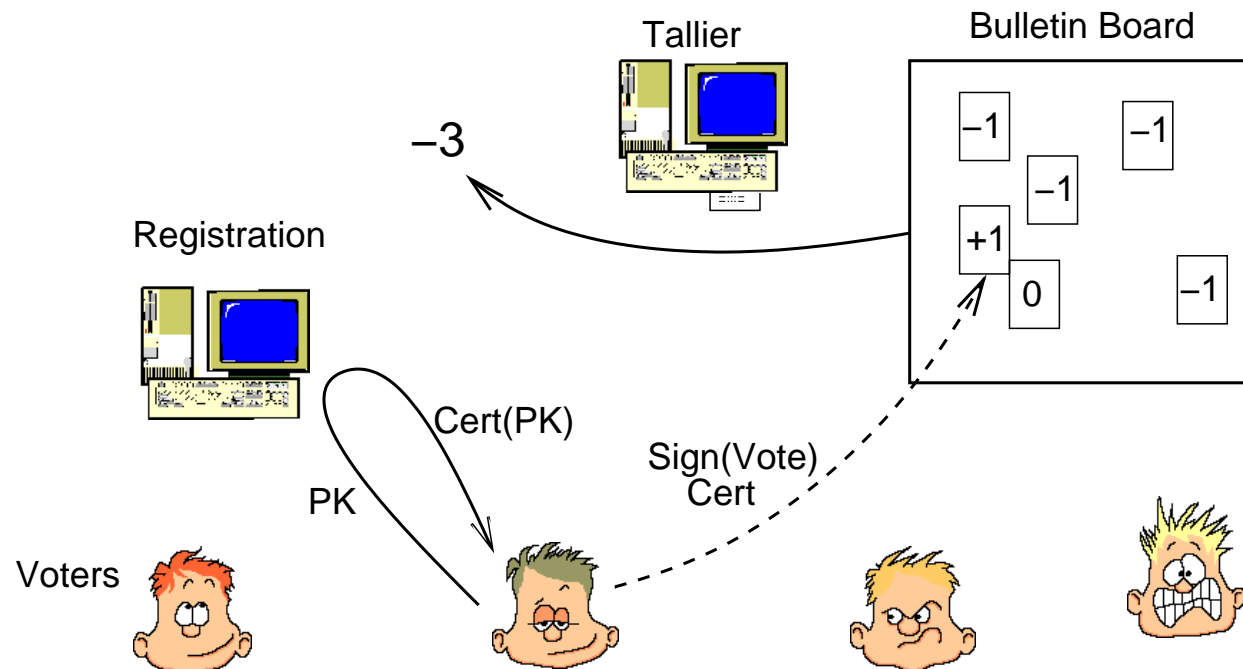
- **Secrecy** (cannot determine voter's vote)
- **Anonymity** (who casts a vote?)
- **Independence** (no partial results)

Voting Schemes based on Mix-Nets



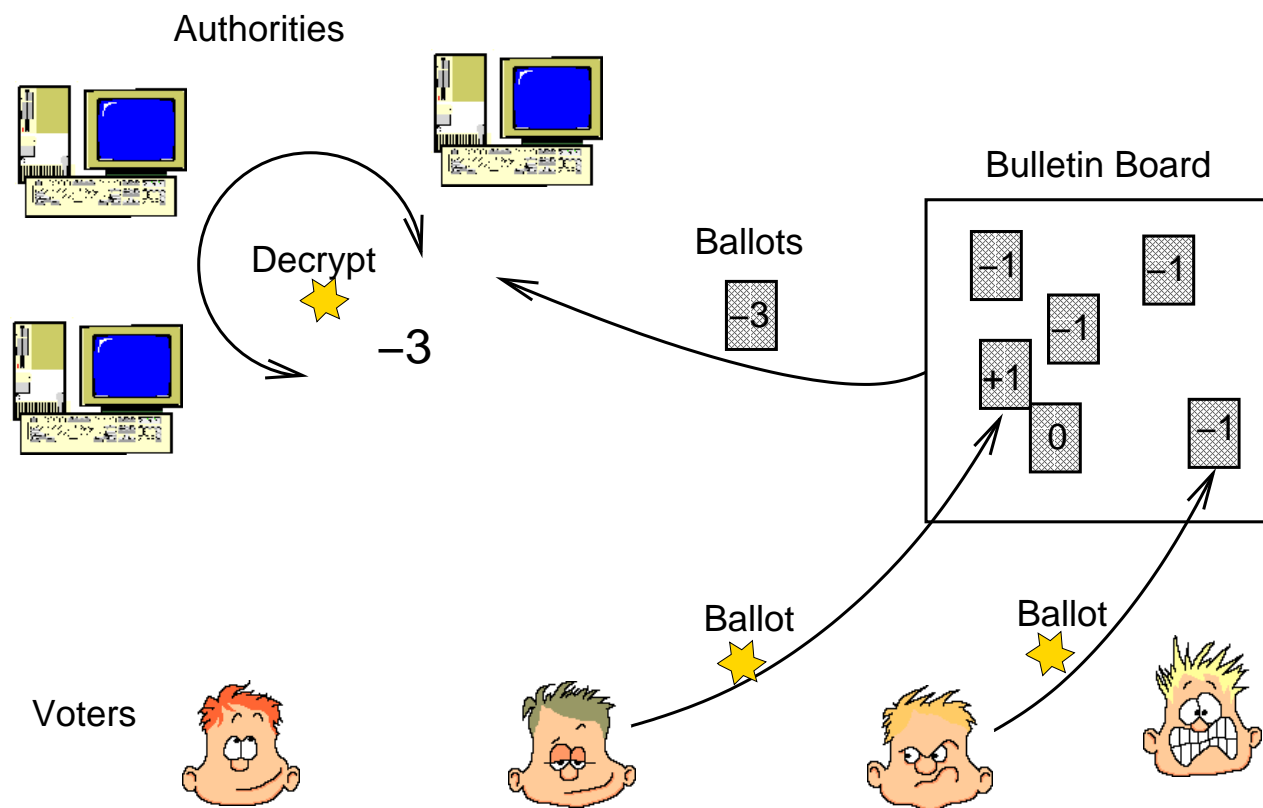
- voters encrypt vote with each authorities' public key
- authorities consecutively decrypt (and prove)

Voting Schemes based on Blind Signatures



- voters get blind signature on PK
- cast authorized vote through anonymous channel
- tallying is public

Voting Schemes based on Homomorphic Encryption

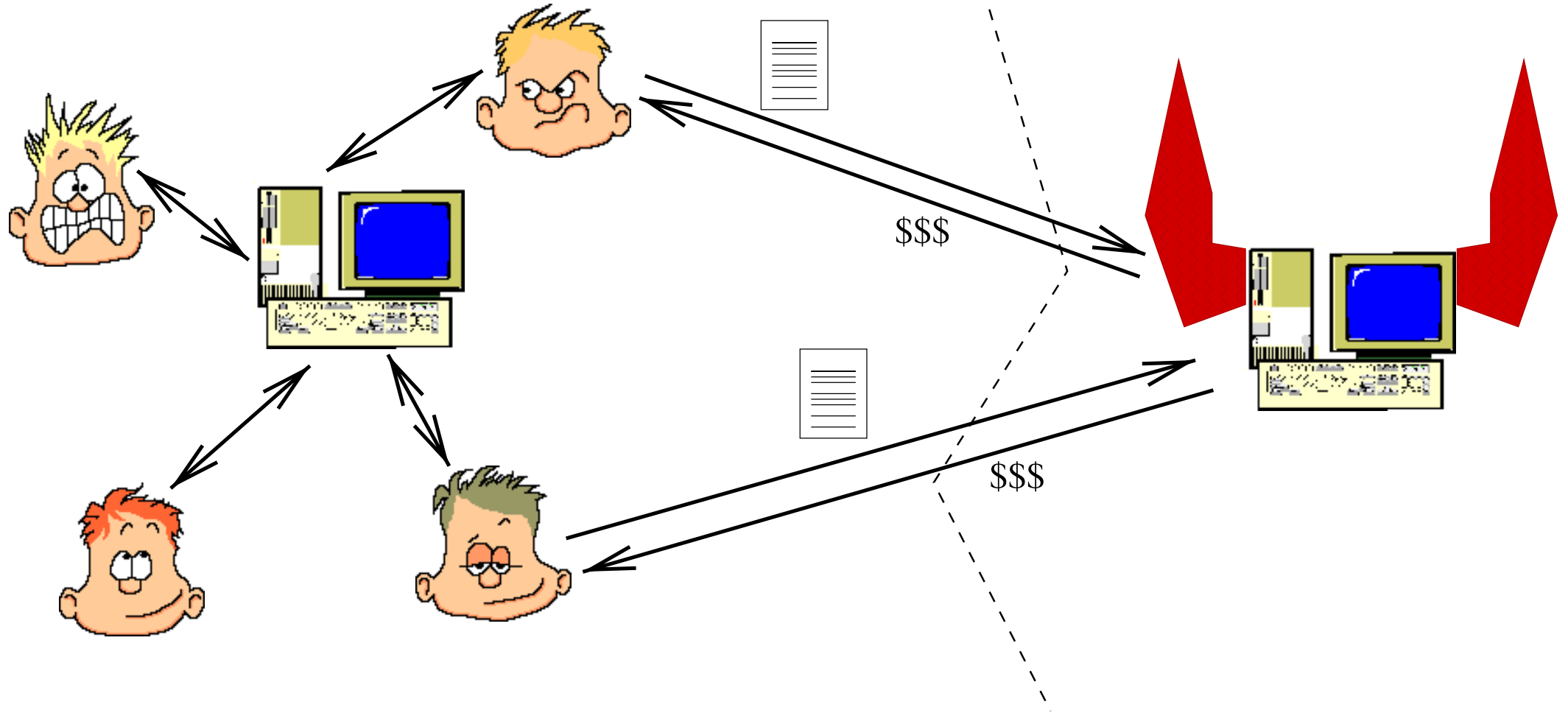


- several authorities, honest quorum
- homomorphic public-key encryption
- decryption function shared among authorities

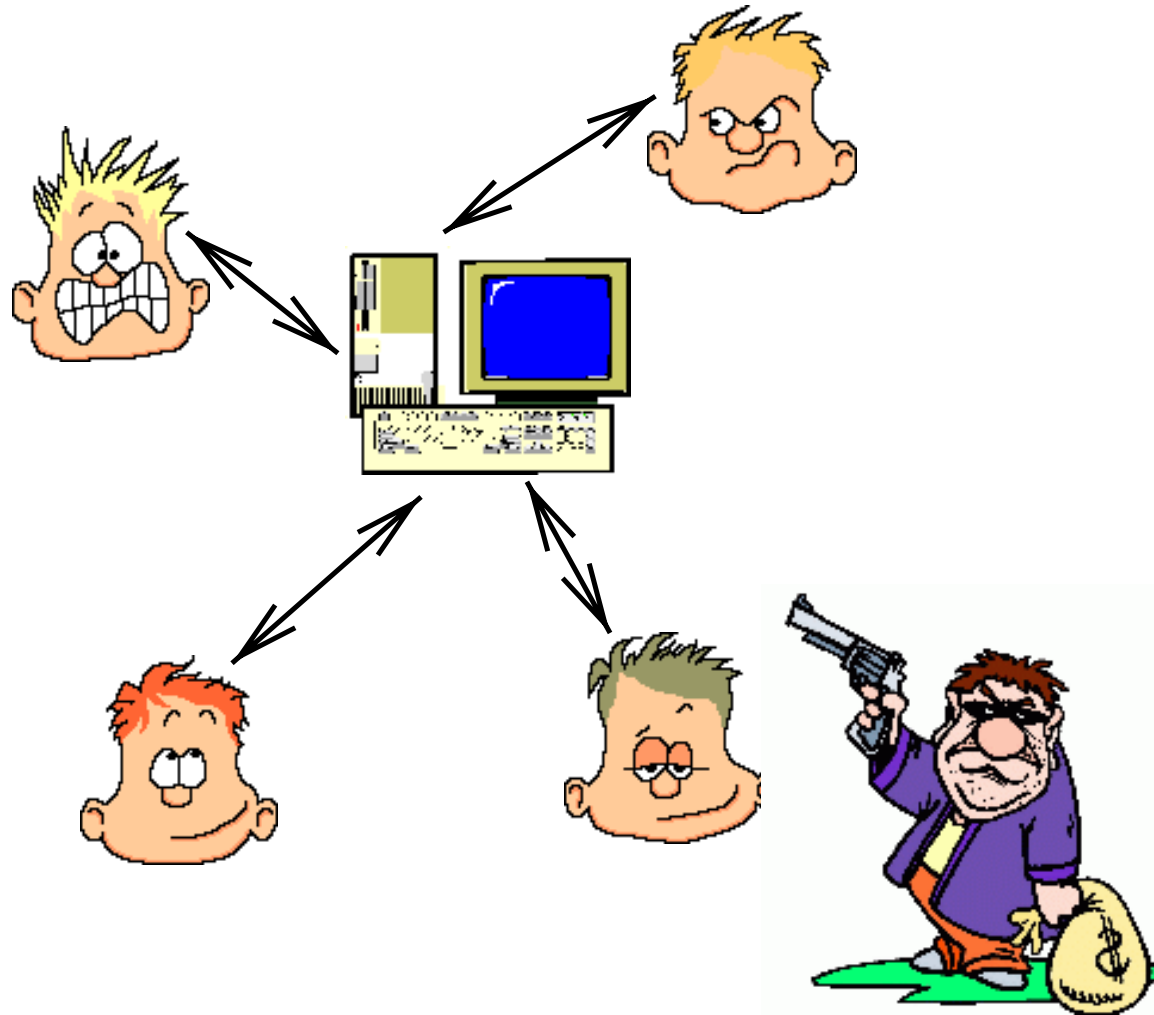
Comparison

	homomorphic encryption	blind signatures	Mix-Net
Mathematical structure	much	no	medium
Voter-interactivities	1 round	> 1 rounds	\geq 1 rounds
Incremental tallying	yes	no	no
Verifiability	yes	??	??
Costs			
voting phase	high	small	medium
tallying phase	small	very small	medium
verification phase	small	local only	high

Vote-Buying



Coercion



Receipt-Freeness

New Requirement

- **Secrecy**: voter **can** keep vote secret
- **Receipt-freeness**: voter **must** keep vote secret

Remarks

- Captures both **vote-buying** and **coercion**
- Impossible for “wild candidates”
- **Impossible in the standard model**

New Assumptions

- **Voting booth**
- **Untappable channels** (many flavors)
- **Erasures** (voter partially honest)
- Others?

Coercion vs Vote-Buying

Receipt-freeness: Voter can construct **false receipts**.

Scenario: Voter can lie, is **caught** with probability ε .

	$\varepsilon \approx 0$	$\varepsilon \approx 0.5$	$\varepsilon \approx 1$
Vote-Buying	no money	no money	money
Coercion	free	coerced	coerced

Receipt-Freeness Definitions I

Trustworthyness of Voter

- (1) honest during protocol execution
- (2) does not delete
- (3) malicious

Interaction with Vote-buyer

- (1) after vote
- (2) before and after vote
- (3) interactively

Receipt-Freeness Definitions II

Reliability of Receipt (when is a string a receipt?)

(1) obeys $\Rightarrow p \approx 1$, lies $\Rightarrow p \approx 0$.

(2) obeys $\Rightarrow p \approx 1$, lies $\Rightarrow p \ll 1$.

Cooperation with Authorities

(1) any authorities cooperate with buyer

(2) known authorities cooperate with buyer

(3) no authority cooperates with buyer

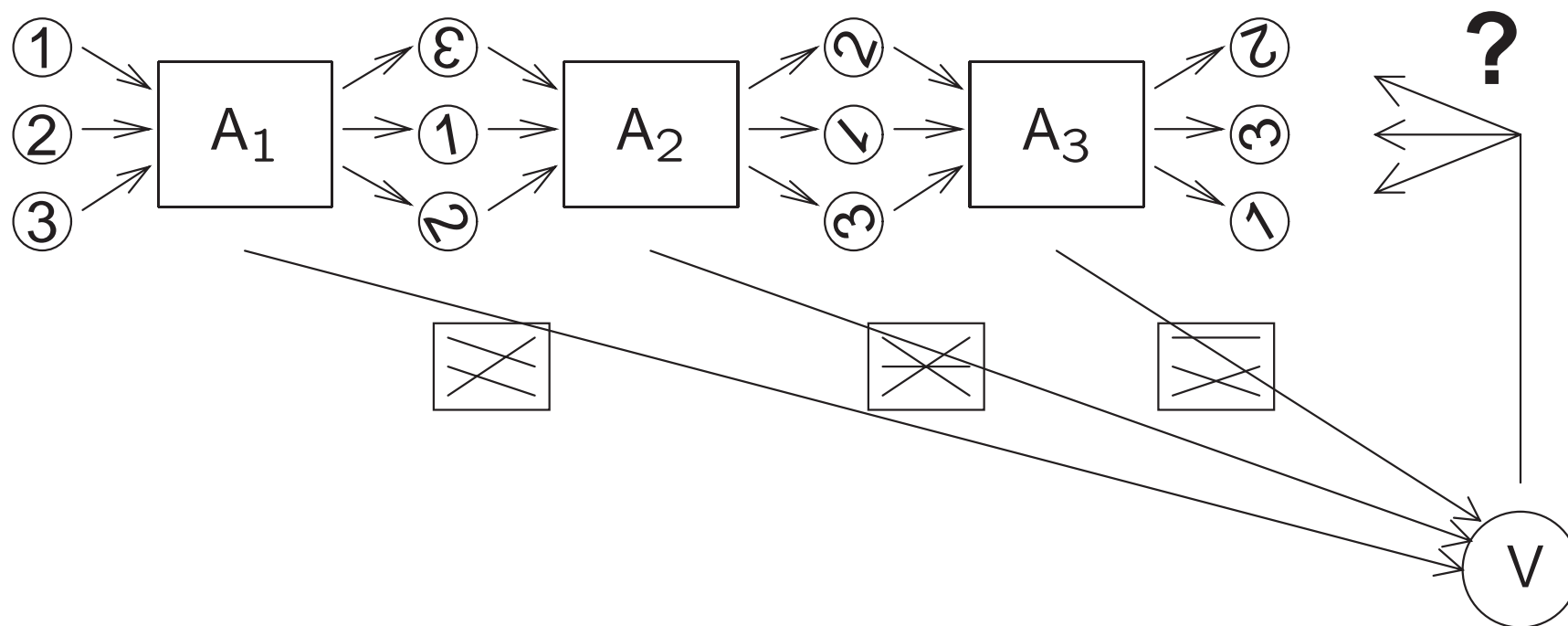
Literature

- Chaum 81 (mixer)
- Park/Itoh/Kurosawa 91 (mixer)
- Fujioka/Okamoto/Ohta 92 (blind signatures)
- X Benaloh/Tuinstra 94 (homomorphic encryption, receipt-free)
- Sako/Kilian 95 (mixer, receipt-free)
- Cramer/Franklin/Schoenmakers/Yung 96 (homo. encryption)
- X Okamoto 96 (blind signatures, receipt-free)
- Cramer/Gennaro/Schoenmakers 97 (homomorphic encryption)
- Okamoto 97 (blind signatures, receipt-free)
- Hirt/Sako 00 (homomorphic encryption, receipt-free)
- X Lee/Kim 00 (homomorphic encryption, receipt-free)

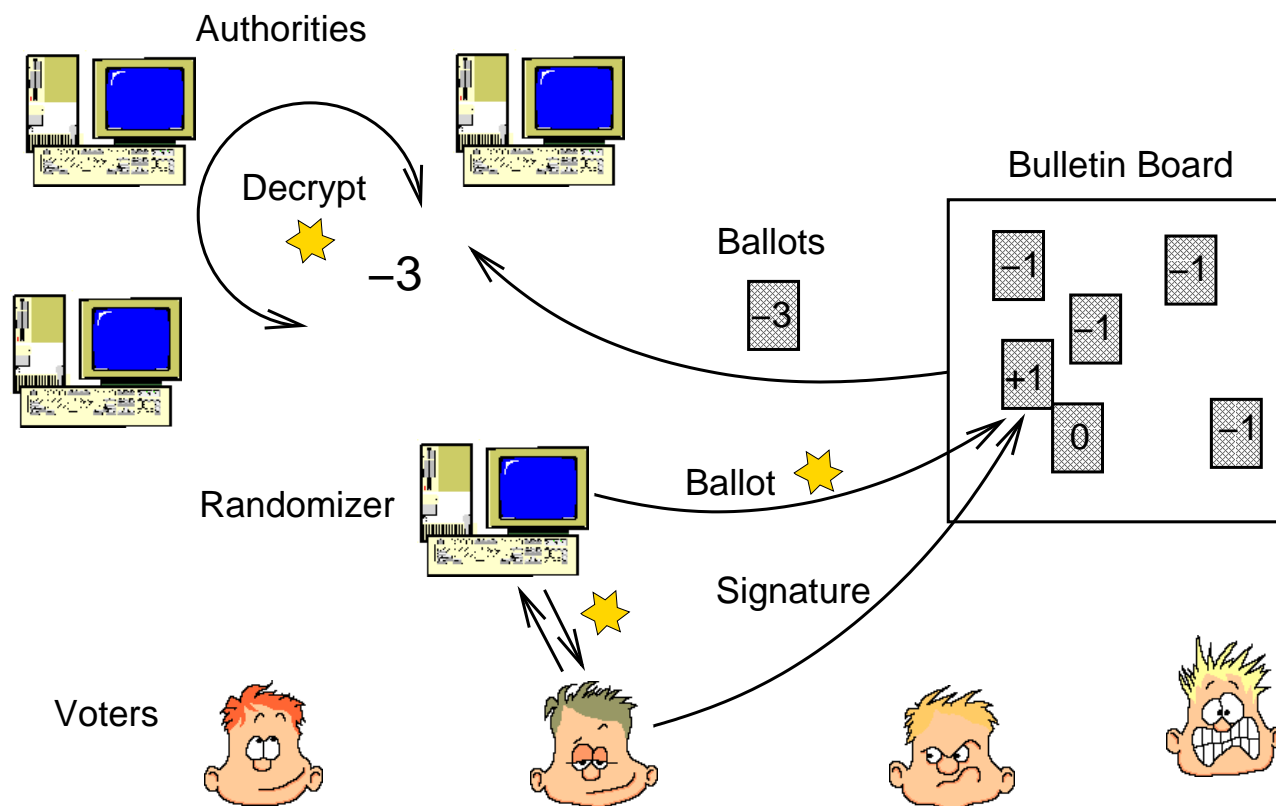
Receipt-Free Vote Generation [HS00]

Idea: **Authorities generate ballots** for each vote. Voter **points** to vote.

E.g.: $\mathcal{V} = \{1, 2, 3\}$, authorities A_1, A_2, A_3 .



Receipt-Free Vote Generation with Randomizers



- randomizer re-encrypts and proves ballot: **randomization proof**
- voter and randomizer **jointly** generate **validity proof**
- **several randomizers** help avoiding denial-of-service attacks