Themis: An On-Site Voting System with Systematic Cast-as-intended Verification an Partial Accountability

*Mikaël Bougon[1], Hervé Chabanne[1], Véronique Cortier[2], **Alexandre Debant[2]**, Emmanuelle Dottax[1], Jannik Dreier[2], Pierrick Gaudry[2], Mathieu Turuani[2]*

[1] IDEMIA, France

[2] Université de Lorraine, CNRS, Inria, LORIA, Nancy, France

CCS Conference
Los Angeles, November 10th 2022

On-site e-voting

Main goal: enhance the trust compared to pure paper-based voting

Security targets:

- **Vote secrecy:** no-one can know who I voted for
- **Verifiability:** no-one can modify the result of the election

On-site e-voting

Main goal: enhance the trust compared to pure paper-based voting

Security targets:

- **Vote secrecy:** no-one can know who I voted for
- **Verifiability:** no-one can modify the result of the election

voting machine can be compromised

On-site e-voting

Main goal: enhance the trust compared to pure paper-based voting

Security targets:

- **Vote secrecy:** no-one can know who I voted for
- **Verifiability:** no-one can modify the result of the election

voting machine can be compromised

New requirements in IDEMIA's use context

- limited access to the technology (the Internet, printers, etc)
- require a high level of robustness
- must cope with strained contexts (risks of corruptions, false accusations, etc)

Themis

Limited access to technology

- use pre-printed paper ballots ➡ do not need printers
- use smart cards and voting machines ➡ given by the service provider
- use a hash-chain to ensure the integrity of the electronic ballot-box ➡ can be monitored offline a posteriori

Themis

Limited access to technology

- use pre-printed paper ballots ➡ do not need printers
- use smart cards and voting machines ➡ given by the service provider
- use a hash-chain to ensure the integrity of the electronic ballot-box ➡ can be monitored offline a posteriori

Require a high level of security and robustness

- verifiability (with cast-as-intended) and vote secrecy
- can always return to a pure paper-based voting system with the same guarantees

Themis

Limited access to technology

- use pre-printed paper balloots ➡ do not need printers
- use smart cards and voting machines ➡ given by the service provider
- use a hash-chain to ensure the integrity of the electronic ballot-box
  ➡ can be monitored offline a posteriori

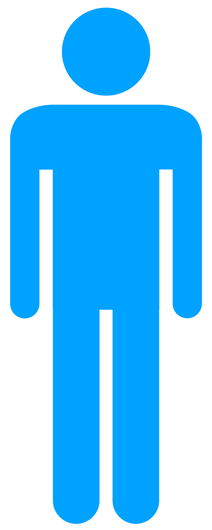
Require a high level of security and robustness

- verifiability (with cast-as-intended) and vote secrecy
- can always return to a pure paper-based voting system with the same guarantees

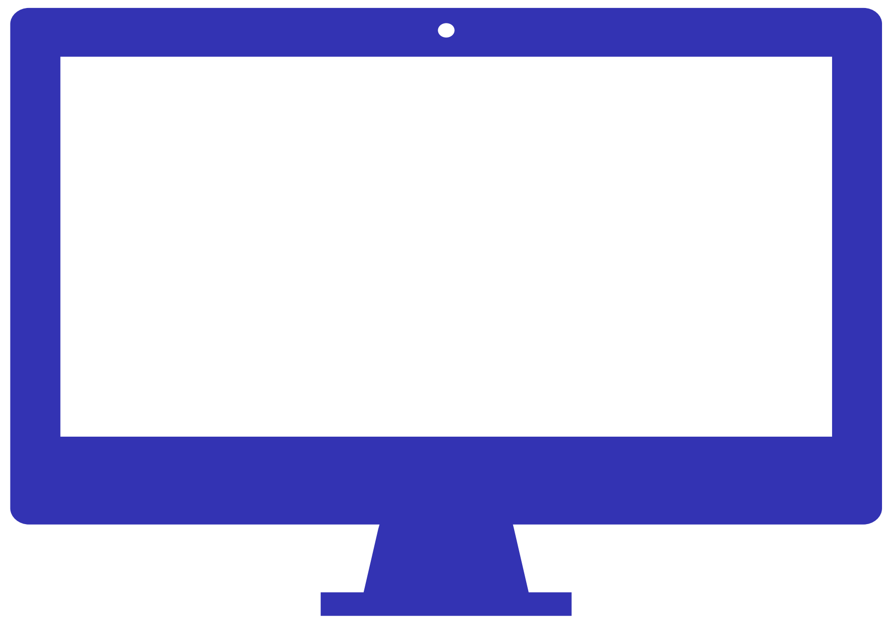
Strained contexts

- implement a dispute resolution procedure to decide who is the culprit
  ➡ proven to never wrongly blame someone
- require the corruption of several authorities to defeat vote secrecy of verifiability
  ➡ proven in symbolic models

Overview of the system

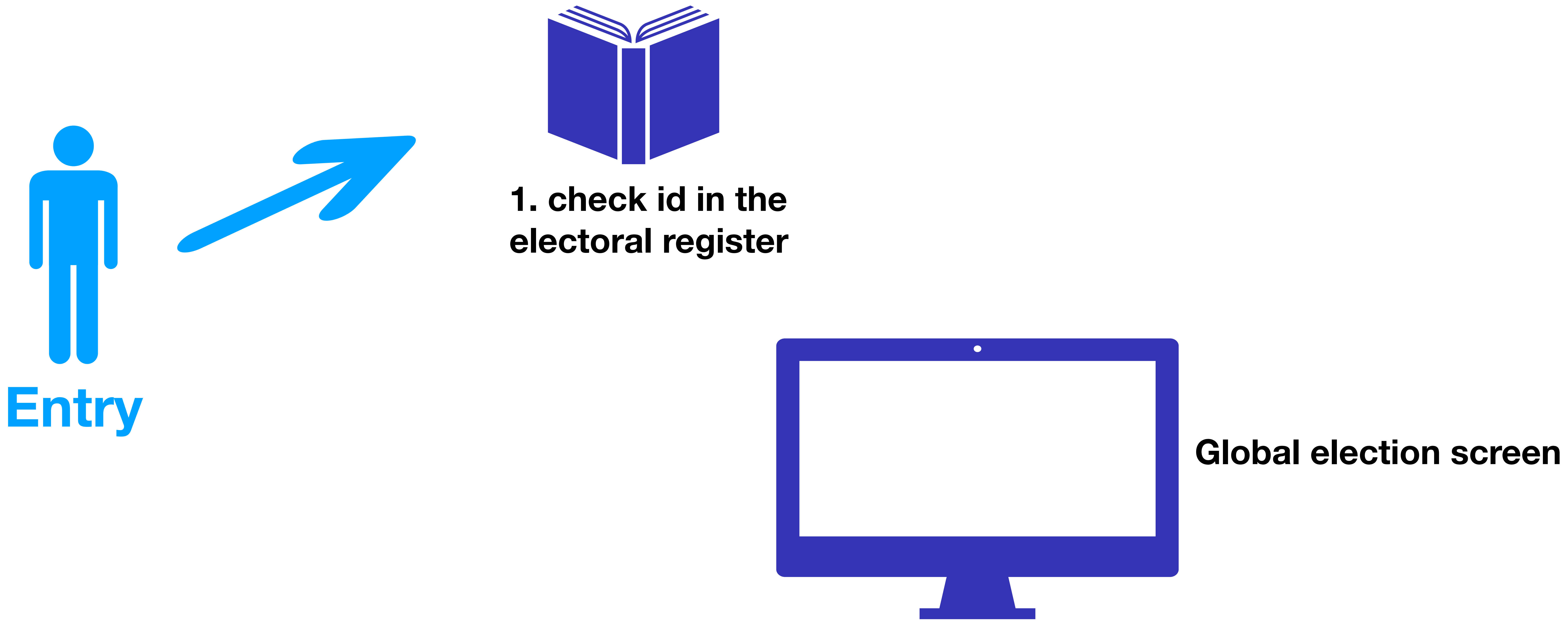


Entry

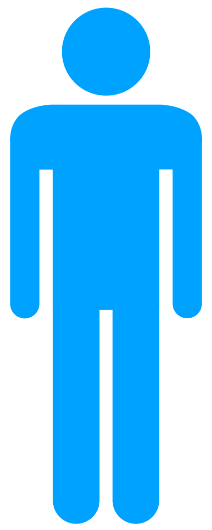


Global election screen

Overview of the system

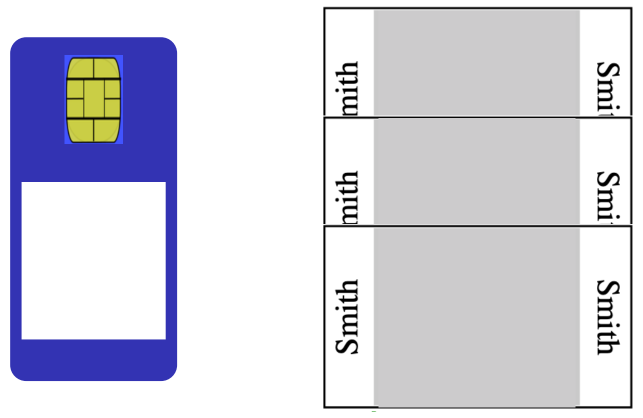


Overview of the system

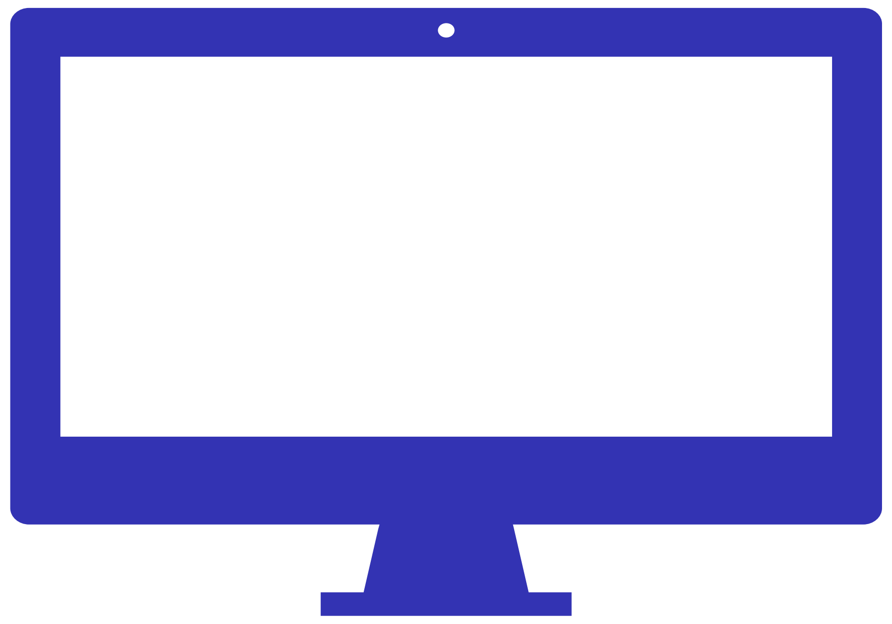


Entry

1. check id in the electoral register

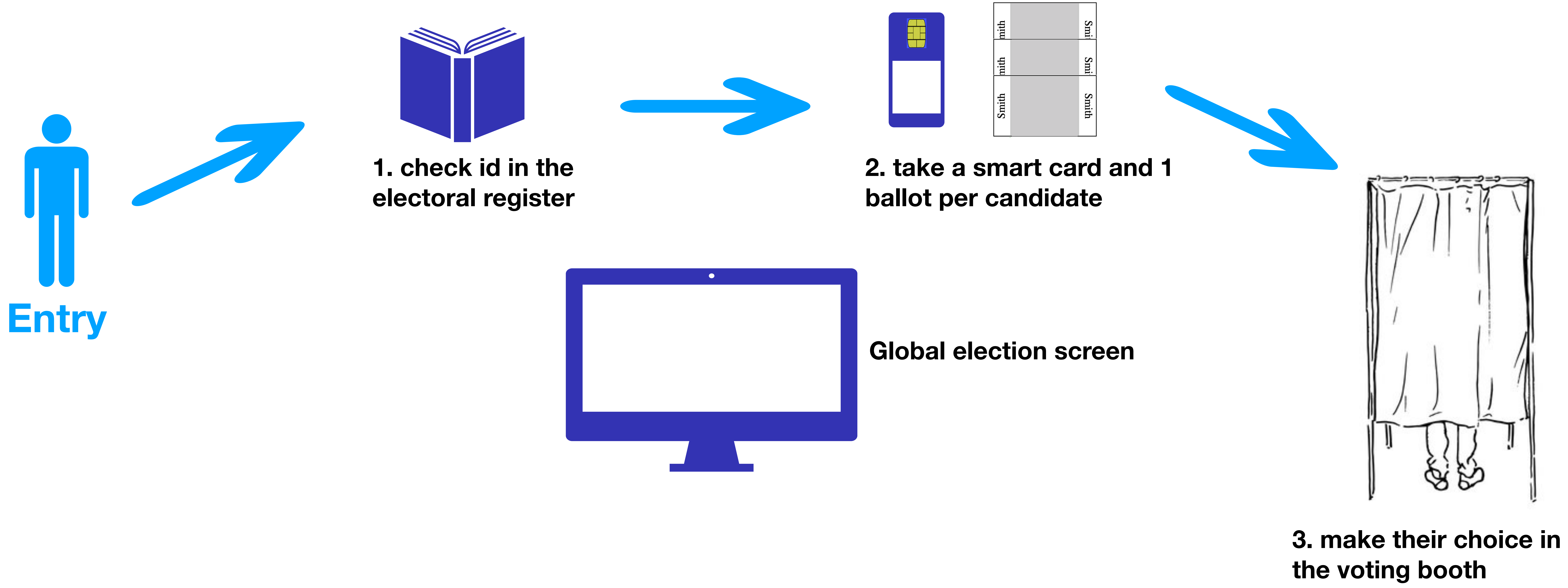


2. take a smart card and 1 ballot per candidate

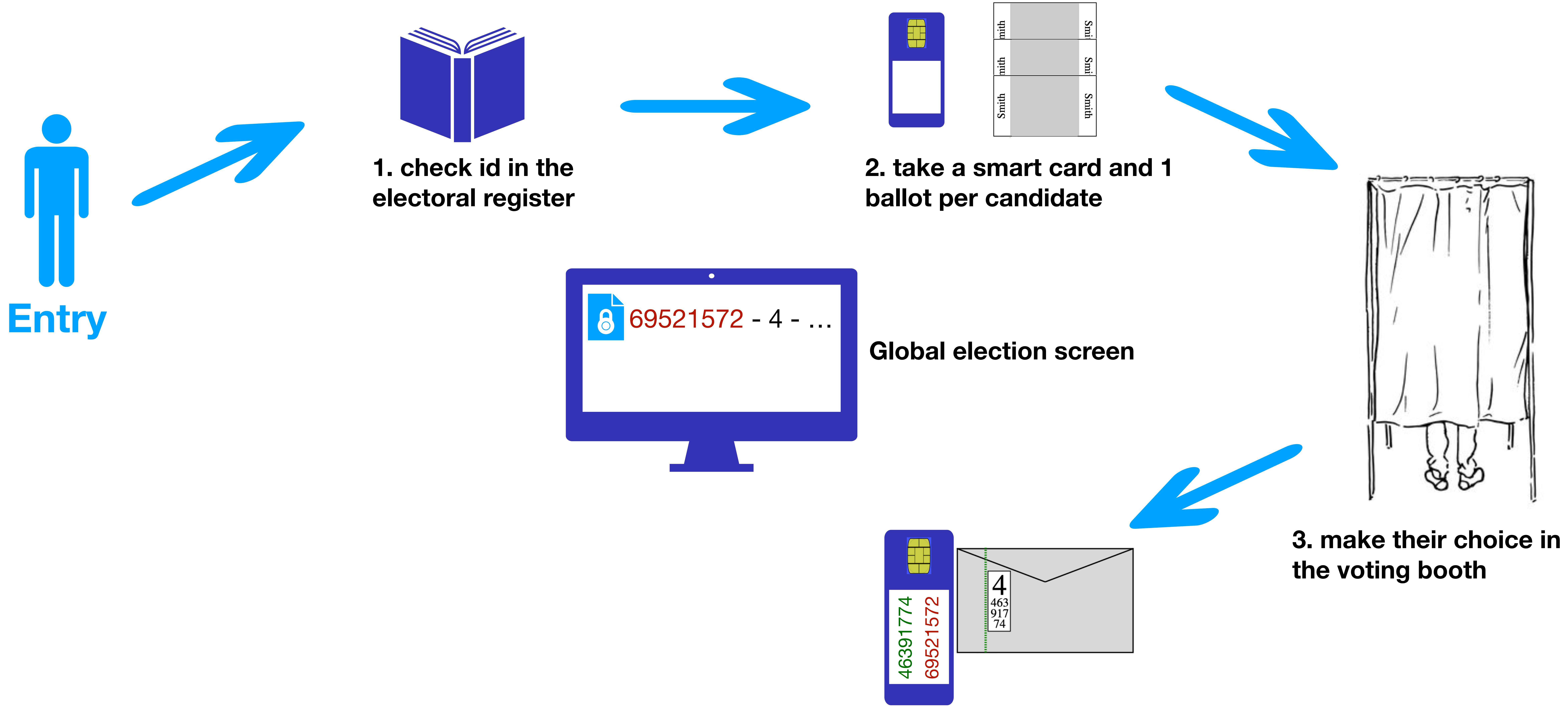


Global election screen

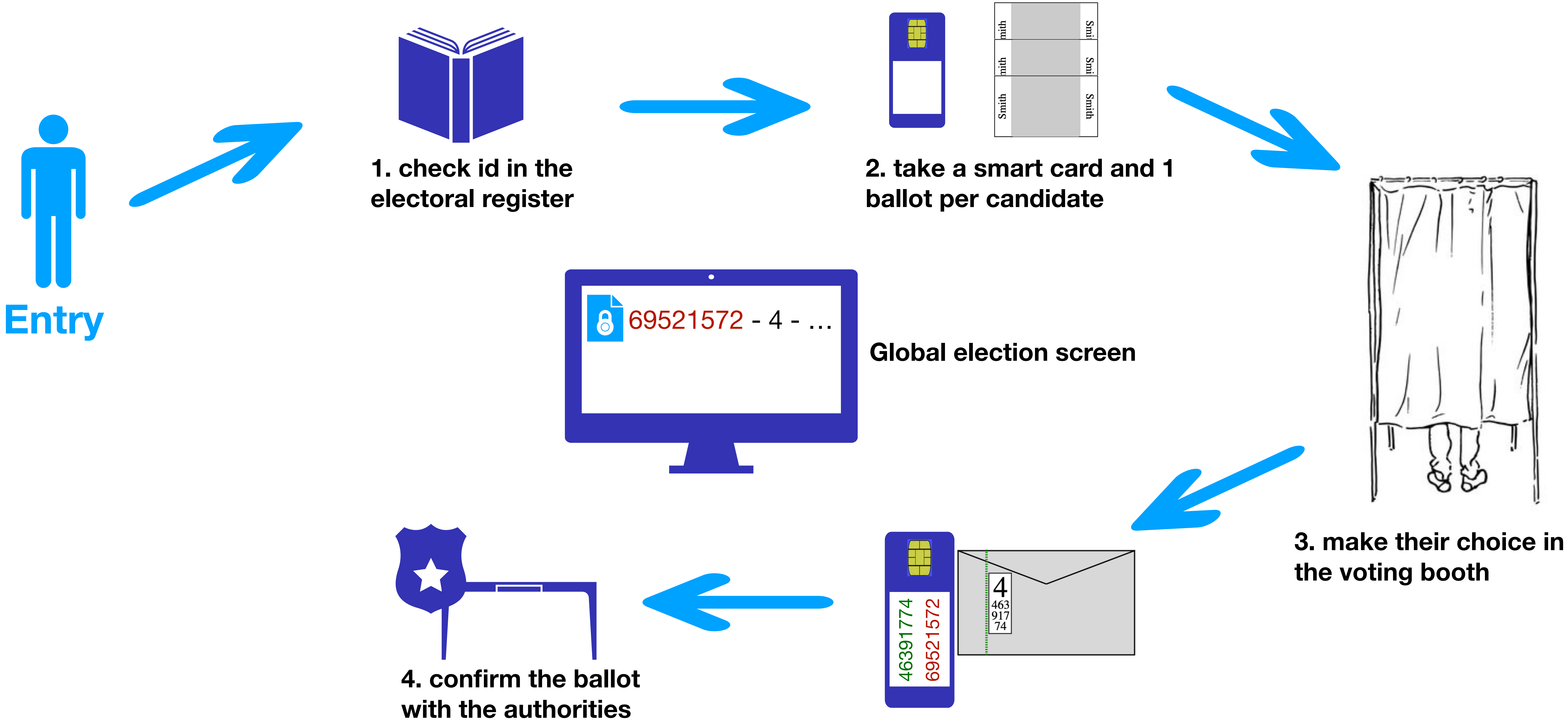
Overview of the system



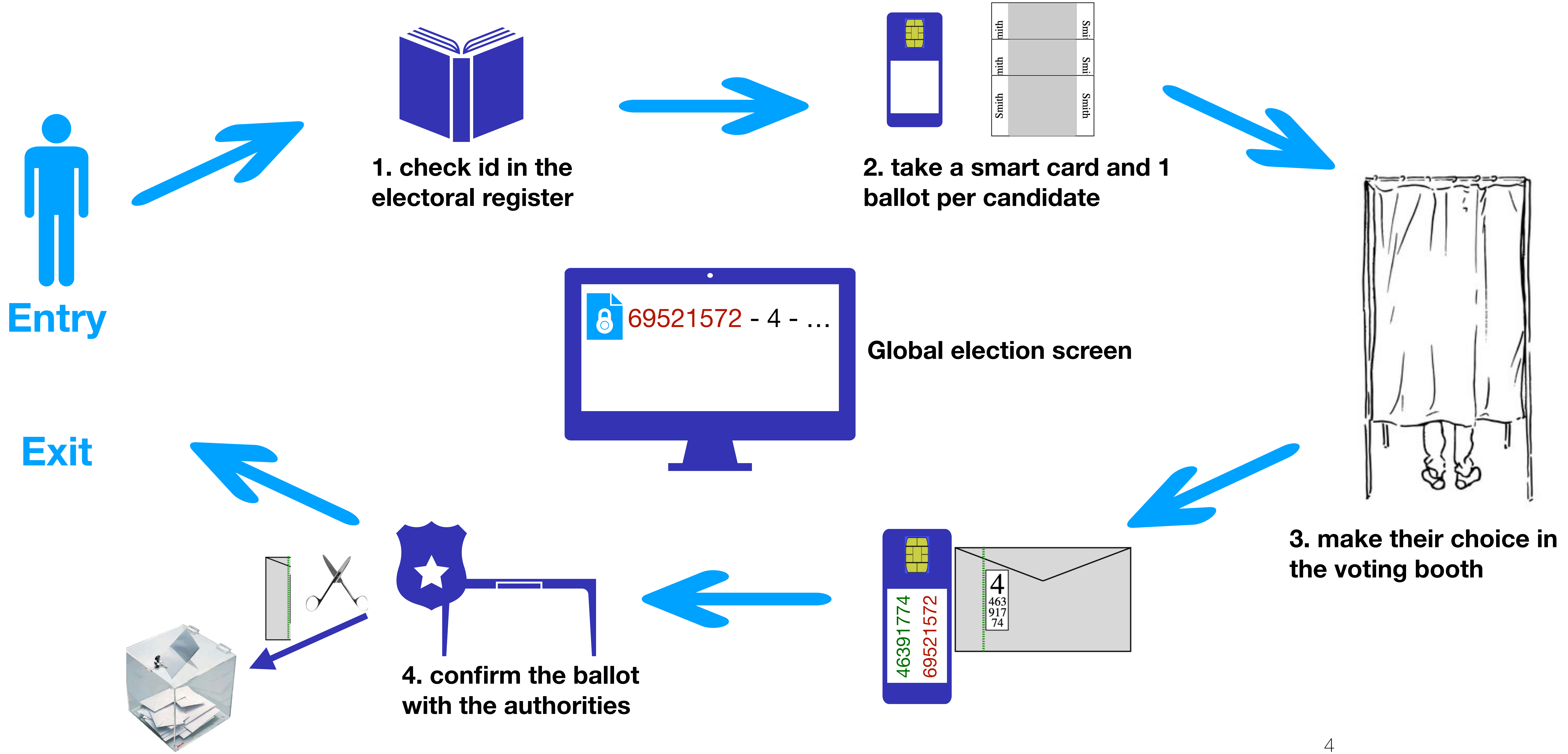
Overview of the system



Overview of the system

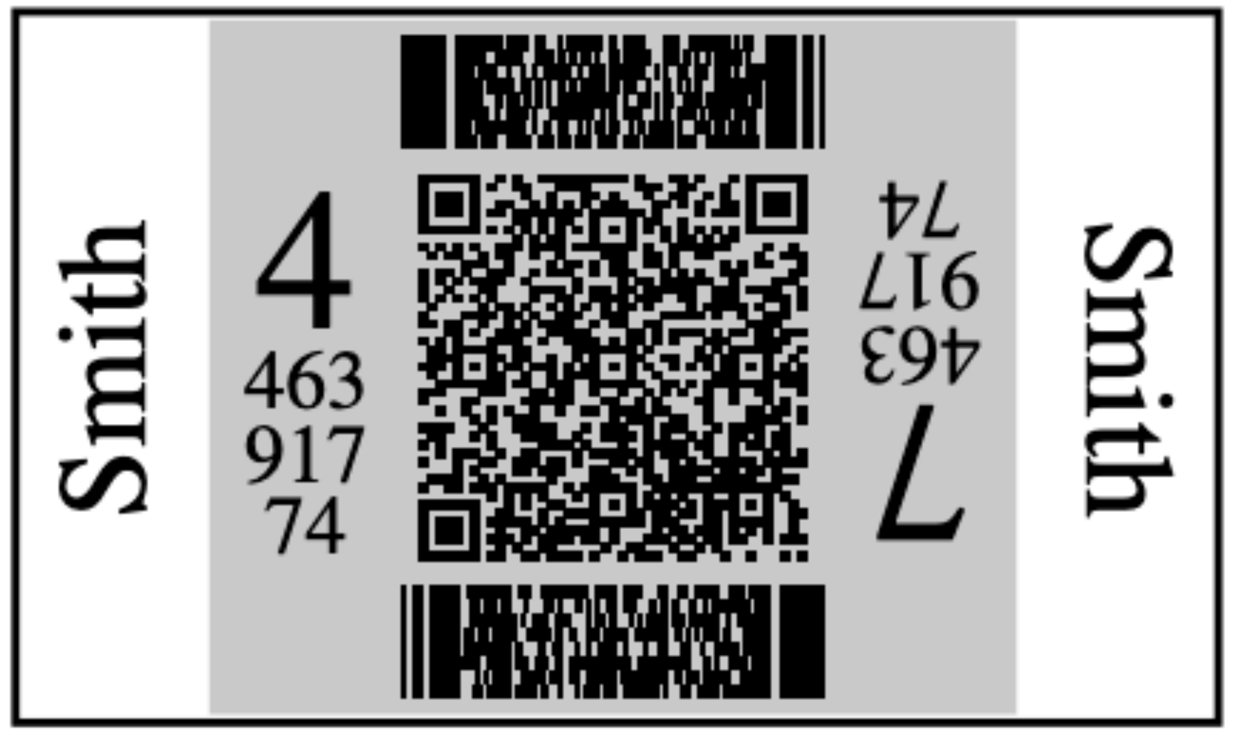


Overview of the system



Well-crafted ballots for cast-as-intended

Cast-as-intended: a corrupted device cannot modify the intended choice of a voter

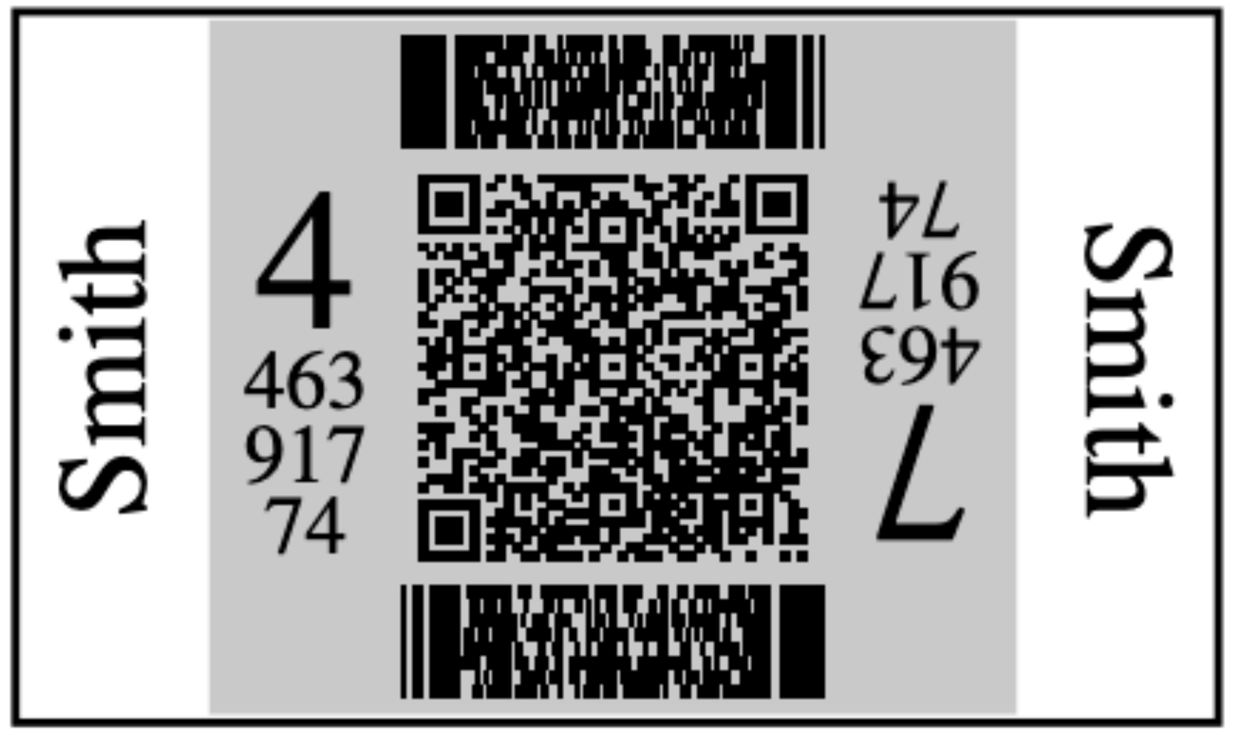


Well-crafted ballots for cast-as-intended

Cast-as-intended: a corrupted device cannot modify the intended choice of a voter

Paper ballot format:

- each candidate is associated to a unique integer
  e.g. Smith = 1
- each ballot for candidate X contains 2 verification codes A and B such that: $X = A + B \mod n$ (for a predefined n)
  e.g. $1 = 4 + 7 \mod 10$

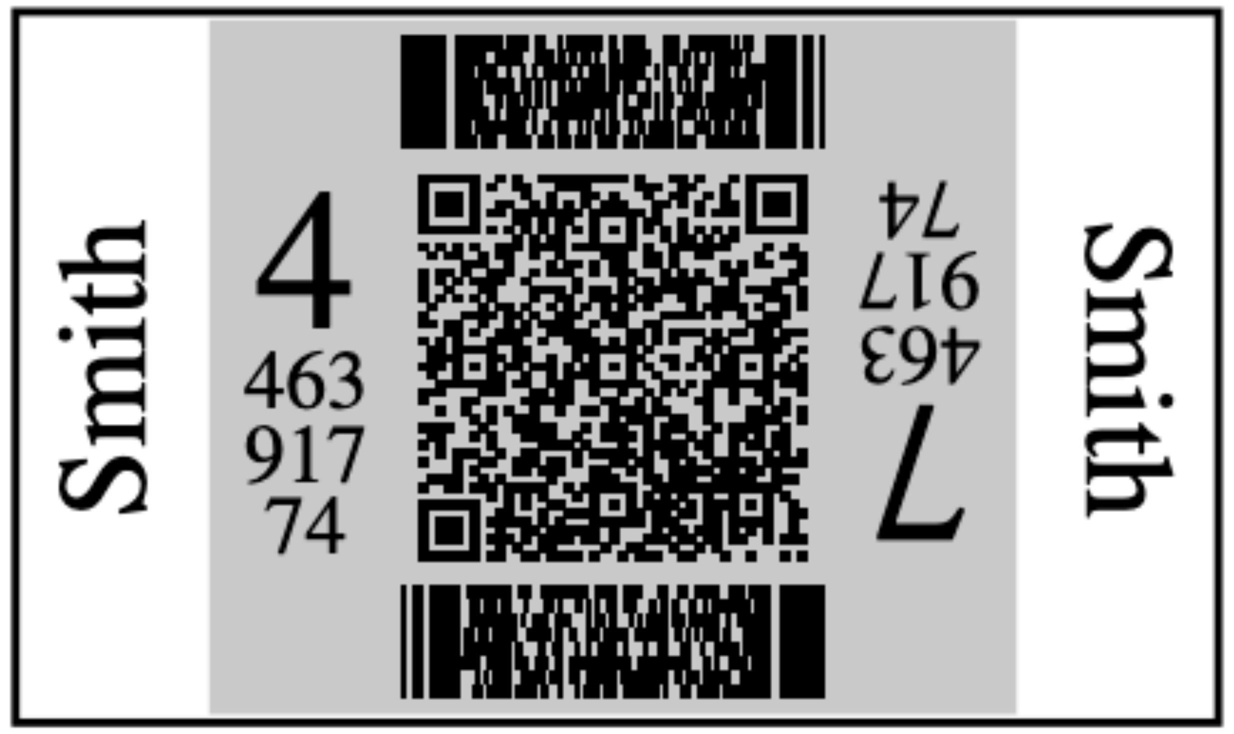


Well-crafted ballots for cast-as-intended

Cast-as-intended: a corrupted device cannot modify the intended choice of a voter

Paper ballot format:

- each candidate is associated to a unique integer
  e.g. Smith = 1
- each ballot for candidate X contains 2 verification codes A and B such that: $X = A + B \mod n$ (for a predefined n)
  e.g. $1 = 4 + 7 \mod 10$



Electronic ballot format:

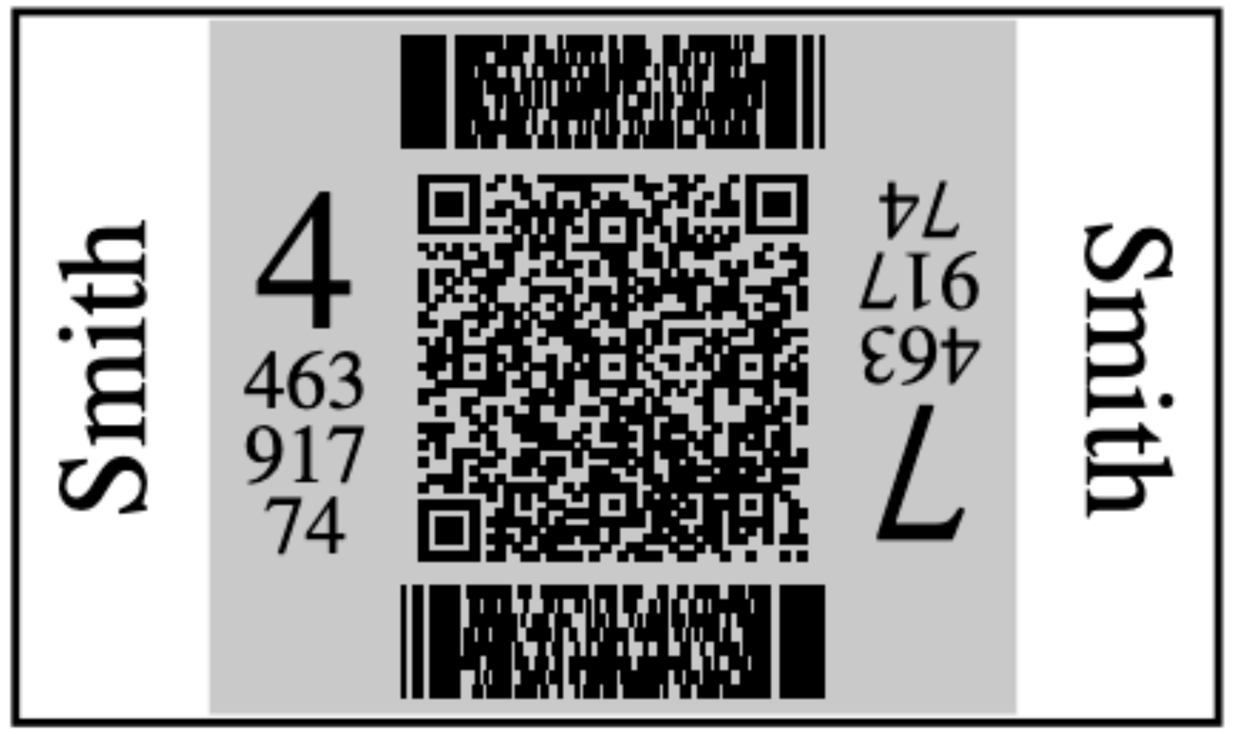
- each ballot contains 3 ciphertexts c_X, c_A, c_B and 1 ZKP π such that
  $\pi = ZKP(ptxt(c_X) = ptxt(c_A) + ptxt(c_B) \mod n)$
  e.g. $c_X = \{1\}_{pkE},\ c_A = \{4\}_{pkE},\ c_B = \{7\}_{pkE}$

Well-crafted ballots for cast-as-intended

Cast-as-intended: a corrupted device cannot modify the intended choice of a voter

Paper ballot format:

- each candidate is associated to a unique integer
  e.g. Smith = 1
- each ballot for candidate X contains 2 verification codes A and B such that: $X = A + B \mod n$ (for a predefined n)
  e.g. $1 = 4 + 7 \mod 10$



Electronic ballot format:

- each ballot contains 3 ciphertexts c_X, c_A, c_B and 1 ZKP π such that
  $\pi = ZKP(ptxt(c_X) = ptxt(c_A) + ptxt(c_B) \mod n)$
  e.g. $c_X = \{1\}_{pkE}, \ c_A = \{4\}_{pkE}, \ c_B = \{7\}_{pkE}$

The voter choses to audit A or B and the smart card must reveal the random used to forge the corresponding encryption c_A or c_B.

Well-crafted ballots for cast-as-intended

Cast-as-intended: a corrupted device cannot modify the intended choice of a voter

Paper ballot format:

- each candidate is associated to a unique integer
  e.g. Smith = 1
- each ballot for candidate X contains 2 verification codes A and B such that: $X = A + B \mod n$ (for a predefined n)
  e.g. $1 = 4 + 7 \mod 10$

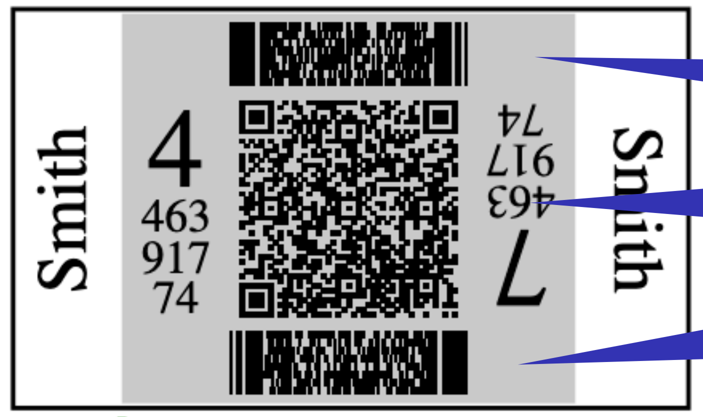
Ballot manipulations are detected with probability $\frac{1}{2}$

Electronic ballot format:

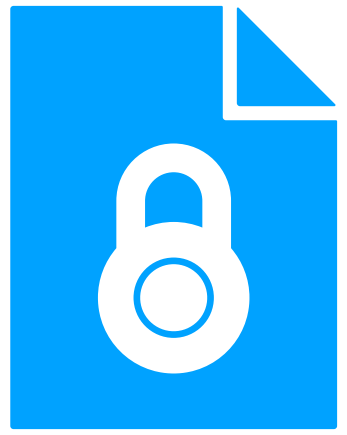
- each ballot contains 3 ciphertexts c_X, c_A, c_B and 1 ZKP π such that
  $\pi = ZKP(ptxt(c_X) = ptxt(c_A) + ptxt(c_B) \mod n)$
  e.g. $c_X = \{1\}_{pkE}$, $c_A = \{4\}_{pkE}$, $c_B = \{7\}_{pkE}$

The voter choses to audit A or B and the smart card must reveal the random used to forge the corresponding encryption c_A or c_B.

Accountability by-design

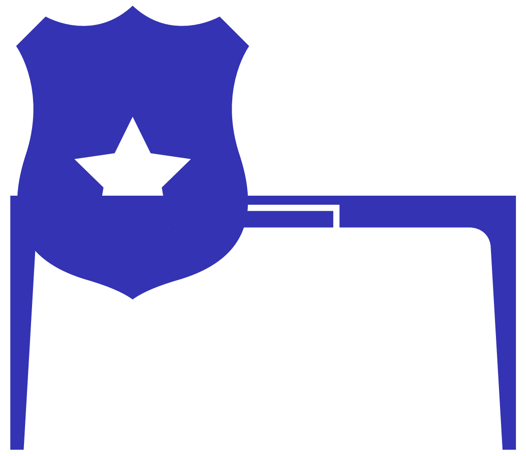


Digital signatures by the printer



Digital signatures by the smart card

A hash chain of blocks signed by the server



Voters and local authorities mutually control their actions

A dispute resolution procedure

- executed when a critical error is detected
- 9 steps:
  - 5 can be executed live
  - + 4 offline because breaks privacy
- can (almost) always deduce the culprit (sometimes a subset of possible culprits)
- protects against false accusations

A formally proven protocol

ProVerif

- An automatic prover for symbolic analysis
- Handle trace-based properties for e.g., verifiability or accountability
- Handle equivalence-based properties for e.g., vote secrecy

A formally proven protocol

ProVerif

- An automatic prover for symbolic analysis
- Handle trace-based properties for e.g., verifiability or accountability
- Handle equivalence-based properties for e.g., vote secrecy

2 main challenges

- Accountability: ProVerif does not support liveness properties
  ➡ carefully define the queries
  ➡ exhaustively identify each possible final state of the protocol by an event

A formally proven protocol

ProVerif

- An automatic prover for symbolic analysis
- Handle trace-based properties for e.g., verifiability or accountability
- Handle equivalence-based properties for e.g., vote secrecy

2 main challenges

- Accountability: ProVerif does not support liveness properties
  ➡ carefully define the queries
  ➡ exhaustively identify each possible final state of the protocol by an event
- Audit mechanism: ProVerif does not support arithmetics in $\mathbb{Z}_n$
  ➡ reachability: over-approximate the "+" operator
  ➡ equivalence: prove a relation preservation

Modeling arithmetics in $\mathbb{Z}_n$

Modelling:

- integers are modeled by abstract atomic values, $x, y, a, b, c, \ldots$
- whenever someone checks $x \stackrel{?}{=} a + b$, we execute the event $isSum(x, a, b)$

Modeling arithmetics in $\mathbb{Z}_n$

Modelling:

- integers are modeled by abstract atomic values, $x, y, a, b, c, \ldots$
- whenever someone checks $x =^? a + b$, we execute the event $isSum(x, a, b)$

Reachability properties:

« For all $x, a \in \mathbb{Z}_n$, there exists a unique $b \in \mathbb{Z}_n$ such that $b = x + a$ »

Restrictions such that

$$isSum(x, a, b) \ \wedge \ isSum(x, a, b') \Rightarrow b = b'$$

$$isSum(x, a, b) \ \wedge \ isSum(x, a', b) \Rightarrow a = a'$$

$$\ldots$$

Modeling arithmetics in $\mathbb{Z}_n$

Modelling:

- integers are modeled by abstract atomic values, $x, y, a, b, c, \ldots$
- whenever someone checks $x =^? a + b$, we execute the event $isSum(x, a, b)$

Reachability properties:

« For all $x, a \in \mathbb{Z}_n$, there exists a unique $b \in \mathbb{Z}_n$ such that $b = x + a$ »

Restrictions such that

$$isSum(x, a, b) \ \wedge \ isSum(x, a, b') \Rightarrow b = b'$$

$$isSum(x, a, b) \ \wedge \ isSum(x, a', b) \Rightarrow a = a'$$

…

Equivalence properties: relation preservation

Lemma (intuition): given two processes P and Q, for all traces $tr_P \in Traces(P)$ and $tr_Q \in Traces(Q)$ such that $tr_P \approx tr_Q$ we have:

$$isSum(x, a, b) \in tr_P \ \Leftrightarrow \ isSum(x, a, b) \in tr_Q$$

(related to the notion of bi-process and diff-equivalence)

Conclusion

Themis is:

- **a verifiable, private, and accountable voting protocol**

- **a formally proven protocol**

- **protocol that can be used in practice**

  ➡ preliminary experiments have been conducted to demonstrate its usability (still require large scale experiments)

Conclusion

Themis is:

- **a verifiable, private, and accountable voting protocol**

- **a formally proven protocol**

- **protocol that can be used in practice**
  - ➡ preliminary experiments have been conducted to demonstrate its usability (still require large scale experiments)

Thank you!