Votegral:
Coercion-Resistant E-voting Without Trusted Hardware

Louis-Henri Merino, Simone Colombo,
Jeffrey R. Allen, and Bryan Ford
Decentralized and Distributed Systems (DEDIS)
dedis@epfl.ch – dedis.epfl.ch

IC3 Blockchain Camp – July 28, 2021

Talk Outline

- Goal: free, people-centric self-governance
- Basics: approaches to coercion resistance
- Signup: governmental or decentralized voting
- Usability: how the user experiences signup
- Technical: what actually happens underneath

Talk Outline

- **Goal: free, people-centric self-governance**
- Basics: approaches to coercion resistance
- Signup: governmental or decentralized voting
- Usability: how the user experiences signup
- Technical: what actually happens underneath

Decentralized Digital Democracy

Will decentralized online systems ever be able to **self-govern** in an egalitarian, democratic fashion?

[Kenneth Hacker, The Progressive Post]

Contrasting Influence Foundations

Wealth-centric

- One dollar, one vote

[Kera]

Person-centric

- One person, one vote

[Verity Weekly]

Contrasting Influence Foundations

Wealth-centric

- Stock corporations
- Loyalty programs
- Online gaming
- CAPTCHA solving
- Proof-of-work
- Proof-of-stake
- Proof-of-X for most X

Person-centric

- Democratic states
- Elected parliaments
- Membership clubs
- Committees
- Town hall meetings
- Direct democracy
- Liquid democracy

Contrasting Influence Foundations

Wealth-centric

✓

Largely Solved

Person-centric

✗

Largely Unsolved

Person-Centric Self-Governance

A few major unsolved questions & challenges:

- Defining a suitable **decentralized architecture**
  - See “Technologizing Democracy…?” [2020]
- Creating Sybil-resistant **proofs of personhood**
  - See “Identity and Personhood…” [2020]
- Scalable participatory **deliberation structures**
  - See “A Liquid Perspective…” [2018]
- Ensuring **freedom** from coercion, vote-buying
  - Topic of this talk

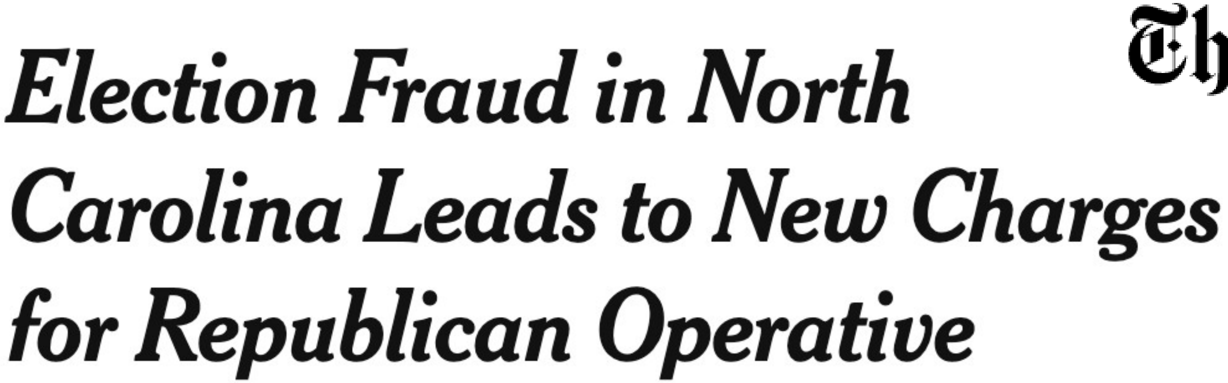
The Coercion, Vote-Buying Problem

How can we know people vote their **true intent** if we can’t secure the environment they vote in?

The Coercion, Vote-Buying Problem

Both **Postal** and **Internet** voting are vulnerable!

The New York Times

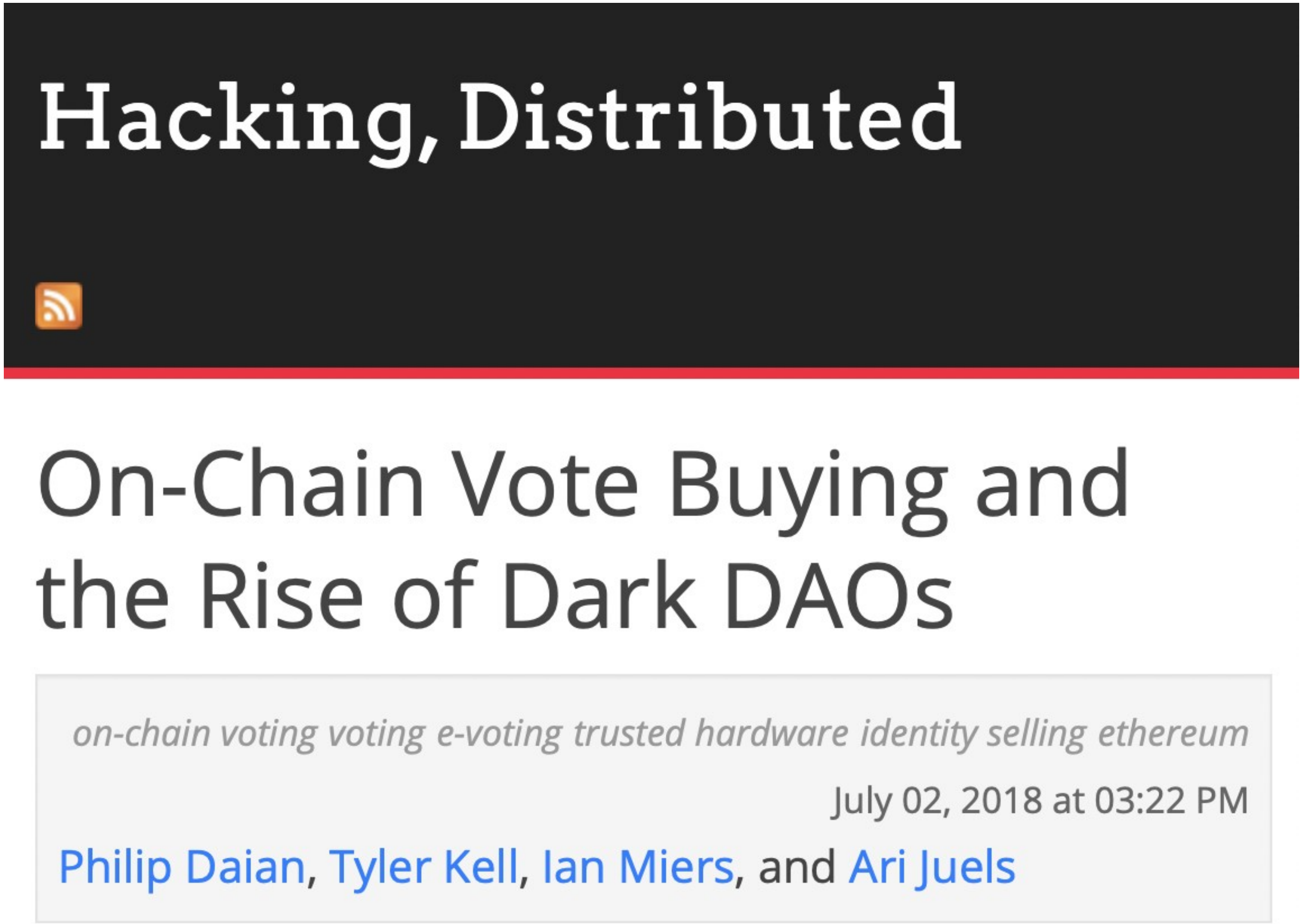


Election Fraud in North Carolina Leads to New Charges for Republican Operative

July 30, 2019

The Coercion, Vote-Buying Problem

Blockchain systems are especially vulnerable!



Hacking, Distributed

On-Chain Vote Buying and the Rise of Dark DAOs

on-chain voting voting e-voting trusted hardware identity selling ethereum

July 02, 2018 at 03:22 PM

Philip Daian, Tyler Kell, Ian Miers, and Ari Juels

Talk Outline

- Goal: free, people-centric self-governance
- **Basics: approaches to coercion resistance**
- Signup: governmental or decentralized voting
- Usability: how the user experiences signup
- Technical: what actually happens underneath

Approaches to Coercion Resistance

- Re-voting (Estonia, Spycher/Haenni/Dubuis, …)
  - Later vote can override an earlier (coerced) vote
  - Key limitation: true preference must be cast *last*
    - Coercer can keep voter under surveillance until deadline

- Fake credentials (JCJ, RSV, …)
  - User can get both *real* and *fake* voting credentials
    - Fake credentials “work” but cast votes that don’t count
  - Can give or sell fake credentials to any coercer

Coercion Resistance, JCJ-Style

JCJ tradition: voters get *real* and *fake* credentials

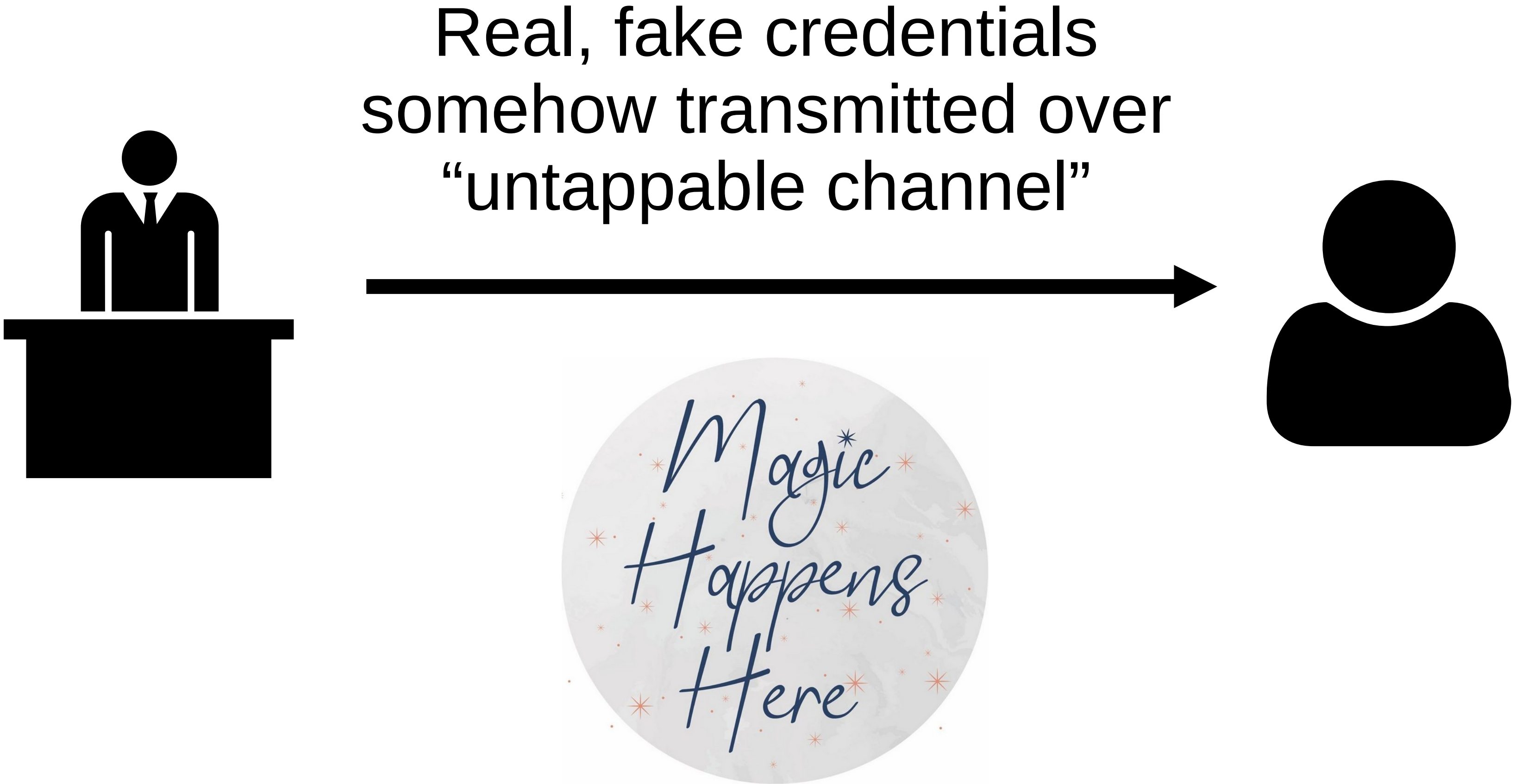
- Can give or sell fake credentials to any coercer

Some key challenges with JCJ

- How do voters *securely* get real credentials?
- Usability: needs complex cryptographic dances
- Quadratic computation cost (mostly solved)
- Bulletin board flooding attacks (mostly solved)

The Signup Problem: First Cut

The scenario typically *assumed* in theoretical work

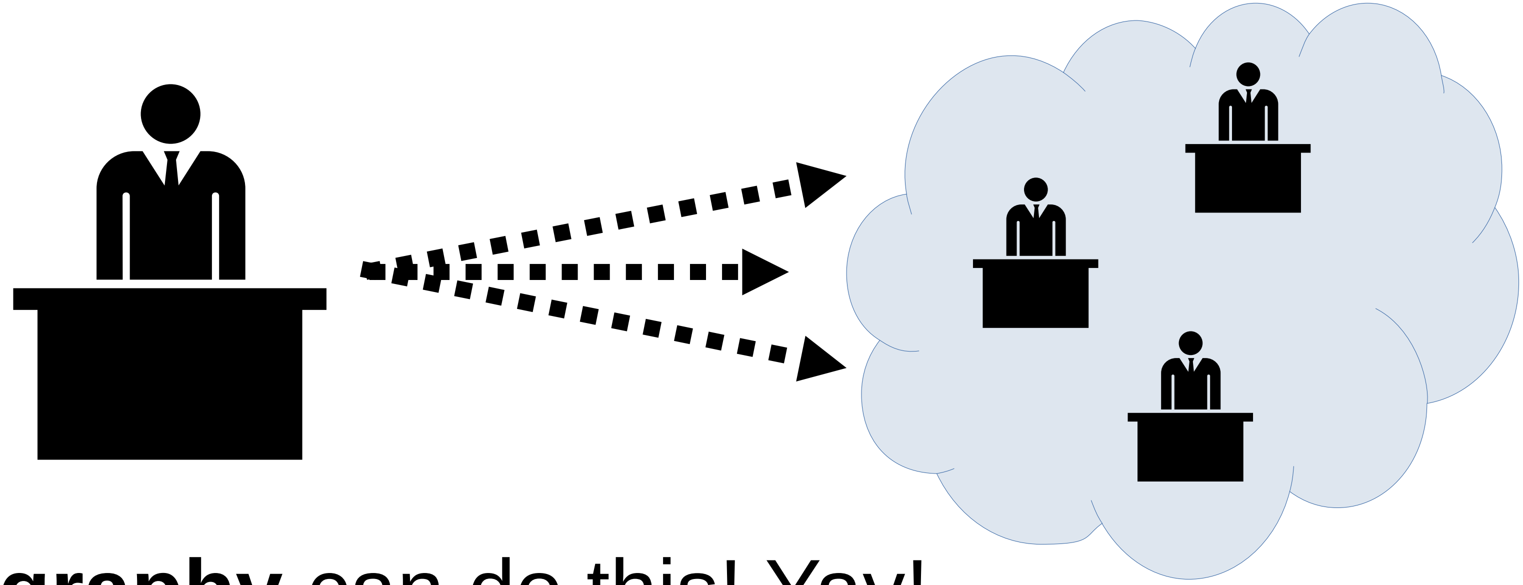


Real, fake credentials
somehow transmitted over
“untappable channel”

First Main Problem

We want things to be *decentralized* – i.e., don’t want to trust a single election authority!

So we *decentralize* the election authority by splitting its role over multiple parties…

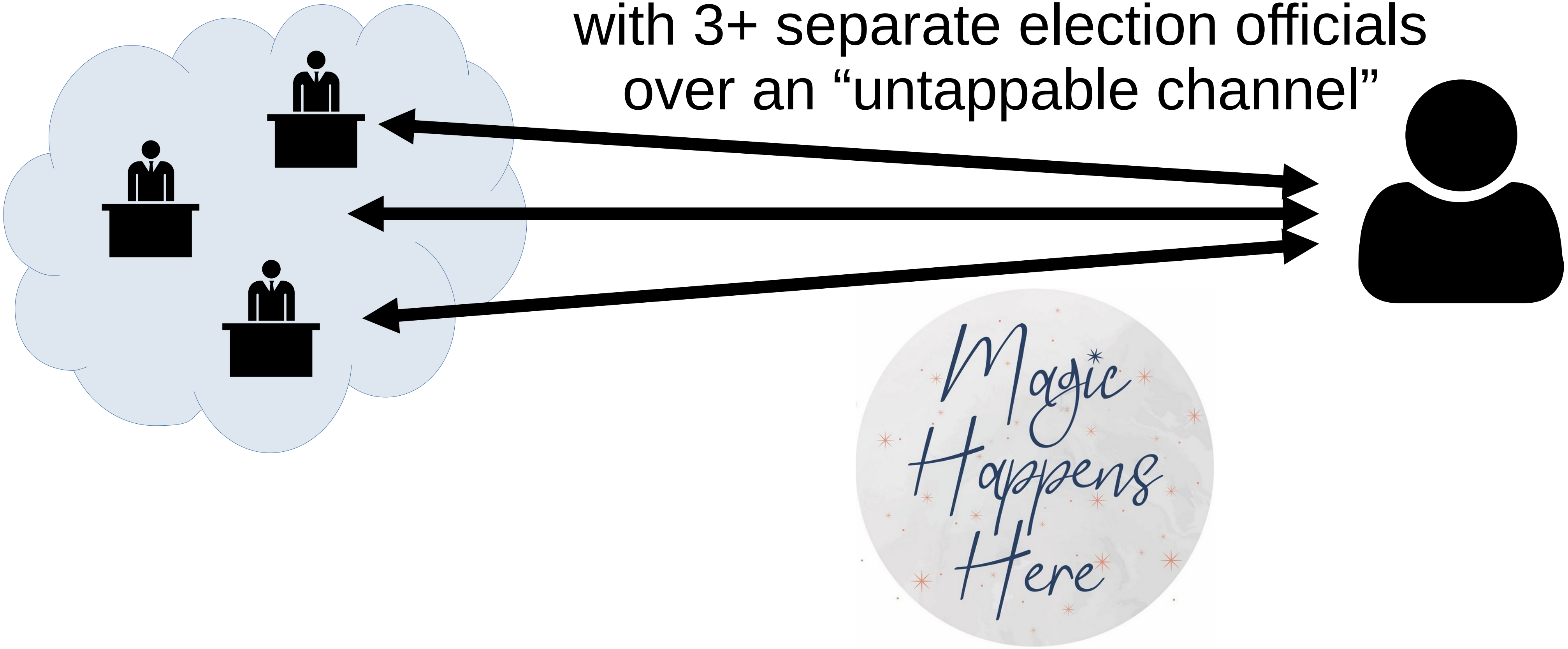


Cryptography can do this! Yay!

The Signup Problem: Next Cut

The scenario typically *assumed* in theoretical work

Ordinary, unsophisticated voter
performs elaborate cryptographic dance
with 3+ separate election officials
over an “untappable channel”



Closest-to-Practical Precedent

JCJ in the Civitas E-voting system

- [Neumann/Volkamer ‘12],[Neuman et al ‘13]

Assumes every voter has **trusted hardware**

- Specifically, a **smart card** that can perform the elaborate cryptographic dance for the user

Could work, but (a) costly, and (b) defeats goal of transparency, independent verifiability of E-voting

The Continuing Challenge

Can we make coercion-resistant E-voting…

- Usable: no elaborate cryptographic dances?
- Secure: no single points of compromise?

That is the Votegral’s goal.

Talk Outline

- Goal: free, people-centric self-governance
- Basics: approaches to coercion resistance
- **Signup: governmental or decentralized voting**
- Usability: how the user experiences signup
- Technical: what actually happens underneath

Votegral Use-Cases

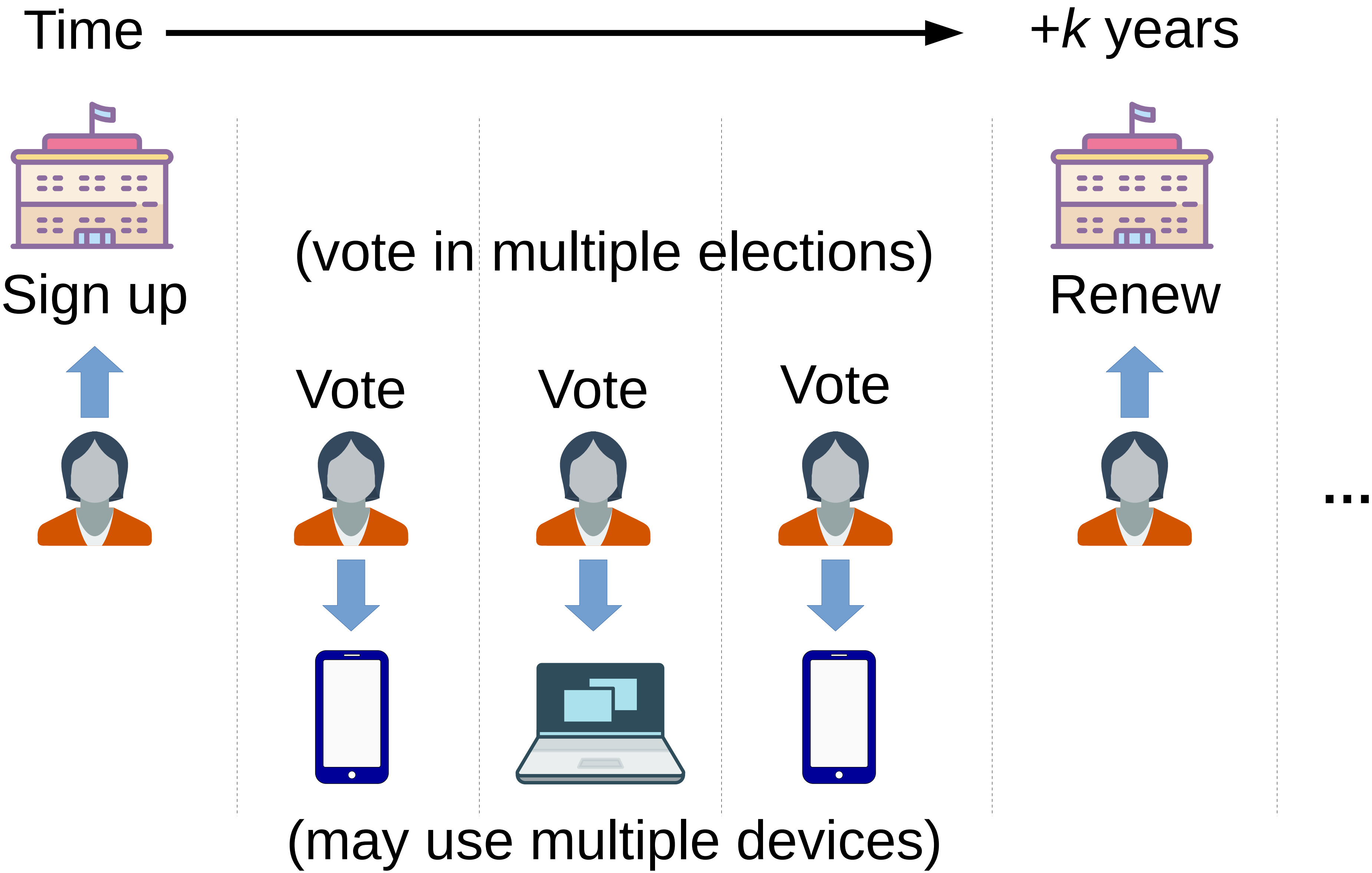
Could in principle be deployed either by:

- Governments, for E-voting in public elections
- Decentralized systems w/ proof of personhood

Difference is when in-person “signup” happens

- Governmental: periodically at a suitable office
- Decentralized: periodically at pseudonym party

Government use-case: outline



Time → +*k* years

Sign up

(vote in multiple elections)

Vote

Vote

Vote

Renew

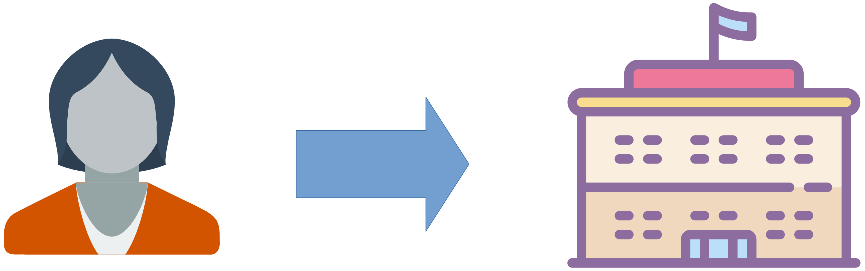
...

(may use multiple devices)

In-person E-voting signup

To use E-voting, voter must visit designated office in person to sign up or renew **every few years**:

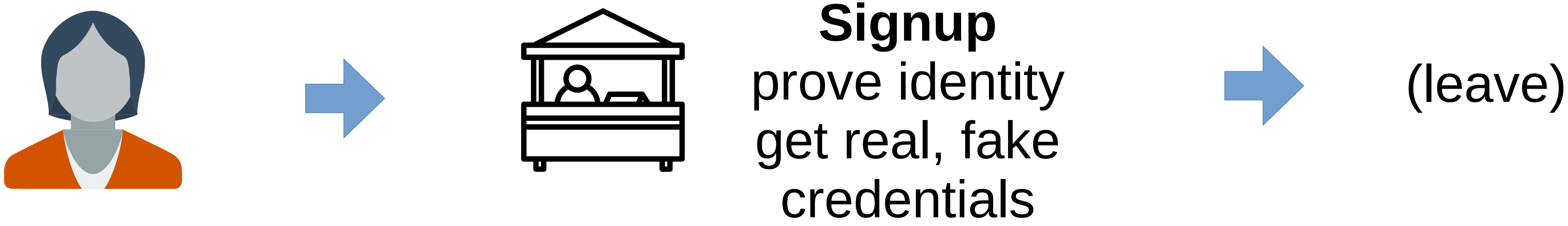
- **Locals:** residents services or ID card office
- **Expats:** embassy, consulate, authorized notary



Might be coincident with obtaining or renewing voter’s national ID card, passport, drivers license

Signup process outline

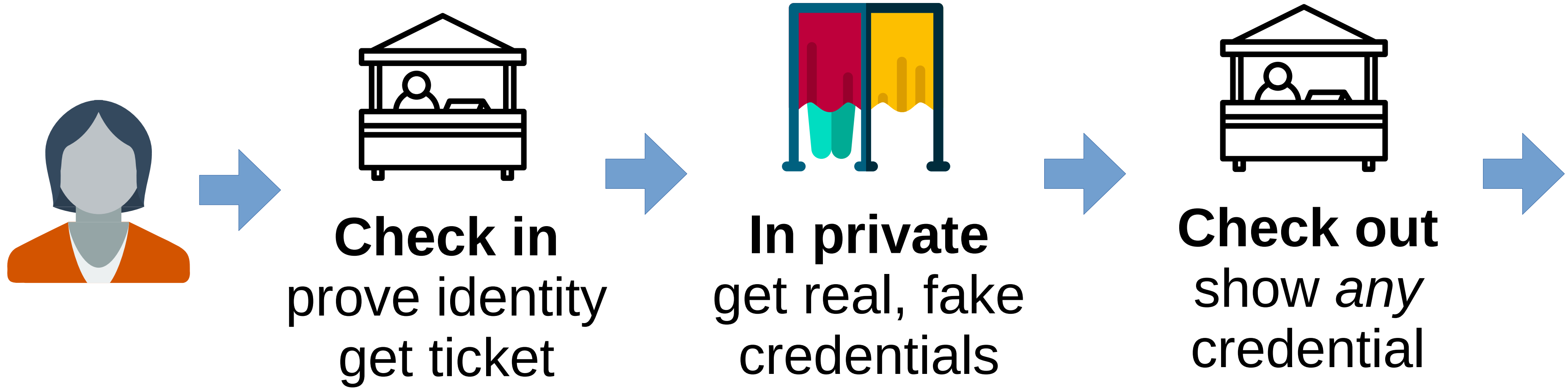
E-voting signup, low coercion threat (e.g., CH?)



Signup
prove identity
get real, fake
credentials

(leave)

In-person voting or high coercion threat (not CH?)



Check in
prove identity
get ticket

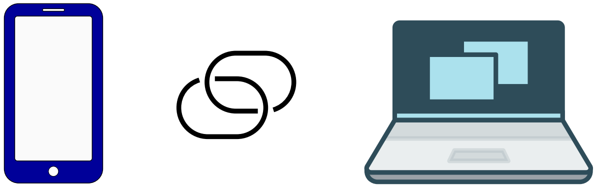
In private
get real, fake
credentials

Check out
show *any*
credential

E-voting across personal devices

Voter can link several trusted personal devices

- **Cast** votes on any linked personal device
- **Check** recent voting record on any device



Cast-as-intended protection: assumes *not all* voter’s personal devices compromised together

- But one device can compromise vote privacy

In-person signup: acceptable cost?

Cost/benefit to voters:

- **Cost:** one in-person visit every few years
- **Benefit:** instant voting in frequent elections
- **Benefit:** cast & verify votes across devices

Cost/benefit to governments:

- **Cost:** offer signup service in local offices
- **Benefit:** save ballot printing and mailing costs
- **Benefit:** no dependence on international mail

Decentralized use-case: outline

Suppose we build a blockchain system using pseudonym parties as 1-per-person stake basis

- Mining/voting power distributed evenly in each epoch among all *people* who show up in-person
- “Proof-of-Personhood: Redemocratizing Permissionless Cryptocurrencies” [2017]

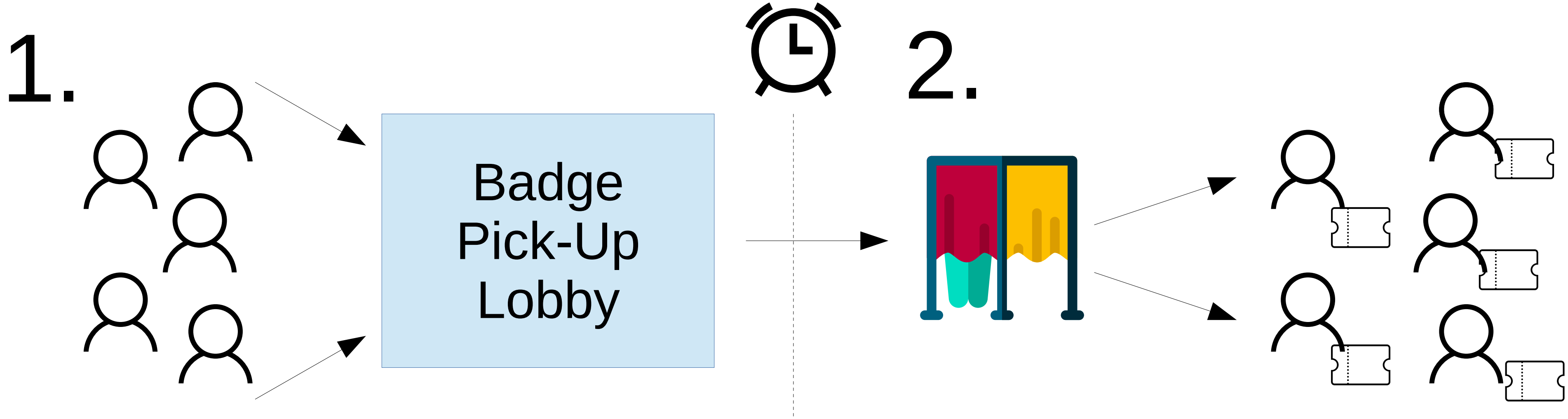
Can we ensure these *people* aren’t just minions paid by a whale to show up and push an agenda?

Pseudonym Parties with Votegral

To get a token, attendees must arrive and enter a closed or cordoned-off *lobby* by a set deadline

At deadline, entrance doors closed: *no re-entry*

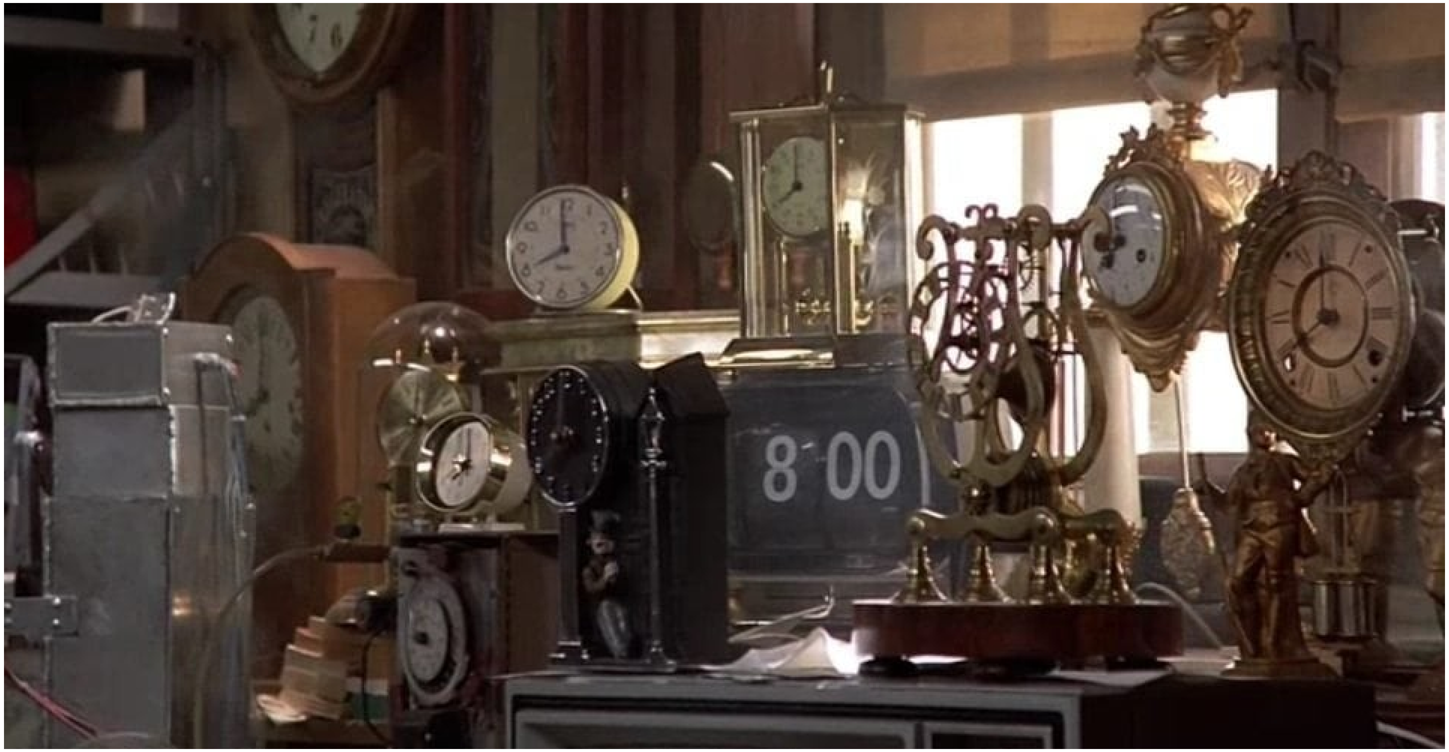
- Attendees file out from lobby to "main event"
- Via a "privacy booth" to get *real* & *fake* tokens



Pseudonym Parties: Scaling

Federation of PoP groups might hold *concurrent* events with *simultaneous* arrival deadlines

- No one can physically attend two at once



Votegral’s key contribution

Make ***signup*** usable, verifiable, coercion-resistant

- Assume in-person signup is an acceptable cost
- Treat in-person signup like in-person voting: a *private but verifiable choice* among alternatives
  - Voting is a choice between candidates or options
  - Signup is a meta-choice between *voting channels*

In-person “voting” techniques can secure signup!

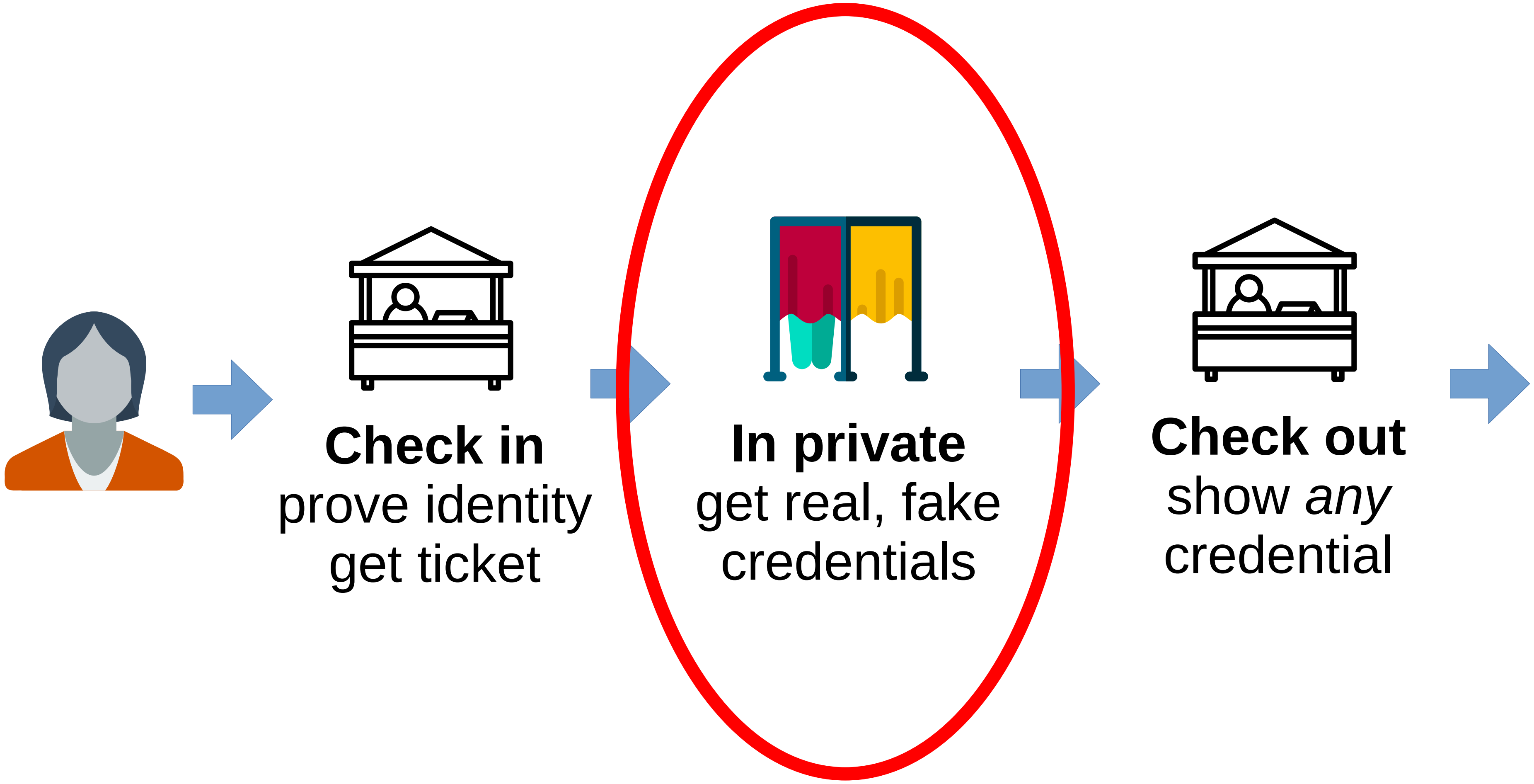
- Ensure voter can *verify* choice but can’t *prove* it

Talk Outline

- Goal: free, people-centric self-governance
- Basics: approaches to coercion resistance
- Signup: governmental or decentralized voting
- **Usability: how the user experiences signup**
- Technical: what actually happens underneath

What is in the “privacy booth”?

High coercion threat case as example



Check in
prove identity
get ticket

In private
get real, fake
credentials

Check out
show *any*
credential

Demo video

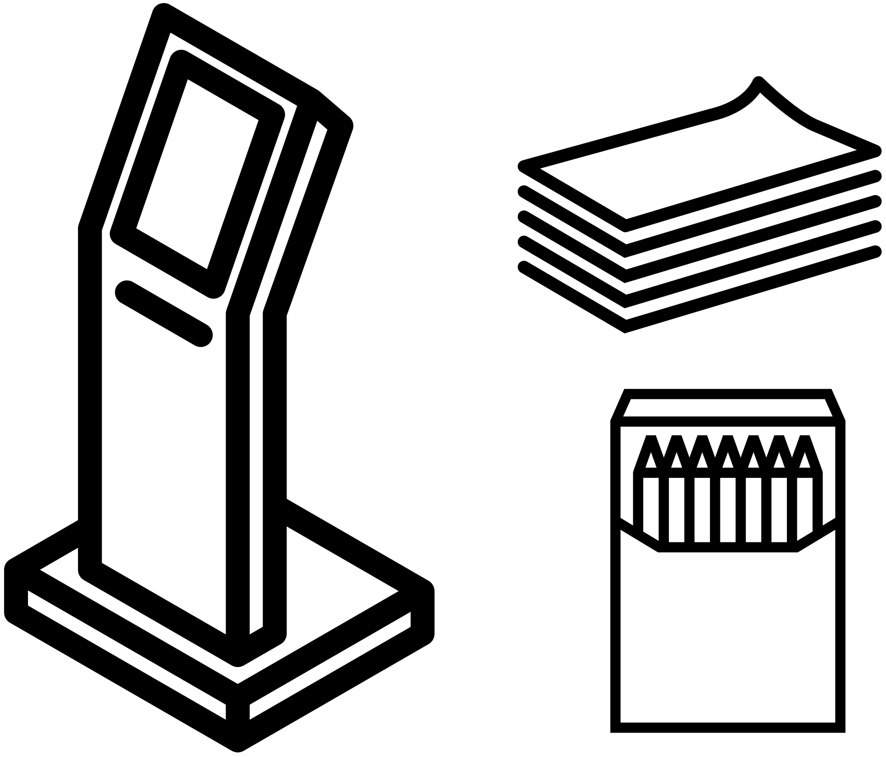
https://votegral.org/demo/

password: **Wahl**

What is in the “privacy booth”?

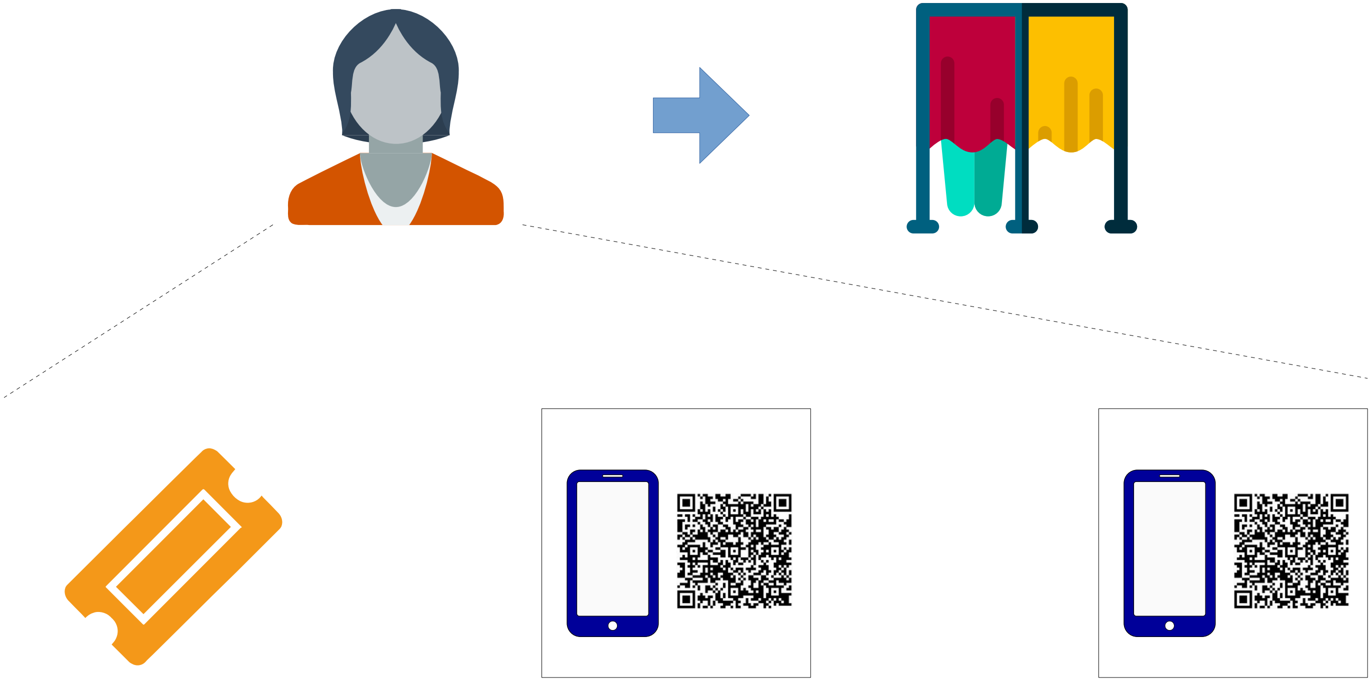
Kiosk or terminal with scanner, receipt printer, stack of envelopes with printed QR codes, pencils

- Could be used for signup *and* in-person voting



What happens in “privacy booth”?

Enter booth with check-in ticket, voting device(s) or printed device IDs (QR-coded public key)



Check-in ticket

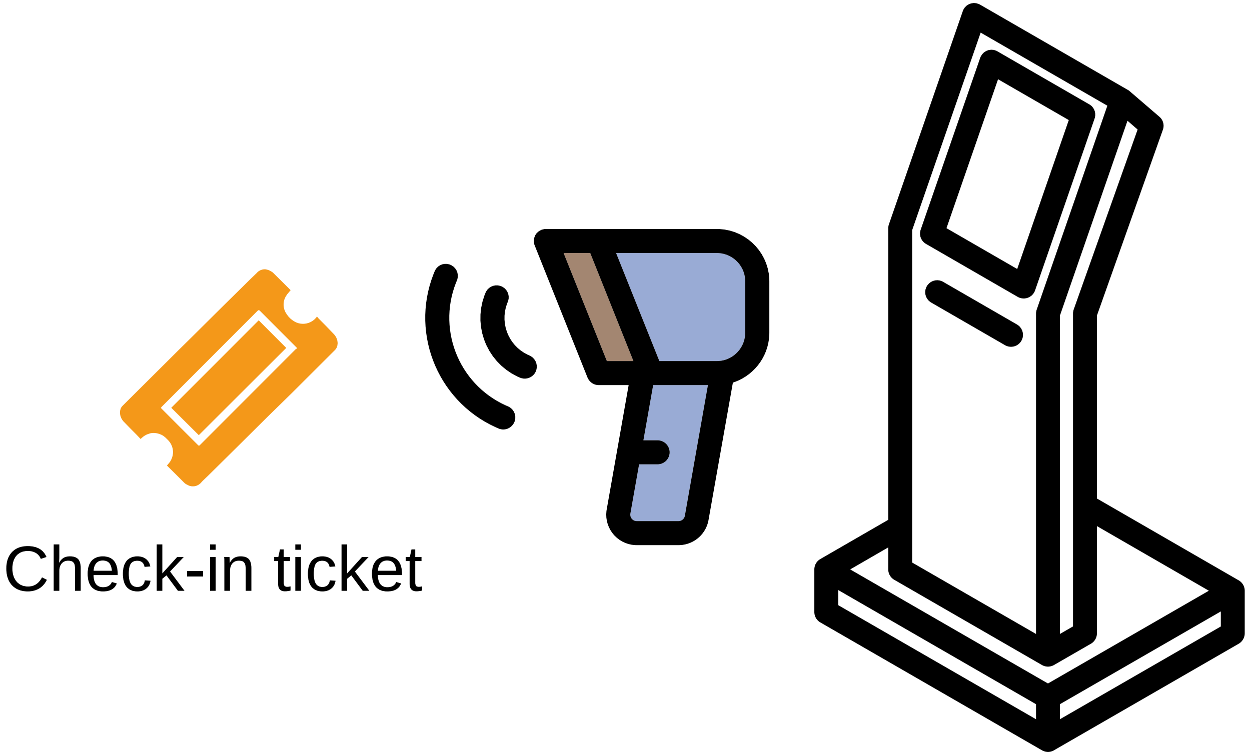
Voter-trusted device ID

Coercer’s device ID

What happens in “privacy booth”?

Enter booth with check-in ticket, voting device(s) or printed device IDs (QR-coded public key)

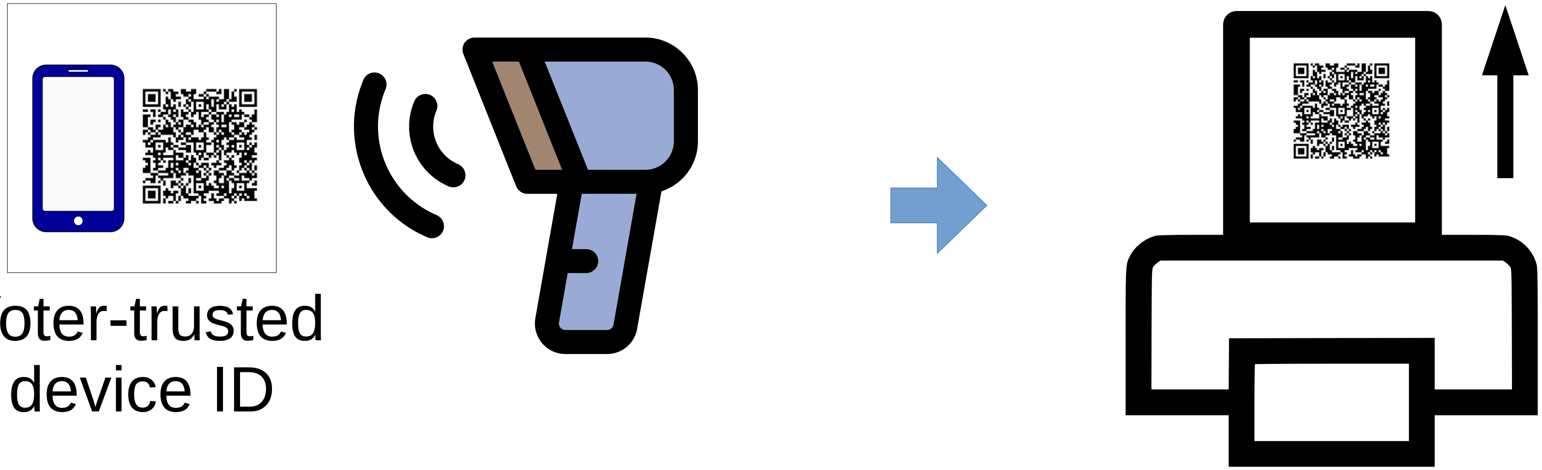
- Terminal asks for, scans voter’s check-in ticket



What happens in “privacy booth”?

Enter booth with check-in ticket, voting device(s) or printed device IDs (QR-coded public key)

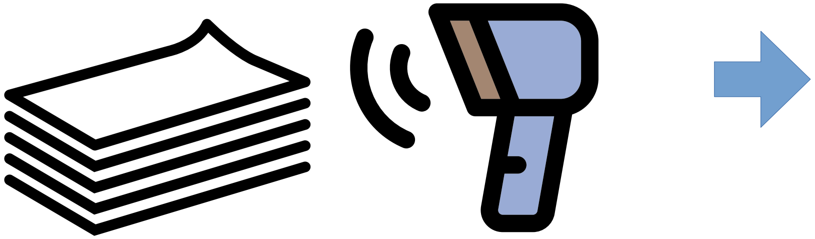
- Terminal asks for, scans voter’s check-in ticket
- Terminal scans device ID for **real** credential
  - Prints QR code on **first half** of receipt

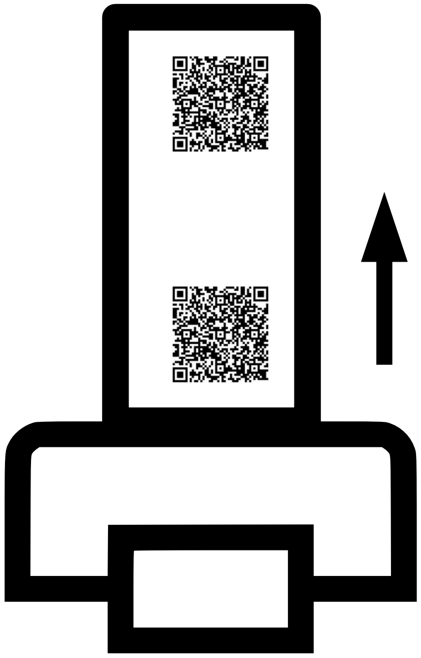


What happens in “privacy booth”?

Enter booth with check-in ticket, voting device(s) or printed device IDs (QR-coded public key)

- Terminal asks for, scans voter’s check-in ticket
- Terminal scans device ID for **real** credential
  - Prints QR code on **first half** of receipt
- Terminal asks user to choose and scan any **envelope** from stack
  - Prints QR code on **rest of** receipt

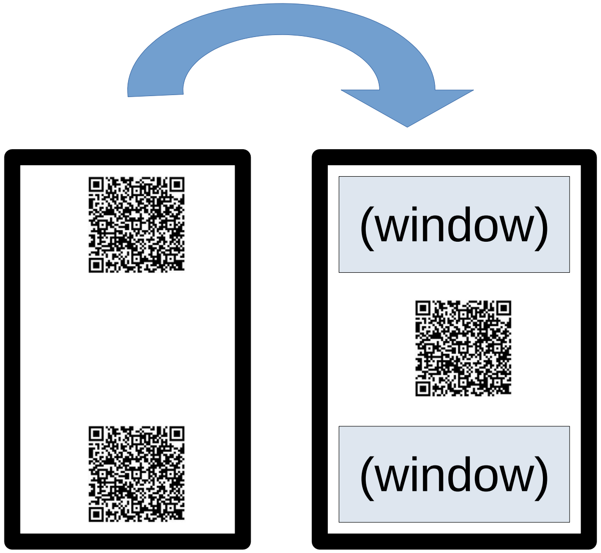




What happens in “privacy booth”?

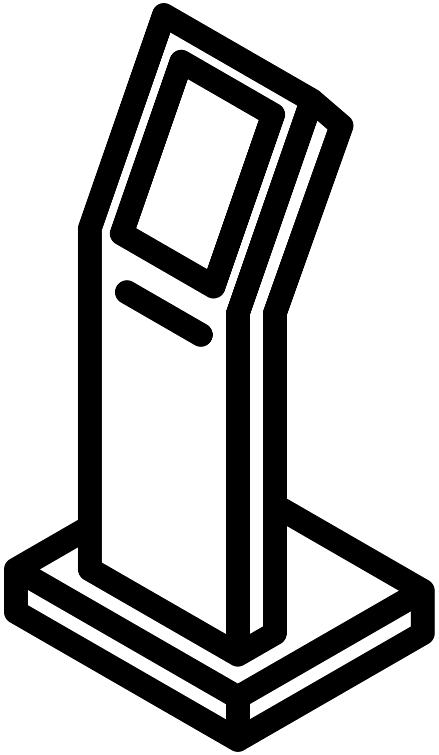
Enter booth with check-in ticket, voting device(s) or printed device IDs (QR-coded public key)

- Terminal asks for, scans voter’s check-in ticket
- Terminal scans device ID for **real** credential
  - Prints QR code on **first half** of receipt
- Terminal asks user to choose and scan any **envelope** from stack
  - Prints QR code on **rest of** receipt
- Terminal prompts user to insert **receipt** into **envelope**



What happens in “privacy booth”?

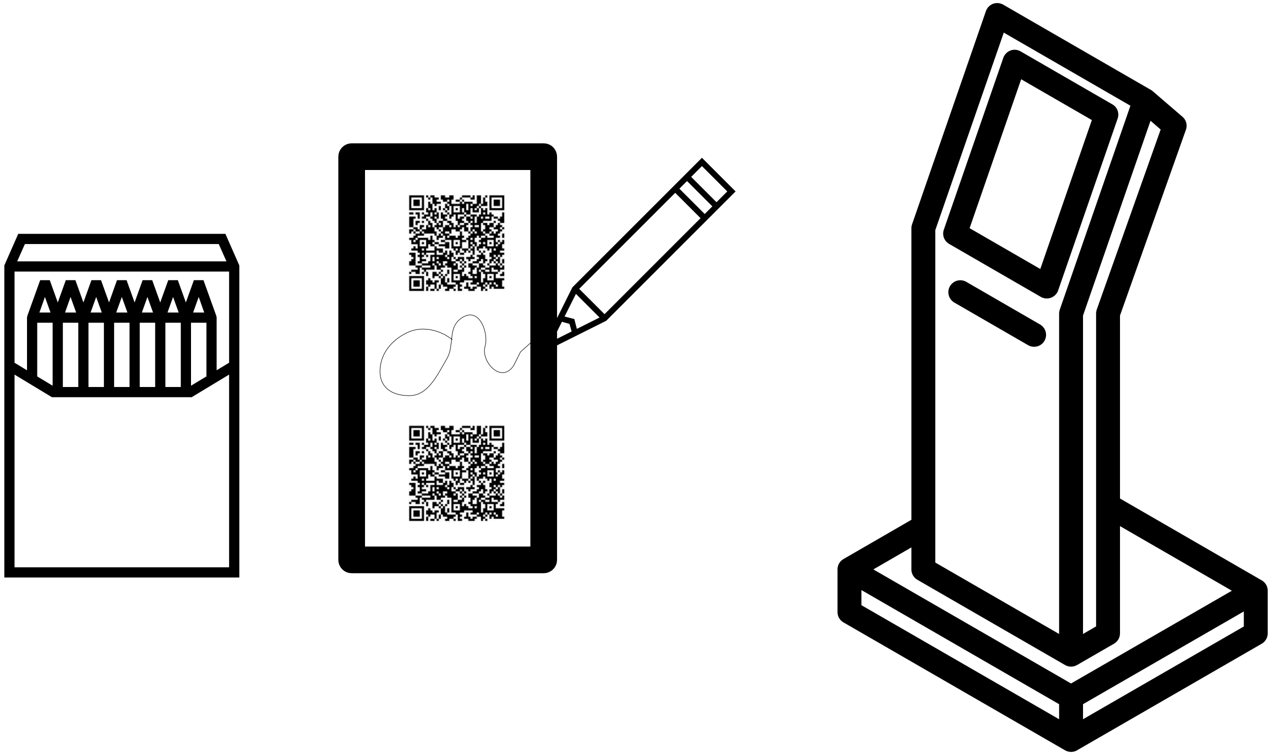
Ask if voter wants a **test credential**? If yes…



What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

- Ask voter to **mark** the **real credential** to help remember which it is



What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

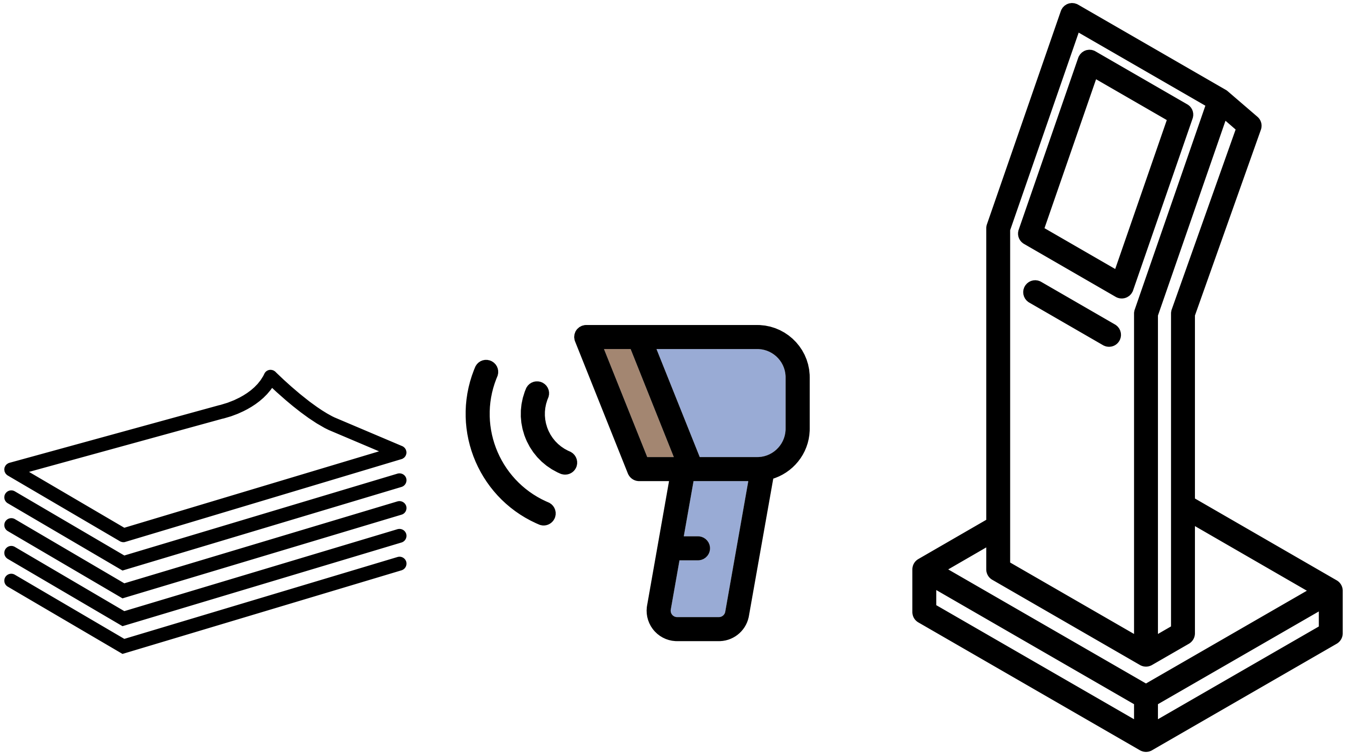
- Ask for, scan device ID for **test** credential
  - Coercer’s device ID if under coercion
  - Or kids’ devices, friends’, …

Coercer’s
device ID

What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

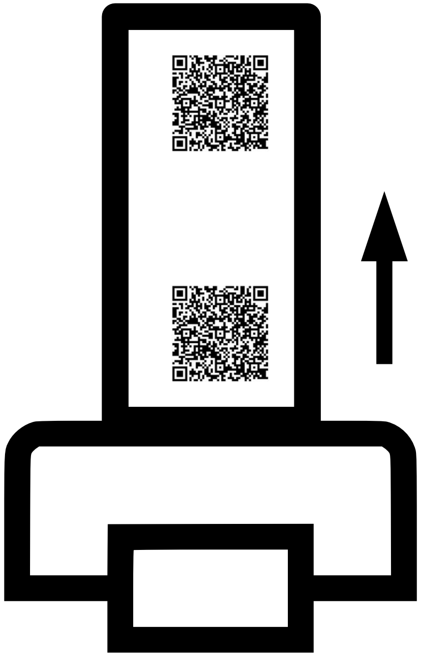
- Ask for, scan device ID for **test** credential
- Ask voter to choose and scan any **envelope**



What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

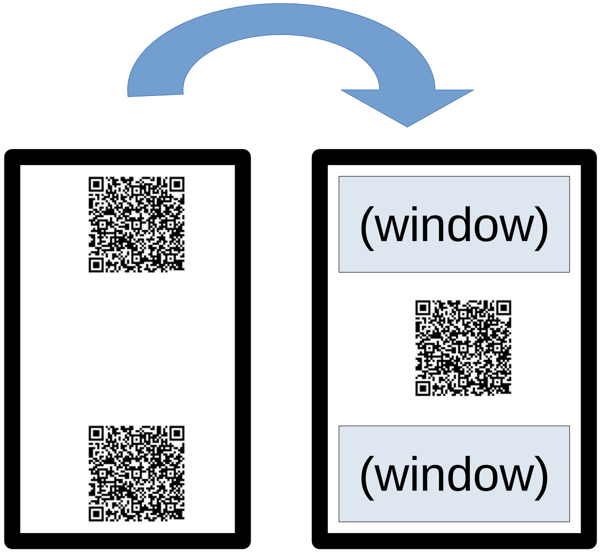
- Ask for, scan device ID for **test** credential
- Ask voter to choose and scan any **envelope**
- Print **entire receipt** at once



What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

- Ask for, scan device ID for **test credential**
- Ask voter to choose and scan any **envelope**
- Print **entire receipt** at once
- Ask user to insert **receipt** into **envelope**



What happens in “privacy booth”?

Ask if voter wants a **test credential**? If yes…

- Ask for, scan device ID for **test credential**
- Ask voter to choose and scan any **envelope**
- Print **entire receipt** at once
- Ask user to insert **receipt** into **envelope**

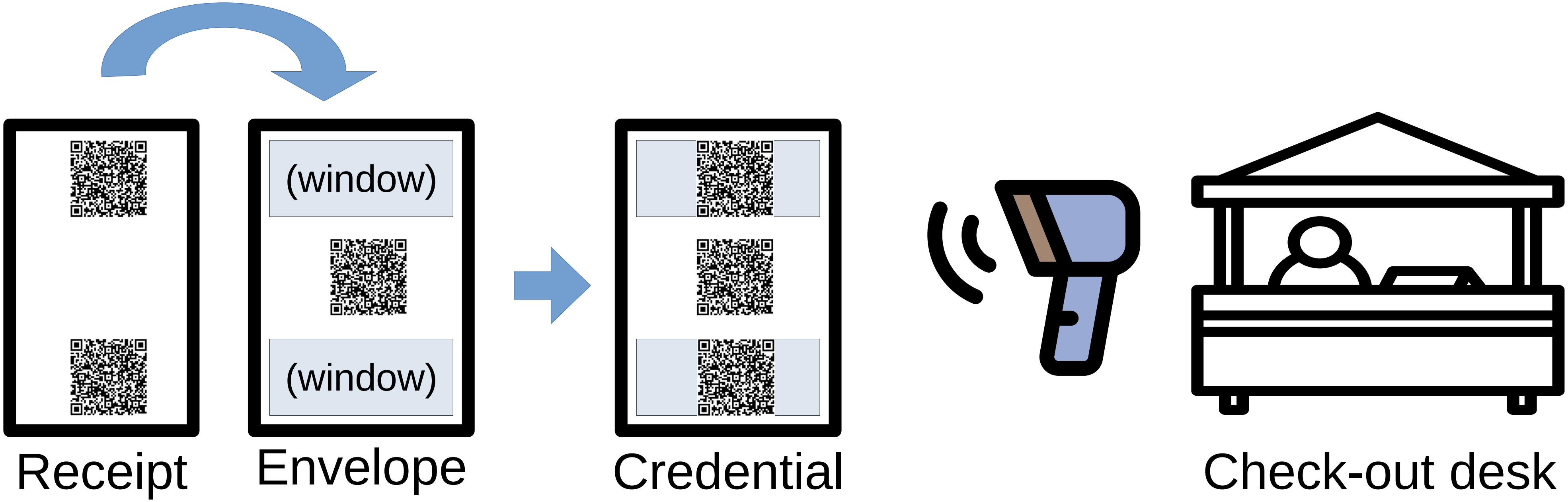
Ask if voter wants another **test credential?**

- If yes, repeat process above (to random quota)

Check-out and subsequent voting

Voter presents *any* credential (e.g., coercer’s) at check-out desk for official to scan

- Activates *all* credentials, real and fake



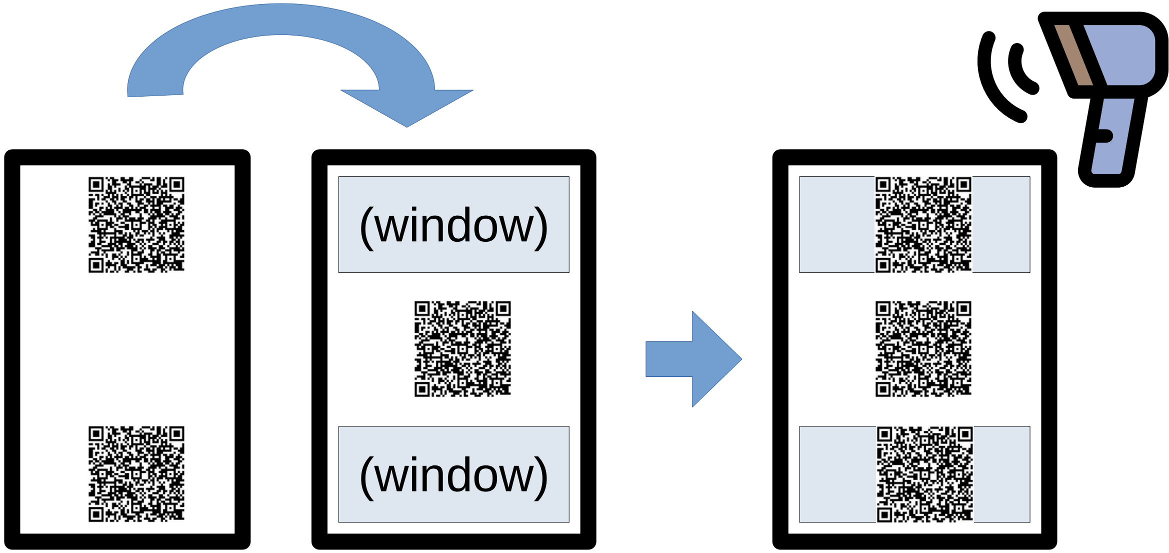
Check-out and subsequent voting

Voter presents *any* credential (e.g., coercer’s) at check-out desk for official to scan

- Activates *all* credentials, real and fake

At leisure, voter scans credential(s) with device(s)

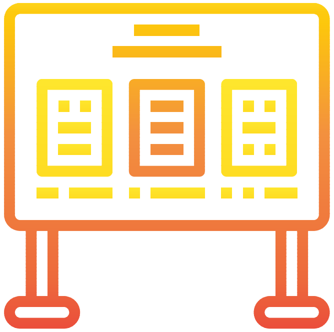
- **Any** device can check validity
- **Associated** device can cast votes, read prior votes



Talk Outline

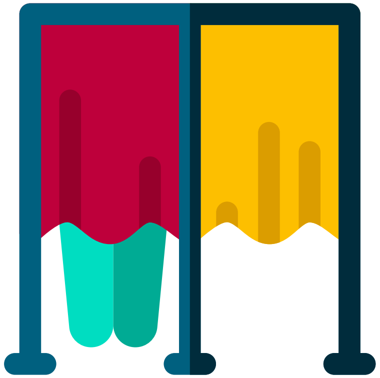
- Goal: free, people-centric self-governance
- Basics: approaches to coercion resistance
- Signup: governmental or decentralized voting
- Usability: how the user experiences signup
- **Technical: what actually happens underneath**

Votegral system components



Electronic bulletin board
(e.g., public blockchain)

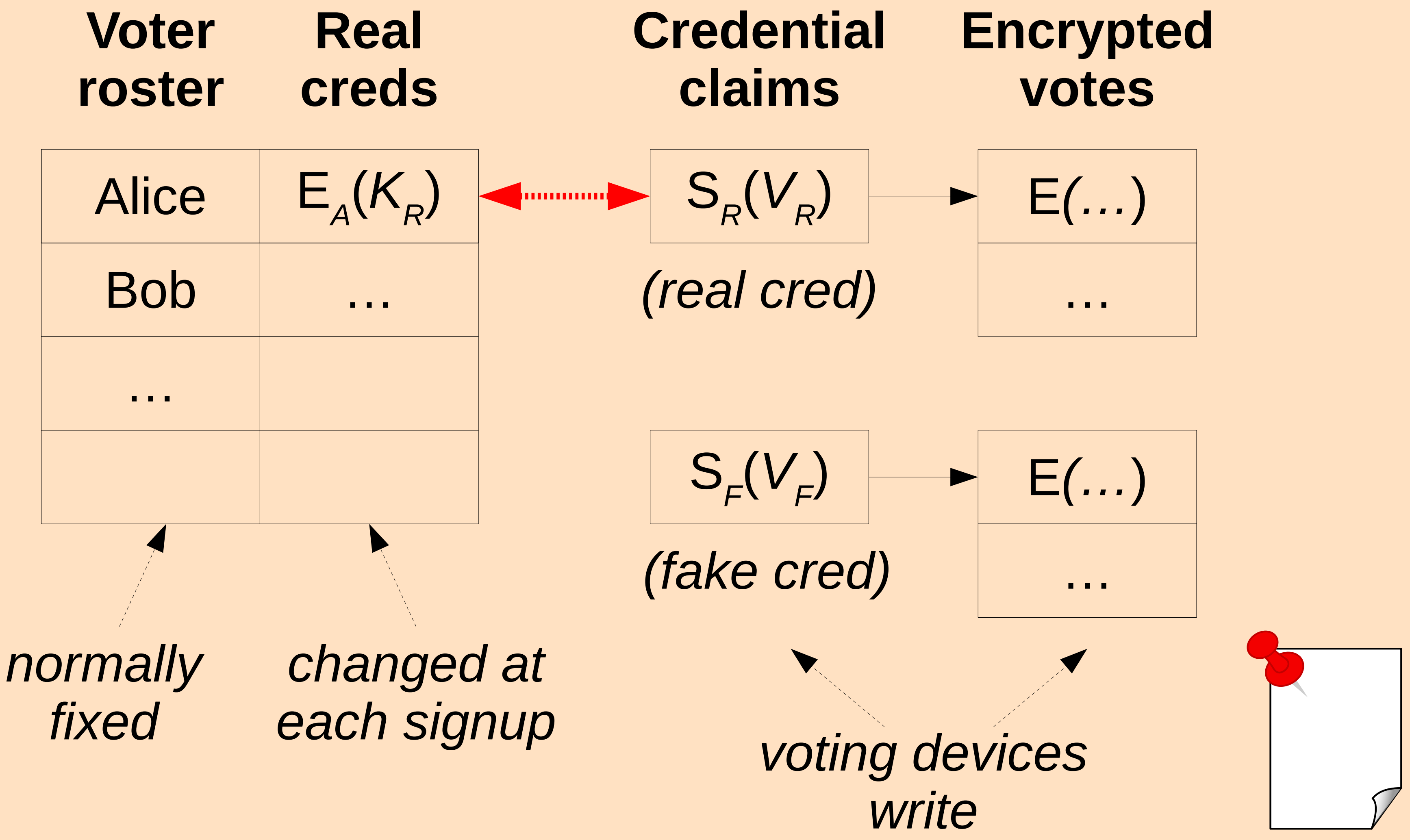
Authority/tally servers



Physical office/booth
with signup terminal

Voter’s personal device(s)

What’s on the bulletin board?



Voter roster	Real creds
Alice	$E_A(K_R)$
Bob	…
…	

Credential claims

$S_R(V_R)$

(real cred)

$S_F(V_F)$

(fake cred)

Encrypted votes

E(…)

…

E(…)

…

normally fixed

changed at each signup

voting devices write

What does signup device do?

Signup device trusted **only** for coercion resistance

- Air-gapped, sees no private info about voters, holds no secrets that can cast or decrypt votes

Signup produces 2 encryptions + interactive ZKP

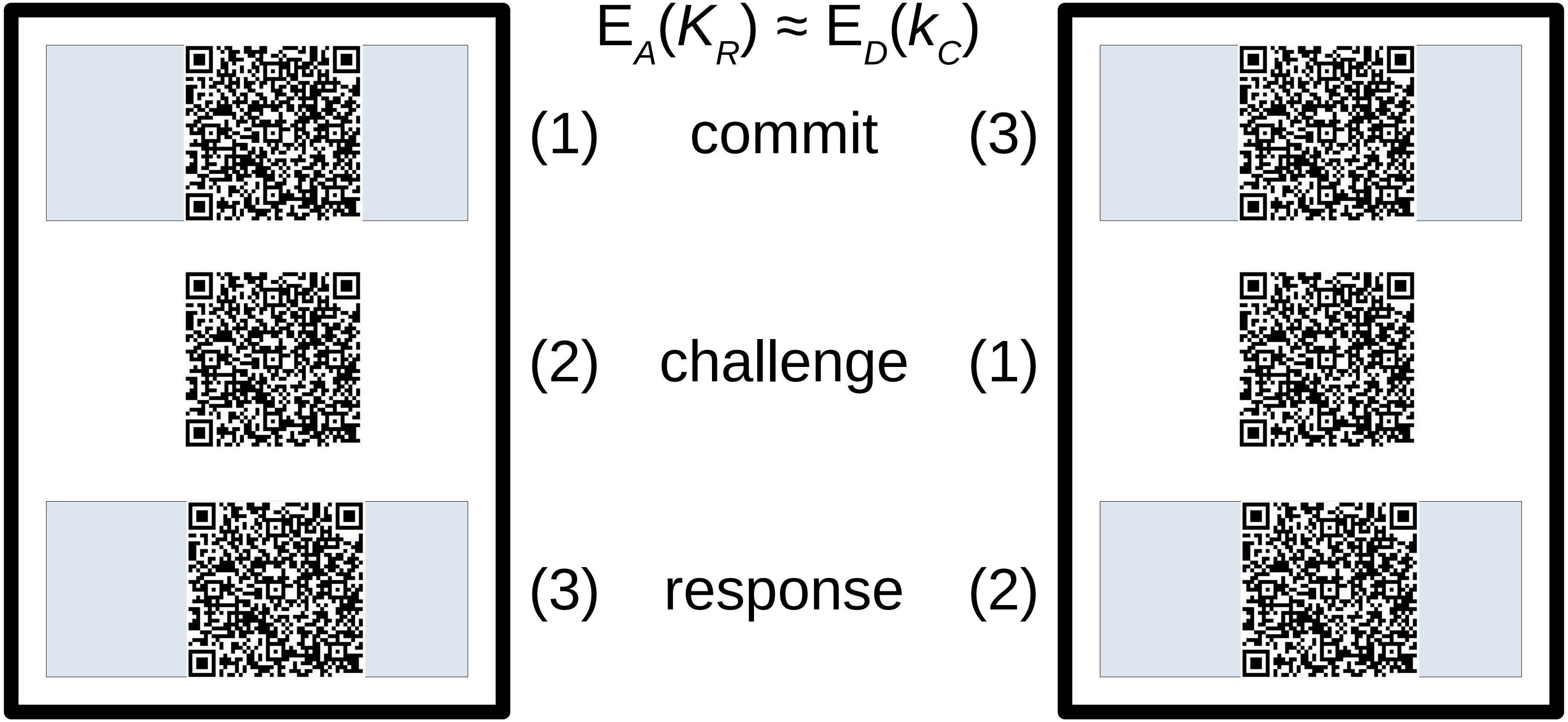
- $E_A(K_R)$: **Real** cred’s public key, Enc → **Authority**
  - Written to **voter roster entry** on BB at check-out
- $E_D(k_c)$: **This** cred’s private key, Enc → **Device**
  - Device can use **once** to create voting profile on BB
- ZKP: **Real** or **fake** interactive ZKP that $K_C = K_R$

Credentials have interactive ZKPs

Real or **fake** proofs that credential matches roster

- Distinguishable only via **order of generation**

$E_A(K_R) \approx E_D(k_C)$



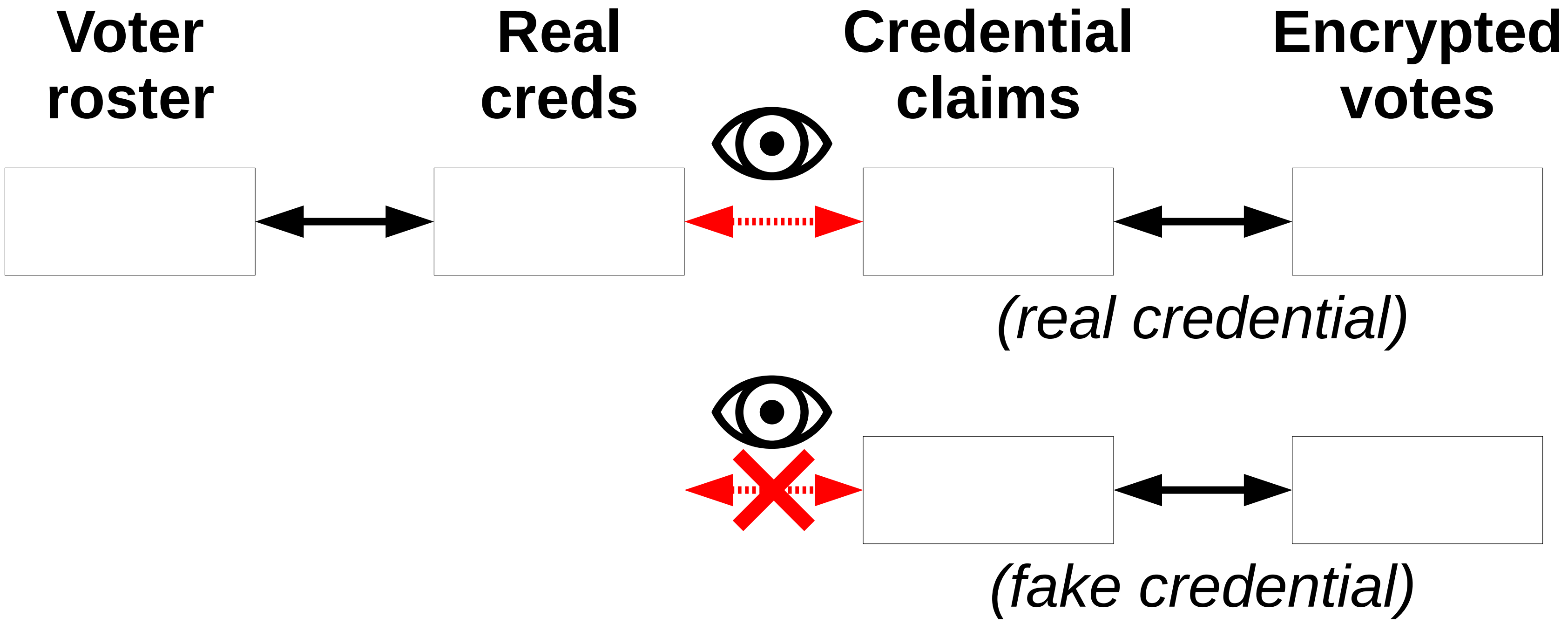
(1)	commit	(3)
(2)	challenge	(1)
(3)	response	(2)

Real credential **Fake credential**

End-to-end vote verification

Every step in signup + voting process is verifiable

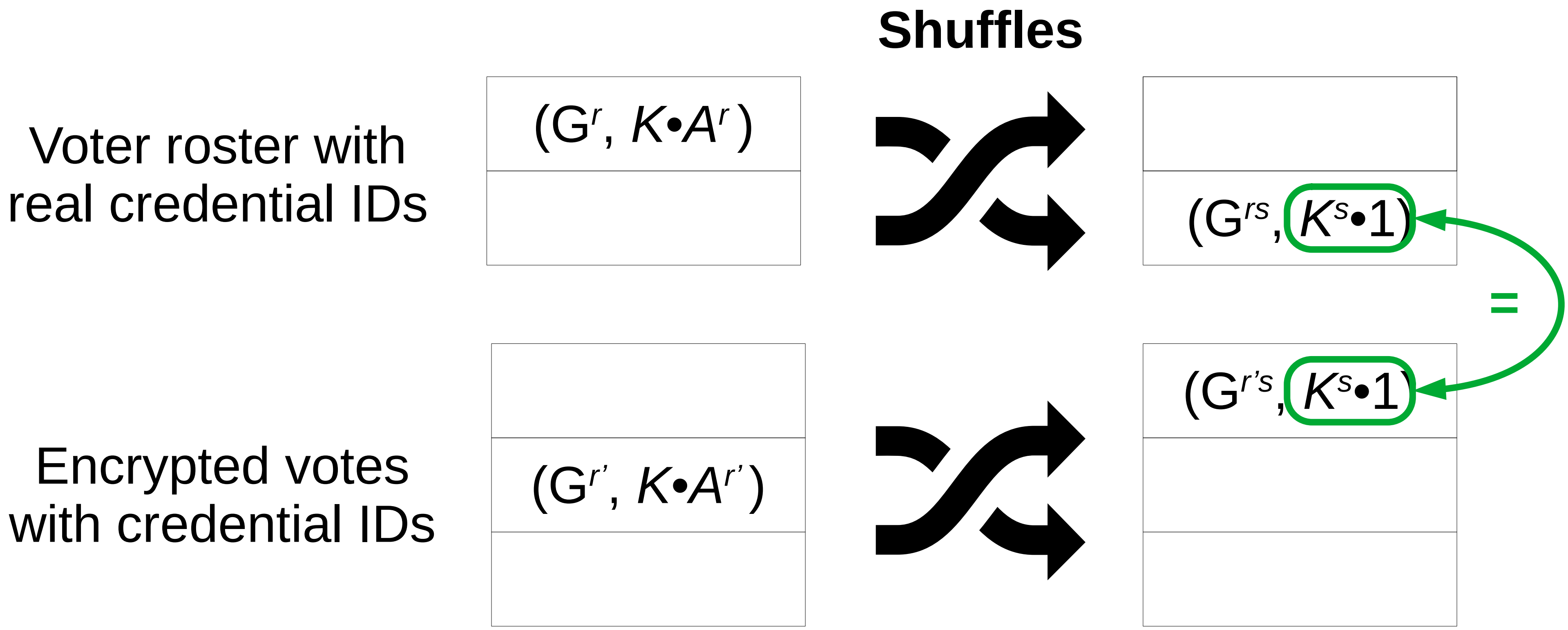
- But “critical link” of **voter roster** ↔ **cred claim** is verifiable only *interactively* by voter in private



Vote tallying process

Uses distributed rewriting of randomized ElGamal ciphertexts into convergent Pohlig-Hellman

- Fully verifiable, splittable, used in PSI protocols



Shuffles

Voter roster with real credential IDs

$(G^{r}, K \bullet A^{r})$

$(G^{rs}, K^{s} \bullet 1)$

=

Encrypted votes with credential IDs

$(G^{r'}, K \bullet A^{r'})$

$(G^{r's}, K^{s} \bullet 1)$

Vote tallying process

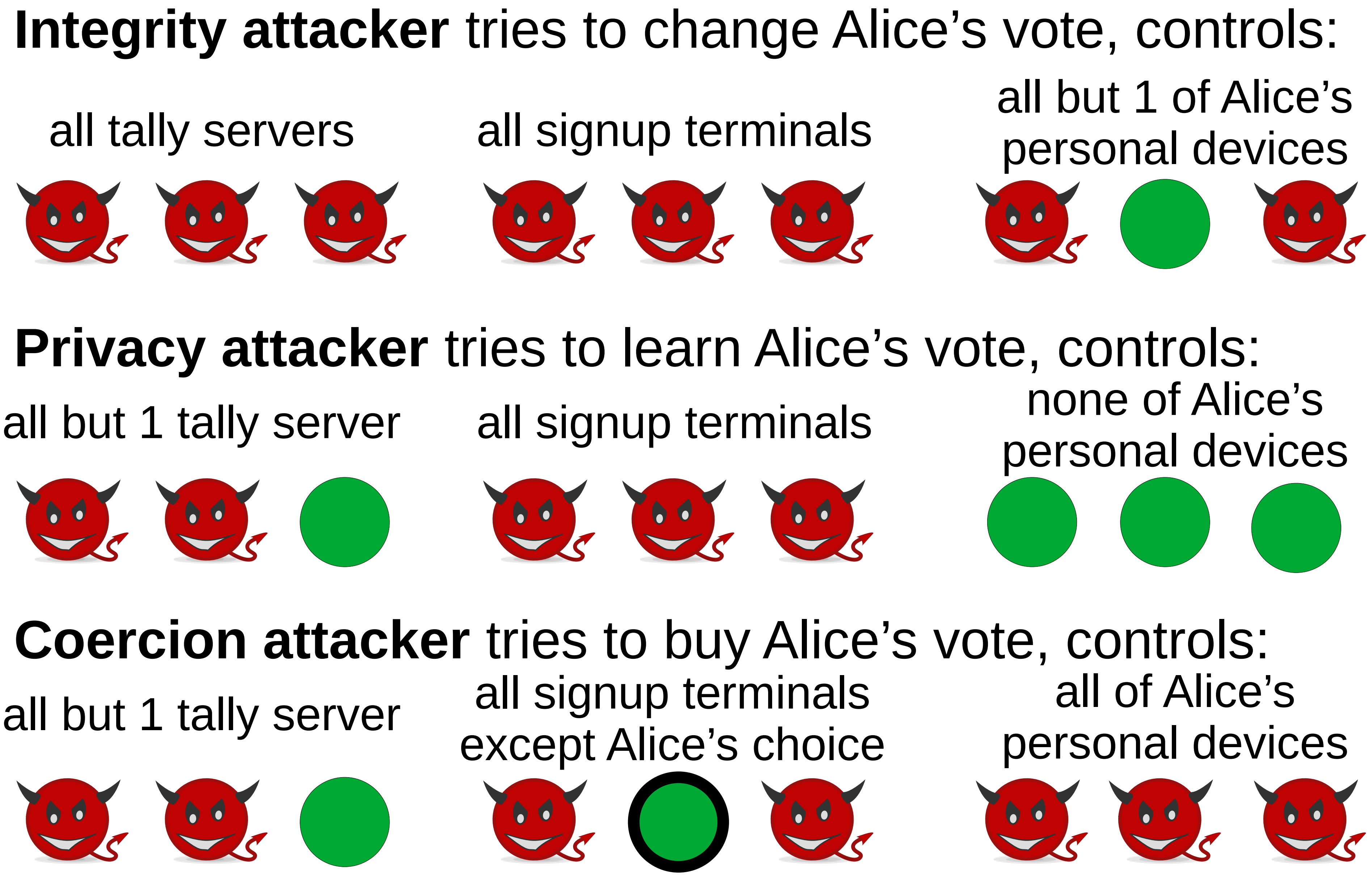
Uses distributed rewriting of randomized ElGamal ciphertexts into convergent Pohlig-Hellman

- Fully verifiable, splittable, used in PSI protocols

Useful properties:

- Naturally linear-time: just match output cred IDs
- Doesn’t leak whether a given voter cast a vote
- Supports well-known keys, e.g., party-line votes

Threat model summary



Integrity attacker tries to change Alice’s vote, controls:

all tally servers: 😈 😈 😈

all signup terminals: 😈 😈 😈

all but 1 of Alice’s personal devices: 😈 🟢 😈

Privacy attacker tries to learn Alice’s vote, controls:

all but 1 tally server: 😈 😈 🟢

all signup terminals: 😈 😈 😈

none of Alice’s personal devices: 🟢 🟢 🟢

Coercion attacker tries to buy Alice’s vote, controls:

all but 1 tally server: 😈 😈 🟢

all signup terminals except Alice’s choice: 😈 🟢 😈

all of Alice’s personal devices: 😈 😈 😈

Summary of relevant features

- End-to-end **verifiability**, minimize required trust
- Coercion-resistant signup via **interactive ZKPs**
- Signup devices **untrusted** for integrity, privacy
- Credentials have **no toxic waste**: discardable
- Linear-time tallying with **last-minute roster**
- Limited credentials per user → **no BB flooding**
- If anything goes wrong, just **signup again**

Votegral: Conclusion

Adapts in-person voting-like process for signup: make coercion-resistant *choice of voting channel*

- Supports governmental or decentralized voting
- Voters get real and fake credentials at signup
  - Learn which is which only **interactively** in private
- Use to vote in multiple subsequent elections
  - Fully-dematerialized voting, check on other devices
- End-to-end verifiability, minimally trusted signup

We **can** make coercion resistance secure, usable!