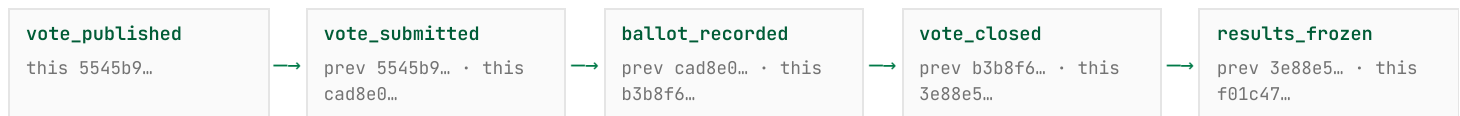


Elections built to survive a challenge.

Every election here writes a permanent, verifiable record of itself as it runs. This page explains the machinery in plain terms. **The claim is not that tampering is impossible. The claim is that tampering cannot succeed silently.**

EVERY ACTION BECOMES A LINK IN A CHAIN

When anything happens in an election, the database itself writes an entry: the ballot goes live, a member votes, a deadline moves, the count freezes. Each entry gets a SHA-256 fingerprint, and **each fingerprint includes the fingerprint of the entry before it**. Change any entry, even one character, and its fingerprint no longer matches, and every entry after it points at a fingerprint that no longer exists. The log is append-only: the database refuses edits and deletions of past entries for everyone, including us. We tested that refusal against our own administrator access.



Verification recomputes every fingerprint. It runs from a button on the election dashboard, on every certificate, and automatically every night.

EVERY BALLOT IS FINGERPRINTED THE MOMENT IT IS CAST

In the same instant a ballot is stored, the database computes a fingerprint of its content and commits it into the chain. The fingerprint is salted with a secret held on the ballot record, so **reading the log never reveals how anyone voted**. From then on, anyone with access can recompute the stored ballot against the cast-time promise. We ran the drill in production: edited a stored answer directly in the database, and verification named the exact ballot that no longer matched.

THE CHANGES PEOPLE WORRY ABOUT ARE WATCHED

Disputes rarely start with forged ballots. They start with a moved deadline or a changed weight. These entries are written by triggers inside the database. **No application code path can skip them.**

vote_settings_changed

Deadline, quorum, eligibility, or secrecy changed on a live election, with old and new values.

voter_record_changed

A voter's weight or contact identity changed mid-election, flagged with whether they had voted.

voter_status_reverted

A member who had voted was marked as not having voted. No legitimate flow does this.

voter_record_removed

A member was removed from a running election's roster, including whether they had already voted.

SEALED ELECTIONS SEPARATE WHO FROM WHAT, PERMANENTLY

Where statutes require it (California and Florida association elections among them), a sealed ballot is stored with **no reference to the voter at all**. The poll book records that a member voted. The ballot box holds ballots. Nothing joins the two, and the chain names who voted, never which ballot. At close the box is shuffled and rewritten in random order, and a checksum of the whole box, computed before and after, must match or the close refuses to happen. That checksum enters the chain and every later check recomputes the box against it. The tally opens only after close, for everyone, including the organizer.

CLOSING FREEZES THE COUNT, AND A NAMED PERSON SIGNS IT

At close the results are computed once and frozen with their own SHA-256 fingerprint. The database refuses any later change to them. Where an inspector of elections is appointed, a named person certifies, and their signed certification records the results fingerprint, the ballot count, and **the head fingerprint of the entire chain at the moment of signing.**

WHO CAN CHANGE WHAT

IF THIS PERSON TRIED...	WHAT THEY CAN DO	WHAT THEY CANNOT DO QUIETLY
A voter	Cast their own ballot, hold their own confirmation.	Vote twice, vote as someone else, or see anyone else's ballot.
An organizer	Run the election: roster, schedule, deadline moves, record fixes.	Change any of that invisibly. Every sensitive change lands in the chain with old and new values. Ballot content and frozen results are never editable.
vote.direct, operating the platform	Everything an organizer can, plus chain-logged support access.	Edit or delete a past log entry. The database refuses the write, for us too, and a nightly sweep re-verifies every election and alerts on any anomaly.
Total database control	In theory: rewrite everything and recompute every fingerprint.	Make the rewrite agree with the certificates, audit packages, signed certifications, and confirmations already in other people's hands. Copies outside the database are the final control.

WHAT WE WILL NOT DO

- ✗ **No receipt proves how you voted.** A receipt that proves content is a tool for pressuring and buying votes, so we do not issue one.
- ✗ **No live tally on sealed elections.** Watching a running tally against the poll book is a way to unmask voters, so it does not exist.
- ✗ **No editing our own audit log.** There is no admin tool for it. The database refuses the write at a layer below our application.
- ✗ **No "trust us."** Verification is a button on the dashboard, a fingerprint on the certificate, and documents in your hands.

THE RECORDS YOU CAN HOLD

Results certificate. Outcome, participation, verification methods, the chain head fingerprint, and integrity status. Made for the board packet.

Inspector certification. A named person's signed statement binding the results fingerprint and the chain head.

Audit package. The full chained log, reconciliations, ballot fingerprint checks, and the sealed box comparison, sealed by its own fingerprint.

Voter confirmation. Each voter's own proof their ballot was recorded, held by them, not by us.