



Deep Well Labs

INDEPENDENT RESEARCH + ENGINEERING

Chronicle Verification Technical Brief

The technical companion to the integrity brief, for a reader who intends to check.

Public edition. This document states the company's position and the reasoning behind it. It is a working document, not a specification, and nothing in it is a claim that a described capability is implemented. Where Deep Well Labs describes what could become possible, that is research; where Origo describes what it does, that is product. The distinction is the subject of the Boundary Charter.

SEPTEMBER 2026

DEEP WELL LABS, INC.

DEEPWELLLABS.COM

What was checked, and when

On 1 September 2026 at 17:10:11.777Z, a Chronicle chain of 264 preserved entries was published: the head of the chain was folded into a Merkle root, that root was signed, and the signature was submitted to two independent RFC 3161 timestamp authorities, both of which answered.

On 3 September 2026 the resulting pair of files, the export and its receipt bundle, was checked outside the system that produced them. The result was `verified`, through 1 September 2026 at 17:10:12.000Z.

Nothing was consulted to reach that result except those two files. No account, no network request, no call back to Origo.

The tool that checked it

A single file of about 95 kilobytes. No dependencies, no installation, no build step, and no network access of any kind. It runs in a browser opened from a downloads folder, or from a USB stick, with the machine disconnected. A copy ships inside every Chronicle export, so the evidence and the means of checking it stay together.

It shares no code with the system that produces the evidence. That is a requirement rather than an accident. A checker built from the producer's own encoder cannot detect a mistake in that encoder, because both halves would be wrong in the same direction and would agree with each other. The two implementations were written separately, from the specification and from the relevant RFCs, and the duplication between them is the mechanism.

What it refuses

A tool that only ever says yes has not been tested. Each of the following was demonstrated, and each is held in place by a test that fails if the behaviour regresses:

- a record altered after preservation, and a record whose capture provenance was relabelled while its text was left untouched;
- an entry removed from the middle of a chain, a chain renumbered, and entries from one chain presented under another chain's identity;
- a signed root that verifies against a key nobody accepts;
- a timestamp that covers a different signed statement, which is the shape that back-dating actually takes: obtain a timestamp, then change what it is presented beside;
- a timestamp whose own contents were altered after issue;
- a timestamp from an authority the reader has not accepted, which is reported as unadjudicated rather than either trusted or rejected.

The last of those matters more than it looks. Timestamps are cheap and anyone can obtain one over any bytes at all. A checker that accepted whichever authority a file happened to name would confirm evidence its own author had manufactured.

The check depends on inputs the evidence cannot supply

Two questions cannot be answered from the files alone: whether the key that signed the root is one anybody accepts, and whether the authority that timestamped it is. Both answers are supplied to the tool. Neither is read from the bundle.

This was demonstrated rather than asserted. Withholding either input, and changing nothing else, moves the result from `verified` to `incomplete`: no failure is reported, because nothing is wrong with the record, and no date is reported, because none was established. A result that a file could produce for itself would be worth nothing, so the check was run in a configuration where it could not.

What this does not establish

Stated here rather than left for a reader to discover.

It does not establish that a human wrote the content, that the content is true, who preserved it, or when the thought occurred. Chronicle is tamper-evident and never tamper-proof: a party holding the keys to the system can rebuild a chain, and this evidence makes that detectable rather than impossible.

Coverage is not continuous. A record is covered from the moment it is included in a publication, and publication runs on a schedule, so there is a window between one publication and the next. Records preserved before the first publication on 1 September 2026 hold a position in their chain that was never independently timestamped.

Two authorities answered at publication. One of them can currently be adjudicated by the shipped tool, because the other operates a root that is in no public trust store. The redundancy exists in the publication and does not yet exist in the checking.

The receipt bundle is produced today by the founder-operated export, which is the only export route Origo offers. A self-serve route does not exist yet.

Origo is in private alpha and not yet generally available.

Checking this document

The verification described above is recorded, with the exact command, the verbatim output, the exit code, the trust inputs and the discrepancies, in `chronicle-independent-verification-2026-09-03.md`. The export it was run against has SHA-256 `14529b5b5c0d14adbc01b4a6b34db8df29132824f7f9d33beed83003b802df34`, and is the same file, unchanged, that returned an incomplete result on 1 September 2026 when the checking procedure was only partly implemented.

The tool is `apps/chronicle-verify` in the Origo repository. The specification it was written from is `packages/chronicle-core/contracts/integrity-evidence-v1.md`. The cases it must reject are listed in the conformance document beside it, and the fixtures that exercise them are published with real

computed values rather than illustrative ones, so a third implementation can be checked against the same corpus.

Deep Well Labs, Inc. advances the science and infrastructure of human thought. Origo is a product of Deep Well Labs, Inc. This public edition is derived from the company's internal source document and may be revised. deepwelllabs.com · phil@deepwelllabs.com