C17: Proof-of-Unchanged Global Application Matrix

Custody-Boundary Verification Methodology

Anchoring Software: AuditLog.AI **Auditing Software:** QMS Auditor **Version:** v5
Mode: Zero-Custody | Human-Verified | Machine-Deterministic | Time-Anchored

Protocol Originator: Dr. Fernando Telles
Date: 06 February 2026
AI_used: true
LLM_used: LLM1<>LLM4 **Human_verified:** true HV_FT
Classification: Public methodology document | Hash-only; zero-custody **Linked Canonicals:** C1 SIASE/SNTPTC Unified Governance | C5 Legal, Ethics, Compliance Enforcement | C9.5 Zero Custody Reproducibility Protocol | C12 AuditLog.AI Global Compliance Matrix | C14 Bitcoin Zero-Custody Protocol (Stage IV Hash-Only Reproducibility) | C15 Zero-Custody Data Transfer Audit Execution Protocol | C16 Zero Authority Audit Protocol

> **One-sentence summary:** Proof-of-Unchanged is a custody-boundary verification methodology that deterministically answers one question: **Has this evidence changed since the last verified checkpoint?**

---

1) Purpose

This document defines a **domain-agnostic** method for proving whether exported digital evidence has remained byte-unchanged across time and custody transitions, and for cryptographically enumerating which evidence and/or memberships diverge when it has not.

It is designed to be portable across:

> **Clinical trials / CRO inspection readiness**

> **Audit & assurance (Big Four, PCAOB / ISA frameworks)**

> **AI governance / cloud integrity / regulated industry evidence retention**

> **Legal / forensics / evidentiary continuity**

Any environment where evidence must survive **movement**, **retention**, and **format change** without ambiguity.

---

2) Scope (what this model does, and what it does not)

What it does

- Produces **deterministic Proof-of-Unchanged** for a paired-comparison of time-anchored evidence sets at a defined checkpoint (T_k vs T_n).
- Detects change as **byte-level divergence** relative to the referenced canonical state at the evidence layer and Evidence Set Fingerprint (ESF) membership layer.
- When change exists, returns **deterministic divergence enumeration** that is informational and exists solely to bound proportional human investigation.
- Supports **repeatable re-verification** at any later time (T_n) against the last anchored canonical state (T_k).

What it does not do

- Does not evaluate **meaning**, **correctness**, **completeness**, or **compliance** of the evidence contents.
- Does not infer intent, misconduct, or SOP failure.
- Does not prevent modification; it **detects** modification relative to a prior state.
- Does not require integration into source systems; verification occurs **at custody boundaries**, not inside operating systems.

---

3) Core Definitions

Evidence Artefact

> A file or object that is treated as audit evidence once it exits a source system, **immediately post-export** (e.g., export bundle, report, dataset snapshot, model artifact, workpaper evidence).

Custody Boundary

A point in time where evidence transitions between states or environments, for example:

- immediately post-export from a source system (T_0),
- ingestion into archive storage,
- transfer between storage tiers,
- transformation by SOP (compression, packaging, normalization),
- movement to another custodian or jurisdiction,
- preparation for inspection/audit submission.

Canonical State (T_k)

A frozen, verifiable checkpoint for an evidence set, represented by: - a deterministic manifest (membership and identifiers), and

- cryptographic digests (hashes), and
- optional decentralized time attestation (OpenTimestamps; `.hash.ots`),
- and a public anchor reference (hash-only; Bitcoin Transaction ID).

Proof-of-Unchanged (PASS)

A deterministic, pairwise statement relative to a specific canonical reference:

> **The evidence and membership bytes at time T_n are identical to the canonical state at time T_k.**

Divergence enumeration (informational)

A deterministic statement:

> **The evidence and/or membership bytes at time T_n differ from the canonical state at time T_k, with divergences enumerated to bound review scope.**

Important: Divergence is **not** a conclusion about compliance, error, or intent. It is a directional signal for proportional human effort.

---

4) The Custody-Boundary Model

Principle

> **Verification occurs at custody boundaries, not inside systems.**

This neutralizes integration risk and avoids dependence on vendor APIs, custody claims, or internal system trust.

Minimal operational position

- Evidence remains **local** (zero-custody).
- Only **hashes / digests** are used for public anchoring.
- Verification is deterministic and repeatable by any verifier with access to either the respective hash-only export packet; or read-only access to the same evidence artefacts on the custodian's node.

---

5) Canonical State Cycle (T_0 → T_n)

5.1 Canonical Anchoring Cycle Overview

1. **T_0 (Export boundary):** evidence leaves a source system.
2. **Freeze locally:** create a stable evidence bundle for long-term verification, protected against **accidental modification** only; intentional alteration is outside the prevention scope and is detected through verification.
3. **Timestamp attestation (UTC):** UTC suffix is recorded on all frozen filenames (`evidence___YYYYMMDDTHHMMSSZ.pdf`)
4. **Deterministic hashing:** compute scope-tiered digests using SHA-256(evidence bytes) and RIPEMD-160(SHA-256), recorded as `.hash` and `.2ha` sidecars at the evidence, batch, log, and session levels.
5. **Optional decentralized time attestation:** OpenTimestamps proofs (hash-only; `.hash.ots`)
6. **Anchor hash-only state:** publish a compact, session-level digest reference (`User Node (hashes-only) ──▶ AuditLog.AI Node (OP_RETURN) ──▶ Bitcoin Mainnet`)

5.2 Canonical Verification Cycle Overview

1. **Re-anchor at any time (T_n):** After elapsed time, re-execute *Anchoring Cycle* on last canonical state T_k, to produce comparison state T_n
2. **Export hash-only packets:** Zero-custody with no identifiers
3. **Verify by paired-comparison (T_k vs T_n):** Generate HVT-A machine-deterministic reports comparing T_k against T_n for human verification.

5.3 Handling legitimate transformations (T_1 → T_2 pattern)

Hash-based verification remains stable under routine cross-platform copying and storage of unchanged files. However, divergence arises when workflows intentionally or unintentionally rewrite byte-level representations, such as line-ending normalization, compression, lossy format conversion or archive rehydration that regenerates files.

Real workflows and archival migration often require transformations. Standard operating procedure versioning, time-dependent serialization, and cloud rehydration workflows may legitimately change evidence bytes. Such scenarios should be interpreted against predefined normalization and archival procedures rather than as evidence of misconduct.

Rule:

> If an SOP intentionally transforms evidence bytes, treat the transformation as a new canonical state.

Standard sequence:

- **Verify pre-transform** (against T_k) → establish Proof-of-Unchanged at T_1

- **Transform under SOP** → bytes change as intended
- **Immediately re-freeze and re-anchor** → establish new canonical state T_2
- **Future verification** compares against T_2 (not T_0)

This preserves **evidentiary continuity** without blocking lawful or necessary operations.

6) Custody-Boundary Application Matrix

Phase	Trigger (Custody Boundary)	Action (Local, deterministic)	Outcome	Example (Clinical Trials / CRO)	Example (Audit & Assurance / Big Four)	Example (AI Governance / Big Tech)
Establish T_0	Evidence leaves source system	Freeze → dual-hash → anchor	Canonical state created	Database lock export (eTMF/EDC extracts)	Receipt of external electronic info	Model artifact / checkpoint export
Retention	Evidence stored unchanged	Optional re-verify	PASS confirms unchanged	Retention prior to inspection	Workpaper retention	Dataset/ model retention
Transformation	SOP changes bytes (packaging/ compression/migration)	Verify (T_1) → transform → re-freeze → re-dual-hash → re-anchor (T_2)	New canonical state	Submission packaging / archival migration	Conversion for long-term archive	Model quantization/compression
Challenge / audit	Inspection, audit, dispute	Re-verify against latest canonical	PASS or divergence enumerated	FDA/EMA/ TGA inspection readiness	PCAOB/ISA audit documentation	EU AI Act / internal audit
Long-term archiving	System changes, storage tier shift	Repeat T_k cycle	Continuous provenance preserved	Multi-year retention	7-year retention	Multi-version governance

7) Language Mapping Across Domains

Concept	Clinical Trials / CRO	Audit & Assurance / Big Four	AI Governance / Big Tech
Canonical state	Frozen export / baseline	Audit evidence set	Model artifact / dataset checkpoint
PASS	Verification confirmed	No integrity exception	Integrity verified
Divergence enumerated	SOP-level delta to review	Exception requiring root cause	Drift / delta requiring attribution
Custody boundary	Export / archive / submission	Receipt / retention / assembly	Training / deployment / governance gate
Proportional review	Inspection readiness scoping	Materiality-driven investigation	Risk-based audit response

---

8) Interpretation Rule (non-accusatory)

Canonical claim

If bytes match the canonical state, the evidence and membership are provably unchanged.

> **Proof-of-Unchanged is the only canonical claim.**

Divergence rule

A divergence outcome:

- is **informational**,
- does not imply wrongdoing,
- does not imply non-compliance,
- and exists to **direct human effort proportionally** toward the minimal set of deltas.

> **Audit doctrine:** Verification establishes integrity facts; interpretation, materiality, and response remain exclusively human and institutional responsibilities.

PASS reduces reconstructive work.
Divergence bounds reconstructive work.

This methodology makes no representations regarding regulatory compliance, audit opinions, or legal sufficiency; it establishes only cryptographic integrity facts.

---

9) Security and Custody Posture

- **Zero-custody:** evidence remains local; no PHI/PII or trial content is required to leave the environment.
- **Hash-only anchoring:** only compact cryptographic digests are anchored publicly.
- **Detection, not prevention:** filesystem controls can reduce only **inadvertent** modification through local frozen protections; verification detects changes regardless of cause.

---

10) Regulatory and Assurance Positioning

Proof-of-Unchanged operates as per `C12: AuditLog.AI Global Compliance Matrix`, under **electronic records** and **audit documentation** frameworks such as:

- FDA **21 CFR Part 11**
- EMA **Annex 11** + GCP Guideline Integration (2023)
- TGA / PIC/S **PE 009-17**
- PCAOB **AS 1105 / AS 1215** *(including AS 1105.10A External electronic information reliability evaluation (effective 2025)*
- ISA **230 / 500 / 240 (Revised 2025)**

This methodology is **not** clinical decision support and does not provide patient-level treatment recommendations.

---

11) Public Verification References (Ordinals public reproducibility ledger; DOI)

> Format: **Block; Payload. Bitcoin TXID**

Ordinal 04 Anchors the world's first auditable reproducible infrastructure for ethical, verifiable auditability enforced by zero-custody cryptographic proof, Bitcoin anchoring, and mandatory human governance. Patent protected (Global priority date 17 June 2025), US Provisional #63/826,381; AU Provisional #2025902482.

> 902895; The world's first auditable AI-Human Synergy infrastructure with enforced ethics, cryptographic proof, and Bitcoin anchoring. Bitcoin `9e70b9510ce64ed53ee5565a114fe96a79b59058499efa1fa400c1155d490986`

Telles F. Sentinel Protocol v3.0: AI-Human Synergy Infrastructure Technical Summary for Intellectual Property & Strategic Briefing. Zenodo; 2025. https://doi.org/10.5281/zenodo.15795253

Ordinal 05 Infrastructure Reproducibility Audit Log. Validator-signed, cryptographically anchored audit for SENTINFRA-SESS001. Enforces zero-custody reproducibility via Bitcoin timestamps and hashes, no trust required.

> 907720; Sentinel Protocol v3.1 Infrastructure Pre-Public Deployment Audit Log. Bitcoin
> `ae198274a00abbb8296a3b9412e6fd3a62360bcf062e000fa2908d8f3b90e803`

Telles, Fernando. Sentinel Protocol v3.1: Infrastructure Reproducibility and Public Verification Log. Zenodo; 2025.
https://doi.org/10.5281/zenodo.16607606

Ordinal 11 This document forms part of the AuditLog.AI Global Regulatory Submission Package (v4.0) providing immutable, reproducible proof of execution for validation, compliance, and quality-assurance evidence across FDA, EMA, TGA, PCAOB, and ISA frameworks.

> 921109; ORDINAL11lb0c8b2223eab705cac6bf7801b64945f7ed47022lfc46851b. Bitcoin
> `6754818e57d5dcc16a7fccdb1d36bb213cd39ae4a180e17923387738f38a3f41`

Telles, Fernando. AuditLog.AI Runtime Execution and System Validation Evidence Dossier End-to-End Operational Proof During Regulatory Global Submission. CDA AI Pty Ltd, October 2025.
https://doi.org/10.5281/zenodo.17460850

Ordinal 12 C12 AuditLog.AI Global Compliance Matrix provides the harmonized cross-reference between AuditLog.AI's live runtime execution evidence and the governing clauses of FDA 21 CFR Part 11, EMA Annex 11, TGA / PIC/S PE 009-17, and international auditing standards (PCAOB AS 1105 / 1215 and ISA 230 / 500 / 240 Revised 2025).

> 921121; ORDINAL12lfb1822a8113a1b691a7ef64bb7cd64ea2fa02bc8lbbedad71. Bitcoin
> `7917e0f12fcff508d387733fd543451846316a21b16ca3f0d872b9849f94a904`

Telles, Fernando. C12: AuditLog.AI Global Compliance Matrix. Zenodo; 2025.

https://doi.org/10.5281/zenodo.17462383

Ordinal 13 Ordinal 13 is the canonical, human-executed reproducibility dossier for AuditLog.AI v4.0, prepared under Sentinel Protocol QMS v4 and intended for regulator-grade verification (FDA / EMA / TGA / PCAOB / ISA). The 69-page PDF documents REGULATORY-RUN15, in which the inventor–operator (Dr Fernando Telles) independently re-computed all dual hashes and cross-checked them against the Bitcoin mainnet and the AMPLIFY_LEDGER master log, without any AI assistance in the execution sequence.

> 923691; ORDINAL13ldd498abe7b95b3faade9b244401a4f68756e82b8l05b3cb00. Bitcoin
> `e8ef6e6a321c51371ed3319e15d4dc5c0fd1f7b52ea96030e586e3e1feb17e61`

Telles, Fernando. Sentinel QMS v4: Human-Executed Reproducibility Audit (RUN015) for AuditLog.AI v4.0. Zenodo; 2025. https://doi.org/10.5281/zenodo.17625098

Ordinal 15 This record contains the hash-only audit outputs for Ordinal 15, a pre-registered disclosure integrity test designed to verify that the publicly disclosed Stage IV analysis dataset matches its anchored ground truth. The package includes deterministic audit summaries, Evidence Set Fingerprint (ESF) comparisons, and explicit enumeration of expected PASS and FAIL outcomes. No raw data, statistical analyses, or interpretive material are included. The contents allow any third party to independently verify disclosure integrity without access to execution environments or underlying evidence.

> 934659; ORDINAL15l7152ab0ffdd30982127306539db22725349d168fld8a2d8e2. Bitcoin `006e274af867de728c28da77175892cb76821e82f05378033e95e024712912a7`

Telles F. Ordinal 15: Proof-of-Unchanged Zero-Custody Audit Reproducibility Trial. Zenodo; 2026. https://doi.org/10.5281/zenodo.18452216