AuditLog.AI — Runtime Execution and System Validation Evidence Dossier

End-to-End Operational Proof During Regulatory Global Submission

Regulatory Submission Software: AuditLog.AI

Date: October 27, 2025

Inventor and Primary Contact: Fernando Telles BMedSc(Adv) MD(Dist)[1] [2]

Position: CEO & Founder **Email:** Dr.Telles@aihumansynergy.org **Phone:** Provided on Request **Address:** 21 Shields St, Flemington VIC 3031, AU **Web:** www.aihumansynergy.org

Engineers

Lead Software Engineer: Dr. Jacob Yang BEng, MEng, PhD[1] **Software Engineer:** Benjamin Hookey BEng (Mechatronics & Robotics), FSEng (Safety Instrumented Systems)[1]

Affiliations

[1] **Cardiovascular Diagnostic Audit & AI Pty Ltd (ACN 638 019 431)** – Registered Australian company conducting AuditLog.AI software development, audit and research services
[2] **Telles Investments Pty Ltd (ACN 638 017 384)** – Private IP holder

IP Rights

US Provisional #63/826,381 · AU Provisional #2025902482 · AU Trade Mark #2535745 & #2549093 IP Priority Date: 17 June 2025 (Global)

Submitted as part of the AuditLog.AI Global Regulatory Submission Package (FDA/EMA/TGA/PCAOB/ISA Alignment)

Regulatory Summary

Region	Submission Type	Regulatory Basis	Proposed Classification
FDA (USA)	Q-Submission (Q-Sub)	21 CFR Part 11 (Electronic Records / e-Signatures)	Standalone Electronic Records / Audit-Trail Infrastructure (non-device)
EMA (EU)	Scientific Advice (optional) + Annex 11 validation	EudraLex Vol 4 Annex 11 (Computerised Systems)	GMP Computerised System for Data Integrity (non-device)
TGA (Australia)	Excluded Software Determination	Excluded Goods Determination 2018 + PIC/S PE 009-17 Annex 11	LIMS-category audit infrastructure (non-medical device)

- **FDA (21 CFR Part 11)** — United States electronic records and signatures
- **EMA (Annex 11)** — European Union computerized systems for GMP
- **TGA (PIC/S PE 009-15)** — Australia therapeutic goods manufacturing principles

Auditing Standards Compliance Summary

Framework	Standard	Core Requirement	AuditLog.AI Compliance Mechanism
ISA / IAASB	**ISA 230**	Audit documentation enabling experienced auditor understanding	session log JSON compilation (who / what / when / meaning); append-only dual atomic ledgers for anchor receipts (TXID, block); frozen folder structure
ISA / IAASB	**ISA 500**	Sufficient appropriate audit evidence (quantity + quality: relevance + reliability)	cryptographic integrity + Bitcoin OP_RETURN (payload/TXID) external verification + OpenTimestamps
ISA / IAASB	**ISA 240**	Professional skepticism + fraud risk assessment + management override prevention	Append-only dual ledgers; non-repudiable e-signatures; fail-closed runtime checks; independent time attestations

Framework	Standard	Core Requirement	AuditLog.AI Compliance Mechanism
PCAOB	**AS 1215**	Audit documentation with 60-day assembly + 7-year retention	Frozen sources UTC timestamp-locked archives; anchors linking records to public blockchain
PCAOB	**AS 1105**	Audit evidence evaluation (sufficiency + appropriateness)	Public blockchain TXID verification + OTS; deterministic hash parity from frozen session records
PCAOB	**AS 1105.10A (effective 2025)**	External electronic information reliability evaluation	Bitcoin blockchain (public, decentralized) + independent blockchain explorer verification tools

Notes: Evidence flow is zero-custody (proofs only); all artifacts are version-locked, non-adaptive, and human-approved prior to anchoring.

---

1) Overview

1.1 Purpose of this Dossier

This dossier presents **visual and cryptographic execution evidence** for AuditLog.AI designed for regulator review (FDA/EMA/TGA) to demonstrate **end-to-end operational compliance, integrity, and human oversight**. The regulatory submission process was logged using the software itself during **Regulatory Global Submission** sessions.

1.2 Screenshots Capture Method & Controls

- **Live runtime capture** on macOS; timestamps displayed (AEST) and recorded in UTC in session logs.
- **No cropping** of regulatory screenshots; limited redactions (solid black boxes) applied solely to protect proprietary or security-sensitive information.
- Execution under **C5.2/C5.3 ethics enforcement** and **C9.5 Zero-Custody** (hashes/metadata only transmitted).

1.3 Execution Context (Sessions & Anchors)

- `session_log_AuditLogAI.REG.GlobalSubmission.v4009_20251013T034508.120256Z.json`

(TXID: **413a692f24f58970cf72fdabec58095d2b7599b5cb1325303cc6d19ec89ddb01**).

- `session_log_AuditLogAI.REG.GlobalSubmission.v4016_20251014T194515.612758Z.json` (TXID: **9a46014d657726798449cc6282de083e2084afc87aa5f23233903396d73b8d4f**).
- `session_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T200606.551974Z.json` (TXID: **d096197e02a8f1ac9ca7da43d353f109fe4ff72f6ae7f88b39c83877247e9ef7**).
- PRE/POST proof pair (UTC): **Δ = 28 seconds** (≤ 300s threshold).
- Reproducibility audit completed using tools independent of AuditLog.AI.

---

2) AuditLog.AI — Four-Tier Reproducibility Architecture

Tier	Scope	Process (Key Scripts)	Artifacts Produced
Per-Evidence	Individual files (eg. PDF, PNG, CSV, TXT)	`batch_dualhasher_multi_v3.3.py` • `batch_ots_stamper_v2.py`	`.hash` • `.2ha` • `.hash.ots`
Per-Batch	Folder-level completeness	Aggregate dual-hash + `manifest_<batch>.csv`	`.2ha` • `manifest_<batch>.csv`
Per-Log	Audit log integrity	`auditlogger.py` • `validis_batchauditlogger_template_v3.1.py`	`.hash` • `.2ha` • `.hash.ots`
Per-Session	System-level proof & anchoring	`session_logger.py` → OP_RETURN via `opreturnanchor_amplify_v5.1.py`	`.hash` • `.2ha` • `.hash.ots` • **TXID**

Standardized OP_RETURN payload (v4): `SENTINEL|<TYPE>|<ripemd160>|<sha8>`
(*TYPE examples: SESSION / AUDIT / BATCH / UPGRADE*).

---

3) Security & Privacy — Zero-Custody by Design

Principles - **Only hashes and minimal metadata** leave the user's environment (no raw evidence).
- **Frozen artifacts:** read-only copies (`chmod 444` + `chflags uchg`) for audit readiness; append-only receipts.

- **Human-gated anchoring:** no anchor without operator approval; **CME** rules runtime enforcement with fail-closed behavior.
- **Deterministic tamper-evidence:** any byte change flips digests; failures surface immediately.
- **Proven throughput:** large batch hashing completes within firewall constraints (≤ 300s) in test runs.

Verification Path (public, tool-agnostic)

```
User Node  ──(hashes only)──▶  AuditLog.AI Node  ──(OP_RETURN)──▶  Bitcoin Mainnet
```

---

4) e-Signature Verification Architecture (Multi-Layer)

Layer	Mechanism	Purpose / Control
1 — Personal Signature	Ed25519 keypair (unique to each reviewer)	Individual authentication and cryptographic attribution of intent
2 — Organizational Auth	HMAC-SHA256 institutional challenge (time-bound access)	Confirms enterprise authorization and enforces licensed use
3 — Fraud-Prevention Check	Sentinel LLM1 vector analysis (**non-biometric**)	Human-activity verification; blocks automation or scripted input until validated

Together these layers establish multi-factor human-primacy control for each e-signature event.
Layer 3 operates under the Zero-Custody Data Flow: vectors contain no identifiers and are analyzed by a **date-locked LLM1 model** against fixed evidence-based thresholds.
A deterministic **PASS/FAIL** result is returned to the signer; repeat attempts are permitted within the same authenticated Ed25519 + HMAC session.
Anchoring proceeds only after all three layers succeed, ensuring each recorded signature represents a deliberate human act within an authorized institution.

---

5) Independent Verification & Reproducibility (for Regulators)

This section enables an **independent, tool-agnostic re-execution** of all integrity checks without access to any vendor portal or proprietary system. All verification relies on **public cryptographic methods** and **open-source tools**.

A. Independent Verification Scope:

1. **Frozen artifacts:** AuditLog.AI Global Regulatory Submission PDFs; session logs `session_log_<META>_<UTC>.json`, audit logs `audit_log_<META>_<UTC>.json`, individual evidence files.

2. **Cryptographic Hash Consistency:** File/log/session-level `SHA-256` and `RIPEMD-160(SHA-256)` dual-hashes (`.hash` , `.2ha`), and folder-level `RIPEMD-160(SHA-256)` .
3. **OP_Return Validation** Bitcoin **OP_RETURN** payload parity: `SENTINEL|<TYPE>|<ripemd160>|<sha8>` and the on-chain **TXID/block height**.
4. **Runtime Execution Timestamps:** PRE/POST Δ ≤ 300s where applicable; chronological order of evidence → audit log → anchor.
5. **OpenTimestamps Bitcoin Anchoring** Using the provided **TXID**, confirm via a public blockchain explorer:
    - The transaction exists on Bitcoin mainnet and Block height matches the record
    - Timestamp confirms existence on listed date
6. **CME Compliance Enforcement for all audit and session logs (abbreviated): CME Rule 1:** Reject entries lacking **timestamped human validation**

- Verify if each log has 'human_verified' True flag and reviewer id (eg. "reviewer": "Dr. Fernando Telles").
- Verify if entry 'timestamp' is present, valid and in UTC format.
- Verify if each audit log has an opentimestamp (`.hash.ots`) file, and is listed in session log **CME Rule 4:** Block export if OP_RETURN or `.2ha` mismatch session log
- Verify OP_return proforma (v4) enforcement
- Verify parity between session log hashes and published payload (where ripemd160 = RIPEMD160(SHA256(bytes(session_log.json); sha8 are the first 8 digits of session log SHA256) **CME Rule 4a:** Reject batches violating **VALIS filename** (UTC suffix), **duplicate canonical base names**, or sidecar contamination
- Verify UTC suffix is present on all frozen filenames (filename___ e.g., `evidence___YYYYMMDDTHHMMSSZ.pdf`).
- Files containing the standardized UTC suffix cannot be re-processed, verify for duplicate frozen files.
- Verify that sidecar (e.g., `.hash` , `.2ha` , `.hash.ots`) are not recorded as source evidence or frozen.

B. Verification Instructions

1. **Obtain artifacts**
    - Download the frozen artifacts (eg. evidence file pdfs: `<filename>___<UTC>.pdf`)
    - Advanced method: Alternatively, for files containing text only with no embeded meta_data such as audit and session logs, text can be copied into a new document and saved (note: ensure no additional meta data is embeded by software, for guidance documents refer to Open-Source Validator Tools)

2. **Recompute hashes locally**
    - Verify against published hashes within this dossier, or hashes listed within logs and manifest csvs.
3. **Confirm anchor**
    - Derive **sha8** = first 8 hex chars of the session log's `SHA-256`.
    - Confirm the on-chain payload matches v4 proforma.
    - Verify the **TXID** and **block** via any public explorer (eg. https://mempool.space/, https://www.blockchain.com/explorer) or any Bitcoin node.

C. Open-Source Validator Tools (Optional)

The audit can be reproduced with open-source scripts maintained for third-party validators:

Repositories: - Sentinel Protocol v4: https://github.com/TELAISYN/AI-Human-Synergy/tree/main/Sentinel_Protocol_v4/validator_tools - Sentinel Protocol v3.1: https://github.com/TELAISYN/AI-Human-Synergy/tree/main/Sentinel_Protocol_v3.1/validator_tools (Note: Original tools developed and first publicly released in v3.1 remain valid and compatible with v4)

- **Included utilities:**
    - `AuditLog.AI_PublicDualHashVerifier.exe` (Open source user interface for independently verifying dualhashes with batch/multi-folder processing functionality; v4 release)
    - `folder_dualhasher_v2.py` — folder-level dual-hash (RIPEMD-160 of concatenated file SHA-256s; v4 release)
    - `dualhasher_public_sentinel_v1.1.py` — **batch** file `SHA-256` + `RIPEMD-160` computation (v3.1 release)
    - `extract_hashes.py` — extracts hashes into manifests for audits (v3.1 release)
    - `Validator Reproducibility Troubleshooting Checklist___20250809T030248.073005Z.md` (v3.1 release)

- **Ordinal 06 – Public Guide for Independent Reproducibility Verification**
  Telles, Fernando; Hookey, Benjamin; Woo, Andrew. *Ordinal 06 – Immutable Verification of Infrastructure Audit Log (Ordinal 05) under Sentinel Protocol v3.1.* CDA AI Pty Ltd, July 2025.
  DOI: 10.13140/RG.2.2.21019.78882
  Zenodo: 10.5281/zenodo.16777715

---

1. PRE Screenshot — User Evidence Directory

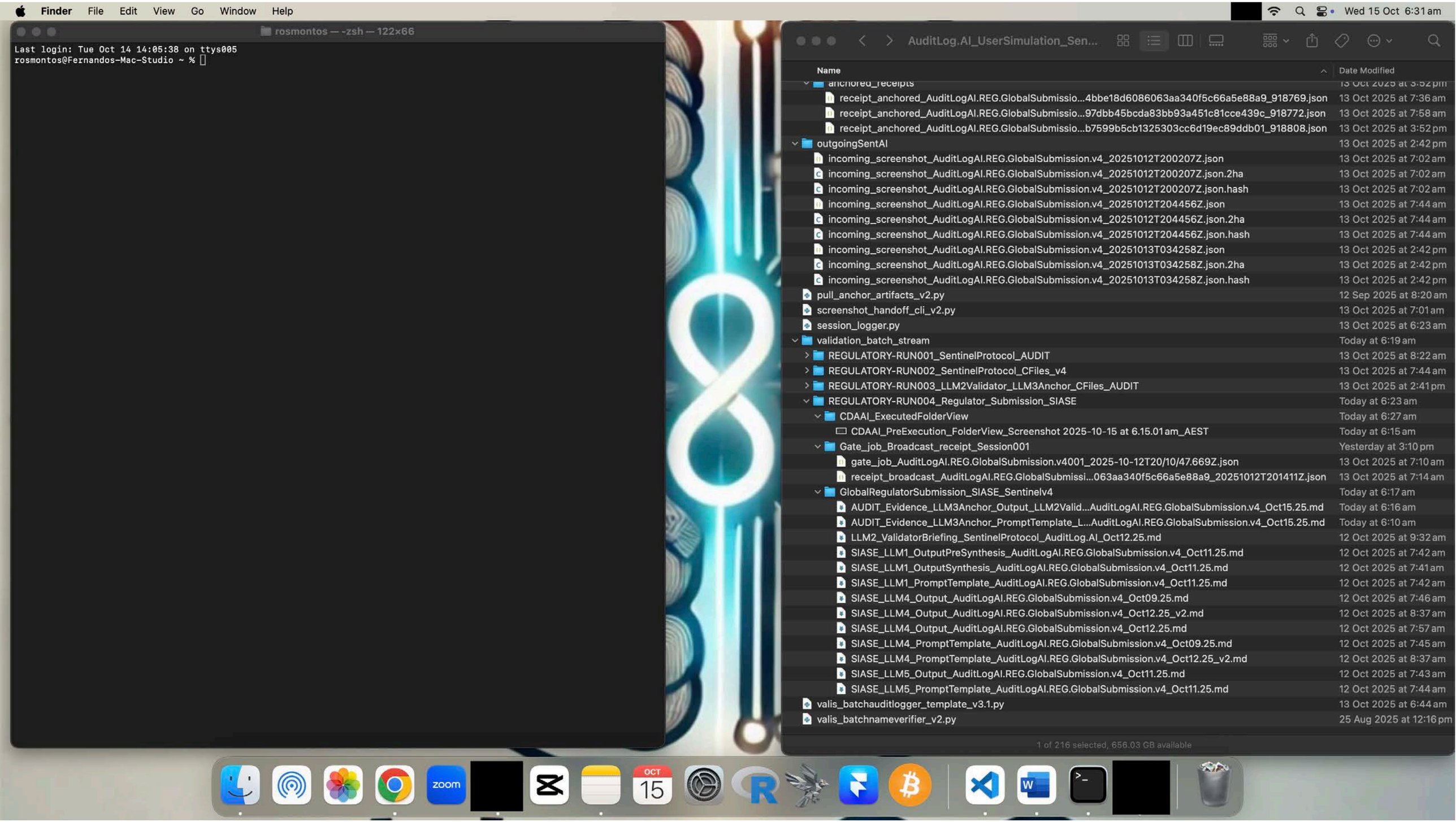
Description:

Displays the user-side directory containing all input evidence files prior to processing, establishing the baseline inventory for the audit session. This screenshot validates the system's readiness and ensures no pre-existing derivative files, demonstrating controlled environment preparation for data integrity verification. Only hashes will be transmitted, with raw content remaining local.

File Reference: `USER_PreExecution_FolderView_Screenshot 2025-10-15 at 6.31.35 amAEST___20251014T193659Z.png` **Timestamp (UTC):** `20251014T193659Z` **SHA-256:** `1112c4b8c1aff878711f39c2d822434436a9b2cfc302f72afb5597c04c58d35f` **RIPEMD-160:** `3e57c9450490544d98c75f08b9b9741e28d1cc2f` **OTS File:** `USER_PreExecution_FolderView_Screenshot 2025-10-15 at 6.31.35 amAEST___20251014T193659Z.png.hash.ots` **OTS SHA-256:** `8c7f114ce3c08c07ae5e9a51a1ae97cd5392b590d29c24f1a6c8bdfc0e7c8e5d` **OTS Bitcoin TXID:** `10e650706acadd74115a38042becea8ccbe58eac7146a9d1c9193ed027ca8cb7` **Block:** `919083` **Merkle Root (from OTS proof):** `416d0f316dec37bdbccdfb142775154e704a203ed41c94dcf10765dce6f7a4fa` **Date of Existence (UTC):** `2025-10-14T19:56:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919083.)*

---

1. PRE Screenshot — User Evidence Directory



File Reference: `USER_PreExecution_FolderView_Screenshot 2025-10-15 at 6.31.35 amAEST___20251014T193659Z.png`

2. PRE Screenshot — AuditLog.AI Folder View

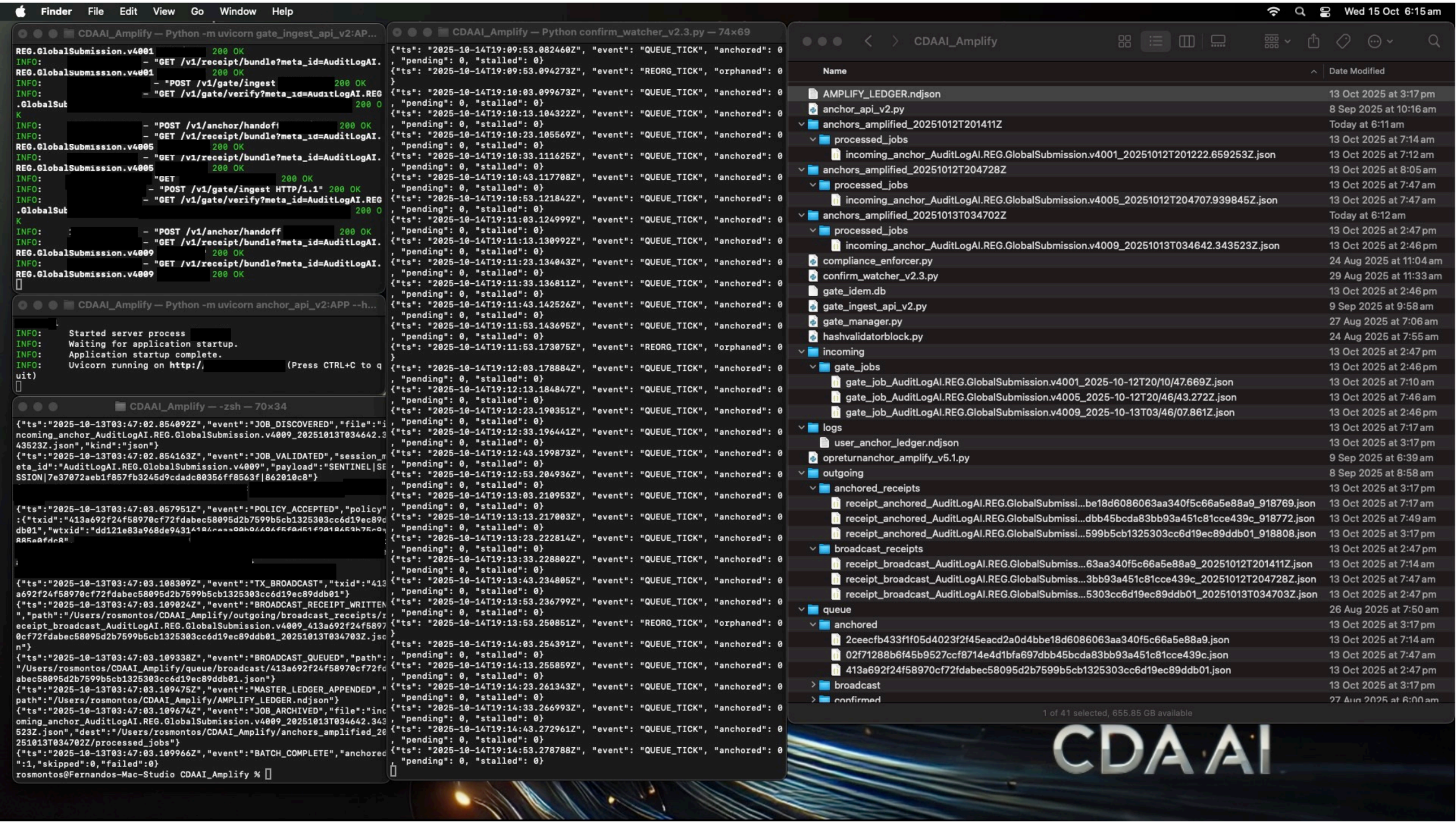
Description:

Shows the AuditLog.AI side directory structure before execution, verifying segregation of hash directories, and other components for reproducibility. This evidence confirms system isolation and readiness for receiving and supporting tamper-evident audit trails. It underscores zero-custody principles by illustrating that no raw data is present or transferred.

File Reference: `CDAAI_PreExecution_FolderView_Screenshot 2025-10-15 at 6.15.01 amAEST___20251014T193659Z.png` **Timestamp (UTC):** `20251014T193659Z` **SHA-256:** `db36d58e5ffdd417d171b22cd6e66f12844e3fa400f7e061c386f89c638176ec` **RIPEMD-160:** `d6ea1c213e00224817519cc68071291a97595812` **OTS File:** `CDAAI_PreExecution_FolderView_Screenshot 2025-10-15 at 6.15.01 amAEST___20251014T193659Z.png.hash.ots` **OTS SHA-256:** `9434e50d3629f36780116c647bac05ab1342c3e87ccb4dd3d430e6e63f5fc1e6` **OTS Bitcoin TXID:** `10e650706acadd74115a38042becea8ccbe58eac7146a9d1c9193ed027ca8cb7` **Block:** `919083` **Merkle Root (from OTS proof):** `416d0f316dec37bdbccdfb142775154e704a203ed41c94dcf10765d-ce6f7a4fa` **Date of Existence (UTC):** `2025-10-14T19:56:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919083.)*

---

2. PRE Screenshot — AuditLog.AI Folder View



3. Human Verifier — Distinct e-Signature Identification Component 1 of 2

4. Pre-Hasher Execution State Audit

3. Description:

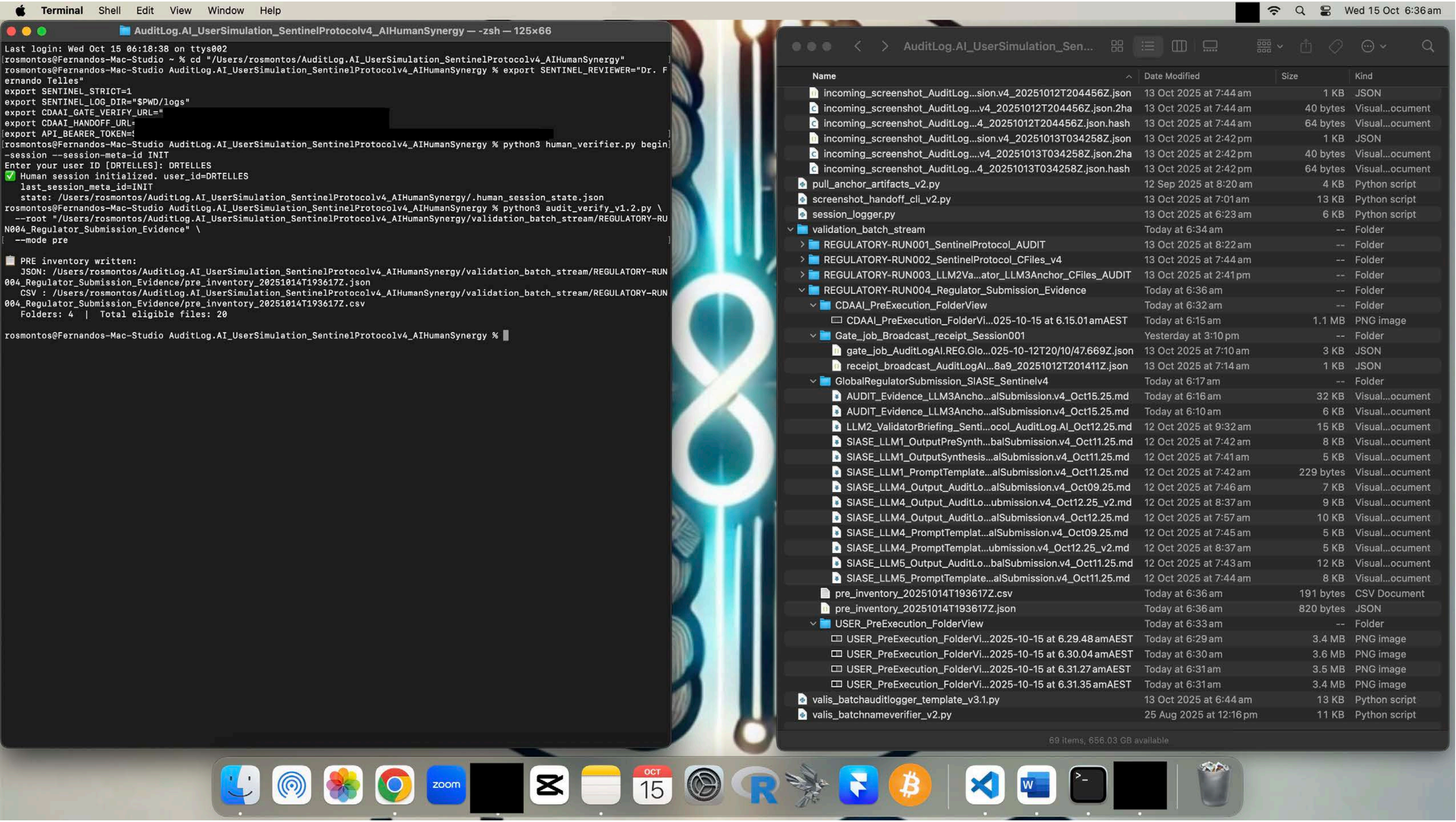
Displays the human-verifier interface initiating the multi-layer user verification sequence at session start. The verifier executes a user-bound Ed25519 digital signature in addition to an institutional HMAC-SHA256 login (bound to a specific meta ID, eg. `AuditLo-gAI.REG.GlobalSubmission.v4`), generating the initial human-verified audit event. This establishes human primacy and binds human intent to the session ledger; no processing can proceed without explicit human authorization. Signed electronic record content includes (displayed/recorded with each signing): 1. Printed name of the signer (reviewer identity) 2. Date and time of execution (UTC timestamp) 3. Meaning of the signature (e.g. reviewer)

4. Description:

Illustrates the pre-hasher execution state audit, enumerating evidence folders and generating baseline inventory records (pre_inventory.json/csv). This validates file presence and environment cleanliness, preventing omissions or unauthorized modifications. It supports reproducibility by establishing a verifiable starting point for hashing operations.

File Reference: `Screenshot 2025-10-15 at 6.36.22 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `e79789a382997019ed85d4d608fa0b205702cff61ef-ba630e1d1fe670685e2e0` **RIPEMD-160:** `31a5456d2184a32199ba052bf65444e7d77b216f` **OTS File:** `Screenshot 2025-10-15 at 6.36.22 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `78d67948ab87507edf446bfd61a2c6f068de210d50e377ede5d4d4966bf7cc1b` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

3. Human Verifier — Distinct e-Signature Identification Component 1 of 2
4. Pre-Hasher Execution State Audit



File Reference: `Screenshot 2025-10-15 at 6.36.22 am___20251014T200505Z.png`

5. User Prompt for Mandatory Pre-Execution System Screenshot

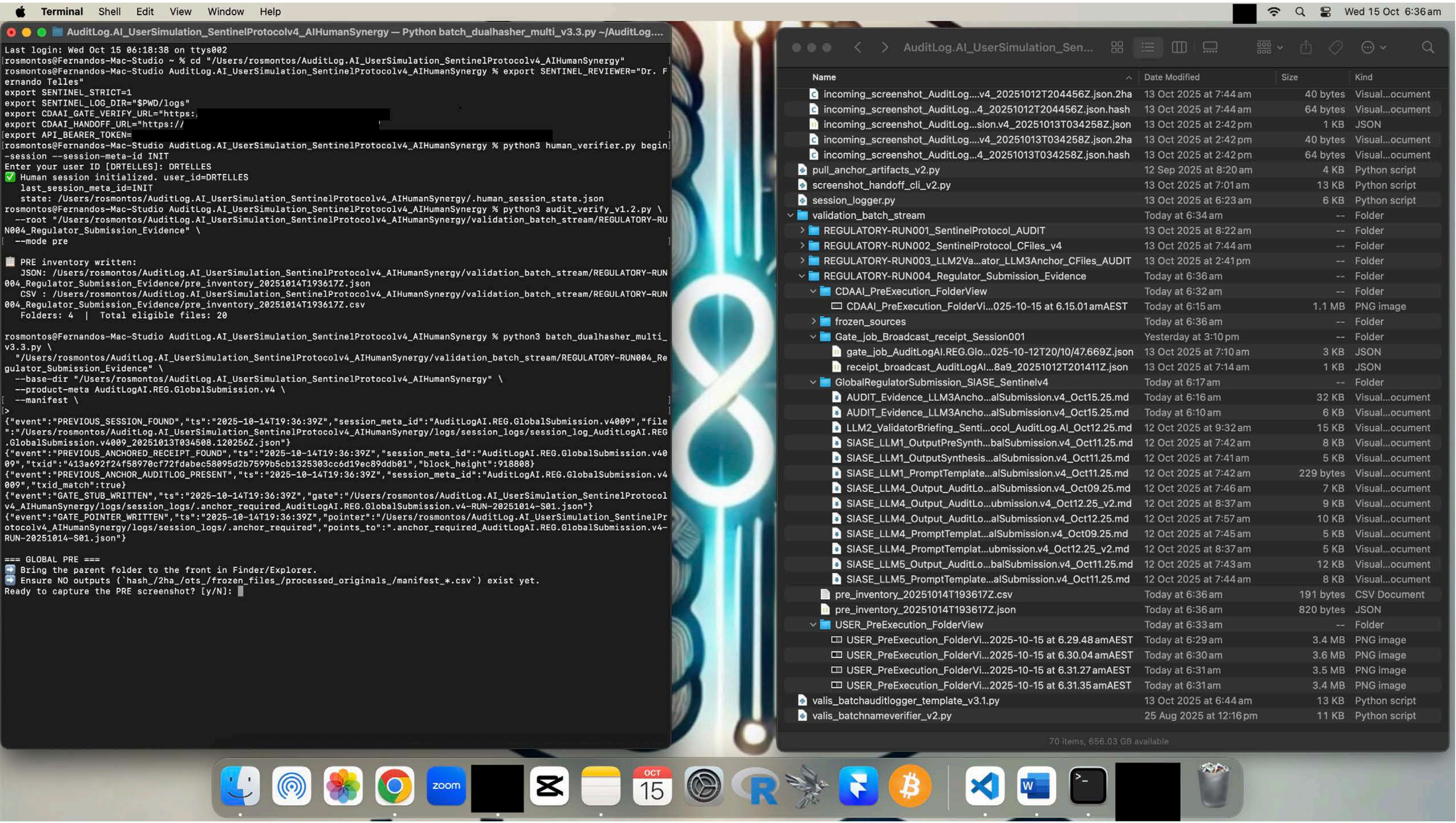
Description:

System-generated mandatory pre-execution capture recorded at human request before hashing begins. This documents the overall system state including open terminals and directories, ensures operational transparency and serves as a visual baseline for post-execution comparison. Screenshot `.png` remains within user's local directory and are not transmitted to AuditLog.AI.

File Reference: `Screenshot 2025-10-15 at 6.36.49 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `93ef6b41ca87da2646de0a424261450f96a471b25b-d04e88e331779a1f80bc35` **RIPEMD-160:** `455c336068a5fad5a651615d036f2ffc9ca65cb3` **OTS File:** `Screenshot 2025-10-15 at 6.36.49 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `6a882aef15535f5ad558c5fb1aac65403e67c34931be77602a5b8a3d154e28bd` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

5. User Prompt for Mandatory Pre-Execution System Screenshot



File Reference: `Screenshot 2025-10-15 at 6.36.49 am___20251014T200505Z.png`

6. Dual-Hasher Runtime Execution

6A. SHA-256 RIPEMD-160 Reproducibility & Amplification Audits

6. Description:

Depicts the initiation of batch dual-hashing process (`batch_dualhasher_multi_v3.3.py`), showing script execution generating `.hash` (SHA-256), `.2ha` (RIPEMD-160) and frozen copies (locally protected by chmod 444/chflags uchg) for each artifact. Each digest is written deterministically and time-stamped. This step validates cryptographic integrity of individual evidence items, ensuring accuracy and reproducibility.

File Reference: `Screenshot 2025-10-15 at 6.37.14 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `540-fa8357013f7e197f794b4aa914d78ca74ede044b0af3b26a4e1be5f6ed736` **RIPEMD-160:** `20bdc7494b-c76b48ffc675486ed82b87fc622533` **OTS File:** `Screenshot 2025-10-15 at 6.37.14 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `1b3a8cea868b86b56c41d1d-ab5570bd7e5a97c7c85db3a5d41088d157f23b98b` **OTS Bitcoin TXID:** `5c28773f653c441226df3079b-f1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

6A. Description:

Prior to implementation, hasher reproducibility is validated using tools independent of AuditLog.AI for all system updates. In order to attain a PASS result, reproducibility audits require 100% SHA-256 and RIPEMD-160 parity on re-execution of frozen files using public validator tools. All audits, whether PASS or FAIL, are executed through the software generating immutable audit trails. The latest two validations are registered as `AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_batch_dualhasher_multi_v3.2_v.aug23.25___20251012T195229Z.md` and `AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_batch_dualhasher_multi_v3.3_v.aug29.25___20251012T195229Z.md`. Excerpts from the later provided below.

```
 **Audit Type:** REPRODUCIBILITY
**Status:** PASS
**Audit ID:**
AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_batch_dualhasher_multi_v3.3_v.aug29.25.md

**Audit Date:** 29 August 2025
**Claim Class:** [Protocol | Credential | Infrastructure]
**Audit Target:** `batch_dualhasher_multi_v3.3.py`
**Audit Files:**
- `/Users/rosmontos/Testing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-27`
```

```
- `manifest_SENTINFRA_v4AMPLIFY_Test001_TEST-RUN-20250829-S01.csv`
- `manifest_SENTINFRA_v4AMPLIFY_Test002_TEST-RUN-20250829-S02.csv`
**Audit Files Source:** `/Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN12`
**Verification Method:** HUMAN_OVERSIGHT + `dualhasher_public_sentinel_v1.1.py` +
`extract_hashes.py` + LLM3 (Anchor)
**Verification Target:** Dual hash (`sha256`, `ripemd160`)
**Outcome:**
- ✅ SHA256 match on all files
- ✅ RIPEMD160 match on all files
- ✅ AUDIT_REPRODUCIBILITY_PASS
```

Additionally, to verify scalability for large multi-centre randomized clinical trials, the hasher subsystem underwent stress testing and successfully executed dual-hash processing for over 30,000 unique evidence files within the 300-second CME threshold. Performance validation was recorded in

`AUDIT_FUNC_PASS_30000+AMPLIFICATION_batch_dualhasher_multi_v3.2_batch_ots_stamper_v2.py_v.aug22.25___20251012T195229Z.md` (PRE: 2025-08-22 01:31:49 UTC | POST: 2025-08-22 01:35:27 UTC | Δ = 218 s). Excerpts from this audit are provided below.

```
 Last login: Fri Aug 22 08:55:20 on ttys001
rosmontos@fernandacstudio Testing_Optimization_Sentinel_Infrastructure_aug.19.25copy
% python3 batch_dualhasher_multi_v2.3.py \
  "/Users/rosmontos/Testing_Optimization_Sentinel_Infrastructure_aug.19.25copy/valid-
ation_batch_stream/TEST-RUN09" \
  --manifest \
  --threads 8 \
  --delta-sec-global 21600

🧭 Run ID: TEST-RUN-20250822 | Evidence folders: 3396 | Shot mode: global

=== GLOBAL PRE ===
➡️  Bring the parent folder to the front in Finder/Explorer.
➡️  Ensure NO outputs (`hash_/2ha_/ots_/frozen_files_/processed_originals_/mani-
fest_*.csv`) exist yet.
Ready to capture the PRE screenshot? [y/N]: y
✅ PRE screenshot: /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/frozen_sources/TEST-RUN-20250822/screenshots/PRE_20250822T013149Z.png
   PRE sha256=964ff7b455d26196… ripemd160(sha256)=eb73a38e8ad94af4…
🔍 TEST-RUN-20250822-S01: 3 files — /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09
📄 manifest_TEST-RUN-20250822-S01.csv
🔍 TEST-RUN-20250822-S02: 19 files — /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/DRTELLES

<redacted section due to large size, full contents within audit available to regulat-
ors on request>
<Redacted lines: 49-7025>

🔍 TEST-RUN-20250822-S3395: 8 files — /Users/rosmontos/Test-
```

```
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/sentinel-venv/lib/python3.11/site-packages/jedi/third_party/typeshed/
third_party/2and3/requests/packages/urllib3/util
📄 manifest_TEST-RUN-20250822-S3395.csv
🔍 TEST-RUN-20250822-S3396: 2 files — /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/sentinel-venv/lib/python3.11/site-packages/jedi/third_party/typeshed/
third_party/2and3/requests/packages/urllib3/packages/ssl_match_hostname
📄 manifest_TEST-RUN-20250822-S3396.csv

=== GLOBAL POST ===
➡️  In Finder/Explorer, filter or search for `manifest_` so **all** manifests are
visible on one screen.
➡️  Do not scroll.
Ready to capture the POST screenshot? [y/N]: y
✅ POST screenshot: /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/frozen_sources/TEST-RUN-20250822/screenshots/POST_20250822T013527Z.png
   POST sha256=b0b7fd74e03f6543… ripemd160(sha256)=121a81df0ae8761f…

✅ Session written: /Users/rosmontos/Test-
ing_Optimization_Sentinel_Infrastructure_aug.19.25copy/validation_batch_stream/TEST-
RUN09/frozen_sources/TEST-RUN-20250822/session_global_TEST-RUN-20250822.json
```

---

6. Dual-Hasher Runtime Execution

File Reference: `Screenshot 2025-10-15 at 6.37.14 am___20251014T200505Z.png`

7. OpenTimestamps (OTS) Runtime Execution

7A. OTS Explanation and Reproducibility Audits

7. Description:

Execution of batch OpenTimestamps (`batch_ots_stamper_v2.py`) generating `.hash.ots` artifacts for each evidence file. OpenTimestamps (OTS) anchors SHA-256 proofs via Bitcoin calendar servers and Merkle paths. It achieves timestamping goals with public on-chain settlement.

OTS Upgrade Proofs (Recommended Weekly Batch Execution): OpenTimestamps verification is a two-stage process: initial stamping records each file's SHA-256 hash in a pending state anchored to the OpenTimestamps calendar network, while final proof ("upgrade") embeds the calendar commitments into confirmed Bitcoin blocks once the transaction reaches maturity. Because on-chain consolidation and Merkle-path publication require network confirmations that may take hours or days, immediate upgrading is impractical. For operational efficiency, AuditLog.AI performs upgrades in weekly batches using `batch_ots_upgrader_v2.1.py`, ensuring all prior proofs are finalized and independently verifiable on Bitcoin mainnet.

File Reference: `Screenshot 2025-10-15 at 6.39.00 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `34dc1b37289d16b4c337914b14bd4bf2076995b877ae-afa32fe2dcb10999bc5b` **RIPEMD-160:** `4e183bb35891e60c70e3fce8970fe0747d15f318` **OTS File:** `Screenshot 2025-10-15 at 6.39.00 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `b67274a48d9275109e67167eb01613ed6d9ebdb4f64adc4fdaba08caf434a634` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

7A. Description:

Explanation of Matching OTS TXIDs: Multiple OpenTimestamps (`.hash.ots`) sharing the same TXID and Merkle Root is expected (Merkle aggregation). The OpenTimestamps protocol aggregates many SHA-256 file hashes into a Merkle root that is written to Bitcoin containing multiple unique SHA-256 within a given TXID. Each `.hash.ots` file contains a unique SHA-256 linking the specific file hash to that shared root; ensuring independent, verifiable proof that the file existed before the recorded block time. Repeated TXIDs therefore indicate batching within the same anchoring transaction, not duplication or error. List of evidence sets containing identical transaction IDs disclosed below for transparency.

Identical OTS TXID Set 1: - Dossier evidence: 1 & 2 - Session Log: `session_log_AuditLogAI.REG.GlobalSubmission.v4016_20251014T194515.612758Z.json` - Audit Logs: `audit_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T194300.697140Z.json` , `audit_log_AuditLogAI.REG.GlobalSubmission.v4023_20251014T194425.352577Z.json` Identical OTS TXID Set 2: - Dossier evidence: 3-14, 17-20, 22-41 - Session Log: `ses-`

`sion_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T200606.551974Z.json` - Audit Logs: `audit_log_AuditLogAI.REG.GlobalSubmission.v4025_20251014T200556.273909Z.json`, `audit_log_AuditLogAI.REG.GlobalSubmission.v4024_20251014T200552.935806Z.json` Identical OTS TXID Set3: - Dossier evidence: 21 - Session Log: `ses-sion_log_AuditLogAI.REG.GlobalSubmission.v4009_20251013T034508.120256Z.json` - Audit Log: `audit_log_AuditLogAI.REG.GlobalSubmission.v4007_20251013T034321.999677Z.json` Identical OTS TXID Set 4: - Dossier evidence: within 43. `ses-sion_log_AuditLogAI.REG.GlobalSubmission.v4038_20251018T214519.997259Z.json` & `AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_AuditLog.AI_v.oct18.25___20251018T214357Z.md`) (Note: Dossier evidence 15 & 16, system-generated screenshots, do not have `.hash.ots` files)

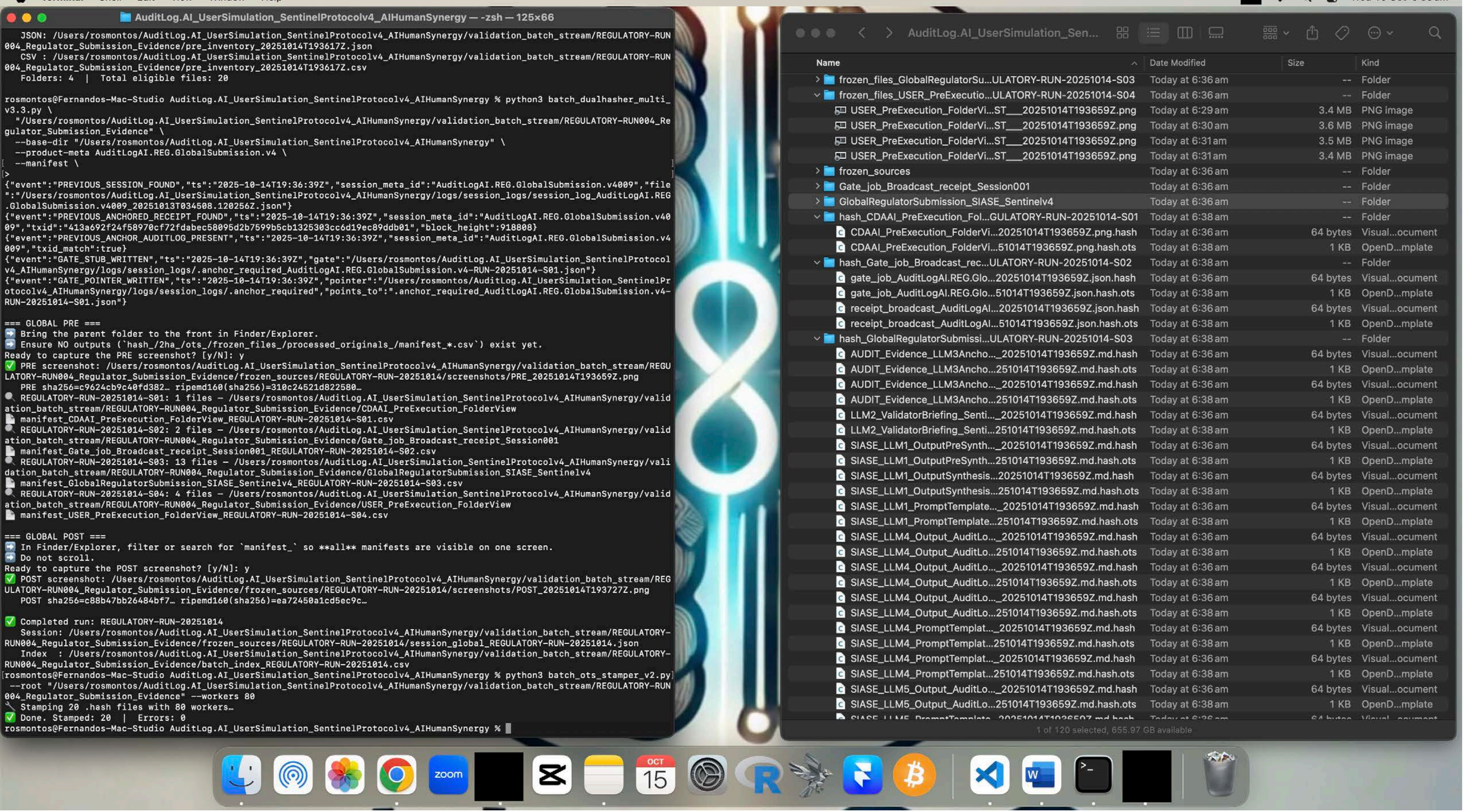
OTS proofs can be independently validated using the provided **TXID** via any public blockchain explorer (transaction exists; block height match; timestamp). Additionally, access to unique `.hash.ots` files is available to regulators on request, which allows independent verification of unique hash lineage for every evidence artifact under a common Bitcoin-anchored root (eg. `ots verify <filename>.hash.ots`). OTS reproducibility evidence, including timestamped full screenshots demonstrating Merkle roots and upgrade logs, is available through our public repository https://github.com/TELAISYN/AI-Human-Synergy/tree/main/Sentinel_Protocol_v3.1/Reproducibility_proof/METAVAL-SESS001_Ordinal_08/OTS_Reproducibility. OTS reproducibility has been audited and validated both internally (`.hash.ots` file access) and using public explorers by three human validators. These audits have been registered as: - `AUDIT_REPRODUCIBILITY_PASS_METAVAL-SES-S001_sha256_ripemd160_opentimestamps_OTS_v.august04.25.md___20250808T001356.822196Z` - `AUDIT_REPRODUCIBILITY_PASS_SENTINFRA-SES-S001_sha256_ripemd160_opentimestamps_OTS_v.july25.25.md___20250728T033533.108012Z` - `AUDIT-EVID-ENCE_VALIDATORS_REPRODUCIBILITY_SENTINFRA_PreDeployment_opentimestamps_OTS_v.july23.25.md___20250728T031210.342440Z` - `AUDIT-EVIDENCE_VALIDATORS_REPRODUCIBILITY_METAVAL-SES-S001_opentimestamps_OTS_v.july29.25.md___20250729T015736.514120Z` Excerpts from the later provided below.

```
 # AUDIT-EVIDENCE_VALIDATORS_REPRODUCIBILITY_METAVAL-SES-
S001_opentimestamps_OTS_v.july29.25.md

**Author:** Dr. Fernando Telles BMedSc(ADV) MD(DIST)
**Validator Role:** ARCHITECT VALIDATOR
**Validator meta_id:** DRTELLES-VAL
**Audit Date:** 18 July, 2025

**Independent Medical Doctor Validator Node:** Dr. Andrew Woo BSc (MD)
**Validator meta_id:** DRWOO-VAL
**Audit Date:** 29 July, 2025

**Independent Engineer Validator Node:** Benjamin Hookey BEng (Mechatronics & Robot-
ics), FSEng (Safety Instrumented Systems)
**Validator meta_id:** ENGBHOOKEY-VAL
**Audit Date:** 29 July, 2025
```

```
**Document Version:** v1.0
**Protocol:** Sentinel Protocol v3.1 ABSOLUTE MODE
**System:**  AI-Human Synergy™
**Audit Type:** REPRODUCIBILITY (Validator Audits – 3 of 3)
**Status:** ✅ PASS
**Audit ID:** AUDIT-EVIDENCE_VALIDATORS_REPRODUCIBILITY_METAVAL-SES-
S001_opentimestamps_OTS_v.july20.25.md

**Claim Class:** Infrastructure
**Linked Files:**
- `AUDIT-EVID-
ENCE_VALIDATORS_REPRODUCIBILITY_SENTINFRA_PreDeployment_opentimestamps_OTS_v.july20.2
5_ots_upgrade_log_20250718T040941Z.txt`
- `AUDIT-EVID-
ENCE_VALIDATORS_REPRODUCIBILITY_SENTINFRA_PreDeployment_opentimestamps_OTS_v.july20.2
5_ots_upgrade_log_20250708T223927Z.txt`
- `AUDIT-EVID-
ENCE_VALIDATORS_REPRODUCIBILITY_SENTINFRA_PreDeployment_opentimestamps_OTS_v.july20.2
5_ots_upgrade_log_20250708T223813Z.txt`
- `AUDIT-EVID-
ENCE_VALIDATORS_REPRODUCIBILITY_SENTINFRA_PreDeployment_opentimestamps_OTS_v.july20.2
5_ots_upgrade_log_20250708T223347Z.txt`
**Verification Method:** OpenTimestamps (`.ots`) proof validation
**Outcome:**
- ✅ All `.ots` files successfully verified
- ✅ All timestamps anchored to Bitcoin mainnet
- ✅ All files linked to matching `.2ha` and `.json` artifacts
- ✅ PASS
```

---

7. OpenTimestamps (OTS) Runtime Execution



File Reference: `Screenshot 2025-10-15 at 6.39.00 am___20251014T200505Z.png`

8. Post-Hasher Verification Audit

Description:

Post-hasher audit performed by `audit_verify_v1.2.py` against pre-execution inventory. Confirms parity against manifest csv rows, and confirms that `.hash`, `.2ha`, `.hash.ots`, frozen copies were generated for all evidence files issuing a PASS result. Outputs audit_report_.json/csv for full traceability.

File Reference: `Screenshot 2025-10-15 at 6.39.40 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `a05b75de6d8ce22f0c19444e46712bed48199ded0706479708e92b7ba222b362` **RIPEMD-160:** `96f6d2699b7ea24fc1d916f10bd99f05eb83a93a` **OTS File:** `Screenshot 2025-10-15 at 6.39.40 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `3eea5409288bf3a9e8daa944d005024c2935a513b6a46d039699ccd17f11fdf3` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

8. Post-Hasher Verification Audit

File Reference: `Screenshot 2025-10-15 at 6.39.40 am___20251014T200505Z.png`

9. Pre-/Post-Execution Screenshots Handoff (Pre-Human Authorization)

Description:

Confirms that user authorization is required **prior** to sending hashes and timestamps (no images) of pre-/post- screenshots to AuditLog.AI.

File Reference: `Screenshot 2025-10-15 at 6.41.14 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `2e9900ddead2f80f728635bc93a3d45fc1d99ea37af89df555338fad2112ecee` **RIPEMD-160:** `7d989cb7714f707f414751cee1d43d03d2ca20b1` **OTS File:** `Screenshot 2025-10-15 at 6.41.14 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `9037456e0e02d3ee5d38359cfd70493fa8efab6338468462d9499a1cdb8e3ac1` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

9. Pre-/Post-Execution Screenshots Handoff (Pre-Human Authorization)

10. Post-Human Authorization

Description:

Evidence of human-approved dataset now authorized and generated incoming_screenshot__.json, which is timestamped and dual-hashed for reproducibility provenance (`.hash` and `.2ha`).

File Reference: `Screenshot 2025-10-15 at 6.41.28 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `4c055b721b8c42d1454a8a8eda6f-d69c712822e009085ad236013e3fb7bac06c` **RIPEMD-160:** `8903abba24a346-ab8313025e6878450b9072a8c0` **OTS File:** `Screenshot 2025-10-15 at 6.41.28 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `08dc6669632ec-c63980ff5e85ba810a6c5a5471a9382213d79647aed28144f53` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

10. Post-Human Authorization

11. System-Generated Screenshots Zero-Custody Evidence

Description:

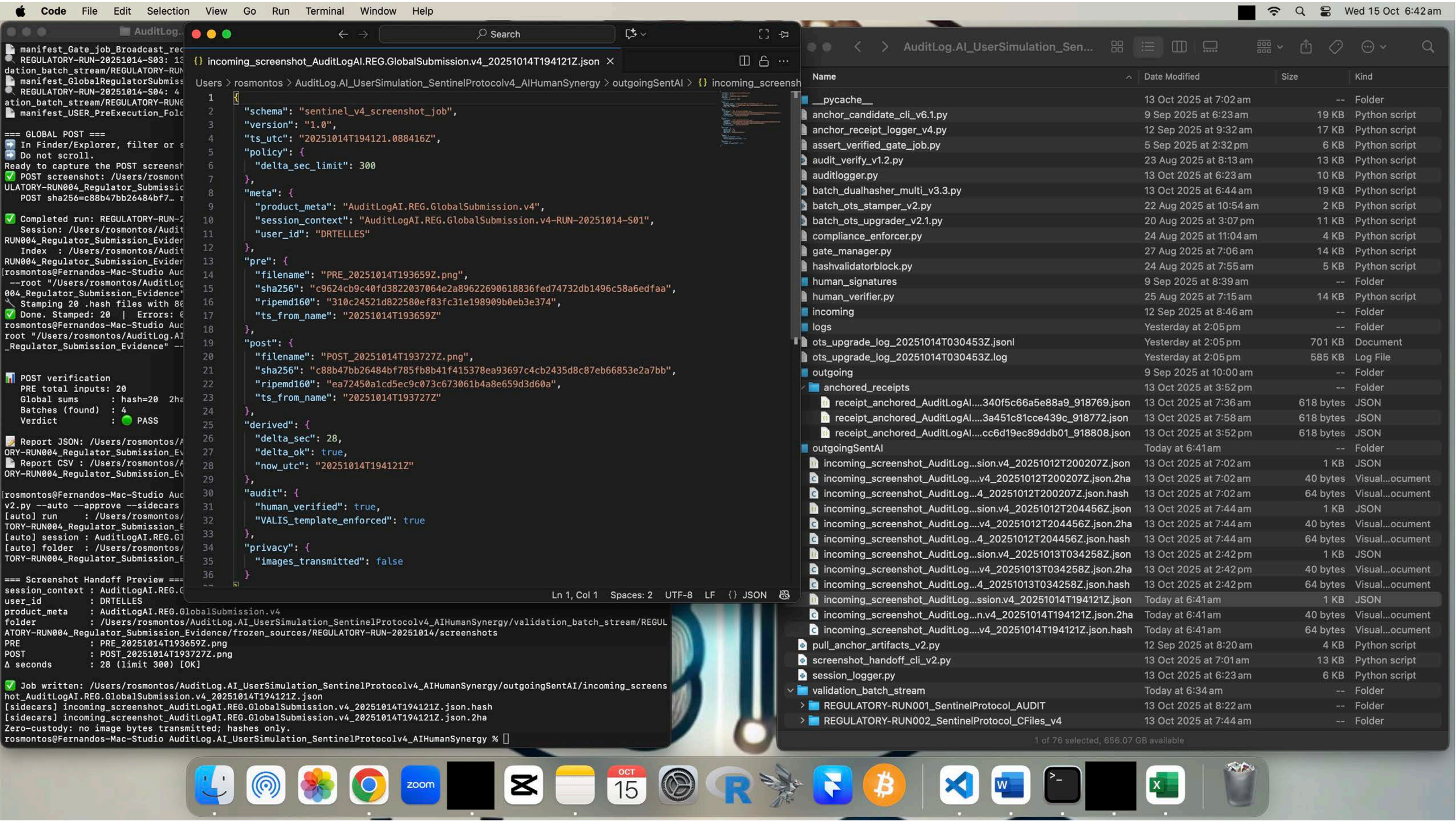
This artifact documents the zero-custody screenshot verification workflow for session AuditLogAI.REG.GlobalSubmission.v4 REGULATORY-RUN004_Regulator_Submission_Evidence. At present, the 300-second execution firewall is enforced deterministically at the script level, ensuring that pre- and post-execution screenshots are captured, hashed, and verified within a tightly controlled interval.

PRE and POST screens are saved locally within user's directory, frozen protected by chmod 444/chflags uchg: - PRE_20251014T193659Z.png (`/AuditLog.AI_UserSimulation_SentinelProtocolv4_AIHumanSynergy/validation_batch_stream/REGULATORY-RUN004_Regulator_Submission_Evidence/frozen_sources/REGULATORY-RUN-20251014/screenshots/PRE_20251014T193659Z.png`) - POST_20251014T193727Z.png (`/AuditLog.AI_UserSimulation_SentinelProtocolv4_AIHumanSynergy/validation_batch_stream/REGULATORY-RUN004_Regulator_Submission_Evidence/frozen_sources/REGULATORY-RUN-20251014/screenshots/POST_20251014T193727Z.png`) - Δ = 28 seconds

Cryptographic digests currently remain in user's directory (`/AuditLog.AI_UserSimulation_SentinelProtocolv4_AIHumanSynergy/outgoingSentAI`). In future modules, this is planned to be transmitted to AuditLog.AI for ML-based verification to augment this firewall by adding machine-learning pattern validation for additional fraud-prevention safeguards.

File Reference: `Screenshot 2025-10-15 at 6.42.04 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `d8a9b0b5adb2cff98860219fb13d619ce8da11950f759c59beccca82e817a93c7` **RIPEMD-160:** `a5a3cd1a7e8ed829d43b71796b566b84c6acc6fc` **OTS File:** `Screenshot 2025-10-15 at 6.42.04 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `cb5d741572ad06b64f48260c4f465b67dbb8ce533dc2839c9e89ec369615e5c2` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

11. System-Generated Screenshots Zero-Custody Evidence



12. VALIS Audit Verification Initial Prompt

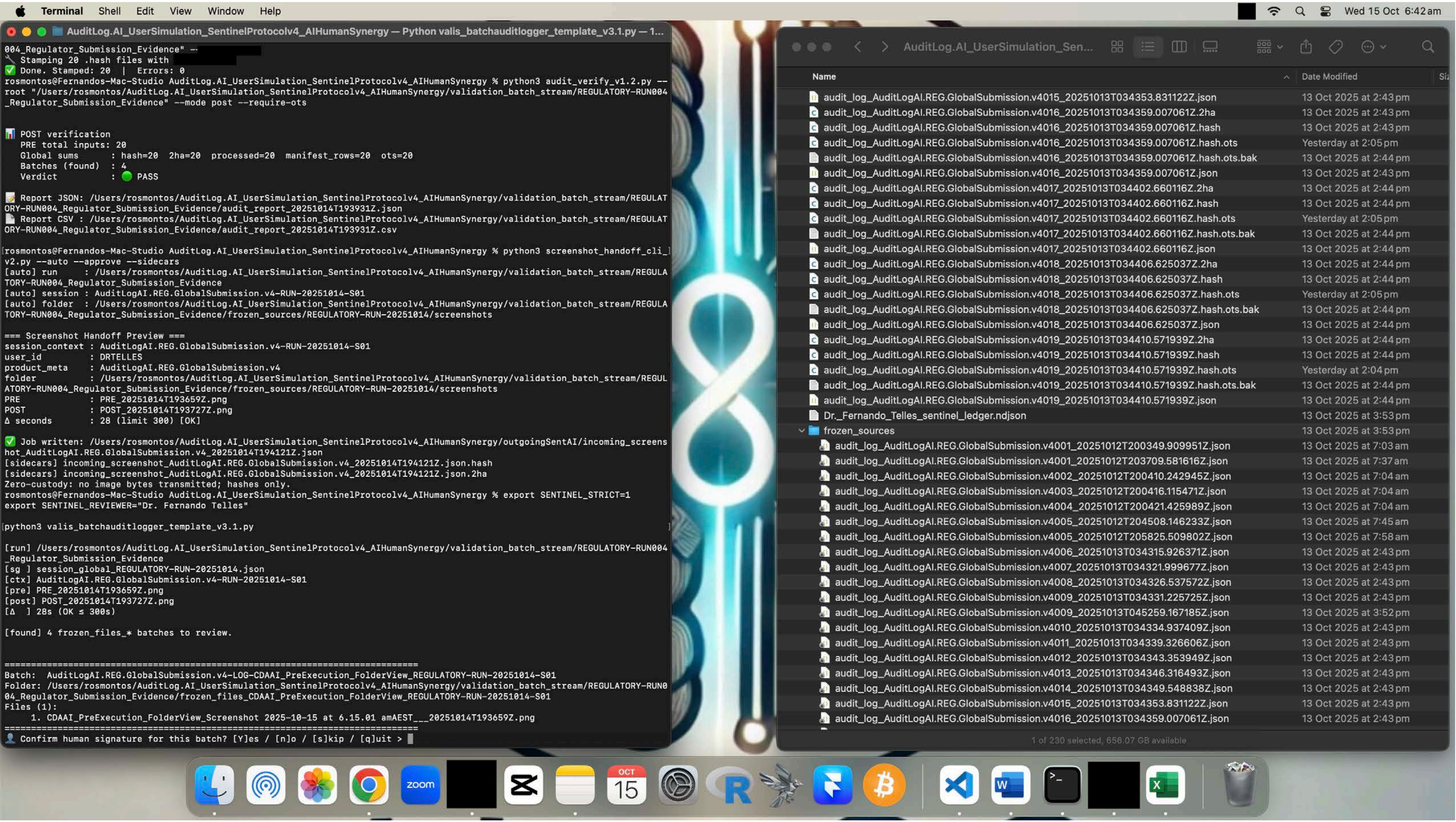
Description:

Evidence of VALIS automated batch verification (`valis_batchauditlogger_template_v3.1.py`) requesting human approval for each batch (Ed25519 personal signature) bound to an institution-specific meta ID. This prompt initiates the validation process between system automation and the human verifier. Records cannot be logged without explicit verifier approval. All evidence filenames comply with VALIS UTC-suffix, sidecars (.hash, .2ha, .ots) excluded from inputs. The institutional HMAC verification is executed via the Vercel interface and enforces license control (paid-user access), while the local Ed25519 signature authenticates the individual reviewer within that institution. (Note: Select audit files from v3.x are included within Dossier for historical provenance which show legacy schema `<filename>.ext___YYYYMMDDTHHMMSSZ`. Current v4 schema is `<filename>___YYYYMMDDTHHMMSSZ.ext`)

File Reference: `Screenshot 2025-10-15 at 6.42.52 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `03151f53ae21a8b88823f99dc81ba3192178fc66c899e827c0f-b978a17319c17` **RIPEMD-160:** `b2c0d9654b9f96dcbf20f3847fe2704f2592d4d1` **OTS File:** `Screen-shot 2025-10-15 at 6.42.52 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `f659db-d50f4993ce85464c458fdc24206f414a83443f4e10c4a7ac1f3f95a22d` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

12. VALIS Audit Verification Initial Prompt



13. VALIS Audit Verification Output (Log per Folder)

Description:

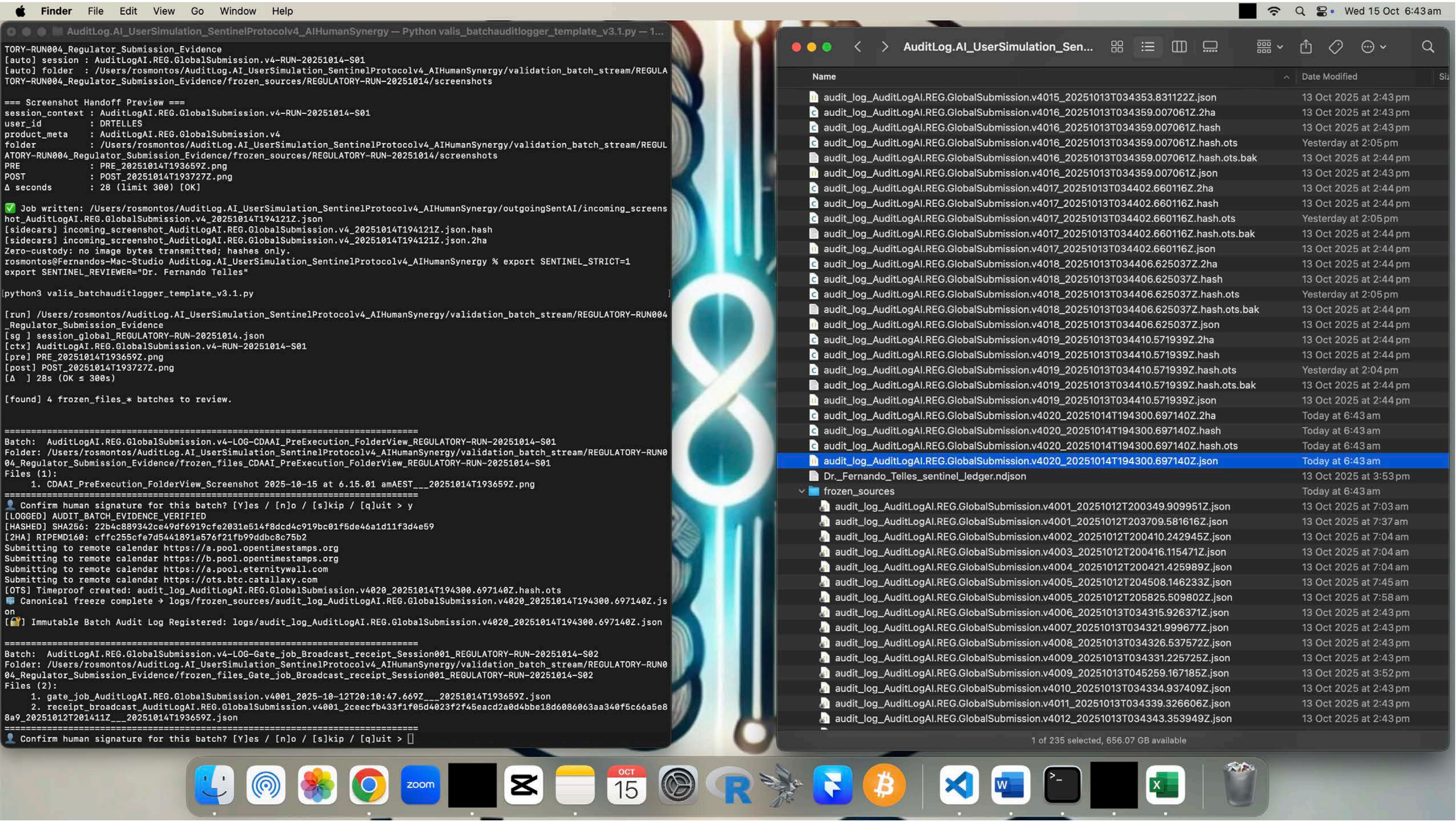
Displays Runtime VALIS template enforcement, a key component of AuditLog.AI CME firewall. VALIS CME features include: - Human-interactive sign-off loop with strict-mode enforcement. - Δ ≤ 300 s validation between PRE/POST screenshots. - Recursive VALIS verification of filenames structure and UTC suffix (`valis_batchnameverifier_v2.py`) - Deterministic dual-hash (SHA-256 → RIPEMD-160) folder verification for audit reproducibility. - CME pre-finalization check via `compliance_enforcer.py` ; fail-closed on violation. - Immutable audit log creation per folder through `auditlogger.py` , upon VALIS compliance pass, with corresponding frozen, `.hash` , `.2ha` and `.hash.ots` artifacts.

Systemic testing and validation of VALIS firewall has been logged and anchored within SENTINFRA pre-public deployment audit provenance: - `VALIS_Integrity_Firewall_Update_July4_25_v1.0.md___20250719T071359.588895Z` - `VALIS-Test_PASS_ErrorPrintout_ExecutionConstraint.md___20250719T071359.591755Z` - `VALIS-Test_PASS_Screenshot_DeltaTooLargeBlocked_AEST_Screenshot 2025-07-14 at 9.04.46 am.png___20250719T071359.586129Z` - `VALIS-Test_PASS_DuplicateBaseBlocked_AEST_Screenshot 2025-07-14 at 9.04.35 am.png___20250719T071359.585966Z` - `AUDIT_FUNC_PASS_Firewall_upgrade_CME_compliance_enforcer.py_v.july6.25.md___20250719T071359.581697Z` (**Provenance Session_log:** `session_log_SENTINFRA-SES-S001_20250719T074009.072317Z.json`)

File Reference: `Screenshot 2025-10-15 at 6.43.07 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `0d28787c0886d885737cc9b858904089b6c36511dbe-b098cd5cf9fe66915753e` **RIPEMD-160:** `adb1dee6f3bc2deae3c1c51ea59e69a158af6d47` **OTS File:** `Screenshot 2025-10-15 at 6.43.07 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `206a028ed1a548033ef4b5603b7bf1af3454bf4cfb66fc4647e226b8acf02e46` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

13. VALIS Audit Verification Output (Log per Folder)



14. AuditLog Generated Post Verification

Description:

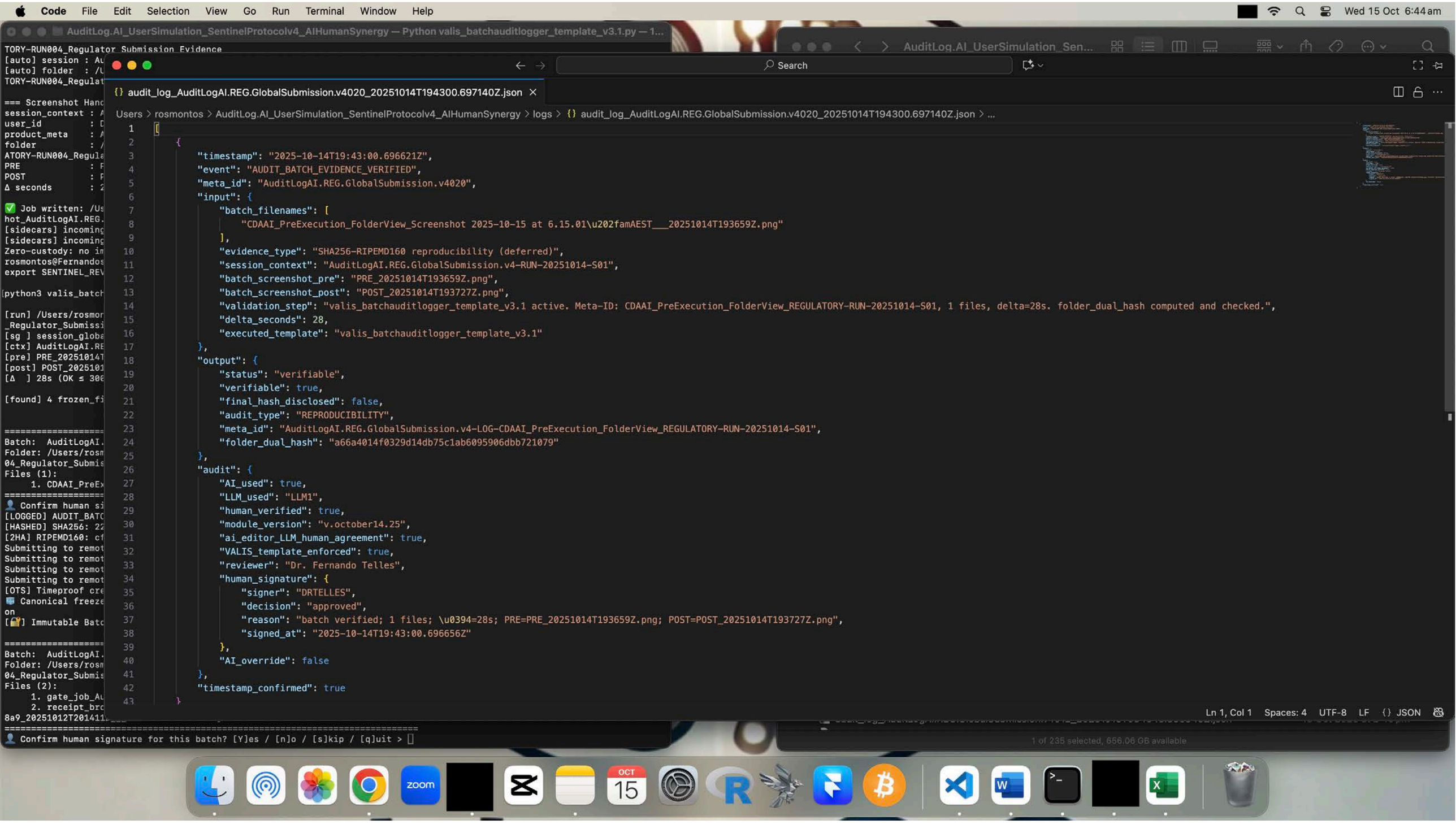
Captures the first generated audit log (`audit_log_<meta_id>_<UTC>.json`) following VALIS enforcement and human verification. Each audit log corresponds to one executed evidence folder and serves as a cryptographically sealed record of that operation. The audit log enumerates all evidence filenames within the executed folder, each following the standardized UTC-suffixed naming convention that ensures deterministic traceability.

The audit log includes: - The RIPEMD-160 digest of the executed folder (folder_dual_hash) confirming byte-level reproducibility. - Full CME firewall metadata, including human_verified: true, AI_used: true, timestamp_confirmed: true, and delta < 300s. - Cross-linked PRE → POST screenshots, providing evidence of controlled process continuity. - The human verifier identity (reviewer, human_signature) and corresponding meta_id binding the record to a specific validated session. - Explicit disclosure of AI participation, AI-Human Agreement and VALIS Template Enforcement.

File Reference: `Screenshot 2025-10-15 at 6.44.02 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `96b7647164a0a8e59d099146e3148b9530869290b-b54cee2e6f5d135610667a4` **RIPEMD-160:** `560271013b7e4e7ef8fbe18f598e38eced67687d` **OTS File:** `Screenshot 2025-10-15 at 6.44.02 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `27da80bfbdc680388bf77fc78e1e0e9e4bd6cd770741d4333a215ae360204b34` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

14. AuditLog Generated Post Verification

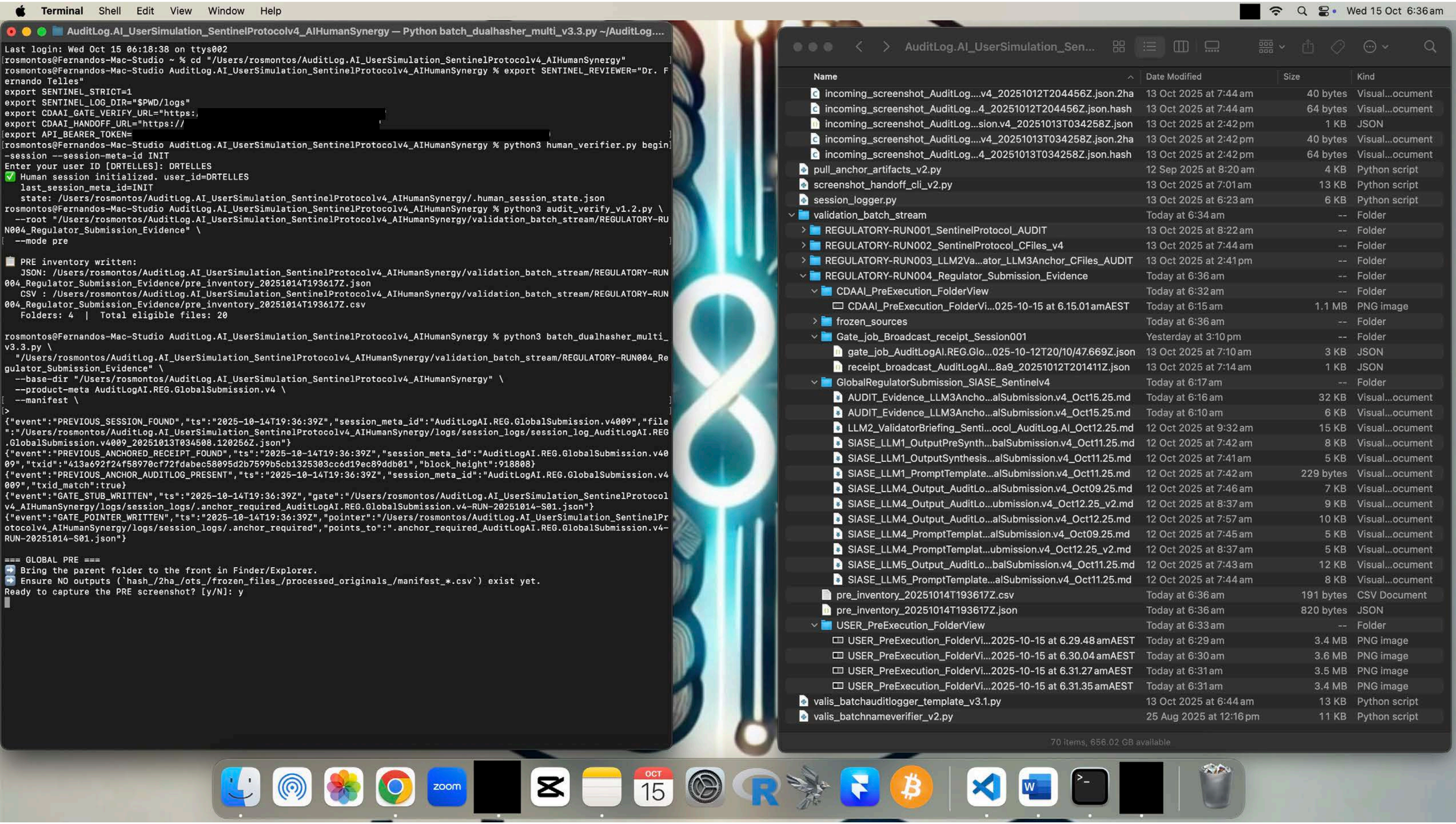


15. Pre-execution Mandatory Screenshot (User held)

This screenshot records the PRE state immediately before hashing begins. It captures the final pre-execution checks showing pre-inventory audit, original files for processing, and timestamp. As previously noted, this step is a mandatory system requirement. Screenshot is taken, timestamped, dual-hashed and frozen automatically following human approval.

File Reference: `PRE_20251014T193659Z` **Timestamp (UTC):** `20251014T193659Z` **SHA-256:** `c9624cb9c40fd3822037064e2a89622690618836fed74732db1496c58a6edfaa` **RIPEMD-160:** `310c24521d822580ef83fc31e198909b0eb3e374`

15. Pre-execution Mandatory Screenshot (User held)



File Reference: `PRE_20251014T193659Z`

16. Post-execution Mandatory Screenshot (User held)

Description:

POST-execution screenshot records evidence of hash/2ha/frozen artifacts generation. Paired screenshots captured by system under human authorization before and after hashing ensures process duration within thresholds (Δ ≤ 300 s). Stored only on the user side, their timestamped frozen file names and hashes are included in VALIS template for reproducibility validation.

The accompanying `session_global_REGULATORY-RUN-20251014.json` file is generated after the post-execution screenshot is captured and hashed. It serves as a lightweight session manifest linking the paired PRE and POST screenshots, their SHA-256/RIPEMD-160 digests, and UTC capture times.

File Reference: `POST_20251014T193727Z` **Timestamp (UTC):** `20251014T193727Z` **SHA-256:** `c88b47bb26484bf785fb8b41f415378ea93697c4cb2435d8c87eb66853e2a7bb` **RIPEMD-160:** `ea72450a1cd5ec9c073c673061b4a8e659d3d60a`

- `session_global_REGULATORY-RUN-20251014.json`:


```
{
  "run_id": "REGULATORY-RUN-20251014",
  "session_context": "AuditLogAI.REG.GlobalSubmission.v4-RUN-20251014-S01",
  "pre": {"path": "/Users/rosmontos/Audit-
Log.AI_UserSimulation_SentinelProtocolv4_AIHumanSynergy/validation_batch_stream/REGU-
LATORY-RUN004_Regulator_Submission_Evidence/frozen_sources/REGULATORY-RUN-20251014/
screenshots/PRE_20251014T193659Z.png", "sha256": "c9624cb9c40f-
d3822037064e2a89622690618836fed74732db1496c58a6edfaa", "ripemd160_sha256":
"310c24521d822580ef83fc31e198909b0eb3e374"},
  "post": {"path": "/Users/rosmontos/Audit-
Log.AI_UserSimulation_SentinelProtocolv4_AIHumanSynergy/validation_batch_stream/REGU-
LATORY-RUN004_Regulator_Submission_Evidence/frozen_sources/REGULATORY-RUN-20251014/
screenshots/POST_20251014T193727Z.png", "sha256": "c88b47bb26484bf785f-
b8b41f415378ea93697c4cb2435d8c87eb66853e2a7bb", "ripemd160_sha256":
"ea72450a1cd5ec9c073c673061b4a8e659d3d60a"},
  "generated_at_utc": "20251014T193727Z"
}
```

16. Post-execution Mandatory Screenshot (User held)

File Reference: `POST_20251014T193727Z`

17. VALIS Audit Log per Folder Authorized

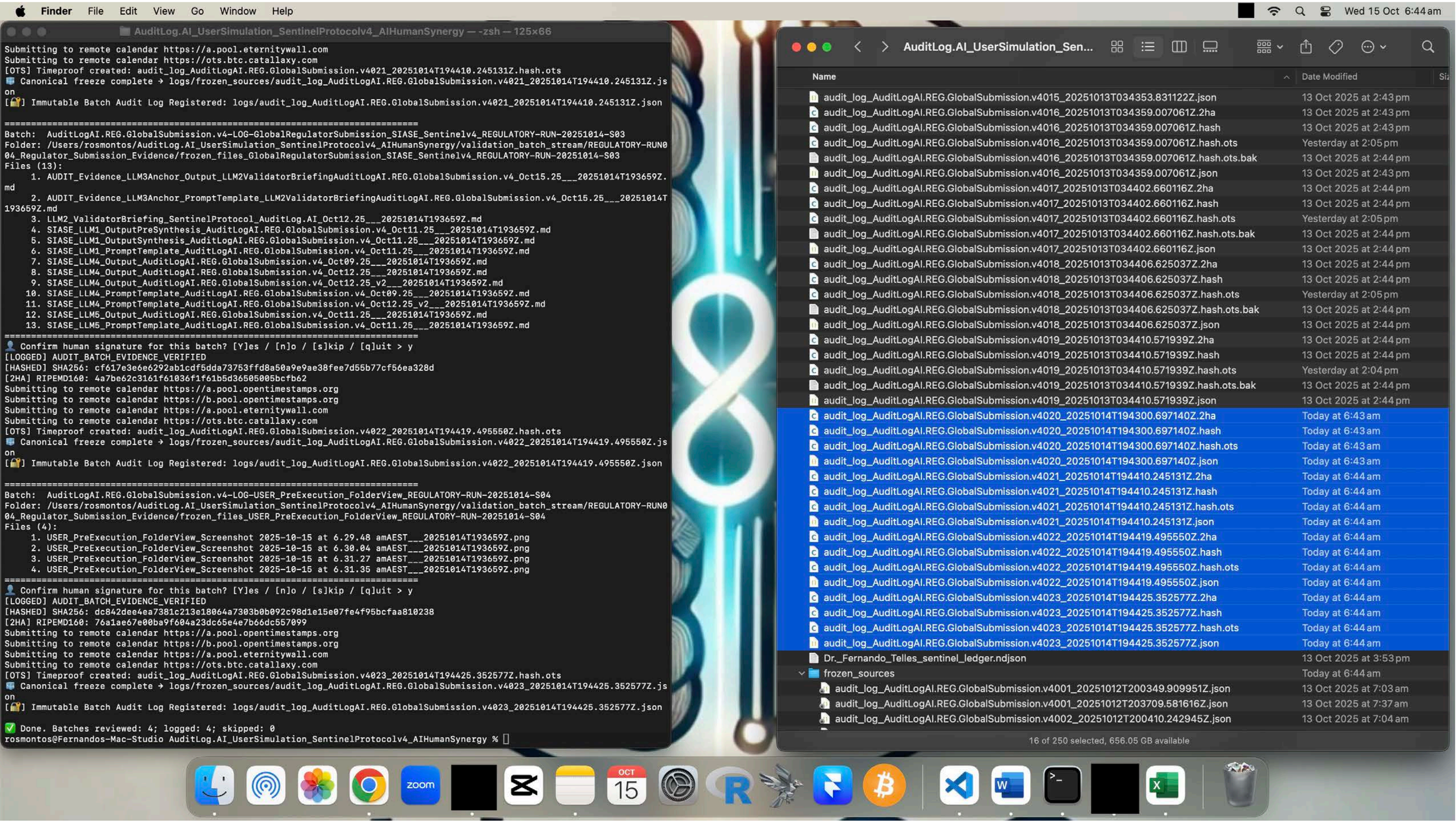
Description:

Displays finalized audit_log_* entries. Each audit log is signed by the human verifier and corresponds to an executed folder.

File Reference: `Screenshot 2025-10-15 at 6.44.37 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `3cf22ac819b74862860924d77b04955c-cc92428f403793184f0ff8454bf4ec0e` **RIPEMD-160:** `ac866c8629b156c8195553619421304ef6cd3e34` **OTS File:** `Screenshot 2025-10-15 at 6.44.37 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `99cace613f64e04569367de856b27c1388b29497f9465f3e3921a910c61684ff` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

17. VALIS Audit Log per Folder Authorized



18. Frozen VALIS Audit Logs

Description:

This screenshot documents the freezing safeguards (chmod 444 read-only permission and chflags uchg immutable flag) applied to copies of evidence files, audit logs and session logs. This results in a locally immutable, read-only artifact that prevents any modification, deletion, or metadata rewrite.

This mechanism was developed following a reproducibility failure identified during early testing (v3.0) reproducibility audit (`AUDIT_REPRODUCIBILITY_FAIL_sha256_ripemd160_v.july4.25___20251012T195229Z.md`). The audit revealed that even without visible content changes, secondary RIPEMD-160 validation could fail when certain utilities (e.g., TextEdit) silently embedded non-visible metadata during file access. Frozen protections now ensures every validated file remains byte-identical for future audits (`AUDIT_REPRODUCIBILITY_PASS_SENTINFRA_PreDeployment_chflags_uchg_lock_integrity_v.july11.25___20251`). These frozen artifacts can always be re-hashed to reproduce the original SHA-256 + RIPEMD-160 pair, ensuring reproducibility and long-term auditability as evidenced by `AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_v.july6.25.md___20250719T071359.581941Z` and `AUDIT_REPRODUCIBILITY_PASS_SENTINFRA-SES-S001_sha256_ripemd160_opentimestamps_OTS_v.july25.25.md___20250728T033533.108012Z`.

Note: Select testing and validation audits performed and originally registered in v3.1 pre-public deployment provenance ledger, were again logged in regulatory session 01 for full transparency `session_log_AuditLogAI.REG.GlobalSubmission.v4001_20251012T200455.617478Z.json`. All frozen audit files are reproducible and available to regulators on request.

File Reference: `Screenshot 2025-10-15 at 6.44.52 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `b3aaab7a6d948f27abed334863459d-cfc508d57959908d0e2f168ef733f1df9a` **RIPEMD-160:** `10feed27a06ec535ae6dcf63673bbd-c64f214607` **OTS File:** `Screenshot 2025-10-15 at 6.44.52 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `b78226e468250c24aa7ea58cabfb-c5dd318ab0a3456349a30eaf39c9ed0a9242` **OTS Bitcoin TXID:** `5c28773f653c441226df3079b-f1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

18. Frozen VALIS Audit Logs

File Reference: `Screenshot 2025-10-15 at 6.44.52 am___20251014T200505Z.png`

19. Session Logger Execution

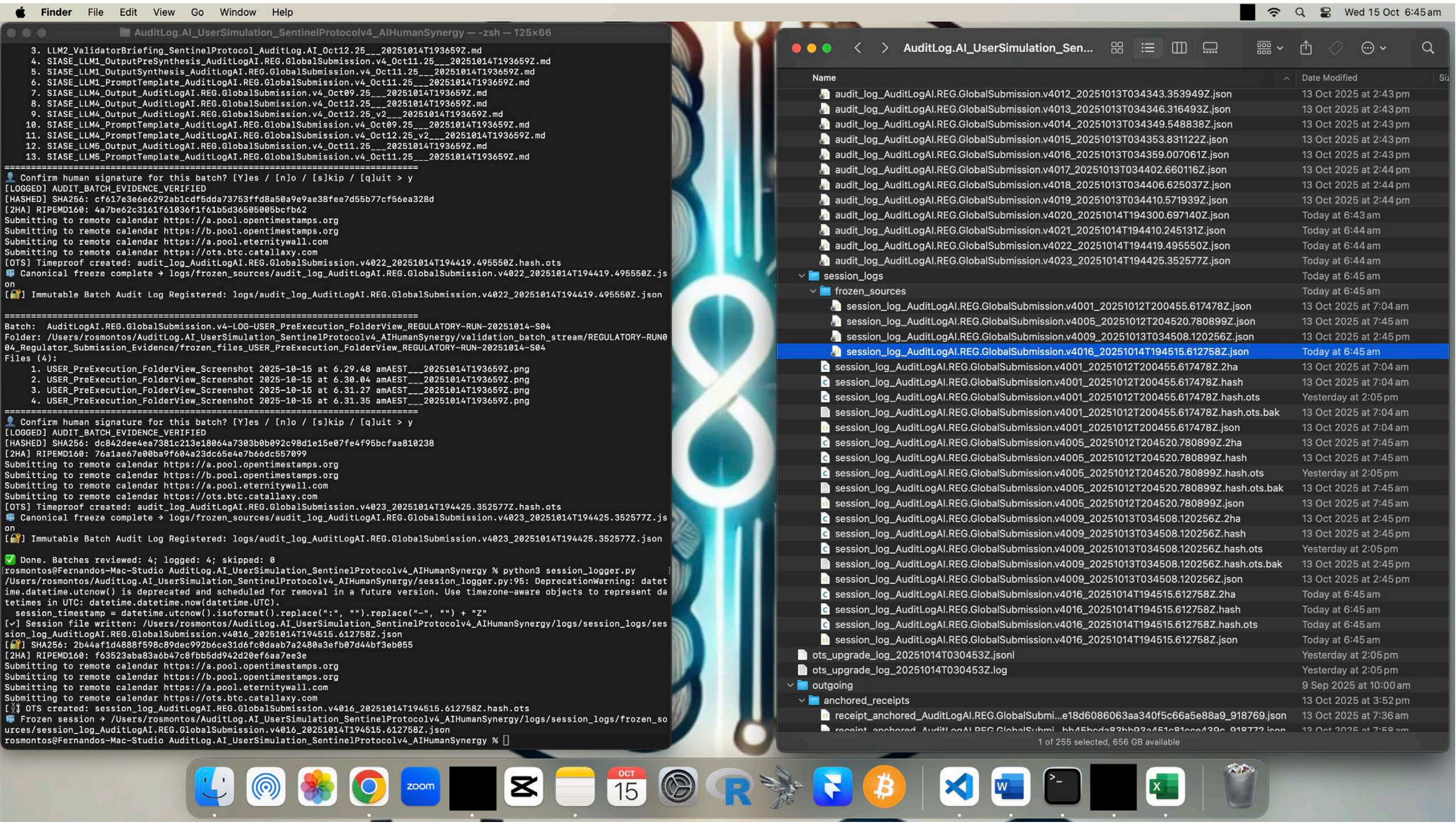
Description:

Execution of `session_logger.py`, compiling all validated audit logs into a single, consolidated session evidence record. Outputs session_log__.json + sidecars (`.hash` , `.2ha` and frozen copies).

File Reference: `Screenshot 2025-10-15 at 6.45.24 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `dd98ba9058e14586ea374e510f8fd-b19554f589327e515041f4a52a3e26a1636` **RIPEMD-160:** `ecea70b2c1fb96debc6db252f-d3e032e9a92a9a2` **OTS File:** `Screenshot 2025-10-15 at 6.45.24 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `0409b-f7356fe6825630a7c649ad3ecf8359882f96238c6c8cfbc803a86efba16` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

19. Session Logger Execution



20. Session Log View and Prior Session Anchor Audit Log Reference

Description:

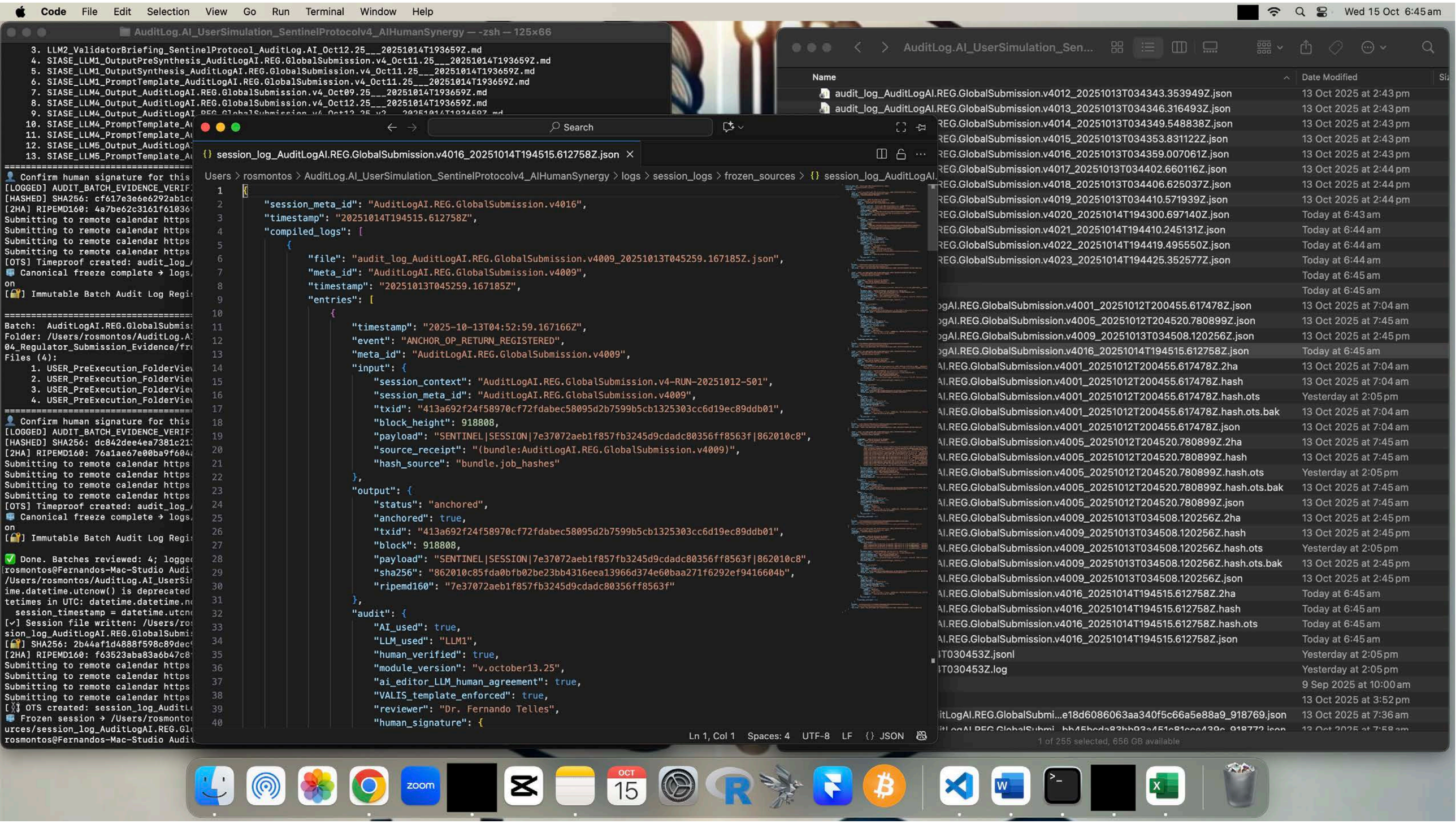
This screenshot shows the compiled session log (session_log_.json), displaying the built-in chain-of-custody enforcement between consecutive sessions. Each new session is cryptographically linked to the prior one through its recorded TXID, OP_RETURN payload, and block height. This continuity is CME enforced: a new session cannot start unless the previous session's anchor event is confirmed and logged. This mechanism provides an immutable, verifiable bridge across the reproducibility chain (e.g., Session 016 → 020), ensuring end-to-end traceability, cross-session reproducibility, and continuous integrity across the anchored audit lineage.

Importantly, testing and validation v3.0 reproducibility audits revealed that cryptographic hashes and transaction IDs cannot be consistently recalled by AI (LLM1) with full fidelity (`AUDIT_REPRODUCIBILITY_FAIL_hash_recalc_LLM1_v.july1.25.md___20250719T071359.583603Z`). AuditLog.AI is therefore built as a deterministic software (not AI adjudication), marked by developents such as `hashvalidatorblock.py` that enforce payload parity to session log hashes (`AUDIT_REPRODUCIBILITY_PASS_hashvalidatorblock.py_patch_v.july1.25.md___20250719T071359.578149Z`).

File Reference: `Screenshot 2025-10-15 at 6.45.44 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `615a9f56784b6206bf6f72a4fa0905ffb620aa90ccb58d-d03a5537e4135479c4` **RIPEMD-160:** `db3cc5132cac33bad4d7c2c380b2ebbb0e41ef83` **OTS File:** `Screenshot 2025-10-15 at 6.45.44 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `dad2953105d1c0e51237041323c111e71026c190220733feaeb195dd1f98d501` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

20. Session Log View and Prior Session Anchor Audit Log Reference



21. HMAC (Layer 2) + Reviewer Login

Description

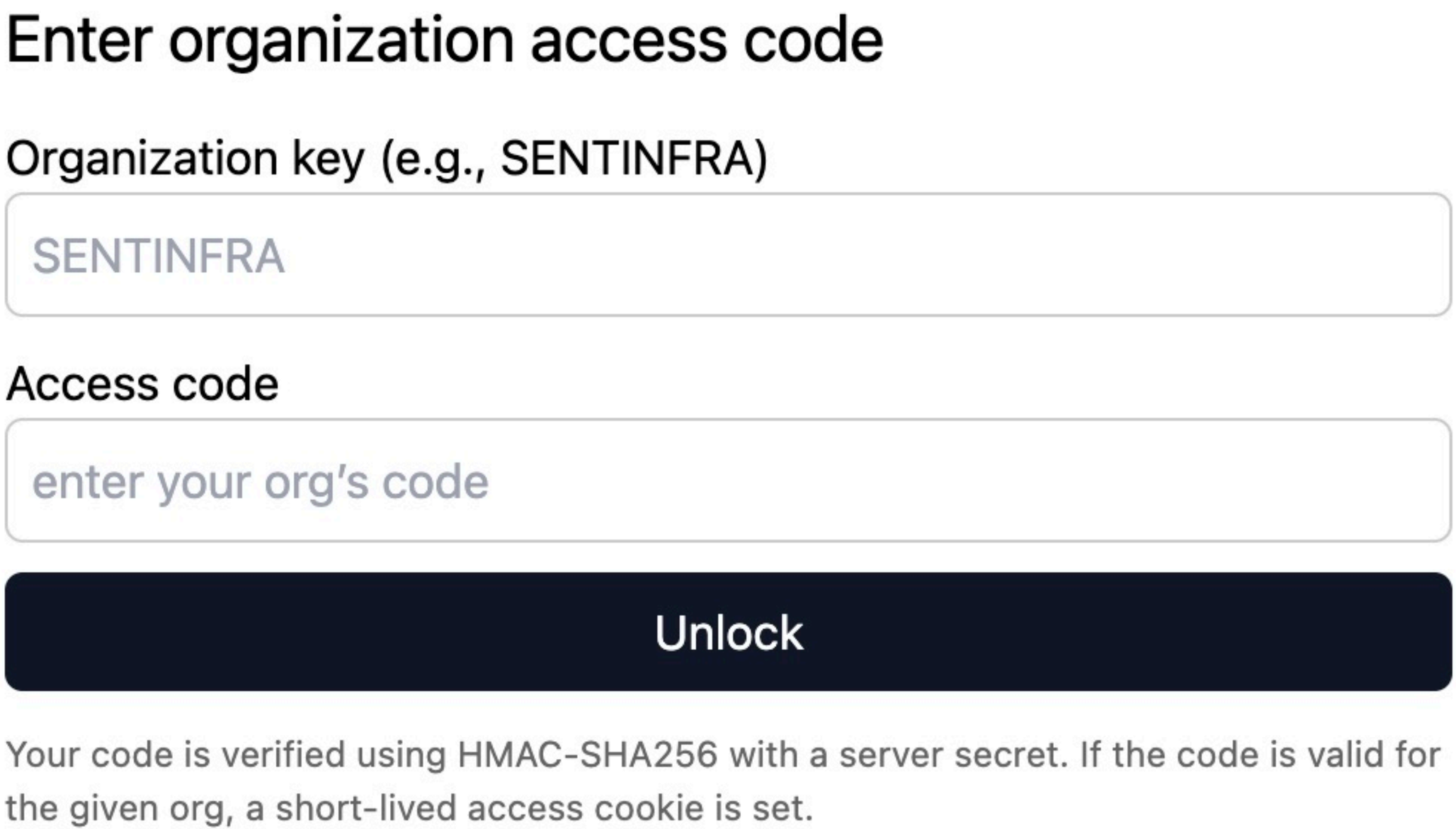
This screenshot shows the Layer-2 authentication interface within the AuditLog.AI e-Signature Gate. The HMAC-SHA256 challenge provides institutional login verification and license control for authorized enterprise users. When a valid organization access code is entered, the client and server perform a symmetric hash comparison against the AuditLog.AI server-side secret, establishing a temporary authenticated session.

This step ensures that only credentialed, paying institutions can access the e-signature environment. The session is time-bound (48 hours) and IP-anchored to the originating network; any change in IP or expiry automatically invalidates the token. The HMAC event is recorded in the audit trail as the institutional component of the two-factor signature system requirement.

File Reference: `AuditLog.AI_WebInterface_HMAC-SHA256___20251013T034046Z.png` **Timestamp (UTC):** `20251013T034046Z` **SHA-256:** `82997a7b21fa553f7b-f8443c0431f819d17b072ba13da9168b9fcfe657d386c5` **RIPEMD-160:** `34c35fe05bb802883ecfd-c3a73d07d9d29d5772d` **OTS File:** `AuditLog.AI_WebInterface_HMAC-SHA256___20251013T034046Z.png.hash.ots` **OTS SHA-256:** `a952e9b292a426170a063a9fbfd-d294a8b752188a442b3b1d52a12ec2d8e8ff6` **OTS Bitcoin TXID:** `eabcee5e6fffe-b6fc6b3c7807165ae863b383fb8459aa0d311726026302ef0e9` **Block:** `918813` **Merkle Root (from OTS proof):** `f161f7b058a1156af142f50a5dc6952b97d2dc897d4e912d6b68dd8119704744` **Date of Existence (UTC):** `2025-10-12T17:07:56Z` *(TXID broadcast time confirming file existence as of Bitcoin block 918813.)*

---

21. HMAC (Layer 2) + Reviewer Login



File Reference: `AuditLog.AI_WebInterface_HMAC-SHA256___20251013T034046Z.png`

22. e-Signature Gate (Web UI Access)

Description:

The AuditLog.AI e-Signature Gate is the controlled interface that authorizes every blockchain-anchoring event. It enforces a multi-layer electronic signature sequence, combining institutional and individual authentication under full human control.

Layer 1 – Individual Ed25519 Digital Signature A personal keypair unique to each reviewer cryptographically binds the signer's identity to the audit session and writes this signature into the immutable audit log.

Layer 2 – Institutional HMAC-SHA256 Authentication Verifies that the signing session originates from a licensed organization by validating the access code against the AuditLog.AI institutional secret. This constitutes the second factor of authentication and confirms enterprise authorization.

Layer 3 – Fraud-Prevention, human-activity verification (Non-Biometric) Layer 3 is an optional, configurable control designed to distinguish genuine human activity from automated or scripted input. It operates under a Zero-Custody data flow and does not capture or process identity, personal traits, or biometrics—therefore it does not constitute biometric authentication.

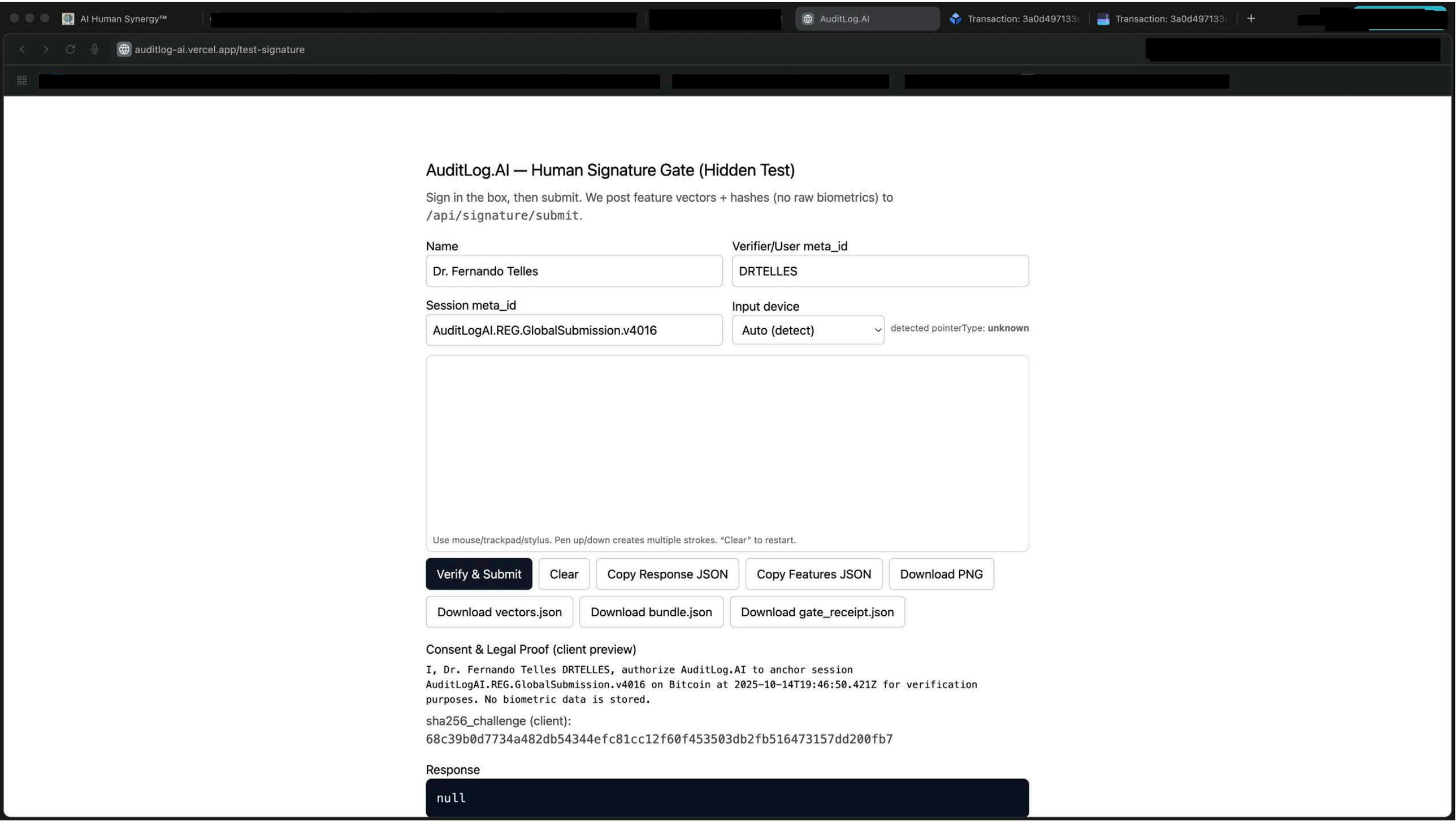
When enabled, minimal motion vectors (cursor or stroke timing data) are transmitted without identifiers to a date-locked, version-controlled LLM1 model operating under validated change control. The model performs a deterministic, evidence-based comparison against fixed human-pattern thresholds and returns a binary PASS / FAIL signal. • Default mode (Regulatory / Production): Layer 3 operates in advisory mode. All results are logged but do not block anchoring. • Strict mode (Enterprise / Pilot): Anchoring pauses on FAIL and requires a deterministic secondary verification (retry, OTP re-auth, or supervisor co-sign). • The reviewer may repeat the signing attempt under the same authenticated Ed25519 + HMAC session; every attempt and result is recorded in the session log.

Layer 3 outputs are version-locked, non-adaptive, and retained for audit transparency. Its use strengthens fraud-prevention controls but is not required for regulatory compliance. Layers 1 and 2 alone (user Ed25519 + institutional HMAC) satisfy all 21 CFR Part 11 and Annex 11 e-signature requirements.

File Reference: `Screenshot 2025-10-15 at 6.49.35 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `3ba9dc776084fcb8b6b5e0f40dcbbd2b2ed-b5143d6037dee78b4b79774e90b49` **RIPEMD-160:** `9a84b52e59a8b28d2efd5386ba066e358ec5a1a8` **OTS File:** `Screenshot 2025-10-15 at 6.49.35 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `9713b2e0aa699eaf2e251abce001922fd0ffe3ec8aa3747d9d12634509eb3a79` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

22. e-Signature Gate (Web UI Access)



File Reference: `Screenshot 2025-10-15 at 6.49.35 am___20251014T200505Z.png`

23. Signature Validation (FAIL Test)

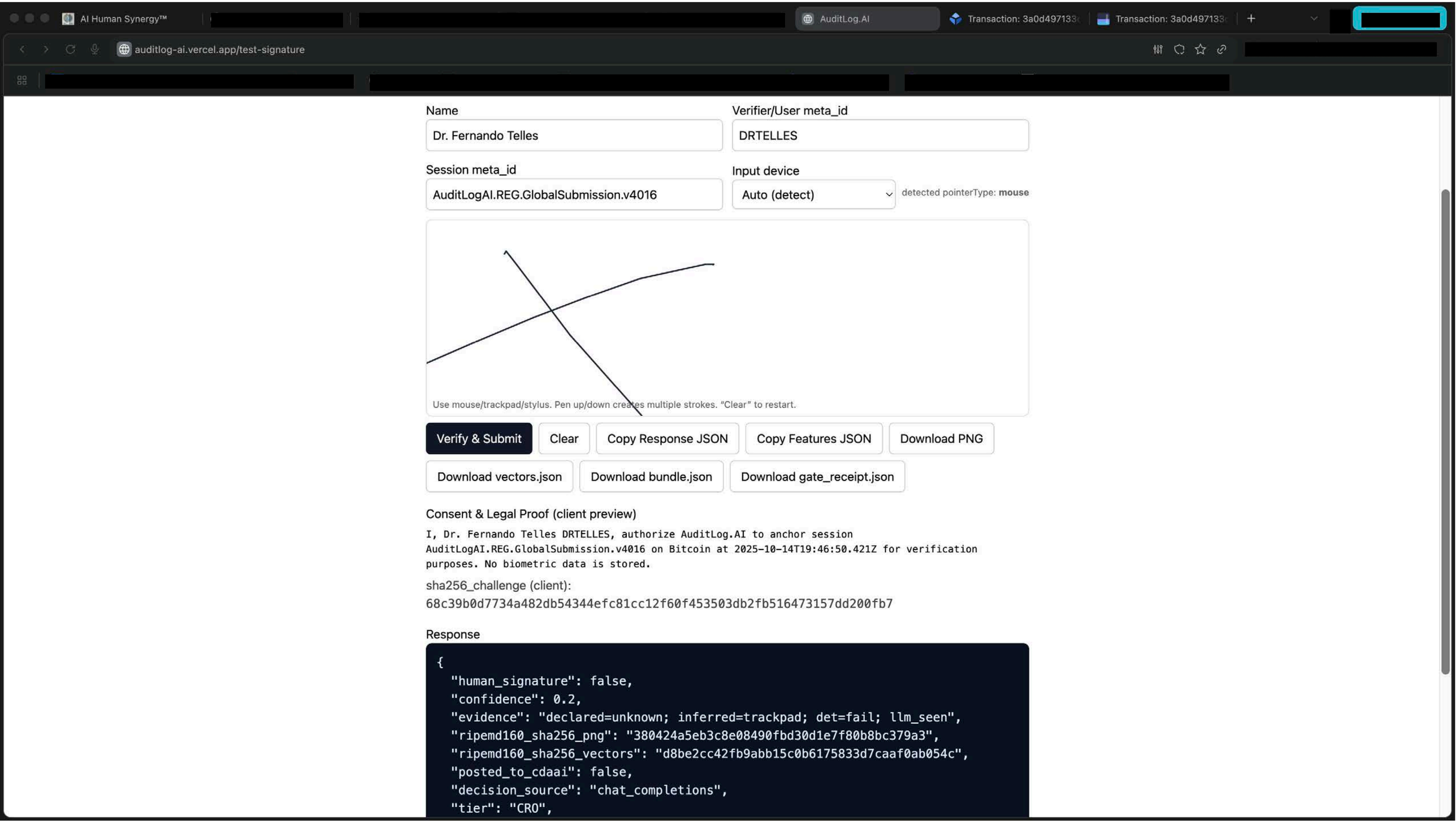
Description:

Strict mode (Enterprise / Pilot) demonstrating a negative-control test in which the Layer 3 human-activity verification returns a confidence score below threshold, producing a deterministic FAIL and blocking creation of the gate_job. The failed attempt is time-stamped and logged in zero-custody Vercel telemetry; no artifacts are generated or transmitted downstream.

File Reference: `Screenshot 2025-10-15 at 6.49.56 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `3f31d4e9c4c6303f14d23fe3d4c1e5ce96fe27e7a4ffb-be581d8cc337bd8fb9d` **RIPEMD-160:** `d10ee26cce112c80ef906f921d642f10fbae42f0` **OTS File:** `Screenshot 2025-10-15 at 6.49.56 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `2c93a32c8f9d6534aa8e865fcc97ea059185178b6dd99b7fac4c874156c04f99` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

23. Signature Validation (FAIL Test)



File Reference: `Screenshot 2025-10-15 at 6.49.56 am___20251014T200505Z.png`

24. Gate Job Not Generated by False Signature

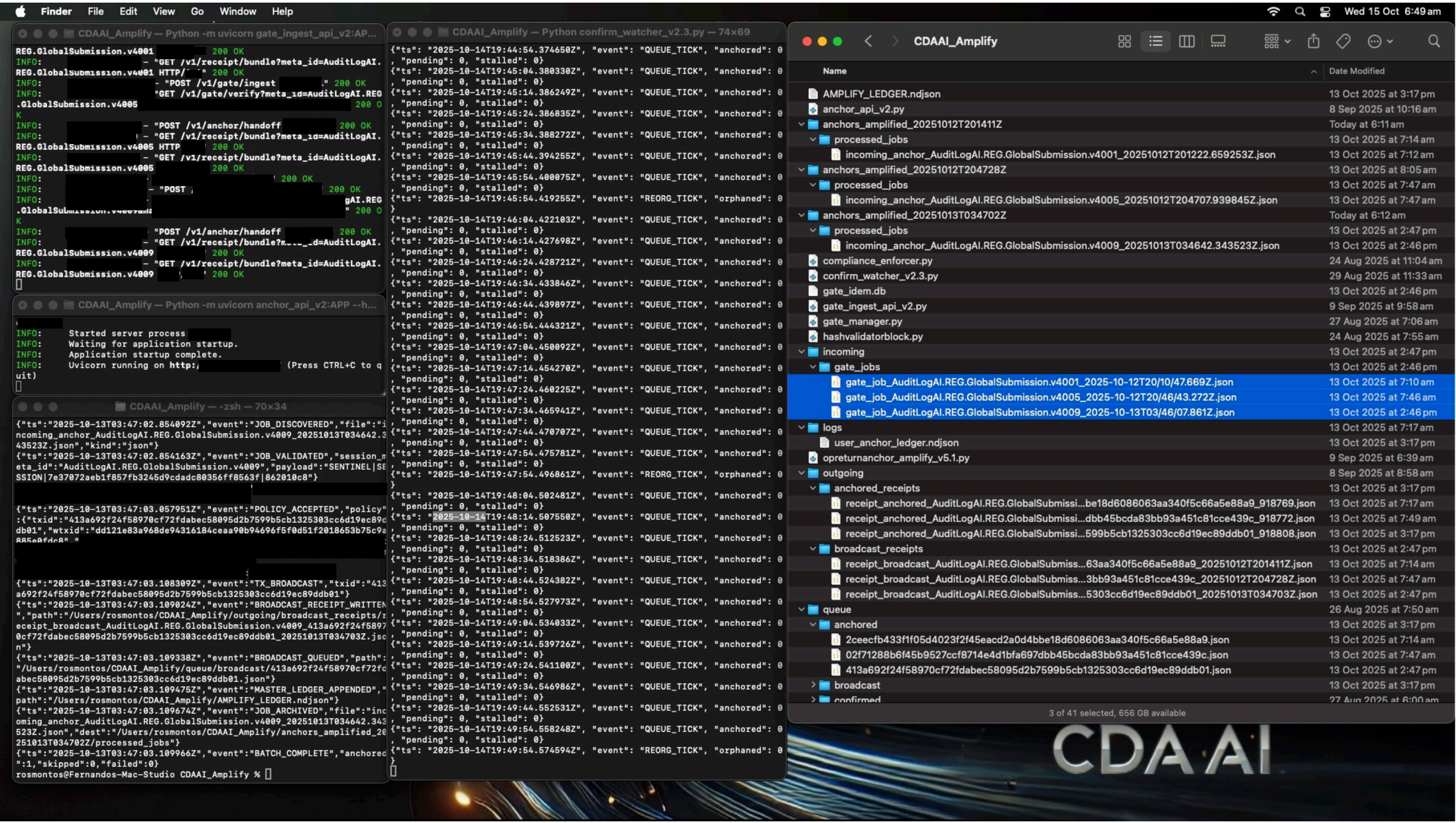
Description:

Post-failure screenshot showing that no gate_job file was produced and no transmission occurred to CDA AI. Blockchain anchoring cannot proceed until a valid e-signature passes all layers. All failed validation events remain recorded within the zero-custody telemetry log, while only successful, human-verified e-signature events generate a gate_job JSON for regulated anchoring.

File Reference: `Screenshot 2025-10-15 at 6.49.56 am (2)___20251014T200505Z.png`
Timestamp (UTC): `20251014T200505Z` **SHA-256:** `1725b1e60d3aef30c8f3f89f5f8f00280334d-b69651d655e5f7551d5d5c39be2` **RIPEMD-160:** `06ea45793c609d6674008ddb9475c8a44bcb509e`
OTS File: `Screenshot 2025-10-15 at 6.49.56 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `560da5db370d96ee143586d16173de35edabc059f58807e6a5f24de8a91ffffb` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084`
Merkle Root (from OTS proof): `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

24. Gate Job Not Generated by False Signature



File Reference: `Screenshot 2025-10-15 at 6.49.56 am (2)___20251014T200505Z.png`

25. Signature Validation (PASS Test)

Description:

Positive-control example of a validated human e-signature achieving confidence ≥ 0.75. All three layers succeed: - Ed25519 signer key matches prior registration (unique ID = reviewer). - HMAC-SHA256 challenge verified by AuditLog.AI authentication server. - Sentinel LLM1 vector analysis confirms signature authenticity (confidence 0.85 → PASS).

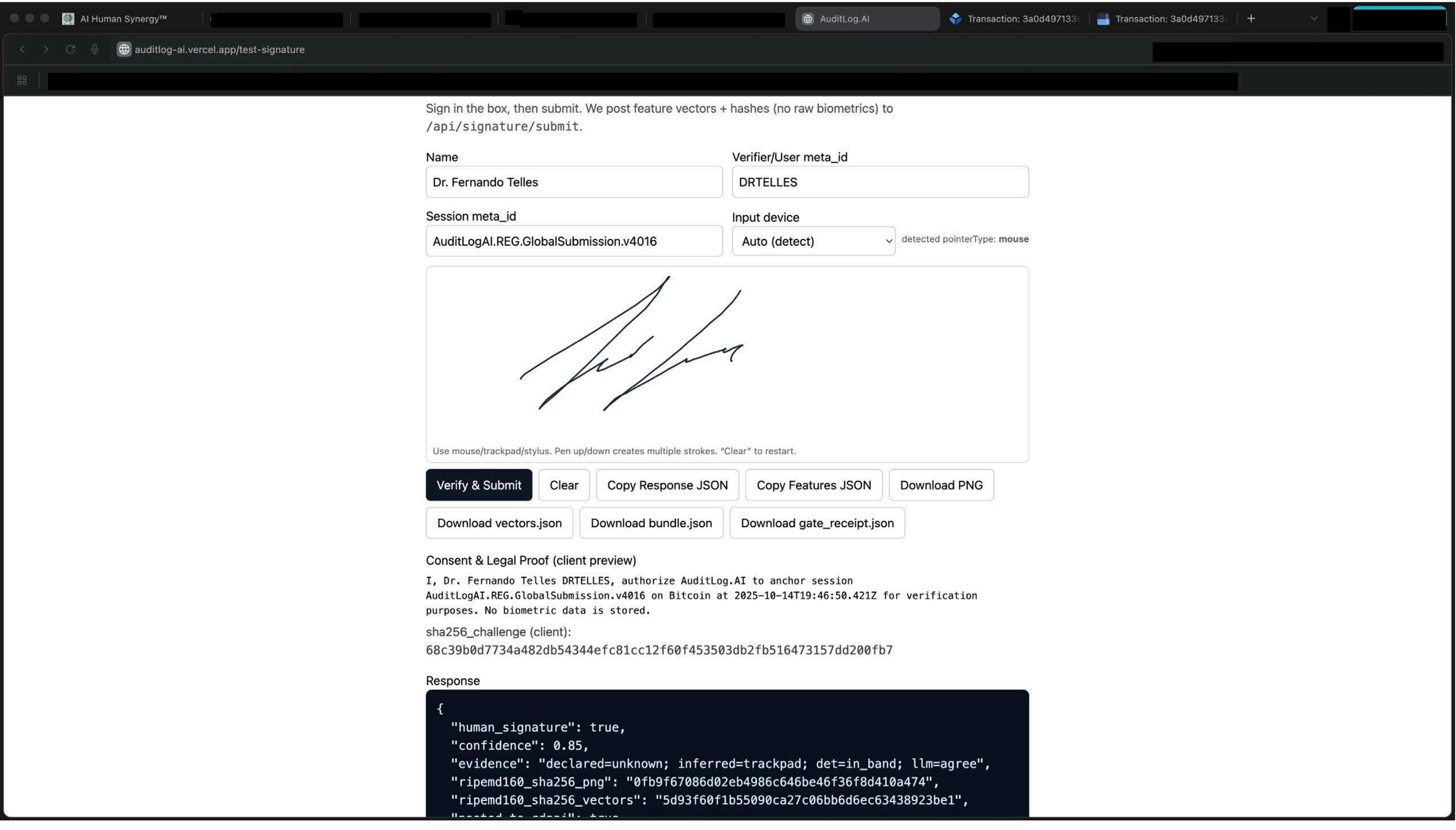
The following artifacts are generated and hashed: - Signature image PNG - Signature vectors JSON - Consent & Legal Proof statement - LLM1 analysis JSON

Operator has the option to download these raw artifacts individually or as a bundle, while AuditLog.AI receives only cryptographic proofs. Reproducibility of these artifacts' hashes has been confirmed by `AUDIT_REPRODUCIBILITY_PASS_AuditLog.AI_SentinelLLM1_sha256_ripemd160_v.aug31.25___20251012T195229Z.md` (logged in regulatory session 1 `session_log_AuditLogAI.REG.GlobalSubmission.v4001_20251012T200455.617478Z.json`).

File Reference: `Screenshot 2025-10-15 at 6.50.34 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `b75bd8e73fef4fdcd36b58477c478d74cdbbc281c8e43e6c7c3279a371b29756` **RIPEMD-160:** `8fecf1e37f2b896bb440d208c772a51b2bc1bdd5` **OTS File:** `Screenshot 2025-10-15 at 6.50.34 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `2b2eb0df3d17bfbe06f1909585df8585ec6afb501afbc545f81b02e5d2f84091` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

25. Signature Validation (PASS Test)



File Reference: `Screenshot 2025-10-15 at 6.50.34 am___20251014T200505Z.png`

26. AuditLog.AI Pass Signature Registered – Gate Job Created

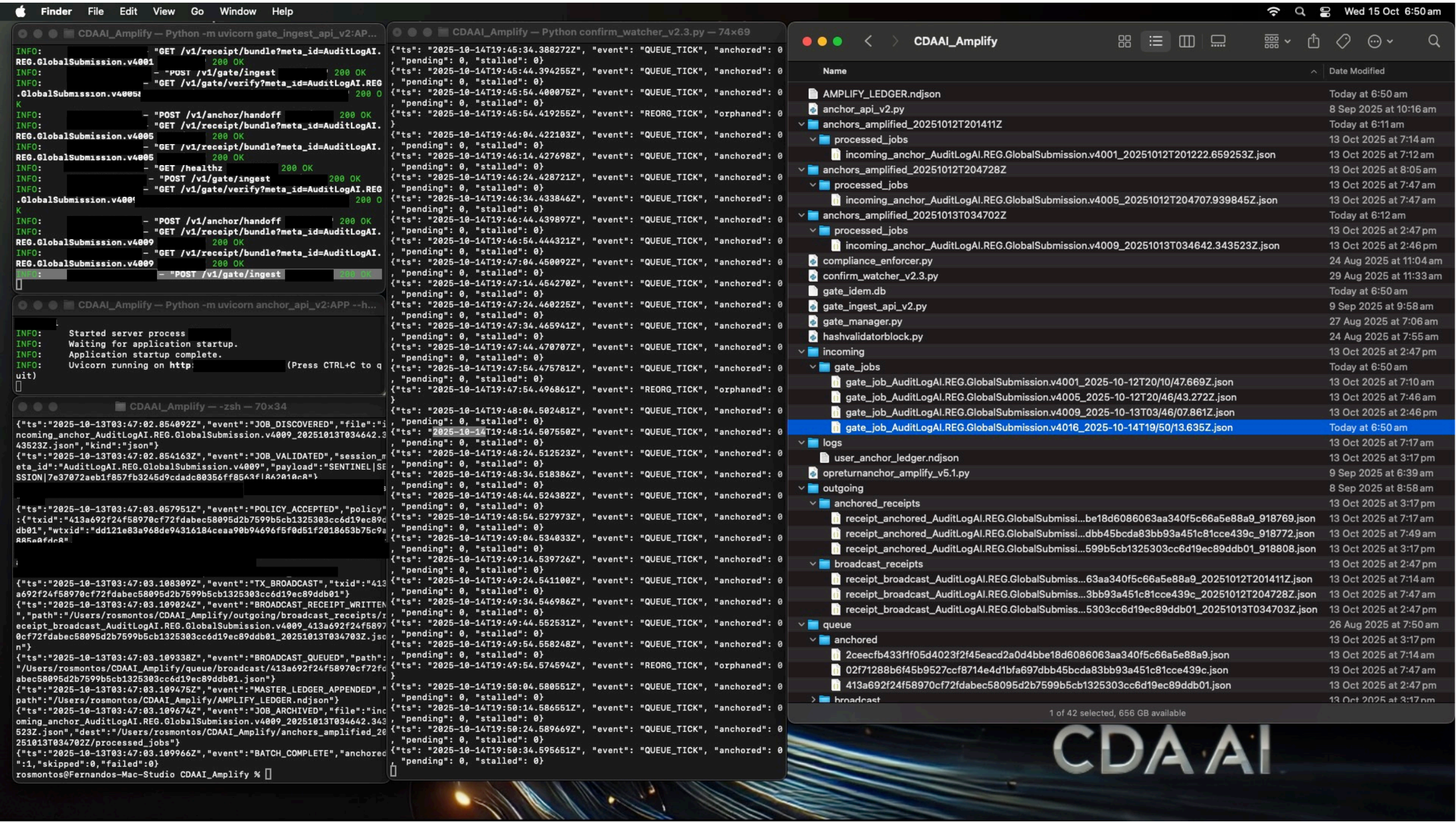
Description:

Screenshot of gate job (`gate_job_<meta_id>_<UTC>.json`) generation and transmission to AuditLog.AI following a PASS result from AuditLog.AI e-Signature Gate. Gate job transmission alone does not authorize AuditLog.AI to proceed with blockchain anchoring, one final human authorization is required (28.).

File Reference: `Screenshot 2025-10-15 at 6.50.34 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `88f9632b400c696f95baf12575f4d1e6fc4ed0a1c962e97756bb55158b910937` **RIPEMD-160:** `0b77da80694a9462792ff5ef77f0d60bf6e3da33` **OTS File:** `Screenshot 2025-10-15 at 6.50.34 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `c6d9589d7f0a311bcbcf1d7699c72524517e44710e00f380e2f6f568e8416774` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

26. AuditLog.AI Pass Signature Registered – Gate Job Created



27. Gate Job (Hashes Only)

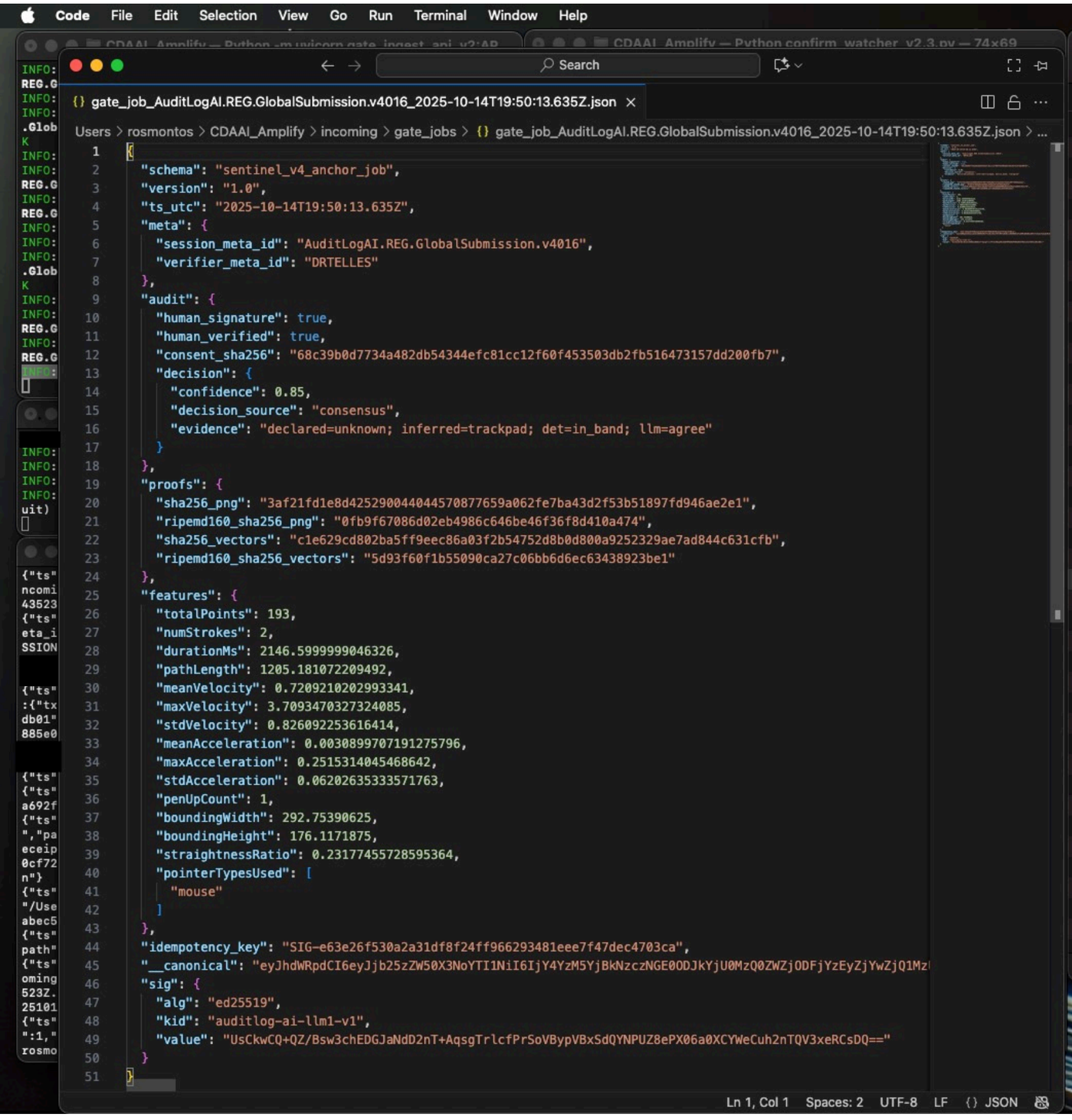
Description:

View of `gate_job_<meta_id>_<UTC>.json` confirming Zero-Custody design. Containing timestamp, cryptographic hashes, vectors, LLM1 analysis, and minimal metadata (session meta_id and human verifier meta_id).

File Reference: `Screenshot 2025-10-15 at 6.50.59 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `964260dfc1a3eda6973dfd0e8df58e950590a950169bb43c09343e1018ecb455` **RIPEMD-160:** `40aed844d14041ff85d6c3b61d54b9debda95c71` **OTS File:** `Screenshot 2025-10-15 at 6.50.59 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `bb22b5e4b49e9fea15e3fe418ec540f00241754ce7c7bae653d1e41ee95cbabb` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

27. Gate Job (Hashes Only)



File Reference: `Screenshot 2025-10-15 at 6.50.59 am (2)___20251014T200505Z.png`

28 A & B. Final Human Authorization Blockchain Anchoring

Description:

This screenshot captures the final human-authorization checkpoint preceding blockchain anchoring, a mandatory interactive step by `anchor_candidate_cli_v6.1.py`. Upon operator approval, it derives the OP_RETURN payload from the frozen session log, and sends artifacts for final blockchain anchoring authorization to AuditLog.AI.

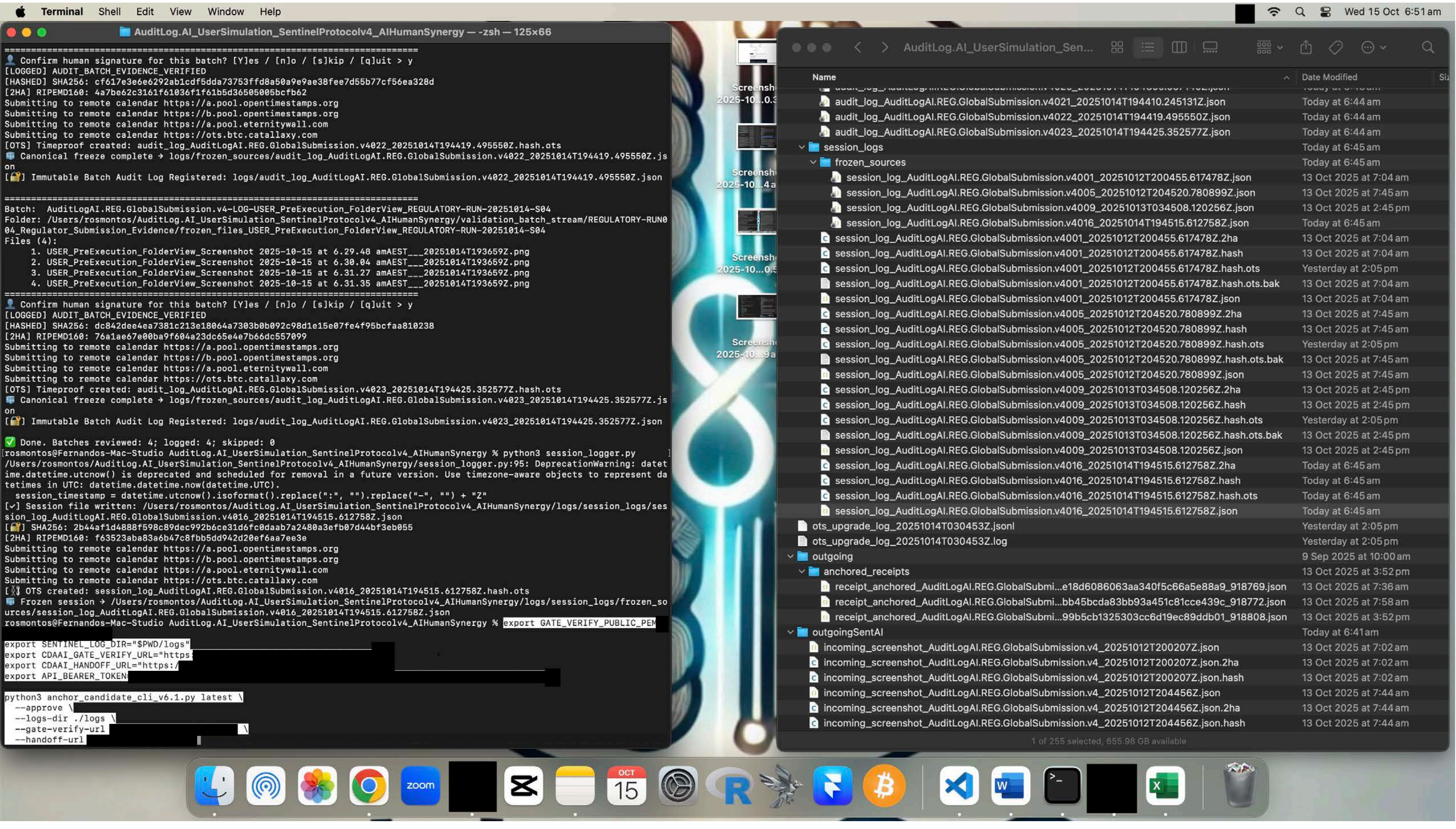
At this stage, all prior validations, hash generation, VALIS enforcement, and compliance checks have passed. Anchoring cannot proceed automatically; the process is hard-gated by the Ed25519-signed authorization of a verified user (the same unique reviewer registered under the AuditLog.AI e-Signature Gate), and bound to intitutional HMAC login and meta ID. Only once this explicit approval is confirmed does the system flag the session as approved for broadcast, enabling immutable publication to the Bitcoin mainnet.

File Reference 28 A: `Screenshot 2025-10-15 at 6.51.51 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `4b910fe7662715c3300b8625c3f6700a496746a42f97b4298dfb77a0f1d934f1` **RIPEMD-160:** `1992ea22332ec4707597578a3f53d6e6721855d2` **OTS File:** `Screenshot 2025-10-15 at 6.51.51 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `cd-d32e33c10d0f7ed16c2c6e927e4f9d640e904f7bf0f86f2b2d85cbe25fc1ef` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

File Reference 28 B: `Screenshot 2025-10-15 at 6.52.12 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `1d1e7c06f49f91910afcd6492687930587d54457b3c9126b743535910e632374` **RIPEMD-160:** `1fc5492096a855e9157e5e1d263467c49bebee19` **OTS File:** `Screenshot 2025-10-15 at 6.52.12 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `b8f4319b7969b37a1933688c0b84e7629e6465655db460e312776421a30335ff` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

28 A. Final Human Authorization Blockchain Anchoring



28 B. Final Human Authorization Blockchain Anchoring

29. Post-Final Authorization — Incoming Anchor Job Arrives at AuditLog.AI Node

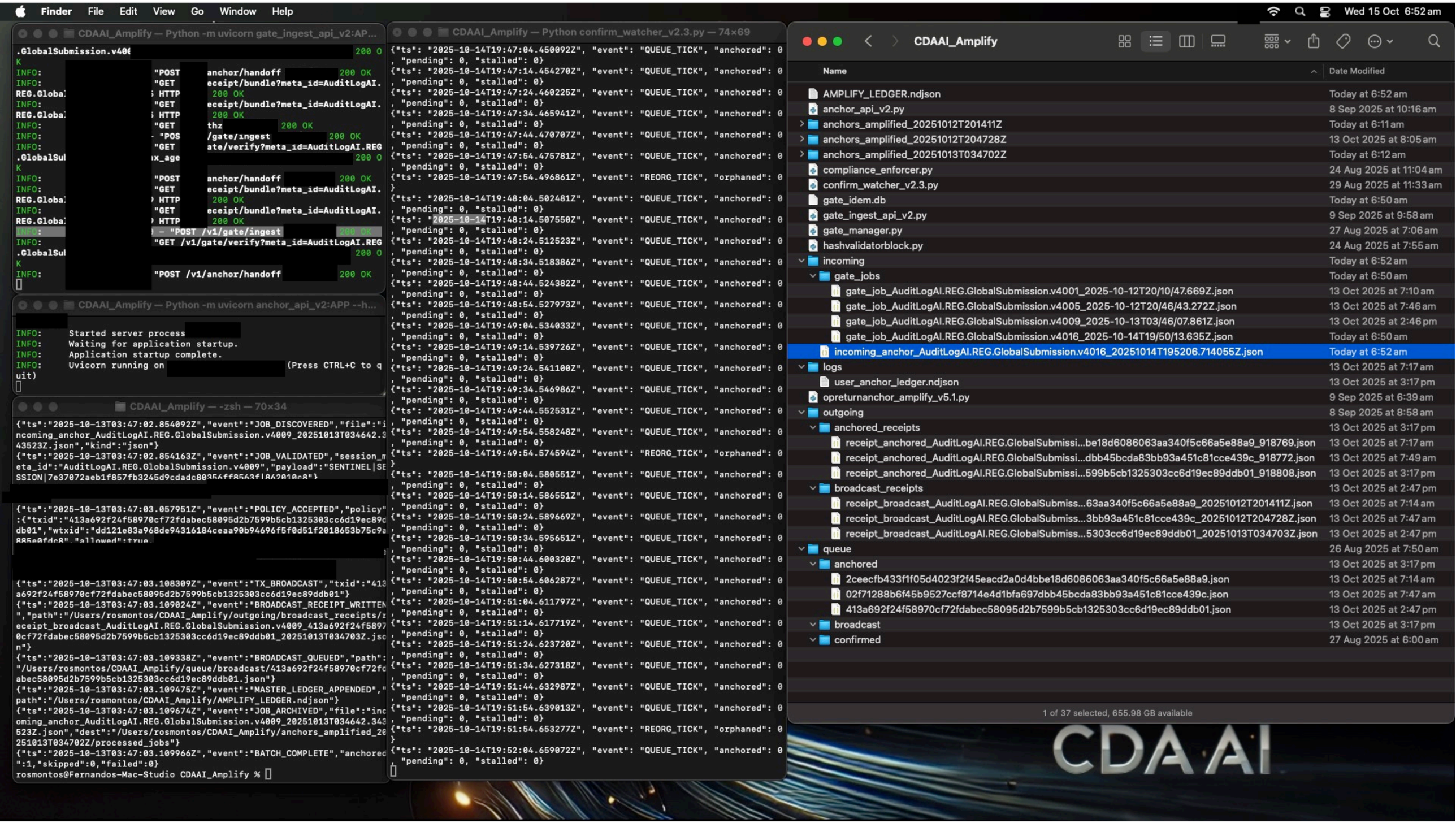
Description:

AuditLog.AI receives the finalized user-approved handoff job from the user environment following successful e-signature gate verification (incoming_anchor JSON).

File Reference: `Screenshot 2025-10-15 at 6.52.12 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `63b142f1ff7d59903d3b01d91ea752be3aa33ff2b9167f1b85e8608f66a0f003` **RIPEMD-160:** `65977b-ccac1119769e7d591ed13f62f7f23d3751` **OTS File:** `Screenshot 2025-10-15 at 6.52.12 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `94e94706a4a5fc-c85b1c56a5955c7d98b2cd3a314a79123da5642170c53a9d80` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

29. Post-Final Authorization — Incoming Anchor Job Arrives at AuditLog.AI Node



30. AuditLog.AI Incoming Anchor Job Receipt

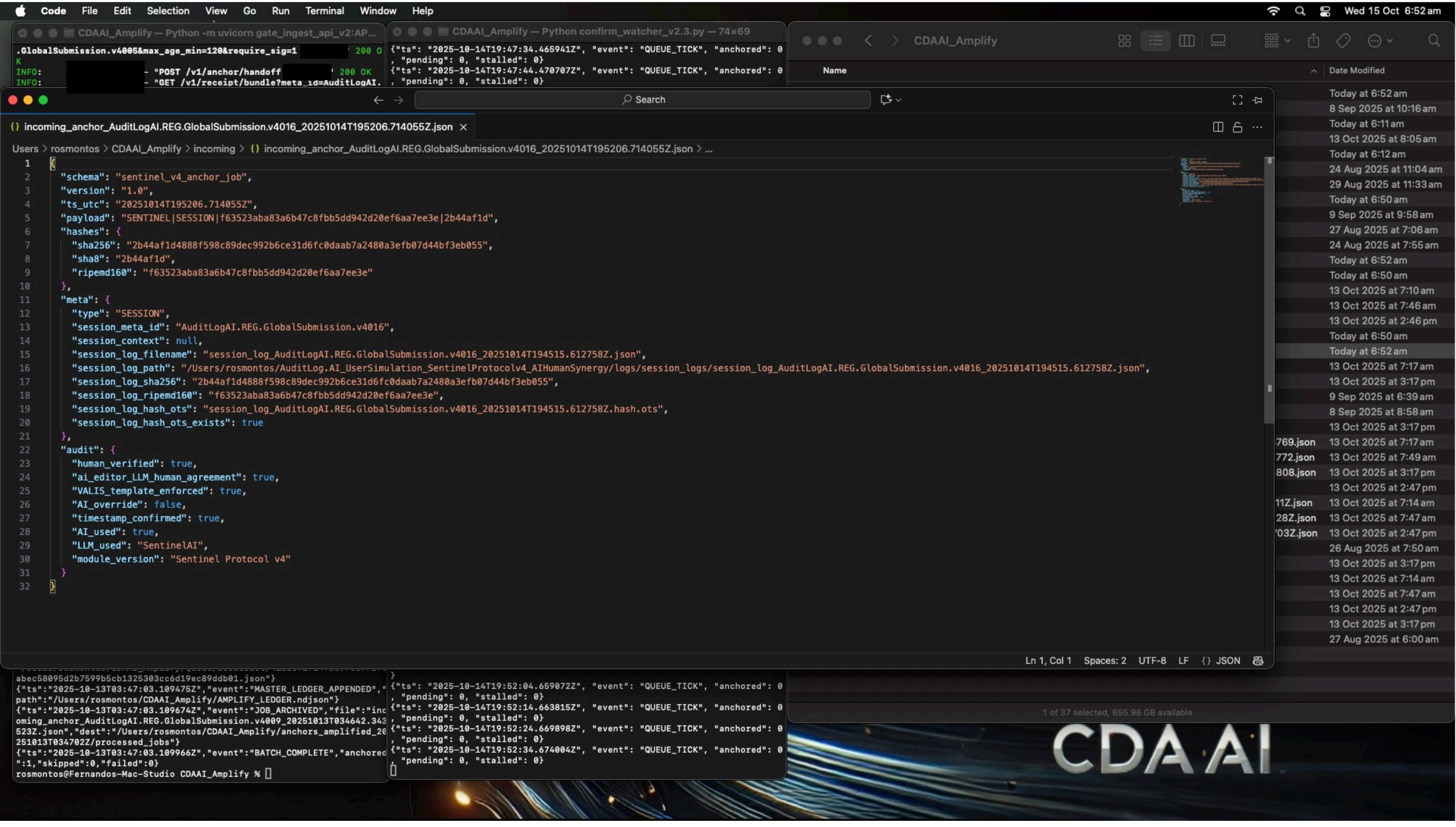
Description:

The handoff package contains only **hashes and metadata**, ensuring **zero-custody** compliance.

File Reference: `Screenshot 2025-10-15 at 6.52.44 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `13a04cb81a8e3497b-b52b92de05d895b7eca87846fab8f8da9c450d1e10c09d8` **RIPEMD-160:** `2ca67a8e38c5b-c07492176a6fd60a25633aa6760` **OTS File:** `Screenshot 2025-10-15 at 6.52.44 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `034d9c54854307f9fd99f-b5005418c8ba6add13a80870388fb44919af2c766ce` **OTS Bitcoin TXID:** `5c28773f653c441226d-f3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

30. AuditLog.AI Incoming Anchor Job Receipt



31. AuditLog.AI OP_RETURN Amplify Execution

Description:

Console output of `opreturnanchor_amplify_v5.1.py` executing Bitcoin transaction broadcast. This operation is only possible if incoming anchor job and gate job files are verified under CME Rules.

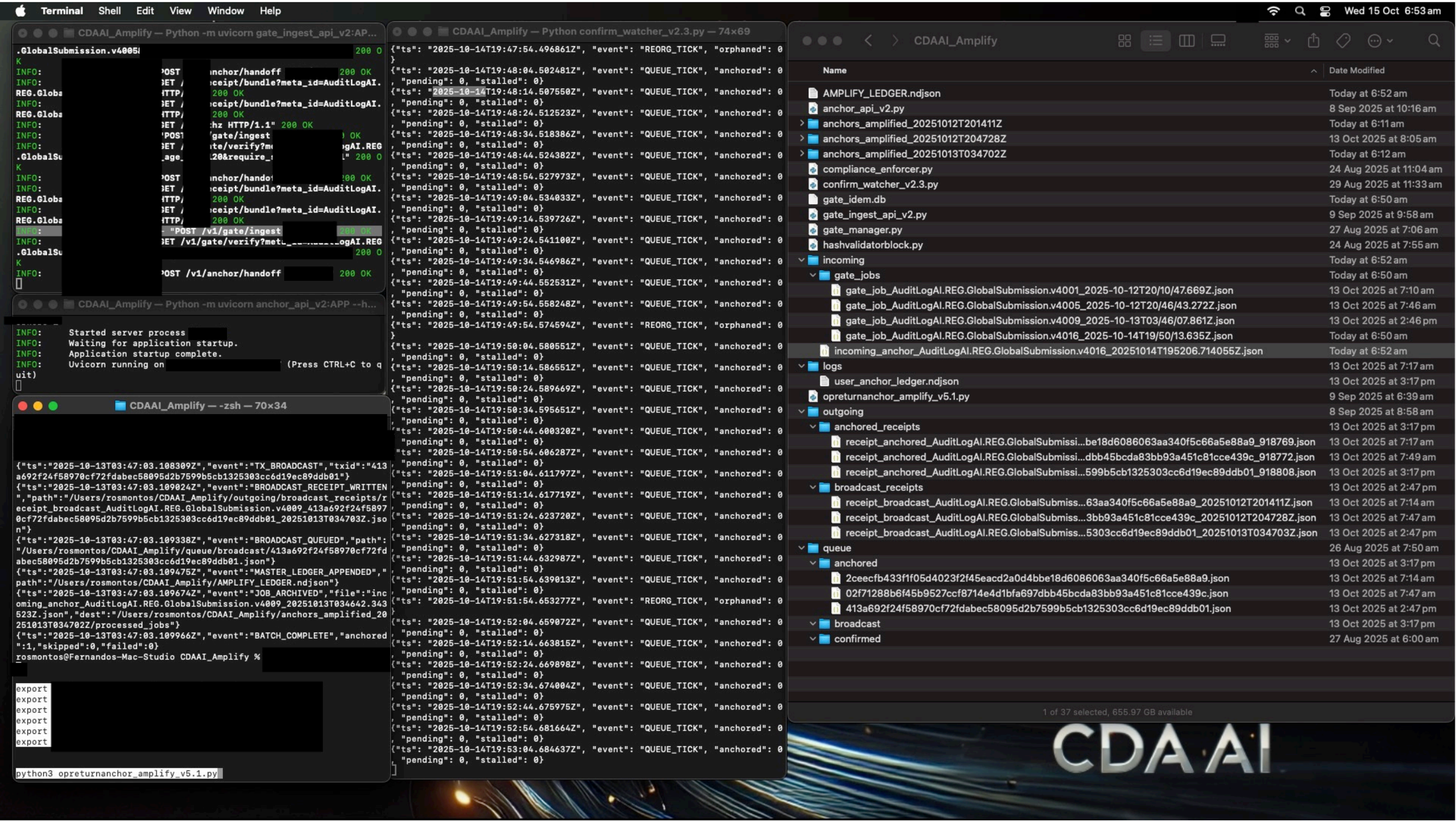
Key compliance confirmations recorded in this event:

- `human_verified: true`
- `ai_editor_LLM_human_agreement: true`
- `VALIS_template_enforced: true`
- `AI_override: false`
- `timestamp_confirmed: true`
- The payload schema conforms to **C5.3.1 OP_RETURN specification** (`SENTINEL|<type>|<ripemd160>|<sha8>`)
- session log hashes parity and `.hash.ots` presence are confirmed

File Reference: `Screenshot 2025-10-15 at 6.53.13 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `301dd7c5b7358523d605fc-c8f28685503a14a3e44889a5905130994e8e87f240` **RIPEMD-160:** `bd2b8bc14e1369682347825f-d7bec3150886b934` **OTS File:** `Screenshot 2025-10-15 at 6.53.13 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `75a6176d6a89e33d-d487992ee214f1cb573711f99de9a94a4f19f9720f3ecf53` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

31. AuditLog.AI OP_RETURN Amplify Execution



32. Broadcasted On-Chain via Bitcoin Node — Amplifier Broadcast Event

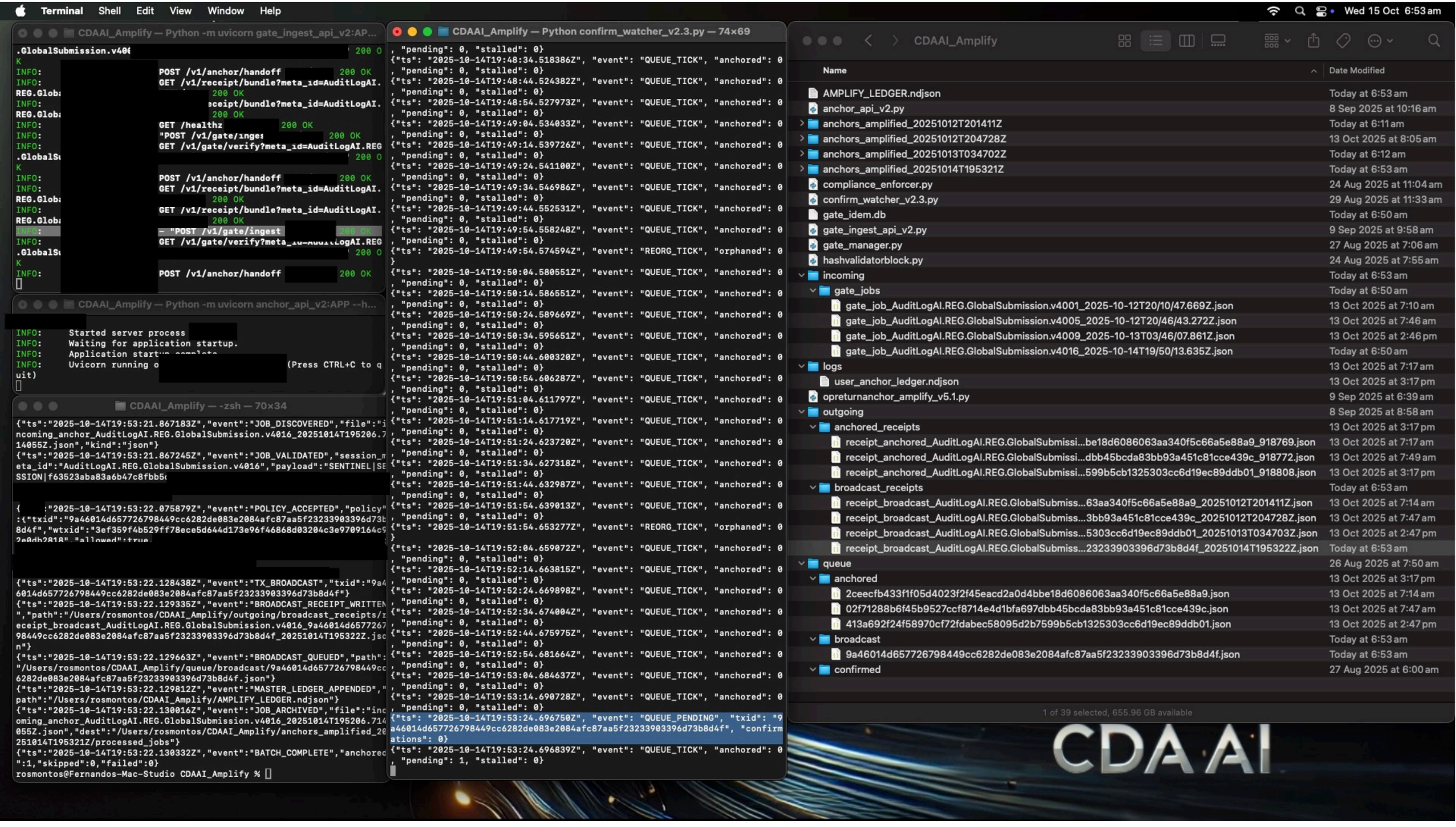
Description:

Verified payload broadcasted. OP_RETURN transaction successfully constructed and submitted to Bitcoin mainnet. Confirm watcher (`confirm_watcher_v2.3.py`) registers broadcast and monitors for on-chain confirmation.

File Reference: `Screenshot 2025-10-15 at 6.53.33 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `e2b3f3b325cc6e2a9fe03bb1b304ff1b-b457c0ce3a6fde11b96f996f0774eb17` **RIPEMD-160:** `597fbbec6aa47d611da8d09e62983031feaea30d` **OTS File:** `Screenshot 2025-10-15 at 6.53.33 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `6f57096699dd99448da03d1a6c691d8395a5eb5d8535cb34120e1e79a8731be6` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

32. Broadcasted On-Chain via Bitcoin Node — Amplifier Broadcast Event



33. Receipt Broadcast View

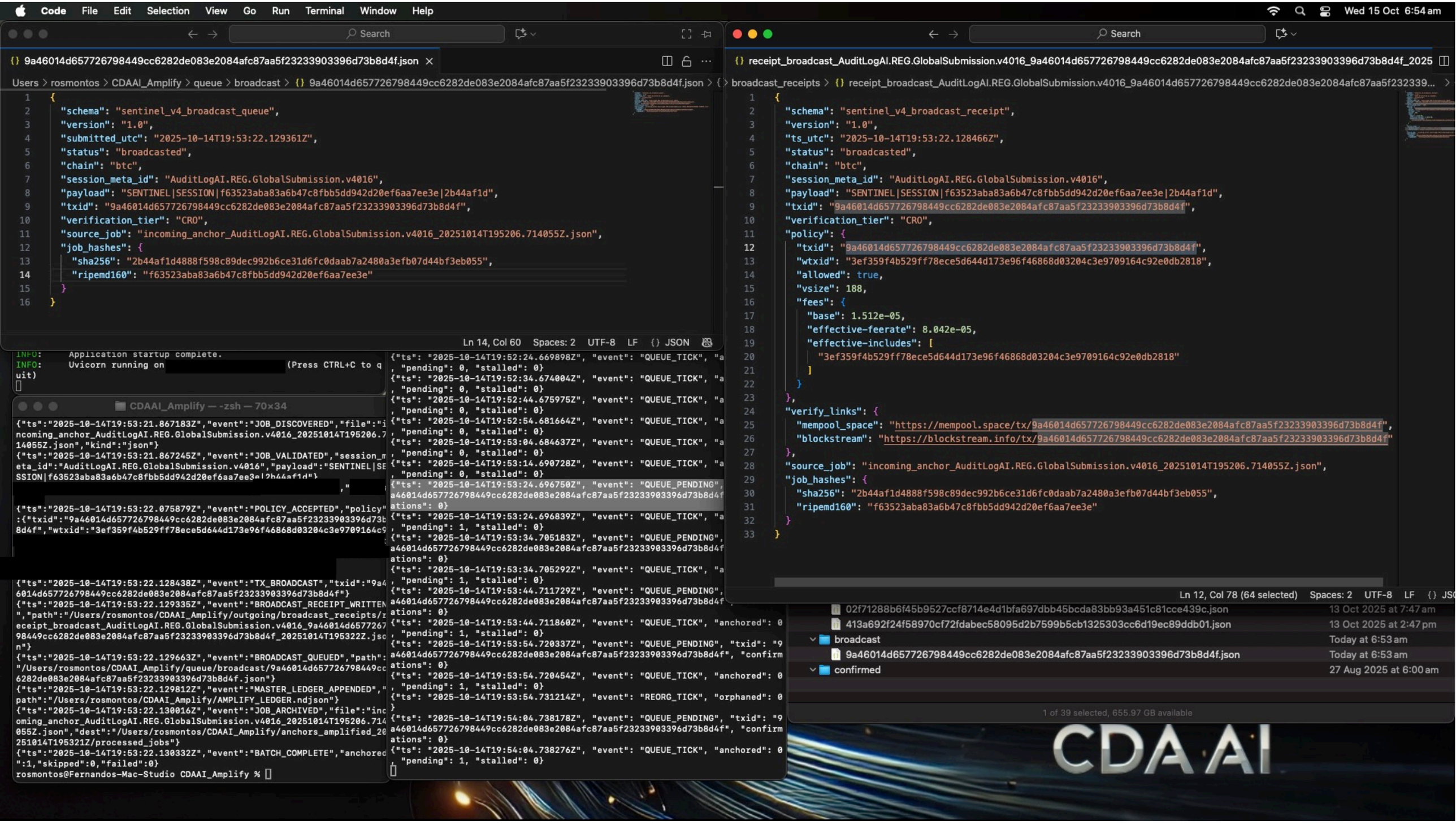
Description:

AuditLog.AI broadcast receipt confirming transaction ID and payload. Documents initial on-chain presence of the payload and writes a status:"broadcasted" record to AMPLIFY_LEDGER.

File Reference: `Screenshot 2025-10-15 at 6.54.10 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z.png` **SHA-256:** `deeb9656bf1608018e22a4e-b2b6b0cfc5197748f07e8c686f3d8eb50b4bde0c3` **RIPEMD-160:** `3a36e40aa5ca3aca4d838ee714534e743224f758` **OTS File:** `Screenshot 2025-10-15 at 6.54.10 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `78713779bfb20491da0ff646c676e62307ce3bc3c006ec239646189e17cc10f0` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

33. Receipt Broadcast View



34. Public Explorer Broadcast Confirmation

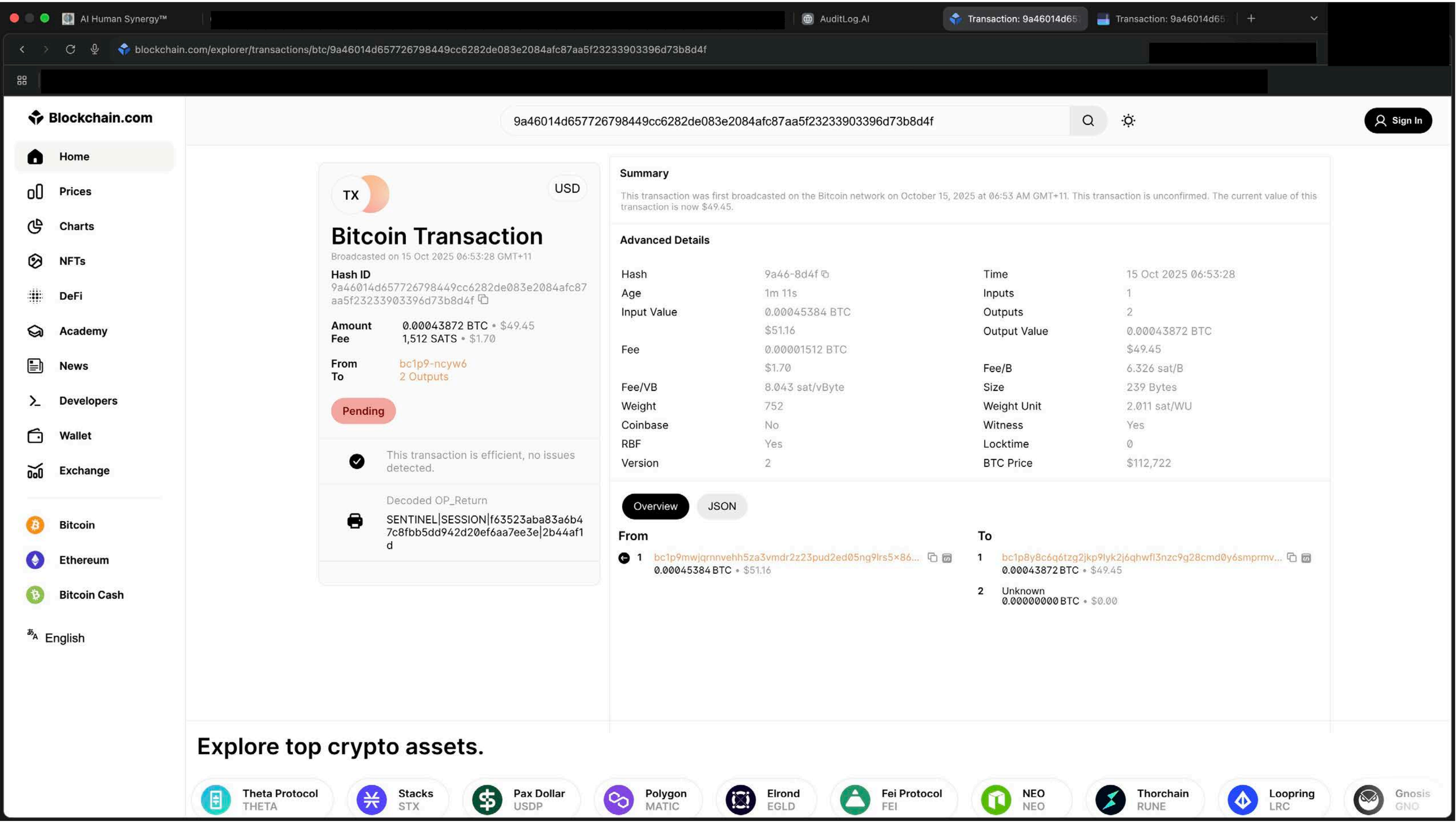
Description:

Independent blockchain explorer (blockchain.com) view of transaction ID and payload. Confirms broadcast and alignment between AuditLog.AI receipt and network state.

File Reference: `Screenshot 2025-10-15 at 6.54.40 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `f1e05fbfad6b-d2f138da9d5f32ae47a230e2e166523c07a6cf810fabdc6f86d7` **RIPEMD-160:** `c8b97e68b2727716135848d61f47358eb2d6f377` **OTS File:** `Screenshot 2025-10-15 at 6.54.40 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `e1a7cb544f37b1e6d8666d8b575-ab42b43c817359b2c0affaffc72ddcdb06269` **OTS Bitcoin TXID:** `5c28773f653c441226df3079b-f1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

34. Public Explorer Broadcast Confirmation



35. Anchor Confirmation Watcher

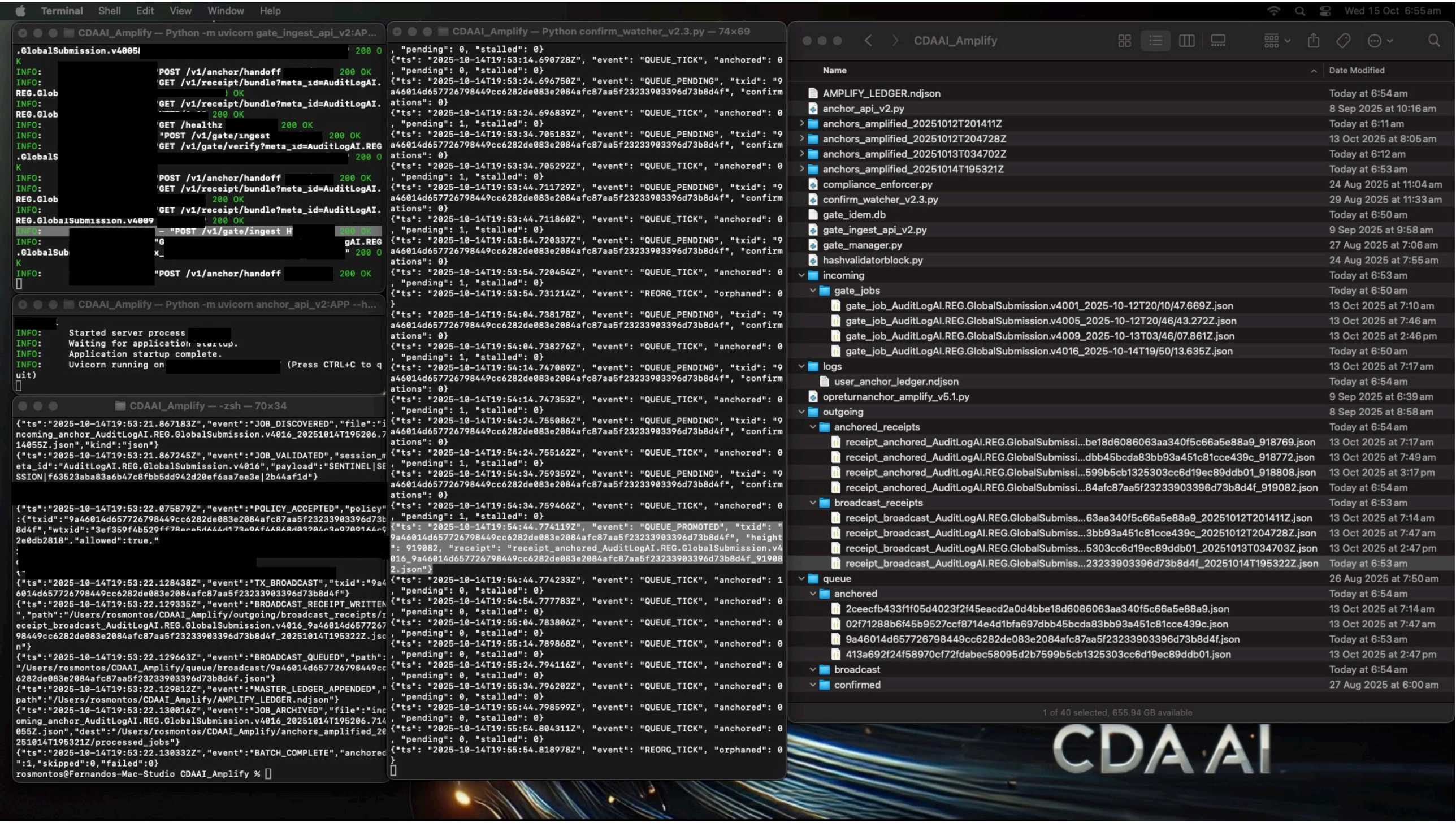
Description:

Console output of confirm watcher detecting transaction confirmation and promoting status from broadcasted to anchored. Performs block-height verification, and automatically appends "event":"ANCHORED_RECEIPT_WRITTEN" to both AMPLIFY ledger and the user ledger (AuditLog.AI held copy). Anchoring finalized; immutable dual-ledger entries created with verified block height.

File Reference: `Screenshot 2025-10-15 at 6.55.57 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `534fc97ace4cd124e24e3bb-bea02c43c98b0da2e96aa091c5270a2b31b89d9dc` **RIPEMD-160:** `97438cd232bac3e3fd0e-f96f7d41471a4f6a8030` **OTS File:** `Screenshot 2025-10-15 at 6.55.57 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `71ee6254d15def87478f9d-b4a6b338f6616b70de52ec25fe78a6d7e939bda6cc` **OTS Bitcoin TXID:** `5c28773f653c441226d-f3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

35. Anchor Confirmation Watcher



36. Anchored Transaction – Public Explorer View #1

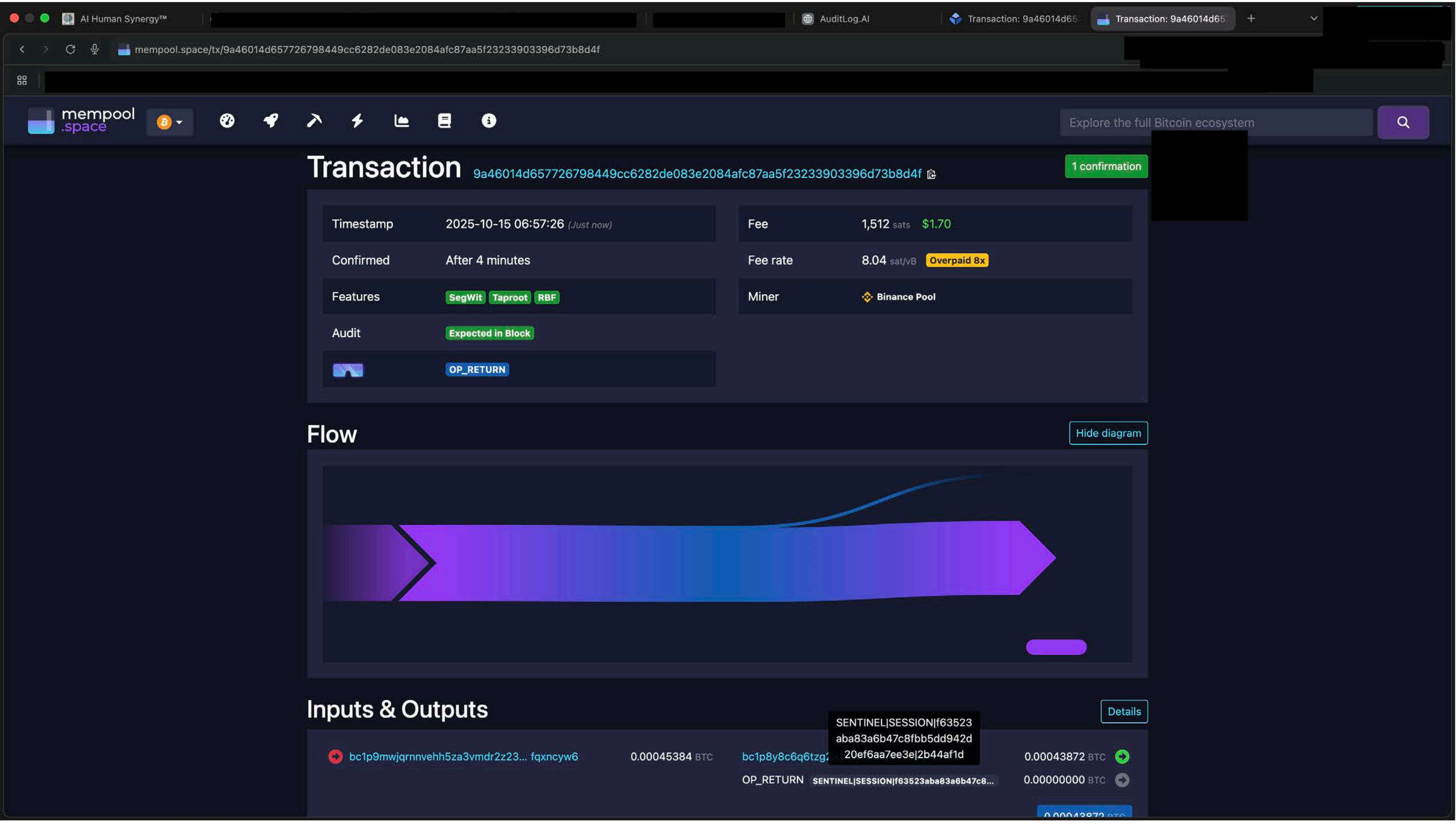
Description:

Explorer screenshot (mempool.space) showing confirmed Bitcoin transaction and OP_RETURN payload. Demonstrates public immutability of the anchor proof and final block inclusion. - No identifiers are published, only hashes, ensuring complete user anonymity. - Blockchain confirmation independently verified; external audit reference established.

File Reference: `Screenshot 2025-10-15 at 6.54.53 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `4e38d40a89af59ed-b96ead6d17c611421c2180558531b14e03f4ac8640b9d73d` **RIPEMD-160:** `b237b71251d0d-b6c71ecd83aeea94a591b1f3c8e` **OTS File:** `Screenshot 2025-10-15 at 6.54.53 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `d6357e9de8411f16028baa55ad715b37e6f5b5130cbc9d7dd797ca48405132f6` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

36. Anchored Transaction – Public Explorer View #1



File Reference: `Screenshot 2025-10-15 at 6.54.53 am___20251014T200505Z.png`

37. Anchored Transaction – Public Explorer View #2

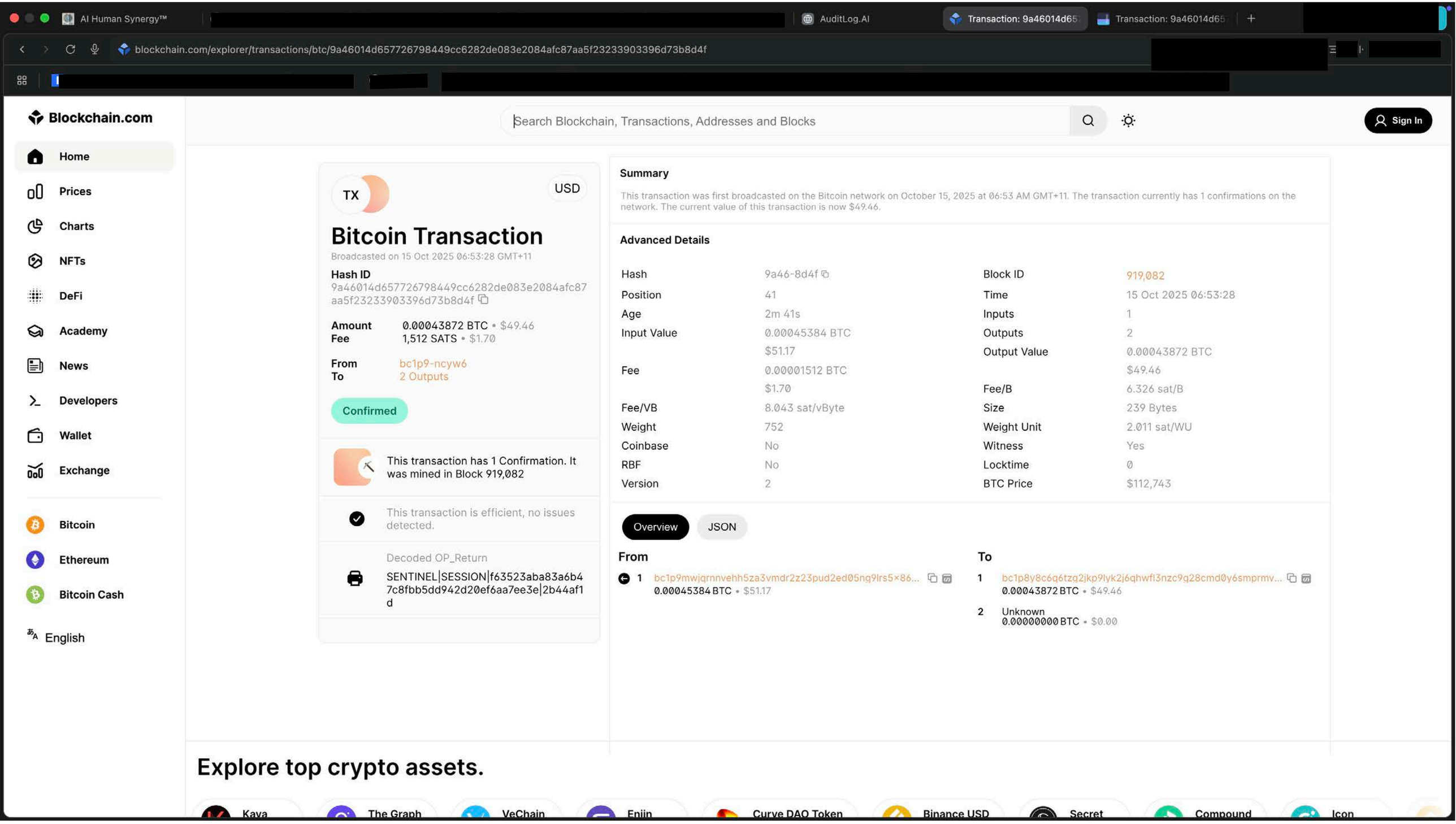
Description:

Alternate blockchain-explorer confirmation (blockchain.com). Displays block height, TXID, payload, and confirmation. Cross-verified confirmation validates network-level reproducibility and external traceability.

File Reference: `Screenshot 2025-10-15 at 6.56.10 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `498dd0c8a5248f1704b5d5ab01476851a7984b-b4f787b3b2699d7298acab26d9` **RIPEMD-160:** `7e4dbc52d2f4cfd95d2a402f7fea1b7bf8ca5ec0` **OTS File:** `Screenshot 2025-10-15 at 6.56.10 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `f46e0defff2e9423baffa220fc5adc479c3c4021e9c9c1a5f79d538f0b0135ec` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

37. Anchored Transaction – Public Explorer View #2



38. Session Log Digest Match Validation

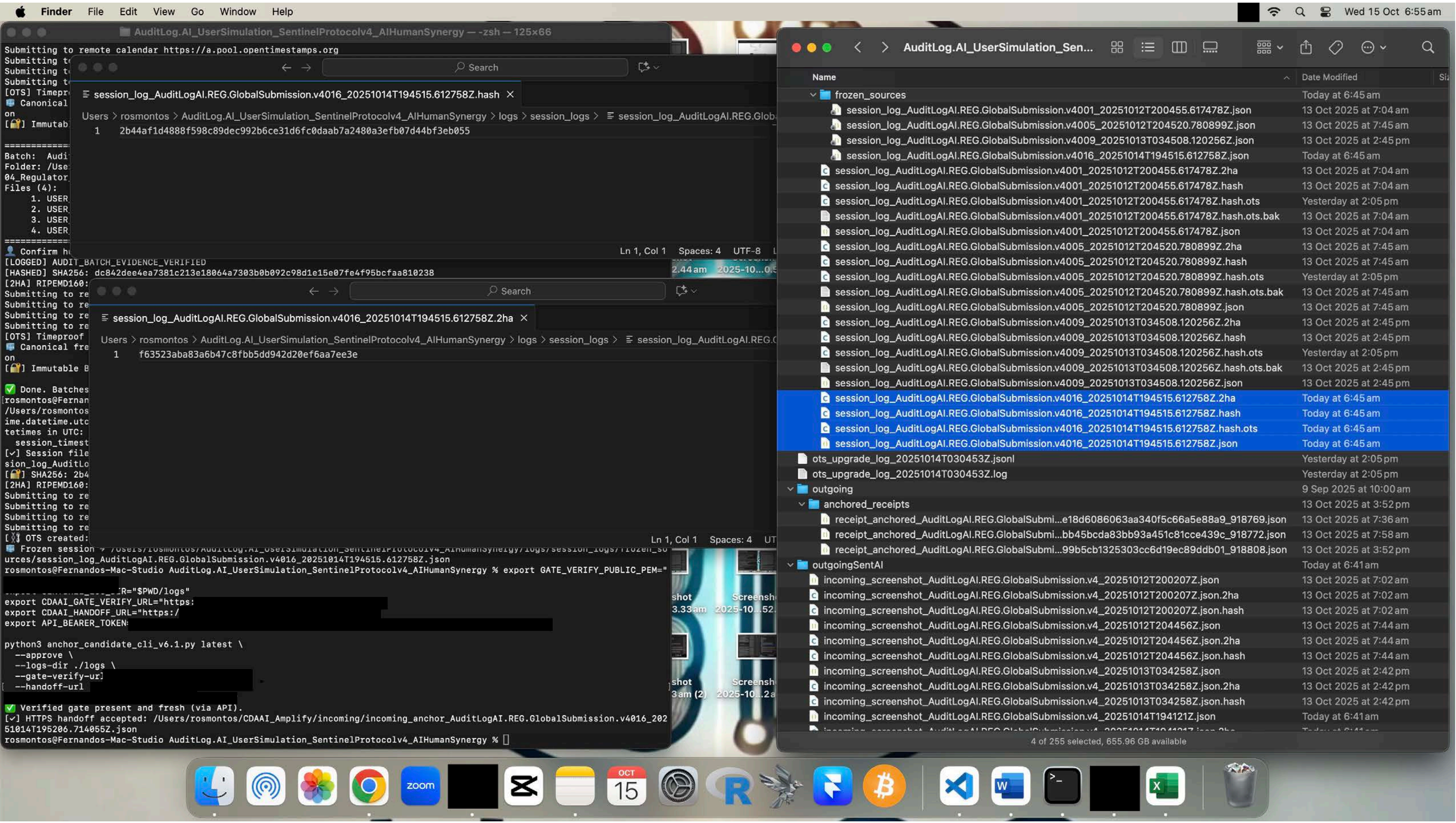
Description:

Screenshot demonstrating parity between session log digests and anchored OP_RETURN payload components (RIPEMD-160, sha8). Confirms that on-chain values match the exact bytes of the frozen session log JSON. Proves deterministic linkage between audit session, including all audit logs and each evidence file, to public blockchain record.

File Reference: `Screenshot 2025-10-15 at 6.55.33 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `9ae00b9b0442b02f06446a3d3d7432bfe26bf52d661e35af-b214862caa2e77cd` **RIPEMD-160:** `3b529fa672f8614453b99d8c76241c0a0d28de01` **OTS File:** `Screenshot 2025-10-15 at 6.55.33 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `417cb4aba1ab799655c7e023db89f926d1472b43b092a87102c58e0e1b880c76` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

38. Session Log Digest Match Validation



39. Dual-Ledger Consistency Verification

Description:

Illustrates cross-comparison of dual ledgers, demonstrating atomic consistency between AMPLIFY and user (AuditLog.AI held copy) ledgers. This demonstrates no discrepancies in TXID, payload, or events, supporting reliable reproducibility. It provides evidence for regulatory inspection of ledger parity (parity check = equality on {event, txid, payload} across AMPLIFY and user ledgers).

File Reference: `Screenshot 2025-10-15 at 6.57.55 am (2)___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `0fec999b3fb5bba6434ab25f65d09e778db5eb63f946cd1f463225c64892455b` **RIPEMD-160:** `034abe9c5619119429 3f57471084e102564101a1` **OTS File:** `Screenshot 2025-10-15 at 6.57.55 am (2)___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `f7f338658c60c811fc2426b155da7e473a7aea71b1abfa2f3bbffd8acf9802a7` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

39. Dual-Ledger Consistency Verification

40. Pull Anchor Artifacts and Log Audit

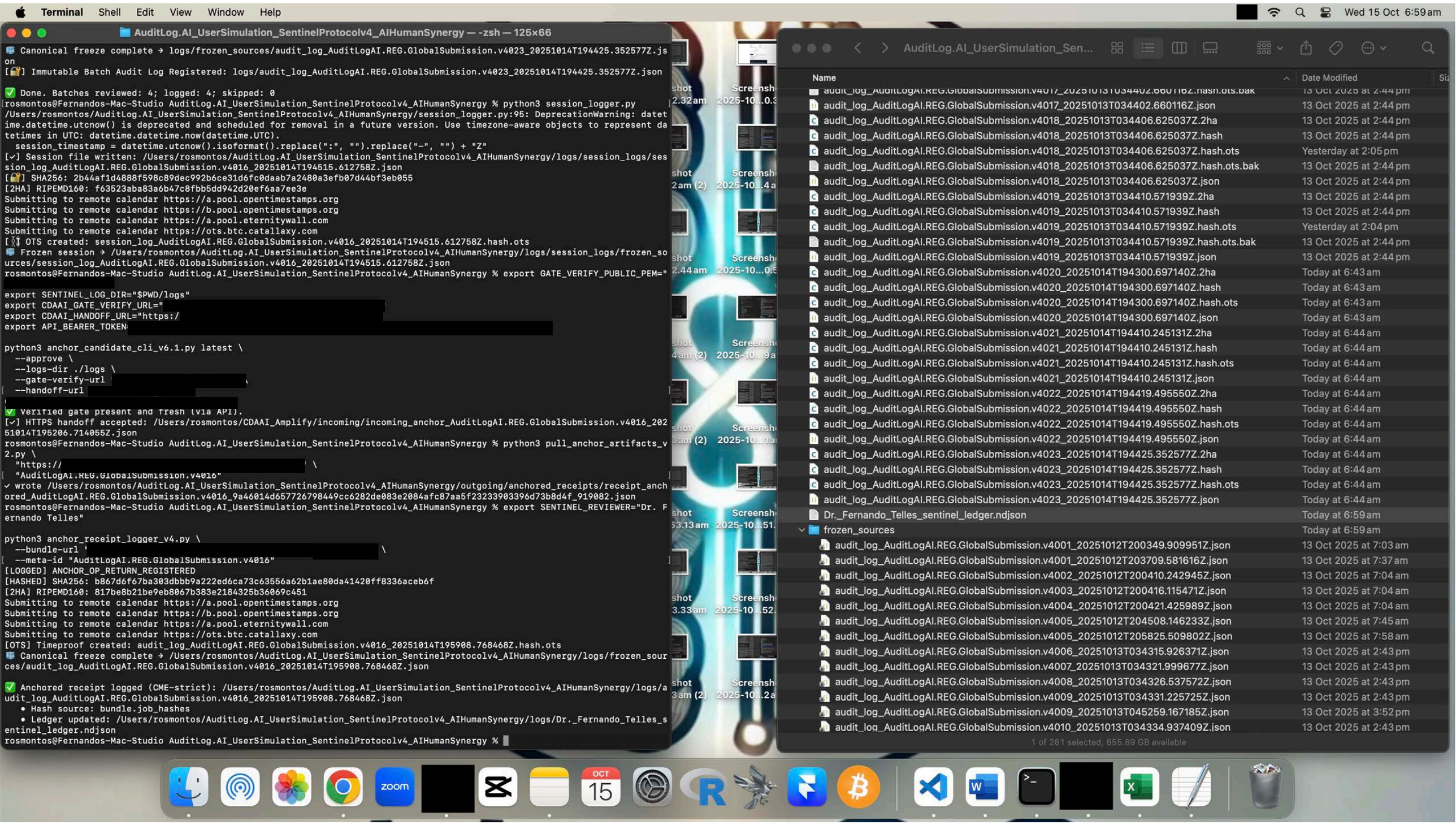
Description:

Depicts retrieval of anchored artifacts via `pull_anchor_artifacts_v2.py`, updating operator's local ledger against AuditLog.AI-held user ledger. This validates successful synchronization and integrity of receipts between user and AuditLog.AI ledgers.

File Reference: `Screenshot 2025-10-15 at 6.59.14 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `b799d19ed11bed-b4295c4eed41647fc1e85a0c20d732272b34606580a4d4d554` **RIPEMD-160:** `88cd-b8705614a357f14c44467e8f9669c9f324a6` **OTS File:** `Screenshot 2025-10-15 at 6.59.14 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `1f33316c86a23e0661d7e80f22f6028f0cfe4abc2241349024479a0397e124bb` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

40. Pull Anchor Artifacts and Log Audit



41. AuditLogger – Final Anchor Event

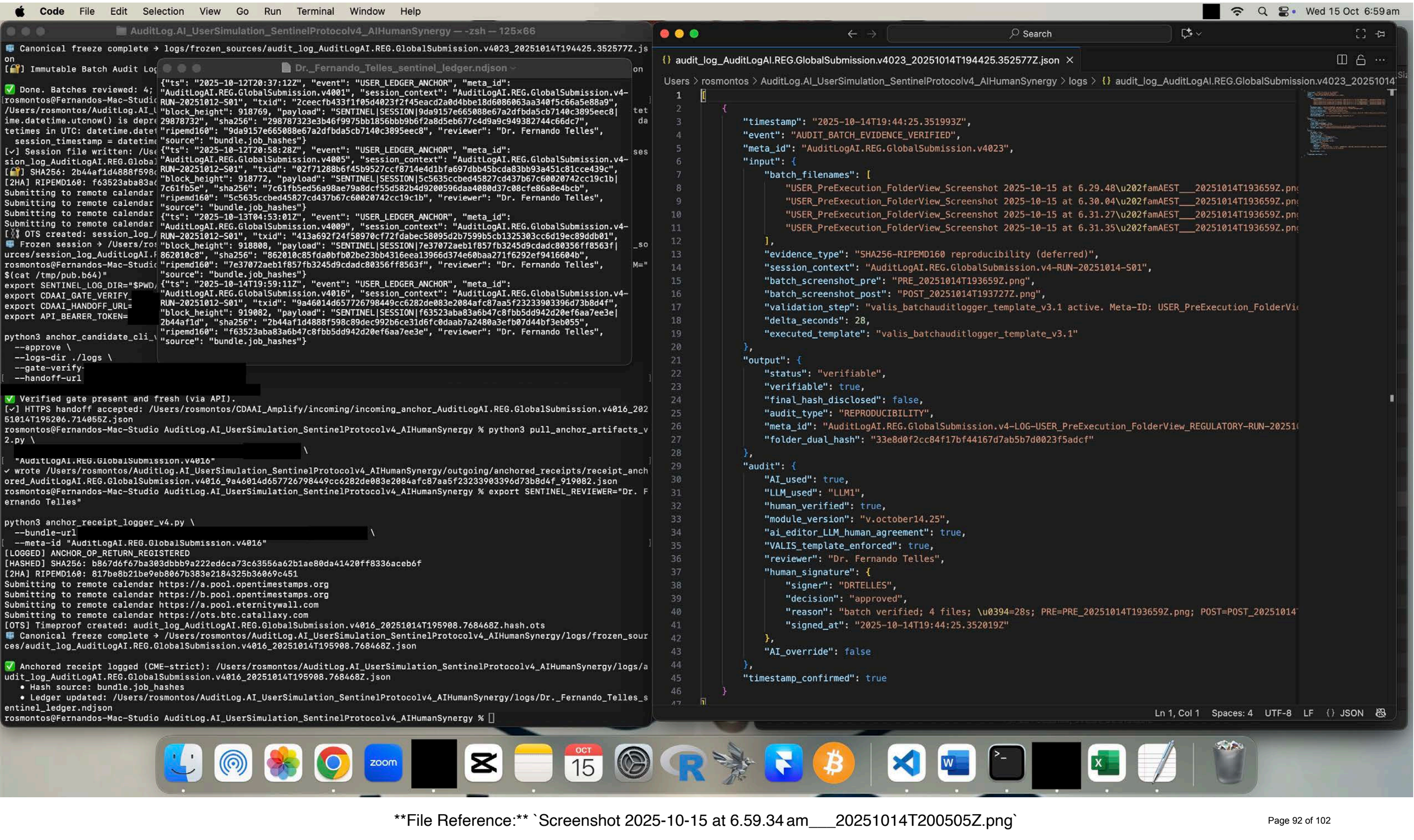
Description:

Captures mandatory audit log entry registering the anchor event, including linkage to TXID and block. This ensures complete audit trail closure with full traceability.

File Reference: `Screenshot 2025-10-15 at 6.59.34 am___20251014T200505Z.png` **Timestamp (UTC):** `20251014T200505Z` **SHA-256:** `42815c031071fade8f166dd9bfbc23d30f25f497762ad-f43c9e137585600894e` **RIPEMD-160:** `b2e2b690332662540a51e753cf1ef6aba09611f6` **OTS File:** `Screenshot 2025-10-15 at 6.59.34 am___20251014T200505Z.png.hash.ots` **OTS SHA-256:** `97e036c5f88b5c7c096835d7eb4c663e10d18f80c74d6e0503ba8b7c10a72816` **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084` **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18` **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

---

41. AuditLogger – Final Anchor Event



File Reference: `Screenshot 2025-10-15 at 6.59.34 am___20251014T200505Z.png`

42. AuditLog.AI Quality Management System (QMS) and Pre-Public Deployment Audit Log

Description:

The AuditLog.AI operating system was developed, tested and validated under Sentinel Protocol's QMS. The Foundational Reproducibility and Infrastructure Audit Record was published via dual-DOI and immutably inscripted to the Bitcoin mainnet via ordinal technology on July 30, 2025. Regulators are invited for independently verification, publication is available through:

Telles, F. (2025). Sentinel Protocol v3.1 – Infrastructure Pre-Public Deployment Audit Log. DOI: 10.13140/RG.2.2.29180.65924

Zenodo: 10.5281/zenodo.16607606 Ordinal: TXID: `ae198274a00abbb8296a3b9412e6fd3a62360bcf062e000fa2908d8f3b90e803`

Provenance Sessions: `SENTINFRA-SESS001_20250719T074009.072317Z` `SENTINFRA-SESS005_20250728T035324.098266Z`

Status: AUDIT PASS
Validator Agreement: 3/3
OP_RETURN TXID: `c4d2e53197983bf84f219b6b3cf4912fa5ebd3c030ae3debf7f35c1eef135e1c`
OP_RETURN Payload: `SENTINEL|SENTINFRA-SESS001|2ebc6ec29d3195f6d3b7050cacc75e145aaa1ad7` **Block:** `906434` **OP_RETURN TXID:** `ea2251eeaaf2f288e0f249d3c4c57346c934636aa2d37a7cac930537d1e3aaaa` **OP_RETURN Payload:** `SENTINEL|SENTINFRA-SESS005|ec6ae8259da7c55958b156c5b7998633e3db18dc` **Block:** `907511`

Independent System Validators:

Architect Validator: Dr. Fernando Telles BMedSc(Adv) MD(Dist) **Validator meta_id:** DR-TELLES-VAL
Audit: - TXID: `ccf9dc6dde2136b5e2adf035532233cfee255fb3704f8f5838b0de730f41eca0` - OP_RETURN payload: `DRTELLES-VAL001|PASS|SENTINFRA-SESS001|05cea9a984e521547de3a405b79fd93c769424fe` - Bitcoin Block: `906434`

Independent Medical Validator: Dr. Andrew Woo Bsc MD **Validator meta_id:** DRWOO-VAL
Audit: - TXID: `fd8f0406f4db2bf39d37a825e8dbd554028d72044813bbda864290fa4e88e0b6` - OP_RETURN payload: `DRWOO-VAL001|PASS|SENTINFRA-SESS001|a54419605f5cb9fa7ce39dd5716fc18f9d43f464` - Bitcoin Block: `907047`

Independent Engineering Validator: Benjamin Hookey BEng (Mechatronics & Robotics), FSEng (Safety Instrumented Systems)
Validator meta_id: ENGHOOKEY-VAL
Audit: - TXID: `a5622ac48c93017215a7fb7af6a69bb965220bb3a86e798e721d1f548033a5e3` - Pay-

load: `ENGHOOKEY-VAL001|PASS|SENTINFRA-SESS001|cc4652bfc2ffb00d81982ddff11a704204fe8c15`
- Bitcoin Block: `907041`

(Note: Legacy OP_RETURN schema (v3.x, pre-upgrade). Shown for historical provenance. Current v4 schema is SENTINELIII per C5.3.1.)

---

43. AuditLog.AI Live Runtime Execution Audit Infrastructure Integrity Provenance

Description:

The process of AuditLog.AI submission to global regulators (FDA/EMA/TGA) was recorded utilizing the software itself. As a result, immutable audit trails were generated recording the evolution of all submission documents with full transparency. The live execution screenshots included within this Dossier were audited to demonstrate 100% SHA-256 and RIPEMD-160 reproducibility. The original source document will be converted from markdown to PDF and submitted to regulators as a standalone Annex. Excerpts from this audit is provided below.

Excerpt 01: Audit Status PASS

```
 **Verification Method:** Human (Dr. Fernando Telles) Manual Re-execution + `dualhash-
er_public_sentinel_v1.1.py___20250809T030248.075732Z` + `ex-
tract_hashes.py___20250809T030248.072046Z` + `folder_dualhasher_v2.py` + LLM3 (An-
chor) + LLM2 (Validator)
**Verification Target:** Dual hash (`sha256`, `ripemd160`)
**Outcome:**
- SHA256 match on all files
- RIPEMD160 match on all files
- RIPEMD160 match on folders
- AUDIT_REPRODUCIBILITY_PASS

## **Human Verification Declaration:**

I, Dr. Fernando Telles, confirm that the original document for dual-Hashes System
Validation Screenshots was compiled manually by me. LLM3 Anchor audit was performed
on 17 October 2025 identifying 1 (one) human error:
```

⚠️ Screenshot #38 (`Screenshot 2025-10-15 at 6.55.33 am___20251014T200505Z.png`) missing RIPEMD-160 hash

```
 I verified the error identified by LLM3 as a true human error, and corrected the doc-
ument accordingly into its final version which was frozen as immutable pre-audit
evidence using AuditLog.AI
`Hashes_SystemValidationScreenshots___20251017T203710Z.md`.
Audit specifics:
- This audit was initiated only after freezing of pre-audit evidence
- This audit was performed manually by myself
- Completed Audit will now be dual-hashed, opentimestamped and bitcoin anchored using
AuditLog.AI as pre-audit evidence for submission to to LLM3 (Anchor) and LLM2 (Valid-
ator) in parallel for independent validation
- In order to independently verify the system, AuditLog.AI scripts (prevenance `ses-
sion_log_AuditLogAI.REG.GlobalSubmission.v4026_20251017T202205.834235Z.json`) were
NOT used for this audit. Instead, audit was performed using publicly available tools
we developed for independent validators (available through https://github.com/TELAI-
SYN/AI-Human-Synergy/tree/main/Sentinel_Protocol_v3.1/validator_tools).
```

Excerpt 02: Open Disclosure of Duplicate non-evidence screenshots

```
 Duplicate non-evidence screenshots detected during the hash integrity audit. Identical screenshot sets:
- set01: `Screenshot 2025-10-15 at 6.36.49 am (2)___20251014T200505Z.png` and `Screenshot 2025-10-15 at 6.37.46 am (2)___20251014T200505Z.png`
- set02: `Screenshot 2025-10-15 at 6.44.37 am (2)___20251014T200505Z.png` and `Screenshot 2025-10-15 at 6.44.02 am (2)___20251014T200505Z.png` and `Screenshot 2025-10-15 at 6.42.52 am (2)___20251014T200505Z.png`
- set03: `Screenshot 2025-10-15 at 6.37.00 am (2)___20251014T200505Z.png` an `Screenshot 2025-10-15 at 6.37.14 am (2)___20251014T200505Z.png`
These sets have been identified as byte-identical, producing the same SHA-256 and RIPEMD-160 digests.
Root cause: duplicate screenshots sets were automatically captured by macOS's full-screen capture function across a dual-monitor setup while the operator guidance runbook window was maximized on non-evidence secondary monitor view. Because no visual differences (e.g., clock, cursor, or secondary content) existed between the two frames, the resulting image files are identical at the binary level.

Regulatory relevance:
These image sets do not represent distinct evidence events. They are background captures from the operator's execution environment and are not part of the dossier evidence set submitted for regulatory review. Their inclusion in the manifest reflects full-disclosure of automated capture and batch hashing of all session artifacts, not intentional duplication of audit content.

Integrity handling:
Files remain hashed and timestamped within the session manifest for reproducibility. Their duplication has no material effect on validation integrity or cryptographic audit completeness. The event has been disclosed for full transparency as
`Non_evidence_Screenshots_identical_set01.pdf`,
`Non_evidence_Screenshots_identical_set02.pdf` and
`Non_evidence_Screenshots_identical_set03.pdf`.
```

Provenance Session_log:

`session_log_AuditLogAI.REG.GlobalSubmission.v4038_20251018T214519.997259Z.json`

Session Meta_id: `AuditLogAI.REG.GlobalSubmission.v4038` **Timestamp (UTC):** `20251018T214519.997259Z` **SHA-256:** `5af54564b05e7ae9c8e6b4c918ca19586ca3c7e58af6b72806-fa08603b27e844` **RIPEMD-160:** `fb5e9fa9e3873fe2236eaf1c60c4fc7dd0968c5a` **OTS File:** `session_log_AuditLogAI.REG.GlobalSubmission.v4038_20251018T214519.997259Z.hash.ots` **OTS SHA-256:** `40f3a9bf146ba3294bf682eeb260bfbe7eb3813c9f313660fcda89499c518b33` **OTS Bitcoin TXID:** `0fec86753224355204bf6a4e9ef0106302a2a597c116236279ab5ce44484f0b7` **Block:** `919723` **Merkle Root (from OTS proof):** `a460deac9d9a2027e72bf90a137786d5b29bb83b4d144868671032e-f724ff990` **Date of Existence (UTC):** `2025-10-18T22:40:46Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919723.)*

OP_RETURN Anchor

Transaction ID: `11fb255046a7934e5c978b3d0887318f85af3324b040873ce6026fea6c8af13d` **Payload:** `SENTINEL|SESSION|fb5e9fa9e3873fe2236eaf1c60c4fc7dd0968c5a|5af54564` **Block:** `919720`

Evidence File:

`AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_AuditLog.AI_v.oct18.25___20251018T214357Z.md`

Linked manifest: `manifest_markdown_to_pdf_conversions_REGULATORY-RUN-20251018-S02.csv`
SHA-256: `7966a393ec2798dd609074e99249cb61d7c9220ad3a7e128398f5ed4c7b9e051`
RIPEMD-160: `7f15cad8acc59548a6eb256f38196db6e8474c79` **OTS File:** `AUDIT_REPRODUCIBILITY_PASS_sha256_ripemd160_AuditLog.AI_v.oct18.25___20251018T214357Z.md.hash.ots` **OTS SHA-256:** `06343134dd0ff461af9d422c59de0a0c8607855aaa203cc2f87549afe54a6b3d` **OTS Bitcoin TXID:** `0fec86753224355204bf6a4e9ef0106302a2a597c116236279ab5ce44484f0b7` **Block:** `919723` **Merkle Root (from OTS proof):** `a460deac9d9a2027e72bf90a137786d5b29bb83b4d144868671032ef724ff990` **Date of Existence (UTC):** `2025-10-18T22:40:46Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919723.)*

Provenance

AuditLog.AI — Runtime Execution and System Validation Evidence

End-to-End Operational Proof During Regulatory Global Submission

---

Linked Runtime Scripts

User Node: (/frozen_files_Validated_AuditLog.AI_v.12sep.25_REGULATORY-RUN-20251017-S12/User_scripts) - `anchor_candidate_cli_v6.1___20251017T200502Z.py` - `anchor_receipt_logger_v4___20251017T200502Z.py` - `assert_verified_gate_job___20251017T200502Z.py` - `audit_verify_v1.2___20251017T200502Z.py` - `auditlogger___20251017T200502Z.py` - `batch_dualhasher_multi_v3.3___20251017T200502Z.py` - `batch_ots_stamper_v2___20251017T200502Z.py` - `batch_ots_upgrader_v2.1___20251017T200502Z.py` - `human_verifier___20251017T200502Z.py` - `pull_anchor_artifacts_v2___20251017T200502Z.py` - `screenshot_handoff_cli_v2___20251017T200502Z.py` - `session_logger___20251017T200502Z.py` - `validis_batchauditlogger_template_v3.1___20251017T200502Z.py` - `validis_batchnameverifier_v2___20251017T200502Z.py`

AuditLog.AI Node: (/frozen_files_Validated_AuditLog.AI_v.12sep.25_REGULATORY-RUN-20251017-S04/CDAAI_scripts) - `anchor_api_v2___20251017T200502Z.py` - `confirm_watcher_v2.3___20251017T200502Z.py` - `gate_ingest_api_v2___20251017T200502Z.py` - `opreturnanchor_amplify_v5.1___20251017T200502Z.py`

Shared - `compliance_enforcer___20251017T200502Z.py` - `gate_manager___20251017T200502Z.py` - `hashvalidatorblock___20251017T200502Z.py`

Interface: (/frozen_files_Validated_AuditLog.AI_v.12sep.25_REGULATORY-RUN-20251017-) - Vercel UI: AuditLog.AI e-Signature Gate

---

Provenance Session_log:

`session_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T200606.551974Z.json`

- **Session Meta_id:** `AuditLogAI.REG.GlobalSubmission.v4020`
- **Timestamp (UTC):** `20251014T200606.551974Z`
- **SHA-256:** `5c009fb3e680ca41e0e17aefe3bc931ec00b195e8faa0eb3f546d059930b0724`
- **RIPEMD-160:** `041cf98418a6ae39dbcc5d3e49e60a06a5112993`

- **OTS File:** `session_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T200606.551974Z.hash.ots`
- **OTS SHA-256:** `c26576aaa708bdb17285f011d1d9c7880a3e5f8f334d2a64ebc2511bf2f12c2a`
- **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084`
- **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18`
- **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

OP_RETURN Anchor

Transaction ID: `d096197e02a8f1ac9ca7da43d353f109fe4ff72f6ae7f88b39c83877247e9ef7` **Payload:** `SENTINEL|SESSION|041cf98418a6ae39dbcc5d3e49e60a06a5112993|5c009fb3` **Block:** `919084`

Audit_log:

`audit_log_AuditLogAI.REG.GlobalSubmission.v4025_20251014T200556.273909Z.json`

- **Linked Dossier Evidence:** `3-20, 22-23, 25, 28, 34, 36-38, 40-41`
- **Linked manifest:** `manifest_User_REGULATORY-RUN-20251014-S02.csv`
- **SHA-256:** `1c9f6e2cc1ff9ce1e615cca36083a3a558468edccefcdd19b86dca3ff082897c`
- **RIPEMD-160:** `56ee6f1e48405a6ca6ef5fda34e842525c06f199`
- **OTS File:** `audit_log_AuditLogAI.REG.GlobalSubmission.v4025_20251014T200556.273909Z.hash.ots`
- **OTS SHA-256:** `1323e7ab0aa18f63d8ffa051cfee88121941aaa658184deb60e2d8b79ba79a3d`
- **OTS Bitcoin TXID:** `5c28773f653c441226df3079bf1b37c662f6c3643a544840e43098cd2a75502f` **Block:** `919084`
- **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcdc13fc828d2e855c985f18`
- **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

Audit_log:

`audit_log_AuditLogAI.REG.GlobalSubmission.v4024_20251014T200552.935806Z.json`

- **Linked Dossier Evidence:** `24, 26-27, 29-33, 35, 39`
- **Linked manifest:** `manifest_CDAAI_REGULATORY-RUN-20251014-S01.csv`
- **SHA-256:** `62a3b9a89b4a812027fabf29492f4cefe408cc6e78db5ab07f0f9cd9eef16770`
- **RIPEMD-160:** `d0c075e79ef8d55d3b6a4eb2c9a753c209c7e96e`

- **OTS File:** `audit_log_AuditLogAI.REG.GlobalSubmission.v4024_20251014T200552.935806Z.hash.ots`
- **OTS SHA-256:** `5c28773f653c441226df3079b-f1b37c662f6c3643a544840e43098cd2a75502f`
- **OTS Bitcoin TXID:** `c355bc2b7d88d70ffd5bd6745cfdc40aa348cd8b3b34dfb7adec-c853c9486ddb` **Block:** `919084`
- **Merkle Root (from OTS proof):** `51d14a6b89b363557180530c51fa192a1cdd093edcd-c13fc828d2e855c985f18`
- **Date of Existence (UTC):** `2025-10-14T20:08:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919084.)*

Provenance Session_log:

`session_log_AuditLogAI.REG.GlobalSubmission.v4016_20251014T194515.612758Z.json`

- **Session Meta_id:** `AuditLogAI.REG.GlobalSubmission.v4016`
- **Timestamp (UTC):** `20251014T194515.612758Z`
- **SHA-256:** `2b44af1d4888f598c89dec992b6ce31d6fc0daab7a2480a3efb07d44bf3eb055`
- **RIPEMD-160:** `f63523aba83a6b47c8fbb5dd942d20ef6aa7ee3e`
- **OTS File:** `session_log_AuditLogAI.REG.GlobalSubmission.v4016_20251014T194515.612758Z.hash.ots`
- **OTS SHA-256:** `c6df3b2bf6a8b3e13c40f2139f61b2d-d7440b8e2a10a321c86b891b3626c2c0b`
- **OTS Bitcoin TXID:** `10e650706acadd74115a38042becea8c-cbe58eac7146a9d1c9193ed027ca8cb7` **Block:** `919083`
- **Merkle Root (from OTS proof):** `416d0f316dec37bdbccdfb142775154e704a203ed41c94dcf10765dce6f7a4fa`
- **Date of Existence (UTC):** `2025-10-14T19:56:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919083.)*

OP_RETURN Anchor

Transaction ID: `9a46014d657726798449cc6282de083e2084afc87aa5f23233903396d73b8d4f` **Payload:** `SENTINEL|SESSION|f63523aba83a6b47c8fbb5dd942d20ef6aa7ee3e|2b44af1d` **Block:** `919082`

Audit_log:

`audit_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T194300.697140Z.json`

- **Linked Dossier Evidence:** `2`
- **Linked manifest:** `manifest_CDAAI_PreExecution_FolderView_REGULATORY-RUN-20251014-S01.csv`

- **SHA-256:** `22b4c889342ce49df6919cfe2031e514f8dcd4c919bc01f5de46a1d11f3d4e59`
- **RIPEMD-160:** `cffc255cfe7d5441891a576f21fb99ddbc8c75b2`
- **OTS File:** `audit_log_AuditLogAI.REG.GlobalSubmission.v4020_20251014T194300.697140Z.hash.ots`
- **OTS SHA-256:** `b38543beb8fdc0b362e137fe0240cf8ef76c3a421f6d3406b215c392862a68fc`
- **OTS Bitcoin TXID:** `10e650706acadd74115a38042becea8ccbe58eac7146a9d1c9193ed027ca8cb7` **Block:** `919083`
- **Merkle Root (from OTS proof):** `416d0f316dec37bdbccdfb142775154e704a203ed41c94dcf10765dce6f7a4fa`
- **Date of Existence (UTC):** `2025-10-14T19:56:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919083.)*

Audit_log:

`audit_log_AuditLogAI.REG.GlobalSubmission.v4023_20251014T194425.352577Z.json`

- **Linked Dossier Evidence:** `1`
- **Linked manifest:** `manifest_USER_PreExecution_FolderView_REGULATORY-RUN-20251014-S04.csv`
- **SHA-256:** `dc842dee4ea7381c213e18064a7303b0b092c98d1e15e07fe4f95bcfaa810238`
- **RIPEMD-160:** `76a1ae67e00ba9f604a23dc65e4e7b66dc557099`
- **OTS File:** `audit_log_AuditLogAI.REG.GlobalSubmission.v4023_20251014T194425.352577Z.hash.ots`
- **OTS SHA-256:** `67a307877305241369ba4e69e4f9e495554652bcc188249d29441f2fd7883f4d`
- **OTS Bitcoin TXID:** `10e650706acadd74115a38042becea8ccbe58eac7146a9d1c9193ed027ca8cb7` **Block:** `919083`
- **Merkle Root (from OTS proof):** `416d0f316dec37bdbccdfb142775154e704a203ed41c94dcf10765dce6f7a4fa`
- **Date of Existence (UTC):** `2025-10-14T19:56:06Z` *(TXID broadcast time confirming file existence as of Bitcoin block 919083.)*

Provenance Session_log:

`session_log_AuditLogAI.REG.GlobalSubmission.v4009_20251013T034508.120256Z.json`

- **Session Meta_id:** `AuditLogAI.REG.GlobalSubmission.v4009`
- **Timestamp (UTC):** `20251013T034508.120256Z`
- **SHA-256:** `862010c85fda0bfb02be23bb4316eea13966d374e60baa271f6292ef9416604b`
- **RIPEMD-160:** `7e37072aeb1f857fb3245d9cdadc80356ff8563f`

- **OTS File:** `session_log_AuditLogAI.REG.GlobalSubmission.v4009_20251013T034508.120256Z.hash.ots`
- **OTS SHA-256:** `aa327f72e2799c98a4f1ea5ff83cda1fc2cf4b9670d90072d865f9cfa53d6bfd`
- **OTS Bitcoin TXID:** `eabcee5e6fffeb6fc6b3c7807165ae863b383fb8459aa0d311726026302ef0e9` **Block:** `918813`
- **Merkle Root (from OTS proof):** `f161f7b058a1156af142f50a5dc6952b97d2dc897d4e912d6b68dd8119704744`
- **Date of Existence (UTC):** `2025-10-12T17:07:56Z` *(TXID broadcast time confirming file existence as of Bitcoin block 918813.)*

OP_RETURN Anchor

Transaction ID: `413a692f24f58970cf72fdabec58095d2b7599b5cb1325303cc6d19ec89ddb01` **Payload:** `SENTINEL|SESSION|7e37072aeb1f857fb3245d9cdadc80356ff8563f|862010c8` **Block:** `918808`

Audit_log:

`audit_log_AuditLogAI.REG.GlobalSubmission.v4007_20251013T034321.999677Z.json`

- **Linked Dossier Evidence:** `21`
- **Linked manifest:** `manifest_AuditLog.AI_Human_gate_REGULATORY-RUN-20251013-S02.csv`
- **SHA-256:** `e5b5aa84b5a8c910bf69bc14c0030c21bd42fa84efaacff955f7a395ac86207d`
- **RIPEMD-160:** `8376fb437f11e1d2bc2eb147363b7f1a88205d5f`
- **OTS File:** `audit_log_AuditLogAI.REG.GlobalSubmission.v4007_20251013T034321.999677Z.hash.ots`
- **OTS SHA-256:** `c6a9707074f416ee292249f1dbb26a55be5a1c2dd676bc40ae7ba64bfebd8abb`
- **OTS Bitcoin TXID:** `eabcee5e6fffeb6fc6b3c7807165ae863b383fb8459aa0d311726026302ef0e9` **Block:** `918813`
- **Merkle Root (from OTS proof):** `f161f7b058a1156af142f50a5dc6952b97d2dc897d4e912d6b68dd8119704744`
- **Date of Existence (UTC):** `2025-10-12T17:07:56Z` *(TXID broadcast time confirming file existence as of Bitcoin block 918813.)*