Synthetic evidence. Real cryptographic checks.

A clean-room technical explanation for prospective users and design partners.

Purpose. This browser-based reference demonstrates Capture → Seal → Investigate → Prove. It is a prospect-discovery and technical-inspection asset, not a production integration or an independently audited LoopGrid release.

01 / WHAT IS SIMULATED

Four fictional workflows cover a voice refund, mortgage income-file update, insurance reserve adjustment and enterprise role grant. A deterministic script stands in for an agent; no LLM or voice provider is called. Identities, authority grants, policies and reviewers are fictional. A browser-memory adapter simulates downstream state changes or failures. No real customer system is contacted.

02 / WHAT ACTUALLY RUNS

Each new run creates a non-extractable Ed25519 browser private key. The recorder hashes protected event content with SHA-256, creates workspace/decision-scoped links, signs each event and signs a checkpoint binding the event count and final hash. The browser independently recomputes commitments and verifies signatures against a separately supplied signer fingerprint. The published fixture was signed offline; its private key is not distributed.

03 / PORTABLE, INDEPENDENTLY CHECKED

Reference exports include the manifest, JSONL events, signed checkpoint, public key and complete JSON bundle. The separate Python verifier supports only the documented synthetic format, requires an external trust pin, uses the established cryptography package and makes no network calls. No compatibility with production LoopGrid SDKs or exports is asserted.

04 / WHAT VERIFICATION CHECKS

Content integrity, sequence and chain continuity, event signatures, workspace/decision identity, signer identity and the signed checkpoint. Field changes, missing or truncated events, and wrong keys are rejected. Re-signing with another key can produce mathematically valid signatures but fails against the original trusted fingerprint. A bundle cannot establish its own signer trust.

05 / WHAT IT DOES NOT ESTABLISH

A valid bundle establishes integrity and provenance of captured records under the expected key. It does not prove upstream truth, capture completeness, genuine human approval, business correctness or legal compliance. A signed mock response is not evidence that a real bank, insurer, payment processor or EHR acted. Browser timestamps are not trusted timestamps; demo keys and policy logic are not hardened production controls.

06 / RELATIONSHIP TO LOOPGRID v0.8

Existing project materials describe a Design Partner baseline with REST/SDK integration, signed evidence, privacy modes and offline verification. This deliverable does not independently validate that backend, production signing boundary or export protocol. A real pilot should connect one authorized workflow through trusted server-side capture, record authoritative downstream responses, and validate actual product exports, privacy and key management before using live data.

REPRODUCE THE INDEPENDENT CHECK

```
python -m pip install cryptography
python verify_synthetic.py evidence.zip \
--pin-file trusted-pin.txt
```

Install the dependency before going offline. Verification exits 0 on success and 1 on failure. Obtain the expected signer fingerprint independently. Exact canonicalization and signature rules are in PROTOCOL.md.