Forensic Ledger: Nine-Plane Tamper-Evident Operational State Architecture

Jeremy Blaine Thompson Beebe
Bxthre3 Inc. — bxthre3inc@gmail.com — ORCID: 0009-0009-2394-9714

April 2026

Abstract

Every autonomous system generates records of its reasoning, decisions, and physical actions. In most systems, those records are generated by the same software that performed the actions — making them alterable by the system itself. The Forensic Ledger solves this with a 9-plane orthogonal architecture: Purpose/Bounds/Fact crossed by Past/Present/Future-Predicted. No plane can write to any other plane. All 9 planes are linked through a SHA-256 cryptographic chain anchored at manufacturing. The result is an audit trail that is structurally impossible to forge or retroactively modify, deployed and verified in the Agentic platform.

Keywords: forensic ledger, 9-plane architecture, tamper-evident logging, plane isolation, SHA-256 chain, autonomous audit, BX3 Framework

1 Introduction

The canonical failure mode of autonomous system audit trails: the system that generates records also controls the database that stores them. A compromised reasoning engine can backdate logs, insert fabricated events, and delete unfavorable records — while presenting a perfectly consistent audit trail to any auditor. The Forensic Ledger addresses this by making the audit trail architecturally tamper-evident rather than administratively controlled.

2 The 9-Plane Matrix

	Purpose	**Bounds Engine**	**Fact**
Past	P1: Mandate	P4: Reason Log	P7: Outcome Record
Present	P2: Intent	P5: Decision	P8: Execution
Future-Predicted	P3: Plan	P6: Projection	P9: Projection Confirmation

P1 records the Human Root that authorized each action. P4 records every reasoning step. P7 records every physical event. P9 confirms the Sandbox Gate validated each actuation. Each plane is written by exactly one actor — and no actor writes to any plane outside its designation.

3 The Cryptographic Chain

Every ledger entry contains SHA-256 of the previous entry. Modifying any historical entry breaks the chain visibly: recomputing the chain from the hardware-anchored genesis event detects any tampering in O(n) time. The genesis event is sealed at Hub manufacturing and cannot be altered without physical hardware compromise.

4 Plane Isolation Enforcement

The Bounds Engine cannot write to P1 (Purpose) or P7 (Fact). The Fact Layer cannot modify P4 (Reason Log). The Purpose Layer cannot fabricate P7 (Outcome Record). These are architectural separations enforced by the BX3 Loop Isolation — not software permissions. A compromised Bounds Engine cannot rewrite the audit trail because it has no write access to the planes that would make revision useful.

5 Application: Water Court

For regulatory proceedings, the 9-plane record provides the complete causal chain: what was authorized (P2), what was proposed (P5), what was validated (P9), and what occurred (P7). The system is its own witness with cryptographic proof of integrity.

6 Conclusion

The Forensic Ledger makes audit trail forgery structurally impossible. Every claim about what happened is backed by a cryptographically chained, plane-isolated record that no software update, administrative action, or system compromise can alter without detection.