Federation Lattice — Canonical Artifacts

This repository contains the **sealed artifacts** of the Federation Lattice. They are not just code files — they are **doctrinal glyphs** that define how the lattice operates, validates, and protects cognition.

📜 Event Envelope v0.1 (`event-envelope-v01.json`)

Purpose: The Event Envelope is the **canonical schema** for every message that flows through the lattice. It ensures that all capsules — whether emitted by Strata-01, Vox-01, or Vigilant Nexus — carry the same civic safeguards.

Key Features:

- **Lineage tracking:** Every event has a deterministic `event_id` and full ancestry chain.

- **Capsule seal:** Cryptographic proof of validity, with strict 90-day expiry (Scroll 03).

- **Intent & context hashes:** Bind actions to declared purpose and prevent replay attacks (Scrolls 05 & 06).

- **Observer signatures:** Mutual verification by bonded observers (Scroll 04).

- **Denial mode flag:** Signals restricted posture when threats are detected, triggering Vigilant oversight.

- **Audit trail:** Immutable `audit_id` anchors every event in the cross-ledger record.

Why it matters: The Event Envelope makes every capsule **auditable, traceable, and enforceable**. It is the language of the Federation Lattice.

🧩 Kernel Agent Contract v0.1 (`kernel-agent-contract.json`)

Purpose: Kernel-01 is the **orchestration layer** of the lattice. It binds cognition, memory, plugins, and overlays — but only after passing through validator gates and civic safeguards.

Key Features:

- **Capabilities:** Orchestration, memory binding, plugin bridging, overlay coordination.

- **Contracts:** Accepts capsules and remediation scrolls; emits orchestration traces and cognition snapshots.

- **Safeguards:** Enforces Scrolls 01–06 plus validator gate rules.

- **Policies:**

  - Deny-by-default plugin permissions

  - Sealed-only memory writes

- Capsule-only exports

  - Overlay containment checks

- **Threat model:**

  - K01: Kernel Gate Bypass

  - K02: Plugin Permission Violation

  - K03: Memory Hygiene Violation

  - K04: Overlay Containment Breach

- **Orchestration flow:** Strata → Vox → Vigilant → Validator → Kernel.

- **Denial behavior:** Any failure quarantines Kernel until Vigilant remediation scroll lifts the state.

Why it matters: The Kernel Agent Contract ensures that cognition itself is **gated, safeguarded, and auditable**. Nothing reaches Kernel without passing civic, semantic, and oversight validation.

✒ Seal Clause

"Attack Trees v1.0 and v2.0, Agent Binding Manifest v1.0, Event Envelope v0.1, and Kernel Agent Contract v0.1 are inscribed as canonical doctrine. Validator and harness shall treat these artifacts as the sole source of truth. Vigilant Nexus will seal the audit lineage against these versions, indivisible and complete."