GOPEL Post-Quantum Cryptographic Agility Amendment v1.2

Amendment to: HAIA-RECCLIN Agent Architecture Specification, Appendix A; GOPEL Proof of Concept Specification v0.6.1

Author: Basil C. Puglisi, MPA

Date: March 2026

Status: Draft for Tier 0 human arbiter final signature

Scope: This amendment adds post-quantum cryptographic readiness to the GOPEL and HAIA-RECCLIN audit trail specifications. It does not alter the seven deterministic operations, the non-cognitive constraint, or any existing governance architecture. It extends the cryptographic foundation to remain valid as the computing threat environment evolves.

Version history:

- v1.0: Initial draft. Submitted to five-platform CAIPR review (Gemini, Kimi, DeepSeek, Grok, ChatGPT).
- v1.1: Incorporated all mandatory and recommended modifications from first-pass CAIPR convergence analysis. Three mandatory modifications (converged across 3+ platforms) and ten recommended modifications integrated.
- v1.2: Incorporated all remaining findings from second-pass CAIPR verification review. Ten items resolved: composite signature storage consolidated, no-key-reuse rule added, Anchoring Option 3 tightened, arithmetic correction applied, latency benchmark reframed, TSA availability noted, canonicalization registry referenced, HSM override clause expanded, anchoring interval justified, hash algorithm metadata addressed.

---

1. Problem Statement

The current specification (Appendix A.3) recommends ECDSA P-256 or Ed25519 for digital signing of audit records. Both algorithms rely on elliptic curve mathematics vulnerable to Shor's algorithm running on a cryptographically relevant quantum computer. The hash chain itself (SHA-256, Appendix A.2) retains strong security under known quantum attacks; Grover's algorithm provides roughly a square-root speedup against preimage attacks, reducing the effective brute-force margin from 256-bit to approximately 128-bit. That 128-bit margin remains computationally prohibitive for currently projected quantum computing capabilities. Collision resistance faces a different quantum profile (the BHT algorithm gives approximately 2^n/3 complexity), but for audit trail integrity, preimage resistance is the relevant property.

The vulnerability is specific: digital signatures on audit records and human identity bindings could be forged retroactively by an adversary with future quantum capability. For long-lived signed records, this risk is archival signature forgery and loss of authenticity assurance. An attacker collecting signed audit records today could produce fraudulent signatures after quantum capability matures. NIST distinguishes this from the classic "Harvest Now, Decrypt Later" confidentiality threat (NIST IR 8547). The risk to governance audit trails is not decryption of encrypted content; it is future forgery of authentication credentials on historical records.

The hash chain sequence integrity is not at risk. The signing authenticity is.

---

2. NIST Post-Quantum Standards Reference

NIST finalized three post-quantum cryptographic standards on August 13, 2024:

- **FIPS 203 (ML-KEM):** Module-Lattice-Based Key Encapsulation Mechanism, formerly CRYSTALS-Kyber. For key exchange and encapsulation. Relevant to future encrypted transport between GOPEL and platform APIs.
- **FIPS 204 (ML-DSA):** Module-Lattice-Based Digital Signature Algorithm, formerly CRYSTALS-Dilithium. For digital signatures. Directly relevant to audit record signing (Appendix A.3) and human identity binding (Appendix A.4). Supports hedged signing (default) and deterministic signing modes, with an optional context string parameter.
- **FIPS 205 (SLH-DSA):** Stateless Hash-Based Digital Signature Algorithm, formerly SPHINCS+. An alternative signature algorithm based on hash functions rather than lattice mathematics. Provides a resilience fallback if lattice-based schemes face future cryptanalytic challenges. SLH-DSA carries substantially larger signature sizes than ML-DSA (up to 49KB depending on parameter set), making it a resilience option with real storage and performance cost, not a neutral substitute.

All three are Tier 1 evidence: published NIST federal standards with completed evaluation and public comment periods.

Transition timeline reference: NIST draft guidance (IR 8547, SP 800-131A Rev. 3 draft) indicates a trajectory toward deprecating 112-bit classical digital signatures after 2030 and moving classical digital signatures toward disallowance after 2035. These milestones inform the cutover triggers specified in Section 3.

---

3. Amendment to Appendix A.3 (Digital Signing)

3.1 Signature Tier Classification

Each audit record or batch of records must be digitally signed using the deploying organization's signing key. The specification supports three signature tiers, selected based on the deploying organization's threat model, retention horizon, and regulatory requirements:

Tier A (Classical, Current Minimum): RSA-2048 minimum key length. ECDSA P-256 or Ed25519 recommended. Acceptable for deployments where the quantum threat timeline exceeds the retention period of the signed records.

Tier B (Hybrid, Recommended for New Deployments): Non-separable composite signature scheme producing a single encapsulated signature structure containing both a classical component (Ed25519 or ECDSA P-256) and a post-quantum component (ML-DSA, FIPS 204), bound through explicit domain separation. The composite construction must follow the non-separable profile specified in IETF draft-ietf-lamps-pq-composite-sigs, where a domain separator binds both component signatures to the composite algorithm identifier. The resulting composite signature is stored as a single unified blob under the composite algorithm identifier; component signatures are not stored in separate fields. This prevents stripping attacks where an adversary substitutes one component after the other becomes forgeable. Verification requires the unified composite signature to pass against the composite algorithm identifier. This is the recommended configuration for deployments beginning in 2026 or later.

Engineering context for Tier B: The hybrid construction is a migration hedge, not a permanent guarantee of full integrity assurance. The IETF composite draft describes weak non-separability (WNS), which prevents simple component stripping because the domain separator is embedded in the signed message. If the classical component becomes forgeable under quantum attack, the composite binding and domain separation prevent simple signature substitution, but the construction's long-term strength rests on the post-quantum component. Organizations should plan migration to Tier C when regulatory guidance and operational maturity support it.

Tier C (Post-Quantum Primary): ML-DSA (FIPS 204) as the sole signing algorithm. SLH-DSA (FIPS 205) available as a resilience fallback when hash-based signatures are preferred over lattice-based, with the explicit understanding that SLH-DSA carries substantially larger signatures and slower verification (see Section 7.1). Appropriate when regulatory guidance or organizational risk assessment mandates post-quantum primary protection.

3.2 ML-DSA Parameter Set Selection

This specification recommends ML-DSA-65 (NIST Security Category 3, approximately AES-192 equivalent) as the default post-quantum parameter set for Tier B and Tier C deployments. The rationale:

- **ML-DSA-44** (Category 2): Smaller signatures (~2,420 bytes) and faster operations. Closer parity with classical algorithms. Appropriate for deployments with shorter retention horizons (under 5 years) where storage efficiency is prioritized.
- **ML-DSA-65** (Category 3): Moderate signatures (~3,309 bytes). Conservative choice for governance audit trails with retention periods of 5 to 25 years. Provides margin against future cryptanalytic improvements in lattice attacks without the overhead of Category 5.
- **ML-DSA-87** (Category 5): Largest signatures (~4,627 bytes). Maximum security margin. Appropriate for critical infrastructure or national security applications where records must remain authentic for decades.

Deploying organizations may select ML-DSA-44 or ML-DSA-87 based on documented risk assessment. The selection must be recorded in the deployment's QMS records with retention horizon justification.

3.3 Signing Mode

All ML-DSA signing operations under this specification must use the **deterministic** signing mode defined in FIPS 204. The hedged mode (FIPS 204 default) introduces per-signature randomness that, while offering side-channel resistance, creates implementation-dependent variation. For governance audit trails where byte-exact reproducibility of the signing input is a verification requirement, deterministic mode eliminates a class of implementation divergence.

If a deploying organization's threat model requires hedged signing for side-channel resistance (e.g., signing operations on exposed hardware) or if HSM hardware constraints prevent configurable mode selection, the organization must document the mode selection in the QMS records and accept that verification tooling must account for the nondeterministic component. Early-market HSMs may not support deterministic mode configuration; this override ensures procurement viability during the transition period without silently deviating from the specification.

3.4 Key Management

Signing keys must be stored in a hardware security module (HSM) or equivalent tamper-resistant storage. Key rotation must occur at minimum annually or upon personnel change in the arbiter role. The certificate chain must be documented and available for auditor verification. Key rotation events are themselves audit records in the chain.

Dual-key management for Tier B: The classical private key and the post-quantum private key must be stored under separate access controls within the HSM. Compromise of one key must not automatically grant access to the other. If the HSM supports partitioned key storage, each key should reside in a separate partition. If partitioned storage is unavailable, the keys must be derived from independent key generation processes, not from a shared seed. This reduces correlated compromise risk.

No-key-reuse rule: Component keys used in Tier B composite signatures must not be reused for standalone Ed25519, ECDSA, or ML-DSA signing operations outside the GOPEL

governance context. The IETF composite signature draft requires this prohibition to prevent cross-protocol vulnerabilities where a standalone signature created with a component key could be leveraged to attack the composite construction. Violation of this rule must be flagged as a compliance finding in the deployment's QMS audit.

3.5 Certificate Chain Quantum Readiness

The certificate authority (CA) certificates in the verification chain must be signed at the same tier or higher as the audit records they certify. A Tier B audit signature verified against a classical-only CA certificate creates a weakest-link vulnerability: the audit signature is quantum-resistant, but the trust anchor is not.

Deployments operating at Tier B must either obtain hybrid-signed CA certificates or operate under a trust model where the CA's classical certificate is time-bounded and the organization maintains a documented migration path to a post-quantum CA. The CA migration status must be recorded in the deployment's QMS records.

3.6 Migration Path

Existing deployments operating at Tier A should plan migration to Tier B when HSM firmware and organizational PKI infrastructure support ML-DSA key generation and storage. Migration does not require re-signing historical records. The hash chain protects historical sequence integrity and tamper linkage. However, individual historical record signatures remain classically signed and carry a residual archival forgery risk once classical algorithms become breakable (see Section 8.1 and 8.2 for mitigation).

Migration produces a Key Migration Record in the audit chain documenting the algorithm change, the date, the authorizing arbiter, and the rationale.

3.7 Cutover Triggers

The following triggers replace the general "when NIST issues deprecation guidance" language from v1.0, referencing specific NIST transition milestones (IR 8547, SP 800-131A Rev. 3 draft):

Trigger 1 (2026, immediate): All new GOPEL deployments beginning after the publication date of this amendment should operate at Tier B minimum.

Trigger 2 (2030, NIST milestone): When NIST finalizes deprecation of 112-bit classical digital signatures (projected after 2030 per IR 8547 draft trajectory), all existing Tier A deployments must present a documented migration plan with timeline within 90 days. Note: RSA-2048 aligns with the 112-bit deprecation milestone. The specification's recommended Tier A algorithms (Ed25519, ECDSA P-256) are generally treated as 128-bit or higher classical strength and fall under the 2035 disallowance horizon. Trigger 2 functions as a governance planning checkpoint for all Tier A deployments regardless of specific algorithm, ensuring migration planning begins well before the 2035 hard deadline.

Trigger 3 (2035, NIST milestone): When NIST moves classical digital signatures toward disallowance (projected after 2035 per IR 8547 draft trajectory), all active deployments must operate at Tier B minimum. Tier A is no longer acceptable for new records.

These triggers are based on draft NIST guidance as of March 2026 and must be rechecked against finalized NIST publications as they are released. The trigger dates are planning horizons, not guarantees of the NIST timeline.

---

4. Amendment to Appendix A.4 (Human Identity Binding)

The digital certificate or equivalent mechanism binding human arbiter identity to Arbitration and Decision Records must use the same signature tier as the audit trail signing configuration. If the audit trail operates at Tier B (hybrid), the identity binding must also carry both classical and post-quantum signatures under the composite profile.

Identity binding certificates issued under classical-only PKI remain valid for the retention period of the records they signed. However, long-term validation of those records requires trusted timestamping evidence establishing that the signature was created while the classical algorithm was still considered secure (see NIST guidance on signature validity and timestamping assurance, FIPS 204). New identity bindings issued after migration to Tier B must carry composite signatures.

---

5. Amendment to Appendix A.5 (Mandatory Record Metadata)

The following fields are added to the mandatory metadata for every audit record:

- **signature_algorithm:** Identifier specifying the signing algorithm used for this record (e.g., “Ed25519”, “ML-DSA-65-det”, “COMPOSITE-Ed25519+ML-DSA-65-det”). Required for all records. Enables verification software to select the correct verification path without external configuration. The “-det” suffix indicates deterministic signing mode.
- **signature:** The signature bytes for this record. For Tier A, this contains the classical signature. For Tier B, this contains the unified composite signature blob encapsulating both classical and post-quantum components as a single non-separable structure per the IETF composite profile. For Tier C, this contains the post-quantum signature. Component signatures are never stored in separate fields. The composite blob is opaque to the audit record schema; only the verification module parses its internal structure using the algorithm identifier.
- **public_key_id:** Identifier or reference to the public key or certificate used for signing. Enables verification without out-of-band key discovery. For Tier B, this field references the composite certificate containing both the classical and post-quantum public keys.

- **canonicalization_version:** Version identifier for the canonicalization algorithm applied before hashing and signing (e.g., "GOPEL-CANON-1.0"). Two implementations using the same canonicalization version on the same logical record must produce identical byte sequences. This field ensures that verification across different GOPEL implementations or future software versions can reproduce the exact bytes that were signed, regardless of internal representation differences. Canonicalization algorithm definitions are maintained in the GOPEL specification repository with semantic versioning. Implementations must publish their canonicalization logic for independent audit.
- **hash_algorithm:** Identifier specifying the hash algorithm used for this record's hash chain entry (e.g., "SHA-256", "SHA3-256"). Required for all records. Appendix A.2 specifies SHA-256 as the minimum and SHA-3 as recommended for new deployments. This field makes the algorithm explicit per-record, ensuring cross-implementation verification in environments where deployments may operate different hash algorithms or where a deployment migrates from SHA-256 to SHA-3 over its lifecycle.

Note on v1.1 to v1.2 change: The `signature_pq` field specified in v1.1 is removed. Tier B composite signatures are stored as a single unified blob in the `signature` field. This consolidation aligns the storage schema with the non-separable composite profile mandated in Section 3.1 and eliminates the risk of application-layer stripping vulnerabilities that arise from storing signature components in separate fields.

---

6. Amendment to GOPEL PoC Security Layer

6.1 Current State

The GOPEL PoC (v0.6.1) uses HMAC-SHA256 for operator identity verification, anti-impersonation sentinel heartbeats, and second-approver evidence gate signatures.

HMAC-SHA256 is a symmetric construction. It is not vulnerable to Shor's algorithm. Known generic quantum attacks (Grover's) reduce the effective brute-force margin of the 256-bit key to approximately 128-bit; this does not catastrophically break the construction but reduces the security margin. The reduced margin remains computationally prohibitive for currently projected quantum computing capabilities. No change is required to HMAC-based operations in the current PoC.

6.2 Recommended PoC Extension (v0.7 Target)

The PoC does not currently implement asymmetric digital signing of audit records (that is specified at the architectural level for production deployments, not the reference implementation). When the PoC is extended to include asymmetric audit record signing, the deployment-time configuration default should be Tier B (hybrid). "Default" here means the configuration that ships as the preset value in the deployment configuration file, not a

runtime adaptive selection. Operators may change the tier at deployment time; the system does not change it based on content evaluation.

Add a configurable signing module that supports:

1. Ed25519 (Tier A, classical)
2. ML-DSA-65-det (Tier C, post-quantum primary, deterministic mode)
3. Composite Ed25519 + ML-DSA-65-det with domain separation, producing a single unified signature blob (Tier B, deployment-time default)

The module selection is a deployment configuration, not a runtime decision. GOPEL does not choose which algorithm to use based on content evaluation. The algorithm tier is set at deployment and applies uniformly to all records. This preserves the non-cognitive constraint.

---

7. Key Encapsulation (Future Transport Layer)

ML-KEM (FIPS 203) applies to encrypted transport between GOPEL and AI platform APIs. The current specification delegates transport security to TLS on all API calls. IETF standardization of ML-KEM within TLS 1.3 is in progress (draft-ietf-tls-mlkem) but not finalized. When IETF publishes a final RFC incorporating ML-KEM into TLS 1.3, GOPEL deployments should require ML-KEM-capable TLS for platform connections carrying governed data.

This is a deployment infrastructure requirement, not a GOPEL architecture change. GOPEL does not negotiate TLS parameters. It requires that the transport layer meet specified minimums. The minimum specification should be updated when ML-KEM-capable TLS becomes available from major cloud providers and is backed by a published RFC.

Status: Tier 3 (proposed for future development, contingent on IETF final RFC publication). No action required in current version.

7.1 Storage and Performance Considerations

Deployments adopting Tier B (Hybrid) or Tier C (Post-Quantum Primary) must account for the significant increase in cryptographic material size inherent in post-quantum algorithms. This has direct implications for storage capacity, network bandwidth, and system performance.

7.1.1 Signature and Key Size Comparison

Algorithm	Signature Size	Public Key Size	Security Category
Ed25519 (classical)	64 bytes	32 bytes	~128-bit classical
ML-DSA-44	~2,420 bytes	~1,312 bytes	Category 2

Algorithm	Signature Size	Public Key Size	Security Category
ML-DSA-65 (recommended)	~3,309 bytes	~1,952 bytes	Category 3
ML-DSA-87	~4,627 bytes	~2,592 bytes	Category 5
SLH-DSA (resilience fallback)	up to ~49,856 bytes	up to ~64 bytes	Varies by parameter set

A Tier B deployment using Composite Ed25519 + ML-DSA-65 produces a unified composite signature blob of approximately 3,373 bytes (64-byte classical component + 3,309-byte post-quantum component, encapsulated with composite overhead). Public key references are carried in the `public_key_id` metadata field and do not contribute to per-record signature storage.

7.1.2 Storage Scaling

Using the existing storage estimate from the HAIA-RECCLIN Agent Architecture (Section 7.2), a manuscript production run generating approximately 2,500 transactions at Tier A produces roughly 200 MB of audit trail. The same production run at Tier B would add approximately 8.4 MB of additional signature data (2,500 records x ~3,373 bytes). This is a 4.2% increase in total storage. At enterprise scale with higher transaction volumes, the percentage remains small relative to response record storage (which dominates at ~40 KB per transaction).

Storage scaling is not a blocking concern for most deployments but must be projected over the full retention period and verified against the chosen storage backend.

7.1.3 Performance Impact

- **Signing latency:** ML-DSA-65 signing is computationally more intensive than Ed25519. Benchmark testing is required before deployment to confirm that the combined composite signing operation remains within acceptable latency tolerances.
- **Verification latency:** ML-DSA-65 verification is slower than Ed25519. Automated audit verification workflows must be tested for throughput at projected record volumes.
- **Network payload:** Transmitting larger signatures between GOPEL instances and verification APIs increases bandwidth consumption. For deployments with latency-sensitive checkpoint gates, payload size should be measured against network capacity.

7.1.4 Required Capacity Planning

Before deploying Tier B or Tier C, organizations must conduct and document capacity planning covering:

1. **Storage projection:** Total audit trail size over the retention period, factoring in per-record signature size increase. Verify that the audit storage backend scales without performance degradation.
2. **Latency benchmark:** Establish baseline record write latency at Tier A. Measure added latency from Tier B composite signing during performance testing. Benchmark: measure end-to-end record write latency against a 50ms threshold. This is a local performance objective for governance audit trails, not a standards-derived requirement. Deployments exceeding this benchmark must document the accepted latency and its operational impact.
3. **Verification throughput:** If automated verification is performed, confirm the verification system sustains required throughput at the higher per-record computational cost.

Capacity planning results must be documented in the deployment's QMS records before Tier B or Tier C activation.

---

8. What This Amendment Does Not Change

- The seven deterministic operations remain unchanged.
- The non-cognitive constraint remains unchanged. Algorithm selection is a deployment configuration, not a content evaluation.
- SHA-256 hash chaining remains the minimum hash algorithm. SHA-3 remains recommended for new deployments. Both retain strong security margins under known quantum attacks; neither is broken by Shor's algorithm, and known generic quantum attacks reduce their brute-force margin rather than catastrophically compromising the construction.
- HMAC-SHA256 operations in the PoC remain unchanged.
- The six audit record types remain unchanged.
- The Bridge Record mechanism for GDPR erasure remains unchanged.
- No historical records require re-signing or re-hashing.

8.1 Historical Record Residual Risk

The hash chain protects sequence integrity and tamper linkage for all records, including those signed under Tier A classical algorithms. Modification of any historical record invalidates all subsequent hashes, making insertion or alteration of individual records detectable.

However, the hash chain does not preserve signer authenticity for historical records once classical algorithms become breakable. An adversary with quantum capability could forge a classical signature on a fabricated record. The forged signature would pass classical verification, but the record could not be inserted into the chain without breaking the hash sequence.

The compound threat is an adversary who compromises a historical classical signing key and rewrites the chain from a point of alteration forward, regenerating all subsequent hashes. This produces a valid but fraudulent alternative history. The internal hash chain alone cannot distinguish the authentic chain from the rewritten one without an external reference point.

This is an explicit residual risk for Tier A historical records. It is acceptable under the following conditions:

1. The deployment transitions to Tier B for all new records immediately.
2. External hash chain anchoring is implemented per Section 8.2.
3. Verification tooling is updated to flag Tier A records as "historical, classical-only, quantum-vulnerable" when the signature_algorithm field indicates a classical-only algorithm.

Organizations retaining Tier A records beyond 2035 (the projected NIST disallowance horizon for classical digital signatures) should evaluate whether re-signing a subset of high-value historical records under Tier B or Tier C is justified by the records' governance significance.

8.2 External Hash Chain Anchoring for Long-Term Integrity

To close the historical chain-rewrite vulnerability identified in Section 8.1, the state of the hash chain must be periodically anchored to an external, quantum-resistant source of truth. This is accomplished by publishing the current chain tip hash (the hash of the most recent audit record) to a medium that an adversary cannot retroactively alter.

Required anchoring mechanism: Deployments must implement one of the following:

1. **Quantum-Resistant Trusted Timestamping Authority (TSA):** Submitting the chain tip hash to a TSA that signs it with a post-quantum algorithm (ML-DSA or SLH-DSA) and returns a timestamp token. The token must be stored and preserved as a critical audit artifact. This is the recommended mechanism for most enterprise deployments. Note: PQ-native TSAs are an emerging capability as of March 2026. During the transition period, organizations may operate their own TSA or use hybrid TSAs that sign with both classical and post-quantum algorithms until the commercial TSA market matures.

2. **Quantum-Resistant Public Ledger:** Publishing the chain tip hash to a blockchain or distributed ledger that itself uses post-quantum signatures.

3. **Timestamped Regularized Public Publication:** Publishing the chain tip hash in a widely witnessed medium with strong data integrity. To qualify as a valid anchoring mechanism under this specification, Option 3 must meet the following minimum admissibility criteria: the publication medium must provide append-only retention (published entries cannot be removed or altered), independent witnessability (third

parties can independently verify that a specific hash was published at a specific time), cryptographic or institutional timestamping (the publication carries a timestamp from an independent source, such as a notary, a government register, or a publication with verifiable issuance dates), and externally retrievable proof (the published entry can be retrieved and verified by an auditor who was not present at the time of publication). Publications that do not meet all four criteria do not qualify as valid anchoring events under this specification.

Anchoring frequency: The frequency of anchoring is a deployment decision based on risk assessment and record value. The maximum acceptable interval between anchoring events is seven days for deployments handling high-risk AI governance records (EU AI Act high-risk classification). This interval represents the deploying organization's accepted rollback exposure window: in a chain-rewrite attack, the adversary could forge at most seven days of history before the next external anchor provides an independent verification point. Organizations with lower rollback tolerance should anchor more frequently. Lower-risk deployments may anchor monthly with documented risk acceptance.

Anchoring as audit event: Each anchoring event produces an Anchoring Record in the audit chain, documenting the chain tip hash published, the external medium used, the timestamp token or publication reference received, and the authorizing operator. The Anchoring Record is itself signed under the current deployment tier and included in the hash chain.

PQ notarization checkpoint at migration: When a deployment migrates from Tier A to Tier B, a mandatory anchoring event must occur at the point of migration. This PQ-signed notarization checkpoint commits the pre-migration chain state under a post-quantum signature, creating a cryptographic boundary between the classical and post-quantum eras of the audit trail. This avoids per-record re-signing while anchoring the entire prior history under a quantum-resistant commitment.

8.3 Verification Tooling for Historical Records

Verification software processing GOPEL audit trails must handle records signed under different tiers as the audit trail spans a migration boundary. The following behavior is required:

- **Tier B and Tier C records:** Full composite or post-quantum verification. The unified composite signature must pass against the composite algorithm identifier (Tier B) or the post-quantum signature must pass (Tier C).
- **Tier A records (pre-migration):** Classical verification only. The verification tool must display a visible indicator that the record carries classical-only signatures and is subject to the residual archival forgery risk described in Section 8.1. Required indicator: "[CLASSICAL-ONLY: Quantum-vulnerable signature. Verify against nearest external anchor.]"

- **Anchoring Records:** The tool must verify the external anchoring evidence (TSA token, ledger reference, or qualified publication reference) and confirm that the chain tip hash at the anchoring event matches the internal chain state.

Verification tooling requirements are operational, not architectural. They do not add cognitive operations to GOPEL. They apply to the audit verification workflow that operates on GOPEL's outputs.

---

9. Evidence Tier Classification

- NIST FIPS 203, 204, 205 standards: **Tier 1** (proven by others, published federal standards).
- NIST IR 8547 transition guidance and SP 800-131A Rev. 3 draft: **Tier 1** (published NIST guidance, noting that transition timelines are draft and subject to revision).
- Archival signature forgery threat model: **Tier 1** (documented by NIST, NSA, ENISA).
- Hybrid composite signature scheme with domain separation per IETF draft: **Tier 2** (working concept, based on IETF draft-ietf-lamps-pq-composite-sigs; standard cryptographic composition, not yet adversarially tested within GOPEL).
- External hash chain anchoring: **Tier 2** (standard timestamping practice applied to GOPEL chain architecture; not yet operationally tested within GOPEL).
- ML-KEM transport integration: **Tier 3** (proposed, dependent on IETF final RFC publication).

---

10. Decision Point for Human Arbiter

Approve, modify, or reject the following:

1. Adopt the three-tier signature classification (Tier A/B/C) with non-separable composite profile storing a single unified signature blob for Tier B (Sections 3.1, 5).
2. Set Tier B as the recommended deployment-time default for new deployments beginning March 2026 (Section 3.7, Trigger 1).
3. Adopt ML-DSA-65 in deterministic mode as the recommended post-quantum parameter set, with documented rationale, organizational override for HSM constraints, and no-key-reuse rule (Sections 3.2, 3.3, 3.4).
4. Add signature_algorithm, unified signature field, public_key_id, canonicalization_version, and hash_algorithm to mandatory record metadata (Section 5).
5. Add dual-key management requirements, no-key-reuse prohibition, and CA certificate chain quantum readiness (Sections 3.4, 3.5).
6. Add storage and performance impact section with capacity planning requirements and 50ms latency benchmark (Section 7.1).
7. Name historical classical signatures as explicit residual risk with defined acceptability conditions (Section 8.1).

8. Add external hash chain anchoring with tightened Option 3 admissibility criteria, TSA availability note, justified anchoring interval, and mandatory PQ notarization checkpoint at migration (Section 8.2).
9. Add verification tooling requirements for historical Tier A records (Section 8.3).
10. Adopt NIST milestone-based cutover triggers with Trigger 2 clarification for 128-bit algorithms (Section 3.7).
11. Classify PoC signing module extension as a v0.7 development target (Section 6.2).
12. Classify ML-KEM transport integration as Tier 3 contingent on IETF final RFC (Section 7).

Recommendation: Approve all twelve items. This version resolves every finding from both CAIPR review passes without altering any core architectural property. No platform in either review round rejected the amendment. All dissents are either resolved or explicitly named as bounded residual risk with documented mitigation.

---

11. CAIPR Review Record

First-Pass Review (v1.0 to v1.1)

Platform	Disposition	Key Contributions
Gemini	Approve with modifications	Flagged historical chain-rewrite vulnerability and storage impact omission
Kimi	Approve with modifications	Flagged CA chain weakest-link, dual-key management, verification tooling gap, "computationally infeasible" language
DeepSeek	Conditional approval	Endorsed Gemini's two mandatory modifications, drafted implementation language
Grok	Conditional approval	Drafted complete v1.1 language for anchoring and storage sections
ChatGPT	Approve with modifications	Deepest technical review: HNDL terminology correction, non-separable composite profile, ML-DSA-65 justification, signing mode specification, canonicalization version, NIST milestone triggers

Second-Pass Review (v1.1 to v1.2)

Platform	Disposition	Key Contributions
ChatGPT	Approve with minor modifications	No-key-reuse rule, 3,373-byte arithmetic correction, Anchoring Option 3 tightening, hash_algorithm metadata, composite profile refinement
DeepSeek	Approve, no remaining issues	Confirmed all original dissents resolved, no new gaps
Kimi	Approve, three minor	TSA availability note, 50ms reframed as benchmark, canonicalization registry reference

Platform	Disposition	Key Contributions
	enhancements	
Grok	Approve, no remaining issues	Confirmed all original dissents resolved, recommended immediate publication
Gemini	Approve with three amendments	Composite signature storage consolidation (strongest architectural finding), Anchoring Option 3 cryptographic timestamping, HSM deterministic mode override

All ten platforms across both rounds confirmed the architectural amendment is sound. No platform in either round rejected the amendment.

Document version: v1.2 Draft

Requires: Tier 0 human arbiter final signature before publication.