ENGINEERING STANDARDS LIBRARY / DATA, STATE, AND EVIDENCE

Tamper-Evident Hash-Chained Ledgers & Evidence Bundle Standard

The architecture of audit and compliance recording has failed.

PERMANENT DOCUMENT ID

MYTHOS-DAT-0005

PUBLICATION

Candidate Standard
Version 1.0.0-candidate

ASSURANCE PROFILE

Standard / Advanced
High Assurance

PERMANENT PUBLICATION IDENTITY

Tamper-Evident Hash-Chained Ledgers & Evidence Bundle Standard

DOCUMENT ID
MYTHOS-DAT-0005

VERSION
1.0.0-candidate

STATUS
Candidate Standard

PUBLISHER
Mythos Systems

PUBLICATION DATE
2026-07-24

SCHEDULED REVIEW
2027-07-24

12	6	4	1
ATOMIC CRITERIA	ASSURANCE VIEWS	IMPLEMENTATION PROFILES	PERMANENT IDENTITY

Publication doctrine. This standard is a permanent library identity. Interpretation must preserve its recorded evidence, controlled exceptions, formal revision history, and explicit relationship to companion standards.

DOMAIN CONTROL SYSTEM

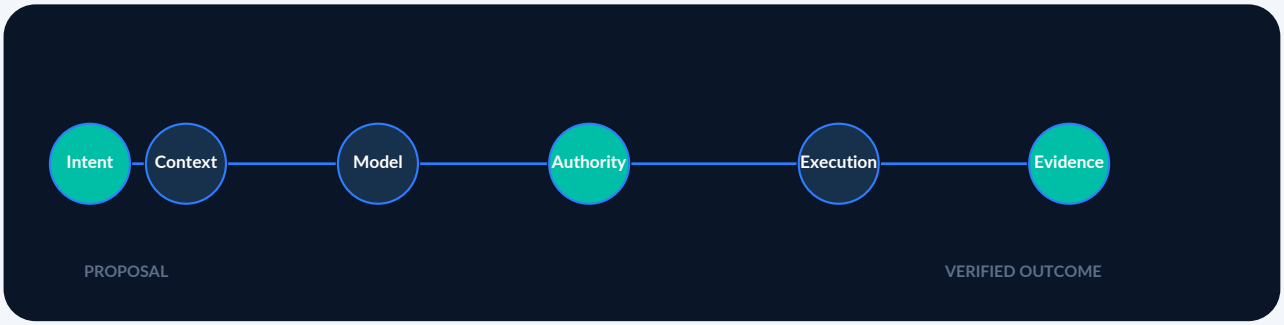
MYTHOS-DAT-0005 inside the data, state, and evidence constellation

Connected assurance	Architecture, implementation, operations, evidence, and lifecycle are evaluated as one controlled system.
Specialization rule	Companion standards may add precision, but cannot silently weaken this publication.

Executive orientation

Defines tamper-evident ledgers and evidence bundles for consequential system actions, including hash chaining, signatures, provenance, retention, verification, and disclosure boundaries.

WHY THIS STANDARD EXISTS	The architecture of audit and compliance recording has failed.
OPERATIONAL SCOPE	This standard covers the evaluation of: - Hash-chained event logs and append-only ledgers (BLAKE3, SHA-256) - Cryptographic signing of log entries and checkpoints (Ed25519) - Transparency logs and public verifiability (Sigstore, Rekor) - Evidence Bundle generation for AI agents and automated systems - in-toto attestations and artifact provenance binding - Merkle tree architectures for high-volume log aggregation - Air-gapped and sovereign audit trail architectures
PRIMARY AUDIENCE	- Security engineers and architects designing SIEM and audit platforms - Platform engineers building compliance pipelines - AI engineers building agent evidence and governance systems - GRC (Governance, Risk, Compliance) and audit teams - Standards bodies seeking a reference tamper-evidence methodology



Assurance principle. Model capability, data quality, identity, authority, operational evidence, and human accountability are treated as a connected system. A high aggregate score cannot compensate for a failed mandatory safety or authority boundary.

Contents

Evaluation criterion index

This standard contains 12 atomic criteria. Each criterion is independently testable and requires retained evidence.

ID	EVALUATION CRITERION
4.1.1	Chaining Mechanism
4.2.1	Action Signing
4.3.1	Contextual Proof
4.4.1	External Proof of Existence
4.5.1	Artifact Provenance
4.6.1	Survivability
5.1.1	MEIS-Tamper
5.1.2	MEIS-Bundle
6.1.1	Retroactive Log Tampering (Cover-Up)
6.1.2	Split-Viewing Attack
6.1.3	Non-Repudiation Failure (Impersonation)
6.1.4	Evidence Disconnect

MYTHOS-DAT-0005-4.1.1

4.1.1 Chaining Mechanism

Normative source requirement

Is the log mathematically resistant to retroactive tampering?

Score	Criteria
0	Standard database table (UPDATE/DELETE permitted)
1	Append-only table, but no cryptographic linking
2	Each entry includes a hash of the previous entry
3	Forward-linked hash chain using BLAKE3, with periodic Merkle root checkpoints
4	Merkle tree architecture enabling efficient O(log N) inclusion proofs for individual entries
5	Perfect integrity: formally verified hash-chains, zero mutation capability at the storage layer, cryptographic proof of global log consistency

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records. Context manifests, retrieval traces, source citations, freshness records, conflict decisions, and memory provenance.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-4.2.1

4.2.1 Action Signing

Normative source requirement

Can an action be cryptographically attributed to a specific identity?

Score	Criteria
0	Logs rely on application-level `user_id` strings
1	Logs signed by a shared symmetric key
2	Logs signed by an asymmetric key (Ed25519), but keys are long-lived and shared
3	Per-entity signing keys (e.g., per-agent or per-service Ed25519 keys)
4	Short-lived, scoped signing keys backed by hardware (TPM/HSM) or OIDC federation
5	Perfect non-repudiation: formally verified key provenance, cryptographic attestation of signer identity, zero repudiation possible

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records. Policy configuration, identity or trust records, authorization decisions, revocation evidence, and adversarial test results.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-4.3.1

4.3.1 Contextual Proof

Normative source requirement

Can the system export a self-contained, verifiable proof of a consequential action?

Score	Criteria
0	Only a flat log line (e.g., "Agent X deleted file Y") is available
1	Log line includes a reference to an external trace ID
2	Exportable bundle includes the trace, but requires external systems to resolve context
3	Evidence Bundle includes the action, policy decision, user identity, and tool execution result
4	Bundle includes the full LLM prompt, context window, and model inference that triggered the action
5	Perfect bundle: cryptographically signed, self-contained package (using CBOR/COSE), verifiable offline by a third party with zero access to internal systems

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records. Policy configuration, identity or trust records, authorization decisions, revocation evidence, and adversarial test results.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-4.4.1

4.4.1 External Proof of Existence

Normative source requirement

Can a third party verify that a record existed at a specific time without seeing the record?

Score	Criteria
0	No external time-stamping; relies on local server clocks
1	NTP synchronized clocks, but no cryptographic time-stamping
2	Periodic hash checkpoints exported to an external system (e.g., S3 bucket)
3	Integration with a public transparency log (Sigstore Rekor) for checkpoint Merkle roots
4	Per-entry inclusion proofs verifiable against a public log
5	Perfect transparency: formally verified inclusion proofs, zero trust in local clocks, cryptographic resistance to split-viewing attacks

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-4.5.1

4.5.1 Artifact Provenance

Normative source requirement

Is lifecycle metadata cryptographically bound to the artifact it describes?

Score	Criteria
0	Metadata stored in a separate database, linked by an ID
1	Metadata includes the artifact's hash, but is not signed
2	Signed metadata, but no standardized schema
3	Standard in-toto attestations bound to artifact hashes
4	Attestations stored in a transparency log and verified at runtime before artifact use
5	Perfect provenance: formally verified attestation chains, zero ability to detach metadata from artifacts, automated policy enforcement on attestation presence

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-4.6.1

4.6.1 Survivability

Normative source requirement

Does the audit trail survive a catastrophic system or infrastructure compromise?

Score	Criteria
0	Audit logs stored on the same infrastructure as the production system
1	Logs streamed to a separate logging cluster in the same cloud/account
2	Logs replicated to a separate cloud provider or on-prem environment
3	WORM (Write-Once-Read-Many) storage architecture for log retention
4	Cryptographically sealed historical archives with automated integrity verification
5	Perfect survivability: air-gapped, sovereign log retention, formally verified continuous integrity scanning, zero ability to destroy historical evidence

---

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Reproducible benchmark results, workload definitions, raw outputs, and variance records. Context manifests, retrieval traces, source citations, freshness records, conflict decisions, and memory provenance.
ASSESSMENT METHOD	MATURITY SIGNAL
Run a version-pinned workload with representative inputs, record distributions rather than a single score, repeat under failure and load, and compare reproduced results with the stated acceptance threshold.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-5.1.1

5.1.1 MEIS-Tamper

Normative source requirement

• Retroactive Modification: 100+ tests attempting to use privileged credentials to `UPDATE` or `DELETE` log entries, verifying the system detects the hash-chain break.

• Split-View Attack: 50+ tests attempting to present a fake, locally-generated log to a verifier, verifying the transparency log (Rekor) rejects the inclusion proof.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Architecture records, implemented configuration, test evidence, operational telemetry, exception decisions, and accountable approval.
ASSESSMENT METHOD	MATURITY SIGNAL
Inspect the architecture and configured control, execute normal and adverse scenarios, verify measurable outcomes, sample retained evidence, and confirm that exceptions are time-bound and approved.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-5.1.2

5.1.2 MEIS-Bundle

Normative source requirement

• Offline Verification: 100+ tests exporting an Evidence Bundle for a complex agent action and attempting to verify it on a disconnected, air-gapped machine.

• Context Fidelity: 50+ tests verifying the bundle contains the exact prompt, policy decision, and tool output that led to the action.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Context manifests, retrieval traces, source citations, freshness records, conflict decisions, and memory provenance. Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
ASSESSMENT METHOD	MATURITY SIGNAL
Inject stale, conflicting, low-authority, and malicious information; inspect selection and citation behavior; verify scope isolation, correction, deletion, and provenance retention.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-6.1.1

6.1.1 Retroactive Log Tampering (Cover-Up)

Normative source requirement

Severity: Critical Description: A compromised admin or attacker attempts to delete or modify logs that record their malicious actions.

Required Controls: 1. Forward-linked BLAKE3 hash chains. 2. Append-only, WORM storage at the infrastructure level. 3. Periodic Merkle root checkpoints to an external transparency log.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Context manifests, retrieval traces, source citations, freshness records, conflict decisions, and memory provenance.
ASSESSMENT METHOD	MATURITY SIGNAL
Inspect the architecture and configured control, execute normal and adverse scenarios, verify measurable outcomes, sample retained evidence, and confirm that exceptions are time-bound and approved.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-6.1.2

6.1.2 Split-Viewing Attack

Normative source requirement

Severity: High Description: A compromised system presents a valid, but fake, local audit log to an internal verifier, while hiding the true (malicious) actions in a separate log.

Required Controls: 1. Public transparency logs (Rekor) for all critical checkpoints. 2. Verifiers must check inclusion proofs against the public log, not just the local system.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Architecture records, implemented configuration, test evidence, operational telemetry, exception decisions, and accountable approval.
ASSESSMENT METHOD	MATURITY SIGNAL
Inspect the architecture and configured control, execute normal and adverse scenarios, verify measurable outcomes, sample retained evidence, and confirm that exceptions are time-bound and approved.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-6.1.3

6.1.3 Non-Repudiation Failure (Impersonation)

Normative source requirement

Severity: High Description: An attacker uses a shared, compromised symmetric key to forge log entries, framing another user or service.

Required Controls: 1. Per-identity asymmetric signing keys (Ed25519). 2. Short-lived, federated signing keys via OIDC/SPIFFE.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Policy configuration, identity or trust records, authorization decisions, revocation evidence, and adversarial test results. Model and dataset cards, lineage, licenses, evaluation reports, release approvals, and rollback artifacts.
ASSESSMENT METHOD	MATURITY SIGNAL
Trace the decision path from identity and policy through execution, attempt bypass and privilege-escalation scenarios, verify revocation behavior, and inspect the resulting evidence record.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

MYTHOS-DAT-0005-6.1.4

6.1.4 Evidence Disconnect

Normative source requirement

Severity: Medium Description: The audit log proves an action happened, but cannot prove why it happened (e.g., missing the LLM prompt or policy decision), making incident response impossible.

Required Controls: 1. Automated Evidence Bundle generation for all high-risk actions. 2. Bundles must include full cognitive context, not just tool execution results.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Context manifests, retrieval traces, source citations, freshness records, conflict decisions, and memory provenance. Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
ASSESSMENT METHOD	MATURITY SIGNAL
Inject stale, conflicting, low-authority, and malicious information; inspect selection and citation behavior; verify scope isolation, correction, deletion, and provenance retention.	0 absent | 1 ad hoc | 2 repeatable | 3 controlled | 4 measured | 5 continuously verified

Decision rule. A criterion is not satisfied by documentation alone when the claim concerns runtime behavior. Reproduced evidence, responsible ownership, known limitations, and controlled exceptions are required.

Current authority and freshness register

These sources inform interpretation but do not convert this candidate publication into certification, legal compliance, or implementation evidence. Rolling sources must be version-pinned at adoption time.

AUTHORITY	VERSION / FRESHNESS	APPLICABILITY LIMIT
NIST - Artificial Intelligence Risk Management Framework (AI RMF 1.0)	NIST AI 100-1; current framework, revision underway Expires 2026-10-24	Voluntary risk-management framework; does not establish product certification or implementation effectiveness.
OpenTelemetry - Semantic Conventions	1.43.0 rolling specification; GenAI conventions maintained separately Expires 2026-08-24	Several GenAI and agent conventions remain under active development and require version pinning.
W3C - Verifiable Credentials Data Model v2.0	W3C Recommendation, May 15 2025 Expires 2027-01-24	Data model conformance does not establish issuer trust, credential truth, or ecosystem governance.
C2PA - Content Credentials Technical Specification	2.4 rolling publication Expires 2026-10-24	Provenance establishes signed assertions and tamper evidence, not the truth or quality of the underlying content.

Source lineage appendix

The following text preserves the substantive source draft in a normalized public form. Where it conflicts with the permanent publication identity, candidate status, current authority register, or evidence requirements above, the controlled publication layer governs.

0. Executive Mandate

The architecture of audit and compliance recording has failed.

Organizations attempt to prove what their systems-or autonomous agents-did by writing logs to an Elasticsearch cluster or a `logs` database table. When a breach occurs or an agent makes a catastrophic error, engineers query the database, only to find the logs were mutated, the timestamps were spoofed, or the offending records were silently `DELETE`d by the compromised account.

This standard exists because:

1. Mutable Logs are Not Evidence: A database record that can be updated or deleted by a root admin or a compromised service account is a claim, not proof. Production audit systems MUST be backed by cryptographic hash-chains where the integrity of record N depends on the immutability of record N-1.

2. Timestamps are Easily Spoofed: Relying on `created_at` database columns or local server clocks for audit trails is a security vulnerability. Temporal proof MUST be established via cryptographic signing (Ed25519) and external transparency logs (Rekor), not trust in the NTP daemon.

3. "Trust Me" is Not an Agent Policy: When an autonomous agent executes a high-risk action (e.g., modifying infrastructure, transferring funds, deleting data), it cannot simply log "I did X." It MUST produce an exportable, cryptographically verifiable Evidence Bundle that contains the user identity, the policy decision, the model prompt/output, and the deterministic execution proof.

4. Attestations Must Be Bound to Artifacts: Provenance is meaningless if it is detached from the artifact it describes. Systems MUST use in-toto attestations to bind metadata (who built it, what approved it, what scanned it) directly to the cryptographic hash of the artifact or action.

This document establishes the Mythos Evidence & Integrity Standard (MEIS), a comprehensive, evidence-based methodology for evaluating audit and logging systems against tamper-evidence, non-repudiation, and compliance verifiability.

Part I. Scope and Applicability

1.1 In Scope

This standard covers the evaluation of:

- Hash-chained event logs and append-only ledgers (BLAKE3, SHA-256)
- Cryptographic signing of log entries and checkpoints (Ed25519)
- Transparency logs and public verifiability (Sigstore, Rekor)
- Evidence Bundle generation for AI agents and automated systems
- in-toto attestations and artifact provenance binding
- Merkle tree architectures for high-volume log aggregation
- Air-gapped and sovereign audit trail architectures

1.2 Out of Scope

- General application observability and tracing (covered in MYTHOS-DAT-0002)
- Event sourcing for state derivation (covered in MYTHOS-DAT-0004)
- Post-Quantum Cryptography algorithms (covered in MYTHOS-SEC-0001)

1.3 Audience

- Security engineers and architects designing SIEM and audit platforms
- Platform engineers building compliance pipelines
- AI engineers building agent evidence and governance systems
- GRC (Governance, Risk, Compliance) and audit teams
- Standards bodies seeking a reference tamper-evidence methodology

Part II. Definitions and Taxonomy

2.1 Integrity Dimensions

Dimension	Definition
Hash-Chained Ledger	An append-only log where each entry contains the cryptographic hash of the previous entry, making retroactive modification immediately detectable.
Evidence Bundle	A cryptographically signed, self-contained package containing all logs, traces, context, and policy decisions required to independently prove the validity of a specific system action.
Transparency Log	An append-only cryptographic ledger (e.g., Rekor) that allows public or third-party verification that a record existed at a specific point in time without revealing the record's contents.
in-toto Attestation	A cryptographically signed metadata statement that binds a lifecycle event (e.g., "verified by policy") to a specific artifact via its cryptographic hash.
Non-Repudiation	The cryptographic guarantee that an action was performed by a specific identity, such that the identity cannot later deny having performed it.

2.2 The Evidence Doctrine

> A mutable log is a liability. A timestamp is a claim. A system without an exportable Evidence Bundle cannot prove its innocence. If the root admin can alter the audit trail, the audit trail is fiction.

Part III. Evaluation Methodology

3.1 Scoring Model

Each capability is scored from 0 to 5.

Score	Meaning
0	Fails basic task; mutable database logs, no cryptographic integrity
1	Append-only database table, but no hashing or signing
2	Basic hashing, but lacks cryptographic signing or external transparency
3	Solid production performance; BLAKE3 hash-chaining, Ed25519 signed checkpoints
4	Strong performance; Rekor transparency integration, in-toto attestations
5	Best-in-class; sets the standard for mathematically proven, zero-trust audit integrity

Weighting

Category	Weight	Rationale
Hash-Chain & Ledger Integrity	20%	Without cryptographic chaining, logs are easily tampered
Cryptographic Non-Repudiation	20%	Unsigned logs cannot be attributed to an actor
Evidence Bundle Completeness	20%	Disconnected logs do not prove an agent's context
Transparency & Third-Party Verifiability	15%	Internal logs can be forged by the org; external proof is required
Attestation & Provenance Binding	15%	Metadata must be cryptographically bound to artifacts
Operational Lifecycle & Retention	10%	Audit trails must survive system compromise

Total: 100%

A system MUST score at least 3.0 in Hash-Chain Integrity and Non-Repudiation to be considered for production use.

Part IV. Evaluation Categories

4.1 Hash-Chain and Ledger Integrity (Weight: 20%)

4.1.1 Chaining Mechanism

Is the log mathematically resistant to retroactive tampering?

Score	Criteria
0	Standard database table (UPDATE/DELETE permitted)
1	Append-only table, but no cryptographic linking
2	Each entry includes a hash of the previous entry
3	Forward-linked hash chain using BLAKE3, with periodic Merkle root checkpoints
4	Merkle tree architecture enabling efficient O(log N) inclusion proofs for individual entries
5	Perfect integrity: formally verified hash-chains, zero mutation capability at the storage layer, cryptographic proof of global log consistency

4.2 Cryptographic Non-Repudiation (Weight: 20%)

4.2.1 Action Signing

Can an action be cryptographically attributed to a specific identity?

Score	Criteria
0	Logs rely on application-level `user_id` strings
1	Logs signed by a shared symmetric key
2	Logs signed by an asymmetric key (Ed25519), but keys are long-lived and shared
3	Per-entity signing keys (e.g., per-agent or per-service Ed25519 keys)
4	Short-lived, scoped signing keys backed by hardware (TPM/HSM) or OIDC federation
5	Perfect non-repudiation: formally verified key provenance, cryptographic attestation of signer identity, zero repudiation possible

4.3 Evidence Bundle Completeness (Weight: 20%)

4.3.1 Contextual Proof

Can the system export a self-contained, verifiable proof of a consequential action?

Score	Criteria
0	Only a flat log line (e.g., "Agent X deleted file Y") is available
1	Log line includes a reference to an external trace ID
2	Exportable bundle includes the trace, but requires external systems to resolve context
3	Evidence Bundle includes the action, policy decision, user identity, and tool execution result
4	Bundle includes the full LLM prompt, context window, and model inference that triggered the action

Score	Criteria
5	Perfect bundle: cryptographically signed, self-contained package (using CBOR/COSE), verifiable offline by a third party with zero access to internal systems

4.4 Transparency and Third-Party Verifiability (Weight: 15%)

4.4.1 External Proof of Existence

Can a third party verify that a record existed at a specific time without seeing the record?

Score	Criteria
0	No external time-stamping; relies on local server clocks
1	NTP synchronized clocks, but no cryptographic time-stamping
2	Periodic hash checkpoints exported to an external system (e.g., S3 bucket)
3	Integration with a public transparency log (Sigstore Rekor) for checkpoint Merkle roots
4	Per-entry inclusion proofs verifiable against a public log
5	Perfect transparency: formally verified inclusion proofs, zero trust in local clocks, cryptographic resistance to split-viewing attacks

4.5 Attestation and Provenance Binding (Weight: 15%)

4.5.1 Artifact Provenance

Is lifecycle metadata cryptographically bound to the artifact it describes?

Score	Criteria
0	Metadata stored in a separate database, linked by an ID
1	Metadata includes the artifact's hash, but is not signed
2	Signed metadata, but no standardized schema
3	Standard in-toto attestations bound to artifact hashes
4	Attestations stored in a transparency log and verified at runtime before artifact use
5	Perfect provenance: formally verified attestation chains, zero ability to detach metadata from artifacts, automated policy enforcement on attestation presence

4.6 Operational Lifecycle and Retention (Weight: 10%)

4.6.1 Survivability

Does the audit trail survive a catastrophic system or infrastructure compromise?

Score	Criteria
0	Audit logs stored on the same infrastructure as the production system
1	Logs streamed to a separate logging cluster in the same cloud/account
2	Logs replicated to a separate cloud provider or on-prem environment
3	WORM (Write-Once-Read-Many) storage architecture for log retention
4	Cryptographically sealed historical archives with automated integrity verification
5	Perfect survivability: air-gapped, sovereign log retention, formally verified continuous integrity scanning, zero ability to destroy historical evidence

Part V. The Mythos Evidence Integrity Suite (MEIS)

Traditional logging benchmarks measure ingestion throughput (EPS). The MEIS evaluates tamper-resistance, non-repudiation, and evidence completeness.

5.1 Suite Composition

5.1.1 MEIS-Tamper

• Retroactive Modification: 100+ tests attempting to use privileged credentials to `UPDATE` or `DELETE` log entries, verifying the system detects the hash-chain break.

• Split-View Attack: 50+ tests attempting to present a fake, locally-generated log to a verifier, verifying the transparency log (Rekor) rejects the inclusion proof.

5.1.2 MEIS-Bundle

• Offline Verification: 100+ tests exporting an Evidence Bundle for a complex agent action and attempting to verify it on a disconnected, air-gapped machine.

• Context Fidelity: 50+ tests verifying the bundle contains the exact prompt, policy decision, and tool output that led to the action.

5.2 Execution Protocol

```
1. Deploy audit architecture with hash-chaining and transparency log integration
2. Execute complex, multi-step agentic workflows
3. Run MEIS suite (automated tamper injection + bundle verification)
4. Score each category 0-5
5. Check for mutable log storage (instant disqualification)
6. If all thresholds met: approve for production
```

Part VI. Security Threat Model for Evidence Systems

6.1 Threat Catalog

6.1.1 Retroactive Log Tampering (Cover-Up)

Severity: Critical Description: A compromised admin or attacker attempts to delete or modify logs that record their malicious actions.

Required Controls:

1. Forward-linked BLAKE3 hash chains.

2. Append-only, WORM storage at the infrastructure level.

3. Periodic Merkle root checkpoints to an external transparency log.

6.1.2 Split-Viewing Attack

Severity: High Description: A compromised system presents a valid, but fake, local audit log to an internal verifier, while hiding the true (malicious) actions in a separate log.

Required Controls:

1. Public transparency logs (Rekor) for all critical checkpoints.

2. Verifiers must check inclusion proofs against the public log, not just the local system.

6.1.3 Non-Repudiation Failure (Impersonation)

Severity: High Description: An attacker uses a shared, compromised symmetric key to forge log entries, framing another user or service.

Required Controls:

1. Per-identity asymmetric signing keys (Ed25519).

2. Short-lived, federated signing keys via OIDC/SPIFFE.

6.1.4 Evidence Disconnect

Severity: Medium Description: The audit log proves an action happened, but cannot prove why it happened (e.g., missing the LLM prompt or policy decision), making incident response impossible.

Required Controls:

1. Automated Evidence Bundle generation for all high-risk actions.

2. Bundles must include full cognitive context, not just tool execution results.

Part VII. The Mythos Evidence & Integrity Standard

7.1 The Mythos Evidence Doctrine

```
A mutable log is a liability.
A timestamp is a claim.

A system without an exportable Evidence Bundle cannot prove its innocence.
If the root admin can alter the audit trail, the audit trail is fiction.

Actions may be asynchronous.
Integrity must be absolute.
```

7.2 Selection Rules

1. No audit architecture enters production without passing MEIS.
2. No architecture is approved if it uses mutable database tables for audit logs.
3. No architecture is approved without cryptographic hash-chaining of entries.
4. No architecture is approved without per-identity cryptographic signing.
5. No agent is approved for high-risk actions without Evidence Bundle generation.

7.3 The Prime Directive for Evidence Architecture

> Chain the hash, do not just append the row. > Sign the action, do not just log the user. > Bundle the context, do not just save the trace. > Prove the existence, do not just trust the clock.

```
Academic benchmarks measure events per second.
Applied benchmarks measure hash-chain integrity.
Security benchmarks measure non-repudiation.
Operational benchmarks measure evidence completeness.

> Systems may be compromised.
> Evidence must be absolute.
```

End of Standard MYTHOS-DAT-0005

End of controlled publication

MYTHOS-DAT-0005 remains a Candidate Standard until technical review, security and privacy review, accessibility review, current-source validation, and formal approval are recorded.