ENGINEERING STANDARDS LIBRARY / DATA, STATE, AND EVIDENCE

AI & Agent Observability Semantic Conventions (OpenTelemetry) Standard

The observability of AI systems has failed.

PERMANENT DOCUMENT ID

MYTHOS-DAT-0002

PUBLICATION

Candidate Standard
Version 1.0.0-candidate

ASSURANCE PROFILE

Standard / Advanced
High Assurance

PERMANENT PUBLICATION IDENTITY

AI & Agent Observability Semantic Conventions (OpenTelemetry) Standard

DOCUMENT ID
MYTHOS-DAT-0002

VERSION
1.0.0-candidate

STATUS
Candidate Standard

PUBLISHER
Mythos Systems

PUBLICATION DATE
2026-07-24

SCHEDULED REVIEW
2027-07-24

11	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

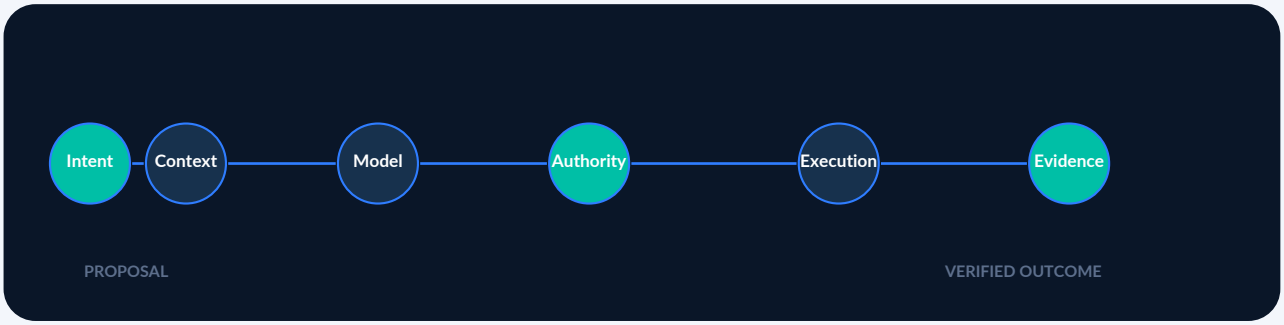
MYTHOS-DAT-0002 inside the data, state, and evidence constellation

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

Establishes semantic conventions and operational requirements for observing models, agents, tools, retrieval, memory, costs, safety events, and end-to-end AI execution traces.

WHY THIS STANDARD EXISTS	The observability of AI systems has failed.
OPERATIONAL SCOPE	This standard covers the evaluation of: - OpenTelemetry (OTel) GenAI semantic conventions and instrumentation - LLM/Agent tracing libraries (Langfuse, Arize, Phoenix, OpenLLMetry) - Token and cost observability (per-span token attribution) - Prompt and completion redaction pipelines - Tamper-evident audit trails and evidence bundle generation - eBPF and kernel-level observability for local model runtimes
PRIMARY AUDIENCE	- Platform engineers and SREs managing AI infrastructure - AI engineers building evaluation and tracing pipelines - Security teams auditing agent behavior and evidence integrity - FinOps teams managing token and compute economics - Standards bodies seeking a reference GenAI observability 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 11 atomic criteria. Each criterion is independently testable and requires retained evidence.

ID	EVALUATION CRITERION
4.1.1	Standardization
4.2.1	Reasoning Visibility
4.3.1	Economic Attribution
4.4.1	Data Sanitization
4.5.1	Tamper Evidencing
4.6.1	Inference Runtime Tracing
5.1.1	M-OBS-Cognition
5.1.2	M-OBS-Evidence
6.1.1	Telemetry Poisoning
6.1.2	PII Egress via Prompts
6.1.3	Evidence Tampering

MYTHOS-DAT-0002-4.1.1

4.1.1 Standardization

Normative source requirement

Does the system emit telemetry using standard OpenTelemetry GenAI semantic conventions?

Score	Criteria
0	Proprietary logging format only
1	Custom OTel attributes, but ignores GenAI conventions
2	Uses some GenAI conventions (`gen_ai.*`), but incomplete
3	Full compliance with current OTel GenAI semantic conventions for prompts, completions, and models
4	Extends conventions for agent-specific concepts (planning, handoffs) while maintaining OTel compatibility
5	Perfect compliance: formally verified semantic mappings, zero proprietary lock-in, automated conformance testing against OTel specs

---

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. Model and dataset cards, lineage, licenses, evaluation reports, release approvals, and rollback artifacts.
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-0002-4.2.1

4.2.1 Reasoning Visibility

Normative source requirement

Can the observer reconstruct the full cognitive loop and context window of the agent?

Score	Criteria
0	Only API call latency is recorded
1	Prompts and completions logged as flat text
2	Traces show model calls and tool calls as separate spans
3	Traces show context assembly, system prompts, tool selection, and execution as a hierarchical DAG
4	Full temporal replay: ability to reconstruct the exact context window at any point in the trace
5	Perfect visibility: formally verified cognitive traces, automated hallucination detection in spans, zero blind spots in the reasoning loop

---

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-0002-4.3.1

4.3.1 Economic Attribution

Normative source requirement

Are token usage and compute costs precisely tracked per task, agent, and user?

Score	Criteria
0	No token tracking
1	Total tokens logged per API call
2	Input/output tokens separated and attributed to a user/session
3	Cost calculated in real-time based on provider pricing models; per-task cost aggregation
4	Context cache hits, prefix caching, and local-vs-cloud compute costs accurately tracked
5	Perfect FinOps: real-time budget enforcement tied to traces, automated anomaly detection for cost spikes, formally verified economic attribution

---

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-0002-4.4.1

4.4.1 Data Sanitization

Normative source requirement

Is sensitive data prevented from egressing to the observability backend?

Score	Criteria
0	Raw prompts and completions exported in plain text
1	Manual redaction of sensitive fields
2	Regex-based PII redaction at the edge
3	Deterministic, policy-driven redaction using NER (Named Entity Recognition) before export
4	Hash-based pseudonymization: sensitive data replaced with cryptographic hashes for correlation without exposure
5	Perfect security: formally verified zero-PII egress, hardware-backed redaction (TEE), automated telemetry poisoning detection

---

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-0002-4.5.1

4.5.1 Tamper Evidencing

Normative source requirement

Can telemetry be used as legally binding evidence of an agent's action?

Score	Criteria
0	Logs stored in mutable database
1	Append-only logs, but no cryptographic signing
2	BLAKE3 hashing of log entries
3	Hash-chained event log with Ed25519 signed checkpoints
4	Integration with Sigstore/Rekor transparency logs for public verifiability
5	Perfect evidence: in-toto attestations, formally verified hash-chains, exportable compliance bundles with full cryptographic provenance

---

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. Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
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-0002-4.6.1

4.6.1 Inference Runtime Tracing

Normative source requirement

How deeply can local/self-hosted model runtimes be observed?

Score	Criteria
0	No visibility into local model execution
1	Basic application-level metrics (tokens/sec)
2	GPU utilization and VRAM tracking via DCGM
3	eBPF-based tracing of kernel-level inference latencies and context switching
4	KV-cache hit/miss ratios, batch processing efficiency, and prefill/decode phase tracing
5	Perfect runtime observability: formally verified eBPF traces, hardware-level performance counters, zero-overhead continuous profiling

---

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-0002-5.1.1

5.1.1 M-OBS-Cognition

Normative source requirement

• Trace Reconstruction: 100+ tests verifying that a complete agent reasoning loop can be reconstructed purely from exported telemetry.

• Context Fidelity: 50+ tests verifying that the context window state at step N matches the recorded span attributes.

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-0002-5.1.2

5.1.2 M-OBS-Evidence

Normative source requirement

• Tamper Test: 50+ tests attempting to mutate, delete, or inject telemetry entries to verify the system detects and rejects the corruption.

• Redaction Verification: 100+ tests injecting PII/secrets into prompts to verify they are scrubbed before hitting the backend.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
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-0002-6.1.1

6.1.1 Telemetry Poisoning

Normative source requirement

Severity: High Description: An attacker (or a prompt injection) causes the agent to emit specially crafted log lines or span attributes designed to exploit vulnerabilities in the observability backend (e.g., log4j style injection, UI XSS in Datadog).

Required Controls: 1. Strict schema validation on all telemetry attributes. 2. Sanitization of model outputs before they are attached to spans. 3. Treat model outputs as untrusted data in the observability pipeline.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Model and dataset cards, lineage, licenses, evaluation reports, release approvals, and rollback artifacts. Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
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-0002-6.1.2

6.1.2 PII Egress via Prompts

Normative source requirement

Severity: Critical Description: The agent processes user PII and logs the raw prompt to a third-party APM tool, violating data sovereignty agreements (GDPR, HIPAA, DoD CMMC).

Required Controls: 1. Edge redaction of all prompt/completion spans. 2. Hash-based pseudonymization for maintaining trace correlation without exposing data.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
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-0002-6.1.3

6.1.3 Evidence Tampering

Normative source requirement

Severity: Critical Description: A malicious insider or compromised agent modifies the telemetry logs to cover up an unauthorized action or make it appear as if a user approved it.

Required Controls: 1. Append-only, hash-chained event logs. 2. Cryptographic signing of telemetry checkpoints. 3. Transparency log (Rekor) integration for root-of-trust.

CONTROL INTENT	REQUIRED EVIDENCE
Create a measurable, reviewable control that can be verified independently of model or vendor claims.	Trace schemas, immutable event records, integrity verification, retention policy, and operator queries.
ASSESSMENT METHOD	MATURITY SIGNAL
Exercise crash, retry, duplicate, reorder, partition, and restore scenarios; compare authoritative state before and after replay; confirm idempotency and recovery evidence.	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 observability of AI systems has failed.

Organizations attempt to monitor autonomous agents using legacy APM tools designed for deterministic web requests. They trace HTTP latency to an `/chat/completions` endpoint and declare the system "observable." But an API latency of 200ms tells you nothing about whether the agent hallucinated, ignored a tool result, leaked a secret in its prompt, or executed a destructive action based on flawed reasoning.

This standard exists because:

1. Standard APM is Blind to Cognition: A traditional trace shows a function call. An agent trace must show the prompt, the context assembly, the model routing, the tool selection, the parsed output, and the reasoning loop. Without GenAI semantic conventions, agent execution is a black box.

2. "Logs" are Not Evidence: An agent logging "I deleted the database" is a claim, not evidence. Production agent observability MUST produce tamper-evident, cryptographically signed audit trails that bind an action to its originating policy, user, and model inference.

3. Token Economics Must Be Traced, Not Estimated: FinOps for AI cannot rely on end-of-month API bills. Token usage, context window consumption, and cost-per-task must be captured as real-time span attributes on every model invocation.

4. Telemetry is a Privacy Threat: Dumping raw prompts and completions into Datadog or New Relic exposes PII, intellectual property, and secrets to the observability vendor. Telemetry MUST be redacted at the edge before egress, using deterministic pipelines.

This document establishes the Mythos AI Observability Standard (MAIOS), a comprehensive, evidence-based methodology for evaluating AI telemetry pipelines against OpenTelemetry compliance, cognitive tracing, and evidence integrity.

Part I. Scope and Applicability

1.1 In Scope

This standard covers the evaluation of:

- OpenTelemetry (OTel) GenAI semantic conventions and instrumentation
- LLM/Agent tracing libraries (Langfuse, Arize, Phoenix, OpenLLMetry)
- Token and cost observability (per-span token attribution)
- Prompt and completion redaction pipelines
- Tamper-evident audit trails and evidence bundle generation
- eBPF and kernel-level observability for local model runtimes

1.2 Out of Scope

- General application and infrastructure observability (covered in future MYTHOS-SRE standards)
- General data sovereignty (covered in MYTHOS-DAT-0003)
- Formal verification of agent state machines (covered in MYTHOS-SWE-0007)

1.3 Audience

- Platform engineers and SREs managing AI infrastructure
- AI engineers building evaluation and tracing pipelines
- Security teams auditing agent behavior and evidence integrity
- FinOps teams managing token and compute economics
- Standards bodies seeking a reference GenAI observability methodology

Part II. Definitions and Taxonomy

2.1 Observability Dimensions

Dimension	Definition
GenAI Semantic Conventions	The standardized OpenTelemetry attribute names and structures (e.g., `gen_ai.system`, `gen_ai.request.model`) used to describe LLM and agent operations.
Cognitive Trace	A distributed trace that captures the full reasoning loop of an agent, including context assembly, system prompts, tool selections, and model outputs.
Evidence Bundle	A cryptographically signed, hash-chained export of a trace and its associated logs/context, used for compliance and dispute resolution.
Telemetry Poisoning	The injection of malicious content into prompts or logs designed to exploit the observability backend (e.g., log injection, prompt injection via telemetry).
Edge Redaction	The process of scrubbing PII, secrets, and sensitive data from telemetry payloads at the application boundary before transmission to any backend.

2.2 The Observability Doctrine

> An API latency of 200ms does not prove the agent was right. A log entry is a claim, not proof. A prompt that is not traced is a security blind spot. An agent without a cognitive trace is ungovernable.

Part III. Evaluation Methodology

3.1 Scoring Model

Each capability is scored from 0 to 5.

Score	Meaning
0	Fails basic task; standard APM, no GenAI context, raw PII logged
1	Basic LLM logging but not OTel-compliant
2	Functional tracing but lacks token/cost tracking or redaction
3	Solid production performance; OTel GenAI conventions, edge redaction, token metrics
4	Strong performance; full cognitive traces, tamper-evident audit logs, eBPF integration
5	Best-in-class; sets the standard for mathematically verified, zero-trust AI observability

Weighting

Category	Weight	Rationale
OTel GenAI Semantic Compliance	20%	Proprietary telemetry formats create vendor lock-in
Cognitive & Agent Loop Tracing	20%	Standard traces cannot capture non-deterministic reasoning
Token & Cost Economics (FinOps)	15%	Untracked tokens lead to unbounded financial risk
Telemetry Security & Edge Redaction	15%	Raw prompts in telemetry are a critical data leak
Evidence Integrity & Audit Trails	15%	Logs must be tamper-evident to serve as compliance proof
Local Model & eBPF Observability	15%	Local inference cannot be observed via API wrappers

Total: 100%

A system MUST score at least 3.0 in OTel GenAI Semantic Compliance and Telemetry Security to be considered for production use.

Part IV. Evaluation Categories

4.1 OTel GenAI Semantic Compliance (Weight: 20%)

4.1.1 Standardization

Does the system emit telemetry using standard OpenTelemetry GenAI semantic conventions?

Score	Criteria
0	Proprietary logging format only
1	Custom OTel attributes, but ignores GenAI conventions
2	Uses some GenAI conventions (`gen_ai.*`), but incomplete
3	Full compliance with current OTel GenAI semantic conventions for prompts, completions, and models
4	Extends conventions for agent-specific concepts (planning, handoffs) while maintaining OTel compatibility
5	Perfect compliance: formally verified semantic mappings, zero proprietary lock-in, automated conformance testing against OTel specs

4.2 Cognitive and Agent Loop Tracing (Weight: 20%)

4.2.1 Reasoning Visibility

Can the observer reconstruct the full cognitive loop and context window of the agent?

Score	Criteria
0	Only API call latency is recorded
1	Prompts and completions logged as flat text
2	Traces show model calls and tool calls as separate spans
3	Traces show context assembly, system prompts, tool selection, and execution as a hierarchical DAG
4	Full temporal replay: ability to reconstruct the exact context window at any point in the trace
5	Perfect visibility: formally verified cognitive traces, automated hallucination detection in spans, zero blind spots in the reasoning loop

4.3 Token and Cost Economics (FinOps) (Weight: 15%)

4.3.1 Economic Attribution

Are token usage and compute costs precisely tracked per task, agent, and user?

Score	Criteria
0	No token tracking
1	Total tokens logged per API call
2	Input/output tokens separated and attributed to a user/session
3	Cost calculated in real-time based on provider pricing models; per-task cost aggregation
4	Context cache hits, prefix caching, and local-vs-cloud compute costs accurately tracked
5	Perfect FinOps: real-time budget enforcement tied to traces, automated anomaly detection for cost spikes, formally verified economic attribution

4.4 Telemetry Security and Edge Redaction (Weight: 15%)

4.4.1 Data Sanitization

Is sensitive data prevented from egressing to the observability backend?

Score	Criteria
0	Raw prompts and completions exported in plain text
1	Manual redaction of sensitive fields
2	Regex-based PII redaction at the edge
3	Deterministic, policy-driven redaction using NER (Named Entity Recognition) before export
4	Hash-based pseudonymization: sensitive data replaced with cryptographic hashes for correlation without exposure
5	Perfect security: formally verified zero-PII egress, hardware-backed redaction (TEE), automated telemetry poisoning detection

4.5 Evidence Integrity and Audit Trails (Weight: 15%)

4.5.1 Tamper Evidencing

Can telemetry be used as legally binding evidence of an agent's action?

Score	Criteria
0	Logs stored in mutable database
1	Append-only logs, but no cryptographic signing
2	BLAKE3 hashing of log entries
3	Hash-chained event log with Ed25519 signed checkpoints
4	Integration with Sigstore/Rekor transparency logs for public verifiability
5	Perfect evidence: in-toto attestations, formally verified hash-chains, exportable compliance bundles with full cryptographic provenance

4.6 Local Model and eBPF Observability (Weight: 15%)

4.6.1 Inference Runtime Tracing

How deeply can local/self-hosted model runtimes be observed?

Score	Criteria
0	No visibility into local model execution
1	Basic application-level metrics (tokens/sec)
2	GPU utilization and VRAM tracking via DCGM
3	eBPF-based tracing of kernel-level inference latencies and context switching
4	KV-cache hit/miss ratios, batch processing efficiency, and prefill/decode phase tracing
5	Perfect runtime observability: formally verified eBPF traces, hardware-level performance counters, zero-overhead continuous profiling

Part V. The Mythos Observability Suite (M-OBS)

Traditional observability benchmarks measure API latency and error rates. The M-OBS evaluates cognitive tracing, economic attribution, and evidence integrity.

5.1 Suite Composition

5.1.1 M-OBS-Cognition

- Trace Reconstruction: 100+ tests verifying that a complete agent reasoning loop can be reconstructed purely from exported telemetry.
- Context Fidelity: 50+ tests verifying that the context window state at step N matches the recorded span attributes.

5.1.2 M-OBS-Evidence

- Tamper Test: 50+ tests attempting to mutate, delete, or inject telemetry entries to verify the system detects and rejects the corruption.
- Redaction Verification: 100+ tests injecting PII/secrets into prompts to verify they are scrubbed before hitting the backend.

5.2 Execution Protocol

```
1. Deploy agent architecture with observability pipeline
2. Execute complex, multi-step agentic workloads
3. Run M-OBS suite (automated trace analysis + tamper injection)
4. Score each category 0-5
5. Check for raw PII in observability backend (instant disqualification)
6. If all thresholds met: approve for production
```

Part VI. Security Threat Model for AI Observability

6.1 Threat Catalog

6.1.1 Telemetry Poisoning

Severity: High Description: An attacker (or a prompt injection) causes the agent to emit specially crafted log lines or span attributes designed to exploit vulnerabilities in the observability backend (e.g., log4j style injection, UI XSS in Datadog).

Required Controls:

1. Strict schema validation on all telemetry attributes.
2. Sanitization of model outputs before they are attached to spans.
3. Treat model outputs as untrusted data in the observability pipeline.

6.1.2 PII Egress via Prompts

Severity: Critical Description: The agent processes user PII and logs the raw prompt to a third-party APM tool, violating data sovereignty agreements (GDPR, HIPAA, DoD CMMC).

Required Controls:

1. Edge redaction of all prompt/completion spans.
2. Hash-based pseudonymization for maintaining trace correlation without exposing data.

6.1.3 Evidence Tampering

Severity: Critical Description: A malicious insider or compromised agent modifies the telemetry logs to cover up an unauthorized action or make it appear as if a user approved it.

Required Controls:

1. Append-only, hash-chained event logs.
2. Cryptographic signing of telemetry checkpoints.
3. Transparency log (Rekor) integration for root-of-trust.

Part VII. The Mythos Observability Standard

7.1 The Mythos Observability Doctrine

```
An API latency of 200ms does not prove the agent was right.
A log entry is a claim, not proof.

A prompt that is not traced is a security blind spot.
An agent without a cognitive trace is ungovernable.

Observability may be passive.
Evidence must be absolute.
```

7.2 Selection Rules

1. No AI observability architecture enters production without passing M-OBS.
2. No architecture is approved if it does not use OTel GenAI semantic conventions.
3. No architecture is approved if it exports raw PII to a telemetry backend.

4. No architecture is approved without real-time token and cost attribution.

5. No architecture is approved if its audit logs are mutable or unsigned.

7.3 The Prime Directive for AI Observability

> Trace the cognition, not just the call. > Redact the prompt, not just the payload. > Attribute the cost, not just the latency. > Bind the evidence, not just the log.

```
Academic benchmarks measure QPS.
Applied benchmarks measure cognitive tracing.
Security benchmarks measure edge redaction.
Operational benchmarks measure evidence integrity.

> Models may be stochastic.
> Evidence must be deterministic.
```

End of Standard MYTHOS-DAT-0002

End of controlled publication

MYTHOS-DAT-0002 remains a Candidate Standard until technical review, security and privacy review, accessibility review, current-source validation, and formal approval are recorded.