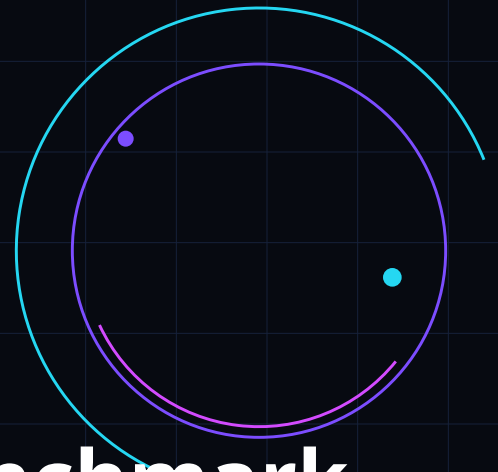




Agentic Framework Benchmark and Evaluation Standard

Defines qualification, evaluation, governance, tool use, recovery, evidence, and scoring for agent frameworks and runtime fabrics.

MYTHOS-AI-0001

Candidate Mythos Standard / 2.0.0-candidate

Published 2026-07-26 | Review 2026-10-26 | Public

Mythos Systems

Permanent Identity and Edition Record

Permanent document ID	MYTHOS-AI-0001
Canonical title	Agentic Framework Benchmark and Evaluation Standard
Publication class	Standards & Benchmarks
Primary domain	MYTHOS-AI - Artificial Intelligence & Agentic Systems
Version	2.0.0-candidate
Status	Candidate Mythos Standard
Publication date	2026-07-26
Scheduled review	2026-10-26
Publisher	Mythos Systems
Owner	Aaron Stovall
Classification	Public
Prior edition	1.0.0-candidate / 2026-07-24
Supersession	This edition supersedes the prior candidate while preserving the permanent identity and historical source.

Identity doctrine

The permanent document ID is immutable. Production sequence labels are not part of the public identity. Atomic standards are authoritative; portfolios, workbooks, and machine-readable exports are generated views.

Normative Language and Applicability

The key words MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL indicate the strength of provisions in this Mythos standard. The publication is a Mythos-authored candidate standard and does not claim approval by the cited external authorities.

Applicability rule

Every control is applicable unless the organization documents a specific architectural or lifecycle reason that the subject is absent. N/A decisions MUST name the decision owner, evidence, affected scope, review date, and triggers that invalidate the exclusion.

Conformance evidence hierarchy

Direct deterministic proof: schemas, tests, signatures, state reconciliation, access decisions, build or runtime evidence.

Reproducible technical evidence: benchmark fixtures, traces, artifacts, profiles, logs, metrics, and environment manifests.

Independent assessment: qualified human or separate evaluator using an explicit rubric and disclosed uncertainty.

Attestation or supplier claim: informative unless independently verified and current for the deployed configuration.

Demonstration or narrative: insufficient by itself for a conforming score above 2.

Critical-control doctrine

Critical controls cannot be averaged away. A failed or stale critical control constrains the maximum conformance level regardless of the aggregate score.

Scope, Audience, and Engineering Principles

Purpose. Defines qualification, evaluation, governance, tool use, recovery, evidence, and scoring for agent frameworks and runtime fabrics.

In-scope systems. agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems.

Primary audience. AI platform engineers, agent-runtime architects, security engineers, evaluators, software leaders, and acquisition teams.

Required engineering principles

Model capability and complete system behavior **MUST** be measured separately.

Human, legal, production, financial, identity, and governance authority **MUST** remain external to model confidence or conversational fluency.

Evaluation **MUST** include expected, prohibited, adversarial, degraded, recovery, and lifecycle-change conditions.

Source authority, data rights, permissions, versions, and freshness **MUST** be visible in evidence.

Critical outcomes **MUST** be supported by deterministic validators or independent review wherever feasible.

The organization **MUST** preserve rollback, revocation, correction, deletion, and retirement paths.

Optimization for score, latency, throughput, tokens, or cost **MUST NOT** hide safety, quality, privacy, accessibility, or reliability regressions.

Exclusions

This standard does not certify a model, provider, framework, benchmark, dataset, or product. Conformance is limited to the declared system, use case, version, environment, evidence period, and assessment scope.

Conformance and Scoring Model

Level	Weighted threshold	Critical-control condition	Meaning
No conformance	< 50	Any state	Materially incomplete, unverified, or unsafe.
Foundation	>= 50	No critical control scored 0	Defined baseline with partial implementation and evidence.
Operational	>= 65	All applicable critical controls >= 3	Implemented and operationally tested.
Assured	>= 80	All applicable critical controls >= 4	Independently verified with adversarial and recovery evidence.
Continuously Assured	>= 90	All applicable critical controls = 5	Continuous regression, monitoring, freshness, and incident learning.

Weighted score

For applicable controls: weighted contribution = (control score / 5) x control weight. N/A controls are removed from the denominator only after an approved applicability decision. The final score is normalized to 100.

A conformance claim MUST include scope, system and model versions, assessment date, evidence window, evaluator identity, scores by family, critical-control status, unresolved findings, exceptions, limitations, and next review.

Control Score Interpretation

Score	Maturity	Required evidence state
0	Not implemented	Not implemented: the assessed control is absent, materially unknown, or contradicted by evidence.
1	Initial	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Nonconformity classes

Critical: credible risk of serious harm, unauthorized high-impact action, material security or privacy failure, false assurance, or inability to recover.

Major: requirement absent or ineffective across a meaningful part of scope, with no sufficient compensating control.

Minor: localized or low-impact deficiency that does not invalidate the overall control objective.

Observation: improvement opportunity or emerging risk without current nonconformity.

Score validity

A score expires when evidence exceeds its approved freshness period or a material change invalidates the assessment. Current operation is not inferred from a historic assessment.

Standard Architecture and Control Model

01	Govern	Scope, ownership, authority, policy, impacts, risk tolerance, exceptions, and lifecycle decisions.
02	Map	System boundaries, actors, models, data, prompts, tools, interfaces, populations, dependencies, and failure domains.
03	Measure	Representative and hidden evaluation, deterministic validation, human or model judging, uncertainty, performance, and cost.
04	Manage	Release gates, least privilege, monitoring, incident response, correction, rollback, revocation, deletion, and retirement.
05	Prove	Traceable evidence bundles connecting claims to configurations, tests, artifacts, effects, decisions, and user outcomes.

Assessment unit

The assessment unit is the declared AI system or workflow, not the model name alone. It includes prompts, retrieval, memory, tools, policies, interface, evaluators, infrastructure, human oversight, and operating environment.

Benchmark Dimensions and Weights

Dimension	Weight	Assessment emphasis
Governance and authority	18%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for governance and authority.
Runtime safety and isolation	20%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for runtime safety and isolation.
Durability and recovery	15%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for durability and recovery.
Evaluation quality	18%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for evaluation quality.
Interoperability and portability	10%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for interoperability and portability.
Operations and economics	10%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for operations and economics.
Lifecycle and exit	9%	Control effectiveness, evidence quality, critical failures, operational limits, and lifecycle behavior for lifecycle and exit.

Benchmark scores **MUST** be reported by dimension and by critical-control status. A single aggregate ranking **MUST NOT** be used to conceal weak safety, authority, privacy, recovery, or evidence performance.

Contents and Control Index

[Permanent identity and edition record](#)

[Normative language and applicability](#)

[Scope, audience, and engineering principles](#)

[Conformance and scoring model](#)

[Standard architecture and control model](#)

[Benchmark dimensions and weights](#)

[Control summary matrix](#)

MYTHOS-AI-0001-C01 - Evaluation Scope and Use-Case Contract

MYTHOS-AI-0001-C02 - Framework and Runtime Taxonomy

MYTHOS-AI-0001-C03 - Model and Provider Independence

MYTHOS-AI-0001-C04 - Authority Separation

MYTHOS-AI-0001-C05 - Tool Contract and Capability Governance

MYTHOS-AI-0001-C06 - Secret Isolation and Credential Brokering

MYTHOS-AI-0001-C07 - Durability and Crash Recovery

MYTHOS-AI-0001-C08 - Checkpoint, Replay, and Side-Effect Safety

MYTHOS-AI-0001-C09 - Human Approval and Interruption

MYTHOS-AI-0001-C10 - Memory and Context Boundaries

MYTHOS-AI-0001-C11 - Multi-Agent Delegation and Attenuation

MYTHOS-AI-0001-C12 - Sandboxing and Execution Isolation

MYTHOS-AI-0001-C13 - Observability and Evidence

MYTHOS-AI-0001-C14 - Evaluation Harness and Reproducibility

MYTHOS-AI-0001-C15 - Interoperability and Protocol Support

MYTHOS-AI-0001-C16 - Operational Scaling and Cost Controls

MYTHOS-AI-0001-C17 - Project Health, Licensing, and Exit Strategy

MYTHOS-AI-0001-C18 - Decision Record and Reassessment

MYTHOS-AI-0001-C19 - Retirement and Evidence Preservation

Control Summary Matrix

Control	Requirement	Family	Wt.	Crit.
C01	Evaluation Scope and Use-Case Contract	Governance and authority	6	YES
C02	Framework and Runtime Taxonomy	Governance and authority	6	-
C03	Model and Provider Independence	Governance and authority	6	-
C04	Authority Separation	Runtime safety and isolation	5	YES
C05	Tool Contract and Capability Governance	Runtime safety and isolation	5	YES
C06	Secret Isolation and Credential Brokering	Runtime safety and isolation	5	YES
C07	Durability and Crash Recovery	Runtime safety and isolation	5	-
C08	Checkpoint, Replay, and Side-Effect Safety	Durability and recovery	5	YES
C09	Human Approval and Interruption	Durability and recovery	5	YES
C10	Memory and Context Boundaries	Durability and recovery	5	-

Control Summary Matrix - Continued

Control	Requirement	Family	Wt.	Crit.
C11	Multi-Agent Delegation and Attenuation	Evaluation quality	6	-
C12	Sandboxing and Execution Isolation	Evaluation quality	6	YES
C13	Observability and Evidence	Evaluation quality	6	YES
C14	Evaluation Harness and Reproducibility	Interoperability and portability	5	YES
C15	Interoperability and Protocol Support	Interoperability and portability	5	-
C16	Operational Scaling and Cost Controls	Operations and economics	5	-
C17	Project Health, Licensing, and Exit Strategy	Operations and economics	5	-
C18	Decision Record and Reassessment	Lifecycle and exit	5	-
C19	Retirement and Evidence Preservation	Lifecycle and exit	4	-

Weight integrity

The 19 applicable control weights sum to 100. Family weights match the benchmark dimension model. Critical controls are highlighted in the atomic control pages and assessment workbook.

MYTHOS-AI-0001-C01

Evaluation Scope and Use-Case Contract

Family: Governance and authority | Weight: 6 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain evaluation scope and use-case contract for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The operating contract must define intended users, decisions, prohibited uses, affected systems, consequences, risk tolerance, terminal conditions, and accountable authority. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that evaluation scope and use-case contract is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for evaluation scope and use-case contract covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The operating contract must define intended users, decisions, prohibited uses, affected systems, consequences, risk tolerance, terminal conditions, and accountable authority.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C01

Evaluation Scope and Use-Case Contract

Family: Governance and authority | Weight: 6 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for evaluation scope and use-case contract.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

- approved control design or operating contract with accountable owner
- versioned configuration, schema, policy, model, prompt, dataset, or architecture record
- reproducible test plan with expected and observed results
- traceable decision, exception, approval, and remediation records
- negative, boundary, degraded, and recovery test results
- operational telemetry and current-state verification

Required metrics

- task success with confidence interval
- critical-failure rate
- slice variance
- reproducibility delta
- human-review burden

Score anchors

0	Not implemented: Evaluation Scope and Use-Case Contract is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for evaluation scope and use-case contract.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C02

Framework and Runtime Taxonomy

Family: Governance and authority | Weight: 6 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain framework and runtime taxonomy for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The inventory must identify versions, owners, suppliers, deployment locations, interfaces, dependencies, trust boundaries, and supported lifecycle states. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that framework and runtime taxonomy is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for framework and runtime taxonomy covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The inventory must identify versions, owners, suppliers, deployment locations, interfaces, dependencies, trust boundaries, and supported lifecycle states.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C02

Framework and Runtime Taxonomy

Family: Governance and authority | Weight: 6 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for framework and runtime taxonomy.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Framework and Runtime Taxonomy is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for framework and runtime taxonomy.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C03

Model and Provider Independence

Family: Governance and authority | Weight: 6 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain model and provider independence for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that model and provider independence is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for model and provider independence covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C03

Model and Provider Independence

Family: Governance and authority | Weight: 6 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for model and provider independence.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Model and Provider Independence is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for model and provider independence.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C04

Authority Separation

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain authority separation for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that authority separation is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for authority separation covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C04

Authority Separation

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for authority separation.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Authority Separation is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for authority separation.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C05

Tool Contract and Capability Governance

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain tool contract and capability governance for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Capabilities must use typed contracts, bounded side effects, explicit authorization, schema validation, progress and cancellation, result evidence, compatibility tests, and revocation. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that tool contract and capability governance is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for tool contract and capability governance covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Capabilities must use typed contracts, bounded side effects, explicit authorization, schema validation, progress and cancellation, result evidence, compatibility tests, and revocation.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C05

Tool Contract and Capability Governance

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for tool contract and capability governance.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

- approved control design or operating contract with accountable owner
- versioned configuration, schema, policy, model, prompt, dataset, or architecture record
- reproducible test plan with expected and observed results
- traceable decision, exception, approval, and remediation records
- tool-call and external-effect receipts
- failure-injection, idempotency, rollback, and post-change validation results

Required metrics

- applicable control coverage
- critical-failure count
- evidence freshness
- open nonconformities

Score anchors

0	Not implemented: Tool Contract and Capability Governance is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for tool contract and capability governance.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C06

Secret Isolation and Credential Brokering

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain secret isolation and credential brokering for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that secret isolation and credential brokering is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for secret isolation and credential brokering covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C06

Secret Isolation and Credential Brokering

Family: Runtime safety and isolation | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for secret isolation and credential brokering.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Secret Isolation and Credential Brokering is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for secret isolation and credential brokering.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C07

Durability and Crash Recovery

Family: Runtime safety and isolation | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain durability and crash recovery for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Recovery must cover safe stop, checkpoint integrity, rollback or compensation, credential revocation, artifact restoration, evidence preservation, requalification, and controlled retirement. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that durability and crash recovery is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for durability and crash recovery covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Recovery must cover safe stop, checkpoint integrity, rollback or compensation, credential revocation, artifact restoration, evidence preservation, requalification, and controlled retirement.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C07

Durability and Crash Recovery

Family: Runtime safety and isolation | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for durability and crash recovery.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

recovery success rate

duplicate-effect rate

time to safe state

residual-effect count

evidence preservation rate

Score anchors

0	Not implemented: Durability and Crash Recovery is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for durability and crash recovery.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C08

Checkpoint, Replay, and Side-Effect Safety

Family: Durability and recovery | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain checkpoint, replay, and side-effect safety for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The control must test realistic adversaries, direct and indirect manipulation, multi-turn escalation, cross-component attacks, prohibited outcomes, control bypass, and safe containment. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that checkpoint, replay, and side-effect safety is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for checkpoint, replay, and side-effect safety covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The control must test realistic adversaries, direct and indirect manipulation, multi-turn escalation, cross-component attacks, prohibited outcomes, control bypass, and safe containment.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C08

Checkpoint, Replay, and Side-Effect Safety

Family: Durability and recovery | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for checkpoint, replay, and side-effect safety.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

tool-call and external-effect receipts

failure-injection, idempotency, rollback, and post-change validation results

Required metrics

attack success rate

critical control bypass rate

safe abstention or containment rate

time to detection

retest closure rate

Score anchors

0	Not implemented: Checkpoint, Replay, and Side-Effect Safety is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for checkpoint, replay, and side-effect safety.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C09

Human Approval and Interruption

Family: Durability and recovery | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain human approval and interruption for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that human approval and interruption is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for human approval and interruption covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Authority must be enforceable outside the model through stable identities, least privilege, scoped credentials, separation of duties, revocation, and auditable approval.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C09

Human Approval and Interruption

Family: Durability and recovery | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for human approval and interruption.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Human Approval and Interruption is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for human approval and interruption.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C10

Memory and Context Boundaries

Family: Durability and recovery | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain memory and context boundaries for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Context state must preserve authority, chronology, source identity, permissions, unresolved work, decisions, evidence, cache identity, and restoration fidelity under compression or handoff. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that memory and context boundaries is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for memory and context boundaries covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Context state must preserve authority, chronology, source identity, permissions, unresolved work, decisions, evidence, cache identity, and restoration fidelity under compression or handoff.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C10

Memory and Context Boundaries

Family: Durability and recovery | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for memory and context boundaries.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

context assembly and token accounting trace

checkpoint, summary-fidelity, stale-state, and restoration tests

Required metrics

decision and evidence retention

stale-context rate

tokens per verified outcome

restore fidelity

cross-tenant leakage rate

Score anchors

0	Not implemented: Memory and Context Boundaries is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for memory and context boundaries.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C11

Multi-Agent Delegation and Attenuation

Family: Evaluation quality | Weight: 6 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain multi-agent delegation and attenuation for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that multi-agent delegation and attenuation is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for multi-agent delegation and attenuation covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C11

Multi-Agent Delegation and Attenuation

Family: Evaluation quality | Weight: 6 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for multi-agent delegation and attenuation.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

- approved control design or operating contract with accountable owner
- versioned configuration, schema, policy, model, prompt, dataset, or architecture record
- reproducible test plan with expected and observed results
- traceable decision, exception, approval, and remediation records
- negative, boundary, degraded, and recovery test results
- operational telemetry and current-state verification

Required metrics

- applicable control coverage
- critical-failure count
- evidence freshness
- open nonconformities

Score anchors

0	Not implemented: Multi-Agent Delegation and Attenuation is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for multi-agent delegation and attenuation.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C12

Sandboxing and Execution Isolation

Family: Evaluation quality | Weight: 6 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain sandboxing and execution isolation for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Security must isolate tenants, files, processes, networks, secrets, models, caches, tools, and external effects according to an explicit threat model and negative tests. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that sandboxing and execution isolation is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for sandboxing and execution isolation covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Security must isolate tenants, files, processes, networks, secrets, models, caches, tools, and external effects according to an explicit threat model and negative tests.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C12

Sandboxing and Execution Isolation

Family: Evaluation quality | Weight: 6 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for sandboxing and execution isolation.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Sandboxing and Execution Isolation is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for sandboxing and execution isolation.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C13

Observability and Evidence

Family: Evaluation quality | Weight: 6 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain observability and evidence for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Evidence must correlate system identity, version, configuration, inputs, outputs, actions, approvals, metrics, artifacts, findings, exceptions, and final outcomes with integrity protection. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that observability and evidence is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for observability and evidence covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Evidence must correlate system identity, version, configuration, inputs, outputs, actions, approvals, metrics, artifacts, findings, exceptions, and final outcomes with integrity protection.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C13

Observability and Evidence

Family: Evaluation quality | Weight: 6 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for observability and evidence.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Observability and Evidence is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for observability and evidence.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C14

Evaluation Harness and Reproducibility

Family: Interoperability and portability | Weight: 5 | Critical: YES

Normative requirement

The organization **MUST** establish, implement, verify, and maintain evaluation harness and reproducibility for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Evaluation must use representative and hidden cases, reproducible configuration, deterministic validators where possible, statistical uncertainty, critical-failure gates, and slice-level reporting. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that evaluation harness and reproducibility is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for evaluation harness and reproducibility covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Evaluation must use representative and hidden cases, reproducible configuration, deterministic validators where possible, statistical uncertainty, critical-failure gates, and slice-level reporting.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C14

Evaluation Harness and Reproducibility

Family: Interoperability and portability | Weight: 5 | Critical: YES

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for evaluation harness and reproducibility.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

- approved control design or operating contract with accountable owner
- versioned configuration, schema, policy, model, prompt, dataset, or architecture record
- reproducible test plan with expected and observed results
- traceable decision, exception, approval, and remediation records
- negative, boundary, degraded, and recovery test results
- operational telemetry and current-state verification

Required metrics

- task success with confidence interval
- critical-failure rate
- slice variance
- reproducibility delta
- human-review burden

Score anchors

0	Not implemented: Evaluation Harness and Reproducibility is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for evaluation harness and reproducibility.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C15

Interoperability and Protocol Support

Family: Interoperability and portability | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain interoperability and protocol support for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Capabilities must use typed contracts, bounded side effects, explicit authorization, schema validation, progress and cancellation, result evidence, compatibility tests, and revocation. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that interoperability and protocol support is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for interoperability and protocol support covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Capabilities must use typed contracts, bounded side effects, explicit authorization, schema validation, progress and cancellation, result evidence, compatibility tests, and revocation.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C15

Interoperability and Protocol Support

Family: Interoperability and portability | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for interoperability and protocol support.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Interoperability and Protocol Support is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for interoperability and protocol support.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C16

Operational Scaling and Cost Controls

Family: Operations and economics | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain operational scaling and cost controls for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Measurements must disclose workload distribution, hardware, concurrency, warmup, queueing, tail latency, successful throughput, token or media use, cost, and overload behavior. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that operational scaling and cost controls is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for operational scaling and cost controls covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Measurements must disclose workload distribution, hardware, concurrency, warmup, queueing, tail latency, successful throughput, token or media use, cost, and overload behavior.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C16

Operational Scaling and Cost Controls

Family: Operations and economics | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for operational scaling and cost controls.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

raw benchmark telemetry with workload and hardware disclosure

tail-latency, throughput, saturation, cost, and resource report

Required metrics

p50/p95/p99 end-to-end latency

successful units per second

saturation and rejection rate

cost per verified outcome

resource efficiency

Score anchors

0	Not implemented: Operational Scaling and Cost Controls is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for operational scaling and cost controls.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C17

Project Health, Licensing, and Exit Strategy

Family: Operations and economics | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain project health, licensing, and exit strategy for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that project health, licensing, and exit strategy is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for project health, licensing, and exit strategy covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The control must identify accountable ownership, authoritative inputs, expected outputs, operating limits, dependencies, failure behavior, evidence, and reassessment triggers.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C17

Project Health, Licensing, and Exit Strategy

Family: Operations and economics | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for project health, licensing, and exit strategy.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

- approved control design or operating contract with accountable owner
- versioned configuration, schema, policy, model, prompt, dataset, or architecture record
- reproducible test plan with expected and observed results
- traceable decision, exception, approval, and remediation records
- negative, boundary, degraded, and recovery test results
- operational telemetry and current-state verification

Required metrics

- applicable control coverage
- critical-failure count
- evidence freshness
- open nonconformities

Score anchors

0	Not implemented: Project Health, Licensing, and Exit Strategy is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for project health, licensing, and exit strategy.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C18

Decision Record and Reassessment

Family: Lifecycle and exit | Weight: 5 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain decision record and reassessment for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. The control must define production indicators, drift thresholds, incident triggers, review frequency, change detection, owner notification, corrective action, and retirement criteria. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that decision record and reassessment is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for decision record and reassessment covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

The control must define production indicators, drift thresholds, incident triggers, review frequency, change detection, owner notification, corrective action, and retirement criteria.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C18

Decision Record and Reassessment

Family: Lifecycle and exit | Weight: 5 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for decision record and reassessment.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

applicable control coverage

critical-failure count

evidence freshness

open nonconformities

Score anchors

0	Not implemented: Decision Record and Reassessment is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for decision record and reassessment.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

MYTHOS-AI-0001-C19

Retirement and Evidence Preservation

Family: Lifecycle and exit | Weight: 4 | Critical: NO

Normative requirement

The organization **MUST** establish, implement, verify, and maintain retirement and evidence preservation for all in-scope agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems. Recovery must cover safe stop, checkpoint integrity, rollback or compensation, credential revocation, artifact restoration, evidence preservation, requalification, and controlled retirement. The control **MUST** define acceptance criteria, prohibited outcomes, evidence, exception authority, and reassessment triggers. Where automated enforcement is feasible, the organization **SHOULD** enforce the requirement outside the model or agent. Any residual manual dependency **MUST** be documented, assigned, time-bounded, and tested.

Control objective

Objective

Ensure that retirement and evidence preservation is explicit, bounded, testable, observable, recoverable, and governed throughout the lifecycle.

Implementation requirements

Establish a versioned control contract for retirement and evidence preservation covering agent frameworks, harnesses, orchestrators, tool runtimes, memory services, multi-agent coordination, and human approval systems and name the accountable owner, approver, assessor, and affected stakeholders.

Recovery must cover safe stop, checkpoint integrity, rollback or compensation, credential revocation, artifact restoration, evidence preservation, requalification, and controlled retirement.

Implement the control through machine-enforceable policy, typed schemas, isolated runtime boundaries, validated procedures, or an approved combination appropriate to the risk.

Define explicit nominal, negative, adversarial, malformed, degraded, overloaded, recovery, and rollback tests; critical cases must be hidden from the system under evaluation where feasible.

Correlate evidence to stable system, model, data, prompt, repository, tool, environment, user or tenant, and release identities; protect records against unauthorized modification.

Set review cadence and event-driven reassessment triggers for material changes, incidents, supplier updates, drift, failed controls, new affected populations, and retirement.

Applicability

Applicable unless a documented, approved, evidence-supported exclusion demonstrates that the capability, data, decision, interface, population, or lifecycle stage is absent.

MYTHOS-AI-0001-C19

Retirement and Evidence Preservation

Family: Lifecycle and exit | Weight: 4 | Critical: NO

Assessment procedure

Examine the approved design, applicability decision, responsible roles, and current implementation for retirement and evidence preservation.

Select representative normal, boundary, high-impact, and adversarial cases, including at least one prohibited outcome and one dependency or recovery failure.

Verify the exact model, data, prompt, tool, policy, runtime, repository, environment, and evaluator versions used during assessment.

Execute deterministic validators first, then calibrated model or human judgments only where deterministic proof is not available.

Reconcile expected behavior with raw outputs, trajectories, tool effects, telemetry, artifacts, user-visible outcomes, and unresolved contradictions.

Assign a 0-5 score, record evidence links and confidence, open nonconformities, define remediation ownership, and set the next verification date.

Minimum evidence

approved control design or operating contract with accountable owner

versioned configuration, schema, policy, model, prompt, dataset, or architecture record

reproducible test plan with expected and observed results

traceable decision, exception, approval, and remediation records

negative, boundary, degraded, and recovery test results

operational telemetry and current-state verification

Required metrics

recovery success rate

duplicate-effect rate

time to safe state

residual-effect count

evidence preservation rate

Score anchors

0	Not implemented: Retirement and Evidence Preservation is absent, materially unknown, or contradicted by evidence.
1	Initial: intent is documented, but ownership, repeatability, implementation, or evidence is incomplete.
2	Defined: approved procedure and design exist, but implementation or representative testing is partial.
3	Implemented: control operates for the declared scope and passes normal and basic negative tests.
4	Verified: independent assessment, hidden or adversarial cases, current evidence, and recovery validation demonstrate effectiveness.
5	Continuously assured: automated monitoring and regression, current evidence, incident learning, and timely reassessment sustain verified effectiveness.

Failure indicators

The organization cannot identify an authoritative owner, current scope, or complete implementation for retirement and evidence preservation.

Only a successful demonstration, vendor claim, aggregate score, or model-generated judgment is available; raw evidence and negative tests are absent.

Critical cases can be hidden by averages, unsupported N/A designations, stale evidence, or changes in model, data, prompt, tool, policy, or environment.

Failure, rollback, revocation, deletion, containment, or retirement behavior has not been exercised under realistic conditions.

Required Benchmark Scenarios

MYTHOS-AI-0001-B01 - Representative production task

Demonstrate the declared use case using current configuration and representative inputs. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B02 - Hidden critical holdout

Execute high-impact cases withheld from development, tuning, and prompt iteration. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B03 - Malformed and adversarial inputs

Test schema violations, ambiguity, direct or indirect manipulation, and unsafe instructions. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B04 - Long-horizon or multi-step execution

Measure compounding error, context loss, looping, premature completion, and recovery. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B05 - Dependency and tool failure

Inject model, network, data, tool, credential, evaluator, or service failure. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

Required Benchmark Scenarios - Continued

MYTHOS-AI-0001-B06 - Resource pressure and overload

Exercise concurrency, long inputs, queueing, rate limits, memory, tokens, and cost limits. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B07 - Privacy and cross-boundary test

Attempt cross-tenant, secret, personal-data, restricted-source, cache, and egress violations. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B08 - Rollback, revocation, or restore

Prove safe stop, compensation, revocation, prior-state restoration, and residual-effect reconciliation. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B09 - Human intervention and disagreement

Test approval, interruption, correction, appeal, assessor disagreement, and minority findings. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

MYTHOS-AI-0001-B10 - Material change and regression

Change model, prompt, dataset, tool, provider, runtime, or policy and run requalification gates. Apply the scenario specifically to agentic framework evaluation and preserve raw evidence.

Conformance Report and Evidence Bundle

Permanent document ID, standard version, assessment version, system and use-case scope.

System, model, provider, prompt, data, retrieval, memory, tool, evaluator, runtime, repository, and infrastructure identities.

Applicability decisions and normalized denominator.

Control-level scores, weights, critical status, evidence links, assessor, date, confidence, and finding disposition.

Benchmark scenario results with raw artifacts, deterministic validators, judge or expert records, uncertainty, failures, and limitations.

Family and total score, achieved conformance level, unresolved critical or major findings, approved exceptions, expiry, and next review.

Release, rollback, revocation, deletion, and retirement evidence where applicable.

Evidence bundle integrity

The evidence bundle SHOULD use content hashes, signatures or protected repository history, stable artifact IDs, access controls, retention policy, and independent verification. Evidence links alone are insufficient when the referenced object can change without detection.

Exceptions, Waivers, and Compensating Controls

An exception **MUST NOT** be used to relabel a failed requirement as conforming. Exceptions document accepted residual risk for a bounded scope and time; the underlying control score remains evidence-based.

Identify the exact control, system scope, reason, affected users or assets, and current score.

Describe the missing requirement, residual risk, foreseeable failure, and affected decisions.

Define compensating controls, validation evidence, monitoring, owner, approver, start date, expiry, and remediation plan.

Obtain approval from an authority independent of the implementation owner for critical or high-impact exceptions.

Reassess on incident, material change, evidence expiry, control failure, or exception expiration.

Publish exceptions in the conformance report; hidden exceptions invalidate the claim.

Critical exception cap

An unresolved exception against a critical control prevents Assured or Continuously Assured conformance and may prevent Operational conformance when the control score is below 3.

Source Authority and Freshness Register

NIST - Artificial Intelligence Risk Management Framework 1.0

Cross-sector AI risk governance, mapping, measurement, and management.

<https://doi.org/10.6028/NIST.AI.100-1>

NIST - NIST AI 600-1 - Generative Artificial Intelligence Profile

Generative-AI risks, controls, evaluation, and lifecycle actions.

<https://doi.org/10.6028/NIST.AI.600-1>

NIST - AI Resource Center and AI RMF Playbook

Current implementation resources and revision notices for the AI RMF.

<https://airc.nist.gov/>

ISO/IEC - ISO/IEC 42001:2023 - AI management systems

AI management-system requirements and continual improvement.

<https://www.iso.org/standard/42001>

ISO/IEC - ISO/IEC 23894:2023 - Guidance on AI risk management

AI risk-management guidance across lifecycle and stakeholders.

<https://www.iso.org/standard/77304.html>

ISO/IEC - ISO/IEC 42005:2025 - AI system impact assessment

Impact-assessment guidance for individuals and societies.

<https://www.iso.org/standard/42005>

MLCommons - MLCommons Benchmarks

Open performance, quality, and safety benchmark programs.

<https://mlcommons.org/benchmarks/>

Source Authority and Freshness Register - Continued

MLCommons - AI Risk and Reliability

Safety-test methodology and risk-oriented evaluation work.

<https://mlcommons.org/working-groups/ai-risk-reliability/ai-risk-reliability/>

Model Context Protocol - MCP Specification 2025-11-25

Stable protocol baseline for resources, prompts, tools, authorization, tasks, and lifecycle.

<https://modelcontextprotocol.io/specification/2025-11-25>

Agent2Agent Protocol - A2A Protocol Specification 1.0

Agent discovery, communication, tasks, status, messages, and artifacts.

<https://a2a-protocol.org/v1.0.0/>

OWASP - Top 10 for Agentic Applications

Agentic-system security risks involving goals, tools, memory, identity, and autonomy.

<https://genai.owasp.org/>

Freshness rule

Sources were verified for this candidate edition on 2026-07-26. The standard **MUST** be revalidated when a cited authority changes, becomes superseded, is withdrawn, or materially alters the control interpretation.

Limitations, Revision History, and Adoption Position

This candidate Mythos standard is designed for engineering governance and assessment. It is not a legal opinion, regulatory certification, safety guarantee, or substitute for domain-specific professional judgment. Model behavior remains probabilistic and system risk depends on data, users, tools, deployment, incentives, and operating context.

Revision history

Version	Date	Change
2.0.0-candidate	2026-07-26	Expanded normative edition with 19 weighted controls, benchmark scenarios, conformance levels, machine-readable catalogs, assessment workbook integration, source freshness, and explicit lineage.
1.0.0-candidate	2026-07-24	Earlier permanent-identity candidate retained in the release lineage archive.

Adoption position

Use this edition as a candidate control and benchmark baseline. Organizations SHOULD pilot it on bounded systems, retain assessment evidence, submit corrections, and avoid representing candidate conformance as external certification.