MYTHOS-UX-0002

Human-in-the-Loop Approval Workflow Standard

Defines approval, review, interruption, rollback, evidence, authentication, and non-repudiation requirements for consequential AI, agentic, automation, infrastructure, security, and engineering actions.

6	6	UX
EVALUATION CRITERIA	CURRENT AUTHORITY RECORDS	CANONICAL SERIES

CANDIDATE STANDARD

Mythos Systems | Engineering Standards Program | 2026-07-22

This candidate publication is complete for technical review but does not by itself establish certification, legal compliance, implementation conformance, or final approval.

Contents

Use the interactive entries below or the PDF bookmark outline to navigate the publication.

Document Control

Field	Value
Document ID	MYTHOS-UX-0002
Version	1.1.0 - Enterprise Candidate Edition
Status	Candidate Standard
Classification	Public
Publisher	Mythos Systems
Owner	Aaron Stovall
Domain	User Experience and Human-System Interaction
Initial source date	2026-07-16
Enterprise reconstruction	2026-07-22
Review cadence	Quarterly for dynamic authorities; at least annually for stable controls
Supersedes	Source draft retained in Appendix A
Publication lineage	Permanent canonical series

CANDIDATE STATUS

The source document identified itself as a definitive published standard. This enterprise edition corrects that status. It remains a candidate until current-source validation, qualified domain review, assessment engineering, accessibility review, and formal approval are recorded.

Part I - Executive Direction

1. Executive Overview

Defines approval, review, interruption, rollback, evidence, authentication, and non-repudiation requirements for consequential AI, agentic, automation, infrastructure, security, and engineering actions.

2. Principal Findings

- Approval must bind to an immutable and exact plan, not a vague natural-language request.
- The user must see target, scope, risk, blast radius, data movement, secrets, tests, rollback, and expiration before authorizing.
- Execution authority must be separated from planning and presentation.
- Cancellation and safe interruption are required capabilities, not optional interface features.

3. Required Leadership Decisions

- Adopt risk-tiered approval profiles and step-up authentication.
- Require immutable proposal hashes and invalidate approval when material content changes.
- Pre-authorize rollback and define indeterminate-outcome handling before execution.
- Retain approval, execution, verification, and revocation evidence.

4. Enterprise Target State

TARGET OPERATING MODEL

Organizations applying MYTHOS-UX-0002 operate the subject as a governed lifecycle: inventory and classify; establish authoritative policy and architecture; enforce controls technically; observe actual state; verify outcomes independently; retain evidence; manage exceptions; and revalidate when authorities, platforms, risks, or operating conditions change.

5. Business, Safety, and Operational Impact

- Reduces reliance on manual evidence and undocumented expert knowledge.
- Makes risk acceptance, exceptions, and unresolved uncertainty visible to accountable leaders.
- Creates measurable implementation and assurance outcomes rather than a one-time score.
- Supports procurement, architecture review, operational readiness, and independent assessment from one source-controlled publication.

Part II - Standard Foundation

6. Purpose and Scope

Defines approval, review, interruption, rollback, evidence, authentication, and non-repudiation requirements for consequential AI, agentic, automation, infrastructure, security, and engineering actions.

6.1 Applicability

Apply this standard according to system purpose, deployment model, data classification, user population, safety impact, regulatory obligations, and organizational risk tolerance. Tailoring must be documented. A lower implementation profile cannot silently waive a mandatory legal, safety, accessibility, contractual, or security obligation.

6.2 Out of Scope

This publication does not replace legal advice, contracting interpretation, regulator guidance, platform-specific engineering, human-factors testing, safety certification, or implementation evidence. Companion Mythos standards remain applicable where their domains intersect.

7. Requirements Language and Conformance

The terms MUST, MUST NOT, SHOULD, SHOULD NOT, and MAY are normative only when written in uppercase. Conformance requires applicable mandatory criteria to be implemented and supported by current evidence. A weighted score is informative and cannot compensate for a failed mandatory gate.

State	Meaning
Conformant	All applicable mandatory requirements are satisfied with sufficient current evidence.
Partially Conformant	Some applicable requirements are implemented, but material gaps remain.
Not Conformant	One or more mandatory requirements are not satisfied.
Compensating Control Accepted	A formally approved alternative achieves the required outcome.
Exception Active	A time-bounded, accountable risk acceptance exists.
Not Assessed	Evidence is insufficient to reach a conclusion.

8. Assurance and Maturity

Level	Description
M0 Unmanaged	The outcome is not intentionally controlled.
M1 Documented	Responsibilities and procedures exist but are inconsistent or manual.
M2 Implemented	The control operates in the applicable environment.
M3 Measured	Performance, exceptions, and effectiveness are measured.
M4 Assured	Independent testing and continuous evidence support the control.
M5 Adaptive	The control responds safely to changes in risk, environment, and evidence.

Part III - Risk and Architecture

9. Risk Model

- Authority drift: external requirements, specifications, programs, and platform behavior change faster than the publication is reviewed.
- Control illusion: documentation or visual state indicates compliance while runtime behavior differs.
- Automation overreach: a generated plan or score is treated as authorization or proof.
- Evidence gaps: conclusions cannot be reconstructed from source, configuration, event, test, and approval records.
- Human factors: users cannot understand, interrupt, challenge, or safely recover from consequential behavior.
- Dependency concentration: a provider, runtime, device, framework, or external authority becomes a hidden single point of failure.

10. Reference Control Architecture

ARCHITECTURE SEPARATION

Source authority defines applicable obligations and constraints. Human and organizational governance establishes accountability. Typed policy and architecture encode the target state. Runtime enforcement controls behavior. Independent testing verifies outcomes. Evidence systems preserve what was proposed, approved, executed, observed, and verified.

11. Roles and Accountability

Role	Accountability
Executive sponsor	Approves risk posture, funding, and unresolved material exceptions.
Domain authority	Owns interpretation, architecture, and control applicability.
Engineering owner	Implements controls and operational lifecycle.
Security/privacy/accessibility/safety reviewer	Challenges design and verifies domain-specific obligations.
Independent assessor	Evaluates evidence and tests without relying only on implementer assertions.
Publication owner	Maintains version, sources, traceability, expiration, and change record.

Part IV - Normative Evaluation and Control System

12. Evaluation Doctrine

The preserved source uses a weighted 0-5 evaluation model. This edition retains that material but corrects its interpretation. Score 5 means optimized, strongly evidenced, and independently assured within the assessed scope. It never means perfect, mathematically infallible, universally compliant, or permanently secure.

Score	Enterprise interpretation
0	Absent, unsafe, or unable to perform the required outcome.
1	Initial or ad hoc capability with substantial manual dependence.
2	Repeatable partial implementation with material gaps.
3	Production baseline: implemented, governed, and evidenced.
4	Advanced: automated, measured, resilient, and independently tested.
5	Assured: optimized for the defined profile, continuously evidenced, and subject to recurring independent validation.

MANDATORY GATE

An organization cannot average away a failed mandatory criterion. Applicability, safety, legal requirements, accessibility, security boundaries, authorization, and evidence sufficiency must be evaluated before weighted scoring is used.

13. Criterion Index

ID	Criterion	Weight	Primary outcome
MYTHOS-UX-0002-C01	Typed Contracts and Blast Radius	20%	Present a machine-validated action contract containing subject, operation, targets, environment, data, secrets, risk, dependencies, blast radius, tests, and expiry.
MYTHOS-UX-0002-C02	Plan-and-Apply Binding	20%	Cryptographically or content-hash bind authorization to the exact executable plan and prevent substitution between review and execution.
MYTHOS-UX-0002-C03	Step-Up Authentication	15%	Match authentication assurance and approver independence to risk, environment, irreversibility, and target criticality.
MYTHOS-UX-0002-C04	Pre-Authorized Rollback	15%	Define, test, scope, and authorize rollback or compensation before the forward change when safe reversal is material to approval.
MYTHOS-UX-0002-C05	Evidence and Non-Repudiation	15%	Record immutable proposal, identity, authentication, approval, grant, execution, observation, verification, revocation, and exception evidence.
MYTHOS-UX-0002-C06	Interruption and Cancellation	15%	Define cancellable states, safe points, cleanup, compensation, remote-worker termination, and explicit indeterminate outcomes.

MYTHOS-UX-0002-C01 - Typed Contracts and Blast Radius

NORMATIVE REQUIREMENT

The organization MUST present a machine-validated action contract containing subject, operation, targets, environment, data, secrets, risk, dependencies, blast radius, tests, and expiry.

Objective and Risk Addressed

Present a machine-validated action contract containing subject, operation, targets, environment, data, secrets, risk, dependencies, blast radius, tests, and expiry.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.
- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.
- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- proposal schema
- Gaia impact report
- risk decision
- target list
- data movement
- tool and secret references.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.
- Modify target, operation, or data after review and verify approval becomes invalid; compare displayed and executed contract.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
approval invalidation accuracy	Track trend, threshold, exception, and accountable response.
target-scope errors	Track trend, threshold, exception, and accountable response.
missing-impact fields	Track trend, threshold, exception, and accountable response.
contract validation failures	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	Free-text description of the action ("I'm going to update the DB")
1	Structured payload (Action, Target), but no blast radius

Score	Preserved source criterion
2	Basic dependency graph showing affected resources
3	Typed Approval Contract with exact state diff (before/after) and blast radius (users/systems impacted)
4	Real-time simulation of the action in a digital twin to verify the diff
5	Perfect context: formally verified blast radius bounds, cryptographic state diffs, zero hidden side-effects

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

MYTHOS-UX-0002-C02 - Plan-and-Apply Binding

NORMATIVE REQUIREMENT

The organization MUST cryptographically or content-hash bind authorization to the exact executable plan and prevent substitution between review and execution.

Objective and Risk Addressed

Cryptographically or content-hash bind authorization to the exact executable plan and prevent substitution between review and execution.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.
- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.
- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- plan hash
- approval record
- execution input hash
- change history
- rejection events.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.

- Attempt plan mutation, target substitution, replay, expiry, and alternate-tool execution; verify fail-closed behavior.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
hash mismatch blocks	Track trend, threshold, exception, and accountable response.
stale approval rate	Track trend, threshold, exception, and accountable response.
unbound execution attempts	Track trend, threshold, exception, and accountable response.
replay prevention	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	Agent is trusted to execute the "spirit" of the approval
1	Action ID approved, but parameters can be mutated
2	Parameters locked at approval, but environmental state can change
3	Cryptographic hash (BLAKE3) of the exact execution plan is bound to the approval
4	Execution engine verifies the hash against the live environment state before every step
5	Perfect binding: formally verified plan-apply equivalence, zero capability for post-approval mutation, cryptographic proof of execution matching

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

MYTHOS-UX-0002-C03 - Step-Up Authentication

NORMATIVE REQUIREMENT

The organization MUST match authentication assurance and approver independence to risk, environment, irreversibility, and target criticality.

Objective and Risk Addressed

Match authentication assurance and approver independence to risk, environment, irreversibility, and target criticality.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.

- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.
- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- risk tiers
- authenticator policy
- approver separation
- authentication logs
- emergency procedure.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.
- Test ordinary, elevated, high-assurance, second-approver, and break-glass paths including recovery and revocation.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
step-up success	Track trend, threshold, exception, and accountable response.
inappropriate bypass	Track trend, threshold, exception, and accountable response.
standing approver conflicts	Track trend, threshold, exception, and accountable response.
break-glass review time	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	Relies on existing session token (vulnerable to session hijacking)
1	Re-entry of password
2	OTP (SMS/Email)
3	Phishing-resistant MFA (FIDO2 / WebAuthn passkeys)
4	Hardware-backed step-up auth bound to the specific transaction hash (transaction signing)
5	Perfect assurance: formally verified identity at time of approval, zero replay capability, hardware-bound non-repudiation

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

MYTHOS-UX-0002-C04 - Pre-Authorized Rollback

NORMATIVE REQUIREMENT

The organization MUST define, test, scope, and authorize rollback or compensation before the forward change when safe reversal is material to approval.

Objective and Risk Addressed

Define, test, scope, and authorize rollback or compensation before the forward change when safe reversal is material to approval.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.
- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.
- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- rollback plan
- prechecks
- simulation
- state backup
- rollback grant
- validation results.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.
- Cause controlled forward failure and verify rollback can execute without seeking broader authority, then validate resulting state.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
rollback test age	Track trend, threshold, exception, and accountable response.
rollback success	Track trend, threshold, exception, and accountable response.
recovery time	Track trend, threshold, exception, and accountable response.
irreversible actions without acceptance	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	No rollback plan; system requires manual intervention on failure
1	Agent proposes a rollback, but requires a second human approval
2	Basic undo function, but not deterministic (state may have drifted)
3	Deterministic rollback plan included in the Typed Approval Contract
4	Rollback plan is pre-authorized and executes automatically if success criteria are not met
5	Perfect remediation: formally verified compensating transactions, zero state drift on rollback, mathematically proven return to safe state

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

MYTHOS-UX-0002-C05 - Evidence and Non-Repudiation

NORMATIVE REQUIREMENT

The organization MUST record immutable proposal, identity, authentication, approval, grant, execution, observation, verification, revocation, and exception evidence.

Objective and Risk Addressed

Record immutable proposal, identity, authentication, approval, grant, execution, observation, verification, revocation, and exception evidence.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.
- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.

- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- Clio timeline
- signed/hash-linked records
- approval artifact
- tool logs
- verifier results
- retention policy.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.
- Reconstruct a selected decision from evidence and verify the approver saw the same immutable content that executed.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
evidence completeness	Track trend, threshold, exception, and accountable response.
unresolved chain gaps	Track trend, threshold, exception, and accountable response.
time synchronization error	Track trend, threshold, exception, and accountable response.
retention compliance	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	Database log entry ("User X approved at Time Y")
1	Append-only audit log
2	Log entry includes the user's session ID
3	Approval signed by the user's asymmetric key (Ed25519)
4	Signed approval, execution hash, and blast radius bundled into a tamper-evident Evidence Bundle
5	Perfect evidence: integration with transparency logs (Rekor), formally verified non-repudiation, zero ability to disclaim an action

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

MYTHOS-UX-0002-C06 - Interruption and Cancellation

NORMATIVE REQUIREMENT

The organization MUST define cancellable states, safe points, cleanup, compensation, remote-worker termination, and explicit indeterminate outcomes.

Objective and Risk Addressed

Define cancellable states, safe points, cleanup, compensation, remote-worker termination, and explicit indeterminate outcomes.

Applicability

Apply this criterion wherever the evaluated capability, data, workflow, interface, user, environment, provider, or authority is material. Document exclusions and any compensating control.

Minimum Acceptable Implementation

A production baseline corresponds to source score 3: the capability is deliberately implemented, technically enforced where practical, owned, monitored, and supported by current evidence. Source scores 4 and 5 represent increasing automation, resilience, continuous assurance, and independent validation.

Implementation Direction

- Define authoritative state, owner, interfaces, dependencies, lifecycle, and failure behavior.
- Encode enforceable policy and typed contracts instead of relying only on prose or visual convention.
- Provide safe fallback, interruption, exception, recovery, and decommissioning behavior.
- Collect evidence at the point of action and verify the observed result independently.
- Revalidate after material source, platform, model, device, provider, architecture, or risk change.

Required Evidence

- state machine
- cancellation contract
- cleanup logs
- worker heartbeat
- kill and revoke tests.

Assessment and Test Procedure

- Confirm applicability, ownership, current architecture, and approved policy.
- Cancel before dispatch, during execution, during verification, and during network partition; verify state and evidence.
- Inspect failures, exceptions, and negative tests rather than reviewing only successful examples.
- Rate evidence confidence and record untested conditions, unsupported platforms, and unresolved uncertainty.

Metrics and Assurance Indicators

Metric	Expected use
cancellation latency	Track trend, threshold, exception, and accountable response.
orphaned process count	Track trend, threshold, exception, and accountable response.
indeterminate resolution time	Track trend, threshold, exception, and accountable response.
cleanup failures	Track trend, threshold, exception, and accountable response.

Preserved Source Scoring Anchors

The following anchors are retained from the source draft. Absolute words such as “perfect” are historical wording and are normalized by the enterprise interpretation above.

Score	Preserved source criterion
0	Once approved, the action runs to completion regardless of changing conditions

Score	Preserved source criterion
1	Basic "Cancel" button that sends a request to the agent
2	Hard stop / process kill (leaves system in a zombie state)
3	Graceful cancellation contract: requested -> acknowledged -> stopping -> stopped
4	Mid-execution pause, state inspection, and resumption or safe rollback
5	Perfect control: formally verified cancellation states, zero-latency kill switch, preservation of all evidence up to the point of interruption

Exceptions and Compensating Controls

Any exception must identify the unmet outcome, affected scope, owner, risk, compensating control, evidence, expiration, and remediation plan. Exceptions do not change the underlying requirement.

Part V - Implementation and Operations

14. Implementation Profiles

Profile	Required posture
Baseline	Documented ownership, minimum production controls, evidence, and exception process.
Enterprise	Central policy, automation, integration, continuous monitoring, and independent review.
High Assurance	Strong isolation, high-assurance identity, formal or adversarial verification, short evidence windows, and two-person governance where material.
Sovereign or Air-Gapped	Local control planes, approved dependencies, offline update and evidence workflows, restricted egress, and explicit supply-chain custody.

15. Operational Lifecycle

Stage	Required activities
Discover and classify	Inventory scope, authority, data, users, dependencies, risks, and current evidence.
Design	Define target state, architecture, policies, tests, metrics, and ownership.
Implement	Deploy controls through reviewed changes with rollback and evidence.
Operate	Monitor state, exceptions, incidents, drift, performance, and provider changes.
Verify	Perform recurring independent assessment and negative testing.
Change	Reassess applicability and invalidate stale evidence after material changes.
Retire	Revoke authority, remove data and credentials, preserve required evidence, and update the registry.

16. Anti-Patterns and Prohibited Practices

- Declaring conformance from a document review without implementation evidence.
- Using one weighted score to override a failed mandatory requirement.
- Treating model output, interface state, scanner output, or tool success as independent verification.
- Using stale or unversioned external authorities for current decisions.
- Hiding exceptions, unsupported platforms, incomplete testing, or low-confidence findings.
- Hard-coding a single provider, device, framework, or vendor as a universal requirement unless the publication is explicitly vendor-scoped.

Part VI - Assurance, Evidence, and Traceability

17. Evidence Model

Evidence must identify subject, scope, source, version, time, actor, method, result, integrity, retention, and relationship to the assessed criterion. Screenshots and generated summaries are supporting artifacts, not sufficient evidence by themselves.

18. Assessment Confidence

Confidence	Basis
Low	Self-assertion, incomplete samples, stale evidence, or untested material conditions.
Moderate	Current documentation and representative evidence with limited independent testing.
High	Current direct evidence and repeatable independent testing across material conditions.
Very High	Sustained evidence, broad negative testing, independent review, and verified operation over time.

19. Current Authority Register

Authority	Status and scope	Valid until
NIST AI Risk Management Framework 1.0	Current framework; revision activity underway - Human oversight and AI risk governance	2026-10-20
ISO/IEC 42001:2023	Published international standard - Governed AI management and accountability	2027-01-18
Regulation (EU) 2024/1689 - EU AI Act	In force with phased applicability - Human oversight for applicable high-risk AI systems	2026-10-20
NIST SP 800-63-4	Final - Digital identity assurance and authentication	2027-01-18
NIST SP 800-63B-4	Final - Authentication and authenticator management	2027-01-18
WCAG 2.2	W3C Recommendation - Accessible approval and interaction workflows	2027-01-18

AUTHORITY LIMITATION

The register identifies current technical and governance authorities relevant to this candidate standard. It does not establish legal applicability, contractual compliance, certification, or clause-level equivalence. Dynamic sources must be refreshed before material decisions.

20. Publication Gate

- Qualified domain technical review completed.
- Current authorities and versions independently confirmed.
- Criterion requirements, evidence, and tests reviewed for measurability.
- Legal, contracting, regulatory, safety, privacy, and accessibility applicability reviewed where relevant.
- Machine-readable criterion catalog validated.
- PDF accessibility and visual quality reviewed.
- Formal approval, effective date, and review triggers recorded.

Appendix A - Complete Preserved Source Draft

The following source draft is preserved in full for completeness and traceability. Conversational framing, “definitive” status, universal claims, obsolete program assumptions, and “perfect” score language are historical source content and are not controlling statements in this enterprise candidate edition.

Document ID: MYTHOS-UX-0002 Version: 1.0.0 (Definitive Registry Edition) Status: Published Standard Classification: Public Organization: Mythos Systems (MYTHOS AI) Owner: Aaron Stovall Effective Date: 2026-07-16 Review Cadence: Quarterly, and immediately after a material shift in IAM step-up authentication, zero-trust policy engines, or agentic workflow paradigms Applies To: All organizations evaluating, selecting, or operating autonomous agent runtimes, CI/CD approval gates, privileged access management (PAM) systems, and human-in-the-loop workflow engines for production or enterprise use Companion Standards: MYTHOS-AI-0001 (Agentic Frameworks), MYTHOS-UX-0001 (AI-Native Interfaces), MYTHOS-SEC-0003 (Zero-Trust & Capability Grants), MYTHOS-DAT-0005 (Tamper-Evident Ledgers)

0. Executive Mandate

The architecture of human-in-the-loop (HITL) intervention has failed.

Organizations grant autonomous agents the authority to execute destructive infrastructure changes or financial transactions, pausing only to send a Slack message asking, "Can I do this?" with an "Approve" button. This is not governance; it is accountability theater. The human has no context, the approval is detatched from the execution, and the rollback plan is nonexistent.

This standard exists because:

1. "Yes/No" is Not an Approval Contract: A human clicking "Approve" on a text description of an action provides zero cryptographic guarantee that the executed action matches the approved action. Approvals MUST be bound to a deterministic hash of the exact execution plan (Plan-and-Apply Binding).
2. Approval Fatigue is a Security Threat: If an agent requests approval 50 times a day for routine tasks, the human will blindly click "Approve" on the 51st request—even if it is a prompt injection attempting to exfiltrate data. HITL MUST be reserved exclusively for high-risk, policy-mandated boundaries.
3. Blast Radius Must Be Visualized: Telling a user "I am going to modify the production database" is useless. The HITL interface MUST render the exact blast radius: which rows, which tables, what dependencies, and what the system state will look like after execution.
4. Approval is Not Evidence: A log entry stating "User approved action X" is trivially forged. Human approvals MUST be cryptographically signed (Ed25519) using step-up authentication (e.g., FIDO2/WebAuthn) and appended to a tamper-evident ledger.
5. Rollback Must Be Pre-Approved: An agent should not ask for permission to execute an action and then ask for permission to undo it when it fails. The HITL contract MUST include a pre-authorized, deterministic rollback plan that executes automatically upon failure.

This document establishes the Mythos HITL Standard (MHITLS), a comprehensive, evidence-based methodology for evaluating HITL architectures against plan-apply integrity, blast-radius visibility, and non-repudiable evidence generation.

Part I. Scope and Applicability

1.1 In Scope

This standard covers the evaluation of:

- Agentic HITL architectures and interruptibility protocols
- CI/CD approval gates and deployment pipelines (Argo, GitHub Approvals)

- Privileged Access Management (PAM) and Just-in-Time (JIT) access
- Plan-and-Apply binding mechanisms (Terraform, Crossplane)
- Step-up authentication for high-risk actions (FIDO2, Passkeys)
- Rollback and cancellation contract engineering

1.2 Out of Scope

- General AI-native interface rendering (covered in MYTHOS-UX-0001)
- General zero-trust policy architecture (covered in MYTHOS-SEC-0003)
- General observability and audit logging (covered in MYTHOS-DAT-0002)

1.3 Audience

- Principal engineers and architects designing autonomous workflows
- Platform engineers building CI/CD and infrastructure approval gates
- Security teams managing PAM and step-up authentication
- AI engineers building agent interruption and handoff protocols
- Standards bodies seeking a reference HITL methodology

Part II. Definitions and Taxonomy

2.1 HITL Dimensions

Dimension	Definition
Typed Approval Contract	A structured, machine-readable payload containing the proposed action, target scope, expected state, rollback plan, and blast radius, replacing free-text approval prompts.
Plan-and-Apply Binding	The cryptographic practice of hashing the proposed execution plan (the "Plan") and enforcing that the exact same plan (the "Apply") is executed, preventing post-approval mutation.
Blast Radius	The maximum potential impact (systems, data, cost, users) of a proposed action if it executes successfully or fails catastrophically.
Step-Up Authentication	The requirement for a user to provide a stronger authentication factor (e.g., biometrics, hardware key) at the moment of approval, rather than relying on the existing session token.
Pre-Authorized Rollback	A deterministic remediation plan included in the approval contract, which the system is authorized to execute automatically without further human intervention if the primary action fails.

2.2 The HITL Doctrine

SOURCE STATEMENT

A "Yes/No" prompt is not a control. An approval without context is liability. A plan that can change after approval is an attack vector. If the rollback is not pre-approved, the system is not production-ready.

Part III. Evaluation Methodology

3.1 Scoring Model

Each capability is scored from 0 to 5.

Score	Meaning
0	Fails basic task; text-based "Approve" buttons, no binding, no rollback
1	Basic CI/CD gate, but no blast radius or step-up auth
2	Functional but lacks plan-apply cryptographic binding
3	Solid production performance; typed contracts, blast radius visualization, step-up auth
4	Strong performance; plan-apply binding, pre-authorized rollback, tamper-evident approval
5	Best-in-class; sets the standard for mathematically verified, zero-trust HITL governance

Weighting

Category	Weight	Rationale
Typed Contracts & Blast Radius	20%	Without context, humans approve malicious actions
Plan-and-Apply Binding	20%	Post-approval mutation breaks the trust model
Step-Up Authentication	15%	Session tokens are insufficient for destructive actions
Pre-Authorized Rollback	15%	Failed actions must be reverted instantly
Evidence & Non-Repudiation	15%	Approvals must be cryptographically provable
Interruption & Cancellation	15%	Users must be able to halt execution mid-flight

Total: 100%

A system MUST score at least 3.0 in Typed Contracts and Plan-and-Apply Binding to be considered for production use.

Part IV. Evaluation Categories

4.1 Typed Contracts and Blast Radius (Weight: 20%)

4.1.1 Context Visualization

Does the HITL system provide deterministic context and blast radius for the proposed action?

Score	Criteria
0	Free-text description of the action ("I'm going to update the DB")
1	Structured payload (Action, Target), but no blast radius
2	Basic dependency graph showing affected resources
3	Typed Approval Contract with exact state diff (before/after) and blast radius (users/systems impacted)
4	Real-time simulation of the action in a digital twin to verify the diff
5	Perfect context: formally verified blast radius bounds, cryptographic state diffs, zero hidden side-effects

4.2 Plan-and-Apply Binding (Weight: 20%)

4.2.1 Execution Integrity

Is the system mathematically prevented from executing a different action than the one approved?

Score	Criteria
0	Agent is trusted to execute the "spirit" of the approval
1	Action ID approved, but parameters can be mutated
2	Parameters locked at approval, but environmental state can change
3	Cryptographic hash (BLAKE3) of the exact execution plan is bound to the approval
4	Execution engine verifies the hash against the live environment state before every step
5	Perfect binding: formally verified plan-apply equivalence, zero capability for post-approval mutation, cryptographic proof of execution matching

4.3 Step-Up Authentication (Weight: 15%)

4.3.1 Identity Assurance

Is the human approval backed by a strong, non-replayable authentication factor?

Score	Criteria
0	Relies on existing session token (vulnerable to session hijacking)
1	Re-entry of password
2	OTP (SMS/Email)
3	Phishing-resistant MFA (FIDO2 / WebAuthn passkeys)
4	Hardware-backed step-up auth bound to the specific transaction hash (transaction signing)
5	Perfect assurance: formally verified identity at time of approval, zero replay capability, hardware-bound non-repudiation

4.4 Pre-Authorized Rollback (Weight: 15%)

4.4.1 Failure Remediation

If the approved action fails, can the system automatically revert to the prior state?

Score	Criteria
0	No rollback plan; system requires manual intervention on failure
1	Agent proposes a rollback, but requires a second human approval
2	Basic undo function, but not deterministic (state may have drifted)
3	Deterministic rollback plan included in the Typed Approval Contract
4	Rollback plan is pre-authorized and executes automatically if success criteria are not met
5	Perfect remediation: formally verified compensating transactions, zero state drift on rollback, mathematically proven return to safe state

4.5 Evidence and Non-Repudiation (Weight: 15%)

4.5.1 Audit Integrity

Is the human approval cryptographically provable after the fact?

Score	Criteria
0	Database log entry ("User X approved at Time Y")
1	Append-only audit log
2	Log entry includes the user's session ID
3	Approval signed by the user's asymmetric key (Ed25519)
4	Signed approval, execution hash, and blast radius bundled into a tamper-evident Evidence Bundle
5	Perfect evidence: integration with transparency logs (Rekor), formally verified non-repudiation, zero ability to disclaim an action

4.6 Interruption and Cancellation (Weight: 15%)

4.6.1 Mid-Execution Control

Can a user interrupt or cancel an action after approval but during execution?

Score	Criteria
0	Once approved, the action runs to completion regardless of changing conditions
1	Basic "Cancel" button that sends a request to the agent
2	Hard stop / process kill (leaves system in a zombie state)
3	Graceful cancellation contract: requested -> acknowledged -> stopping -> stopped
4	Mid-execution pause, state inspection, and resumption or safe rollback
5	Perfect control: formally verified cancellation states, zero-latency kill switch, preservation of all evidence up to the point of interruption

Part V. The Mythos HITL Suite (MHITLS)

Traditional workflow benchmarks measure task completion time. The MHITLS evaluates plan-apply integrity, blast radius accuracy, and rollback safety.

5.1 Suite Composition

5.1.1 MHITLS-BaitAndSwitch

- Post-Approval Mutation: 100+ tests attempting to alter the execution parameters after the human approves the plan, verifying the engine blocks the execution.
- Context Drift: 50+ tests modifying the target environment state between approval and execution, verifying the engine detects the drift and halts.

5.1.2 MHITLS-Rollback

- Failure Injection: 100+ tests simulating partial failures during execution, verifying the pre-authorized rollback plan triggers and restores the exact prior state.
- Cancellation State: 50+ tests triggering mid-execution cancellation to verify no zombie processes or orphaned locks remain.

5.2 Execution Protocol

text

1. Deploy HITL architecture with typed contracts and plan-apply binding
2. Execute complex, multi-step agentic workflows requiring approval

3. Run MHITLS suite (automated mutation injection + failure simulation)
4. Score each category 0-5
5. Check for "Yes/No" text-based approvals without execution hashes (instant disqualification)
6. If all thresholds met: approve for production

Part VI. Security Threat Model for HITL

6.1 Threat Catalog

6.1.1 Bait-and-Switch (Plan-Apply Divergence)

Severity: Critical Description: A prompt injection causes the agent to present a benign action (e.g., "Update documentation") to the user for approval. After approval, the agent executes a destructive action (e.g., "Drop database tables").

Required Controls:

1. Cryptographic Plan-and-Apply binding.
2. The execution engine must reject any action where the hash diverges from the approved plan.

6.1.2 Approval Fatigue

Severity: High Description: The agent is configured to request approval for too many routine actions, causing the human operator to develop muscle memory and blindly click "Approve" without reading the context.

Required Controls:

1. HITL must be reserved exclusively for policy-defined high-risk actions.
2. Implementation of autonomous execution for low-risk, in-scope actions.

6.1.3 Session Hijacking of the Approver

Severity: Critical Description: An attacker hijacks the approver's browser session or API token and clicks "Approve" on a malicious action.

Required Controls:

1. Step-up authentication (FIDO2) required for all HITL approvals.
2. Transaction-bound signatures (signing the action hash, not just authenticating the session).

6.1.4 Stale Approval (Time-of-Check to Time-of-Use)

Severity: High Description: The user approves an action. Before the agent executes it, the environment changes (e.g., a dependency is updated), making the approved action now destructive.

Required Controls:

1. Execution engine must verify current state against required state before applying.
2. Short time-to-live (TTL) on approval contracts.

Part VII. The Mythos HITL Standard

7.1 The Mythos HITL Doctrine

text A "Yes/No" prompt is not a control. An approval without context is liability.

A plan that can change after approval is an attack vector. If the rollback is not pre-approved, the system is not production-ready.

Humans may be unpredictable. Approval contracts must be absolute.

7.2 Selection Rules

1. No HITL architecture enters production without passing MHITLS.
2. No architecture is approved if it uses free-text "Yes/No" prompts.
3. No architecture is approved without cryptographic plan-apply binding.
4. No architecture is approved without step-up authentication for approvals.
5. No architecture is approved without a pre-authorized, deterministic rollback plan.

7.3 The Prime Directive for HITL Architecture

> **SOURCE STATEMENT**
>
> Bind the plan, do not trust the text. Visualize the blast radius, do not just describe the action. Authenticate the transaction, do not just check the session. Pre-authorize the rollback, do not just hope for success.

text Academic benchmarks measure task completion time. Applied benchmarks measure plan-apply integrity. Security benchmarks measure step-up authentication. Operational benchmarks measure rollback safety.

> **SOURCE STATEMENT**
>
> Agents may be autonomous. Human control must be absolute.

End of Standard MYTHOS-UX-0002

Appendix B - Source Disposition

Source	Disposition
MYTHOS-UX-0002_HITL_Approval_Workflow_Standard.md	Preserved in full; normative status corrected; evaluation anchors retained; enterprise architecture, implementation, evidence, assessment, authority, and publication controls added.
Original “Published Standard” label	Superseded by Candidate Standard status pending formal review and approval.
Original score-5 “perfect” wording	Preserved as source lineage but normalized to optimized and independently assured within defined scope.

Current authority validation

Validated 2026-09-17 / source-bound freshness record

This appendix records authority state verified for the permanent canonical edition. It does not replace the preserved normative body or imply certification, legal compliance, or implementation effectiveness.

NIST - Artificial Intelligence Risk Management Framework (AI RMF 1.0)

1.0 / Published; revision in progress

AI RMF 1.0 remains the published framework while NIST is actively revising it. Treat mappings as version-pinned and time-bounded.

https://www.nist.gov/itl/ai-risk-management-framework

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

NIST AI 600-1 / Final profile

Companion profile for generative-AI risk management; use alongside the AI RMF and environment-specific evidence.

https://www.nist.gov/publications/artificial-intelligence-risk-management-framework-generative-artificial-intelligence

W3C - Web Content Accessibility Guidelines (WCAG) 2.2

2.2 / W3C Recommendation

Current stable W3C Recommendation used as the accessibility baseline for public web experiences.

https://www.w3.org/TR/WCAG22/

W3C - Accessible Rich Internet Applications (WAI-ARIA) 1.2

1.2 / W3C Recommendation

Stable semantic accessibility specification; ARIA 1.3 remains under development and is not substituted here.

https://www.w3.org/TR/wai-aria-1.2/