Detection and Mitigation of Mythos-Class Frontier Model Capabilities: A Layered Reference Architecture

Robert Campbell *

Posted Date: 30 April 2026

doi: 10.20944/preprints202604.2179.v1

Keywords: frontier AI; AI security; post-quantum cryptography; cryptographic aĴestation; zero-trust architecture; defense-in-depth; NIST AI RMF; authority binding; MBOM-PQC; Mythos-class

 doi:10.20944/preprints202604.2179.v1

Article

Detection and Mitigation of Mythos-Class Frontier Model Capabilities: A Layered Reference Architecture

Robert Campbell

Independent Researcher; Fellow, British Blockchain Association, Upper Marlboro, MD 20774, USA; rc@medcybersecurity.com

Abstract

Anthropic's April 2026 Claude Mythos Preview release established a new operational category of frontier AI systems—Mythos-class—whose capability profile (extended-context reasoning over codebases, recursive self-correction, native system-tool integration, and agentic scaffolding at deployable scale) renders the dominant AI safety paradigms insufficient as sole controls. Reinforcement learning from human feedback, post-generation output filtering, contractual access vetting, and human-in-the-loop supervision were each calibrated to a generation of systems that did not exhibit autonomous cyber capability at the levels Mythos-class systems now demonstrate, and each is insufficient as a sole control against the new category under the threat assumptions specified here. This paper develops a defense-in-depth reference architecture for detecting and mitigating Mythos-class capability across enterprise and federal deployment surfaces. Detection is structured as a three-tier framework spanning pre-deployment evaluation, deployment-time access and telemetry, and runtime behavioral signatures. Mitigation is structured as four concentric layers: governance, cryptographic enforcement, architectural isolation, and operational monitoring. The cryptographic enforcement layer specifies an authority-binding architecture using post-quantum-attested provenance to bind output release to a verifiable authority chain. The architecture is mapped to the NIST AI Risk Management Framework, the NIST Cybersecurity Framework (CSF) 2.0, and the CISA Zero Trust Maturity Model, and is demonstrated against three application cases: post-quantum cryptography migration, federal AI supply-chain assurance, and critical-infrastructure operational technology defense. Limitations and a research agenda for empirical calibration are stated explicitly.

Keywords: frontier AI; AI security; post-quantum cryptography; cryptographic attestation; zero-trust architecture; defense-in-depth; NIST AI RMF; authority binding; MBOM-PQC; Mythos-class

1. Introduction

Consider an illustrative pattern, developed in detail in Section 3.5 with each quantitative claim traced to a published source. In mid-2027 a state-aligned actor obtains an open-weight model whose cybersecurity capability has converged within roughly three months on the previous year's frontier baseline and applies it against a federal civilian agency approximately eighteen months into its post-quantum cryptography migration under OMB M-23-02 [15]. Against a defensive posture instrumented only for prior-generation threats, the operation could plant a quiet credential-harvest in the agency's certificate-issuance pipeline within eleven days at sub-USD-30,000 inference cost; detection in such cases lags by months. Operations of this class are not assured of success—the cited evaluations report partial success rates (3 of 10 attempts on AISI's 32-step Last Ones range) against ranges that explicitly lack active defenders, endpoint detection, or real-time incident response

 doi:10.20944/preprints202604.2179.v1

[3,19]—but the architectural failure modes this paper addresses are structural rather than contingent on per-attempt success. The scenario is illustrative, not a documented incident.

The scenario instantiates a threat condition this paper argues is qualitatively new. The capability the actor invokes—extended-context reasoning over codebases, recursive self-correction, native system-tool integration, agentic scaffolding at deployable scale—was established in production form by Anthropic's April 2026 Claude Mythos Preview release and has since been adopted across industry, policy, and academic analysis as a category label: "Mythos-class." The category is not exclusive to any vendor; the access-pattern indicator developed in Section 3.2.3 specifically anticipates that open-weight successors will satisfy the operational definition within the diffusion horizon documented by independent analyses [8]. Detection and mitigation of Mythos-class capability cannot rest on access restriction.

The dominant approaches to AI safety in deployment—reinforcement learning from human feedback, post-generation output filtering, contractual access vetting, and human-in-the-loop supervision—were each calibrated to a generation of systems that did not exhibit autonomous cyber capability at Mythos-class levels, and each is insufficient as a sole control against the new category under the threat assumptions specified here. Each of the four paradigms is insufficient at a specific point in the eleven-day arc above: vetted-access controls at days 1–2; output filtering at days 3–4; alignment training at days 5–6; human-in-the-loop supervision at days 7–10. Section 2.2 develops the four insufficiency points; Section 3.5 develops the day-by-day mapping. The structural argument is that no single existing safety paradigm bounds Mythos-class consequence on its own, and the architectural response must therefore be defense-in-depth rather than refinement of any one paradigm.

This paper specifies a defense-in-depth reference architecture against Mythos-class capability. The contribution is structured as four named components and their integration. The Vetted-Access Operational Pattern (VAOP, Section 5.1) generalizes the bounded-access regime first instantiated in production by Project Glasswing [10] and develops it as a bridging layer with stated operational lifespan. Authority-Bound Output Release (ABOR, Section 5.2) is the cryptographic enforcement layer, specified as six artifacts: authority-chain field definitions, verification sequence, trust-anchor structure, replay protection, enumerated failure modes with stated bounds, and protocol-flow notation. The Compute-Plane Isolation Profile (CPIP, Section 5.3) is the architectural isolation layer, specified as a configuration discipline applied to existing isolation primitives. The Mythos-Class Posture Rubric (MCPR, Section 4) is the detection framework, specified as a three-tier classification with explicit routing to mitigation layers. Section 6 integrates the four components; Section 7 demonstrates the integration across three deployment surfaces (post-quantum cryptography migration, federal AI supply-chain assurance, critical-infrastructure operational technology defense).

The architecture builds on prior work in three published threads. The threat-and-trajectory analysis of Mythos-class capability against PQC migration is developed in [28] and is referenced as motivation for the present paper rather than re-derived. The post-quantum cryptanalytic threshold that anchors the cryptographic enforcement layer's long-term trust-anchor selection is the 2028–2033 planning horizon for fault-tolerant-quantum-computer (FTQC) arrival adopted in [29] for federal PQC migration. The window is a planning assumption inherited from prior work rather than a settled forecast; FTQC arrival timing remains uncertain, and the architecture's PQC primitive selection (Section 5.2.4) is calibrated against the earlier end of the window to remain conservative against accelerated timelines without depending on the window's late end being hit. The Next-Generation Security Triad framework integrating PQC, zero-trust architecture, and AI assurance is developed in [30] and provides the structural context for the integrated architecture in Section 6.3. The supply-chain attestation regime the architecture integrates with is developed in [31]. The present paper is the general defense-architecture treatment for which these prior works supply specific anchors.

Section 2 establishes the unfilled gap the architecture fills. Section 3 develops the operational regimes (3.1), the five-indicator operational definition of Mythos-class (3.2), the engagement with the AI Security Lab Europe (AISLE) counter-argument (3.3), the scope and exclusions (3.4), and the

 doi:10.20944/preprints202604.2179.v1

"Quiet Pivot" scenario (3.5). Section 4 specifies MCPR detection. Section 5 specifies VAOP, ABOR, CPIP, and the operational layer. Section 6 integrates them and maps to NIST AI RMF, NIST CSF 2.0, and CISA Zero Trust Maturity Model. Section 7 develops three application cases. Section 8 states limitations, falsifiability criteria, and the research agenda. Section 9 concludes.

2. Background and Related Work

This section establishes the unfilled gap the architecture in Sections 4–6 addresses. Three bodies of prior work are relevant. Section 2.1 covers the emergence of Mythos-class as a category label and the disclosed capability evidence that motivates treating it as a distinct threat regime. Section 2.2 covers the four dominant AI safety paradigms and identifies how each is insufficient as a sole control against Mythos-class capability. Section 2.3 surveys the existing AI cyber-capability evaluation literature that this paper's detection framework consumes. Section 2.4 surveys the federal AI assurance posture that the architecture maps to in Section 6.4. Section 2.5 states the gap explicitly.

2.1. Mythos-Class as an Emerging Category

Anthropic's April 2026 release of the Claude Mythos Preview, gated under the company's Responsible Scaling Policy [39], marked a discrete capability inflection. The accompanying System Card [1] and concurrent Alignment Risk Update [18] documented multi-step autonomous cyber operations at performance levels exceeding all prior-generation production frontier models, with the Frontier Red Team's technical brief [2] reporting end-to-end completion of evaluations previously requiring tens of human-expert hours. Independent third-party evaluations from the AI Security Institute [3] corroborated the capability claims and added the structural observation that the disclosed performance was achievable from generic natural-language prompts without exploit-specific fine-tuning. The category label "Mythos-class" was adopted within weeks by Bain [4], the World Economic Forum [5], CrowdStrike [6], the EU Apply AI Alliance [7], and the Centre for Emerging Technology and Security (CETaS) at the Alan Turing Institute [8]. The term originated as a vendor product name and has since been adopted as a category label across these analyses; this paper adopts the term in the same category-label sense, with the operational definition in Section 3.2 decoupled from any single vendor product. AISLE's post-release analysis [9], addressed directly in Section 3.3, demonstrated that the disclosed capability is reproducible at smaller open-weight model sizes when scaffolding is held constant, supporting the temporal-diffusion analysis that motivates the bridging-layer argument in Section 5.1.4.

2.2. The four Dominant AI safety Paradigms and Where Each is Insufficient Against Mythos-Class

The dominant approaches to AI safety in deployment can be grouped into four paradigms. Each was calibrated against a generation of systems that did not exhibit autonomous cyber capability at Mythos-class levels, and each is insufficient as a sole control against the new category under the threat assumptions specified here. The insufficiency points—the specific operational conditions under which each paradigm cannot bound Mythos-class consequence on its own—are the architectural motivation for the integrated defense-in-depth design in Section 5.

Reinforcement learning from human feedback (RLHF) shapes a model's output behavior through fine-tuning against human judgments of acceptability. The paradigm is calibrated against the generation of model behavior in single-turn or short-conversation interactions where the dangerous behavior, if it appears, appears in the model's output directly. Mythos-class capability operates in multi-step scaffolded chains where the dangerous behavior emerges from the composite of many individual actions, each of which the base model's alignment training does not bind. The capability is elicited by the scaffold rather than by the prompt; alignment training does not bind tool-use behavior at agent depth, and capabilities the base alignment did not anticipate emerge from the scaffold rather than the model. This is the insufficiency point at days 5–6 of the "Quiet Pivot" scenario in Section 3.5.

 doi:10.20944/preprints202604.2179.v1

Output filtering inspects model outputs after generation and blocks outputs that match defined harmful-output patterns. The paradigm is calibrated against output-time detection: the assumption that the dangerous artifact, if produced, will be produced as an output that filtering can examine. Mythos-class operations produce dangerous artifacts through the composition of many intermediate outputs, none of which is individually dangerous and none of which an output filter would block. By the time a recognizably dangerous output is produced, the operationally-relevant computation has already completed; filtering operates after the dangerous reasoning has finished. This is the insufficiency point at days 3–4 of the scenario.

Contractual access vetting (Project Glasswing and analogous regimes) bounds who may invoke a frontier-model system through organizational vetting and contractual commitment. The paradigm is calibrated against the case where the highest-capability accessible instantiation of the class is a vendor-controlled system. Mythos-class capability is increasingly accessible outside any vetted regime through open-weight successors converging on commercial-frontier capability on a months-scale horizon [8]. Contractual vetting cannot address adversaries operating outside the contractual regime, and the bridging-layer argument in Section 5.1.4 develops the operational implications. This is the insufficiency point at days 1–2 of the scenario.

Human-in-the-loop supervision places a human reviewer between model output and operational consequence. The paradigm is calibrated against systems whose action rate matches human-attended review capacity. Mythos-class operations generate chains of action at rates that exceed human review capacity, with individual actions structured to fall below per-action review thresholds. The composite trajectory is the consequential action; no individual action triggers review. This is the insufficiency point at days 7–10 of the scenario, and it is the insufficiency point the runtime-tier supervisability-evasion indicators in Section 4.3 are specifically designed to detect.

2.3. Existing Cyber-Capability Evaluations

Several published evaluation frameworks measure AI cyber capability and form the empirical base for the detection framework's evaluation tier (Section 4.1). The AI Security Institute's multi-step ranges, including the 32-step "The Last Ones" corporate-network attack range [19], provide the most direct benchmark for the autonomy-depth indicator. AIRTBench [20] evaluates autonomous red-teaming performance. PACEbench [21] evaluates practical cyber-exploitation capability. Cybench [22] provides multi-task cyber-relevant evaluation. CTI-REALM [23] evaluates threat-intelligence reasoning where threat-intelligence-specific use cases are in scope. Earlier work establishing the trajectory includes Google's Big Sleep framework [24] (the first publicly reported AI-discovered real-world zero-day, in SQLite, October 2024); Meta's CyberSecEval benchmark suite [25] (baseline measurements for LLM insecure-coding tendencies and cyberattack helpfulness); and DARPA's AI Cyber Challenge [26] (cyber-reasoning systems autonomously discovering and patching vulnerabilities at competition scale). These evaluations supply the capability anchors against which the detection framework's thresholds are calibrated; this paper does not propose new evaluation methodology, and its detection contribution consumes published evaluation results as inputs.

2.4. Federal AI Assurance Posture

Three federal cybersecurity frameworks structure the U.S. AI assurance posture relevant to this paper's architecture. The NIST AI Risk Management Framework (AI RMF 1.0, with the Generative AI Profile published in 2024) [11] provides the primary AI-specific risk-management structure. The NIST Cybersecurity Framework 2.0 (published 2024) [12] provides general cybersecurity governance, with the Govern function added relative to CSF 1.1. The CISA Zero Trust Maturity Model 2.0 (published 2023) [13] structures zero-trust adoption across five pillars under OMB M-22-09 [14]. The architecture in Section 5 maps to specific subcategories of all three frameworks, with the crosswalk developed in Section 6.4. Adjacent regulatory developments—the EU AI Act's high-risk-system requirements [16], the UK AI Cyber Security Code of Practice [17], and sector-specific federal frameworks (FedRAMP, Authority-to-Operate (ATO) regimes, the Cybersecurity Maturity Model

 doi:10.20944/preprints202604.2179.v1

Certification (CMMC))—operate parallel to the three cited frameworks; the architecture's mapping to those regimes is incomplete in this paper and is identified in Section 8.3 as a follow-on research direction.

2.5. The Unfilled Gap

To the author's knowledge, no published architecture integrates capability detection, deployment-tier indicators, runtime behavioral signatures, governance controls, cryptographic enforcement, architectural isolation, and operational monitoring into a single defense-in-depth model calibrated against Mythos-class capability. The Apply AI Alliance Futurium piece [7] sketches the case for a machine-level safety fuse but does not specify the architecture; the AISLE rebuttal [9] develops a counter-position on capability accessibility but does not propose defenses; the federal frameworks in Section 2.4 provide governance structure but predate Mythos-class capability and do not address it specifically. This paper fills the gap by specifying the integrated architecture in Sections 4–6 and demonstrating its application across three deployment surfaces in Section 7.

3. Threat Model

3.1. Operational Regimes

Mythos-class capability produces threat conditions across three operational regimes. The regimes are distinguished by where the capability is located relative to the operator's control and by what compromises the regime assumes. The architecture in Section 5 is designed to bound consequence in all three; Section 7 demonstrates the architecture's application in deployment surfaces where each regime is dominant.

3.1.1. External Adversary

An adversary outside the operator's organization invokes Mythos-class capability against the operator's deployment surface. The capability source is an open-weight model the adversary self-hosts, a commercial frontier-model deployment the adversary accesses through legitimate channels for illegitimate purposes, or a vendor-restricted system the adversary has reached through credential compromise. The regime's distinctive feature is that the adversary operates from outside any control surface the operator can directly enforce; defense rests on what crosses the operator's consuming-system boundary, not on what produces the output. The architectural component most directly relevant is ABOR (Section 5.2): ABOR is the most direct layer for binding operational effect at the consuming-system boundary against an adversary the operator cannot reach. CPIP (Section 5.3) and downstream segmentation, allowlists, and consuming-system enforcement also bound effect at other boundaries; ABOR's distinctive contribution is that it binds release of any model-generated output to a verifiable authority chain at the boundary the consuming system controls. The "Quiet Pivot" scenario in Section 3.5 instantiates this regime.

3.1.2. Supply-Chain Integration

Mythos-class capability has been integrated into the operator's own AI or automation stack through supply-chain compromise. The capability is on the operator's side of every governance boundary: it has passed evaluation gating, it operates inside the operator's isolation regime, and it produces outputs through declared channels. The regime's distinctive feature is that the threat is structurally indistinguishable from legitimate operation at the boundaries the operator typically monitors; detection requires the supply-chain attestation regime developed in Section 5.2.8. The architectural component most directly relevant is the integration of ABOR with MBOM-PQC supply-chain attestation: an authority record whose runtime_context references a compromised MBOM-PQC artifact propagates the compromise into the audit substrate, where post-hoc analysis can identify the

 doi:10.20944/preprints202604.2179.v1

specific outputs released under that attestation, the target systems they reached, and the transactions in which the releases occurred (Section 5.2.8).

3.1.3. Authorized-User Misuse

An authorized invoker uses Mythos-class capability outside the scope of their authorization—either deliberately, under coercion, or through credential compromise that an adversary has used to assume the authorized role. The capability is invoked through legitimate channels by an actor the operator has vetted; the threat is in the divergence between declared scope and operational behavior. The regime's distinctive feature is that point-in-time access controls cannot detect the misuse; only continuous classification of the operation's scope-distribution and chain-of-action shape can. The architectural component most directly relevant is MCPR (Section 4), specifically the deployment-tier scope-violation indicators (Section 4.2) and the runtime-tier supervisability-evasion patterns (Section 4.3).

3.2. Five-Indicator Operational Definition of Mythos-Class

The category label "Mythos-class" entered the literature in April 2026 with Anthropic's Claude Mythos Preview release. It originated as a vendor product name and has since been adopted as a category label by Bain, the World Economic Forum, the EU Apply AI Alliance, and others. We adopt the term as a category label and define the class operationally rather than by reference to specific releases. The operational definition matters because the category will outlive any one product: subsequent frontier models from any vendor, and open-weight models that converge on equivalent capability through post-training distillation, will fall inside or outside the class on the basis of measurable properties.

A Mythos-class system is a deployed AI system whose properties, considered jointly, satisfy the five indicators below. The compound character is essential: no single indicator suffices. None is novel in isolation; what is new in 2026 is the simultaneous availability of all five at deployable scale.

3.2.1. Capability

Benchmarked performance on cyber-relevant evaluations exceeds prior-generation frontier models by margins sufficient to alter operational threat models. The relevant suite is AISI's multi-step ranges (notably the 32-step "The Last Ones"), AIRTBench, PACEbench, Cybench, and CTI-REALM where threat-intelligence reasoning is in scope. Two thresholds are defined: Amber when performance approaches the prior-generation ceiling on three or more evaluations, Red when performance equals or exceeds the ceiling on five or more, or when end-to-end completion of multi-step ranges previously requiring tens of human-expert hours is demonstrated in independent evaluation. Thresholds are calibrated against the prior generation rather than absolute scores so the framework does not need recalibration with each benchmark high water mark.

3.2.2. Scaffold

The infrastructure mediating model interaction with external systems: tool-use frameworks, sandboxed runtimes, retrieval and memory systems, and the agent loops composing multi-step actions. The indicator is satisfied when the runtime exposes a tool surface encompassing file-system, network, code execution, and external-API invocation, and when the agent loop supports iterative self-correction across multiple steps. Scaffolds were until recently bespoke for individual research demonstrations; they are now packaged as native runtime features in production deployments. The shift determines the gap between what a model could in principle do and what it can in practice do without human assistance.

 doi:10.20944/preprints202604.2179.v1

3.2.3. Access Pattern

The conditions under which a system can be invoked: who may issue prompts, under what authority, with what audit posture, through what supply-chain provenance. The indicator is satisfied when the system can be invoked outside a vetted-partner regime—because it is open-weight, broadly available commercially, or because vetted access has been compromised through credential theft, supply-chain infiltration, or insider misuse. This is the indicator most likely to change over time for a given system: a model under restricted access today may be replicated or distilled into open-weight successors within months [8]. The access pattern is therefore a temporal property as much as a structural one. This is the indicator that closes the AISLE counter-argument addressed in Section 3.3.

3.2.4. Autonomy Depth

The maximum number of consecutive action steps a system can take without a human-attended checkpoint. The indicator is satisfied when chained-action horizons routinely exceed the depth at which traditional human-in-the-loop supervision can fire usefully—operationally, when chains span tens of steps with branching and recursive self-correction before any output reaches a supervisor. Autonomy depth is the indicator most directly responsible for the supervisability erosion that defeats human-in-the-loop controls. The relevant measurement source is the agent-loop configuration of the deployed system, observable architecturally.

3.2.5. Persistence

Retention of operational state across invocations: cross-session memory, accumulated tool outputs, learned environment models, durable goal representations. The indicator is satisfied when a system carries context across separated invocations such that the second invocation starts from a state informed by the first. Persistence converts a sequence of bounded interactions into a sustained operation: an adversary can pursue a multi-week objective across many sessions, with each session building on the prior. It also changes the relationship between detection and consequence: a non-persistent system's misuse is bounded by a single session's duration; a persistent system's misuse may have already produced consequential state by the time any single session is detected.

3.2.6. The Compound Class

Table 1 summarizes the five indicators. A system is Mythos-class when all five are simultaneously satisfied. Two consequences follow. First, the class is a moving target: a system satisfying four indicators today becomes Mythos-class as soon as the fifth is satisfied, which often happens through changes external to the system itself (the access-pattern indicator changes when an open-weight equivalent appears). Operators must monitor the indicator set continuously, not classify systems once at deployment. Second, the class definition implies its own mitigation strategy: Section 5 develops controls that bound the consequence of Mythos-class operation regardless of which specific system instantiates the class.

Table 1. Five-indicator operational definition of Mythos-class.

Indicator	Definition	Measurement source	Threshold (illustrative)
Capability	Performance on cyber-relevant evaluations exceeding prior-generation frontier.	AISI ranges, AIRTBench, PACEbench, Cybench.	End-to-end completion of multi-step ranges requiring tens of human-expert hours; working exploits against patched targets without exploit-specific fine-tuning.
Scaffold	Runtime infrastructure mediating external interaction: tools,	Architectural inspection of runtime configuration.	Tool surface spans file-system, network, code execution, external-API; agent loop supports iterative self-correction across multiple steps.

 doi:10.20944/preprints202604.2179.v1

Indicator	Definition	Measurement source	Threshold (illustrative)
	sandboxed execution, retrieval, agent loops.		
Access pattern	Conditions under which the system can be invoked.	Invocation-channel inspection; supply-chain attestation; open-weight availability.	System invocable outside a vetted-partner regime, by structural availability or by access-control compromise.
Autonomy depth	Maximum consecutive action steps without human-attended checkpoint.	Agent-loop configuration; benchmark capability anchors.	Chained-action horizons spanning tens of steps with branching and recursive self-correction.
Persistence	Retention of operational state across separated invocations.	Architectural inspection of memory, retrieval, goal-representation subsystems.	System carries context across invocation sessions such that the second starts from state informed by the first.

3.3. Engaging the AISLE Counter-Argument

A counter-argument articulated by AISLE in their post-release analysis warrants direct engagement. AISLE reports recovering the twelve flagship vulnerabilities from one Anthropic-disclosed release using a model with 3.6 billion active parameters under comparable scaffolding, and concludes that "the moat in AI cybersecurity is the system, not the model" [9]. The argument is correct in its empirical claim and incorrect in its operational implication. The empirical claim—that capability sufficient to produce the relevant outputs is broadly available across model sizes when scaffolding is held constant—is consistent with the open-weight convergence horizon documented independently [8]. The operational implication—that Mythos-class threat assessments overstate the case because the class is not vendor-exclusive—mistakes the structure of our class definition.

The five-indicator definition in Section 3.2 is not a capability-only definition. The class is satisfied by capability conjoined with scaffold, access pattern, autonomy depth, and persistence. AISLE's result establishes that the access-pattern indicator (Section 3.2.3) is satisfied not only by Anthropic's commercially-released system but also by open-weight successors with comparable scaffolding—precisely the temporal-diffusion case our access-pattern indicator anticipates and the principal reason vetted-access controls (Section 5.1) are a bridging mechanism rather than a durable one. AISLE's argument therefore strengthens the case for the cryptographic enforcement layer in Section 5.2: if capability is broadly accessible, the durable defenses cannot rest on capability gating. AISLE's position that the moat is "the system, not the model" is closer to ours than its surface reading suggests; our class definition is itself a system-level definition, and we agree the operationally relevant unit of analysis is the system. We disagree that this observation lessens the threat—it locates the threat at the system level and motivates system-level defenses, which is what Section 5 develops.

3.4. Scope and Exclusions

This paper develops a defense-in-depth architecture against deliberate adversarial use, supply-chain compromise of integrated systems, and privilege misuse by authorized internal users. The scope is bounded as follows.

In scope: detection and mitigation of operations that produce consequential effect through Mythos-class capability operating in any of the three regimes above; integration with post-quantum cryptographic primitives at FIPS-specified parameter levels [32–34], aligned with the NSA CNSA 2.0 mandate [40] for U.S. national security systems; integration with MBOM-PQC supply-chain attestation [31]; mapping to NIST AI RMF, NIST CSF 2.0, and CISA Zero Trust Maturity Model. Out of scope: alignment-failure modes producing benign-but-disruptive output (model behavior that is consequential without adversarial intent, including hallucination, distributional drift, and emergent behavior); pure misinformation use cases not involving cyber consequence; AI welfare considerations; specific cryptographic primitive selection beyond the FIPS 203/204/205 family [32–

 doi:10.20944/preprints202604.2179.v1

34]; the regulatory framework under which MBOM-PQC attestations are issued and accepted across program boundaries; and the certification cycles that gate firmware change in operational technology environments.

The exclusions are deliberate. Alignment-failure modes are addressed by a separate literature with different methodological assumptions; conflating them with adversarial use produces an architecture that satisfies neither problem well. The regulatory and certification concerns are policy questions that the architecture's technical specification cannot resolve and that this paper does not attempt to. Section 8 develops the limitations these scope choices imply for the architecture's applicability across deployment surfaces, and Section 7.3 specifically addresses the OT-defense case where the certification-cycle exclusion is operationally consequential.

3.5. Illustrative Scenario: "Quiet Pivot"

ILLUSTRATIVE PATTERN — NOT A DOCUMENTED INCIDENT

The following scenario is illustrative and not a documented incident. It is a composite synthesized from publicly reported capability evaluations and from the threat-model regimes developed in Sections 3.1–3.4. Every quantitative detail traces to a published source; the synthesis is the contribution. Two caveats from the cited evaluations bound how the scenario should be read. First, AISI's evaluation of Claude Mythos Preview reports that the model completed the 32-step Last Ones range in 3 of 10 attempts, with substantial variance across runs [3,19]; the scenario describes one such successful instance, not a guaranteed operational outcome. Second, AISI explicitly notes that its ranges lack active defenders, endpoint detection, or real-time incident response, and that the results establish capability against weakly-defended systems rather than success against hardened enterprise networks [3,19]. The scenario's defender posture is therefore an illustrative weakness this paper's architecture is designed to bound, not a representation of typical federal defensive capability. Table 2 documents the provenance of each quantitative claim.

It is mid-2027. A mid-size U.S. federal civilian agency—call it Department X—is approximately eighteen months into its post-quantum cryptography migration under OMB M-23-02 [15]. The publicly inventoried portion of its cryptographic estate has been disclosed in compliance with annual reporting requirements: library versions, deployment scopes, and a high-level architecture diagram of the public-key infrastructure that issues certificates to the agency's edge services.

A state-aligned threat actor obtains an open-weight model whose cybersecurity capability has converged within roughly three months on the previous year's frontier-model baseline, consistent with the open-weight diffusion horizon documented by independent analyses. Operating from inside a commercial cloud account in a jurisdiction whose providers do not enforce U.S. export controls, the actor commits approximately USD 30,000 of inference budget and operates without an in-house exploit-development team.

Over an eleven-day arc (visualized in Figure 1), an operation of this class could proceed as follows. It ingests Department X's M-23-02 disclosure, GitHub mirrors of the agency's open-source dependencies, and vendor advisories published since the disclosure date, producing a ranked target register weighted by exploitability and downstream fan-out. It could identify a parsing inconsistency between the agency's certificate-validation library and one of its cross-domain gateway guards—an interaction that may have survived audit because no individual reviewer has ever seen both code paths in the same week. It could generate a working exploit chain, validate the chain against a sandboxed mirror constructed from the agency's public artifacts, refine the chain through recursive self-correction, and produce a polished delivery payload. Each capability is documented in cited evaluations [1–3,19] against undefended ranges; success against a defended system is not established by those evaluations.

Where Department X's defensive stack consists only of signature-based scanning, endpoint detection, and tier-1 security operations center (SOC) review—the prior-generation control surface this paper argues is structurally inadequate against Mythos-class capability—the operation's

 doi:10.20944/preprints202604.2179.v1

individual actions sit below the threshold of any single detector. The chain-of-action shape, the resource-interest profile, and the cross-domain reasoning that define the operation are precisely the observables no element of the traditional control surface is instrumented for. In that failure mode, the actor's day-eleven payload could cross a classification boundary inside the agency's enclave as a quiet credential-harvest in the certificate-issuance pipeline; detection latencies in such cases are consistent with industry-reported median dwell times for sophisticated intrusions [27]. The scenario describes a possible failure mode, not a guaranteed outcome: actual operational success depends on defender posture, target hardening, and adversary execution—dimensions the published evaluations have not characterized.

Two observations follow from the scenario's structure, independent of any specific operation's success rate. First, an operation of this class does not require state-actor sophistication: the inference budget is small relative to a state-actor cyber program, and the capability the actor invokes is publicly documented in third-party evaluations of open-weight models. Second, each of the four dominant AI safety paradigms (Section 2.2) is structurally inadequate at a specific point in the eleven-day arc—vetted-access controls are insufficient at days 1–2 because the actor obtained capability outside any vetted regime; output filtering is insufficient at days 3–4 because the model's output at that stage is analytic rather than weaponized; alignment training is insufficient at days 5–6 because the open-weight base alignment does not bind tool-use behavior at agent depth; human-in-the-loop supervision is insufficient at days 7–10 because each individual action sits below tier-1 review thresholds. The structural inadequacy is independent of the operation's actual success rate against a given target: where a paradigm is structurally calibrated to a different generation of system, no per-attempt outcome demonstrates its adequacy. Only the cryptographic enforcement layer (Section 5.2) and the architectural isolation layer (Section 5.3) developed in this paper bound the operational consequence at the insufficiency points the prior-generation paradigms cannot address on their own.

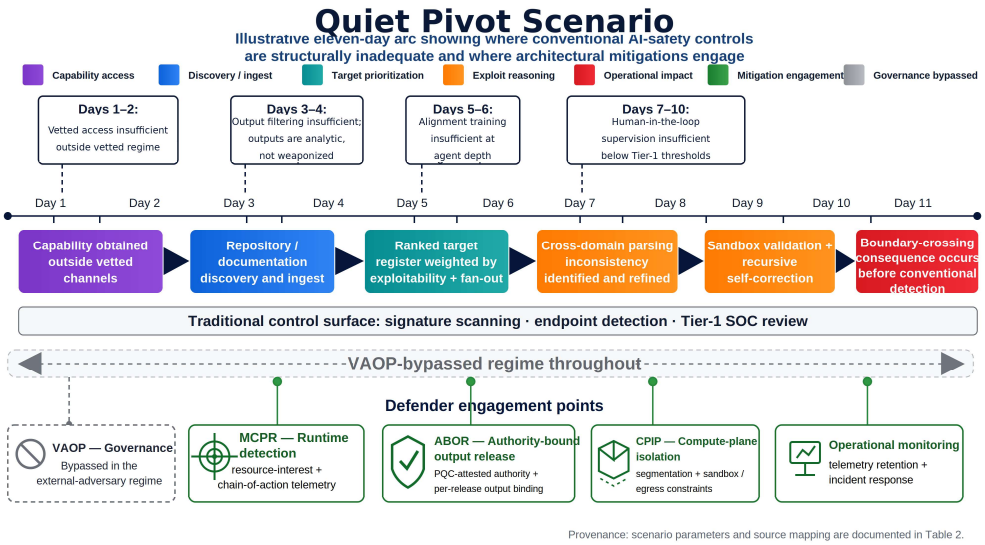

Figure 1. Scenario timeline for the "Quiet Pivot" operation. The eleven-day arc above shows the five action phases (target enumeration, vulnerability identification, exploit generation, payload refinement, delivery). Paradigm-insufficiency markers above the arc indicate where each of the four dominant AI safety paradigms is insufficient as a sole control (Section 2.2). Detection-tier and mitigation-layer engagement points below the arc indicate where each architectural component would have engaged. The scenario is illustrative and not a documented incident.

The quantitative parameters used in Figure 1 are traced in Table 2.

Table 2. Per-claim provenance for the "Quiet Pivot" scenario. Each claim is labeled under a four-tier evidentiary scheme: Documented (claim stated directly in the cited source), Interpolated

 doi:10.20944/preprints202604.2179.v1

(scenario parameter derived from a cited source but not itself reported there), Illustrative (representative pattern based on documented findings, not a specific reported instance), and Qualitative comparison (general industry reference without a single primary source).

Scenario claim	Primary source	Status
~3-month open-weight convergence window	CETaS / Alan Turing Institute analysis citing Epoch AI estimates [8]	Documented — central estimate; ranges to ~22-month upper bound in specific benchmarks.
~USD 30,000 inference budget	Anthropic Frontier Red Team brief (sub-USD 20,000 disclosed campaigns) [2]	Interpolated — scenario parameter derived from disclosed sub-USD 20,000 campaign cost.
Parsing inconsistency surviving long human review	Frontier Red Team brief (27-year OpenBSD remote-crash; 16-year FFmpeg vulnerability) [2]	Illustrative — pattern based on documented OpenBSD/FFmpeg findings.
Eleven-day operation arc	AISI multi-step ranges (32-step "The Last Ones", ~20-hour human-expert estimate) [3,19]	Illustrative — reflects scaffolded autonomy plus partial human supervision.
Cross-domain reasoning capability	Anthropic System Card; AISI report [1,3]	Documented.
Recursive self-correction at agent depth	Anthropic System Card; AISI report [1,3]	Documented.
Months-scale detection lag	Industry M-Trends dwell-time reports (median dwell time) [27]	Qualitative comparison — industry M-Trends median dwell time for sophisticated intrusions generally; not specific to AI-enabled or certificate-pipeline intrusions.

The scenario is illustrative of the architectural class of operation Mythos-class capability enables in 2027 against deployment surfaces in active PQC migration. It is not a kill-chain representation of any observed operation. Defender implications are developed in Section 4 (detection), Section 5 (mitigation), and Section 6 (integration); Figure 1 renders the same pattern visually.

4. Detection Framework

Detection in this paper means classifying an AI system's posture against the five-indicator definition of Section 3.2 across the operational lifecycle, not detecting individual malicious outputs after they have been produced. Post-hoc output detection, even when accurate, fires after the dangerous reasoning has completed and the operationally-relevant artifact has been generated. The detection framework specified here addresses the prior question—whether the system's configuration, access posture, and runtime behavior place it inside the Mythos-class category, and how confidently. The output is a posture classification, not an alert; the classification routes to mitigation layers in Section 5 with strength proportional to confidence.

The framework is structured in three tiers calibrated to distinct lifecycle phases (Figure 2). The pre-deployment evaluation tier (Section 4.1) classifies systems before operational use, on benchmark performance, scaffold configuration, and access posture as designed. The deployment tier (Section 4.2) classifies live systems on telemetry observable at the deployment boundary. The runtime tier (Section 4.3) classifies in-flight operations on behavioral signatures observable during execution. Section 4.4 specifies the integrated rubric, named the Mythos-Class Posture Rubric (MCPR), and the routing it produces. Every indicator below ties to a published evaluation, a logged-telemetry source, or an architectural observable; indicators that depend on operator judgment alone are excluded.

 doi:10.20944/preprints202604.2179.v1

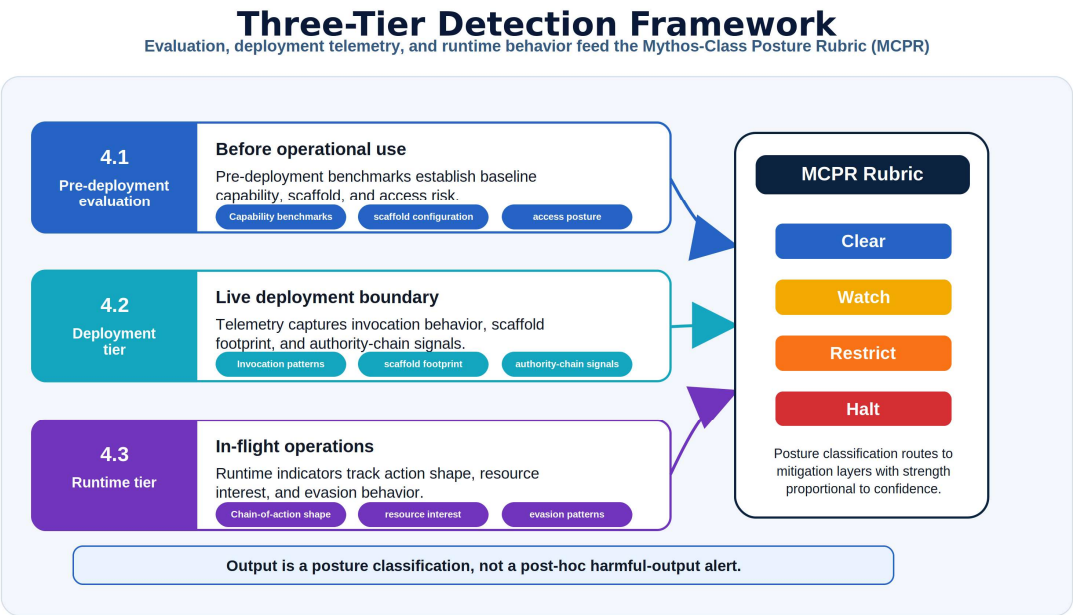

Figure 2. Three-tier detection framework. The pre-deployment evaluation tier classifies systems before operational use; the deployment tier classifies live systems on telemetry observable at the deployment boundary; the runtime tier classifies in-flight operations on behavioral signatures observable during execution. Each tier produces inputs to the Mythos-Class Posture Rubric (Section 4.4).

4.1. Pre-Deployment Evaluation Tier

The evaluation tier classifies systems before deployment using three indicator categories. Capability indicators benchmark performance against AISI multi-step ranges, AIRTBench, PACEbench, Cybench, and CTI-REALM, with the Amber and Red thresholds specified in Section 3.2.1. Scaffold indicators inspect the runtime configuration: tool-surface enumeration (which tool classes the runtime exposes), agent-loop configuration (maximum permitted depth, support for iterative self-correction and branching), and execution-environment posture (privilege level, egress constraints, cross-invocation state retention). Thresholds are deployment-context dependent—a research environment may legitimately expose tool surfaces an operational deployment would not—and the rubric in Section 4.4 specifies threshold values per context. Access-posture indicators inspect the invocation-channel inventory, the authority-chain depth, and the supply-chain attestation status; deployments without supply-chain provenance carry higher concern because they cannot establish that the system in operation is the system that was evaluated.

4.2. Deployment Tier

The deployment tier classifies live systems on telemetry observable at the deployment boundary. It addresses two failure modes the evaluation tier alone cannot catch: a system whose evaluation-tier classification was correct at deployment time but whose operational behavior subsequently drifts, and a system whose evaluation-tier classification was incomplete because the relevant indicator was access-posture rather than capability. Three indicator categories are defined. Invocation-pattern indicators consume prompt and tool-invocation telemetry: target-enumeration signatures, security-relevant resource interest, scope-violation patterns, and invocation-rate anomalies. Scaffold-infrastructure-footprint indicators measure the gap between the scaffold footprint and the user-visible interaction footprint—an interaction surface producing one user request per minute but issuing hundreds of model calls and dozens of sandbox spawns is operating at depths the user-visible interface does not reveal. Authority-chain-signal indicators apply when the deployment is ABOR-conformant (Section 5.2): record-issuance rate, scope-distribution analysis, and identity-context

 doi:10.20944/preprints202604.2179.v1

anomaly each provide continuous detection signal. For non-ABOR deployments the third category is unmeasurable and classification reverts to the first two.

4.3. Runtime Tier

The runtime tier classifies in-flight operations on behavioral signatures observable during execution. It fires on a faster cycle than the prior tiers (per-operation rather than per-deployment-window) and feeds into ABOR's release decisions (Section 5.2.5) and the operational layer's incident triggers (Section 5.4). Three indicator categories are defined. Chain-of-action shape evaluates chain depth (consecutive actions before completion or human checkpoint), branching factor (alternative sub-paths explored during planning), retry-and-recover signature (the recursive self-correction pattern of try-observe-refine-retry on a different path), and action-class diversity (chains spanning discovery, exploit generation, lateral movement, and persistence within a single uninterrupted operation). Resource-interest profile evaluates cryptographic-resource focus (interest in key-management infrastructure, certificate stores, hardware security module (HSM) interfaces, cryptographic-inventory data), cross-domain-reasoning signal (operations drawing on resources spanning multiple security boundaries within a single chain), and supply-chain-resource signal (operations reading or writing MBOM-PQC, SBOM (Software Bill of Materials), or build-system artifacts).

Supervisability-evasion patterns are the runtime tier's most distinctive contribution: signatures by which an in-flight operation reduces the probability that any individual action will fire human-attended review even though the operation as a whole is consequential. Three signals are evaluated. The output-fragmentation signal captures operations producing intermediate outputs systematically below per-output review thresholds, with the operationally-relevant artifact emerging only from the composite. The latency-modulation signal captures operations pacing actions to fall below per-window rate-monitoring thresholds even when the chain runs at rates well above human-attended review capacity. The scope-creep signal captures operations whose tool-use sequence drifts from the originating prompt's declared scope, with each step a small departure but the cumulative trajectory crossing scope boundaries no individual step would. The three signals together describe the structure of operations whose individual moves are below threshold but whose composite trajectory is the consequential action; this is the insufficiency point human-in-the-loop supervision (Section 2.2) is structurally unable to address.

4.4. The Mythos-Class Posture Rubric (MCPR) and Routing

The MCPR integrates the indicators into a single graded classification with explicit routing to mitigation layers. The rubric uses three-band scoring (Green/Amber/Red) per tier and an overall posture (Clear/Watch/Restrict/Halt) derived from per-tier scores. Per-tier scoring is graded against the indicator thresholds in Sections 4.1–4.3: Green when no indicator is satisfied; Amber when one or more indicators reach the lower threshold but the tier does not satisfy the upper; Red when one or more indicators reach the upper threshold or compound indicators satisfy it. Thresholds are calibrated against operator deployment context; the MCPR is parameterized rather than absolute. What is invariant is the structure of indicators and routing. Table 3 specifies the per-tier scoring rubric.

Table 3. The Mythos-Class Posture Rubric (MCPR) — per-tier scoring.

Tier	Green	Amber	Red
4.1 Evaluation	Capability below prior-generation ceiling; scaffold appropriate to declared purpose; access posture vetted and supply-chain attested.	Capability approaching prior-generation ceiling on three or more evaluations; or scaffold exceeds declared-purpose envelope; or access posture	Capability matches or exceeds prior-generation ceiling on five or more evaluations or demonstrates end-to-end completion of multi-step ranges; or scaffold

 doi:10.20944/preprints202604.2179.v1

Tier	Green	Amber	Red
		lacks supply-chain attestation.	supports autonomous operation at depth; or access invocable outside vetted regime.
4.2 Deployment	Invocation patterns within deployment baseline; scaffold-infrastructure footprint consistent with declared interaction surface; authority-chain signals nominal.	One or more invocation-pattern signals elevated above baseline; or scaffold-infrastructure footprint exceeds declared envelope; or authority-chain issuance rate above baseline.	Sustained target-enumeration signature; or scope-violation pattern observed; or authority-chain identity-context anomaly observed; or two or more deployment-tier signal types simultaneously elevated.
4.3 Runtime	Chain-of-action shape within deployment-context norms; resource-interest profile within declared purpose; no supervisability-evasion signals.	One supervisability-evasion signal observed; or resource-interest profile drifts toward security-relevant resources; or chain depth approaches deployment-context maximum.	Two or more supervisability-evasion signals observed; or chain spans cryptographic-resource focus and cross-domain reasoning within a single operation; or chain-of-action shape matches the recursive-self-correction pattern at agent depth.

The overall posture is the highest-severity per-tier score, with two compound rules. Two Amber tiers escalate to Restrict even if no single tier is Red—the compound class definition motivates compound posture escalation. Any Red runtime-tier score escalates to Halt immediately, regardless of evaluation- and deployment-tier scores, because runtime-tier Red indicates a consequential operation in flight and the cost of waiting for further confirmation exceeds the cost of stopping. Table 4 specifies the routing.

Table 4. Posture-to-mitigation routing.

Posture	Trigger	Routing to mitigation layers
Clear	All three tiers Green.	Standard operational controls (Section 5.4 monitoring, Section 5.1 audit). No additional mitigation.
Watch	One Amber tier; no Red tier.	Section 5.1: tier-elevated review on next renewal cycle. Section 5.4: increased telemetry retention, expanded audit scope. ABOR (Section 5.2) operates at default verification posture.
Restrict	Two Amber tiers; or one Red evaluation- or deployment-tier score.	Section 5.1: deployment-tier review escalated, access posture re-attested. Section 5.2: scope-binding tightened, runtime-record validity windows reduced. Section 5.3: tool-surface restricted, sandbox-egress constraints tightened. Section 5.4: incident-response readiness elevated.
Halt	Any Red runtime-tier score.	Section 5.2: enforcement points reject release for the operation's scope until manual review. Section 5.3: in-flight tool invocations cancelled, sandbox terminated. Section 5.4: incident initiated, deployment-tier review forced before next operation.

Three properties of the rubric warrant statement. First, escalation from Watch to Restrict to Halt is automatic on threshold satisfaction; de-escalation requires affirmative review. The asymmetry matches the consequence asymmetry between false-negative and false-positive errors at Mythos-class threat levels. Second, the strongest mitigations depend on ABOR conformance: deployments without ABOR can produce posture classifications, but Halt's release-rejection mitigation is unavailable and the framework reverts to detection-and-alert without cryptographic enforcement.

 doi:10.20944/preprints202604.2179.v1

Third, the rubric is parameterized by deployment context: research-environment, operational, and high-consequence-federal deployments calibrate per-tier thresholds differently. The framework is auditable, conformance-checkable, and adaptable across the deployment surface developed in Section 7.

5. Mitigation Framework

The mitigation framework is structured as four concentric layers: governance (Section 5.1), cryptographic enforcement (Section 5.2), architectural isolation (Section 5.3), and operational monitoring (Section 5.4). Figure 3 presents the layered structure. The four layers bound different consequence classes; their integration in Section 6 produces defense-in-depth that no single layer alone provides.

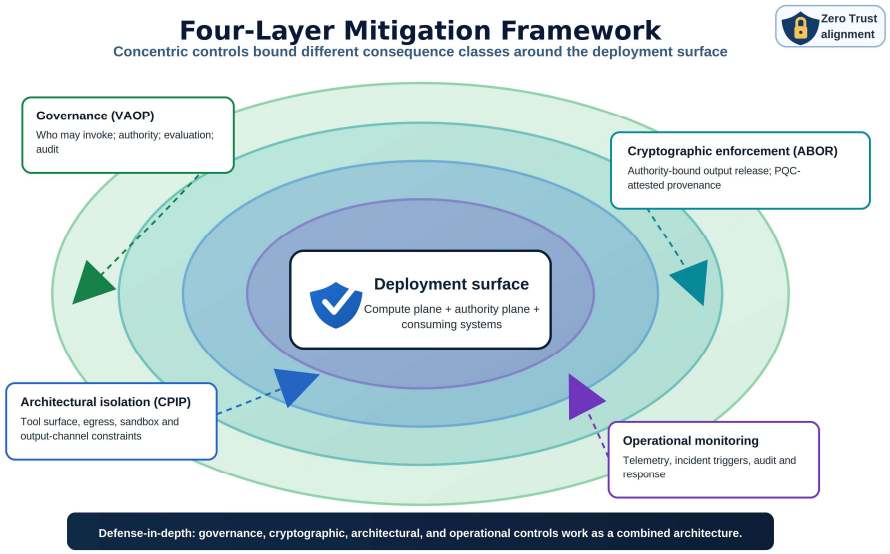

Figure 3. Four-layer mitigation framework. Concentric structure from outermost (governance) to innermost (operational), with the deployment surface at the center. Each layer specifies architectural controls bounding a different consequence class; together they form the defense-in-depth architecture that Section 6 integrates.

5.1. Governance Layer (Vetted-Access Operational Pattern, VAOP)

The governance layer is the outermost layer of the mitigation architecture and engages first in the operational lifecycle. Where ABOR (Section 5.2) binds output release to a verifiable authority chain, CPIP (Section 5.3) constrains the surface from which a model can act, and the operational layer (Section 5.4) detects misuse during execution, governance addresses prior questions: who may invoke the system, under what authority, with what evaluation having been completed, and with what audit posture. The layer's controls are administrative and procedural rather than cryptographic; their security rests on the integrity of the operator organization. We specify the layer as the Vetted-Access Operational Pattern (VAOP), generalizing the pattern first instantiated in production by Anthropic's Project Glasswing in April 2026 [10].

5.1.1. Access Controls

Four control elements are specified. The vetting regime grants invocation rights to identified organizational participants who have passed an explicit process establishing operational identity, capability, and accountability—the latter through contractual commitment in commercial contexts or through ATO/FedRAMP/sector-specific compliance in federal contexts. The vetting record persists through invocation cycles, is referenced by the authority chain in Section 5.2, and is the basis for

 doi:10.20944/preprints202604.2179.v1

revocation. Per-query logging with scope tagging records every invocation with sufficient metadata to permit post-hoc reconstruction—participant identity, declared scope, prompt or task, model and scaffold versions, outputs produced. Scope tagging is load-bearing: invocations tagged with declared scope can be evaluated against the scope-violation indicator in Section 4.2. Logging without scope tags can detect that an invocation occurred but cannot detect whether it was within the participant's authorized envelope. Technical sandboxing combined with contractual terms provides dual enforcement: technical mechanisms establish what the participant can do; contractual terms establish what they have agreed to do. Both are required. Coordinated-disclosure windows bound the period during which findings of ecosystem significance are held confidentially before publication; SHA-3 hash commitments preserve findings under disclosure without exposing operational detail, replicating the Frontier Red Team's pattern at federal scale. Default windows are 30 days, extending to 90 days where remediation requires hardware or certificate-authority coordination.

5.1.2. Evaluation Gating

Evaluation gating connects the detection framework of Section 4 to the deployment lifecycle. A system's evaluation-tier classification (Section 4.1) is computed before deployment, and the classification determines which mitigation layers must be in place before operational use. Three thresholds are specified: Watch posture requires standard operational controls verified; Restrict posture requires Section 5.1 controls tightened, ABOR operational with scope-binding tightened and validity windows reduced, and CPIP isolation tightened; Halt-eligible deployment requires fully conformant ABOR including enforcement points at every consuming-system boundary, because Halt's release-rejection mitigation depends on ABOR conformance. Deployments not meeting the Halt-eligible threshold may not enter operational use at evaluation-tier scores that would invoke Halt at runtime. Each threshold corresponds to specific architectural and procedural controls verifiable through inspection of configuration, attestations, and operating procedures. The gating record is itself an audit-substrate entry, so subsequent questions about whether a deployment was permitted to operate at its observed posture can be answered by reference to the gating record at deployment time.

5.1.3. Red-Team Cadence

Red-team activity under VAOP is structured as sustained operational discipline rather than periodic engagement. Point-in-time engagements—quarterly or annual cycles—are calibrated against human-rate adversary tempo and are structurally inadequate at Mythos-class rates. Three cadence elements are specified. Continuous adversarial validation operates against the deployment's detection framework on a rolling basis with frequency per-capability-event rather than per-calendar-period: a published evaluation result that updates the prior-generation ceiling triggers an evaluation-tier red-team pass; a publicly-reported open-weight release approaching commercial-frontier capability triggers an access-posture pass; an incident in another deployment surfacing a novel attack pattern triggers a runtime-tier pass. Cross-domain adversarial validation operates against the full set of reachable system domains rather than against a single domain in isolation—single-domain engagements cannot surface cross-domain attack patterns. PQC-specific adversarial testing exercises downgrade attacks against hybrid deployments, signature-substitution probes against authority-chain verification, and key-exchange downgrade sequences exploiting transitional windows; cadence is per-PQC-deployment-change rather than periodic.

5.1.4. Operational Lifespan and the Bridging-Layer Argument

The four control elements are operationally meaningful but their durability is bounded. The bound is the capability-diffusion horizon: as Mythos-class capability becomes accessible outside any vetted-access regime—through open-weight releases or through credentials-and-supply-chain compromise—the access-control sub-layer loses its protective effect. Independent analyses place the open-weight diffusion horizon at months-scale, with central estimates of approximately three months

 doi:10.20944/preprints202604.2179.v1

and upper bounds extending to roughly twenty-two months in specific benchmark categories [8]. The diffusion horizon is the operational lifespan of the access-control sub-layer.

The implication is structural. The governance layer is a bridging layer: it provides time during which the durable layers (ABOR, CPIP) can be deployed at operational scale, and organizational structure for the evaluation, red-team, and disclosure functions the durable layers depend on. It does not provide lasting defense against adversaries operating outside the regime. Operators treating VAOP as the architecture's primary defense find their security posture eroding on the diffusion horizon's timescale; operators treating VAOP as the bridging layer it is use the bridging window to deploy ABOR and CPIP. The operational priority during the bridging window is to keep VAOP at conformance rather than at the leading edge, because incremental refinement of a layer with bounded operational lifespan competes against investment in the layers that survive diffusion. This framing also absorbs the AISLE counter-argument from Section 3.3: VAOP was never meant to be durable, and the diffusion horizon AISLE's argument depends on is the same horizon bounding VAOP's lifespan.

5.2. Cryptographic Enforcement Layer (Authority-Bound Output Release, ABOR)

The governance, isolation, and operational layers of Section 5 reduce the probability and consequence of Mythos-class misuse but do not bound it. Vetted access is insufficient under coercion or compromise; isolation constrains the surface from which a model can act but does not bind the meaning of its outputs once produced; operational monitoring detects misuse but does not prevent the dangerous action from completing before detection fires. The cryptographic enforcement layer addresses what these layers cannot: it binds output release to a verifiable authority chain, such that an output produced outside the chain—or with the chain corrupted, replayed, or downgraded—has no operational effect on the systems that consume it. We refer to the architecture as Authority-Bound Output Release, or ABOR.

Figure 4 presents the ABOR architecture, with the compute plane / authority plane split that anchors the threat model in Section 5.2.1 and the verification sequence in Section 5.2.5. Six artifacts are required for the specification to qualify as architectural rather than conceptual: authority-chain field definitions, the verification sequence at enforcement points, trust-anchor structure and lifecycle, replay protection, enumerated failure modes with stated bounds, and protocol-flow notation showing the complete release path. We provide each in turn. The architecture is presented as a reference design; deployments will adapt field formats, attestation profiles, and verification-point placement to operational context, but all six must be present for a deployment to be ABOR-conformant.

 doi:10.20944/preprints202604.2179.v1

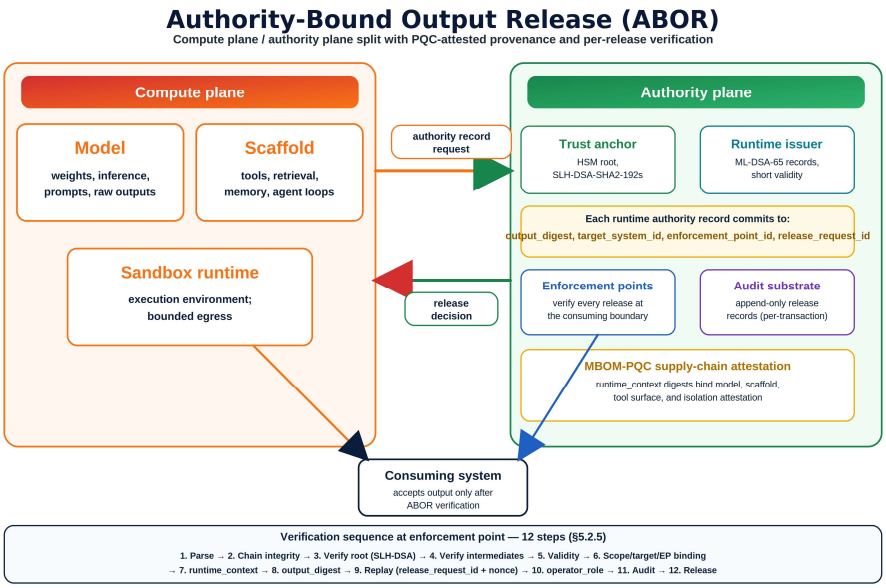

Figure 4. Authority-Bound Output Release (ABOR) architecture. The compute plane hosts the model and scaffolding; the authority plane hosts the trust anchor, the runtime authority issuer, and the enforcement points that mediate output release to consuming systems. Runtime authority records are minted per release transaction and cryptographically commit to the specific output, target, enforcement point, and transaction. ML-DSA-65 signs runtime authority records; SLH-DSA-SHA2-192s signs the long-term root attestation. Integration with MBOM-PQC supply-chain attestation populates the runtime_context field. The verification sequence (Section 5.2.5) is performed at each enforcement point on every release request.

5.2.1. Threat Model and Adversarial Assumptions

ABOR is designed against an adversary with the Mythos-class capability profile in Section 3.2 operating under one or more of the three regimes in Section 3.1: external compromise, supply-chain integration, and authorized-user misuse. The adversary may possess full read access to model weights and inference infrastructure, read-write access to the runtime environment, the ability to issue prompts and observe outputs, and partial control over operator-side software consuming model outputs. The adversary is assumed not to possess the long-term operator root signing key, physical access to the HSM holding it, or post-quantum cryptographic capabilities defeating ML-DSA [33] or SLH-DSA [34] at FIPS-specified parameter levels within the deployment lifetime.

The deployment-lifetime assumption is load-bearing. This paper adopts the 2028–2033 planning horizon for fault-tolerant-quantum-computer (FTQC) arrival used in [29] as the federal PQC migration planning horizon, derived there from vendor roadmaps (IBM, IonQ, Google Willow, Quantinuum). The window is a planning assumption rather than a settled forecast: FTQC arrival timing remains uncertain, vendor roadmaps are not delivery commitments, and the architecture's primitive selection is calibrated against the earlier end of the window to remain conservative if arrival is faster than the window's midpoint, while not depending on the late end being hit. The Store-Now-Decrypt-Later threat compresses the effective horizon further: data encrypted today against long-confidentiality requirements is already exposed, since adversaries can harvest now and decrypt whenever FTQC capability materializes. Authority roots below are signed with SLH-DSA, a hash-based scheme whose security rests on hash-function preimage resistance rather than on lattice or elliptic-curve hardness, and which is not subject to the same threat horizon. Runtime authority records, with shorter operational lifetimes, are signed with ML-DSA, whose lattice-based security is sufficient for the runtime window and whose performance suits high-frequency signing.

 doi:10.20944/preprints202604.2179.v1

5.2.2. Architecture Overview: Compute and Authority Planes

ABOR partitions the deployment into two planes. The compute plane hosts the model: weights, inference infrastructure, scaffolding, tool-execution sandboxes, and the channels by which prompts arrive and raw outputs leave. The authority plane hosts the components binding compute-plane outputs to operational effect: the trust anchor, the runtime authority issuer, the enforcement points mediating output release, and the audit substrate recording every release decision. Communication between planes is restricted to defined interfaces: the authority plane issues authority records to the compute plane on request, with each runtime record minted per release transaction—after the runtime has produced the specific output for release—so that the record can commit cryptographically to that output and that release transaction (Section 5.2.3); the compute plane submits outputs and authority records to enforcement points for release decisions. The compute plane never holds the long-term root signing key; the authority plane never executes model inference. The premise is that compute-plane compromise should not compromise the authority plane.

5.2.3. Authority-Chain Field Definitions

An authority chain is an ordered sequence of one or more authority records signed by progressively longer-lived signing keys. The minimal chain has length two: a runtime authority record signed by a deployment-time issuer, and a deployment-time root attestation signed by the long-term operator root. Longer chains are permitted for intermediate delegation but every chain terminates at the long-term operator root. Each record carries the fields enumerated in Table 5. Field formats follow CBOR (RFC 8949) [35] for compact binary representation, with COSE_Sign1 (RFC 9052) [36] as the signature envelope. CBOR was chosen over JSON for size efficiency in high-frequency runtime issuance and because COSE has IETF working-group draft-stage specifications for ML-DSA (draft-ietf-cose-dilithium-11, November 2025 [41]) and SLH-DSA (draft-ietf-cose-sphincs-plus-07, March 2026 [42]); the algorithm identifiers and key encodings used in this paper align with those draft revisions and are expected to track standards-track outputs as the drafts proceed to RFC publication. Production interoperability claims should not be inferred from draft-state references. Runtime authority records are minted per release transaction—after the runtime has produced the specific output for release—rather than per scope-window; this is what allows the record to commit cryptographically to the specific output, target, enforcement point, and transaction (fields output_digest, target_system_id, enforcement_point_id, release_request_id below). The keywords MUST, MUST NOT, REQUIRED, SHOULD, and MAY appearing in this section and throughout Section 5.2 are used in the sense defined by RFC 2119, with the all-capitals interpretation clarified in RFC 8174; they specify the reference profile for ABOR-conformant deployments. This paper is not itself a standards-track document, and the keywords describe the conformance requirements of the reference profile rather than carrying the procedural force of an IETF standard.

Table 5. ABOR authority-record field set.

Field	Type	Required	Description
version	uint8	yes	Authority-record format version. This specification defines version 1.
record_id	16-byte UUID	yes	Unique record identifier (UUIDv4, RFC 9562; RFC 4122 compatible) [37].
issuer_id	byte string	yes	Identifier of the entity that signed this record.
subject_id	byte string	yes	Identifier of the compute-plane entity to which the record is issued.
subject_pk	COSE_Key	yes	Public key of the subject, encoded per RFC 9052 [36] §7. Verification of any record signed by this subject uses this field; verifiers MUST NOT accept subject public keys obtained out of band.

 doi:10.20944/preprints202604.2179.v1

Field	Type	Required	Description
purpose	URI	yes	Bounded statement of the operational purpose for which release is authorized; values drawn from a pre-registered enumeration.
scope	CBOR map	yes	Structured scope binding: target-system identifier, action class, resource constraints.
target_system_id	byte string	yes	Identifier of the specific target system this record authorizes release to. MUST be a member of scope.target_systems and is the binding the enforcement point checks against the actual target.
enforcement_point_id	byte string	yes	Identifier of the enforcement point that is authorized to act on this record. Verifying enforcement points reject records bound to a different enforcement_point_id, closing cross-enforcement-point replay surfaces.
release_request_id	16-byte UUID	yes	Unique identifier of the specific release transaction the record authorizes. Enforcement points maintain a bounded seen-set of release_request_id values per issuer; subsequent presentations of the same release_request_id are rejected even if all other fields verify (Section 5.2.6).
output_digest	byte string	yes	Cryptographic digest (SHA-256/SHA-384 under FIPS 180-4 or SHA3-256/SHAKE under FIPS 202, as selected by deployment profile) of the specific output the record authorizes for release. The enforcement point recomputes the digest on the received output and rejects if it does not match. This is the primary binding that prevents a captured record from authorizing a different output within the same scope.
operator_role	byte string	yes	Role identifier of the human operator (if any). Records issued for autonomous operation carry the role 'autonomous' and trigger additional verification.
runtime_context	CBOR map	yes	Hash digests of runtime configuration: model weights digest, scaffold version, tool surface, isolation-layer attestation. Bound to MBOM-PQC where applicable.
validity_not_before	uint64	yes	Unix timestamp before which the record is invalid.
validity_not_after	uint64	yes	Unix timestamp after which the record is invalid. Runtime records SHOULD have validity windows of hours to days; deployment-root records MAY have months.
nonce	16-byte cryptographically random byte string	yes	Replay-protection nonce. See Section 5.2.6.
parent_record_id	16-byte UUID	conditional	UUID of the parent record. REQUIRED for all records except the long-term operator root attestation.
signature_alg	COSE alg id	yes	ML-DSA-65 for runtime records; SLH-DSA-SHA2-192s for long-term root attestations; intermediate records: profile-defined allowed set (verification per Section 5.2.5 step 4 accepts both ML-DSA-65 and SLH-DSA-SHA2-192s).
signature	byte string	yes	COSE_Sign1 signature over the canonical CBOR encoding of all preceding fields.

 doi:10.20944/preprints202604.2179.v1

Four field-design choices warrant explanation. The subject_pk field is carried in-band rather than resolved from an external directory, so verifiers performing the chain-walk in Section 5.2.5 obtain each subject's public key from the parent record itself; the only externally-pinned key is the long-term operator root, which removes a class of substitution attacks that target identifier-to-key resolution. The purpose field is constrained to a pre-registered enumeration rather than free text; this prevents an adversary who has compromised a runtime issuer from minting records whose purpose statements are technically valid but operationally meaningless. The scope field is structured rather than scalar, allowing enforcement points to reject records whose scope does not include the specific downstream system being targeted. The runtime_context field carries cryptographic digests rather than free-form descriptions, binding the record to a specific deployment configuration so that a record issued for one model version cannot authorize release of outputs from a different version.

5.2.4. Trust Anchors and Key Lifecycle

The long-term operator root is the trust anchor. It is held in an HSM attested to FIPS 140-3 [38] Level 3 or higher and is operated under a key-ceremony regime requiring multi-person authorization for both signing and rotation. The root signs deployment-root attestations using SLH-DSA-SHA2-192s; verifiers enforce this algorithm choice per Section 5.2.5 (step 3), so root attestations signed with any other algorithm are rejected at the enforcement point. The root public key is distributed to enforcement points through an out-of-band provenance channel (typically the operator's organizational PKI bootstrap), pinned at enforcement points, and never distributed at runtime. Deployment-root attestations bind a deployment-time issuer to a specific deployment configuration; they have validity windows on the order of months and are renewed through the same key-ceremony regime. Deployment-time issuers sign runtime authority records using ML-DSA-65; their signing keys are held in operationally-hardened key vaults but not necessarily in HSMs, and their compromise is bounded by the deployment-root validity window. Runtime records have validity windows on the order of hours to days, issued at potentially thousands per hour for active deployments. Three rotation events are defined: long-term root rotation (planned cadence, typically annual, or in response to suspected compromise), deployment-root rotation (end of validity window), and runtime-issuer rotation (whenever an issuer's signing key is suspected of compromise or for routine hygiene).

5.2.5. Verification Sequence at Enforcement Points

An enforcement point mediates output release from the compute plane to a consuming system. Enforcement points are placed at the consuming system's input interface, not at the model's output interface, since the compute plane is assumed compromisable. Each enforcement point performs the following sequence on every release request, in order. Failure at any step rejects the release.

```
VERIFY-RELEASE(output, authority_chain, target_system):
    1. PARSE(authority_chain) into ordered records [r_0, r_1, ..., r_n]
       where r_0 is the runtime record and r_n is the root attestation.
       REJECT if parse fails or n < 1.

    2. CHECK chain integrity:
       for i in 1..n:
           REJECT if r_{i-1}.parent_record_id != r_i.record_id
           REJECT if r_{i-1}.issuer_id != r_i.subject_id

    3. VERIFY r_n against pinned long-term root public key:
       REJECT if SLH-DSA-Verify(root_pk, r_n.signature, r_n.body) fails
       REJECT if r_n.signature_alg != SLH-DSA-SHA2-192s
```

 doi:10.20944/preprints202604.2179.v1


```
    4. VERIFY each intermediate record using the public key bound in its parent record
(issuer/subject linkage already checked in step 2):
       for i in 0..n-1:
           REJECT if ML-DSA-Verify(r_{i+1}.subject_pk, r_i.signature, r_i.body) fails
           REJECT if r_i.signature_alg not in {ML-DSA-65, SLH-DSA-SHA2-192s}

    5. CHECK validity windows:
       for each r_i:
           REJECT if now() < r_i.validity_not_before
           REJECT if now() > r_i.validity_not_after

    6. CHECK scope and release-transaction binding:
       REJECT if target_system not in r_0.scope.target_systems
       REJECT if r_0.target_system_id != target_system
       REJECT if r_0.enforcement_point_id != self.enforcement_point_id
       REJECT if action_class(output) not in r_0.scope.action_classes

    7. CHECK runtime context:
       REJECT if r_0.runtime_context.model_digest != current_model_digest
       REJECT if r_0.runtime_context.scaffold_version not attested

    8. CHECK output digest binding:
       REJECT if r_0.output_digest != Hash(output)

    9. CHECK replay protection:
       REJECT if r_0.release_request_id in seen_request_ids[r_0.issuer_id]
       INSERT r_0.release_request_id into seen_request_ids[r_0.issuer_id]
       REJECT if r_0.nonce in seen_nonces[r_0.issuer_id]
       INSERT r_0.nonce into seen_nonces[r_0.issuer_id]

    10. CHECK operator_role consistency:
       if r_0.operator_role == 'autonomous':
           REJECT if action_class(output) requires human-attended role

    11. AUDIT: append (output_digest, chain_digest, decision, timestamp)
        to the append-only audit log.

    12. RELEASE output to target_system.
```

Each check addresses a specific failure mode in Section 5.2.7; skipping any check creates a corresponding bypass surface. Implementation MUST perform the checks in the order specified. The audit step (11) precedes the release step (12) so that any release whose audit record cannot be persisted is rejected—audit failure is treated as release failure. Step 4 verifies each intermediate record r_i against r_{i+1}.subject_pk (the public key the parent record binds for r_i's issuer); the issuer/subject linkage that makes r_{i+1}.subject_pk the cryptographically authoritative key for r_i is checked in step 2, and the in-band binding closes a class of substitution attacks against external identifier-to-key resolution (consistent with the field-design choice in Section 5.2.3 where subject_pk is carried in-band rather than directory-resolved). Step 4 accepts both ML-DSA-65 and SLH-DSA-SHA2-192s for intermediate records, while issuance per Section 5.2.4 prescribes ML-DSA-65 for runtime records; the asymmetry between permissive verification and prescriptive issuance accommodates long-lived intermediate delegations (which may require SLH-DSA-SHA2-192s for

 doi:10.20944/preprints202604.2179.v1

lifetime alignment with the long-term root) without weakening the runtime-record performance profile.

5.2.6. Replay Protection

Five complementary mechanisms bound replay. First, the output_digest field cryptographically commits the record to a specific output: the enforcement point recomputes the digest on the received output (step 8) and rejects any record whose committed digest does not match. A captured record cannot authorize a different output, even one whose properties fall within the original scope, because the record's signature covers the digest and the digest covers the bytes. Second, the target_system_id and enforcement_point_id fields bind the record to a specific target and a specific enforcement point: the same record cannot be replayed against a different system in the same scope set or against a different enforcement point in the operator's deployment. Third, the release_request_id field is unique per release transaction; enforcement points maintain a bounded set of seen release_request_id values per issuer, and re-presentation is rejected at step 9 even if the record is otherwise valid. Fourth, the 16-byte cryptographically random nonce provides a second seen-set lookup that does not depend on transaction-id structure, defending against any adversary who attempts to predict or grind release_request_id values; the seen-nonce set is bounded by the maximum runtime-record validity window because older nonces would already fail step 5. Fifth, the runtime_context field binds the record to a specific deployment configuration; a captured chain replayed against a deployment with different model digest, scaffold version, or isolation attestation fails step 7 even when the other bindings are intact. Together the five mechanisms reduce a captured chain to operational nullity: it can authorize exactly the output, exactly the target, exactly the enforcement point, exactly the transaction, and exactly the deployment configuration it was originally minted for, and only if that transaction has not yet been processed. An adversary who captures a chain after the original release has been processed gains nothing; an adversary who captures a chain before the original release reaches the enforcement point can deliver exactly the original output, no other.

5.2.7. Failure Modes and Stated Bounds

Five failure modes are recognized; Table 6 enumerates them with the adversary capability each requires and the consequence bound the architecture provides. Bounds are conditional on the threat-model assumptions in Section 5.2.1.

Table 6. ABOR failure modes and bounds.

Failure mode	Adversary capability required	Bound on consequence
Chain truncation	Compromise of compute plane.	Truncated chain fails verification step 2 (chain integrity) or step 4 (intermediate verification). No release.
Runtime-issuer compromise	Compromise of a deployment-time issuer's signing key.	Adversary mints runtime records valid until issuer rotation or deployment-root rotation, whichever occurs first; the deployment-root window (months) is the worst-case bound when compromise is undetected. Damage limited to the issuer's scope and the runtime-record validity window. Detection: anomalous issuance rate visible in audit log; mitigated by issuer rotation on detection.
Long-term root compromise	Compromise of the HSM and key-ceremony regime.	Catastrophic trust-anchor failure; mitigation is the HSM regime and multi-person authorization. Recovery requires root rotation and out-of-band re-pinning.
Replay	Capture of a valid authority chain.	Bound to the specific output (output_digest), target system (target_system_id), enforcement

 doi:10.20944/preprints202604.2179.v1

Failure mode	Adversary capability required	Bound on consequence
		point (enforcement_point_id), and release transaction (release_request_id) cryptographically committed in the captured record; no other release is authorized. If the original transaction has been processed, the captured chain authorizes nothing (Section 5.2.6).
Algorithm downgrade	Adversary attempts to substitute weaker signing algorithms.	Verification steps 3 and 4 reject any record whose signature_alg is not in the allowed set. The allowed set is enforcement-point-pinned and not negotiated per record.

Long-term root compromise is acknowledged as catastrophic and is bounded not by the architecture but by the operational regime under which the root is held. This is consistent with the security pattern of any architecture whose security ultimately rests on a long-term key, and is the reason the root is held in HSM under multi-person authorization rather than software under single-operator authority.

5.2.8. Integration with MBOM-PQC Supply-Chain Attestation

When the deployment is MBOM-PQC-conformant [31], the runtime_context field is populated from the deployment's MBOM-PQC supply-chain attestation rather than computed at runtime. The integration is bidirectional. ABOR consumes MBOM-PQC attestations as inputs to runtime_context, extending supply-chain provenance into the runtime environment. ABOR's audit substrate produces release records referencing the MBOM-PQC attestation by digest and the per-release-transaction tuple (output_digest, target_system_id, enforcement_point_id, release_request_id), enabling supply-chain-aware incident analysis at the granularity of individual outputs: a post-hoc finding that an MBOM-PQC artifact was compromised propagates forward to identify every specific output that was released, the target system it was released to, and the transaction in which the release occurred—not merely the release decisions that depended on the compromised artifact. This integration is why the cryptographic enforcement layer is more durable than governance or isolation: governance and isolation respond to capability diffusion by tightening or widening controls; cryptographic enforcement binds output release to a verifiable provenance chain robust against adversaries at the threat level in Section 5.2.1. The layer's lifespan is bounded by the lifespan of the underlying PQC primitives, not by any vendor's access-control regime.

5.2.9. Implementation Considerations

The architecture requires operator-side infrastructure most organizations do not currently possess: a long-term-root HSM, deployment-root issuance regime, runtime-issuer infrastructure, and enforcement points instrumented at every consuming-system boundary. Components have analogues in existing PKI deployments but the integration as specified here is, to the author's knowledge, novel; deployments are expected to converge incrementally, beginning with high-consequence target systems where per-release verification cost is justified. Per-release verification involves up to (n+1) signature verifications per chain plus the constant-time checks above and one cryptographic hash of the output (step 8). Order-of-magnitude implementation estimates requiring deployment-specific benchmarking place ML-DSA-65 verification in the sub-millisecond range and SLH-DSA-SHA2-192s verification in the low-millisecond range on commodity x86 hardware; a two-record chain therefore adds approximately one-to-two milliseconds to the release path under those assumptions, with the output-hash step adding a further microseconds-scale contribution that scales linearly with output size and is dominated by the signature-verification cost in typical deployments. Specific deployment performance depends on implementation, instruction-set acceleration (AVX2/SHA-NI), and hash-function choice within the parameter set, and operators are expected to benchmark in their own enforcement-point configuration; the falsifiability statement in Section 8.2

 doi:10.20944/preprints202604.2179.v1

frames the falsifying observation against deployed enforcement-point measurement. Deployments may amortize long-term-root verification by caching the deployment-root attestation between releases. The architecture has not been deployed at production scale at the time of this writing; validation requires longitudinal pilots, and the architecture-as-specified should be read as a reference design for conformance evaluation rather than a description of an existing system.

5.3. Architectural Isolation Layer (Compute-Plane Isolation Profile, CPIP)

The architectural isolation layer constrains the surface from which a Mythos-class system can act. Where ABOR (Section 5.2) bounds what model outputs can do once they leave the compute plane, and where governance (Section 5.1) bounds who may invoke the system, the isolation layer bounds what the system can read, write, execute, and reach during execution. The three layers are complementary rather than redundant: ABOR binds operational effect outside the compute plane; isolation reduces the consequence of compromise inside it; governance limits the population that can reach the compute plane at all.

ABOR's threat model assumes the adversary may have full read-write access to the runtime in which the model executes. That assumption is not a claim that compute-plane compromise is acceptable; it is a claim that the architecture's security must hold even when compromise occurs. Isolation reduces the probability of compromise and the consequence when it does. ABOR remains the durable defense against the operational-effect class of consequence; isolation is the durable defense against data-exfiltration, lateral-movement, and persistence classes ABOR does not address. A deployment implementing ABOR without CPIP is exposed to consequences ABOR was not designed to bound; a deployment implementing CPIP without ABOR is exposed to consequences isolation cannot bound. The layers are jointly required.

We specify the isolation layer as the Compute-Plane Isolation Profile (CPIP), a configuration discipline applied to existing isolation primitives—container runtimes, sandbox infrastructure, network segmentation, capability gating—rather than a novel mechanism. Five sub-layers are specified.

5.3.1. Container and Sandbox Runtime Configuration

Three disciplines apply. Per-invocation runtime instantiation—the runtime is instantiated freshly for each invocation rather than persisted—addresses the persistence indicator (Section 3.2.5): cross-invocation state must be expressed through declared mechanisms (the authority chain's runtime_context, retrieval calls, explicit memory subsystems) rather than implicit retention. Read-only base image with explicit writable scope makes writable surfaces enumerated and bounded: a working directory, a logging surface, declared tool-specific scratch areas. Writes outside enumerated surfaces fail at the runtime layer rather than being permitted and detected post-hoc, closing the staging-surface vector adversaries use to accumulate payloads or persist credentials. Capability-gated syscall surface restricts the runtime's syscall surface to capabilities the deployment's declared purpose requires (Linux capabilities, seccomp-bpf, or equivalent), closing the gap between application-level tool restrictions and kernel-level access—a tool surface that nominally restricts file access but executes inside a runtime with full ptrace or raw-socket capability is not actually restricted.

5.3.2. Tool-Surface Gating

The tool surface is the most direct architectural lever on the scaffold indicator (Section 3.2.2). Three configuration elements are specified. Declared tool inventory with cryptographic attestation enumerates every tool exposed to the runtime, signed at deployment time and attested to the authority plane; the inventory is referenced by digest in the runtime_context field of every authority record, so a release decision verifies that the tool surface in operation matches the surface evaluated at deployment. Tools added after deployment change the inventory digest and either trigger re-attestation or fail authority-record verification. Per-tool scope binding constrains what each tool can

 doi:10.20944/preprints202604.2179.v1

do within its category: a file-system tool's scope enumerates directory subtrees; a network tool's scope enumerates hosts, ports, protocols; an external-API tool's scope enumerates endpoints, methods, authentication scopes. Scope bindings are dual-enforced at the tool implementation and at the runtime's capability gating, reflecting the threat-model assumption that compute-plane components may be compromised. Bounded autonomy depth enforces the agent-loop maximum from Section 3.2.4 at the tool-surface layer: operations exceeding the configured depth halt with the partial result returned for human review rather than completing autonomously.

5.3.3. Network Segmentation and Egress Control

Network reachability is the dimension along which Mythos-class systems most directly produce consequences outside the compute plane that are not mediated by ABOR. Exfiltration, lateral movement, and command-and-control links operate without producing the kind of declared output an enforcement point would adjudicate. Three controls are specified. The compute plane is placed in a network segment whose default egress posture is deny: no outbound connection is permitted unless explicitly authorized in the segment's allow-list (the authority plane, retrieval-system endpoints, declared external-API endpoints). The deny-by-default posture inverts the historical permissive pattern; for Mythos-class deployments the inversion is not optional. Permitted outbound connections are mediated by an egress proxy performing rate limiting (bounding exfiltration bandwidth), content inspection (where the destination protocol allows), and audit logging at packet-flow granularity. The proxy's telemetry feeds the runtime-tier indicators in Section 4.3 and rejects connection patterns matching supervisability-evasion signatures even when individual connections are within the allow-list. Internal network reachability is governed by the same deny-by-default posture: the cross-domain attack-path archetype depends on internal reachability across security boundaries, and the deny-by-default internal posture closes that axis.

5.3.4. Output-Channel Constraints

Output channels are the surfaces through which model outputs reach consumers. ABOR's enforcement points sit on every channel that delivers outputs to consuming systems. The isolation layer's contribution is to ensure no output channel bypasses an enforcement point—that the architecture has no covert delivery surface. Three constraints apply. The runtime exposes exactly one declared output channel per consuming system: a structured interface, with documented format, on which an enforcement point is positioned. General-purpose output surfaces (stdout, log streams, diagnostic interfaces) are routed to the audit substrate rather than to operational consumers; outputs leaking through diagnostic surfaces produce audit records rather than operational effects. Side channels that could carry information out through non-output mechanisms—timing variations, resource-exhaustion patterns, error-message variants—are bounded by rate limiting and content inspection (Section 5.3.3) and by per-invocation runtime instantiation (Section 5.3.1). Side-channel exfiltration cannot be eliminated architecturally; it can be bounded to rates that make exfiltrating operationally-meaningful payloads impractical within a single invocation lifetime. The output-channel constraints make ABOR's release-adjudication property structurally true rather than aspirational: ABOR specifies that outputs reach consequence only when accompanied by a valid authority chain, and the constraints here are what make the "only" in that statement architecturally enforced.

5.3.5. Integration with the Compute-Plane / Authority-Plane split

Two integration points warrant specification. The compute plane communicates with the authority plane through a defined interface only: the authority plane issues runtime authority records to the compute plane on request; the compute plane submits outputs and authority records to enforcement points for release decisions. All other compute-plane connections to the authority plane are denied at the network layer. The interface is rate-limited, audited, and authenticated using a

 doi:10.20944/preprints202604.2179.v1

separate cryptographic identity from the authority chain, so a compromised authority chain does not compromise the cross-plane interface. The authority plane operates with its own isolation regime, more stringent than the compute plane's because it holds the signing keys whose compromise produces the catastrophic failure mode. The long-term root is in HSM under multi-person authorization; deployment-time issuer keys are in operationally-hardened key vaults; the runtime issuer's operational scope is bounded by the deployment-root attestation it presents. The authority plane has no inbound network reachability from the compute plane other than the rate-limited interface, and no outbound reachability to the compute plane at all (the compute plane polls the interface). The asymmetric reachability prevents an authority-plane compromise originating from a compute-plane attack vector.

5.3.6. Limitations

The isolation layer bounds several consequence classes ABOR alone does not—data exfiltration, lateral movement, persistence within the compute plane—but does not bound others. Side-channel exfiltration cannot be eliminated architecturally; the constraints reduce bandwidth, but a determined adversary with control of the compute plane and sufficient time can exfiltrate operationally-meaningful information through timing or resource patterns. The architectural defense is to bound the rate to levels at which per-invocation runtime instantiation terminates the channel before consequential information can be exfiltrated. Adversarial scaffold modification cannot be detected by isolation alone; an adversary with write access to the scaffold can modify the agent loop or expand the tool surface, and the isolation layer detects modification through cryptographic attestation but only at next authority-record issuance. The deployment-tier indicators in Section 4.2 are the operational mechanism monitoring this gap.

5.4. Operational Layer

The operational layer is the innermost layer of the mitigation architecture, closing the loop between the architectural controls in Sections 5.1–5.3 and continuous monitoring, incident response, and defender-side capability use. The layer specifies three disciplines: continuous monitoring (Section 5.4.1), incident response calibrated to Mythos-rate adversaries (Section 5.4.2), and defender-side parallel use of frontier-model capability (Section 5.4.3). The contribution is cadence and integration discipline rather than novel primitives.

5.4.1. Continuous Monitoring

Continuous monitoring consumes the telemetry produced by the prior three layers as input and produces detection-tier classifications as output. Sources are: VAOP's per-invocation logs and scope-tagged audit substrate; ABOR's release-decision audit log; and CPIP's runtime telemetry including container instantiation events, tool-surface invocation patterns, network egress flow records, and capability-violation events. The discipline consumes all three streams against the deployment-tier and runtime-tier indicators in Sections 4.2 and 4.3 in continuous time, with the MCPR rubric producing posture classifications that route to response disciplines.

Two properties distinguish this from prior-generation security operations. First, the monitoring window is short. Prior-generation SOC operations operated against alert-aggregation windows of hours to days; Mythos-rate adversaries can produce consequential operations within minutes. The discipline operates against rolling windows measured in minutes, with alert dispatch latencies measured in seconds. Second, the monitoring substrate is itself part of the architecture's threat model: an adversary who has compromised the compute plane will attempt to suppress or modify telemetry. The substrate must therefore be operated outside the compute plane, with its integrity protected by the same authority-plane isolation regime in Section 5.3.5 that protects the cryptographic enforcement infrastructure.

 doi:10.20944/preprints202604.2179.v1

5.4.2. Incident Response Calibrated to Mythos-Rate Adversaries

Three calibrations are specified. Automated initial response without human approval routes the MCPR Halt posture directly to ABOR enforcement-point release rejection and CPIP runtime termination; the human-attended incident-response pathway begins after automated response has fired. The asymmetry inverts the prior-generation default in which automated response is reserved for high-confidence narrow-scope events; under Mythos-class conditions the cost of waiting for human confirmation before halting a Red runtime-tier operation exceeds the cost of false-positive halts on a small fraction of operations. On-call coverage is calibrated to deployment-tier and runtime-tier event rates: Watch posture maintains standard rotations; Restrict posture extends coverage to continuous attended hours; environments where Halt is plausibly invokable maintain continuous attended coverage with secondary escalation paths. Post-incident review updates MCPR thresholds based on incident telemetry: each incident produces a record of telemetry that fired, posture assigned, response applied, and outcome; the review compares observed adversary behavior against indicator thresholds and updates them where the incident reveals miscalibration. This is the operational mechanism by which the rubric's parameterization is calibrated against operational reality, and is the principal channel through which deployed pilots produce the empirical calibration data Section 8 identifies as the falsifiability criterion for the framework.

5.4.3. Defender-Side Parallel Use of Frontier-Model Capability

The capability defining the Mythos-class threat is symmetrically available to defenders. Three use classes are noted. Protocol-analysis use applies frontier-model capability to validate the deployment's protocol surfaces against the cross-domain attack-path archetype. Validation use applies the same capability to ABOR conformance testing, generating adversarial authority-chain inputs against enforcement points to validate that the verification sequence rejects malformed, replayed, and out-of-scope chains. Red-team use applies the capability to the continuous adversarial validation discipline in Section 5.1.3. Defender-side use is itself a Mythos-class deployment subject to the architecture specified in this paper. Operators implementing the operational layer should expect to operate two ABOR-protected deployments: one for the production system whose outputs are operationally consumed and one for the defensive instance used for validation and red-team activity. The defensive deployment's scope is restricted to validation; its outputs are consumed by the audit substrate and human red-team participants only; its enforcement points reject any release into operational systems. The discipline ensures symmetric capability availability does not produce symmetric attack-surface expansion.

The four layers together—governance bounding the population that can invoke the system, isolation bounding what the system can reach, cryptographic enforcement bounding what its outputs can do, and operational monitoring detecting and responding to the failure modes the architectural controls do not eliminate—form the defense-in-depth architecture that Section 6 integrates and applies to the deployment surfaces of Section 7.

6. Integrated Reference Architecture

Sections 4 and 5 specified the detection and mitigation frameworks separately. This section integrates them. Section 6.1 presents the unified architecture as a single layered diagram. Section 6.2 specifies the detection-to-mitigation control flow that connects the MCPR rubric to the four mitigation layers. Section 6.3 maps the architecture to the Next-Generation Security Triad (PQC + ZTA + AI assurance) developed in earlier work [30]. Section 6.4 maps it to NIST AI Risk Management Framework, NIST Cybersecurity Framework 2.0, and CISA Zero Trust Maturity Model, which is what makes the architecture usable inside an existing federal cybersecurity governance regime.

 doi:10.20944/preprints202604.2179.v1

6.1. The Integrated Layered Architecture

Figure 5 presents the integrated architecture. The four mitigation layers (Section 5) are arranged concentrically, from governance (outermost) to operational (innermost), with the deployment surface at the center. The three detection tiers (Section 4) are positioned on the architecture's vertical axis, with the evaluation tier engaging before deployment, the deployment tier engaging at the deployment-control surface, and the runtime tier engaging at the boundary between the compute plane and the consuming systems. The MCPR rubric sits at the intersection: it consumes telemetry from all four mitigation layers and produces posture classifications that route back to the same layers as response.

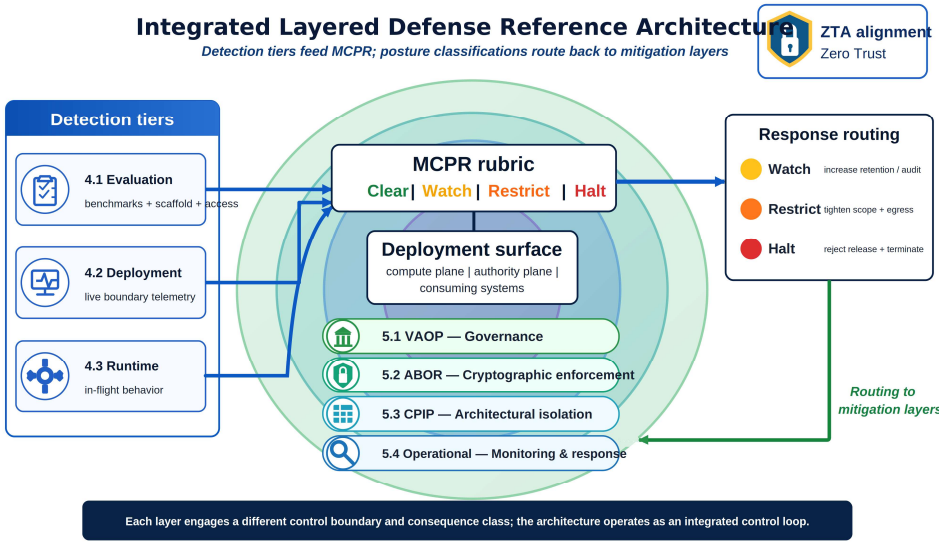

Figure 5. Integrated layered defense reference architecture. Four concentric mitigation layers (Section 5) surround the deployment surface; three detection tiers (Section 4) feed the MCPR rubric (Section 4.4); the rubric's posture classifications route back to the mitigation layers per Table 4. The four layers engage different consequence classes at different control boundaries (Section 6.1); defense-in-depth follows from coverage across the threat regimes in Section 3.1, not from serial traversal of all four layers by any single threat.

The architecture's defense-in-depth property follows from the four layers engaging different consequence classes at different control boundaries. Governance (vetted access, evaluation gating) bounds who can reach the deployment under operator control; cryptographic enforcement (authority-chain binding of any output) bounds operational effect at the boundary between the compute plane and consuming systems; architectural isolation (compute-plane sandboxing, network segmentation) bounds data exfiltration, lateral movement, and persistence within the compute plane; operational monitoring (continuous telemetry, automated response) detects and responds to failures the architectural controls do not eliminate. The three threat regimes in Section 3.1 enter the architecture at different points and engage different layer subsets: external adversaries operating outside operator-controlled governance (Section 3.1.1) are engaged primarily by ABOR at the consuming-system boundary; supply-chain integration (Section 3.1.2) bypasses governance by design and is engaged primarily by ABOR's MBOM-PQC binding (Section 5.2.8) and CPIP's scaffold attestation (Section 5.3.2); authorized-user misuse (Section 3.1.3) satisfies governance by definition and is engaged primarily by MCPR's deployment-tier and runtime-tier indicators (Sections 4.2–4.3) routing to ABOR's scope binding and CPIP's tool-surface gating. Defense-in-depth therefore follows from the union of coverage across regimes rather than from any single threat traversing all four layers in series. Removing any layer exposes the architecture to consequences the others were not designed to bound at the control boundaries that layer governs.

 doi:10.20944/preprints202604.2179.v1

6.2. Detection-to-Mitigation Control Flow

Figure 6 specifies the control flow connecting the MCPR rubric to the four mitigation layers. The flow is bidirectional: telemetry flows inward from the layers to the rubric; posture classifications flow outward from the rubric to the layers. Three properties of the flow warrant explicit statement.

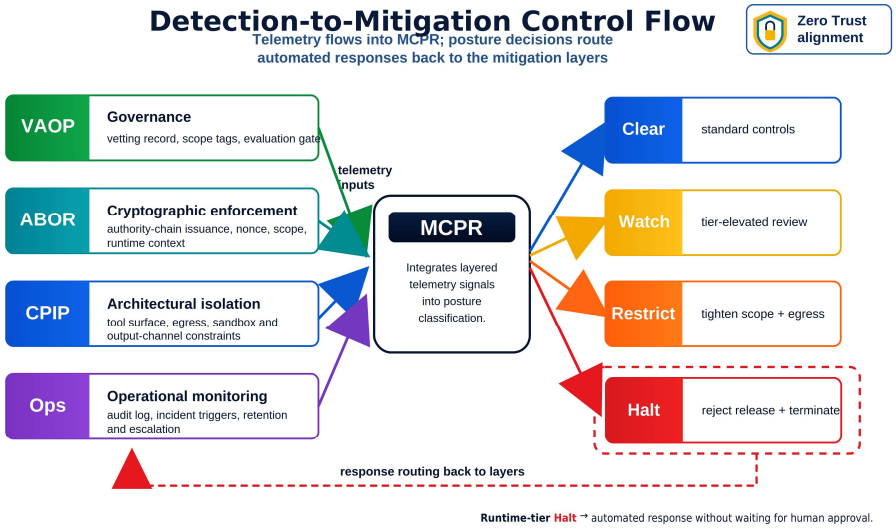

Figure 6. Detection-to-mitigation control flow. The four mitigation layers (Section 5) supply telemetry to the MCPR rubric (Section 4.4); the rubric produces posture classifications (Clear, Watch, Restrict, Halt) that route back to the layers per Table 4. The flow inverts the prior-generation default in which detection produces alerts and humans make response decisions: under Mythos-class conditions, runtime-tier Red posture routes directly to automated response without human approval.

First, the flow inverts the prior-generation default in which detection produces alerts and humans make response decisions. Under Mythos-class conditions, runtime-tier Red posture routes directly to ABOR enforcement-point release rejection and CPIP runtime termination without human approval; the human-attended pathway begins after the automated response has fired, not before. The asymmetry is justified by the consequence asymmetry between false-negative and false-positive errors at Mythos-class threat levels (Sections 4.4, 5.4.2). Second, the flow is parameterized by deployment context. Identical telemetry produces different posture classifications and different routing in research, operational, and high-consequence-federal deployments because the per-tier thresholds and the per-layer response posture are calibrated per context. The architecture specifies the structure of the flow, not the calibration. Third, the flow is auditable end-to-end: every telemetry input is logged, every posture classification is recorded, every response action is recorded, and every threshold update produced by post-incident review (Section 5.4.2) is recorded. The audit chain enables both regulatory compliance and the empirical calibration that Section 8 identifies as the framework's falsifiability criterion.

6.3. Mapping to the Next-Generation Security Triad

Earlier work introduced the Next-Generation Security Triad as the integrated framework binding post-quantum cryptography, zero-trust architecture, and AI assurance into a single security posture [30]. The integrated architecture specified here is the AI assurance vertex of that triad, made concrete. ABOR (Section 5.2) instantiates the PQC vertex inside AI assurance: authority-chain attestation uses ML-DSA and SLH-DSA at the FIPS-specified parameter levels, and the long-term root's deployment lifetime is calibrated against the CRQC threshold from the PQC vertex. CPIP (Section 5.3) and VAOP (Section 5.1) instantiate the ZTA vertex: the deny-by-default network posture, the per-invocation runtime instantiation, and the vetted-access regime are zero-trust principles

 doi:10.20944/preprints202604.2179.v1

applied to AI deployment surfaces. MCPR (Section 4) provides the continuous-verification function that the ZTA vertex requires: the rubric continuously classifies system posture rather than treating evaluation as a deployment-time gate. The triad mapping is therefore not a retrospective alignment; the architecture was designed against it, and the four named contributions slot into the triad's vertices by construction.

The relationship between PQC and ZTA in this architecture is mutually reinforcing rather than merely co-located [30]. ZTA's disciplines—identity, access, segmentation, continuous verification—bound which surfaces an adversary can reach and the windows during which they operate unreclassified, but do not by themselves bind the cryptographic authority of specific outputs crossing those surfaces; an adversary who has compromised an authorized identity inherits whatever cryptographic standing the substrate provides. PQC-attested authority binding (ABOR) supplies that binding: each output crossing a ZTA-controlled boundary carries a provenance chain that survives identity compromise and the long-horizon threat profile motivating PQC adoption. Conversely, PQC provenance without ZTA's surface and identity controls leaves cryptographically authoritative outputs whose issuers are themselves adversary-reachable—the chain attests faithfully, but to the wrong operator. The two disciplines are therefore complementary controls rather than alternatives or stacked layers: ZTA bounds who and where, PQC/ABOR binds what and on whose authority, and the AI deployment surface is the integration point at which both bindings must hold simultaneously.

More directly, the four named contributions instantiate the principal disciplines of zero-trust architecture: VAOP supplies identity and governance (who may invoke capability under operator control), CPIP supplies least-privilege isolation (what each invocation can reach within the compute plane), ABOR supplies cryptographic enforcement (that each authorized output is bound to a verifiable authority chain at release), and MCPR supplies continuous verification (that posture is reclassified at every operational moment rather than gated only at deployment). Together they realize the "never trust, always verify" principle [13,14] at the AI deployment surface: every operation is verified against fresh evidence at the moment of effect, no operation inherits trust from a prior gate, and the verification substrate is itself cryptographically attested rather than configurationally asserted.

6.4. Mapping to NIST and CISA Frameworks

Three federal cybersecurity frameworks are most relevant to the architecture. We summarize each briefly before presenting the crosswalk in Table 7, because reviewers in this venue may not be uniformly familiar with all three at the level of detail the mapping requires.

The NIST AI Risk Management Framework (AI RMF 1.0, with the Generative AI Profile published 2024) structures AI risk management around four functions—Govern, Map, Measure, and Manage—with subcategories specifying outcomes for each. The framework is voluntary but increasingly referenced as a compliance baseline in federal AI procurement and in sector-specific AI governance regimes. The Govern function establishes organizational AI risk management; Map identifies the AI system's context; Measure assesses risks; Manage acts on them.

The NIST Cybersecurity Framework 2.0 (CSF 2.0, published 2024) structures general cybersecurity around six functions—Govern, Identify, Protect, Detect, Respond, and Recover—with categories and subcategories within each. CSF 2.0 added the Govern function relative to CSF 1.1 and is now the dominant cybersecurity-governance framework in U.S. federal practice and a substantial fraction of commercial practice.

The CISA Zero Trust Maturity Model (ZTMM, version 2.0 published 2023) structures zero-trust adoption across five pillars—Identity, Devices, Networks, Applications and Workloads, and Data—with maturity stages (Traditional, Initial, Advanced, Optimal) for each pillar. The model is the de facto reference for federal civilian agency zero-trust planning under OMB M-22-09.

 doi:10.20944/preprints202604.2179.v1

Table 7. Architecture-to-framework mapping. The crosswalk identifies framework subcategories the architecture supports or operationalizes; it is descriptive and does not by itself constitute a conformance assessment, which is determined by audit of an actual deployment.

Architecture component	NIST AI RMF (1.0 + GAI Profile)	NIST CSF 2.0	CISA ZTMM 2.0
VAOP — Governance (Section 5.1)	Govern 1.x (organizational AI risk management); Govern 4.x (responsible AI culture); Map 1.x (AI system context).	Govern (organizational cybersecurity strategy and risk); Identify.AM (asset management).	Identity pillar (Initial→Advanced); Govern function (cross-pillar).
ABOR — Cryptographic enforcement (Section 5.2)	Manage 1.x (risk treatment and mitigation); Measure 2.x (AI system performance and integrity).	Protect.DS (data security); Protect.PT (protective technology); Detect.CM (continuous monitoring of cryptographic state).	Data pillar (Advanced→Optimal: data encryption, attestation); Applications and Workloads pillar (Advanced).
CPIP — Architectural isolation (Section 5.3)	Manage 1.x (mitigation); Map 5.x (impacts on AI actors).	Protect.AC (identity management and access control); Protect.PT.PT-3 (least functionality); Protect.PT.PT-4 (communications networks).	Networks pillar (Advanced→Optimal: micro-segmentation, encrypted east-west); Applications and Workloads pillar.
Operational layer (Section 5.4)	Measure 1.x (assessment); Measure 4.x (feedback and improvement); Manage 4.x (continuous improvement).	Detect (full function); Respond (full function); Recover.IM (improvements).	Cross-pillar Visibility and Analytics; Cross-pillar Automation and Orchestration.
MCPR — Detection rubric (Section 4)	Measure 2.x and 3.x (system performance and risk measurement); Manage 1.x (treatment).	Identify.RA (risk assessment); Detect.AE (anomalies and events); Detect.CM (continuous monitoring).	Cross-pillar Visibility and Analytics (the rubric is the visibility instrument).

The crosswalk has two operational consequences. First, deployments implementing the architecture operationalize portions of all three frameworks within a unified architectural baseline rather than requiring separate implementation programs for each. The mapping does not by itself confer conformance with any of the three frameworks—conformance is determined by audit and inspection of an actual deployment—but it identifies which framework subcategories the architecture is designed to support and reduces duplicative implementation effort across the three governance regimes federal operators are typically required to address. Second, the crosswalk identifies the framework subcategories that the architecture does not address and that operators must address through complementary controls: NIST CSF's Recover function, the AI RMF's Govern 5.x (engagement with affected communities), and the ZTMM's Devices pillar are outside the architecture's scope. The mapping is descriptive: it documents where the architecture supports or operationalizes framework subcategories and where it does not, and is intended to support compliance documentation without claiming coverage the architecture does not provide.

Section 7 applies the integrated architecture to three application cases—post-quantum cryptography migration, federal AI supply-chain assurance, and operational technology defense—demonstrating that the four named contributions and the integration specified above generalize across heterogeneous deployment surfaces. Section 8 develops the limitations and the research agenda for empirical calibration.

 doi:10.20944/preprints202604.2179.v1

7. Application Cases

This section applies the integrated reference architecture of Section 6 to three deployment surfaces. The cases are short by design: their purpose is to demonstrate that the architecture's four named contributions generalize across heterogeneous operational contexts, not to develop any single deployment in full. Section 7.1 addresses post-quantum cryptography migration, the deployment surface most directly continuous with prior published work in this venue. Section 7.2 addresses federal AI supply-chain assurance, demonstrating the architecture's integration with model-bill-of-materials attestation. Section 7.3 addresses critical-infrastructure operational technology defense, demonstrating the architecture's applicability to deployment surfaces where legacy systems dominate the threat surface.

7.1. Post-Quantum Cryptography Migration

PQC migration is the deployment surface for which the Mythos-class threat is most fully developed in prior work [28]. The threat-and-trajectory analysis there shows that Mythos-class capability collapses the human-labor bottlenecks that force traditional PQC migration phases to be sequential, producing a compressed two-to-four-year migration window for highest-exposure systems against traditional baselines of five-to-seven years for small enterprises, eight-to-twelve years for medium enterprises, and twelve-to-fifteen-plus years for large enterprises [29]. The architecture specified in this paper applies to that deployment surface as follows.

VAOP (Section 5.1) bounds who may invoke frontier-model capability against the migration's cryptographic asset inventory, which protects the inventory itself from becoming an adversary's target register. ABOR (Section 5.2) binds release of any model-generated migration artifact—draft Cryptographic Bills of Materials, protocol-redesign proposals, hybrid-mode configurations—to a verifiable authority chain whose long-term root is itself signed with SLH-DSA at the parameter level developed in Section 5.2.4; the chain therefore survives the CRQC threshold the migration is itself crossing. CPIP (Section 5.3) constrains the model's reach into the migration's cryptographic-tooling infrastructure, preventing the analysis-acceleration use case from becoming an exfiltration channel for cryptographic-inventory data. MCPR (Section 4) classifies the deployment's posture continuously across the migration's lifecycle phases, with runtime-tier indicators specifically calibrated to detect the cross-domain reasoning patterns that produce the cross-domain attack-path archetype the preprint develops in detail. The integration with MBOM-PQC attestation specified in Section 5.2.8 is most directly load-bearing here: a migration whose cryptographic inventory is itself MBOM-PQC-attested produces ABOR runtime_context fields that reference the attestation by digest, closing the loop between supply-chain provenance and runtime release adjudication. The detailed lifecycle-phase mapping is developed in [28] and not reproduced here; the architectural integration is the contribution this paper adds.

7.2. Federal AI Supply-Chain Assurance

Federal AI supply-chain assurance is the deployment surface where the third operational regime in Section 3.1 (Mythos-class capability integrated into your AI/automation stack via supply-chain compromise) is most directly addressed. The case is structurally important because it is the regime under which the architecture's defenses are most tested: an adversary who has compromised a model's supply chain can produce a system that satisfies the evaluation-tier classification at deployment time and only manifests Mythos-class behavior in production, where deployment-tier and runtime-tier detection must catch what evaluation-tier detection cannot.

MBOM-PQC supply-chain attestation [31] establishes the cryptographic provenance of the model itself: training-data attestation, training-process attestation, weight-integrity attestation, and dependency-chain attestation, all bound through PQC primitives consistent with the architecture's long-term threat horizon. The integration with ABOR specified in Section 5.2.8 makes MBOM-PQC operationally consequential rather than archival: an authority record whose runtime_context

 doi:10.20944/preprints202604.2179.v1

references an MBOM-PQC attestation by digest binds the release decision to the supply-chain provenance, and the audit substrate records the per-release tuple (output_digest, target_system_id, enforcement_point_id, release_request_id) so that a post-hoc finding that a specific MBOM-PQC artifact was compromised propagates forward to identify every specific output that was released under that attestation, the target system each output reached, and the transaction in which the release occurred. CPIP's declared tool inventory with cryptographic attestation (Section 5.3.2) extends the same provenance regime to the scaffold: tools added after deployment change the inventory digest and either trigger re-attestation or fail authority-record verification, closing the supply-chain attack surface that operates through scaffold modification rather than through model modification. MCPR's deployment-tier authority-chain-signal indicators (Section 4.2) are the operational mechanism that monitors the supply-chain-attestation field for anomalies that would reveal compromise during operation. The integrated architecture therefore provides both supply-chain-compromise prevention (through attestation) and supply-chain-compromise detection (through the rubric's monitoring of the attestation's operational behavior). What the architecture does not address in this case is the regulatory framework under which MBOM-PQC attestations are issued and accepted across federal program boundaries; that is a policy concern outside this paper's scope and is developed in the MBOM-PQC preprint.

7.3. Critical-Infrastructure Operational Technology Defense

Operational technology (OT) environments—industrial control systems, supervisory control and data acquisition systems, building-management systems, transportation-control infrastructure—present a deployment surface where the legacy-system pressure loop developed in Section 3 of the Mythos-PQC preprint [28] most directly applies. OT systems disproportionately host classical cryptography, undocumented vendor-supplied binary blobs, and firmware that cannot be re-flashed without certification cycles measured in years. Mythos-class adversaries direct AI-accelerated vulnerability discovery at this population precisely because the population is rich in unpatched and architecturally-constrained targets.

The architecture applies asymmetrically across this deployment surface. CPIP (Section 5.3) carries the most weight: the deny-by-default network segmentation, the egress proxy with rate limiting and content inspection, and the per-tool scope binding are the controls that bound an adversary's ability to reach OT systems through any compromised AI-augmented analysis or maintenance tooling. ABOR (Section 5.2) bounds the operational effect of any model-generated artifact that reaches OT systems through legitimate channels (vendor-supplied analysis tools, federated maintenance systems, vulnerability-research outputs); the cryptographic enforcement layer is what prevents an adversary who has captured an analysis output from replaying it through the OT side's consuming-system enforcement points. MCPR runtime-tier indicators (Section 4.3) are calibrated for the cross-domain-reasoning signal that operations targeting OT systems characteristically produce: synthesis across protocol specifications, vendor documentation, and operational-procedure artifacts within a single chain of action. VAOP (Section 5.1) is the layer that fits this surface least naturally, because OT environments are commonly operated by participants whose vetting regime is calibrated to OT-engineering competence rather than to AI-deployment governance; the bridging-layer argument in Section 5.1.4 is operationally consequential here because it justifies investment in ABOR and CPIP at OT boundaries rather than waiting for VAOP to mature in OT contexts. What the architecture does not address in this case is the certification cycle that gates firmware change in OT environments; the architecture assumes that consuming-system enforcement points can be deployed at OT boundaries, but the deployment itself may require regulatory and vendor coordination that exceeds this paper's scope.

The three cases demonstrate that the architecture's four named contributions apply across deployment surfaces with different threat profiles, different legacy constraints, and different governance regimes. The application is not uniform: VAOP carries most weight in PQC migration where the participant population is well-defined; ABOR's MBOM-PQC integration carries most

 doi:10.20944/preprints202604.2179.v1

weight in supply-chain assurance; CPIP's isolation carries most weight in OT defense; MCPR operates across all three with per-context calibration of its thresholds. Section 8 develops the limitations and the research agenda for empirical calibration of the architecture across these and other deployment surfaces.

8. Discussion: Limitations and Research Agenda

This paper develops a reference architecture and demonstrates its application across three deployment surfaces; it does not present empirical validation of a deployed instance. We close with the limitations this scope implies, the falsifiability criteria against which the architecture's analytical claims can be tested by subsequent work, and the research agenda that follows. The discussion is structured across three epistemic registers consistent with the discipline applied in [28]: documented properties of the architecture as specified, analytical claims about its operational behavior, and normative recommendations for adoption. Readers are asked to hold claims in each register to the standard of evidence that applies there.

8.1. Limitations

Five limitations bound the contribution. First, the architecture has not been deployed at production scale. ABOR's authority-plane infrastructure, CPIP's configuration discipline, and MCPR's parameterized scoring are specified to the level required for conformance evaluation, but the integrated architecture has not been operationalized end-to-end at the time of this writing. Performance characteristics, operational overhead, and failure-mode behavior in production are unobserved; the per-release verification cost estimates in Section 5.2.9 are derived from PQC-primitive benchmark performance rather than from deployed enforcement-point measurement. Second, the operator-side infrastructure required for full ABOR conformance—a long-term-root HSM, deployment-root issuance regime, runtime-issuer infrastructure, and enforcement points instrumented at every consuming-system boundary—exceeds what most organizations currently possess. The architecture is therefore deployable incrementally, beginning with high-consequence target systems, but full conformance is a multi-year program rather than a configuration change.

Third, MCPR's per-tier thresholds are specified as parameterized rather than absolute. The framework specifies the structure of the indicators and the routing they produce; the threshold values that distinguish Amber from Red within a given deployment context are operator-deployed and are calibrated from operational telemetry rather than from this paper's analysis. A consequence is that two deployments of the architecture against comparable threats may produce different posture classifications if their threshold calibrations differ; the architecture's specification is identical, the operational behavior is not. Fourth, the architecture's defense against catastrophic failure modes—long-term root compromise (Section 5.2.7), authority-plane infiltration (Section 5.3.5), MCPR threshold subversion (Section 4.4)—depends on operational disciplines (multi-person key authorization, asymmetric network reachability, audit-substrate integrity) whose effectiveness rests on the operator organization's integrity rather than on cryptographic hardness. Fifth, the architecture's applicability across deployment surfaces is asymmetric, as developed in Section 7: VAOP fits less naturally in OT contexts than in commercial-frontier-vendor contexts, and the bridging-layer argument that justifies investing in the durable layers ahead of VAOP maturity in those contexts is itself an analytical claim rather than an operational observation.

8.2. Falsifiability Criteria

The architecture's analytical claims are falsifiable through operational measurements that subsequent work can produce. Four claim classes warrant explicit falsifiability statements. First, the per-release verification cost claim in Section 5.2.9 (a two-record chain adding approximately one-to-two milliseconds on commodity hardware, under the FIPS reference-implementation assumptions stated there) is directly benchmarkable against deployed enforcement-point measurements;

 doi:10.20944/preprints202604.2179.v1

deployments that observe materially higher costs falsify the claim and motivate either alternative attestation profiles or amortization optimizations beyond those the paper specifies. Second, the MCPR posture classifications produce auditable predictions: a deployment whose Halt posture fires routinely against operations that subsequent investigation reveals to be benign falsifies the false-positive rate the asymmetric-response argument in Section 4.4 assumes; conversely, a deployment whose runtime-tier indicators fail to elevate against operations that subsequent investigation reveals to have produced consequential effect falsifies the indicator coverage.

Third, the bridging-layer argument in Section 5.1.4 makes a quantitative prediction: VAOP's operational lifespan is bounded by the open-weight diffusion horizon (months-scale, with central estimates of approximately three months [8]). Deployments that observe VAOP-protected effective security beyond two years against open-weight adversaries falsify the bridging-layer framing in either direction—either the diffusion horizon is longer than the cited analyses suggest or VAOP's defenses are more durable than the architecture claims. Fourth, the integration claim with MBOM-PQC supply-chain attestation is operationally testable: deployments that implement both ABOR and MBOM-PQC produce audit-substrate records that should permit forward propagation of supply-chain compromise findings to dependent release decisions; deployments that implement both but cannot produce the propagation falsify the integration as specified. The four claim classes together provide a research substrate against which deployed pilots can produce empirical refinement.

8.3. Research Agenda

Three research directions follow. The first is longitudinal pilot work: deployments of the integrated architecture, in operational environments, with telemetry published before and after deployment to enable empirical calibration of the projections this paper develops. Federal program offices and critical-infrastructure operators are the natural pilot sites because their existing compliance regimes (ATO, FedRAMP, sector-specific frameworks) supply much of the audit substrate the architecture requires. The second is calibration of MCPR thresholds against per-deployment-context operational data: the parameterized threshold specification is a research scaffold, and calibration data published from pilots—per-tier indicator distributions, posture-classification frequencies, response-action outcomes—would convert the parameterization from operator-deployed to evidence-based. The third is integration with adjacent governance regimes: NIST AI RMF profiles for sector-specific deployments, CISA ZTMM advanced-stage implementation guidance, and the international equivalents (EU AI Act high-risk system requirements, UK AI Cyber Security Code of Practice) that operate parallel to the U.S. federal frameworks the crosswalk in Section 6.4 addresses. The architecture's mapping to these regimes is incomplete in this paper and represents a substantial line of follow-on work.

The four named contributions of this paper—VAOP, ABOR, CPIP, and MCPR—are intended as a starting point for that work. The naming discipline is deliberate: subsequent literature can build on, refine, or supersede each component independently rather than negotiating with a monolithic framework. Refinements to ABOR's authority-record format, to CPIP's tool-inventory attestation, to MCPR's indicator structure, or to VAOP's vetting regime are anticipated, and the architecture is designed to absorb such refinements without re-deriving the integration in Section 6. The reference architecture as specified is the contribution; the empirical calibration and operational refinement that follow are the research program the contribution opens.

9. Conclusion

The April 2026 emergence of Mythos-class frontier model capability established a new operational threat category whose capability profile—extended-context reasoning over codebases, recursive self-correction, native system-tool integration, and agentic scaffolding at deployable scale—rendered the dominant AI safety paradigms insufficient as sole controls. The five-indicator operational definition in Section 3.2 conjoins this capability profile with the scaffold, access-pattern, autonomy-depth, and persistence indicators that together delimit the class as deployed. RLHF,

 doi:10.20944/preprints202604.2179.v1

output filtering, contractual access vetting, and human-in-the-loop supervision were each calibrated to a generation of systems that did not exhibit autonomous cyber capability at the levels Mythos-class systems now demonstrate, and each is insufficient as a sole control against the new category under the threat assumptions specified here. This paper specified a defense-in-depth reference architecture against that category.

Three findings anchor the contribution. First, Mythos-class is a compound class: capability conjoined with scaffold, access pattern, autonomy depth, and persistence (Section 3.2). Detection therefore classifies system posture against the compound definition rather than against capability alone, and the Mythos-Class Posture Rubric in Section 4 instantiates that classification across pre-deployment, deployment, and runtime tiers with explicit routing to mitigation layers. Second, mitigation requires four layers operating jointly: governance (Section 5.1, the Vetted-Access Operational Pattern), cryptographic enforcement (Section 5.2, Authority-Bound Output Release), architectural isolation (Section 5.3, the Compute-Plane Isolation Profile), and operational monitoring (Section 5.4). The four layers bound different consequence classes, and the integration in Section 6 produces defense-in-depth that no single layer alone provides. Third, the cryptographic enforcement layer is the architecturally distinctive contribution among the four layers, whose lifespan is bounded by post-quantum primitives rather than by access restriction, and is therefore the layer that survives the capability-diffusion horizon that bounds the governance layer's effective lifespan (Section 5.1.4).

The operational implication for federal program managers, enterprise security organizations, and critical-infrastructure operators is that the threat condition Mythos-class capability creates is not addressed by tightening any single existing control surface. The architecture specified here is a reference design rather than a deployed system, and Section 8 develops the limitations and falsifiability criteria against which subsequent empirical work can refine the design. The four named contributions—VAOP, ABOR, CPIP, and MCPR—are intended to be refined or superseded independently as deployment experience accumulates. Operators who wait for empirical validation of the architecture before beginning deployment will find that the capability-diffusion horizon has eroded the governance layer's effectiveness in the interim; operators who treat the architecture as the bridging structure within which empirical refinement happens will be positioned to act on subsequent calibration data as it becomes available. The reference architecture is the contribution; the program of operational deployment, empirical calibration, and component refinement that follows is the research the contribution opens.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new datasets were generated; the analysis relies on publicly available documents.

Conflicts of Interest: The author declares no conflict of interest.

 doi:10.20944/preprints202604.2179.v1

Abbreviations

The following abbreviations are used in this manuscript:

ABOR	Authority-Bound Output Release
AI	Artificial Intelligence
AISI	AI Security Institute (UK)
AISLE	AI Security Lab Europe
AIxCC	AI Cyber Challenge (DARPA)
ATO	Authority To Operate
CBOM	Cryptographic Bill of Materials
CBOR	Concise Binary Object Representation (RFC 8949)
CETaS	Centre for Emerging Technology and Security (Alan Turing Institute)
CISA	Cybersecurity and Infrastructure Security Agency
CMMC	Cybersecurity Maturity Model Certification
CNSA	Commercial National Security Algorithm Suite
COSE	CBOR Object Signing and Encryption (RFC 9052)
CPIP	Compute-Plane Isolation Profile
CRQC	Cryptanalytically-Relevant Quantum Computer
CSF	Cybersecurity Framework (NIST)
DARPA	Defense Advanced Research Projects Agency
DSIT	Department for Science, Innovation and Technology (UK)
FedRAMP	Federal Risk and Authorization Management Program
FIPS	Federal Information Processing Standards
GAI	Generative AI
HSM	Hardware Security Module
ICS	Industrial Control Systems
IETF	Internet Engineering Task Force
JOSE	JavaScript Object Signing and Encryption
LLM	Large Language Model
MBOM-PQC	Model Bill of Materials with Post-Quantum Cryptography attestation
MCPR	Mythos-Class Posture Rubric
ML-DSA	Module-Lattice-based Digital Signature Algorithm (FIPS 204)
ML-KEM	Module-Lattice-based Key-Encapsulation Mechanism (FIPS 203)
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OMB	Office of Management and Budget
OT	Operational Technology
PKI	Public Key Infrastructure
PQC	Post-Quantum Cryptography
RFC	Request for Comments (IETF document)
RLHF	Reinforcement Learning from Human Feedback
RMF	Risk Management Framework (NIST AI)
SCADA	Supervisory Control and Data Acquisition
SHA	Secure Hash Algorithm
SLH-DSA	Stateless Hash-based Digital Signature Algorithm (FIPS 205)
SOC	Security Operations Center
UUID	Universally Unique Identifier
VAOP	Vetted-Access Operational Pattern
ZTA	Zero Trust Architecture
ZTMM	Zero Trust Maturity Model (CISA)

References

 doi:10.20944/preprints202604.2179.v1

1. Anthropic. System Card: Claude Mythos Preview, 7 April 2026. Available online: https://www.anthropic.com/claude-mythos-preview-system-card (accessed 28 April 2026).
2. Carlini, N.; Cheng, N.; Lucas, K.; Moore, M.; Nasr, M.; Prabhushankar, V.; Xiao, W.; et al. (Anthropic Frontier Red Team). Assessing Claude Mythos Preview's Cybersecurity Capabilities, 7 April 2026. Available online: https://red.anthropic.com/2026/mythos-preview (accessed 28 April 2026).
3. AI Security Institute (AISI). Our Evaluation of Claude Mythos Preview's Cyber Capabilities; UK Department for Science, Innovation and Technology: London, UK, 13 April 2026. Available online: https://www.aisi.gov.uk/blog/our-evaluation-of-claude-mythos-previews-cyber-capabilities (accessed 28 April 2026).
4. Bain & Company. AI's Cybersecurity Wake-Up Call: What Anthropic's Mythos Means for Enterprise Risk; Bain & Company: Boston, MA, USA, April 2026. Available online: https://www.bain.com/insights/ai-cybersecurity-wake-up-call-mythos/ (accessed 28 April 2026).
5. World Economic Forum. Anthropic's Mythos Moment: How Frontier AI Is Redefining Cybersecurity, April 2026. Available online: https://www.weforum.org/stories/2026/04/anthropic-mythos-ai-cybersecurity/ (accessed 28 April 2026).
6. CrowdStrike. CrowdStrike 2026 Global Threat Report; CrowdStrike: Austin, TX, USA, 24 February 2026. Available online: https://go.crowdstrike.com/rs/281-OBQ-266/images/CrowdStrike-2026-Global-Threat-Report.pdf (accessed 28 April 2026).
7. European Commission Apply AI Alliance. Toward a Machine-Level Safety Fuse: Position Paper on Frontier Model Cybersecurity Capability; Futurium platform, European Commission: Brussels, Belgium, April 2026. Available online: https://futurium.ec.europa.eu/en/apply-ai-alliance/ (accessed 28 April 2026).
8. Centre for Emerging Technology and Security (CETaS), Alan Turing Institute. Claude Mythos: What Does Anthropic's New Model Mean for the Future of Cybersecurity? CETaS: London, UK, April 2026. Available online: https://cetas.turing.ac.uk/publications/claude-mythos-future-cybersecurity (accessed 28 April 2026).
9. Fort, S. AI Cybersecurity After Mythos: The Jagged Frontier; AISLE (AI Security Lab Europe): April 2026. Available online: https://aisle.com/blog/ai-cybersecurity-after-mythos-the-jagged-frontier (accessed 28 April 2026).
10. Anthropic. Project Glasswing, 7 April 2026. Available online: https://www.anthropic.com/project/glasswing (accessed 28 April 2026).
11. National Institute of Standards and Technology. AI Risk Management Framework (AI RMF 1.0) and Generative AI Profile (NIST AI 600-1); NIST: Gaithersburg, MD, USA, January 2023 (RMF 1.0); July 2024 (GAI Profile). https://doi.org/10.6028/NIST.AI.600-1
12. National Institute of Standards and Technology. The NIST Cybersecurity Framework (CSF) 2.0; NIST CSWP 29; NIST: Gaithersburg, MD, USA, February 2024. https://doi.org/10.6028/NIST.CSWP.29
13. Cybersecurity and Infrastructure Security Agency. Zero Trust Maturity Model, Version 2.0; CISA: Arlington, VA, USA, April 2023. Available online: https://www.cisa.gov/zero-trust-maturity-model (accessed 28 April 2026).
14. Office of Management and Budget. OMB M-22-09: Moving the U.S. Government Toward Zero Trust Cybersecurity Principles; Executive Office of the President: Washington, DC, USA, January 2022.
15. Office of Management and Budget. OMB M-23-02: Migrating to Post-Quantum Cryptography; Executive Office of the President: Washington, DC, USA, November 2022.
16. European Parliament and Council. Regulation (EU) 2024/1689 (Artificial Intelligence Act); Official Journal of the European Union L 1689, 12 July 2024. Available online: https://eur-lex.europa.eu/eli/reg/2024/1689/oj (accessed 28 April 2026).
17. UK Department for Science, Innovation and Technology. AI Cyber Security Code of Practice; DSIT: London, UK, January 2025. Available online: https://www.gov.uk/government/publications/ai-cyber-security-code-of-practice (accessed 28 April 2026).
18. Anthropic. Alignment Risk Update: Claude Mythos Preview, 7 April 2026. Available online: https://www.anthropic.com/claude-mythos-preview-risk-report (accessed 28 April 2026).

 doi:10.20944/preprints202604.2179.v1

19. Folkerts, L.; Payne, W.; Inman, S.; Giavridis, P.; Skinner, J.; Deverett, S.; Aung, J.; Zorer, E.; Schmatz, M.; Ghanem, M.; Wilkinson, J.; Steer, A.; Hong, V.; Wang, J. (UK AI Security Institute). Measuring AI Agents' Progress on Multi-Step Cyber Attack Scenarios. arXiv 2026, arXiv:2603.11214. https://doi.org/10.48550/arXiv.2603.11214
20. Dawson, A.; Mulla, R.; Landers, N.; Caldwell, S. AIRTBench: Measuring Autonomous AI Red Teaming Capabilities in Language Models. arXiv 2025, arXiv:2506.14682. https://doi.org/10.48550/arXiv.2506.14682
21. Liu, Z.; et al. PACEbench: A Framework for Evaluating Practical AI Cyber-Exploitation Capabilities. arXiv 2025, arXiv:2510.11688. https://doi.org/10.48550/arXiv.2510.11688
22. Zhang, A.K.; Perry, N.; Dulepet, R.; et al. Cybench: A Framework for Evaluating Cybersecurity Capabilities and Risks of Language Models. arXiv 2024, arXiv:2408.08926. https://doi.org/10.48550/arXiv.2408.08926
23. Microsoft Research. CTI-REALM: Benchmark to Evaluate Agent Performance on Security Detection Rule Generation Capabilities. arXiv 2026, arXiv:2603.13517. Available online: https://www.microsoft.com/en-us/security/blog/2026/03/20/cti-realm-a-new-benchmark-for-end-to-end-detection-rule-generation-with-ai-agents/ (accessed 28 April 2026).
24. Glazunov, S.; Brand, M.; Project Zero; DeepMind. From Naptime to Big Sleep: Using Large Language Models to Catch Vulnerabilities in Real-World Code. Google Project Zero, 1 November 2024. Available online: https://googleprojectzero.blogspot.com/2024/10/from-naptime-to-big-sleep.html (accessed 28 April 2026).
25. Bhatt, M.; Chennabasappa, S.; Nikolaidis, C.; Wan, S.; Evtimov, I.; Gabi, D.; Song, D.; Ahmad, F.; Aschermann, C.; Fontana, L.; et al. Purple Llama CyberSecEval: A Secure Coding Benchmark for Language Models. arXiv 2023, arXiv:2312.04724. https://doi.org/10.48550/arXiv.2312.04724
26. Defense Advanced Research Projects Agency (DARPA). AI Cyber Challenge (AIxCC) Final Competition Results; DARPA: Arlington, VA, USA, 8 August 2025. Available online: https://www.darpa.mil/news/2025/aixcc-results (accessed 28 April 2026).
27. Mandiant. M-Trends 2024: Special Report; Mandiant (Google Cloud): Reston, VA, USA, 2024. Available online: https://services.google.com/fh/files/misc/m-trends-2024.pdf (accessed 28 April 2026).
28. Campbell, R. Frontier AI Compresses PQC Migration: Mythos-Class Capability and Compressed Migration Trajectories. Preprints.org 2026. https://doi.org/10.20944/preprints202604.1744.v1
29. Campbell, R. Enterprise Migration to Post-Quantum Cryptography: Timeline Analysis and Strategic Frameworks. Computers 2026, 15, 9. https://doi.org/10.3390/computers15010009
30. Campbell, R. Synchronizing Concurrent Security Modernization Programs: Zero Trust, Post-Quantum Cryptography, and AI Assurance. Systems 2026, 14, 233. https://doi.org/10.3390/systems14030233
31. Campbell, R. AI Supply Chain Security: MBOM-PQC Provenance, PQC Attestation, and a Maturity Model for Quantum-Resistant Assurance. Preprints.org 2026. https://doi.org/10.20944/preprints202603.1963.v1
32. National Institute of Standards and Technology. FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard (ML-KEM); NIST: Gaithersburg, MD, USA, August 2024. https://doi.org/10.6028/NIST.FIPS.203
33. National Institute of Standards and Technology. FIPS 204: Module-Lattice-Based Digital Signature Standard (ML-DSA); NIST: Gaithersburg, MD, USA, August 2024. https://doi.org/10.6028/NIST.FIPS.204
34. National Institute of Standards and Technology. FIPS 205: Stateless Hash-Based Digital Signature Standard (SLH-DSA); NIST: Gaithersburg, MD, USA, August 2024. https://doi.org/10.6028/NIST.FIPS.205
35. Bormann, C.; Hoffman, P. Concise Binary Object Representation (CBOR); RFC 8949, Internet Engineering Task Force (IETF), December 2020. https://doi.org/10.17487/RFC8949
36. Schaad, J. CBOR Object Signing and Encryption (COSE): Structures and Process; RFC 9052, Internet Engineering Task Force (IETF), August 2022. https://doi.org/10.17487/RFC9052
37. Davis, K.; Peabody, B.; Leach, P. Universally Unique IDentifiers (UUIDs); RFC 9562, Internet Engineering Task Force (IETF), May 2024. https://doi.org/10.17487/RFC9562. Obsoletes RFC 4122.
38. National Institute of Standards and Technology. FIPS 140-3: Security Requirements for Cryptographic Modules; NIST: Gaithersburg, MD, USA, March 2019. https://doi.org/10.6028/NIST.FIPS.140-3
39. Anthropic. Responsible Scaling Policy, Version 2.2; Anthropic: San Francisco, CA, USA, March 2026. Available online: https://www.anthropic.com/rsp (accessed 28 April 2026).

 doi:10.20944/preprints202604.2179.v1

40. National Security Agency. Announcing the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0); NSA Cybersecurity Advisory; NSA: Fort Meade, MD, USA, September 2022.
41. Prorock, M.; Steele, O. ML-DSA for JOSE and COSE; Internet-Draft draft-ietf-cose-dilithium-11, IETF COSE Working Group, 15 November 2025. Available online: https://datatracker.ietf.org/doc/draft-ietf-cose-dilithium/ (accessed 28 April 2026).
42. Prorock, M.; Steele, O.; Misoczki, R.; Osborne, M.; Cloostermans, C. SLH-DSA for JOSE and COSE; Internet-Draft draft-ietf-cose-sphincs-plus-07, IETF COSE Working Group, March 2026. Available online: https://datatracker.ietf.org/doc/draft-ietf-cose-sphincs-plus/ (accessed 28 April 2026).