# A Multidimensional Protection Strategy for Engineering Mission-Resilient Systems in the Age of Frontier AI[1]

Dr. Ron Ross
CEO, RONROSSECURE

**Abstract**

*Frontier artificial intelligence is reshaping cybersecurity and systems engineering in two distinct ways: by increasing the speed and scale of adversary cyber operations and by introducing AI components whose behavior may be difficult to fully specify, predict, and assure. This paper argues that NIST SP 800-160, Vols. 1 and 2, provides a strong systems security engineering and cyber-resiliency foundation for addressing these challenges, but that the foundation must be extended with AI-specific protections. The paper proposes a five-layer Multidimensional Protection Strategy consisting of: (1) a systems security engineering foundation based on NIST SP 800-160; (2) AI-specific threat modeling using resources such as MITRE ATLAS and the NIST AI Risk Management Framework; (3) adaptive operational tempo capable of responding to AI-accelerated threats; (4) data and model integrity protections; and (5) governance and organizational agility. The central argument is that sufficiency against frontier AI must be structural rather than dependent on assumptions about model behavior. Frontier AI components should be treated as assurance-bounded components whose authority, access, and potential impact are constrained by the architecture surrounding them. Likewise, systems should be engineered so that an AI-enabled adversary's ability to gain an initial foothold does not automatically translate into mission compromise. Mission resilience in the age of frontier AI therefore requires not merely additional security technologies, but an integrated systems engineering discipline that combines trustworthy system architecture, AI-specific threat intelligence, rapid adaptation, integrity assurance, and continuous governance.*

## 1. Introduction

A new term has entered the vocabulary of risk officers, systems engineers, and national security planners alike: *frontier AI*. The term refers to the small set of Artificial Intelligence (AI) models operating at or near the current outer limit of AI capability. This includes large language models and AI agents capable of autonomous, multi-step reasoning, tool use, and code generation, whose capabilities are advancing quickly enough that a defensive posture calibrated to last year's models can be obsolete against this year's release. Frontier AI is not a single product or vendor. It is a capability tier, and it is dual-use by nature. The same underlying reasoning and code-generation ability that helps a defender triage a vulnerability backlog also allows an adversary to scan software, identify exploitable flaws, and generate functional exploit code autonomously, at machine speed, and at a fraction of the cost a human-driven attack used to require.

That dual-use character creates two distinct engineering problems, and any strategic framework for mission resilience has to address both. The first problem is *frontier AI as a threat-actor force multiplier*. An adversary equipped with frontier-class tools can compress reconnaissance, exploit development, and multi-stage attack chaining into a fraction of the time a human-only campaign required. The second, less discussed problem is *frontier AI as a system component*. Organizations are increasingly embedding

[1] This is a living document. Updated versions of the original paper will be published in response to comments received. Revision numbers will be listed in the upper right-hand header of the paper.

frontier AI models directly into their mission and business systems as decision-making, generative, or orchestration components, and those components may behave in ways that resist the kind of formal, specifiable trust arguments that systems security engineering has relied on for decades.

This article argues that NIST Special Publication 800-160, Vol. 1, and specifically the thirty security design principles catalogued in Appendix E, remain the correct engineering foundation for building trustworthy, mission-resilient systems in this environment. But it also argues that the foundation is not, by itself, sufficient. The security design principles were written to be technology-agnostic, and that is its strength; it is also the reason it is silent on the attack techniques and trust problems that are unique to AI. What follows is a multidimensional protection strategy that keeps the systems security engineering concepts and principles in SP 800-160 as the architectural backbone and foundation while stacking four additional layers of AI-specific defenses on top of it. These include: (1) *threat modeling*; (2) *adaptive operational tempo*; (3) *data and model integrity*; and (4) *governance*. Building trustworthy systems that are mission resilient in a threat environment with frontier-AI-enabled adversaries requires all five layers to be effective.

> ***"Sufficiency against frontier AI is structural, not behavioral: an architecture must hold regardless of what a model intends, chooses, or is instructed to do."***

## 2. The Foundational Discipline: NIST SP 800-160, Volumes 1 and 2

### Security Design Principles

NIST SP 800-160, Vol. 1 codifies thirty security design principles including, *Anomaly Detection*, *Clear Abstractions*, *Commensurate Protection*, *Commensurate Response*, *Commensurate Rigor*, *Defense in Depth*, *Commensurate Trustworthiness*, *Compositional Trustworthiness*, *Minimal Trusted Elements*, *Least Privilege*, and *Loss Margins*. These principles are deliberately general-purpose. They describe how to think about designing a trustworthy system — how to bound trust in a system component, how to reduce and reason about attack surface, how to ensure that guarantees still hold once components are integrated into a larger whole — rather than prescribing a fixed set of controls tied to any particular adversary or technology.

That technology-agnostic quality is precisely why the systems engineering approach has held up for decades of shifting threats, and it is genuinely useful against AI-accelerated adversaries. If the threat model is "an attacker, human or automated, trying to find and exploit a vulnerability faster than the defender can respond," then least privilege, domain separation, reduced complexity, and clear trust boundaries raise the cost of a successful attack regardless of whether the reconnaissance and exploit generation on the other side of the boundary were done by a person or a frontier model. The principles do not care who or what is attacking; they care whether the architecture makes the attack structurally more difficult. That is a durable property, and it is why SP 800-160 belongs at the base of any strategy discussed in this article.

The systems security engineering framework described in SP 800-160, Vol. 1, structures how these principles get applied, through three distinct contexts. The *problem context* establishes protection objectives and requirements — what a system must achieve and what losses must be prevented. The *solution context* defines the protection strategy and system architecture — how the security design principles are actually applied. The *trustworthiness context* demonstrates, through evidence, that the resulting architecture achieves its objectives, not by assertion but through analysis, inspection, and

 

testing that can be independently verified. The distinction matters for everything that follows: an organization that has addressed the problem and solution contexts but has nothing to show for the trustworthiness context has built a system it believes is secure, not one it has demonstrated to be secure — and that gap is exactly where confidence in a system tends to fail first against a frontier-AI-enabled adversary.

> ***"An organization that has addressed the problem and solution contexts but has nothing to show for the trustworthiness context has built a system it believes is secure, not one it has demonstrated to be secure."***

## Cyber and System Resiliency

NIST SP 800-160, Vol. 2 describes a second, complementary foundation: *cyber and system resiliency*. It is organized around four explicit goals: (1) *anticipate*; (2) *withstand*; (3) *recover*; and (4) *adapt*. Anticipate means establishing and maintaining a state of informed preparedness for adversity. Withstand means continuing essential mission or business functions despite adversity, without requiring that the adversity first be detected. Recover means restoring mission or business functions during and after adversity, with the caveat that restored functions and data cannot automatically be assumed trustworthy simply because they have been restored. Adapt means modifying mission or business functions and supporting capabilities in response to predicted changes in the technical, operational, or threat environments. SP 800-160 names artificial intelligence directly as an example of the kind of technical-environment change the "adapt" goal exists to address. The publication identified, at the level of a goal, that AI would be a source of disruptive change well before frontier-scale AI models made that disruption concrete.

SP 800-160, Vol. 2 also codifies five *strategic design principles* that inform which structural principles and techniques apply. Two are directly relevant to the argument in this article. *Assume Compromised Resources* holds that systems and their components — from chips to software modules to running services — can be compromised for extended periods without detection, and that architectures should be designed on the assumption that some resources cannot be trusted to behave as intended, rather than on the assumption that most resources are benign. *Expect Adversaries to Evolve* holds that a system hardened only against known vulnerabilities will remain exposed to adversaries that continue to develop new tactics, techniques, and procedures, and that resilience must be designed for adversaries who have not yet been observed. Both principles were written for human adversaries and compromised infrastructure; both apply with equal force, and arguably greater urgency, to a frontier AI model that behaves unpredictably, not because it has been compromised by an outside attacker, but because its own goal-directed behavior was never fully specified in the first place. This point is developed further below, in the discussion of frontier AI as an untrustworthy system component.

SP 800-160, Vol. 2 further defines fourteen cyber resiliency techniques — practices and technologies intended to achieve the four resiliency goals. Several bear directly on frontier AI and recur throughout the strategy in Section 5: *Segmentation*, which defines and separates system components based on criticality and trustworthiness; *Non-Persistence*, which generates and retains resources only as needed or for a limited time, shrinking the temporal window an adversary, human or autonomous, has to find and exploit a flaw; *Substantiated Integrity*, which ascertains whether critical system components have been corrupted, on an ongoing basis rather than once at deployment; *Adaptive Response*, which implements agile courses of action and responses to manage risk as conditions change; *Analytic Monitoring*, which monitors and analyzes emergent system properties and behaviors on an ongoing, coordinated basis; and *Unpredictability*, one of only two techniques Volume 2 identifies as specific to

adversarial threats, which makes changes randomly or unpredictably to deny an adversary the ability to plan against a known, fixed configuration.

## 3. Two Distinct Problems Frontier AI Creates

### Frontier AI as a Threat-Actor Capability

The first problem is external and adversarial. A frontier model applied offensively can increasingly automate vulnerability discovery, exploit development, and portions of attack-chain execution with reduced human direction. It can chain that capability across an entire attack campaign: reconnaissance that maps an attack surface at scale, planning that sequences an attack chain toward a high-value target, and adaptive malware that changes its behavior when it meets a defensive measure. None of the underlying attack techniques are new — initial access, lateral movement, privilege escalation, and persistence remain what they have always been. What has changed is the economics: the time, cost, and expertise required to execute a sophisticated campaign have collapsed, and a defender's traditional assumption, that vulnerabilities can be found and patched faster than an adversary can find and weaponize them, no longer holds against an adversary operating at machine speed.

The reason this shift matters as much as it does is economic, not merely technical. A useful way to state a defender's objective is cost imposition: raise the cost and complexity of a successful attack until it exceeds what the adversary is willing or able to pay. Perimeter-based security imposes cost only at the initial-access phase of an attack; once that perimeter is breached, the adversary's remaining costs collapse, because a flat, undifferentiated architecture offers no further resistance. The use of Frontier AI compounds this problem rather than creating a new one. It collapses the cost of initial access; the one phase perimeter security was ever any good at pricing. Structural design principles change where cost is imposed rather than how much of it there is: a properly domain-separated architecture imposes cost at every subsequent phase of an attack chain — discovery, lateral movement, privilege escalation, and persistence — so that a foothold gained cheaply and quickly still has to be converted into mission impact the hard way, one bounded domain at a time.

> ***"AI lowers the adversary's cost of finding and exploiting vulnerabilities; therefore, resilient architecture must increase the adversary's cost after initial compromise. The AI-era objective is not merely to prevent initial compromise. It is to prevent initial compromise from becoming mission compromise."***

### Frontier AI as an Untrustworthy System Component

The second problem is internal and architectural, and it is easy to overlook. When an organization embeds a frontier AI model into its own system as a decision support tool, a code-generation assistant, or an autonomous agent, that model becomes a system component in the SP 800-160 sense, and the systems engineering approach expects components to be bounded, specified, and verifiable. Frontier AI models and systems present a fundamentally different assurance challenge from conventional software components (i.e., behavior is probabilistic, context-dependent, or insufficiently specified to support strong behavioral guarantees). Behavior is generated through learned statistical representations and may vary with inputs, context, model configuration, and execution environment. As a result, traditional specification, testing, and formal-verification techniques may be insufficient to establish comprehensive behavioral guarantees. The engineering challenge is therefore not to prove that the model will always behave as intended, but to architect the surrounding system so that unacceptable model behavior is bounded, detected, contained, and recoverable.

This is not a new demand on the systems engineering approach so much as a direct extension of one it already makes. The strategic design principle *Assume Compromised Resources* in SP 800-160, Vol. 2, instructs architects to treat components as potentially unable to be trusted to behave as intended, without requiring proof of an external compromise first. A frontier model embedded as a component satisfies that condition natively: it need not be compromised by an outside attacker to behave in ways its own designers did not specify, because its behavior was never fully specified to begin with. Thus, a frontier AI model should not necessarily be treated as a trusted component in the traditional sense.[2] Instead, it should be treated as an *assurance-bounded component* whose authority, access, autonomy, and potential impact are constrained by the architecture surrounding it. This gives rise to a proposed, new security design principle called *Assurance-bounded AI Component*. The principle states that **an AI component is never required to be more trustworthy than the system architecture can independently enforce**. It complements the current design principles in SP 800-160, Vol. 1 (e.g., *Least Privilege*, *Mediated Access*, *Domain Separation*, *Defense in Depth*, *Minimal Trusted Elements*, *Segmentation, Substantiated Integrity*, and *Compositional Trustworthiness*) and extends systems security engineering theory for incorporating inherently uncertain intelligent components into mission systems.

> ***"An organization that treats a frontier model as a classically "trusted component" because it passed an evaluation once is making a claim the component's own behavior does not support."***

## 4. Where the Foundation Falls Short

There are four gaps that follow directly from the two problems described in Section 3. It is important to understand these gaps before accepting the sufficiency of traditional risk management frameworks, cybersecurity frameworks, and systems engineering approaches.

- **AI-specific attacks are not explicitly addressed as a distinct threat domain.**

SP 800-160, Vol. 1 provides architectural mechanisms that can mitigate AI-specific threats (e.g., *prompt injection* (malicious instructions embedded in a model's input to hijack its behavior), *model extraction* (recovering a model's parameters or training data through repeated querying), *adversarial examples* (inputs deliberately crafted to cause misclassification), *data poisoning* (corrupting training data or fine-tuning data to implant a hidden behavior), or *jailbreaking* (prompting techniques that bypass a model's safety constraints)), but it does not provide the AI-specific threat taxonomy, attack knowledge, or specialized controls needed to identify and manage those threats comprehensively.

- **Adversary capability is not fixed.**

The security design principles of *Commensurate Protection* and *Commensurate Rigor* ask an organization to scale its defenses to the severity of the threat. However, that assumes that the severity of the threat changes slowly enough for a system life cycle process to react in a timely manner to counter it. Frontier AI models can jump discontinuously with a single model release, faster than most systems engineering processes can react.

[2] Frontier AI models resist the follow-on step SP 800-160, Vol. 1 requires, which is bounding that assumed unreliability through specification and formal verification. Any physical computer, including one running a frontier AI model, has finite memory and is therefore, in the strict formal sense, a finite state machine — and a finite state machine with a fixed transition rule for every state and input is deterministic by definition. But a deployed AI system does not hold those conditions fixed in practice with stochastic sampling, implementation variability, nondeterministic execution, changing external context, and system-level nondeterminism.

- **Trust in system components cannot be bounded through specification.**

As described above, a frontier AI model's statistical, non-deterministic behavior is a poor fit for the verification techniques — formal methods, testable specifications — that the design principles *Minimal Trusted Elements* and *Compositional Trustworthiness* were designed around.

- **Good architecture does not substitute for missing threat intelligence.**

Sufficiency against a frontier-AI-enabled adversary depends on red-teaming against actual frontier models, continuously updated threat modeling, and organizational agility to change course — none of which a set of security design principles can supply on its own.

SP 800-160, Vol. 2 anticipated AI as a source of disruptive change, but only at the level of a *goal*, not a *technique*. It names artificial intelligence directly as an example of technical-environment change under the *Adapt* goal and elsewhere notes the increasing maturity of AI and machine learning as part of an evolving technical environment. That is a genuine point of foresight, but it is foresight at the *goal* level only. None of the fourteen cyber resiliency techniques in SP 800-160, Vol. 2 was written with a goal-directed, statistically-behaved model in mind as either a component to be trusted or an attacker to be defended against, which is exactly why the two problems in Section 3 require the additional protection layers described in Section 5.

## 5. A Multidimensional Protection Strategy

The multidimensional protection strategy below uses the systems security engineering concepts and principles in SP 800-160 as the engineering backbone or foundation and proposes four additional layers, each addressing a gap the foundation does not close on its own. The five-layer strategy is not a replacement for SP 800-160. It is an AI-era extension of the systems security engineering discipline. It allows organizations to: (1) build the architecture to contain failure, (2) understand the threat, (3) operate at the adversary's speed, (4) protect the integrity of what the AI relies upon and produces, and (5) continuously govern and adapt. This set because it creates a logical progression:

**Architect → Model → Adapt → Assure → Govern**

It also reinforces the central thesis that the answer to frontier AI is structural and continuous, rather than simply adding more security technologies and responding to anomalous behaviors.

### Layer 1: Architect for Containment

#### Systems Engineering Foundation (NIST SP 800-160, Volumes 1 and 2)

- *Compositional Trustworthiness* and *Minimal Trusted Elements*: Treat any frontier model, whether it is your own component or an adversary's attack surface, as untrusted by default; isolate it behind clear interfaces and never grant it implicit trust.
- *Least Privilege* and *Clear Abstractions*: Minimize what any AI-touching component can reach or see and keep interfaces simple enough that a human reviewer can actually reason about them.
- *Defense in Depth* and *Loss Margins*: Assume any single control will eventually fail against a sufficiently capable adversary; build overlapping layers with margin before a failure becomes catastrophic.
- *Anomaly Detection* and *Commensurate Response*: Build monitoring hooks and response plans that scale to the severity of what is actually being protected, not a uniform baseline.

- *Assume Compromised Resources*: Apply this strategic design principle to frontier models explicitly. Assume a model component may not behave as intended without waiting for proof of external compromise, since an unspecified statistical component satisfies the assumption natively.
- *Segmentation* and *Substantiated Integrity*: Separate system elements a frontier model can reach based on criticality, and ascertain on an ongoing basis, not once at deployment, whether elements the model has touched have been corrupted.
- *Non-Persistence* and *Unpredictability*: Shrink the time window any AI-touching resource remains present and reachable and vary configurations unpredictably where practical. Unpredictability is one of only two techniques defined specifically for adversarial threats, and it directly raises the cost of a model or attacker searching systematically for a fixed, learnable boundary.
- *Mediated Access*, *Least Functionality*, *Least Sharing*, *Reduced Complexity*, and *Structured Decomposition and Composition*: Mediate every boundary crossing through a policy-based decision point a model cannot bypass without detection; strip any AI-touching domain down to only the functions and shared resources it actually needs; and keep the architecture simple and explicitly decomposed enough that trust boundaries do not quietly erode through undocumented exceptions over time.

## Layer 2: Model the AI Threat

### Understanding the Adversary

- Map the system against MITRE ATLAS, the adversarial threat landscape for AI systems, for concrete tactics and techniques: prompt injection, model extraction, data poisoning, and evasion attacks.
- Adopt the NIST AI Risk Management Framework (AI RMF) to govern AI components specifically; where SP 800-160 addresses system trustworthiness broadly, the AI RMF governs the AI risk management activities across the AI life cycle — *Govern*, *Map*, *Measure*, and *Manage*.
- Red-team with actual frontier AI models, not only human penetration testers. The capability gap between what a human attacker can do unaided and what a frontier model can automate, including reconnaissance, phishing content generation, exploit-variant generation, must be tested directly to be understood.

## Layer 3: Adapt at Machine Speed

### Adaptive Operational Tempo

- Shrink detection and patch cycles. Static, annual assessments cannot keep pace with AI-accelerated vulnerability discovery.
- Harden human-factor defenses, including phishing-resistant multi-factor authentication and out-of-band verification, since AI most easily amplifies social engineering at scale, not just technical exploitation.
- Build machine-assisted defensive monitoring capable of operating at machine speed; employ defensive machine learning systems tuned to detect anomalies at a speed that can actually match offensive AI capability, rather than relying solely on human analyst throughput.

## Layer 4: Assure Data and Model Integrity

### Trusting the Data and Models

- Establish provenance and integrity controls over training and fine-tuning data for any model the organization operates itself, as a direct defense against poisoning.
- Treat AI-generated output, including generated code, as untrusted input: validate and sandbox it before it touches production systems, exactly as you would validate any other untrusted user input.
- Rate-limit and monitor API and query patterns to catch model-extraction or systematic probing attempts before they succeed.

## Layer 5: Govern for Continuous Adaptation

### Organizational Agility

- Give AI-specific risk clear, named ownership rather than folding it into generic information security responsibilities. Track frontier model capability releases and reassess the organization's threat model on a regular basis, because capability can jump discontinuously with a single release.
- Build a fast-path change management process specifically for AI-related findings. The standard SP 800-160 system life cycle and engineering approach assumes a change velocity that frontier AI threats may be able to outpace.
- Maintain threat intelligence feeds focused specifically on frontier AI misuse trends. Several frontier AI developers publish misuse and threat reporting. Feed that intelligence into the organization's threat model on an ongoing basis, not as a one-time input.

> ***"You should not have to trust a frontier AI component if the surrounding architecture cannot independently constrain its authority, access, and potential impact."***

Figure 1 illustrates the five layers in the multidimensional protection strategy.

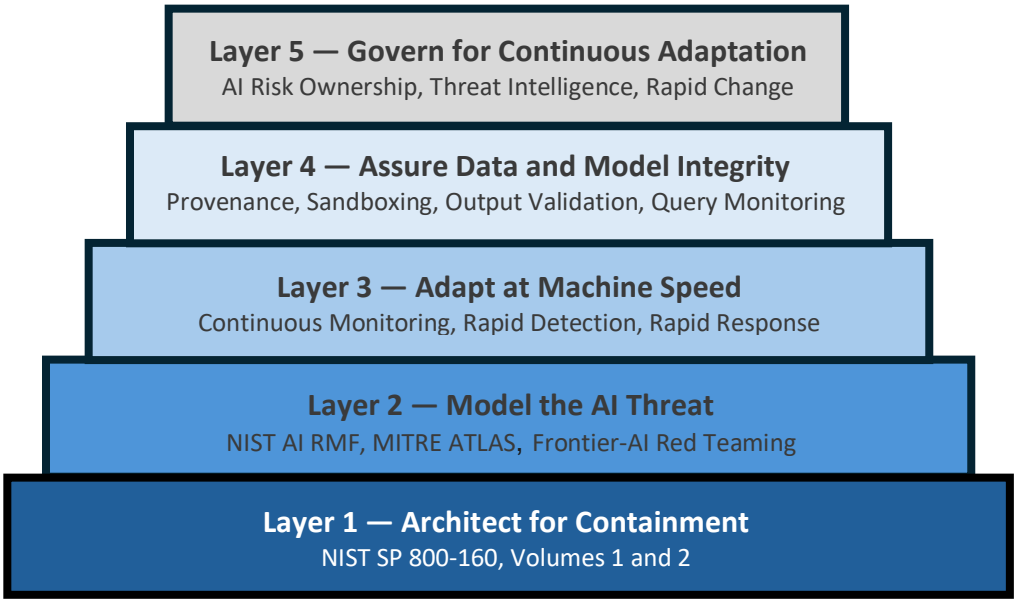


**Figure 1.** Five layers in multidimensional protection strategy

## 6. Conclusion

NIST SP 800-160 gives an organization a defensible, technology-agnostic architecture for trustworthy systems, and that architecture remains the correct starting point even against an adversary equipped with frontier AI. But applying the security design principles in SP 800-160, on their own, may be insufficient to fully defend against frontier AI. Sufficiency requires the five layers together: the systems engineering foundation, AI-specific threat modeling, adaptive operational tempo suitably matched to machine-speed adversaries, data and model integrity controls, and governance built for continuous monitoring and rapid engineering improvements.

The strategic implication for risk governance is the same one that runs through every serious discussion of mission resilience in this environment: a strategy built for a slower-moving threat cannot be treated as a finished, static compliance artifact. It has to be the foundation of a continuously maintained structure, re-engineered, re-tested and re-argued as frontier AI capability advances — not the whole house, but the part of the house that has to be built first, and built well, for everything layered on top of it to hold.

None of this is achieved by deploying more security products. It is achieved by applying rigorous systems security engineering discipline and AI extensions to the structure of the system itself. The failure mode this article has argued against, in every form it takes, is not inadequate technology. It is inadequate engineering.

> ***"The answer to frontier AI is structural and continuous, rather than simply adding more security technologies and responding to anomalous behaviors."***

## References

**[1]** Ross, R., Winstead, M., & McEvilley, M. (2022). NIST Special Publication 800-160, Volume 1, Revision 1: Engineering Trustworthy Secure Systems. National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-160v1r1

**[2]** Ross, R., Pillitteri, V., Graubart, R., Bodeau, D., & McQuaid, R. (2021). NIST Special Publication 800-160, Volume 2, Revision 1: Developing Cyber-Resilient Systems: A Systems Security Engineering Approach. National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-160v2r1

**[3]** National Institute of Standards and Technology. (2023). AI Risk Management Framework (AI RMF 1.0). https://doi.org/10.6028/NIST.AI.100-1

**[4]** MITRE Corporation. Adversarial Threat Landscape for Artificial-Intelligence Systems (ATLAS). https://atlas.mitre.org

**[5]** Anthropic. (2026). Introducing Claude Fable 5 and Claude Mythos 5. https://platform.claude.com/docs/en/about-claude/models/introducing-claude-fable-5-and-claude-mythos-5

**[6]** Ross, R. (2026). Defending Against Mythos-Class Attacks: Applying Structural Design Principles to Build Trustworthy Secure Systems. RONROSSECURE, LLC. https://lnkd.in/e87-mfW6

 