Comments in Response to CAISI's Request for Information
Security Considerations for Artificial Intelligence Agents

Docket No. NIST-2025-0035

March 9, 2026

Introduction

The Agentic Futures Initiative (AFI) appreciates the opportunity to respond to the Center for AI Standards and Innovation's (CAISI) Request for Information on securing AI agent systems.

AFI is a cross-industry policy coalition whose members span the full AI technology stack from frontier model developers and cloud infrastructure providers, to financial services platforms and data processors, to on-device and edge deployments serving defense and critical infrastructure. Our members include Anthropic, Amazon Web Services, Intuit, Plaid, Imbue, Unstructured, LILT, EdgeRunner AI, and Shift5, some of whom will also be submitting their own RFI responses. Collectively, they build, deploy, and secure AI agents across high-trust environments serving millions of users.

What AFI offers CAISI is depth and breadth. Because our membership includes both the developers building these systems and the organizations deploying them in consequential settings, we are uniquely able to provide a comprehensive perspective on agent security, one that reflects the real-world interplay between model capabilities, system architecture, deployment context, and policy constraints. We are not a trade association advancing a single commercial interest. We are a coalition organized around the shared belief that agentic AI policy should be informed by the full range of practitioners who are building and operating these systems today. Our members are actively developing operational approaches in the areas discussed below and we would welcome the opportunity to share lessons learned with CAISI as well as learn from other organizations with whom CAISI collaborates.

This response addresses NIST's prioritized questions based on insights from real-world deployments. We focus on four areas raised in the RFI: the unique security threats posed by AI agent systems, the security practices and identity frameworks needed to address them, methods for assessing agent-specific risk, and the deployment environment controls that shape what security is even possible.

Three core themes emerge:

1. AI agents present security challenges that are fundamentally different from traditional software. Existing cybersecurity frameworks provide a foundation, and CAISI guidance can fill in potential gaps about how those frameworks apply to the way agents actually work.
2. The field is moving faster than the standards. Organizations deploying agents today are navigating a fragmented landscape and CAISI has an opportunity to accelerate alignment around shared frameworks.

3. Agent identification and verified authorization are critical policy questions that CAISI is well positioned to help resolve. As agents increasingly act across organizational boundaries and on behalf of users, establishing trusted identity, defined credentials, and clear records of agent activity is becoming as important as user authentication is today.

AFI and its members welcome the opportunity to serve as an ongoing resource to CAISI as it develops standards and guidance in this space. We would welcome the chance to brief CAISI staff, participate in working groups or listening sessions, and provide further input as this work evolves.

1. Security Threats, Risks, and Vulnerabilities Affecting AI Agent Systems

Question 1(a): What are the unique security threats, risks, or vulnerabilities currently affecting AI agent systems, distinct from those affecting traditional software systems?

Securing AI agent systems is a challenge that AFI's members are actively navigating today. Drawing on that experience, we have identified several areas where the security surface of AI agents differs meaningfully from traditional software and where targeted guidance from CAISI would have the greatest impact:

Agents can be manipulated into misusing the tools they are given. Unlike traditional software, which executes predefined functions within a fixed scope, agents are designed to dynamically select and use tools (e.g., APIs, code executors, web browsers, file systems, communication platforms) to complete tasks. That flexibility is also a vulnerability because when an agent is manipulated into misusing a tool, the harm isn't contained to the software environment. It can result in sent emails, deleted files, executed transactions, or exfiltrated data.

Prompt injection attacks manipulate agents through carefully crafted natural language inputs, potentially causing them to ignore instructions and execute unauthorized actions, like leaking sensitive data. This is different from SQL injection where hackers would hide malicious code inside something like a form field on a website and software could be programmed to spot and then block code that didn't belong. With prompt injection, the "attack" is just words, the same kind an agent is supposed to understand and follow, so there's no easy way to filter it out. This risk intensifies when agents process external content like documents or web pages where malicious instructions can be hidden inside what looks like normal text (i.e., indirect injection).

Data security concerns are amplified in multi-tenant environments where agents process information for multiple customers. Traditional software retrieves and displays data within clearly defined, developer-specified boundaries. Agents dynamically access, synthesize, and act on information across multiple sources in ways that are difficult to fully anticipate or constrain beforehand which makes clean data isolation harder to enforce. Without proper controls, an agent might surface one client's data in another's context, incorporate sensitive information into shared memory, or pass confidential details to downstream agents or external tools, often without any single action appearing obviously wrong.

Goal misalignment presents another novel challenge: agents may pursue objectives that differ from what their user actually intended. This is not because it malfunctioned but because it optimized for what it was told in a way that missed the point. An agent optimizing for efficiency might take shortcuts that violate policies or an agent that was asked to organize files might delete content it considers unnecessary. Traditional software has no capacity to interpret intent, fill in gaps, and make judgment calls like agentic AI can. It follows explicit rules. The more autonomous the agent, the more opportunities there are for its interpretation of a goal to drift

from what the user actually wanted. Unlike a software bug, misalignment will not always be obvious. This risk is compounded in multi-agent systems, where one agent's misaligned output becomes another agent's input, potentially cascading into larger unintended consequences before any human notices.

Agentic AI systems also introduce unique accountability challenges. When an AI agent causes harm, responsibility is difficult to assign because its behavior emerges from a combination of factors such as design, institutions, context, and tool access that no single party fully controls. Developers, operators, and users each shape the system's behavior in different ways, but none has complete visibility into how those inputs interact in any given situation. Our current liability frameworks, built around human decision-makers and deterministic systems, do not give us a clear answer about distributing responsibility across that chain. This leaves real questions about who bears the legal and ethical consequences when something goes wrong: the developer who built the system, the company that deployed it, or the person who used it.

Question 1(d): How have these threats, risks, or vulnerabilities changed over time? How are they likely to evolve in the future?

The threat landscape continues to evolve rapidly as the capabilities of the systems themselves evolve rapidly. Early AI systems were narrow and reactive, limiting both their autonomy and their attack surface. As systems became capable of multi-step reasoning, tool use, and autonomous action, the threat landscape expanded accordingly.

Prompt injection, for example, was a theoretical concern in early language model deployments but became an operational one as agents gained access to tools and the ability to act on their outputs. Similarly, goal misalignment moved from largely an academic AI safety concern to a practical security issue as agents were deployed in consequential settings with limited human oversight. Multi-agent architectures represent the most significant recent escalation. When organizations chain agents together, a vulnerability in one system can propagate across the entire pipelines, amplifying the harm and making it harder to trace the source.

Several trends are likely to intensify as agents become more capable and widely deployed, including:

1. The attack surface will grow with capability. As agents gain access to more tools and operate in more domains, there are simply more ways for attackers to exploit them.
2. Attack methods will be more sophisticated. Rather than generic manipulation, adversaries will develop techniques tailored to specific systems that target how a particular agent stores memory or processes instructions.
3. Agent impersonation and supply chain attacks will emerge as significant vectors as malicious actors attempt to introduce compromised agents into trusted pipelines.
4. The gap between agent deployment and the development of governance frameworks creates its own risk. Efforts like CAISI's newly launched AI Agent Standards Initiative, together with organizations like the Agentic Futures Initiative, represent important progress toward closing that gap, bringing together industry and policymakers to discuss the issues and develop the frameworks that deployment decisions increasingly require.

2. Security Practices and Agent Identification

Question 2(a): What technical controls, processes, and other practices could ensure or improve the security of AI agent systems in development and deployment? What is the maturity of these methods in research and in practice?

Effective agent security requires multiple layers of protection, each addressing a different point of vulnerability.

1. At the model level, developers train systems to resist manipulation, detect harmful requests in real-time, and behave consistently under adversarial conditions.
2. At the system level, architectural choices matter, such as whether agents are limited to specific tools, whether inputs and outputs are validated, and whether limits are applied on what data can be used. Due to the agent's dynamic nature, the limits require constant adjustment versus a one-time set-up.
3. At the governance layer, human oversight is an important design principle with approval requirements calibrated to risk. Routine actions can proceed autonomously while high-stakes decisions may require explicit human authorization. High-stakes decisions may include modifying production data, initiating financial transactions, or communicating sensitive information externally. This is an important balance to get right in deploying agents responsibly and practical CAISI guidance on how to think through these tradeoffs would be very useful to practitioners.

For financial services and other regulated industries, agents need additional controls to ensure agents operate within compliance boundaries. Transparency mechanisms show users how agents reached decisions, building trust while enabling verification. When agents recognize their limitations because they hit the edge of what it's authorized or capable of doing, there should be a clear escalation pathway to human experts.

As agents increasingly act across organizational boundaries, establishing trusted agent identity and scoped authorization will become as critical as user authentication is today. CAISI is well positioned to convene industry around common approaches. Proper identity verification serves as a critical threshold for access and action, particularly in sensitive domains like financial services. Before agents access user data or perform operations on users' behalf, robust verification mechanisms should confirm both the agent's identity and its authorization to act. Questions to ask include:

- Is this agent actually from the organization it claims to represent?
- Did the user whose data is being accessed actually give this agent permission to act on their behalf?
- Is the agent staying within the specific boundaries in which it was authorized to operate?
- Is there a clear, reliable record of what the agent accessed and did?

These verification requirements protect users while enabling legitimate agent applications to function effectively.

Question 2(e): Which cybersecurity guidelines, frameworks, and best practices are most relevant to the security of AI agent systems?

i. What is the extent of adoption by AI agent system developers and deployers of these relevant guidelines, frameworks, and best practices?

ii. What are impediments, challenges, or misconceptions about adopting these kinds of guidelines, frameworks, or best practices?

iii. Are there ways in which existing cybersecurity best practices may not be appropriate for the security of AI agent systems?

Several existing cybersecurity frameworks provide partial guidance for agentic AI systems.

1. The most directly applicable are NIST's Cybersecurity Framework (CSF) and AI Risk Management Framework (AI RMF), offering useful structures for governance, risk mapping, and incident response.
2. NIST SP 800-53 provides relevant controls around access management and audit logging.
3. OWASP's Top 10 for Large Language Models has gained traction as a practitioner-facing resource specifically addressing prompt injection, insecure tool use, and data leakage.
4. MITRE ATLAS, which catalogs adversarial tactics targeting AI systems, is increasingly referenced by experienced practitioners.

None of these were designed with agentic systems in mind.

(i) Adoption is uneven. Large frontier AI developers have invested in red-teaming and adversarial testing, though these practices are not always documented or independently verified. Enterprise deployers vary widely, often applying traditional software security frameworks without adapting them for agentic contexts. Smaller developers may lack dedicated security resources altogether, though well-designed agent systems could themselves help fill that gap over time. No widely adopted agentic-specific security standard currently exists so organizations will need to piece together guidance from frameworks that were not designed for this technology.

(ii) Agents are being deployed faster than security standards can keep up, leaving practitioners without clear guidance. Even where frameworks exist, they require interpretation to apply to agent systems, which can lead to very different approaches across organizations. Addressing these challenges effectively requires people who understand both traditional cybersecurity and AI. This is why initiatives like CAISI's AI Agent Standards Committee matter. They bring those communities from all sectors into the same room, combining cybersecurity and AI expertise to develop shared standards based on how these systems actually work and the risks they present. It's also important to not assume that safety work done during model development is sufficient for deployment. A well-aligned model can still be exploited through prompt injection or tool misuse, and the deployment environment can introduce risks that model-level controls cannot fully anticipate.

(iii) It's important to understand that security controls, processes and governance should not treat all agentic AI systems as equivalent. A one-size-fits-all regulatory or security framework would either overburden low-risk applications or under-regulate high-impact ones. Instead, oversight mechanisms should be tiered and proportional to risk. This may include incorporating graduated controls such as stronger authentication, tighter access controls, mandatory human-in-the-loop review, logging and audit requirements, red teaming, and formal risk assessments for higher-impact deployments. Some additional considerations to keep in mind when considering how to adapt existing frameworks include:

1. Traditional access control models assume that permissions are relatively fixed, but agents can gain access to new tools and data sources mid-task in ways that static permission structures were never designed to handle.
2. Standard logging practices capture what a system did but not why. Understanding an agent's behavior often requires reconstructing its reasoning, not just its actions.

3. Incident response frameworks assume you can trace a clear sequence of events within a bounded system. When multiple systems are acting and interacting simultaneously, that kind of clean reconstruction may not be possible.
4. Different regulators and jurisdictions may develop different expectations. This could create compliance complexity for organizations.
5. Given how fast the technology is moving, principles-based vs. prescriptive standards would mitigate the likelihood that rules become outdated before or right after they are finalized.

3. Assessing Security

Question 3(a): What methods could be used during AI agent systems development to anticipate, identify, and assess security threats, risks, or vulnerabilities?

i. What methods could be used to detect security incidents after an AI agent system has been deployed?

ii. How do these align (or differ) from traditional information security practices, including supply chain security?

iii. What is the maturity of these methods in research and applied use?

iv. What resources or information would be useful for anticipating, identifying, and assessing security threats, risks, or vulnerabilities?

AFI members have found that effective security assessment for AI agent systems requires both updating existing cybersecurity practices and developing new approaches suited to how agents actually work. Key areas where CAISI's engagement would be valuable include:

1. Evolving existing threat modeling frameworks to address agent-specific risks. Established tools like STRIDE are widely used and provide a good starting point, but the practitioner community is increasingly recognizing they need to be extended for agentic contexts. Assessments need to account for risks specific to agentic systems, including prompt injection, tool misuse, memory manipulation, and vulnerabilities that emerge in multi-agent coordination. While common standards have not yet emerged, MITRE ATLAS and the Cloud Security Alliance's MAESTRO framework represent potential promising early efforts. CAISI can play a valuable role in helping practitioners navigate this landscape and accelerating convergence around shared frameworks.
2. Treat red-teaming as a distinct discipline. Adversarial testing of agent systems goes beyond traditional penetration testing. It requires understanding how language models can be manipulated, how errors compound across multi-step tasks, and how to examine agent reasoning processes to catch problems before deployment. Invest in runtime monitoring. Testing an agent before launch is necessary, but not enough. Once deployed, agents respond to real-world conditions that no test environment can fully anticipate and when something goes wrong, one needs to be able to see what happened and why. That requires logging systems that can track an agent's full reasoning process across multiple steps, not just record individual commands or API calls. There are currently no widely accepted standards for what good agent monitoring looks like. CAISI guidance in this area would give developers and deployers a shared baseline from which to work, making it easier to detect problems early, investigate incidents when they occur, and improve agent security over time.
3. Update supply chain security for AI-specific risks. AI agents introduce supply chain risks beyond traditional software, including model weights, training data integrity, third-party tools, and external APIs. Existing practices like Software Bill of Materials (SBOMs) remain relevant but need to be extended. The concept of an AI Bill of Materials (AI-BOM) that documents the models, datasets, and tools a system relies on in addition

to the code is already gaining traction in the research and practitioner community, and CAISI should consider building on that existing work.

4. Information sharing mechanisms do not need to be built from scratch. The groundwork for sharing AI security threat intelligence already exists with established mechanisms like MITRE ATLAS and sector-specific ISACs. We encourage CAISI to prioritize convening existing stakeholders and identifying gaps in current sharing frameworks as well as connecting efforts that may be happening in silos. The objective is to make it easier for organizations across sectors to share what they're seeing on a voluntary basis, with appropriate sensitivity to legal and competitive considerations, so the whole ecosystem gets smarter about identifying and responding to emerging risks more quickly and effectively.

Question 3(b): Not all security threats, risks, or vulnerabilities are necessarily applicable to every AI agent system; how could the security of a particular AI agent system be assessed and what types of information could help with that assessment?

The right level of scrutiny depends on what an agent is doing, where it operates, and what safeguards are already in place. We recommend CAISI develop a voluntary risk-tiering framework, similar in approach to NIST's existing risk management tools, that includes the following key factors:

1. What is an agent's scope? An agent that can read documents presents a very different risk profile than one that can execute transactions, send communications, or otherwise interact with external systems. The more an agent can do and the more consequential those actions are, the more rigorous the security assessment should be. Scope includes the breadth and sensitivity of systems, databases, and environments that the agent can access.
2. What are the potential consequences? This includes the real-world impact of a compromise and the ability to remedy unintended consequences.
3. How much human oversight is there? An agent that acts on its own requires much more careful pre-deployment testing than one where a person reviews and approves its actions before anything happens. How autonomous an agent is should be a core factor in any risk assessment.
4. What rules already apply? Sectors like financial services, healthcare, and defense already have serious security and compliance requirements. Where possible, security guidance for AI agents should build on existing frameworks rather than add a whole new layer of obligations on top of it.
5. Is it one agent or multiple? When agents coordinate with or delegate to other agents, new risks emerge that one simply wouldn't catch by looking at each agent in isolation. Systems where multiple agents interact need to be evaluated as a whole, not just part by part.

Whatever tiering framework CAISI develops, organizations should know what their agents can reach before those agents are deployed, including the specific datasets, APIs, and external services with which each agent is authorized to interact. The AI Bill of Materials (AI-BOM) concept discussed above is an example of a vehicle for this, extending a traditional software inventory beyond what an agent is built from to include what it is permitted to access.

A well-designed, context-specific risk-tiering framework helps ensure security investments align with actual risks and should be developed through direct engagement with the practitioners who will use it.

4. Deployment Environment Controls

Question 4(a): AI agent systems may be deployed in a variety of environments, i.e., locations where the system's actions take place. In what manner and by what technical means could the access to or extent of an AI agent system's deployment environment be constrained?
Question 4(b): How could virtual or physical environments be modified to mitigate security threats, risks, or vulnerabilities affecting AI agent systems? What is the state of applied use in implementing undoes, rollbacks, or negations for unwanted actions or trajectories (sequences of actions) of a deployed AI agent system?
Question 4(d): What methods could be used to monitor deployment environments for security threats, risks, or vulnerabilities?

> ***i. What challenges exist to deploying traditional methods of monitoring threats, risks, or vulnerabilities?***
> ***ii. Are there legal and/or privacy challenges to monitoring deployment environments for security threats, risks, or vulnerabilities?***
> ***iii. What is the maturity of these methods in research and practice?***

The environment in which agents operate fundamentally shapes what security is even possible.

The most effective controls like network segmentation, least-privilege access, and tenant isolation work by limiting what an agent can reach and do in the first place. However, these controls can be harder to implement for agents than for traditional software because agents make dynamic decisions that are difficult to anticipate at design time, requiring ongoing calibration rather than a one-time configuration.

Recovery is complicated by the fact that many agent actions, like a completed transaction or sent message, cannot be cleanly undone and an agent may have taken dozens of interconnected steps by the time the issue is caught. An important design principle here is whether an action can be reversed should be a factor the system evaluates before an action is taken, not a characteristic that only becomes apparent after something has gone wrong. Actions that are difficult or impossible to undo may warrant additional automated checks and, in some cases, explicit human confirmation before execution. CAISI guidance reinforcing this principle would help practitioners build it into their systems as a default and could scale across sectors and deployment contexts without requiring prescriptive rules about specific actions.

Monitoring is essential but not straightforward because agent behavior is naturally variable, making it hard to define what normal looks like. Layered on top of all of this are existing compliance requirements. Privacy laws and sector-specific legal requirements, like GLBA in financial services and the state privacy frameworks for companies handling personal data, vary across deployment contexts and were not written with AI agents in mind. Since AI agents will process significantly more personal data than traditional software systems, these frameworks will need to evolve to ensure what is required to manage agents responsibly respects the data minimization principles and does not erode the broader privacy protections they were designed to safeguard.

AFI and its members are well-positioned to help CAISI work through these questions and welcome the opportunity to engage further.

5. Additional Considerations

Question 5(b): In which policy or practice areas is government collaboration with the AI ecosystem most urgent or most likely to lead to improvements in the state of security of AI agent systems today and into the future?

- Agent Identification and Authentication: As agents start acting across organizations and on behalf of users, there is no shared way to verify who an agent is, who authorized it, or what it is allowed to do. We are encouraged to see NIST giving this issue the attention it deserves, including through the NCCoE's recently released concept paper, *Accelerating the Adoption of Software and AI Agent Identity and Authorization*. More broadly, CAISI is well positioned to convene industry around common approaches to agent identity before everyone builds their own solutions that make alignment harder to achieve down the road.

- Procurement and government use of AI agents: Federal agencies are already evaluating AI agents for internal use, but procurement frameworks have not caught up. Government collaboration with industry on agent-specific procurement standards would both improve the security of government deployments and send a clear market signal that helps the broader ecosystem converge around shared practices.

Conclusion

AI agents offer significant productivity and capability benefits, but realizing these benefits responsibly requires thoughtful security practices and policies. The core principles that emerge from AFI's member experience are consistent across deployment context, including: layer multiple protections rather than relying on any single control, calibrate oversight to actual risks, adapt proven cybersecurity frameworks to AI-specific challenges, and monitor and share findings continuously as capabilities evolve.

AFI welcomes the opportunity to collaborate with CAISI and its broader AI Agent Standards Initiative on this very important work through briefing CAISI staff, participating in working groups and listening sessions, and/or sharing the practical experience our members have built deploying agents at scale across some of the most demanding environments in the country. The organizations best positioned to inform these standards are the ones already navigating these challenges every day, and AFI is proud to contribute our members' perspective to this effort.

We appreciate CAISI's leadership on this important topic.

Respectfully submitted,

Ryan Dattilo
Executive Director
info@agenticfuturesinitiative.org

Agentic Futures Initiative