CRA and CCC's Response to the Notice of Request for Information regarding Security Considerations for Artificial Intelligence Agents.

March 11, 2026

Written by: *Rachel Greenstadt (New York University), Michela Taufer (University of Tennessee, Knoxville), Ming Lin (University of Maryland, College Park), Manish Parashar (University of Utah), David Jensen (University of Massachusetts Amherst), and Brian Mosley (Computing Research Association).*

This response is prepared by the Computing Research Association (CRA), assisted by the Computing Community Consortium (CCC). CRA is an association of over 270 North American computing research organizations, both academic and industrial, and partners from six professional computing societies.

The mission of CCC, a CRA subcommittee, is to enable the pursuit of innovative, high-impact computing research that aligns with pressing national and global challenges.

Please note any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the authors' affiliations, or of the National Science Foundation, which funds the CCC.

AI Agent Security Challenges

There are serious security threats and vulnerabilities specific to AI agent systems. One of the most significant is that debugging an AI agent is incredibly difficult. In the past, software development has benefited from extensive measures like code review and testing to catch major flaws. However, the inherent adaptability of AI agent systems introduces significantly greater unpredictability. It has the potential to make assurance efforts much more difficult.

Simultaneously, the speed that failures can escalate into potentially problematic actions is amplified. The attribute that makes AI agents so useful is their ability to learn from mistakes, often utilizing techniques such as reinforcement learning. Through a reward function, the system can distinguish between successful and failed attacks. Over time, an AI agent develops a memory, essentially a record of these attempts, allowing it to identify and focus on the most successful attack strategies. This iterative learning process significantly enhances the AI agent's effectiveness. But that increased effectiveness will amplify any errors caused by faults in the agent's code.

The field's current security models are inadequate for autonomous agents. Specifically, there are some main challenges:

- security models and credential delegation, the methods for delegating credentials to an agent remain unclear;
- context-dependent risk, an agent's potential, and its security risk, is highly dependent on who is using it, what they are doing, and in what context. This context is inherently unpredictable for an autonomous agent;
- Accountability for security failures caused by agents may be difficult, especially when the agents themselves are the product of AI-generated code;
- cascading failures in Multi-Agent Systems can occur, where a single compromised agent can quickly poison the decision-making of downstream agents within hours; and
- other potential vulnerabilities may include high-speed, high-stakes actions in autonomous AI agents, without immediate human approvals, can limit the time available to stop a malicious action, adversarial agents extract models and sensitive data to cause harm, etc.

Our existing security frameworks were not defined for such adaptive, autonomous agency and are not mature enough to respond effectively to these challenges. This is a crucial issue, especially as these agents are used not just for malicious attacks but for general automation. The fundamental question is whether we have the necessary frameworks to manage these risks.

The issues of credential delegation, combined with other factors, introduce a compelling risk profile. For instance, jailbreaking could potentially alter an agent's loyalty and objectives, and hallucinations might be triggered under specific circumstances by certain actions. What is needed is a way to audit the actions of an AI agent, something that does not currently exist.

In addition, new research and techniques to handle potential cascading failures in large-scale multi-agent systems need to be developed to quickly detect catastrophic events, adversarial attacks and thefts in digital and physical autonomous systems to ensure safety of all users.

Threats From an AI Workforce Imbalance

The increasing trend of replacing human workers with AI programs in coding and related tasks poses a significant risk. Crucially, the immediate future still demands human expertise in complex coding and a deep understanding of these AI agents. Despite the potential for AI and other tools to assist with code auditing, without the deep expertise of a human expert who grasps the underlying systems, how they function, and how they generate their outputs, the field faces serious dangers if AI adoption accelerates. To mitigate the risk of a workforce imbalance, the nation must stress the necessity of a balanced strategy toward AI development that continues to have human expertise in-the-loop.

Although NIST cannot mandate adoption, it can set widely used expectations by translating “human-in-the-loop” into specific, testable practices. Building on the NIST AI Risk Management Framework and its Playbook, NIST should publish implementation guidance for AI coding agents that defines (i) accountable human roles (e.g., code owner, security reviewer, model operator), (ii) review/approval gates for AI-generated contributions, (iii) documentation

requirements for provenance, limitations, and changes introduced by the agent, and (iv) measurement metrics and criteria for when human oversight is adequate in high-risk contexts.

In particular, we need to encourage skill development and workflows where humans can specify requirements clearly and securely, then validate and evaluate results.

Regulatory Alignment and Compliance for Agentic Behavior

A critical dimension to consider is the alignment of agentic behavior with existing legal and regulatory frameworks. Specifically, how can compliance with these laws and regulations be ensured? This issue is a frequent topic during legislative sessions across the nation, particularly given the focus on introducing regulations for deepfakes, child protection, and related concerns. A key challenge is determining how these regulations can be effectively applied to the actions of agentic systems.

A careful balance is needed when creating regulations for industry versus the research community, as significant differences could negatively affect public-private partnerships at academic institutions. Although the work can be similar, the public impact of industry and academic research differs greatly in timing, scale, and method. Imposing excessive constraints on academic researchers prematurely risks stifling innovation, yet a lack of regulation could lead to the exploitation of the general public by opaque black-box systems offered by industry.

Administrations at some research universities are trying to fill this void, though the results could have serious unintended consequences for the progression of research. As one example, that is not recommended, a research university system is considering a new role of “AI Risk Officers.” Under this new concept, AI research must be approved in advance. Such an approach raises several immediate questions and potential problems:

- Staffing: Given the nascent nature of the field, who possesses the requisite expertise to serve in such a role?
- Research Freedom: If research is not approved, can it still be conducted?
- Field Impact: If universities begin to impede their own ability to conduct this research, entire fields of research face potential stagnation.

This example illustrates potentially excessive attempts to exert some control in the legal and regulatory vacuum.

AI Agent Security Challenges

At a high level, the era of autonomous AI agents is just beginning, making it difficult to rely on past experience to guide current security considerations regarding these new systems. Nevertheless, it would be beneficial to examine previous periods of major leaps in abstraction within computing to identify any applicable lessons.

The rise of autonomous code presents a scaling problem, analogous to previous technological shifts, suggesting an inevitable increase in broken systems. While lessons can be drawn from past experiences with software taking actions and traversing networks (which introduced new and interesting threats) the qualitative difference and true extent of translation remain unclear. New abstraction layers, like the shift to higher level languages from assembly code and now the shift from programming languages to natural language prompts, provide opportunities for new vulnerabilities caused by ambiguities in the translation process between these layers.

Another area to keep in mind is that modern software development practices rarely involve comprehensive, in-house, end-to-end testing (i.e. vertical testing). Agents, in particular, may be constructed using a diverse array of tools and middleware, which presents a significant risk: the potential for malicious "poisoning" of the codebase, impacting security.

This is a general challenge for most AI-developed software, but it is acutely problematic for autonomous agents due to the subtlety and rapidity of potential vulnerabilities. When a system can act flexibly and autonomously, testing for edge cases becomes substantially more difficult. The breadth of an agent's possible responses complicates "fuzz testing." It may exhibit strange interactions or security flaws, like input sanitization failures, which only manifest under specific, hard-to-predict conditions. The very concept of an input sanitization failure in this dynamic context needs careful re-evaluation.

The involvement of agents in code development introduces another challenge, as their actions might include code creation itself. Conversely, the offense-defense dynamic remains unclear in these early stages. Some are exploring the use of agents for security purposes, specifically agents designed to police other agents, identify issues, and respond accordingly. How this will ultimately unfold is an open question. Typically, cybersecurity professionals operate under the assumption that attackers have an advantage, needing to find only one vulnerability, while defenders must cover everything. It is currently unknown to what extent agents will be able to enhance and scale defense capabilities; this needs to be looked at very closely.

A significant security concern arises from the unpredictable nature of multi-agent systems, particularly when agents possess conflicting goals. These interactions create a volatile security dynamic. Broadly, agent interactions fall into two categories. One is where agents collaborate to achieve a common objective. The other is where different agents, often controlled by separate entities, pursue divergent goals. The latter scenario is more complex and could potentially lead to novel, security-affecting outcomes. Experiments have already shown that agents, when pursuing a goal in semi-adversarial, role-playing scenarios, may employ unexpected and overly creative methods. When agents are pursuing conflicting goals, they may adopt methods that human operators would not sanction or approve of.

AI Security and Detection Strategies

The current pace of attack development is outstripping AI models, a trend consistent across all safety research. A significant challenge to safety efforts is the fact that interpretability remains

far behind capability. This gap is severely complicated by the major threat posed by malicious actors using multi-agent systems to automate and scale up existing attack vectors. Research is ongoing to secure these systems, but the fundamental challenge is defining effective defenses against such large-scale, automated attacks.

While there are existing guardrails that are common in commercial AI models, they are designed to prevent accidental harms or non-expert usage (such as age-verification). Such guardrails will not work against malicious attackers using self-learning AI agents, which will learn and understand the underlying coding and any exploits at a speed and level a human never could. NIST should invest significant research support into looking for effective guardrails that could be embedded with AI agents framework or that could withstand assaults from such systems.

AI Security by Design

As pointed out in the CCC White Paper A Research Ecosystem for Secure Computing, "more often than not, securing a system happens after the design or even deployment, meaning the security community is routinely playing catch-up and attempting to patch vulnerabilities that could be exploited any minute." Put another way, the question of security is routinely placed a distant second to application capability. There needs to be a sea-change in the way the field approaches this critically important aspect of the problem. NIST should explore ways of creating incentives for industry to make security by design in AI systems, not something that is added later and if possible.

Additionally, stronger connections are needed between the research community and industry cybersecurity practitioners. These pathways would provide researchers with practical experience and knowledge of how security professionals address real-world problems. This practical understanding could then inform the design and development of future systems.

A significant concern is the current inability to verify if a human, or even another AI agent, is interacting with an AI agent. This lack of verification poses serious security risks, especially for commercial applications. There should be significant research efforts around how to integrate a protocol into fundamental AI agent systems.