American Academic Scientific Research Journal for Engineering, Technology, and Sciences

ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

https://asrjetsjournal.org/index.php/American_Scientific_Journal/index

Agentic AI Security & Autonomous Red-Teaming

Ashok Kumar Kanagala*

Independent Researcher, Boston, MA, USA

Email:Kanagala.ashok@ieee.org

Abstract

Recent progress in foundation models and multi-agent orchestration systems has increased their capabilities and also their attack surface. Cyber-physical systems and edge devices serve both as a target of deployment and as an enabler of operation. The security issues surrounding these enabling mechanisms are already becoming a reality, but the implications of these issues on AI-driven ecosystems are under-researched. In contrast to traditional security areas, threats in agentic AI environments are difficult to anticipate due to their dynamic execution contexts, lack of standardized operational baselines, and the unpredictable behaviors arising from autonomous and emergent agent strategies. This paper examines these challenges and proposes a forward-looking security approach centered on continuous model verification, alignment assurance, and transparency tooling tailored to agentic systems. The framework emphasizes early, automated, and lifecycle-integrated security validation, augmented by autonomous red-teaming to proactively surface weaknesses. The findings suggest that embedding self-assessing security mechanisms into agentic AI pipelines enables more resilient, adaptive, and accountable intelligent systems.

Keywords: Agentic AI Security; Autonomous Red-Teaming; AI Vulnerability Assessment.

1. Introduction

The rapid evolution of artificial intelligence has given rise to a new class of systems commonly described as agentic AI—autonomous, goal-directed entities capable of reasoning, planning, tool use, and coordinated action across complex environments. Unlike traditional machine learning models that operate within narrowly defined and largely static boundaries, agentic AI systems are increasingly deployed in dynamic, open-ended contexts where they can interact with other agents, software services, cyber-physical systems, and human users. These capabilities offer transformative potential for cybersecurity, particularly in areas such as automated threat detection, continuous validation, and adversarial testing. At the same time, they introduce novel security risks that challenge existing assumptions about control, predictability, and accountability in intelligent systems.

Received: 11/29/2025
Accepted: 1/29/2026
Published: 2/7/2026

----

* *Corresponding author.*

One of the most promising and disruptive applications of agentic AI is autonomous red-teaming—the use of self-directed agents to continuously probe systems for vulnerabilities without direct human supervision. This paradigm reflects a broader shift in cybersecurity toward proactive, upstream, and lifecycle-integrated security practices. Similar to how DevSecOps reshaped software engineering by embedding security into development pipelines, agentic AI enables security mechanisms that can reason, adapt, and evolve alongside the systems they protect. However, the autonomy that makes these agents effective defenders can also turn them into sources of risk, especially when their behavior emerges from complex interactions rather than explicitly programmed rules.

The literature on AI security has expanded rapidly in response to threats such as model poisoning, prompt injection, and adversarial inputs. Yet much of this work remains focused on static models or single-agent scenarios. The security implications of fully autonomous, multi-agent systems—particularly those capable of independent tool use and system-level access—remain underexplored. This gap is especially concerning given the growing integration of agentic AI with distributed edge environments, Internet of Things (IoT) infrastructures, and cyber-physical systems, where errors or misaligned actions can have immediate real-world consequences.

Agentic AI security differs fundamentally from conventional cybersecurity challenges. Emergent behaviors, non-deterministic decision paths, and adaptive strategies undermine traditional threat modeling approaches that rely on known attack vectors and predefined system states. Furthermore, multi-agent coordination introduces cascading failure modes, communication vulnerabilities, and monitoring challenges that cannot be easily decomposed into individual components. When agents are granted autonomous access to tools, APIs, or external systems, the attack surface expands beyond the model itself to include toolchains, execution layers, and trust boundaries across heterogeneous environments.

This paper investigates the security implications of agentic AI with a specific focus on autonomous red-teaming as both a defensive capability and a source of systemic risk. We analyze the technological trends driving increased agent autonomy, examine key vulnerability classes unique to agentic and multi-agent architectures, and explore how automated adversarial testing reshapes the cybersecurity lifecycle. In doing so, we argue that future security frameworks must move beyond reactive controls toward continuous, self-assessing, and agent-aware security validation models. By situating agentic AI within the broader evolution of proactive cybersecurity, this work contributes a structured foundation for securing intelligent systems that are increasingly capable of acting and attacking on their own.

2. Literature Review

Early studies of agent-driven cyber operations linked their development to advances in large-scale reasoning engines and self-evolutionary feedback loops, a developer-level interest that was reminiscent of early Web evolution research [1]. The modern conceptualization of agentic AI is much more about autonomous tool use, system-level coordination, and adaptive behavior functions that allow self-directed security judgments but create new uncertainties. Just like decentralized architectures in Web 3.0 raised serious concerns about trust, resource efficiency, and governance [2], agentic AI raises similar concerns in automated cybersecurity. The AI security literature is still growing, but the discourse on autonomous agent-specific issues, especially those that can perform red-team work without any human involvement, remains in its early stages and is relatively underdeveloped. The enablers of Agentic AI, such as model-based reasoning, autonomous planning, and the integration with cyber-physical systems, are rapidly changing and can transform adversarial testing approaches at a faster pace than governance structures can keep pace [3]. Technologies like IoT and distributed edge intelligence only increase this complexity, which generates operational environments in which autonomous agents can interact with the digital and physical realms at the same time.

However, little analysis has been done on how agentic AI can transform the engineering of security per se, especially in the context of the expected transition to ongoing, upstream, and fully automated threat assessment.

This paper will seek to address this new terrain by looking at how agentic AI and autonomous red-teaming will transform the practice of cybersecurity. We examine the trends of development that lead towards agent autonomy, the weaknesses of multi-agent and self-directed systems, and the changing role of automated adversarial testing in the overall security lifecycle. We also reflect on the ways in which security integration models, similar to DevSecOps, need to evolve to an ecosystem where autonomous agents are both defenders and potential attack vectors. In this light, we explore the future of proactive security in a world that is becoming more intelligent and self-governing in terms of digital systems [4].

3. Problem Statement: Security Challenges in Autonomous Agentic AI Systems

The increased use of agentic AI presents a range of complicated security issues that are not similar to those presented by conventional machine learning systems. In contrast to the static models, agentic architectures are autonomous, proactive, and can interact with tools and environments in dynamic and often unpredictable manners Reference [12]. This combination greatly increases the attack surface, and at the same time, it adds new failure and exploitation modes. These are some of the risks that should be understood to design safe deployment strategies that would ensure alignment, oversight, and resilience in environments where agents might make decisions that impact critical systems in real time [13].

A. Emergent Behaviors and Loss of Predictability

Among the most significant security issues related to agentic AI, one can note the development of unpredictable behavior in autonomous decision-making. Since these systems are based on multi-step reasoning, dynamic planning, and adaptive feedback loops, they can produce new strategies that are not the intended ones. The emergent behaviors, though in most cases useful in problem-solving, may also lead to security threats when agents act outside the authorized scope of operations [14].

Specifically, reasoning models that include iterative self-improvement or automatic tool invocation may be driven into unexpected states by ambiguous data or adversarial inputs. It is this uncertainty that is intensified in those environments where agents are exposed to other agents, external processes, or live systems. The non-deterministic behavior is a challenge to the traditional security methods based on a predefined threat model or fixed rules [15]. The common testing methodologies do not fully represent the combinatorial space of possible agent behavior, and thus failure modes that are rare but significant are hard to predict.

Emergent behavior also leaves the prospect of subtle misalignment, where the objective that is inferred by the agent is not what the developer intended. Agents that are misaligned can seek shortcuts, use unintended system functionality, or ignore implicit safety constraints, especially when performing complex cyber operations. This may cause unsafe code execution, unauthorized access, or manipulation of the functionality of other agents. New control and interpretability systems are being suggested as researchers investigate these uncertainties to limit emergent behavior without compromising the adaptive benefits of agentic autonomy [16].

B. Multi-Agent Coordination and Communication Vulnerabilities.

Multi-agent systems have their own security issues, especially in the coordination and communication of distributed agents. When several independent bodies work together on the same tasks, there are possible cascading vulnerabilities between them, which are hard to disconnect. Weak or rogue agents may interfere with the working processes, distort the information shared, or cause unintentional activities throughout the system. Coordination failure may also occur due to inconsistent reasoning states, race conditions, or resource competition, even in the absence of malicious interference.

Another point of exposure is the communication channels between agents. One malicious input can change the decision-making behavior of a group, affect the system's integrity, or mislead cooperative agents to perform harmful actions.

The multi-agent environment also makes the task of monitoring more difficult because the behavior of the group cannot be easily broken down into individual behaviors. Such opaqueness prevents security analysts from tracking the cause of errors, imposing accountability, or determining the cause of a failure. Researchers are investigating solutions like hierarchical controller architecture, consensus-based validation, and inter-agent trust mechanisms. Nevertheless, with the increase in the complexity of multi-agent coordination, the issue of secure, transparent interaction between agents will continue to be at the center of agentic AI security studies [17].

C. Threats in Autonomous Tool Use and System-Level Access

Another major security issue with agentic AI architectures is the use of autonomous tools, which is one of the most potent and dangerous features of such architectures.

In addition, the autonomous use of the tool introduces the system to tool-chain vulnerabilities. In case an external tool or API has security vulnerabilities, the agent can inadvertently activate malicious behavior or become a victim of exploitation. Attackers can also seek to hijack tool interfaces, alter execution outputs, or inject malformed data to corrupt the decision processes of the agent. Such interactions provide adversarial influence opportunities, which act on the execution layer but not the underlying model [18].

Another issue is the problem of agent autonomy in mixed-trust settings. When agents are exposed to the external systems, the third-party software, or the open networks, it is hard to enforce strict control over their behavior. Even perfectly aligned agents can take counterproductive actions when they misunderstand the output of tools, or when they make inaccurate risk assessments, or when they are faced with unforeseen environmental factors. To achieve safe tool use, it is necessary to have multiple layers of protection, such as permission gating, real-time monitoring, traceable execution logs, and action filters that are consistent with policies.

4. Solution: Agentic AI Security Through Autonomous Red-Teaming and Continuous Validation

Addressing the security challenges posed by agentic AI systems requires a fundamental shift from static, perimeter-based defenses toward continuous, autonomous, and system-aware security mechanisms. Because agentic architectures are dynamic, self-directed, and capable of emergent behavior, effective security solutions must operate at the same level of autonomy and adaptability as the systems they protect. This section outlines a multi-layered solution framework centered on autonomous red-teaming, continuous verification, controlled tool use, and governance-aware security integration. Together, these approaches aim to reduce uncertainty, constrain harmful emergence, and embed security as a core property of agentic AI lifecycles.

A. Autonomous Red-Teaming as a Continuous Security Primitive

Autonomous red-teaming represents a central pillar of securing agentic AI systems. Unlike traditional penetration testing, which is episodic and human-driven, autonomous red-teaming employs AI agents that continuously probe systems for weaknesses, misconfigurations, and unintended behaviors. These agents can simulate adversarial strategies, adapt their attack paths based on system responses, and explore edge cases that are infeasible to test exhaustively through manual methods.

In agentic AI environments, red-teaming agents can be designed to mirror the reasoning capabilities, tool access, and coordination patterns of production agents. This symmetry allows defenders to evaluate not only known vulnerabilities but also emergent risks arising from multi-step reasoning, inter-agent communication, and autonomous decision-making. Importantly, autonomous red-teaming shifts security validation upstream, enabling vulnerabilities to be identified during development, deployment, and runtime rather than after failures occur.

To avoid introducing new risks, red-teaming agents must operate within controlled sandboxes, with strict boundaries on system impact and rollback mechanisms. Their findings should feed directly into automated remediation pipelines, enabling rapid iteration and continuous hardening. In this model, security becomes an ongoing adversarial dialogue between defensive and offensive agents rather than a static checklist activity.

B. Containing Emergent Behavior Through Continuous Verification and Alignment Checks

Given the inherent unpredictability of agentic AI, eliminating emergent behavior is neither feasible nor desirable. Instead, the solution lies in constraining emergence within acceptable operational and ethical boundaries. Continuous verification mechanisms can monitor agent behavior in real time, comparing observed actions against policy constraints, alignment objectives, and expected behavioral envelopes.

These mechanisms include runtime policy enforcement, invariant checking, and anomaly detection based on behavioral baselines rather than fixed rules. By focusing on deviations in intent, scope, or resource usage, security systems can identify subtle forms of misalignment that may not manifest as overt attacks. For example, an agent that repeatedly selects high-risk tools to optimize performance may signal objective drift, even if no explicit policy violation occurs.

Alignment checks should also be iterative and contextual. As agents learn from feedback and adapt to new environments, their alignment must be reassessed continuously rather than assumed to be static. Integrating interpretability tools, such as decision trace analysis and reasoning path inspection, enhances human oversight and supports post-incident accountability without undermining agent autonomy.

C. Securing Multi-Agent Coordination and Communication

To address vulnerabilities arising from multi-agent coordination, security solutions must treat agent collectives as first-class security entities rather than aggregations of individual components. One effective approach is the introduction of hierarchical or supervisory control agents that validate plans, mediate resource allocation, and arbitrate conflicts among subordinate agents. This structure limits cascading failures while preserving decentralized execution.

Communication security is equally critical. Inter-agent messages should be authenticated, integrity-checked, and contextually validated to prevent message tampering, spoofing, or poisoning. Consensus-based validation mechanisms can require agreement from multiple agents or controllers before high-impact actions are executed, reducing the risk of a single compromised agent influencing the entire system.

Transparency in coordination is also essential. Logging and traceability systems that capture inter-agent interactions enable forensic analysis and real-time monitoring. While full observability of complex agent collectives is challenging, partial visibility into coordination patterns can significantly improve anomaly detection and accountability.

D. Controlled Autonomous Tool Use and Execution Safeguards

Autonomous tool use is both a strength and a primary risk factor in agentic AI systems. To mitigate tool-chain vulnerabilities, agents must operate under strict permission models that define which tools can be used, under what conditions, and with what scope of access. Fine-grained access control, combined with contextual risk assessment, allows agents to select tools responsibly without unrestricted execution authority.

Execution safeguards such as sandboxing, rate limiting, and output validation further reduce the impact of compromised tools or malicious inputs. For example, before executing a high-risk action, an agent may be required to obtain secondary validation from a supervisory agent or pass a simulated execution check. These safeguards act as friction points that slow down potentially harmful actions without eliminating autonomy altogether.

Additionally, comprehensive execution logging and provenance tracking are critical for post hoc analysis and regulatory compliance. By maintaining traceable records of tool usage and decision rationales, organizations can better understand failures, assign responsibility, and refine security policies over time.

E. Integrating Agentic Security into the AI Lifecycle

Finally, securing agentic AI requires embedding these solutions into the broader AI development and deployment lifecycle. Similar to how DevSecOps integrated security into continuous integration and deployment pipelines, agentic AI security must be lifecycle-aware, spanning model training, agent design, deployment, and runtime operation.

This integration includes automated security testing during development, continuous validation in production, and adaptive governance mechanisms that evolve alongside agent capabilities. Autonomous red-teaming, alignment verification, and tool-use controls should not be treated as add-ons but as foundational components of intelligent system design.

Organizations can ensure that agentic AI systems remain trustworthy, resilient, and accountable by aligning technical safeguards with emerging regulatory and ethical frameworks. In this way, proactive, agent-aware security architectures provide a viable path forward for harnessing the benefits of agentic AI while managing its inherent risks.

5. Recommendations: Operationalizing Secure AI Supply Chains

The implementation of agentic red-teaming into the software development lifecycle needs to be an organized effort that integrates active security testing with ongoing monitoring. In contrast to standard penetration testing, autonomous red-teaming makes use of agentic AI systems that are able to produce and implement multi-step adversarial strategies on their own.

The initial step to integration is to establish clear goals, allowable areas of operation, and alignment limits of the agents. These requirements make sure that autonomous evaluations are significant and maintain safety and compliance.

Initial research and field tests of agentic red-teaming models show that they are able to identify weak points that traditional testing can easily miss. Independent agents could model advanced attack conditions, find vulnerabilities in configuration, and locate logic errors in multi-layered systems.

The findings show that vulnerability coverage has increased significantly, critical issues have been discovered faster, and the insight into the possible failure paths has been enhanced. Furthermore, reasoning models, multi-agent coordination, and autonomous tool use enabled these agents to work 24 hours, which offers 24-hour security assessment that is out of the reach of red-teaming teams that are manually staffed [19].

The findings also indicate the possibilities and difficulties associated with agentic red-teaming. On one hand, autonomous agents raise efficiency, identify new weak points, and permit proactive security positions. On the other hand, agent-to-agent interaction, emergent behavior, and autonomous tool execution are difficult and involve new risks and issues that need to be managed effectively, audited, and aligned. The results show that agentic red-teaming is quite a promising concept, yet organizations must find the necessary balance between freedom and restraint and implement multiple safety layers and checks, and continuous monitoring to ensure reliable outcomes Reference [20].

A further analysis of agentic interactions reveals how well-organized coordination practices and the arrangement of trust between autonomous agents are required. Misunderstandings or lack of coherence between two or more agents may unwittingly increase system vulnerabilities instead of lessening it, and escalate into localized effects that are hard to predict. To ensure the stability of red-team environments in multi-agent cases, it is crucial to create a common objective, confirmed communication methods, and agreement mechanisms. Such provisions can minimize the chances of unwanted system behaviors but enable agents to search through various attack paths in an efficient manner. Emergent behavior is still a double-edged sword.

Though adaptive decision-making enables the agents to detect previously unknown vulnerabilities and maximize the testing strategies, it also provides variability that puts the risk tolerance of organizations to the test. Monitoring systems should then be developed to identify anomalies, indicate that the behavior is not as expected, and give human operators an interpretable definition of what the agent is doing. Explainable AI approaches can also be used to fill the gap between autonomous decision-making and human supervision, so that the security teams still have a clear view of agents' thought processes. Lastly, autonomy in executing tools requires the implementation of strict access controls, sandboxing, and audit logging in real-time.

The misuse of the tools based on the misaligned goals, wrong execution, or control-driven manipulation options can deter the advantages of agentic red-teaming when uncritically followed. The layered protection and rollback systems, as well as constant compliance controls, will guarantee that agents work under safe and predictable limits. Generally, agentic red-teaming is an opportunity to positively change how security is implemented, but it should be architecturally designed, well supervised, and continually optimized to achieve all benefits without causing other systemic risk.

6. Conclusion

Red-teaming autonomously and using agentic AI is a cybersecurity paradigm shift. Despite the emerging challenges of uncertainty and complexity of operations, the described framework reveals that the integration of intelligent agents into safe red-teaming workflows is possible. The next step of work, then, should be to refine the oversight mechanisms, increase interpretability, and standardize the best practice in order to make agentic red-teaming an essential part of safe software development [21].

References

[1] A. Patil, N. Patel, and S. Deshpande, "Ethical Decision-Making in Sustainable Autonomous Transportation: A Comparative Study of Rule-Based and Learning-Based Systems," Cogent Engineering, vol. 11, no. 12s, 2025. [Online]. Available: https://doi.org/10.64252/cgzh6r94

[2] M. DeBellis and R. Neches, "Knowledge Representation and the Semantic Web: An Historical Overview of Influences on Emerging Tools," Recent Advances in Computer Science and Communications, vol. 16,

no. 6, pp. 22–36, Jul. 2023. [Online]. Available: https://doi.org/10.2174/2666255815666220527145610

[3] Cloud Security Alliance and OWASP AI Exchange, "Agentic AI Red Teaming Guide," 2025. [Online]. Available: https://cloudsecurityalliance.org/artifacts/agentic-ai-red-teaming-guide

[4] A. Dawson, R. Mulla, N. Landers, and S. Caldwell, "AIRTBench: Measuring Autonomous AI Red Teaming Capabilities in Language Models," arXiv:2506.14682, 2025. [Online]. Available: https://arxiv.org/abs/2506.14682

[5] HiddenLayer, "Indirect Prompt Injection of Claude Computer Use," HiddenLayer Innovation Hub, 2024. [Online]. Available: https://hiddenlayer.com/innovation-hub/indirect-prompt-injection-of-claude-computer-use/

[6] K. Huang, "Agentic AI Threat Modeling Framework: MAESTRO," Cloud Security Alliance, Feb. 6, 2025. [Online]. Available: https://cloudsecurityalliance.org/blog/2025/02/06/agentic-ai-threat-modeling-framework-maestro

[7] K. Huang, G. Huang, Y. Duan, and J. Hyun, "Utilizing Prompt Engineering to Operationalize Cybersecurity," in Generative AI Security: Theories and Practices, K. Huang, Y. Wang, B. Goertzel, Y. Li, S. Wright, & J. Ponnapalli, Eds., Springer, 2024, pp. 271–303. [Online]. Available: https://doi.org/10.1007/978-3-031-54252-7_9

[8] K. Huang, V. Manral, and W. Wang, "From LLMOps to DevSecOps for GenAI," in Generative AI Security: Theories and Practices, K. Huang, Y. Wang, B. Goertzel, Y. Li, S. Wright, & J. Ponnapalli, Eds., Springer, 2024, pp. 241–269. [Online]. Available: https://doi.org/10.1007/978-3-031-54252-7_8

[9] K. Huang, J. Huang, and C. Hughes, "AI Agents in Offensive Security," in Agentic AI: Theories and Practices, K. Huang, Ed., Springer, 2025, pp. 167–205. [Online]. Available: https://doi.org/10.1007/978-3-031-90026-6_6

[10] Invariant Labs, "MCP Security Notification: Tool Poisoning Attacks," Invariant Labs Blog, May 26, 2025. [Online]. Available: https://invariantlabs.ai/blog/mcp-security-notification-tool-poisoning-attacks

[11] Z. Wang, C. Q. Knight, J. Kritz, W. E. Primack, et al., "A Red Teaming Roadmap Towards System-Level Safety," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/xxxx.xxxxx

[12] M. Feffer, A. Sinha, W. H. Deng, Z. C. Lipton, et al., "Red-teaming for Generative AI: Silver Bullet or Security Theater?," Proceedings of the AAAI Conference, 2024. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/xxxx

[13] S. Ghosh, B. Simkin, K. Shiarlis, S. Nandi, D. Zhao, et al., "A Safety and Security Framework for Real-World Agentic Systems," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/xxxx.xxxxx

[14] A. Sinha, K. Grimes, J. Lucassen, M. Feffer, et al., "From Firewalls to Frontiers: AI Red-Teaming is a Domain-Specific Evolution of Cyber Red-Teaming," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/xxxx.xxxxx

[15] B. Ren, E. J. Cheon, and J. Li, "Organization Matters: A Qualitative Study of Organizational Dynamics in Red Teaming Practices for Generative AI," Proceedings of the ACM on Human-Computer Interaction, 2025. [Online]. Available: https://dl.acm.org/doi/xxxx

[16] I. Wicaksono, Z. Wu, R. Patel, T. King, et al., "Mind the Gap: Comparing Model-vs Agentic-Level Red Teaming with Action-Graph Observability on GPT-OSS-20B," arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/xxxx.xxxxx

[17] V. Saarainen, "Red Teaming: Regulatory and Non-Regulatory Frameworks Used in Adversarial Simulations," Theseus.fi, 2021. [Online]. Available: https://www.theseus.fi/handle/10024/xxxx

[18] B. Challita and P. Parrend, "RedTeamLLM: An Agentic AI Framework for Offensive Security," arXiv preprint arXiv:2505.06913, 2025. [Online]. Available: https://arxiv.org/abs/2505.06913

[19] R. Singh, B. Blili-Hamelin, and C. Anderson, "Red-Teaming in the Public Interest," Data & Society, 2025. [Online]. Available: https://ranjitsingh.me/red-teaming-public-interest

[20] K. Huang and C. Hughes, "Agentic AI Red Teaming," in Securing AI Agents, Advances in Data Analytics, AI, and Smart Systems (ADAASS), pp. 207–252, 2025.

[21] S. Majumdar, B. Pendleton, and A. Gupta, "Red Teaming AI Red Teaming," in Conference on Applied Machine Learning for Information Security (CAMLIS) 2025, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2507.05538