State of AI in the Cloud 2026

How AI Adoption, Autonomy, and Attacker Innovation Are Reshaping Cloud Security

Table of Contents

Table of Contents

Introduction

Artificial intelligence is the most significant technology shift since the start of the cloud era. In just a few years, AI systems have gone from limited rollouts to foundational infrastructure embedded across applications, developer workflows, and business operations. This transition is unlocking enormous opportunity, but it is also reshaping the security landscape.

This report captures a snapshot of that transition. With analysis grounded in real-world cloud environments and observed attacker behavior, it explores how AI is being deployed and attacked today. The findings reinforce a critical point: securing AI is not just about protecting models. It's about understanding how AI systems interact with infrastructure, identities, data, and automation, and adapting security practices accordingly.

Executive Summary

AI has moved from exploratory use to infrastructure. In 2026, the defining shift is not whether organizations use AI, but how deeply AI is embedded across cloud environments and how it reshapes risk in practice. Increasingly, AI functions less like a standalone tool and more like a foundational layer embedded across development workflows, orchestration layers, and production systems.

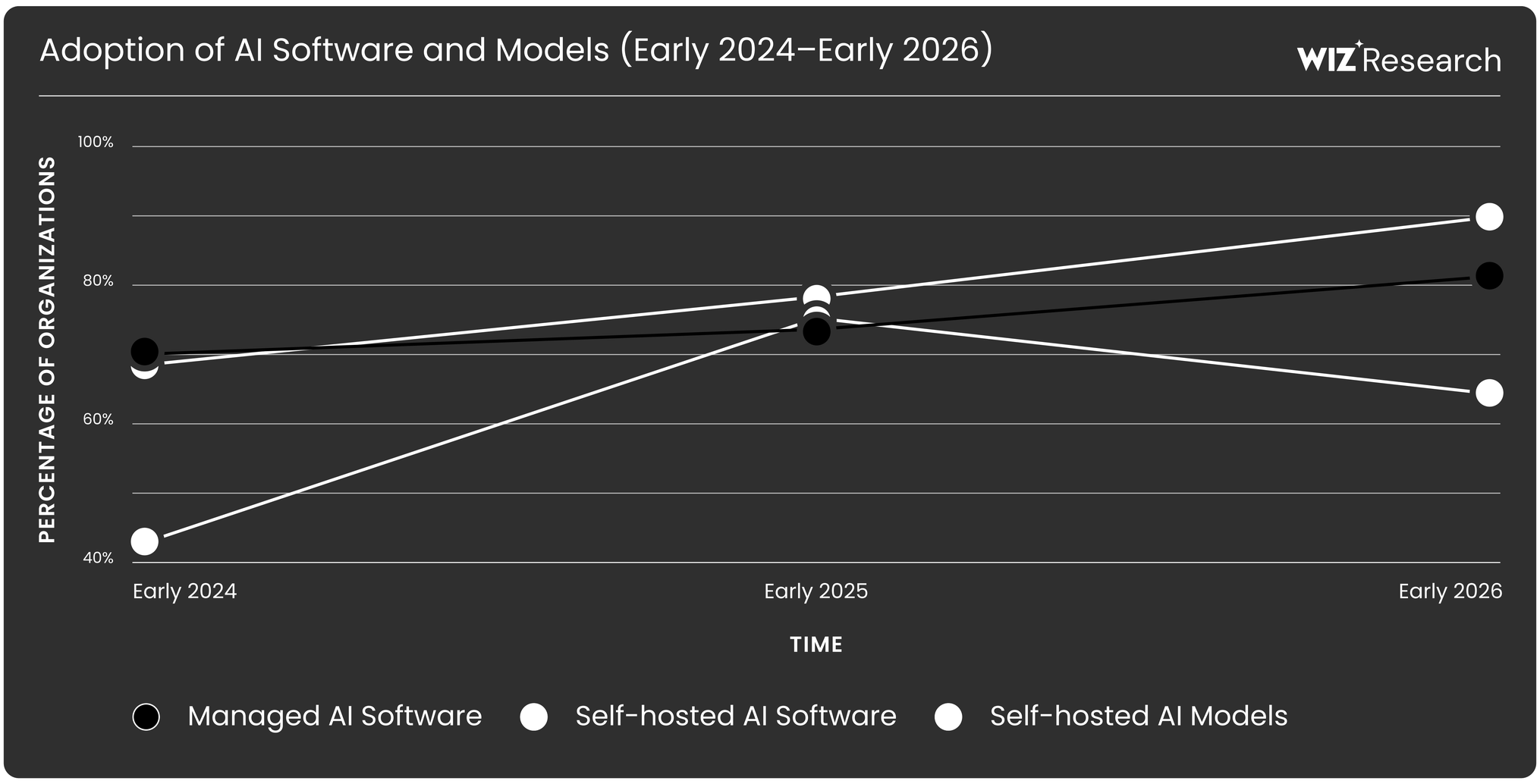
- **AI Is Now Core Cloud Infrastructure:** AI adoption has stabilized at scale. At least 81% of cloud environments we observed use managed AI services, and at least 90% run self-hosted AI software. AI is now present across cloud environments, development workflows, and automation tooling.
- **AI Adoption Extends Beyond Direct Deployment:** Our data shows that 63% of organizations are self-hosting AI models. Among these, 68% ingest such models at least partially through third-party software, and 18% rely exclusively on such transitive AI components. As AI becomes a standard feature of cloud software and development tooling, organizations increasingly inherit AI capabilities through vendors and integrations. The security focus therefore expands from what is explicitly deployed to what is operating across the broader application ecosystem.
- **AI-Powered Development is the Default, with Large Implications for the Ecosystem:** At least 80% of organizations have developers using AI IDE extensions, and 71% have at least one AI coding assistant present. In September 2025, Wiz Research found that roughly one in five organizations using AI-powered vibe-coding platforms had applications affected by systemic security weaknesses. When AI-generated defaults replicate at scale, insecure patterns can become systemic rather than isolated defects.
- **Agents and MCP Server Adoption Expand the Effective Attack Surface:** At least 57% of organizations have deployed self-hosted AI agent technologies, and MCP servers appear in at least 80% of cloud environments, with 5% Internet-facing. These figures reflect rapid adoption of orchestration infrastructure and the emergence of new control-plane layers that must be secured.

- **AI Broadens What Can Be Reached:** AI systems increasingly connect to APIs, data stores, credentials, and operational workflows, extending access paths across cloud environments. As these connections deepen, attackers can more easily move between systems and reach sensitive data through existing integrations, as seen in real-world cases of exposed model servers and abused AI-enabled integrations.
- **AI Is Changing the Economics of Exploitation:** AI is both a target and an accelerant. Observed incidents and benchmark data suggest AI can reduce experimentation cost and accelerate exploit development. Rather than introducing entirely new attack classes, AI compresses timelines, lowers the skill floor, and scales familiar techniques.

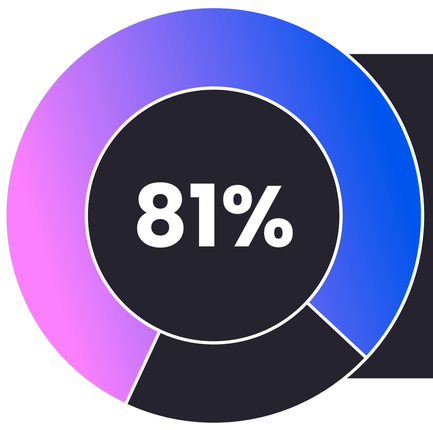
Together, these findings reinforce a central conclusion. AI security is not a future discipline. It is an extension of cloud security that must account for autonomy, automation, and the rapid spread of AI-driven systems across development and production environments.

1. AI Is Now Operational Infrastructure

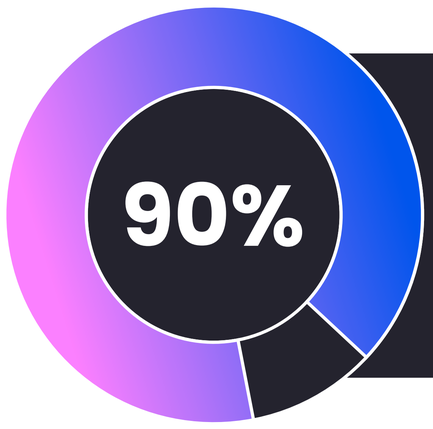
Wiz Research observes AI adoption across nearly every dimension of the cloud stack. Across the cloud environments analyzed by Wiz Research, at least 81% use managed AI services. This figure is up from 74% in early 2025, reflecting continued but steady growth.



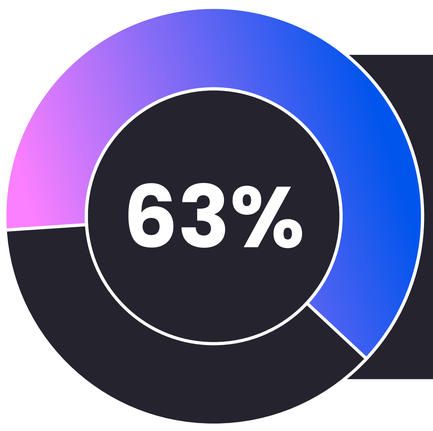
AI usage now spans managed services, self-hosted software, models, and development tooling, reflecting AI's transition from isolated rollouts to widespread operational presence.



At least 81% of organizations are using managed AI services.



At least 90% of organizations leverage self-hosted AI software.



At least 63% of organizations leverage self-hosted AI models.

AI adoption spans every major industry, including highly regulated sectors such as finance, energy, and aerospace, underscoring that AI security is no longer a niche concern but a cross-industry challenge.

However, as AI became more embedded, early indicators suggested governance was already struggling to keep pace with adoption. In Wiz's 2025 Security Readiness Survey, 87% of respondents said they were using AI services, **yet 25% lacked visibility into which AI services were running** in their environment, providing early context for the trends explored throughout this report.

1 From Consumption to Ownership

Managed AI models remain central. At least 81% of organizations use them in some form, and **at least 40% actively deploy them to support agents or automated workflows.**

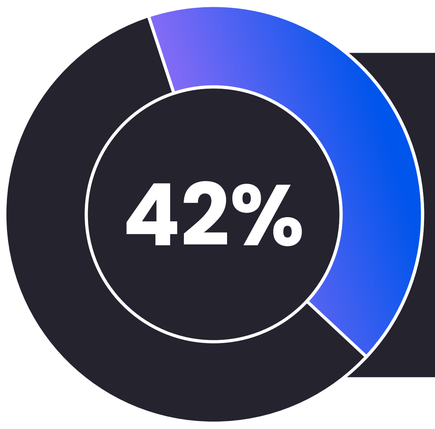
At the same time, self-hosted AI continues to expand. At least 63% of organizations now show evidence of self-hosted AI models, while adoption of self-hosted AI software has grown to at least 90%. This reflects a shift from pure consumption of managed services toward greater ownership and embedding of AI capabilities within cloud environments.

2 AI Ownership Is Often Indirect

However, ownership is not always intentional. At least **68% of organizations running self-hosted models do so at least partially transitively,** meaning the models are bundled within third-party software they deploy. At least **18% rely exclusively on these transitive self-hosted models,** suggesting that some organizations may be operating self-hosted AI components without fully realizing it.

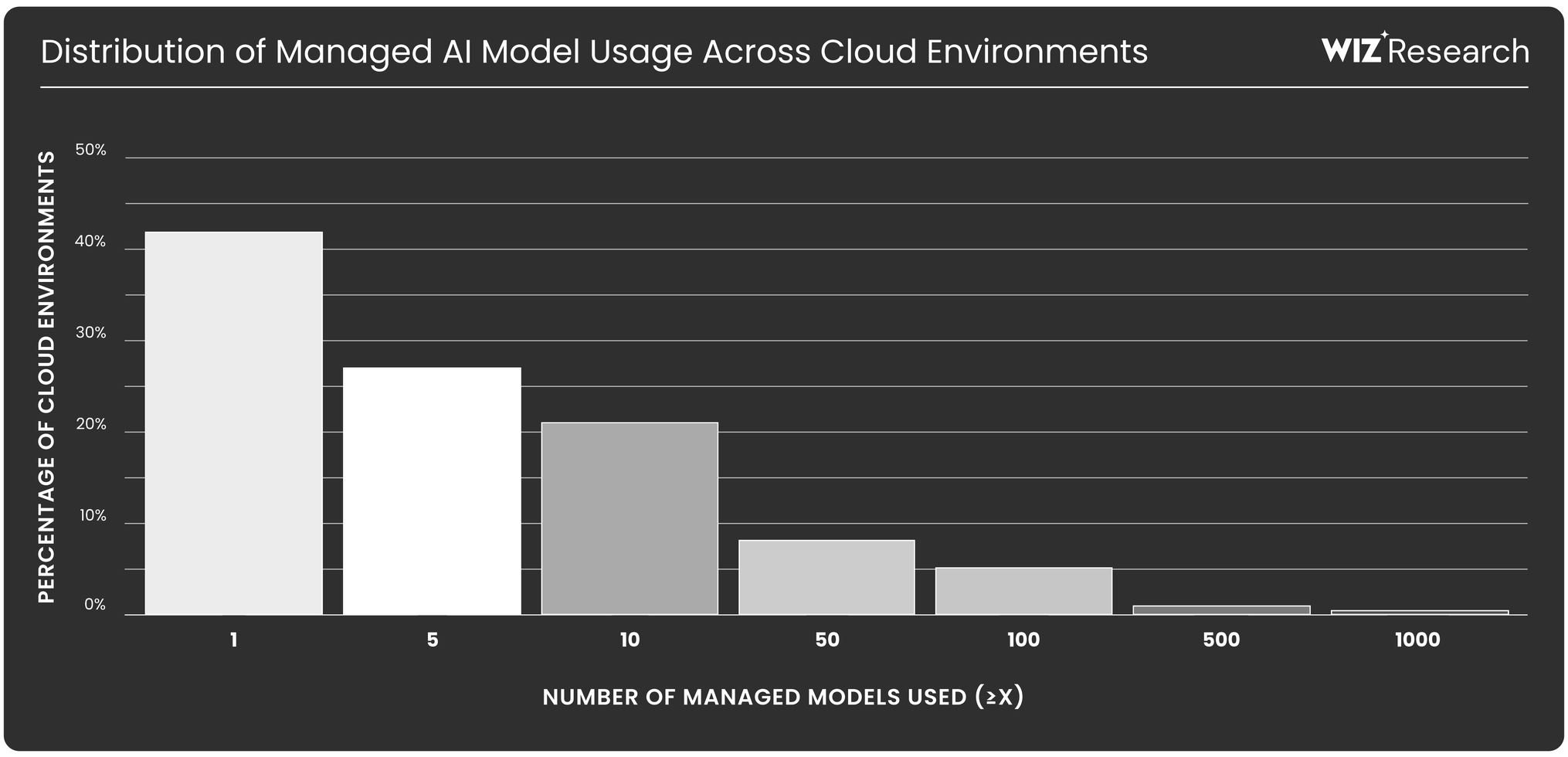
3 Model Usage Patterns Reflect Maturity, Not Just Adoption

Looking beyond whether organizations use AI, model usage patterns reveal how AI is being operationalized across cloud environments. Most organizations using managed AI models rely on fewer than ten distinct models, with 42% depending on a single model.



42% of organizations depend on a single AI model. Fewer than 7% deploy more than 100.

This consolidation around a small number of foundational models further suggests that for many organizations, AI has moved beyond isolated pilots into steady, production-oriented deployment.



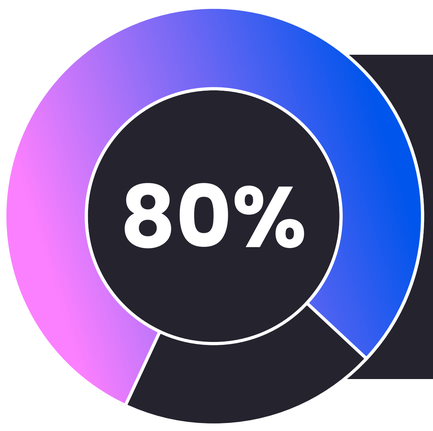
At the same time, a meaningful subset shows deeper engagement. Twenty-one percent of organizations operate ten or more managed models, typically reflecting more advanced use cases, internal model development, or fine-tuning for specialized workloads. Only a limited group, fewer than 7%, deploy more than one hundred managed models, signaling extensive model development, deep adoption of AI, and perhaps more diverse requirements across the organization.

Taken together, these patterns highlight a clear divide in AI maturity. **While adoption is widespread, depth of model usage varies significantly.** Most organizations prioritize a small set of trusted models, while deeper model proliferation remains concentrated among a leading cohort pushing AI further into products and operations.

As AI usage expands through agents, orchestration layers, and AI-assisted development, these maturity differences increasingly shape how organizations experience both opportunity and risk in practice.

2. AI-Assisted Development Is the Default, with Systemic Effects Across Applications

AI-assisted development is now part of standard developer workflows. Wiz Research observes that at least 80% of organizations are operating with AI IDE extensions, reflecting a shift in how code is written, reviewed, and shipped.



At least 80% of organizations use AI IDE extensions. At least 71% use AI coding assistants.

Clarifying definitions:

AI IDE Extension: A plugin added to a developer's integrated development environment (IDE) that enables AI-powered capabilities.

AI coding assistant: An AI-powered assistant that works alongside developers in a step-by-step, human-driven loop. Coding assistants respond to prompts, generate code, explain logic, and assist with problem-solving, but execution remains primarily guided by the developer.

AI Coding Agent: An AI system capable of performing multi-step development tasks autonomously. Agents can plan, execute, and iterate across tools and services with limited human input.

The Key Distinction:

IDE extensions describe the integration surface, while Coding assistants and Agents are two types of AI-powered tools available as IDE extensions.

1 AI-Assisted Development Is Now a Default Behavior

AI coding assistants have become the primary way developers interact with AI. Embedded directly into IDEs and code review workflows, these coding assistants help with code generation, refactoring, and routine development tasks as part of day-to-day work rather than standalone pilot usage.

Wiz Research finds that at least 71% of organizations have at least one AI coding assistant present in their environment. In most cases, coding assistants are adopted bottom-up by developers rather than deployed as centrally managed platforms, meaning AI-assisted development is already widespread even where formal AI initiatives remain limited.

Industry data reinforces how deeply these tools are shaping software production. The GitHub Octoverse 2025 report describes AI as a "day-one" reality for new engineers, with **80% of new developers adopting AI coding assistants within their first week** on the platform. GitHub also observed a 25% year-over-year increase in total code pushes, though it emphasizes these are observational signals rather than proven causal effects of AI coding assistants adoption.

The scale of activity is significant. External analysis from LogicStar AI and ETH Zürich suggests that AI agents now participate in up to 10% of public pull requests, often achieving high merge rates for widely used tools. While methodologies vary, the directional signal is clear: AI is no longer peripheral to development. It is contributing materially to the volume of production code.

2 AI Coding Assistant Adoption Is Broad, but Fragmented

Coding assistant adoption does not converge on a single tool. A small number of well-established coding assistants from major providers, including **GitHub Copilot, Claude Code, and OpenAI Codex,** have strong presence across environments, reflecting trust in familiar ecosystems and mature integrations. At the same time, Wiz Research observes a long tail of open-source, niche, and newly released coding assistantss, often installed by individual engineers to solve specific problems or explore new capabilities. These tools frequently appear outside centralized governance processes, creating pockets of "shadow AI" within development environments.

This fragmentation increases complexity. Security teams must account for multiple AI systems influencing code generation, often without standardized policy enforcement, review controls, or unified telemetry.

3 Speed and Scale Turn Coding Errors into Systemic Risk

The security implications of AI-assisted development are structural, not incidental.

When coding assistant adoption is fragmented and largely unguided, application risk shifts from isolated bugs to repeatable, systemic weaknesses. Applications built rapidly with AI assistance often replicate the same insecure patterns across projects, such as exposed credentials, permissive data access policies, and missing authentication, rather than introducing one-off implementation errors.

In September 2025, Wiz Research found that roughly one in five organizations using AI-powered vibe-coding platforms had applications affected by systemic security issues. These weaknesses stemmed from shared generation patterns and defaults rather than individual developer mistakes.

Wiz subsequently worked with platforms including Lovable to share these findings and improve guardrails, reducing repeated exposure at scale.

Taken together, these patterns show that AI-assisted development has already become routine. AI coding assistants are lightweight, developer-driven, and tightly embedded in everyday workflows, which is why they spread quickly even without centralized oversight.

The risk is not simply that coding assistants exist, but that they scale insecure patterns as easily as they scale productivity. When AI-generated code, configurations, and access patterns are repeated across projects, small mistakes become systemic weaknesses rather than isolated defects.

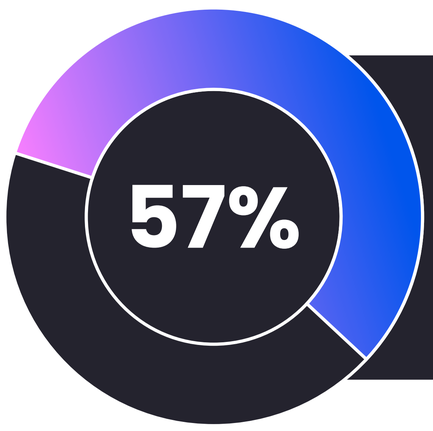
Agents extend this risk further. While coding assistants support developers, agents act on their behalf across tools, data, and services, shifting AI from advisory input to autonomous execution and raising the stakes for governance, visibility, and control.

3. Agents and MCPs Are Expanding the Cloud Attack Perimeter

Beyond models, APIs, and developer coding assistants, organizations are increasingly adopting agents and Model Context Protocol (MCP) servers to orchestrate AI interactions with systems and data. These technologies represent a shift from AI that assists humans to AI that can act autonomously across environments.

1 Agents Are Entering Production, Fast

Wiz Research finds that at least 57% of organizations have deployed at least one self-hosted AI agent technology, indicating strong early adoption for a category that was largely absent from production environments a year ago. Agent frameworks are most commonly used to automate development tasks, connect AI systems to external tools, and orchestrate workflows that span multiple services.

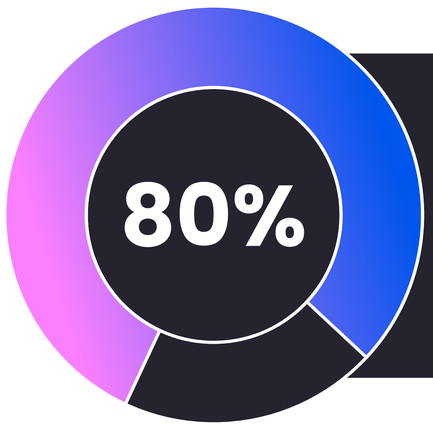


At least 57% of organizations have deployed self-hosted AI agent technologies.

Agent adoption is also notably fragmented. Wiz Research observes a broad range of agentic frameworks and implementations across environments, with **no single framework emerging as dominant.** This diversity reflects a rapidly evolving ecosystem, as organizations explore different approaches to agent design, tooling, and integration rather than converging on a single standard.

2 While Agents Fragment, MCP Is Consolidating

MCP servers show similarly rapid uptake, but with a different adoption pattern. Wiz Research observes MCP servers in 80% of cloud environments, underscoring how quickly model context exchange has consolidated around a common protocol. Within that population, 5% of environments have at least one internet-facing MCP server, highlighting early but tangible exposure risk as orchestration infrastructure moves into production.



MCP servers are in at least 80% of observed cloud environments. Of that group, 5% of have at least one internet-facing MCP server.

3 AI Components Are Accumulating Beyond Direct Deployment

The rise of transitive AI components is expanding the AI supply chain in cloud environments.

Just as open-source dependencies extended software supply chain complexity, embedded AI components introduce additional considerations around model provenance, update cadence, runtime configuration, and external connectivity.

Organizations must now consider:

- Which applications embed local models
- How those models are updated
- What permissions inference services operate under
- Whether those components are Internet-reachable
- How AI-specific vulnerabilities are surfaced and remediated

As AI becomes embedded deeper into applications and infrastructure, visibility into how models are introduced and operated becomes as important as the decision to deploy them directly. The security question shifts from "*What AI provider do we use?*" to "*What AI components are already operating inside our environment?*"

AI is not only adopted. It is accumulated. And accumulation without visibility becomes a governance challenge.

As AI adoption spreads across development, platform, and product teams, it does not converge into a single, centrally managed stack. AI coding assistants may be introduced by developers, agents by platform teams, and embedded AI features by third-party software vendors. Over time, this creates a distributed AI ecosystem composed of many small components rather than a few centralized platforms.

4 Autonomy Expands the Architectural Attack Surface

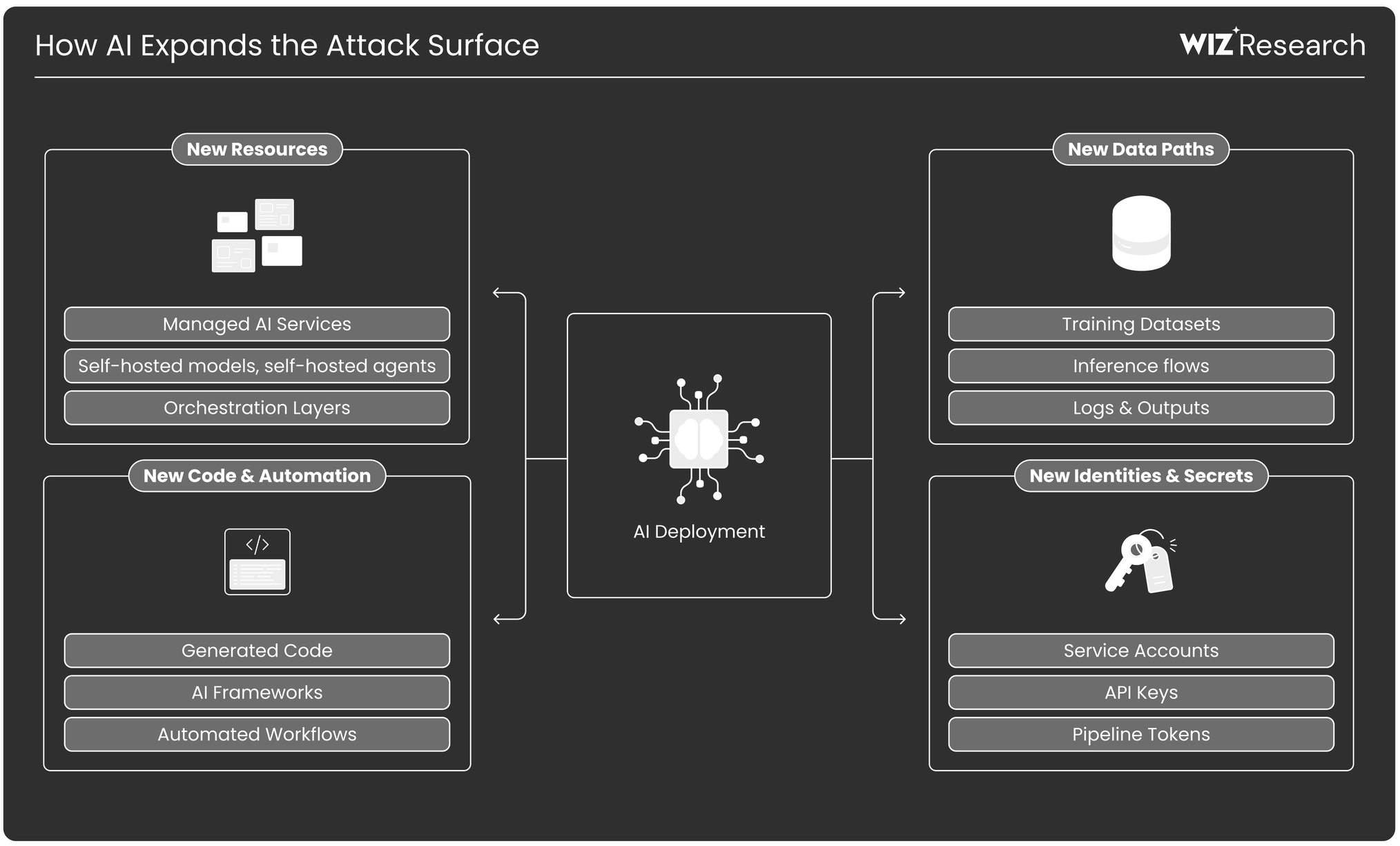
Together, agents and MCP servers expand the architectural footprint of AI within cloud environments. Rather than producing outputs in isolation, these components are designed to coordinate actions across tools and services. As orchestration infrastructure becomes more common, the number of AI-related systems that must be inventoried and secured increases.

Notably, while MCP has rapidly emerged as a shared standard for model context exchange, no equivalent consensus has yet formed around agent-to-agent (A2A) communication. Instead, a wide range of agentic frameworks coexist across environments. Established frameworks such as LangGraph, OpenAI Agents, and Google ADK show meaningful adoption, alongside numerous open-source and bespoke implementations. Protocol-level adoption tells a similar story: A2A appears at a fraction of MCP's prevalence.

This divergence matters. MCP demonstrates how quickly AI infrastructure can consolidate around a standard, while agent coordination highlights where fragmentation and uncertainty persist as organizations push toward more autonomous systems.

4. AI Broadens what is Reachable in Cloud Environments

As we explored in our [2026 Cloud Threats Retrospective,] AI adoption expands the cloud attack surface by embedding familiar risks into new execution layers. Model servers, orchestration layers, development tooling, and specialized APIs are now embedded throughout cloud environments, often operating alongside traditional services but without the same visibility or hardening expectations.



As AI systems become more tightly integrated with sensitive data and privileged automation, misconfigurations and excessive permissions carry greater impact. Existing attack techniques such as credential theft, misconfiguration abuse, and supply chain compromise remain effective against AI-native components, frequently allowing attackers to move laterally into high-value systems.

1 Exposed and Misconfigured AI Infrastructure

Misconfigured AI tooling has already been abused in the wild. Industry research has shown attackers exploiting unsecured AI interfaces to execute AI-generated payloads, demonstrating how exposed AI services can function like publicly accessible administrative consoles. Similar patterns have been observed in self-hosted AI tooling such as ComfyUI, where weak authentication and permissive defaults enable direct abuse.

Research from Censys identified thousands of publicly accessible Ollama instances, highlighting how quickly AI inference servers can become Internet-facing without proper controls. Public model servers effectively expose execution environments that can download models, process arbitrary prompts, and interact with local resources.

In addition to misconfiguration, vulnerabilities in AI infrastructure are also emerging. Wiz Research previously identified Probllama (CVE-2024-37032), a critical vulnerability in Ollama that could allow remote code execution under certain conditions. Workflow automation platforms such as n8n have also disclosed multiple critical security issues, demonstrating that orchestration layers connecting AI systems to external services introduce their own attack paths.

2 Overprivileged AI agents

Organizations employing agents may assign them broad privileges or access to sensitive data by necessity, to enable them to fulfil their required function. However, if they neglect to limit their access to the Internet, allow them to use untrusted external data as input, or fail to secure them against prompt injection attacks, they could accidentally create all the ingredients for the lethal trifecta of AI agents.

With agents rapidly taking on key roles within organizations' product backends, internal tooling, and CI/CD pipelines, the risk of an attacker hijacking an overprivileged agent must become a key consideration in security teams' risk management and security review processes. This is especially true in multi-tenant services, where insufficient isolation could enable an attacker to gain cross-tenant access to other customers' data.

3 Living-off-the-LLM and AI Tooling Abuse

AI tooling present inside environments can also expand the attack surface after initial access. In the s1ngularity supply chain attack, malicious packages injected into the Nx build system abused already-installed AI command-line tools, including Claude, Gemini, and Amazon Q, to support reconnaissance and credential harvesting. Rather than deploying new malware, attackers reused trusted developer utilities already embedded in workflows.

AI infrastructure therefore expands the attack surface in two ways. It introduces new exposed services and orchestration layers, and it provides attackers with powerful tooling already present inside compromised environments.

5. AI Is Changing Attacker Behavior and Economics

AI is enhancing attacker productivity rather than replacing established intrusion techniques. Across incidents and research, AI compresses timelines, reduces manual effort, and scales iteration speed. This shift is already visible across multiple layers of attacker activity.

AI-Powered Malware and Adaptive Execution:
Industry research shows attackers incorporating LLMs directly into malware workflows. Some samples dynamically generate commands at runtime, tailor execution logic to the target environment, or modify payload behavior on the fly. Research from Wiz and other vendors has documented malware leveraging AI services to adapt execution during runtime, reducing reliance on static payloads and complicating detection. These techniques accelerate known tradecraft rather than introducing entirely new attack classes.

Measured Agent Capability in Offensive Testing:
In controlled evaluations of AI agents performing realistic web hacking tasks, state-of-the-art models successfully solved the majority of targeted vulnerabilities at low cost when operating within clearly defined scopes. In broader, less constrained scenarios, performance declined and cost increased, highlighting the continued importance of prioritization and contextual reasoning. These results demonstrate both the growing capability of autonomous agents and their current limitations. The implication is economic: when vulnerability discovery and exploit construction can be automated within scoped tasks, iteration speed increases and experimentation cost declines.

Exploiting Repeatable Weaknesses in AI-Built Applications:
AI-assisted development changes attacker economics by introducing repeatable generation patterns. Wiz's analysis of Base44, revealed systemic design flaws that enabled unauthorized access to private applications, with weaknesses reproducible across environments due to shared generation logic.

Our research into Moltbook also demonstrated how AI-built applications could unintentionally expose sensitive data when generation defaults and guardrails were insufficient. The issue was not a single coding mistake, but a structural weakness that propagated through automated development patterns.

Abuse of AI-Enabled OAuth Integrations:
In a campaign analyzed by Google Threat Intelligence, attackers abused compromised OAuth tokens tied to AI-enabled integrations to access Salesforce environments and exfiltrate data. Rather than exploiting traditional software flaws, attackers leveraged trusted automation paths and existing integrations to move laterally. This reflects a shift toward abusing trust relationships embedded in AI-enabled workflows.

AI-Powered Vulnerability Research and Exploit Development:
Zeroday.cloud was a coordinated whitehat research effort aimed at uncovering previously unknown vulnerabilities in widely deployed cloud software. AI-assisted analysis contributed to the discovery of 13 zero-day vulnerabilities in foundational technologies, including core database systems. The event illustrates how AI can accelerate vulnerability discovery in high-impact software.

Benchmarks such as Cyber Model Arena demonstrate measurable improvements in exploit task performance generation over generation. While AI does not consistently outperform elite human practitioners, capability gains are meaningful and compounding.

AI does not need to replace expert attackers to alter the threat landscape. Even incremental improvements lower the skill barrier, reduce the cost of discovery, and increase the speed of exploit development, augmenting human capability rather than eliminating the need for expertise.

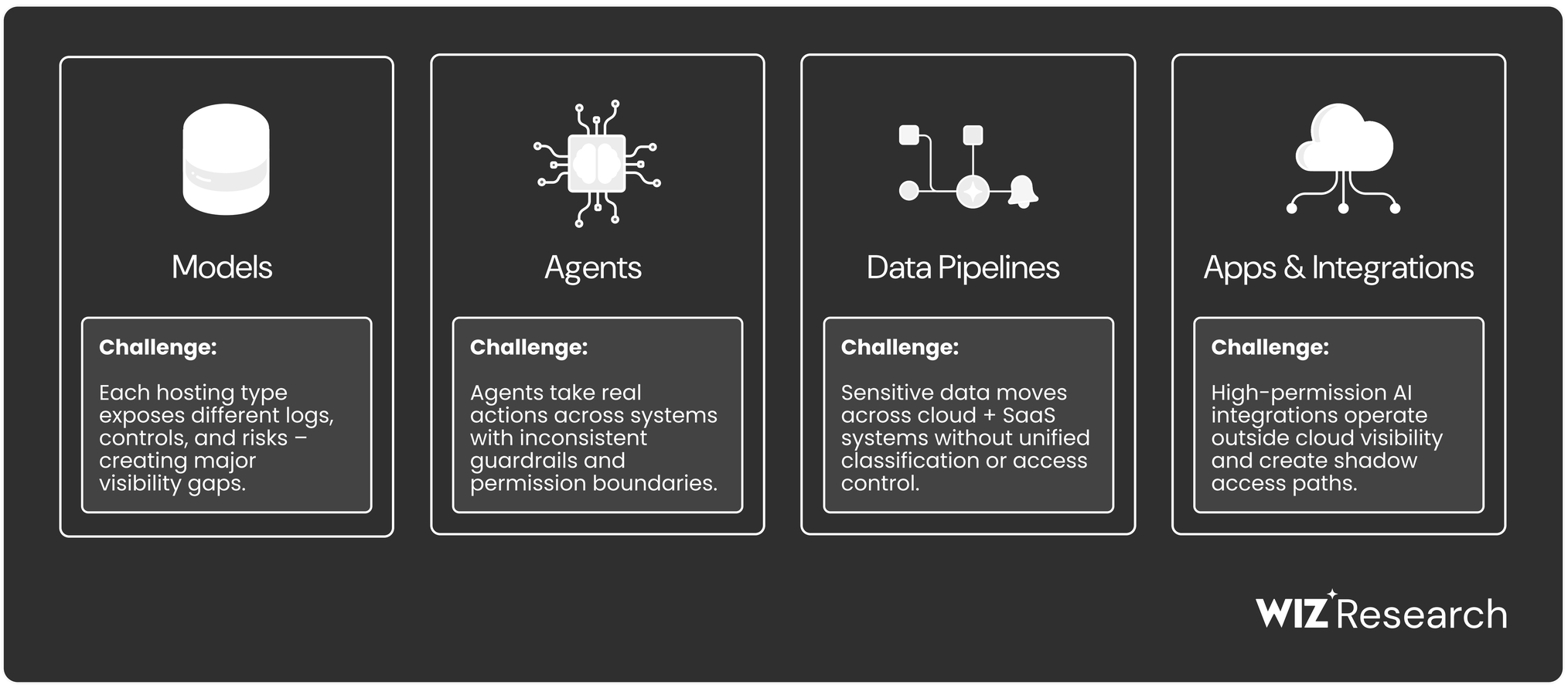
As vulnerability discovery and exploitation become cheaper and more automated, attackers can test more hypotheses, refine techniques more quickly, and scale intrusion attempts with less specialized knowledge. The result is greater volume, faster operational cycles, and sustained pressure on defensive teams.

6. Implications for Defenders in AI-Driven Cloud Environments

AI is already embedded across development workflows, orchestration layers, and production infrastructure. The challenge for security leaders is not predicting future breakthroughs, but managing how AI is already reshaping cloud environments today.

1 Defenders Must Inventory and Secure AI as Core Infrastructure

As agents, orchestration layers, and embedded AI components proliferate, the number of services that must be inventoried, hardened, and monitored increases. AI does not replace traditional cloud risk. It adds new execution layers and new interconnections that compound existing complexity.



Security programs must treat AI components as first-class infrastructure and maintain continuous visibility into where those systems operate and what they can access, subjecting them to the same asset inventory, configuration review, identity governance, and exposure management as any other cloud workload.

2 Organizations Must Extend Governance Across Distributed AI Ownership

AI adoption spans development, platform, product, and data teams. Coding assistants, agents, and embedded AI features may be introduced independently across the organization, often without centralized review.

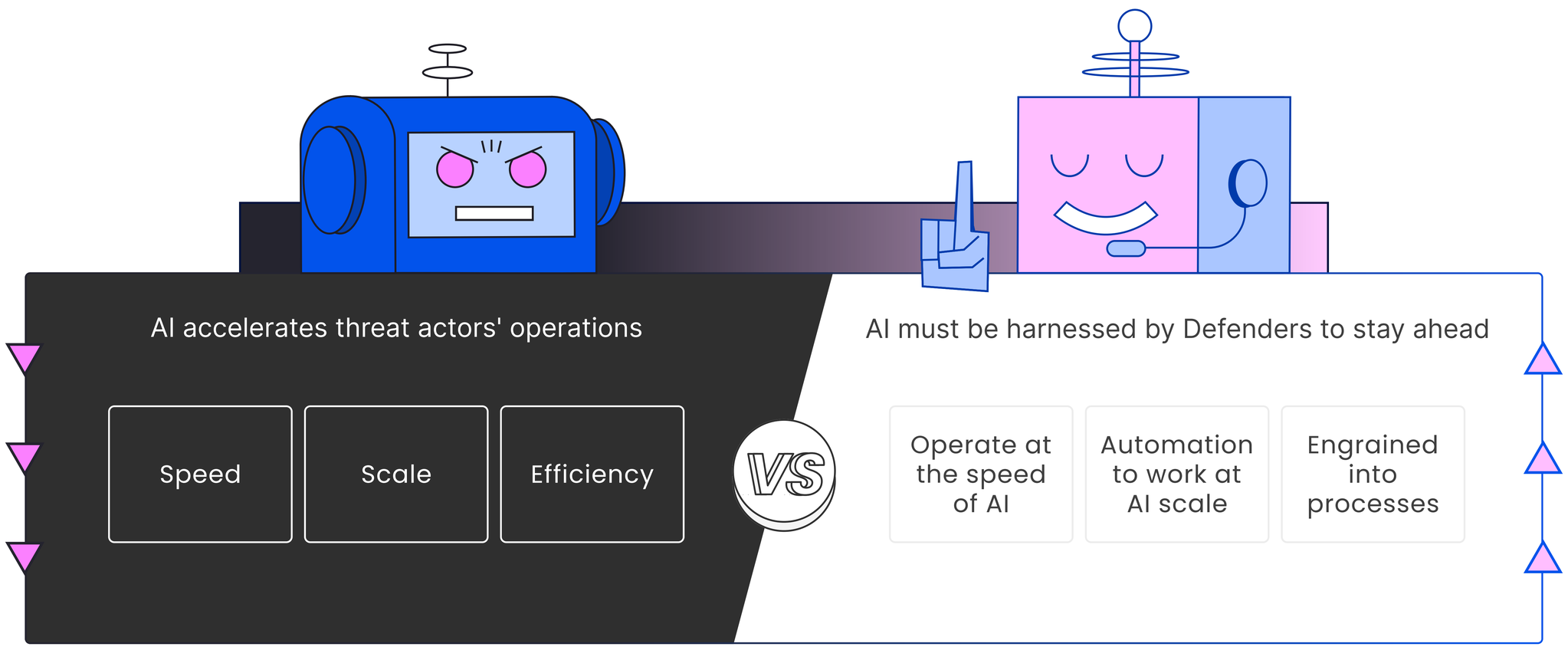
Governance models must evolve to account for distributed ownership, transitive components, and shared accountability. AI security cannot sit exclusively with a single innovation team. It must be integrated into cloud security, application security, and data governance programs.

3 Security Teams Must Adapt to Faster, More Automated Adversaries

Observed incidents and benchmark data suggest AI can reduce experimentation cost, accelerate exploit development, and increase the speed of post-compromise activity. This shift is reinforced by recent advances in frontier models like Anthropic's Claude Mythos, which has demonstrated the ability to autonomously identify vulnerabilities and generate working exploits, compressing the time between reconnaissance, discovery, and execution.

Defenders should assume faster reconnaissance, more automated lateral movement, and higher iteration velocity rather than entirely new categories of attack. As tasks like vulnerability discovery and exploit generation become more efficient, attackers can test more pathways in less time and refine their approach with minimal overhead.

Detection and response programs must adapt to increased volume and speed, focusing on patterns of abuse rather than isolated signals. This includes prioritizing behavioral detection, monitoring automated access patterns, and accounting for the growing role of AI-assisted tooling in both pre- and post-compromise activity.



4 Defenders Must Use Context to Cut Through AI-Driven Complexity

As AI systems proliferate, alert volume and environmental complexity increase. In this environment, raw signal is not enough. Security teams must understand how AI assets connect to identities, permissions, network exposure, data sensitivity, and runtime behavior.

Without this contextual mapping, AI-related findings become additional noise layered onto already crowded dashboards. With it, teams can distinguish between theoretical misconfigurations and material exposure paths.

AI can also assist defenders when applied thoughtfully. Used responsibly, AI-driven analysis can help prioritize alerts, surface high-risk misconfigurations, and reduce time spent triaging low-impact findings.

The competitive advantage will not come from identifying more issues, but from understanding which issues materially increase risk based on real access, real data, and real attack paths.

Questions Security Leaders Should Be Able to Answer

1. Where is AI operating across our cloud environments, including embedded and transitive components?
2. Which AI-related systems are Internet-reachable or connected to sensitive data and automation workflows?
3. What permissions do agents, model servers, and AI services operate under, and are those privileges justified?
4. How is AI ownership distributed across development, platform, and product teams?
5. Can we rapidly detect and respond to threats across AI applications?
6. Can we prioritize AI-related findings based on real business impact rather than alert volume alone?

Conclusion

AI has moved decisively from experimentation to infrastructure. Across cloud environments, it is now a foundational component of how applications are built, deployed, and operated. While the pace of first-time adoption has stabilized, the depth of AI integration continues to increase, introducing new layers of automation, dependency, and risk.

This evolution has not created entirely new classes of security issues as much as it has amplified familiar ones. Misconfigurations, excessive permissions, supply chain weaknesses, and credential misuse now appear in more places and closer to sensitive data and critical workflows. At the same time, attackers are adapting quickly. Advances in AI systems, including models like Claude Mythos that have demonstrated the ability to autonomously identify vulnerabilities and generate working exploits, point to a near-term reality where reconnaissance, discovery, and exploitation cycles are significantly compressed. Attackers are already using AI to accelerate reconnaissance, automate execution, and blend malicious activity into legitimate usage, while also setting their sights on the novel attack surface of organizations' AI infrastructure and tooling.

The defining challenge for defenders is not anticipating the next AI innovation, but maintaining visibility and control over how AI is used today. Security models built for earlier generations of cloud workloads struggle to keep pace with agentic behavior, model-driven workflows, and fragmented AI adoption across development and production. As vulnerability discovery and exploitation become faster and more automated, defenders must contend with higher volumes of activity, shorter feedback loops, and less time to respond.

Organizations that succeed will be those that treat AI security as an extension of cloud security, not a separate discipline. Understanding where AI runs, how it connects to data, identities, and automation, and how those connections can be abused is now essential to managing cloud risk as AI continues to reshape the cloud operating model.

Methodology

This report is based on analysis by Wiz Research across hundreds of thousands of real-world cloud environments throughout 2025, spanning major cloud providers and a wide range of industries.

Findings combine anonymized cloud configuration metadata, AI asset discovery, and hands-on security investigations, supplemented by publicly disclosed incidents and third-party threat intelligence.

AI adoption and exposure were assessed by inventorying AI-related resources such as managed AI services, self-hosted models, AI development tooling (including coding assistants and agents), and orchestration components like Model Context Protocol (MCP) servers. Adoption figures reflect infrastructure and configuration signals observed across cloud environments in the dataset and should be interpreted as lower-bound estimates, not as a measure of global organizational adoption.

Security insights were derived with a focus on misconfigurations, exposure paths, and real-world attacker abuse of AI-related infrastructure observed in production cloud environments.