



The State of AI and API Security:

Navigating the Agentic Era

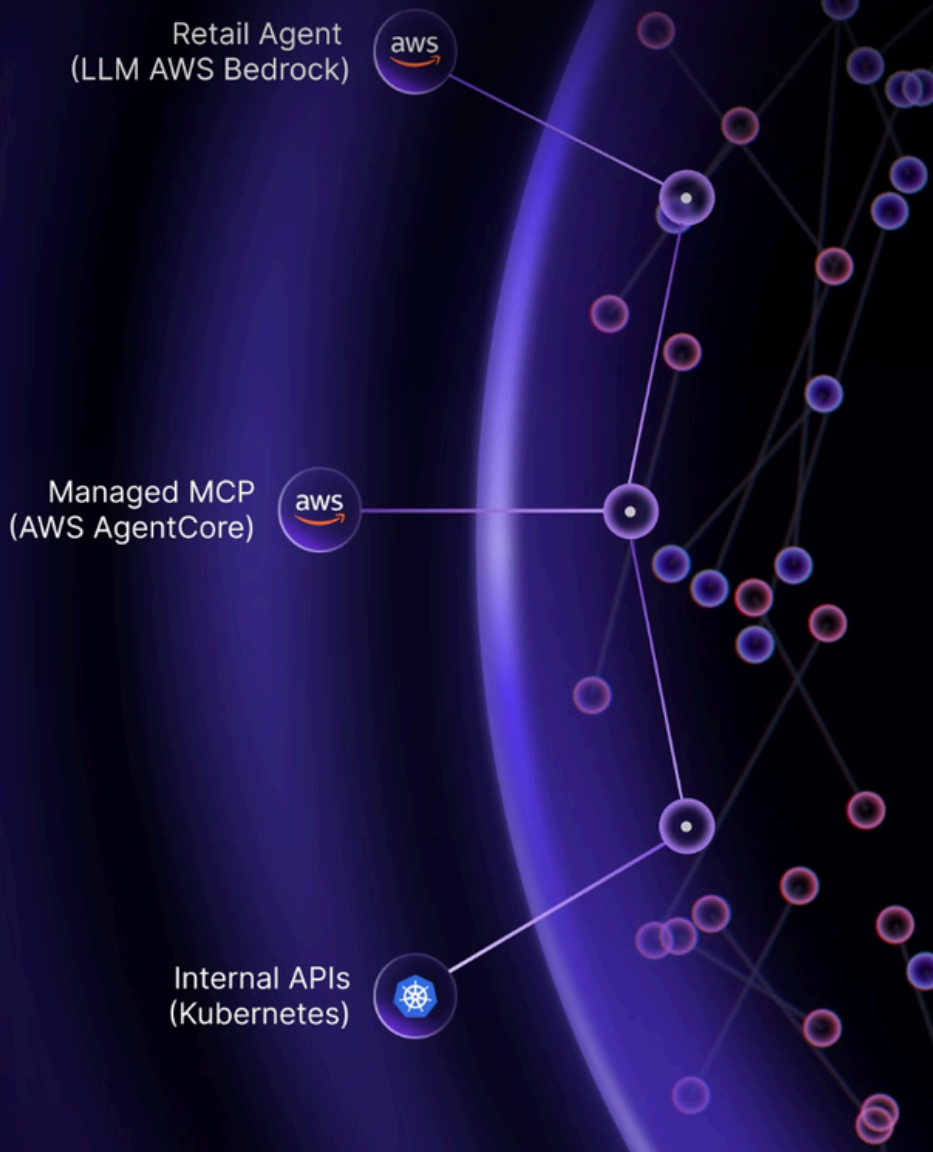


Table of Contents

Executive Summary	2
Key Findings	5
Drivers for AI Adoption	8
API Development Trends	9
API Security Challenges in the Agentic Era.....	10
Salt Labs Analysis of Customer Data	12
Monitoring and Securing APIs.....	14
Generative AI and API Security Risks	17
Measuring ROI in API Security.....	20
Conclusion and Recommendations.....	21
About Salt	23

Executive Summary I

The State of API Security in 1H 2026

The Emerging Agentic Security Gap

Security leaders increasingly recognize that AI systems introduce entirely new operational risks. Our survey shows that boards and executive leadership are now actively scrutinizing AI security:

- **79% report increased executive scrutiny of AI security risks.**
- **69% of boards are concerned about sensitive data leakage through AI prompts or models.**
- **39% are specifically worried about autonomous agents acting without human oversight.**

Despite this boardroom mandate, organizations admit to a severe "Confidence Gap" when relying on traditional security stacks to address AI-driven threats. The visibility gap mentioned earlier is compounded by a profound failure in legacy tooling:

- **About 18% are very confident they can detect attacks using Generative AI.**
- **Crucially, only 24% of respondents find their existing security tools "Very effective" at preventing attacks.**

These findings confirm that the entire legacy security stack is failing. Traditional Application Security Testing (SAST and DAST) lacks the context to validate AI-generated code, while legacy Web Application Firewalls (WAFs) and API Gateways are completely blind to new agent-based infrastructure such as MCP servers. AI security has become inseparable from API security. To safely scale, organizations must build an **Agentic Security Graph** to visualize undocumented machine-to-machine connections, and enforce context-aware security from the developer environment straight through to runtime. Organizations must secure the "Sequence of Intent" rather than rely solely on the static signatures and rate limits of outdated WAFs and Gateways, as APIs evolve into the control plane for autonomous systems.

Executive Summary II

The era of human-centric API consumption is ending. As enterprises rapidly deploy generative AI models and autonomous agents, APIs have evolved from simple integration points into the operational backbone of the AI economy. Every autonomous action, from reasoning to retrieving sensitive data to executing workflows, ultimately occurs through an API call.

This shift means a fundamental truth now defines modern security architecture: You cannot secure AI without securing APIs.

Our 1H 2026 survey of over 300 security leaders reveals that organizations are accelerating API adoption to support AI innovation, but security maturity is struggling to keep pace. Rather than traditional API vulnerabilities, the most alarming findings highlight a growing tension between agentic innovation and enterprise visibility:

- **60% of organizations admit to a profound lack of control over the security of the AI models that generate their application code.**
- **49% are essentially blind to non-human, machine-to-machine traffic, unable to monitor what their autonomous agents are doing.**
- **48% cannot effectively differentiate legitimate AI agents from malicious bots.**
- **As a result of this opacity, 47% have delayed production releases due to concerns about securing the APIs exposed to these autonomous systems.**

As organizations build AI-driven platforms, APIs are expanding rapidly. **Nearly half (47%) of respondents reported API growth of 51 to 100% in the past year**, reflecting the massive expansion of machine-to-machine communication. At the same time, Generative AI is becoming central to software development: **69% of organizations already use GenAI to develop APIs, and 19% plan to adopt it in the next 12 months.**

This transformation introduces a new category of security risk: the **Agentic Action Layer**, the APIs that allow AI agents to interact with enterprise systems.

Executive Summary III

The Path Forward

Organizations that successfully navigate this shift are treating API security as the foundation for AI governance. Leading teams are prioritizing:

- **Mapping the Agentic Security Graph:** Move beyond static API lists to visualize the continuous, multi-pronged relationships across the entire agentic stack.
- **Identity-Aware Intent Analysis:** Monitor runtime machine-to-machine traffic by focusing on the agentic "Sequence of Intent" rather than relying on legacy static signatures.
- **Context-Driven Assurance in the SDLC:** Establish guardrails for AI-generated code directly within the developer environment (IDE) and create a continuous runtime-to-code feedback loop to train local models to prevent the recurrence of architectural errors.
- **Posture governance aligned to modern frameworks:** Mapping risks directly to the OWASP API Top 10 and NIST AI RMF.

When implemented effectively, API security becomes more than a defensive control. It becomes the enabling layer that allows organizations to deploy AI systems safely, confidently, and at scale.

5 Key Findings

APIs Are the Operational Backbone of the Agentic AI Economy

APIs remain the core infrastructure enabling digital transformation and automation. The top drivers for API adoption include:

- **57% – Development efficiency and standardization**
- **56% – Digital transformation initiatives**
- **50% – Platform and system integrations**
- **43% – Cloud migration**

AI is now emerging as a major new driver:

- **32% use APIs to enable AI/ML analytics and automation**
- **14% rely on APIs to support AI agents and autonomous systems**

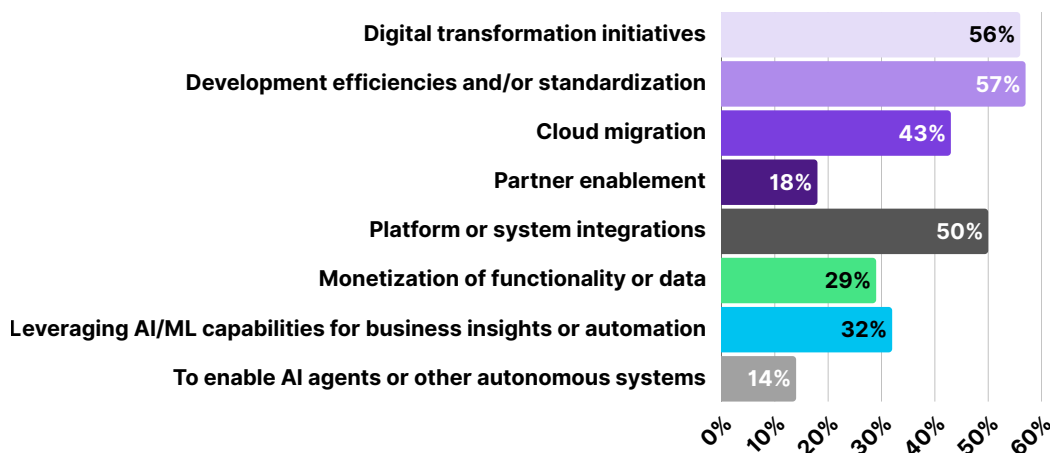
This indicates that organizations are rapidly building the infrastructure needed for **AI-driven enterprise operations**.

API Growth Is Accelerating Rapidly

Organizations reported dramatic increases in API volume:

- **47% experienced API growth of 51–100% in the past year**
- **12% saw growth of 101–200%**
- **30% reported growth up to 50%**

This surge is being driven by microservices architectures, automation initiatives, and the rise of AI-enabled applications. However, rapid growth is outpacing many organizations' ability to maintain visibility and governance.



What are the main drivers behind the use of APIs in your organization?

6 Key Findings II

API Visibility Remains a Critical Gap

Accurate inventory and monitoring remain major challenges:

- **57% are only somewhat confident their API inventory is accurate**
- **23% have fully automated, real-time inventory**
- **19% have low or no confidence in their inventory**

This visibility gap increases the risk of **shadow APIs, undocumented endpoints, and unintended exposure of sensitive data.**

API Security Incidents Are Common

API attacks are not theoretical risks. Our survey found:

- **32% of organizations experienced an API security incident in the past year**
- **47% delayed application deployment due to API security concerns**

Common security issues discovered in production APIs include:

- **Sensitive data exposure (44%)**
- **Vulnerabilities (43%)**
- **Authentication problems (41%)**
- **Credential stuffing or brute force attacks (27%)**

These incidents demonstrate that API security directly impacts **business continuity and innovation velocity.**

Key Findings III

Generative AI Is Transforming Development (and Risk)

Generative AI is already reshaping software development:

- **18% use GenAI for all API development**
- **51% use it for some development**
- **19% plan to adopt it within 12 months**

However, security teams remain concerned about the risks. Top concerns include:

- **60% — Lack of control over the security of AI models used in development**
- **59% — Difficulty securing AI-generated code**
- **57% — Potential for new vulnerabilities introduced by GenAI**

These concerns highlight the need for **secure development guardrails for AI-generated software**.

AI Agents Introduce a New Security Challenge

Organizations deploying AI agents face significant operational risks. Key challenges include:

- **50% – Maintaining reliability for mission-critical automation**
- **49% – Monitoring machine-to-machine traffic**
- **48% – Distinguishing legitimate agents from malicious bots**
- **47% – Securing APIs exposed to autonomous systems**

These findings reinforce the concept of the **Agentic Action Layer**: the APIs that enable AI agents to interact with enterprise systems.

Drivers for API Adoption

Building the Infrastructure for the AI-Driven Enterprise

APIs have long served as the connective tissue of digital transformation, enabling organizations to integrate applications, deliver services, and scale innovation. In 2026, however, APIs are taking on a new role: **powering the operational layer of AI systems**. Modern AI deployments—from generative AI copilots to autonomous agents—depend on APIs to retrieve data, execute workflows, and interact with enterprise infrastructure. As a result, organizations are expanding their API ecosystems not only to support digital services but also to enable **machine-to-machine automation at scale**. Survey responses highlight that traditional drivers of API adoption remain strong while AI-related use cases are rapidly emerging.

Primary Drivers of API Adoption

Security leaders identified the following as the most important drivers behind API usage:

- **57% – Development efficiency and standardization**
- **56% – Digital transformation initiatives**
- **50% – Platform and system integrations**
- **43% – Cloud migration**
- **29% – Monetization of functionality or data**
- **18% – Partner enablement**

At the same time, AI-driven use cases are increasingly shaping API strategy:

- **32% report using APIs to enable AI/ML capabilities** such as analytics and automation.
- **14% say APIs are critical for enabling AI agents or other autonomous systems.**

These findings reinforce a fundamental shift in enterprise architecture: **APIs are no longer just integration tools... they are becoming the execution layer for AI systems.**

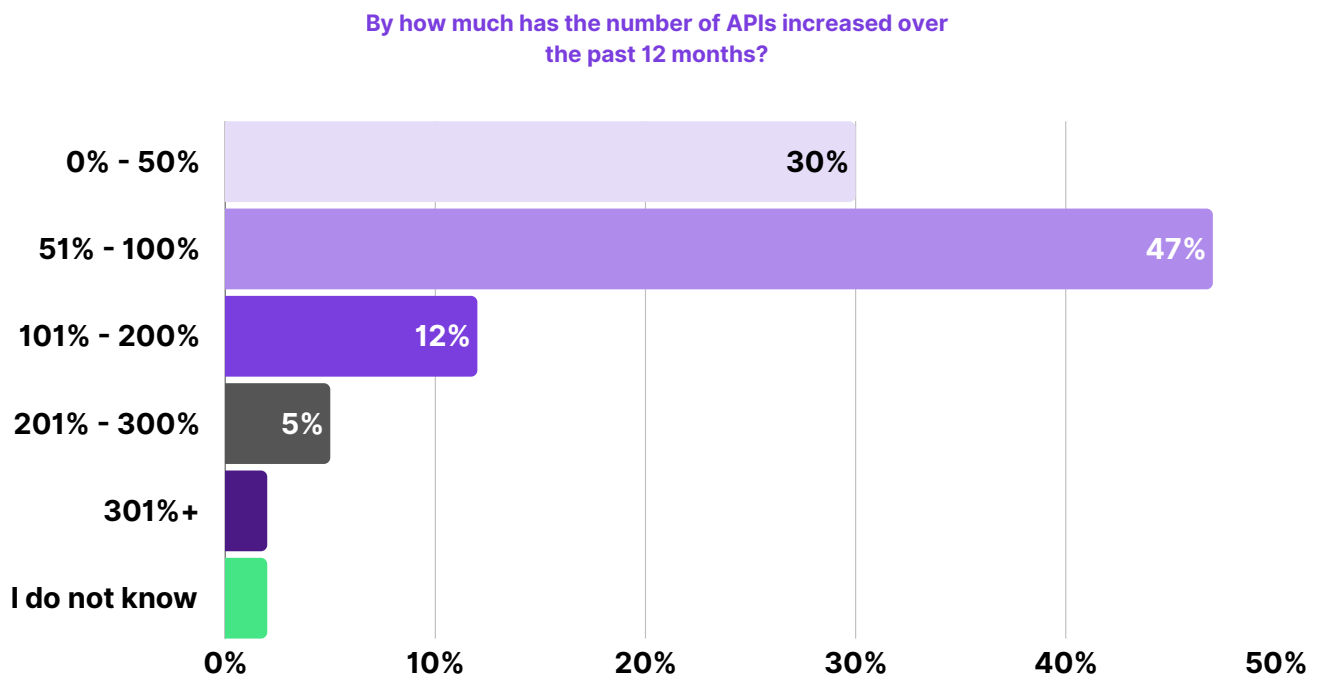
API Development Trends

The Engine of the Agentic Era

The reliance on APIs continues to grow, reflecting their role as the backbone of digital operations and AI innovation. As noted in our key findings, with **nearly half of organizations reporting API growth of 51 to 100%**, this rapid proliferation is no longer just human-driven.

It is fueled by machine-to-machine communication, the adoption of autonomous AI, and the need to enable new digital services at scale. However, it also raises the stakes for security and governance. As APIs multiply across hybrid environments, development velocity is rapidly outpacing security maturity.

APIs have become the central control plane for advanced technologies such as GenAI and autonomous agents. This means the risks tied to rapid API expansion now extend beyond technical vulnerabilities, directly shaping business resilience, regulatory compliance, and customer trust in the Agentic Era.



API Security Challenges in the Agentic Era

API security challenges remain a critical concern for nearly all organizations, with **32% reporting they experienced at least one API security incident in the past 12 months**. As API ecosystems expand to support AI, the attack surface grows exponentially, amplifying the risks of misconfigurations, poor visibility, and insufficient runtime protections.

Types of Security Problems Found

As organizations enable AI agents, common API security flaws become systemic risks. The speed and scale of automation amplify the potential damage:

- **Vulnerabilities (43%):** In an agentic architecture, a vulnerability does not just leak data; it can be weaponized to hijack an AI's logic. Through techniques like indirect prompt injection, attackers can manipulate an LLM to execute malicious API calls on their behalf, turning internal AI tools into insider threats.
- **Sensitive data exposure (44%):** This risk is magnified when APIs grant access to AI models. A single misconfiguration can allow a compromised agent to autonomously exfiltrate entire datasets at machine speed.
- **Authentication problems (41%):** In an automated ecosystem, authentication is no longer just about user logins. A fundamental flaw in machine identity can grant an autonomous AI agent dangerous levels of access, bypassing traditional controls.
- **Account misuse or fraud (20%):** This threat escalates significantly when AI agents are involved, enabling fraudulent transactions or data manipulation at scales and speeds that human attackers cannot replicate.
- **Denial-of-service (14%):** APIs powering critical AI workflows are high-value targets. A successful DoS attack halts automated business processes, resulting in severe operational disruption.
- **Brute forcing or credential stuffing (27%):** Attackers use automated techniques to target the machine identities of AI agents and services, seeking to compromise the core of the enterprise automation fabric.
- **Enumeration and scraping (10%):** Malicious AI can conduct reconnaissance at an unprecedented scale, mapping an organization's entire API attack surface to identify the weakest point of entry.

API Security Challenges in the Agentic Era II

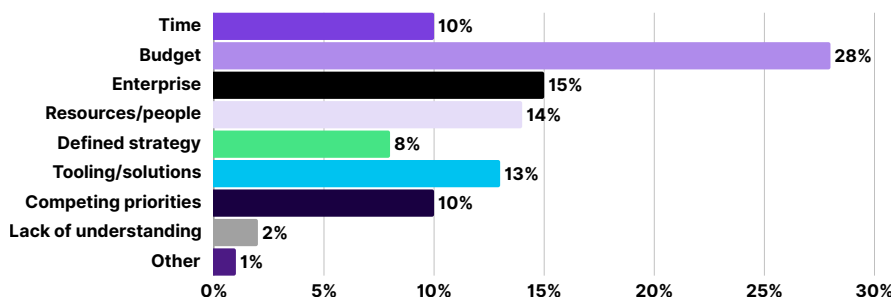
The Machine-Speed Code Generation Challenge

The rapid adoption of AI coding assistants is driving massive productivity, but it is also creating a critical governance gap. According to our survey, **60% of security leaders cite a lack of control over the security of AI models used in development, and 59% struggle to secure AI-generated code.**

Developers and AI agents are spinning up undocumented APIs, MCP servers, and third-party tool integrations at machine speed. Traditional Application Security Testing (SAST and DAST) fails in this environment because these tools lack business-logic awareness to validate complex API architectures and are completely blind to MCP configurations. This creates an unsustainable, reactive security posture where vulnerable logic is generated in seconds, but takes security teams weeks to discover in staging or production.

Biggest Concerns & Obstacles to an Optimal Strategy

Beyond specific incidents, organizations highlighted structural concerns in managing their API programs.



What is the biggest obstacle keeping you from implementing an optimal API security strategy?

Together, these findings reveal a fundamental tension. While APIs are essential to scaling AI, many organizations lack the purpose-built tooling and governance required to secure them effectively. Relying on legacy platforms exacerbates staffing shortages, as security teams are quickly overwhelmed by the operational friction and noise generated by trying to secure machine-speed traffic with human-speed tools.

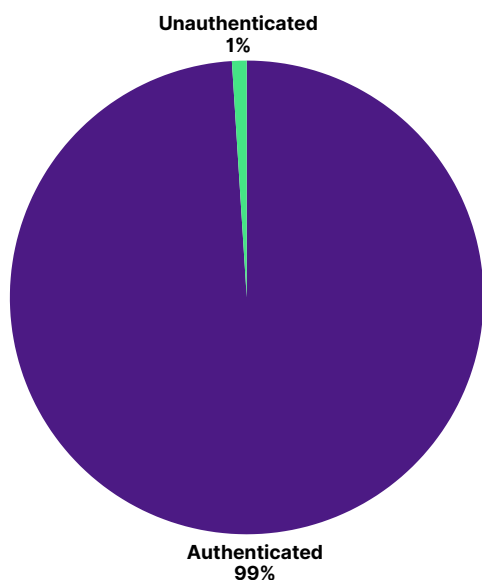
Salt Labs Analysis of Customer Data

Insight from the Front Lines: The New Attacker Playbook

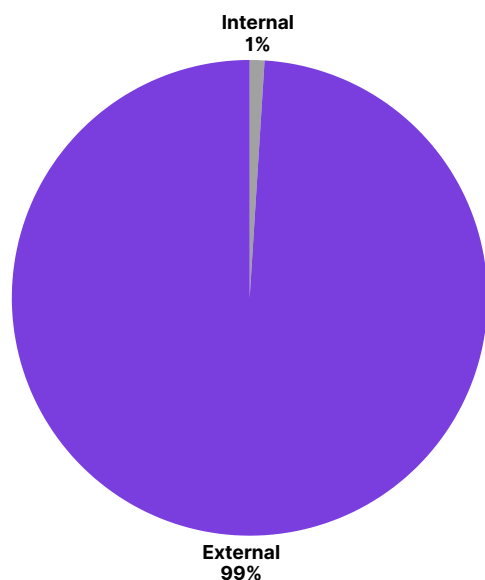
Our analysis of real-world attack data reveals the playbook attackers will use in the AI Agent Economy, and it confirms that perimeter security is becoming irrelevant. The primary threat is no longer an unauthenticated outsider trying to break in, but an authenticated entity. Today, this frequently means a compromised **Machine Identity**, such as a rogue AI agent, an over-permissioned LLM, or a hijacked automated workflow abusing its legitimate access. This represents a paradigm shift that legacy security tools, which rely on human-centric session monitoring, are not equipped to handle. The data shows that a staggering **99% of attack attempts originate from authenticated sources, while 99% target external-facing APIs**.

This proves that traditional security models focused on authentication are insufficient. Organizations must evolve to a model based on continuous monitoring, behavioral anomaly detection, and robust authorization checks to mitigate these modern risks.

Attack attempts from authenticated vs. unauthenticated attackers



Attack attempts against internal and external facing API endpoints



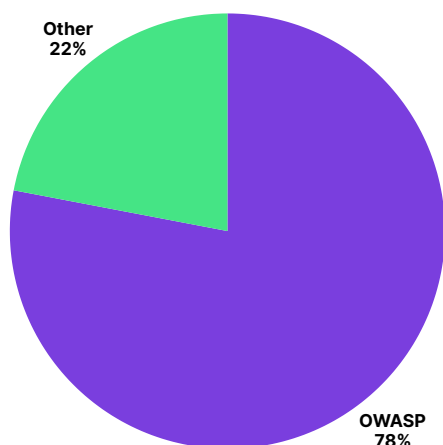
Salt Labs Analysis of Customer Data II

When examining attack techniques, we found that **78% of attempts align with the OWASP API Security Top Ten**, confirming that attackers are exploiting known, common vulnerabilities. The two most dominant vectors are governance and authorization failures:

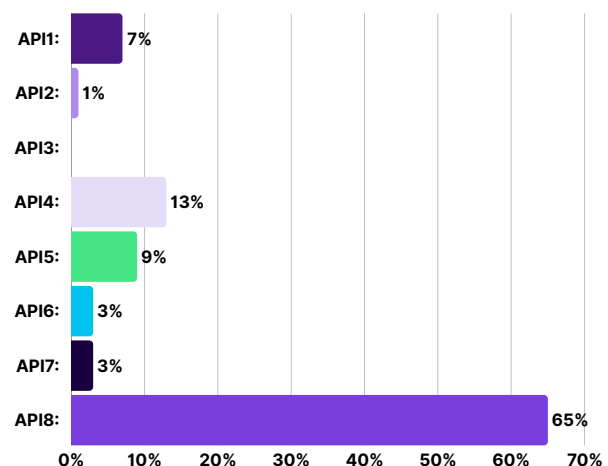
- The Dominant Threat - **API8 (Security Misconfiguration)**: Accounting for **65% of attacks**, these misconfigurations are highly dangerous in an AI context. An over-permissioned API connected to an LLM bypasses traditional RBAC, allowing an agent to query data the end-user shouldn't see.
- The Authorization Gap - **API1 (Broken Object Level Authorization)**: Contributing to **7% of attacks**, BOLA takes on new dimensions with machine-to-machine traffic. Because AI agents can chain multiple APIs together to complete tasks, a single BOLA vulnerability deep in the execution chain can be exploited autonomously at a massive scale.
- Less Common Exploits: In contrast, other issues like **API2 (Broken User Authentication)** makes up 1% and **API7 (Security Monitoring & Logging Failures)** account for just 3% of attempts, indicating that attackers are finding far more success exploiting misconfigurations and authorization flaws.

Overall, the Salt Labs analysis reinforces the urgent need for API-specific security strategies that align with the OWASP Top Ten. Organizations must go beyond traditional perimeter defenses and focus on strong authentication and authorization, proper configuration management, continuous security testing, and posture governance to address the evolving API threat landscape.

Attack attempts leveraging the OWASP API Security Top 10 vs other attack types



Attack attempts that map to the OWASP API Security Top 10



Monitoring and Securing APIs

Visibility Gaps Create Operational Risk

Despite the central role APIs now play in enterprise operations, many organizations still struggle to maintain complete visibility and effective monitoring of their API environments.

Accurate inventory and runtime monitoring are foundational capabilities for any API security program. Yet the survey reveals that these capabilities remain inconsistent across organizations.

API Inventory Confidence

Confidence in API inventories remains limited: **24% report having a fully automated, real-time API inventory. 57% say they are only somewhat confident their inventory is accurate.** Nearly **19% report low or no confidence in their ability to track all APIs.**

In the Agentic Era, this visibility gap is a critical liability. It increases the risk of "Shadow AI", where autonomous agents dynamically create undocumented endpoints or leverage protocols like Model Context Protocol (MCP) to connect to enterprise data outside the view of security teams.

Solving this requires moving beyond flat, legacy API inventories to build a dynamic **Agentic Security Graph**. By continuously mapping the multi-pronged relationships among Large Language Models (LLMs), MCP servers, and foundational APIs, organizations can eliminate blind spots and visualize exactly how autonomous systems interact with their proprietary data.

In the Agentic Era, this visibility gap is a critical liability. It **increases the risk of "Shadow AI"**, where autonomous agents dynamically create undocumented endpoints or leverage protocols like Model Context Protocol (MCP) to connect to enterprise data outside the view of security teams, leading to the unintended exposure of sensitive data.

Monitoring and Securing APIs II

Monitoring Practices

Monitoring practices also vary widely across organizations.

While **26% of organizations monitor APIs continuously in real time**, the majority rely on periodic monitoring processes such as daily scans, weekly reviews, or monthly assessments. These inconsistent monitoring practices create potential blind spots, allowing attackers to exploit vulnerabilities before security teams detect abnormal behavior.

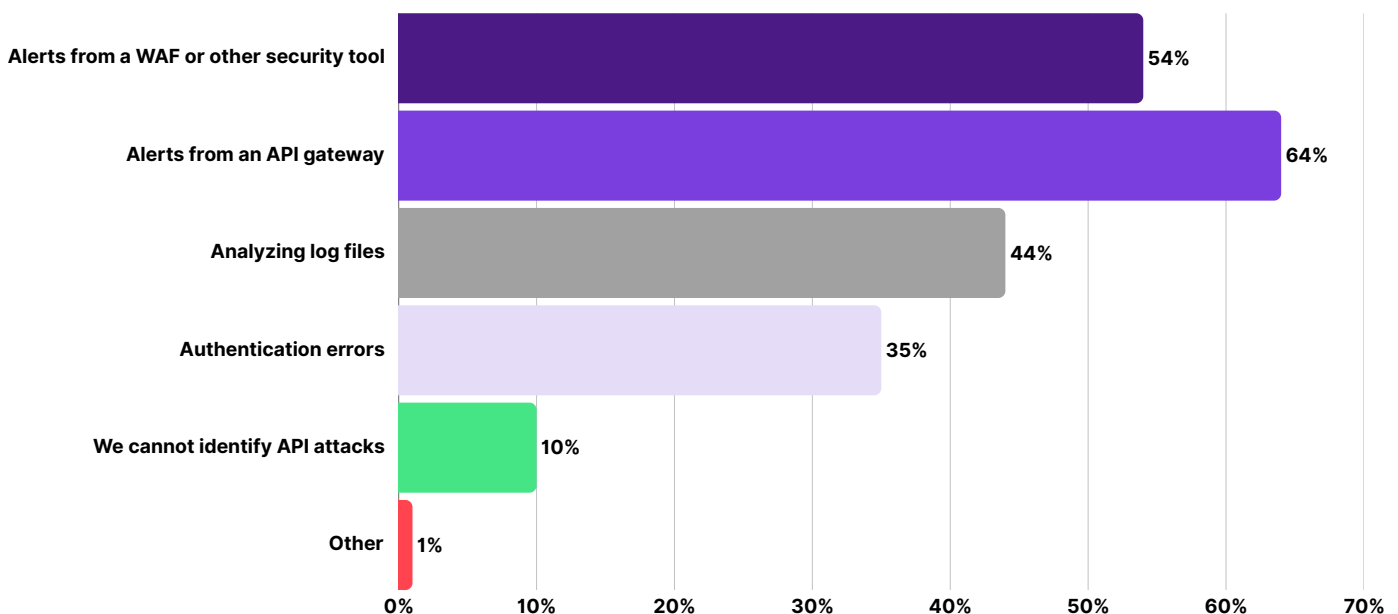
Detection Mechanisms

Organizations rely on a variety of tools to identify API attacks, including:

- **64% – Alerts from API gateways**
- **54% – Alerts from WAFs or other security tools**
- **44% – Log analysis**
- **35% – Authentication error monitoring**

However, **10% of respondents admitted they cannot reliably identify API attacks**, indicating that visibility challenges remain significant.

How do you identify an attack or attacker targeting your APIs? (Pick all that apply)



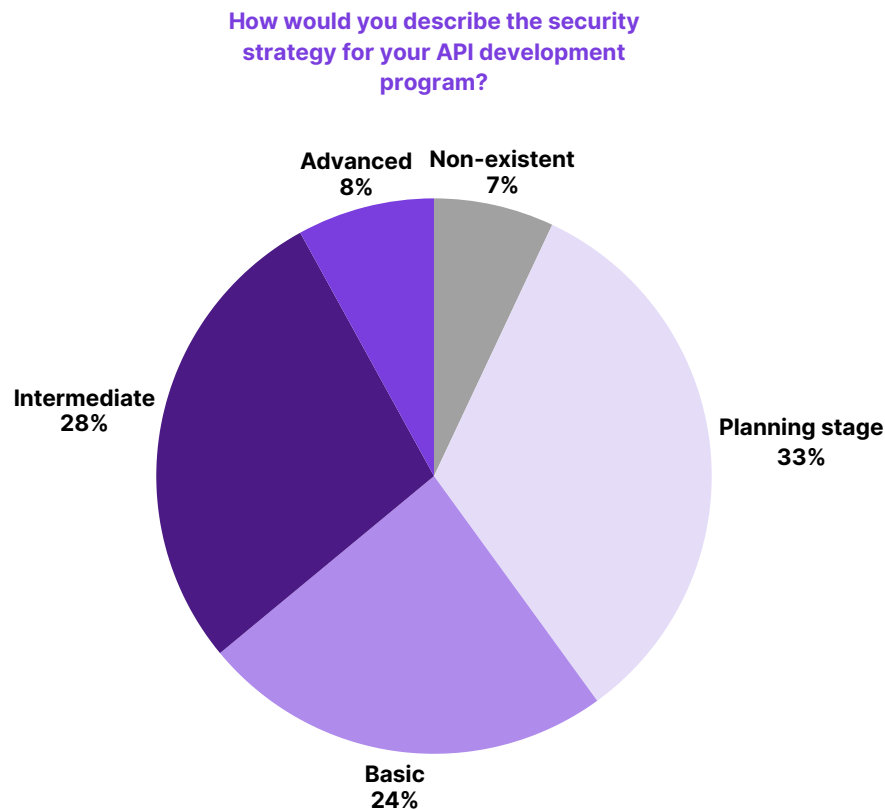
Monitoring and Securing APIs III

Security Program Maturity

API security programs remain uneven across the industry:

- **7% report having no API security strategy.**
- **33% are still in the planning stage.**
- **24% have basic programs focused on manual reviews and risk assessments.**
- **28% have intermediate maturity with testing and gateway protections.**
- **Only 8% report advanced API security programs.**

These findings suggest that while awareness of API risks has increased significantly, many organizations have not yet operationalized mature API security programs.



Generative AI and API Security Risks

The New Attack Surface of the Agentic Enterprise

Generative AI is rapidly transforming software development and operational workflows. Organizations are increasingly using AI to generate code, automate processes, and assist developers throughout the software lifecycle.

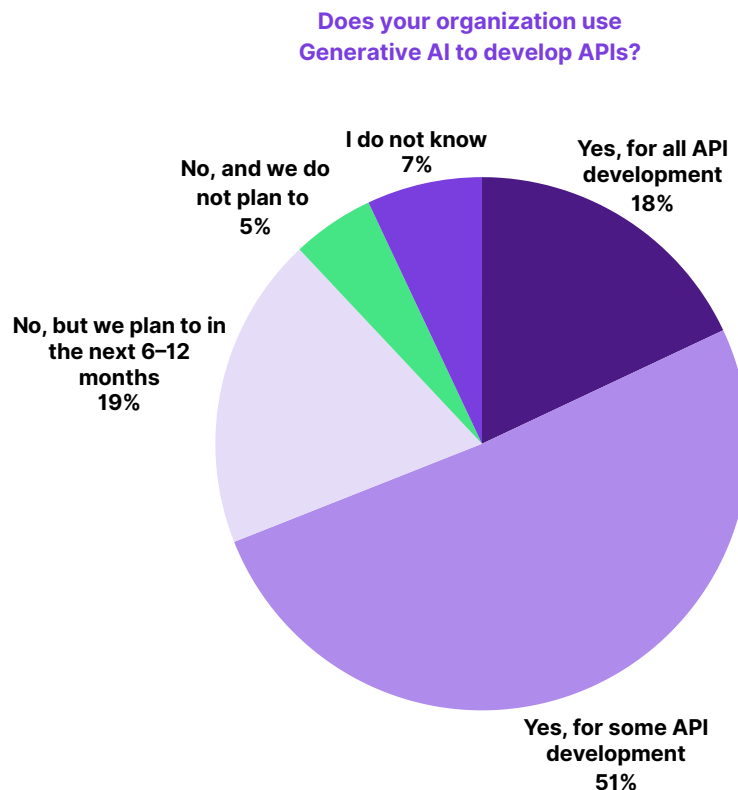
However, the widespread adoption of GenAI also introduces new security risks that organizations must address.

GenAI Adoption in API Development

The survey shows that GenAI has already become deeply integrated into development workflows:

- **18% of organizations use GenAI for all API development.**
- **51% use it for some development.**
- **19% plan to adopt GenAI for development within the next 12 months.**

This means that nearly **90% of organizations are already using or planning to use AI-assisted development tools.**



Generative AI and API Security Risks II

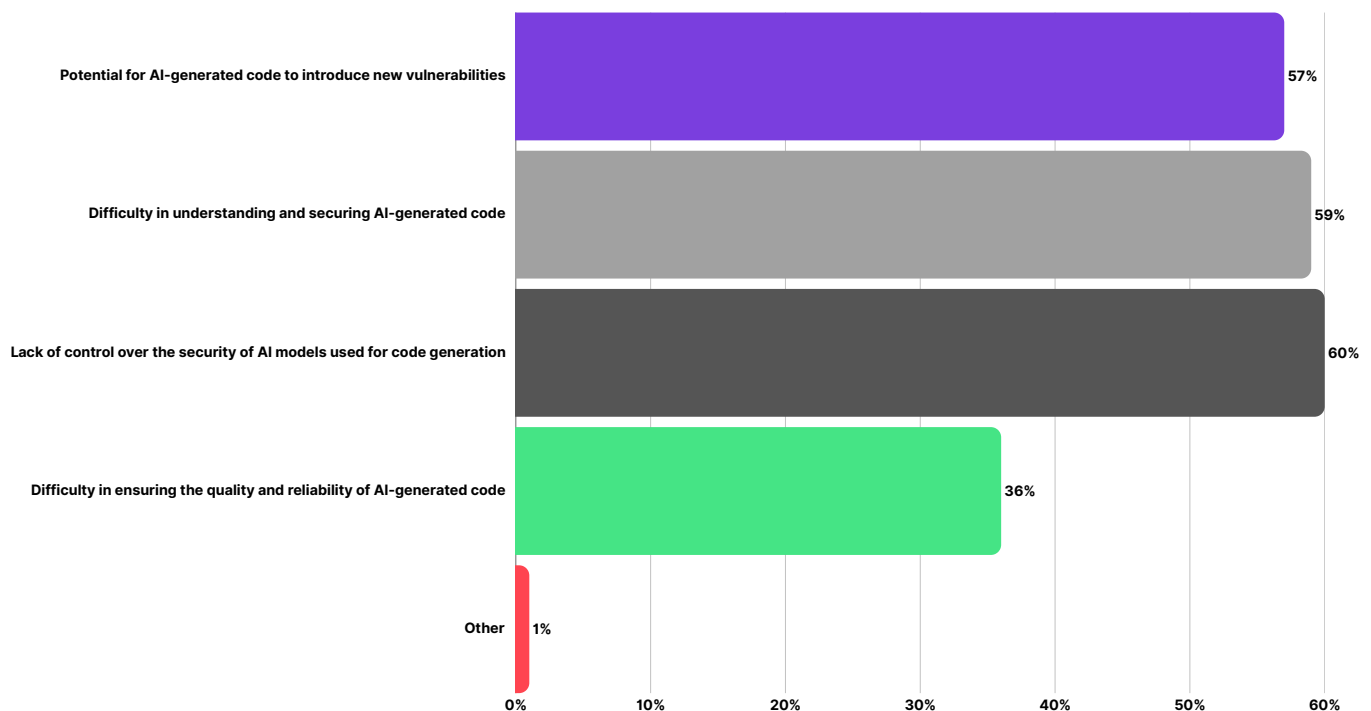
Key Security Concerns

Security leaders identified several risks associated with AI-generated code:

- 60% are concerned about lack of control over the security of AI models used in development.
- 59% worry about the difficulty of understanding and securing AI-generated code.
- 57% fear that AI-generated code could introduce new vulnerabilities.
- 36% cite concerns about reliability and code quality.

These concerns reflect the challenge of integrating AI tools into development pipelines while maintaining secure coding standards.

What security concerns do you have about using Generative AI to develop APIs?



Generative AI and API Security Risks III

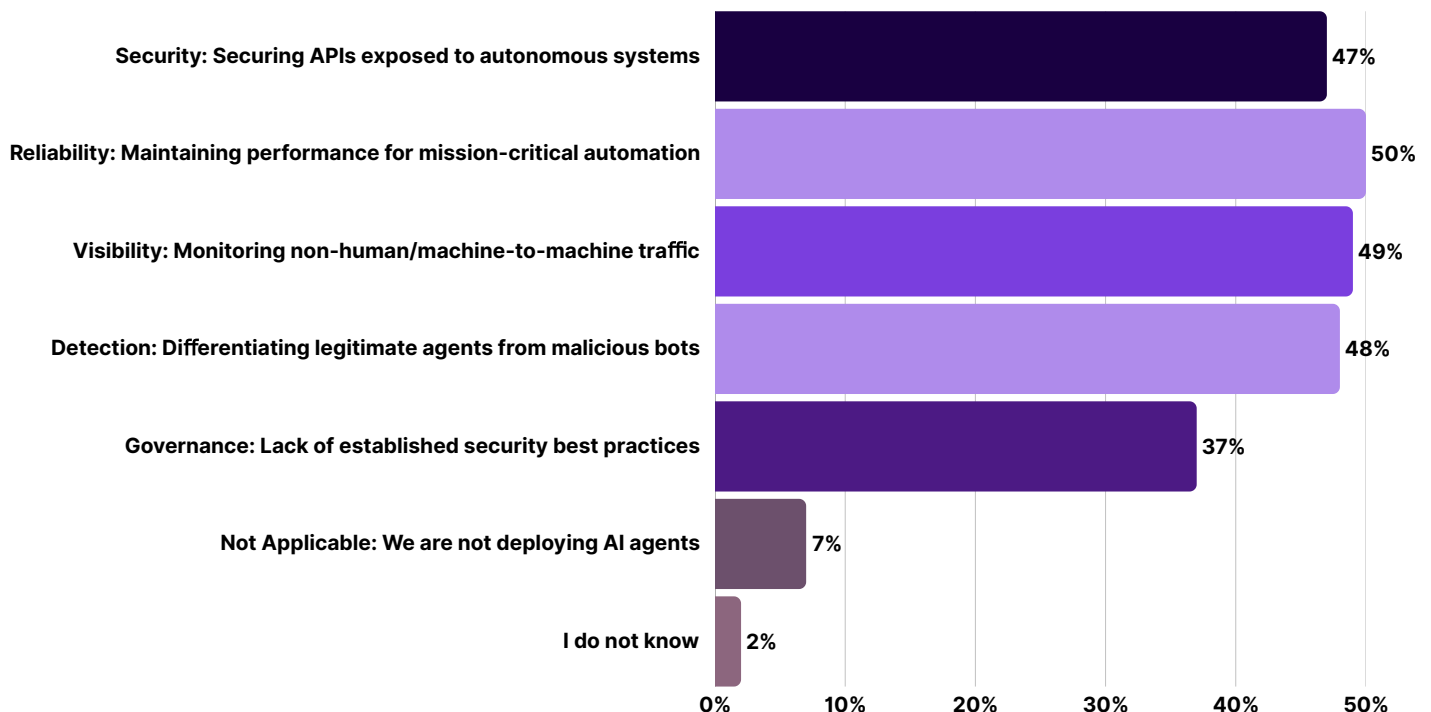
Security Challenges with AI Agents

Beyond development, AI agents introduce entirely new operational risks. Organizations deploying agent-based systems reported several key challenges:

- **50% – Maintaining reliability for mission-critical automation**
- **49% – Monitoring machine-to-machine traffic**
- **48% – Detecting malicious bots impersonating legitimate agents**
- **47% – Securing APIs exposed to autonomous systems**
- **37% – Lack of established governance frameworks**

These results highlight the growing importance of securing what many security leaders now refer to as the Agentic Action Layer—the APIs that allow AI agents to interact with enterprise systems.

What are the most significant challenges for your organization when deploying AI agents that rely on APIs?



Measuring ROI in API Security

From Cost Avoidance to Innovation Enablement

As API ecosystems grow and security investments increase, organizations are under pressure to demonstrate the business value of API security initiatives.

Historically, API security investments were justified primarily through risk reduction and breach prevention. While these outcomes remain critical, many organizations are now recognizing that mature API security programs can also enable faster innovation and safer deployment of new technologies such as AI.

Business Impact of API Security

The survey reveals that API security challenges already affect business operations:

- **47% of organizations have delayed releasing a new application due to API security concerns.**
- **32% experienced an API security incident in the past year.**

These results demonstrate that API security is not just a technical issue—it directly impacts time-to-market, operational continuity, and enterprise risk.

Measuring Value: Security as an AI Enabler

While risk reduction remains a core metric, forward-thinking organizations are measuring the ROI of API security through a new lens: AI Enablement. With 47% of organizations delaying a production release due to API security concerns, a lack of a robust Agentic Security platform is a direct bottleneck to AI adoption. Organizations are increasingly using metrics such as:

- **Speed to safe deployment of LLM-integrated tools**
- **Reduction in "Mean Time to Remediate" (MTTR) for AI-generated code vulnerabilities**
- **Compliance velocity for regulations like the EU AI Act**

Ultimately, robust Agentic Security is no longer viewed merely as a cost center for breach prevention, but as the mandatory governance layer required to safely unleash autonomous AI agents at enterprise scale.

Conclusion and Recommendations

Securing the Agentic Future

The findings of the 1H 2026 State of API Security survey reveal a clear pattern: API adoption is accelerating rapidly, driven by digital transformation and the rise of AI-powered applications. However, security maturity has not kept pace with this growth.

Organizations face several recurring challenges:

- **Limited visibility into API inventories**
- **Inconsistent monitoring practices**
- **Rapid expansion of API ecosystems**
- **Emerging risks associated with AI-generated code and autonomous agents**

To address these challenges, security leaders should prioritize the following actions.

1. Provide the Full Agentic Security Graph

Organizations cannot secure APIs they cannot see, nor can they secure AI agents without seeing what data they touch. Move beyond static API lists to build a dynamic Agentic Security Graph that maps the relationships between Large Language Models (LLMs), Model Context Protocol (MCP) servers, and the underlying APIs.

2. Implement Real-Time Monitoring and Behavioral Detection

Continuous monitoring of API traffic is critical for detecting anomalous behavior, identifying attacks, and protecting machine-to-machine interactions.

3. Adopt Context-Aware Security for AI-Driven Development

Traditional scanners lack context. Organizations must enforce strict API and AI security standards directly within the IDE and CI/CD pipelines, validating AI-generated code before it merges into the main branch.

4. Secure the Agentic Action Layer

APIs that enable AI agents represent a new and rapidly expanding attack surface. Security strategies must evolve to protect autonomous systems interacting with enterprise infrastructure.

5. Align Security Investments with Business Outcomes

Organizations should measure API security success not only by reducing incidents but also by improving resilience, readiness for compliance, and faster innovation.

Conclusion and Recommendations II

Final Note

APIs are now the operational backbone of the AI-driven enterprise.

As organizations deploy autonomous agents, generative AI systems, and automated workflows, APIs are evolving from simple integration points into **the execution layer of modern digital infrastructure.**

Security leaders who recognize this shift, and invest in robust API security strategies, will be best positioned to **enable innovation while protecting their organizations from the emerging risks of the agentic era.**

For additional reading, to further understand how AI agents interact across APIs and where new risks emerge, explore our latest perspective on securing the Agentic Action Layer: [“Everyone Is Deploying AI Agents. Almost Nobody Knows What They're Doing.”](#)



About Salt Security

The Salt Security API Protection Platform secures your APIs across the full API lifecycle. The Salt platform collects a copy of API traffic across your entire application landscape and uses big data, machine learning (ML), and artificial intelligence (AI) to discover all your APIs and their exposed data, stop attacks, and eliminate vulnerabilities at their source.

The Salt platform:

Discovers all APIs and exposed data – Automatically inventory all your APIs, including shadow and zombie APIs, and highlight all instances where your APIs expose sensitive data. Continuous discovery ensures your APIs stay protected even as your environment evolves and changes with agile DevOps practices.

Stops API attackers – Pinpoint and stop threats to your APIs by identifying attackers early, during their reconnaissance phase, and prevent them from advancing. The Salt platform correlates activities back to a single entity, sends a consolidated alert to avoid alert fatigue, and blocks the attacker rather than transactions.

Improves your API security posture – Salt proactively identifies vulnerabilities in your APIs even before they serve production traffic. The platform also uses attackers like pen testers, capturing their minor successes to provide insights for dev teams while stopping attackers before they reach their objective.

About Salt Labs

Salt Labs identifies API threats and vulnerabilities in customer deployments and in the wild. Our in-depth API threat research reports document the steps of an exploit, including the processes and tooling, to reveal an attacker's approach, the details of an exploit, the risk to the business, and the steps an organization can follow to avoid falling victim to a similar attack. We also apply our research findings to improve the ML and AI algorithms at the heart of our API security platform, so that all our customers benefit from our ongoing research. Our industry reports, such as this State of API Security Report, tap empirical and survey data to educate the market on API security trends.