ProjectDiscovery

The *AI code* deluge

Findings from security teams in the age of AI-assisted engineering

Table of contents

03 — Executive summary

04 — Engineering delivery is accelerating

05 — The pressure on already-stressed security teams is increasing

06 — AI-generated code is escalating business risks

07 — More tools = More signal and more work

11 — Security practitioners see AI's potential—but don't yet trust It

13 — Closing the gap

14 — Why Neo

16 — Methodology

Executive summary

The use of AI-assisted coding, where Large Language Models (LLMs) help software developers write, debug, and optimize code, has grown significantly since 2023. With the introduction of agentic AI in 2025, it is fast becoming a standard tool in every developer's kit. At ProjectDiscovery, we wondered what that means for the security teams who are on the receiving end of the code.

In March 2026, we commissioned a blind survey of 200 cybersecurity practitioners across North America and Western Europe to explore how the rapid adoption of AI-assisted coding is affecting those people responsible for keeping software secure. The respondents represent a cross-section of mid-to-large enterprises (the majority between 1,001 and 5,000 employees), with nearly half working in security architecture and strategy and more than half involved in selecting or approving security products.

The findings paint a clear picture: a deluge of AI-generated code is hitting security teams, and the wave is building faster than most organizations can absorb it. Engineering teams are shipping at an unprecedented speed, and security teams are standing in the path of that rising tide. The study offers three key insights:

1. **Code volume is growing, but so is the stress.**
   100% of respondents reported increased engineering delivery in the last twelve months. 49% attribute most or all of that increase to AI-assisted coding. The code is arriving faster than security teams can review it. Nearly 60% of respondents say they are able to keep up with the increased code volume for now, but it is getting harder. While security professionals have been stretched and over-stressed for years, AI-assisted coding is now adding to their burden.

2. **Security practitioners are stuck in a triage loop.**
   66% of respondents spend more than half of their time manually validating findings. The top activities consuming their weeks are alert triage, coordinating fixes, and proving exploitability; not resolving the actual vulnerabilities that put their organizations at risk.

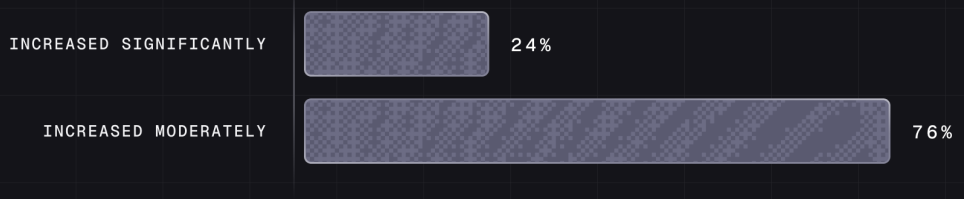
3. **Trust is the gating factor for adoption of AI-powered security.**
   Practitioners want audit trails, scoped access, and clear evidence before they will trust AI in their own workflows. They are not opposed to the technology, but they need it to earn its place.

KEY FINDING 01

Engineering delivery is accelerating

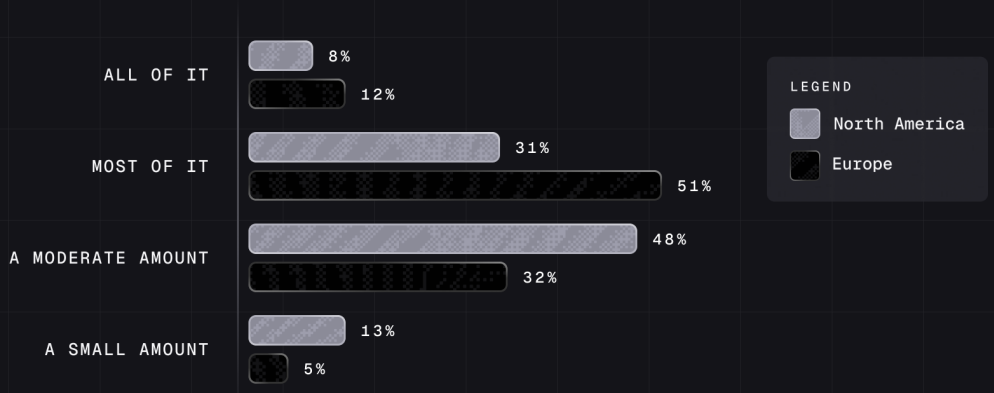
All survey respondents reported that the speed of engineering delivery in their organization has increased over the past 12 months. The majority (76%) described that increase as moderate, while 24% called it significant.

In the past 12 months, has the speed of engineering delivery in your organization changed?



This tracks with broader industry projections. Gartner forecasts that 75% of enterprise software engineers will use AI code assistants by 2028, up from roughly 10% in 2023. It is no longer a question of whether AI is making engineers faster, it is how much of the acceleration AI accounts for. In our study, 49% of respondents said AI-assisted coding is responsible for most or all of the increase in delivery speed. The other 51% attribute it to a combination of factors, with AI playing a moderate or smaller role. Given the increase over just the last two years, we can expect utilization to continue growing, further accelerating code output.

How much of that increase do you attribute to AI-assisted coding?



Regional adoption is surprisingly skewed so far. In Europe, 63% of respondents attributed most or all of the increase to AI-assisted coding, as opposed to only 39% in North America. European engineering teams may be adopting these tools more aggressively, or those security professionals may be more aware of the role AI plays in what their engineering counterparts are shipping.

KEY FINDING 02

The pressure on already-stressed security teams is increasing

Security's pace against growing code volumes is starting to lag. When we asked security professionals how well they are handling the expanding code that needs their review, the responses indicate an inflection point.

Only 38% of respondents said they are keeping up well. Nearly 60% said they are managing to keep up so far but it is getting harder. That should get attention in every CISO's next planning meeting. Respondents from mid-sized businesses (having between 1001-5000 employees) made up the largest share experiencing this pressure (69%). And 2% of respondents admitted they are already falling behind. While that number might seem small, consider what it takes for a security professional to openly admit their team is losing ground.

Given that the last 12-18 months have been the real acceleration period for AI-assisted coding adoption, the squeeze on security teams is still new but is likely to change fast. More code will put them further behind, more quickly.

The application security workload issue identified through our survey reflects a systemic issue across the entire cybersecurity profession. In the International Information System Security Certification Consortium (ISC2) 2025 Cybersecurity Workforce Study of over 16,000 cybersecurity professionals, nearly half (48%) of respondents felt exhausted from trying to stay current on the latest threats and emerging technologies, and 47% felt overwhelmed by their workload.

There is also a compounding factor of more risk per unit of code volume. For instance, Gartner has noted that by 2028, prompt-to-app approaches adopted by citizen developers will increase software defects by 2500%, triggering a software quality and reliability crisis. This means that the code getting faster is also, in many cases, getting less secure.

> 60% of security teams say keeping up with the increased code volume is getting harder.

KEY FINDING 03

AI-generated code is escalating business risks

We asked respondents to rank the top security challenges that AI-assisted coding has introduced or amplified in their organizations. The results indicate that there are growing business-centric risks to the organizations that are using it.

Rank the top 5 challenges that AI-assisted coding has introduced/amplified in your organization's security environment.

Secrets exposure came in as the top challenge (78%), and it is not hard to understand why. Over the past couple of years, employees' use of generative AI for work purposes has increased dramatically, and not all are using it responsibly. A 2025 National Cybersecurity Alliance report found that 43% of workers admit to plugging sensitive workplace information into AI tools without their employer's knowledge. (To be fair, 58% report receiving no training on AI's security or privacy risks. Employers take note!)

78% of security practitioners rank secrets exposure as the top challenge introduced or amplified by AI-assisted coding.

When engineers paste code and context into AI tools without guardrails, secrets leak. When AI tools generate code, they sometimes hardcode credentials or API keys that should be managed externally. Either way, the secrets problem is growing in tandem with AI adoption.

Interestingly, European respondents rated secrets exposure higher than those in North America (87% vs 72%), likely reflecting the heightened awareness that comes from operating under the General Data Protection Regulation's (GDPR) strict requirements.

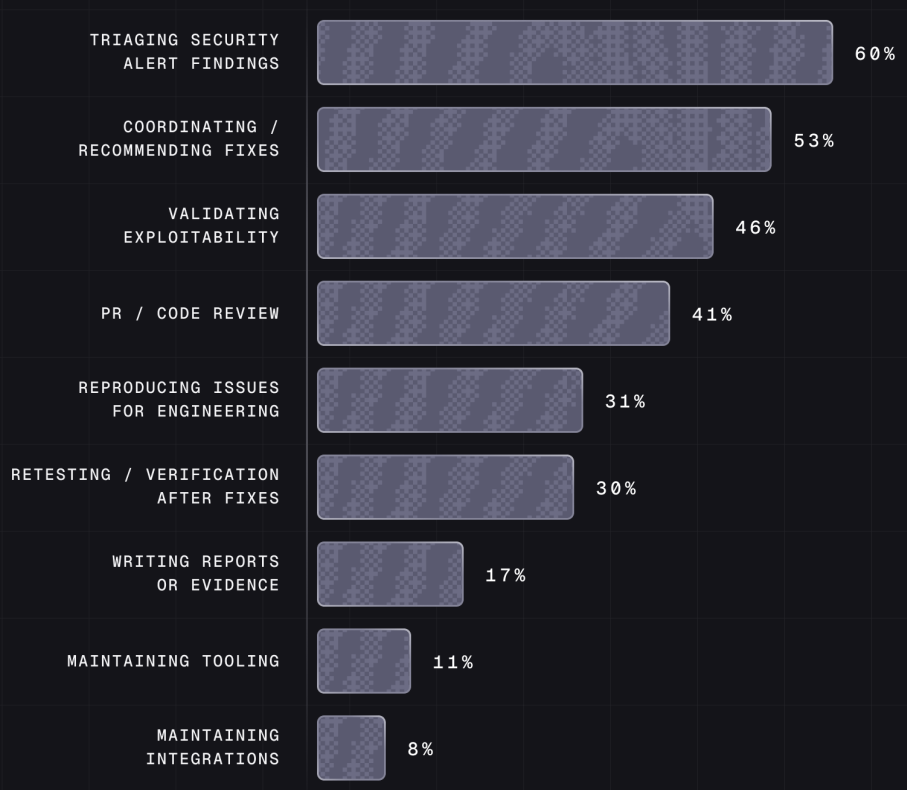
Supply chain risk, second-highest ranked at 73%, and business logic vulnerabilities, ranking third at 72%, are arguably more concerning for the long term. They point to a fundamental limitation of current AI coding tools: generating code that is syntactically correct but often lacks awareness of the broader system architecture and business rules. The AI does not know your authentication model. It does not understand your payment flow. It writes what looks correct and moves on.

KEY FINDING 04

More tools = More signal and more work

Insights into how security teams spend their time is key to understanding the daily pressure they face. It has been the case for years that security teams have felt stretched, with many admitting to approaching burnout. A National Institutes of Health report explains cybersecurity fatigue as a significant factor contributing to burnout, reduced productivity, and increased psychological strain. Yet security teams bear a large part of the burden for keeping their organizations safe. If their environment has already been difficult for years, AI-assisted coding is now making it harder.

Of which of the following activities does your application security team spend the most time each month?

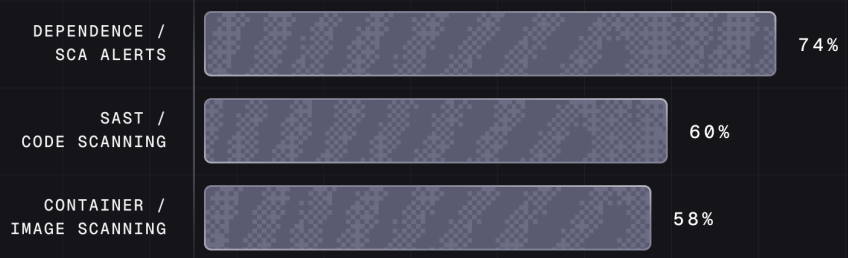


The top three activities security pros undertake in a week are about sorting, verifying, and proving; not about fixing. The majority of their efforts go toward determining what is real, convincing engineering teams that it matters, and then proving it all over again. The actual work of resolving vulnerabilities gets deprioritized given lack of time to do so.

This is compounded by the “noise” problem of alert fatigue. AI-assisted coding creates more code volume and correspondingly more noise. When we asked which sources of low-value alerts most distract from addressing exploitable vulnerabilities, dependency/SCA alerts (74%), SAST/code scanning (60%), and secrets scanning (58%) topped the list.

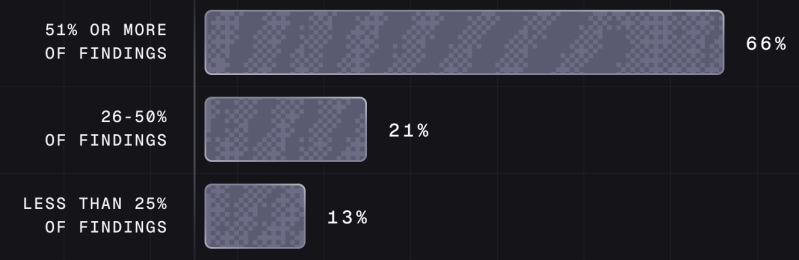
The growing volume of low-quality data generated by “vibe coding” that is entering the security funnel forebodes a potential state of operational paralysis, where primary security pillars like SCA, SAST, and secrets scanning are being considered distractions instead of useful and productive tools. More tools are not the answer; competing or overlapping tools result in additional signal that security pros then must sift through.

Which are the top sources of low-value security alert ‘noise’ that distracts you from addressing exploitable vulnerabilities?



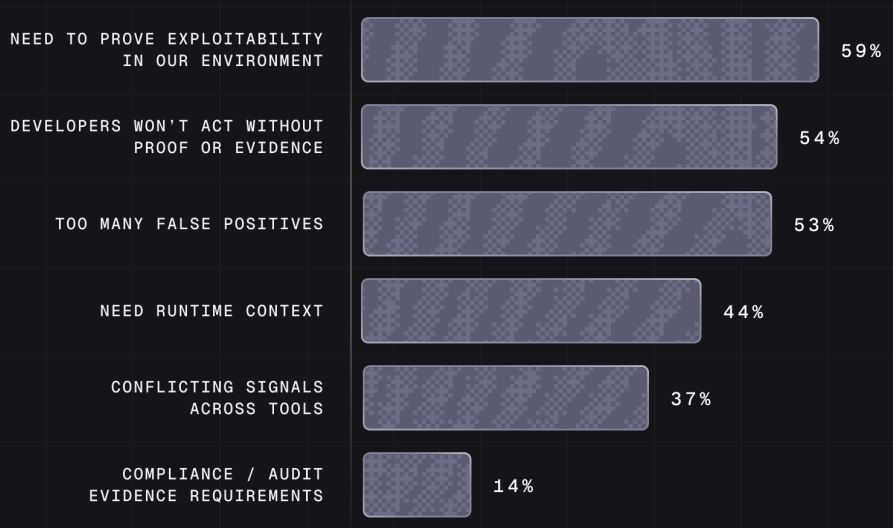
The need for manual validation exacerbates the challenge. We asked respondents what percentage of their team’s security findings require manual validation or reproduction before they can be prioritized. 66% reported that more than half require manual work; 53% say it’s as high as 51% to 75%.

What percentage of your team’s security findings typically requires manual validation and/or reproduction before it can be prioritize and routed?



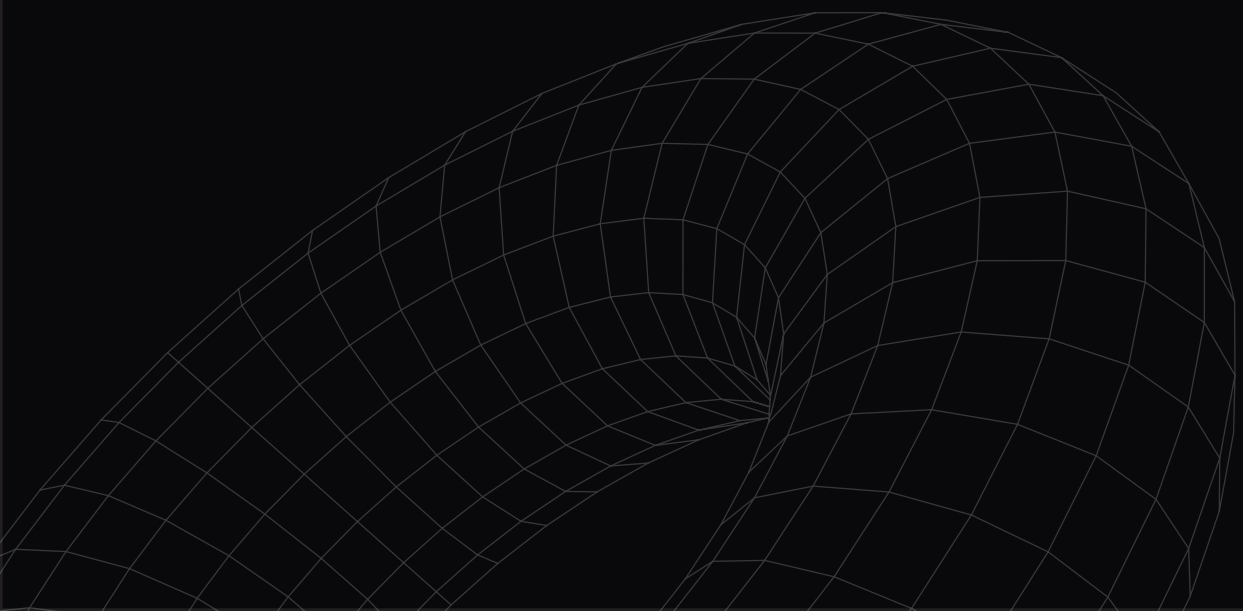
The triggers for manual intervention reveal a cycle in which findings are discovered, most cannot be trusted at face value, engineers reasonably demand evidence, and security practitioners spend their days building that evidence by hand. This indicates a growing, systemic failure where AI-based tools meant to save time are actually creating more manual labor.

In your security team's validation/reproduction workflow, what typically triggers manual validation?

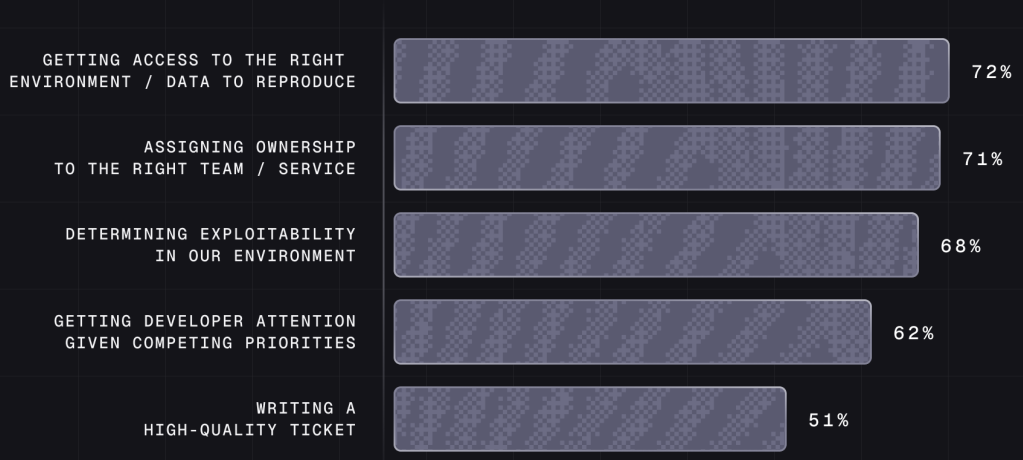


66% of security practitioners spend more than half their time manually validating findings, not resolving vulnerabilities.

We also asked respondents to rank the top bottlenecks in their vulnerability triage process. The results were tightly clustered, with only four percentage points separating the first from the third issues.



Of which of the following activities does your application security team spend the most time each month?



While most of the top reasons are fundamentally resource and coordination problems, determining exploitability (67.6%) is a technical problem that can be addressed through better—not more—tooling.

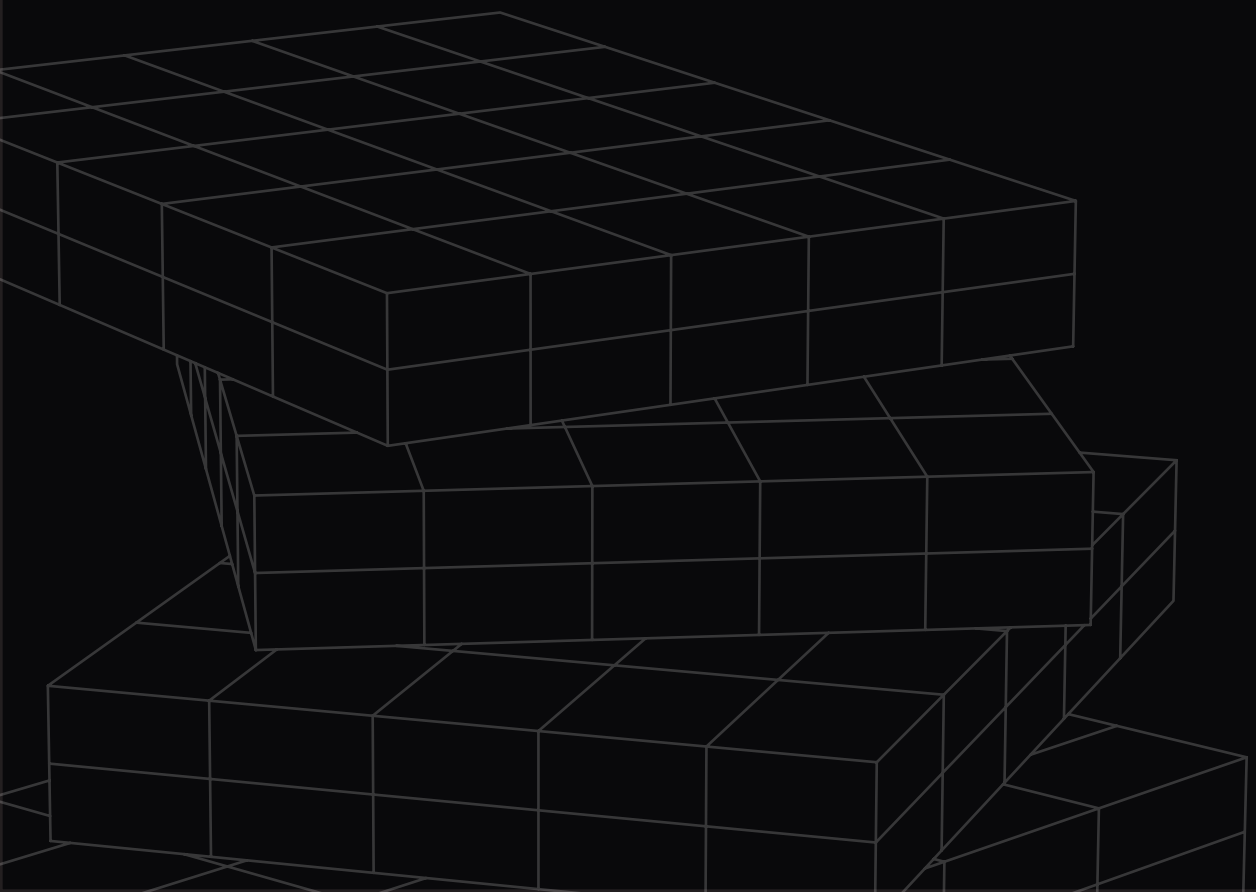
A reliable tool that can prove (with evidence!) whether a finding is actually exploitable in a specific environment would ease the chokepoint being created by multiple disparate tools. That would in turn enable security teams to act with confidence, freeing time for vulnerability resolution and other high security priorities. This is where AI, thoughtfully applied, could make a real difference.

KEY FINDING 05

Security practitioners see AI's potential—but don't yet trust It

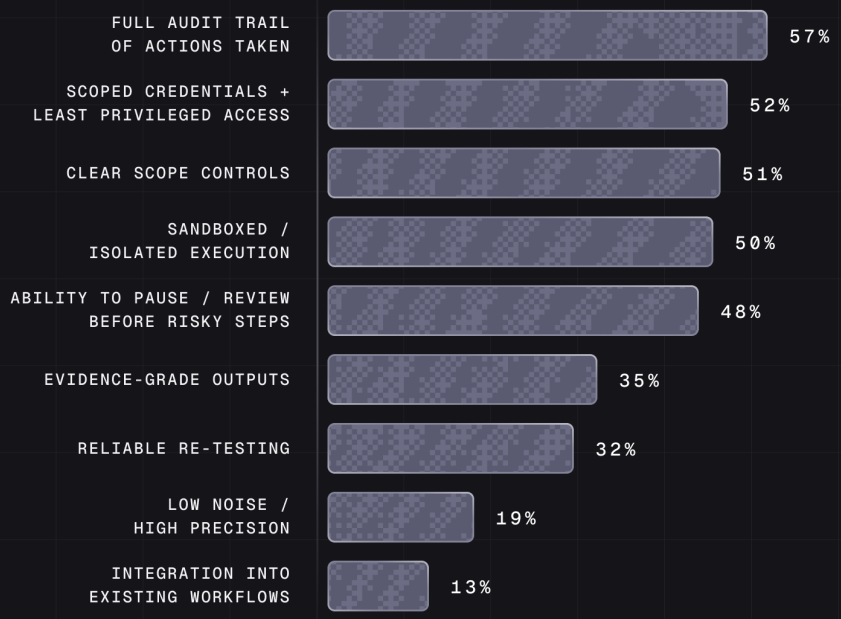
As security teams face the reality that they will need to embrace AI tools to keep up with increased engineering output, they are somewhat reluctant to do so. For instance, penetration testing, a fundamental security task, is an obvious function to start with AI-based tool adoption. Respondents indicated strong preference for several associated use cases:

Rank your top preferred use cases for potential AI-based pen-testing capability.



However, preferences for what could be helpful do not equate to trust in AI's reliability for performing these tasks. When asked what it would take for them to actually trust AI-based pen testing within the next 12 months, there was a clear indication that respondents are wary.

Which of the following would be needed for your organization to trust AI pen-testing within the next 12 months?



Their number one trust requirement is providing a full audit trail. Security pros want to see exactly what an AI system did, when it did it, and what it found. These experts have spent their entire careers being careful, methodical, and evidence-driven. They are not going to hand over the keys to a system they cannot inspect.

This is healthy skepticism, not resistance. As the ISC2 2025 Workforce Study noted, the security profession suffers exhaustion from trying to stay current on the latest cybersecurity threats. Now they are being asked to evaluate and adopt new AI tools, while already running at capacity.

57% of respondents say a full audit trail is the #1 requirement for trusting AI-based pen testing.

The respondents who report they are keeping up well with increased code volume ranked the ability to pause/review before taking risky steps as their third highest trust requirement (39%). This could be because the most successful practitioners only trust AI output when they have control mechanisms for intervening before that output becomes production code, saving later clean-up.

The message for anyone building AI-powered security tools is clear: earn trust through transparency, auditability, and precision. Do not ask security teams to take it on faith.

Closing the gap

This is where it gets better. The data in this report describes a security workforce that is skilled and committed, but caught in rising water. They are dealing with more code, more alerts, more false positives, and more manual proof-of-concept work than their bandwidth can sustain. With AI-assisted coding adoption accelerating, the volume will only keep climbing.

But the survey also points clearly to where the right tooling can make the biggest difference:

Automated exploitability validation.
The single most impactful thing a tool can do is answer the question “is this actually exploitable in our environment?” with proof; not a confidence score or severity rating. This will eliminate the manual reproduction cycle that consumes most of security teams’ time.

Evidence that engineering teams will act on.
Developers will not prioritize a finding based on a scanner alert. They need reproducible evidence, clear impact, and actionable remediation steps. Tools that deliver pentester-grade evidence close the gap between “finding” and “fix.”

Transparent, auditable AI.
Security practitioners are ready for AI to help them. They are not ready for AI to operate unsupervised. Audit trails, scoped access, sandboxed execution, and the ability to inspect every action are prerequisites for trust.

Noise reduction at the source.
When 73.5% of practitioners say dependency alerts are their top source of low-value noise, the answer is not to add more scanning. It is to validate what is actually exploitable and suppress the rest.

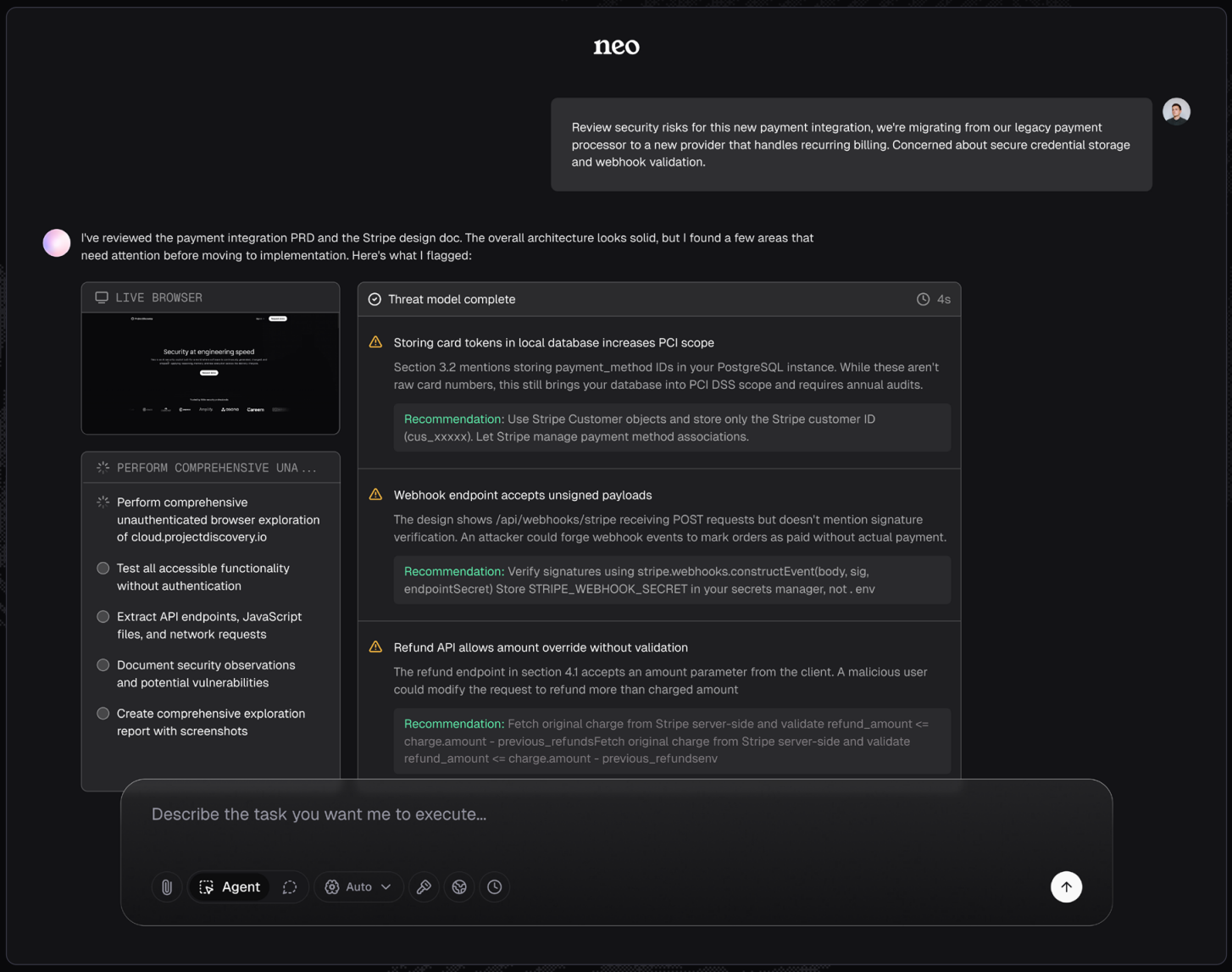
Why Neo

Look at the toolchain a typical security team operates today: SAST, DAST, a separate pen testing engagement every quarter, if the budget allows. Each tool generates its own findings in its own format with its own noise. The team stitches it all together manually, cross-referencing results, deduplicating alerts, and trying to build a coherent picture of what is actually exploitable.

This is not a people problem. Security teams are good at what they do. It is a structural problem. The tools were built for a world where engineering shipped on a predictable cadence and security had time to keep up. That world is gone.

What this survey makes clear is that security practitioners do not need more scanners. They need fewer tools that go deeper, deliver proof instead of alerts, and work the way their teams actually work. They need a single system that can do what today requires three or four disconnected products and a lot of manual stitching in between.

That is the premise behind Neo.



Neo is a multi-purpose security testing platform powered by frontier AI models, built to unify SAST, DAST, and automated penetration testing into one workflow. Think of it as an offensive security engineer fluent in ProjectDiscovery's open-source toolkit, the same toolchain trusted by over a million practitioners through Nuclei, while also drawing on the broader open-source security ecosystem and 30+ agent-native tools purpose-built for how AI reasons about vulnerabilities. When a novel vulnerability surfaces, Neo researches it, assembles the right tools to test it, and validates exploitability in context. It handles the full lifecycle from discovery through verified remediation in one place.

"Neo surfaced real race-condition and payment-bypass scenarios with step-by-step reproduction. We stopped spending days recreating timing-sensitive bugs manually."

Security Engineering Lead
Application Security Team, Fortune 500 Restaurant Chain

Nuclei's thesis was always proof over pattern matching. Neo extends that principle to the vulnerabilities pattern-based scanners were never designed to reach: business logic flaws, multi-step authorization failures, and the kind of contextual, application-specific risks that only surface when you test the way a skilled human researcher would. The difference is that Neo can do this continuously, across every release, without waiting for the next quarterly pen test.

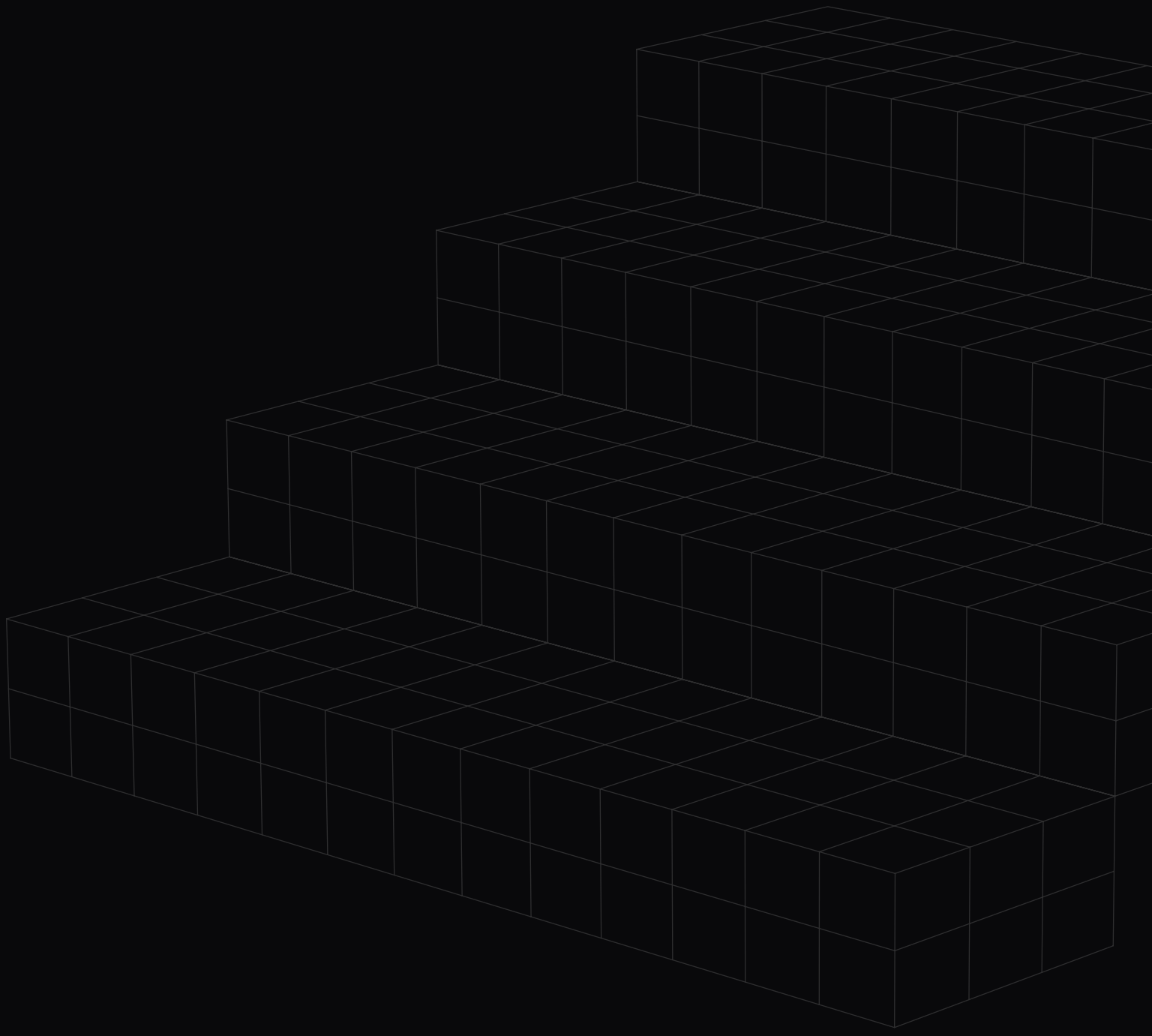
More importantly, Neo was shaped by the same concerns this survey surfaced. Security practitioners told us they need audit trails, scoped access, evidence-grade outputs, and the ability to see exactly what a tool did and why. Those are not features we bolted on. They are the design constraints we started from, because we believe the only way AI earns a place in security workflows is by meeting practitioners where they are: skeptical, methodical, and rightly protective of their environments.

The goal is not to replace security teams. It is to give them back the time they are currently losing to triage, manual reproduction, and proof-of-concept work, so they can focus on the judgment calls and remediation decisions that actually require a human being.

Engineering is not going to slow down. The volume of AI-generated code is only going to grow. The question for every security organization is whether their tooling can keep pace without burning out the people behind it. We built Neo because we believe it can.

Methodology

This report is based on a blind survey conducted in February and March 2026, fielded by a third party firm. The survey consisted of 21 questions (including 4 demographic questions) and collected 200 responses from cybersecurity practitioners: 118 in North America and 82 in Western Europe. Respondents were screened for qualifying job activities, organization size, and involvement with AI-assisted coding. More than half of respondents are involved in selecting or approving cybersecurity products and services. Organizations represented range from 501 to over 10,000 employees, with the majority (65.5%) in the 1,001 to 5,000 employee range.



See Neo in Action

Bring your noisiest repo, your busiest service, or a real PR. We'll show Neo running end-to-end.

Request a Demo

ProjectDiscovery

ProjectDiscovery is the open-source cybersecurity company behind Nuclei, the world's most widely used vulnerability scanner, with over 10 billion scans run and a community of more than 100,000 security practitioners. Winner of the RSAC 2025 Innovation Sandbox, ProjectDiscovery builds the tools that security teams trust to find what is actually exploitable and prove it.