



First Edition

Software Quality Pulse Report

Table of Contents

01 - Introduction	03
02 - Foreword	04
03 - Key Insights	05
04 - Global Quality Trends	09
05 - QA Testing Insights	20
06 - Software Security Insights	40
07 - QA and Security Convergence	63
08 - Survey Details and Methodology	72
09 - Conclusion	78
10 - About Sembi and Portfolio	80

Introduction

The State of Software Quality Has Never Been More Complex—or More Critical

3,800 QA and security professionals share what is really happening on the ground in 2026.

The Sembi Software Quality Pulse Report is a survey-driven research report designed to capture the real state of software quality and security across the industry. Drawing on responses from 3,800 QA engineers, security professionals, developers, and engineering leaders, this report surfaces the trends, challenges, and priorities shaping how teams build, test, and secure software today.

The industry is at an inflection point. AI-assisted development is reshaping how code is written, accelerating release cycles, and blurring the line between quality assurance and security in ways that demand a new approach. Against this backdrop, the insights in this report explore seven core themes:

1. Release velocity and its tradeoffs
2. Automation maturity
3. DevOps integration
4. The reactive state of security
5. AI's uneven impact
6. The persistent talent gap
7. And the growing convergence of QA and security under a unified quality operations model

The challenges ahead are real, and no team is facing them alone. This report is a reflection of an industry in motion and serves as a reminder that the path forward is one we're all charting together.

Foreword

I've spent my career watching software quality evolve from an afterthought bolted onto the end of the development process into a discipline that now sits at the heart of how organizations compete, innovate, and earn their stakeholders' trust. In my experience, that evolution has never moved faster than it is moving now.

What strikes me most in this year's report is not any single statistic, it's the picture they paint together. 3,800 QA engineers, security professionals, developers, and engineering leaders took time to tell us what's really happening inside their organizations. They described an industry under pressure: accelerating release cycles, expanding system complexity, growing skills gaps, and threats that refuse to stand still. These are not new problems, but the stakes have never been higher.

Quality is no longer a technical concern; it is a business imperative. Organizations that understand this are embedding quality and security throughout the development lifecycle, not simply treating them as final gates. Those who don't are falling behind.

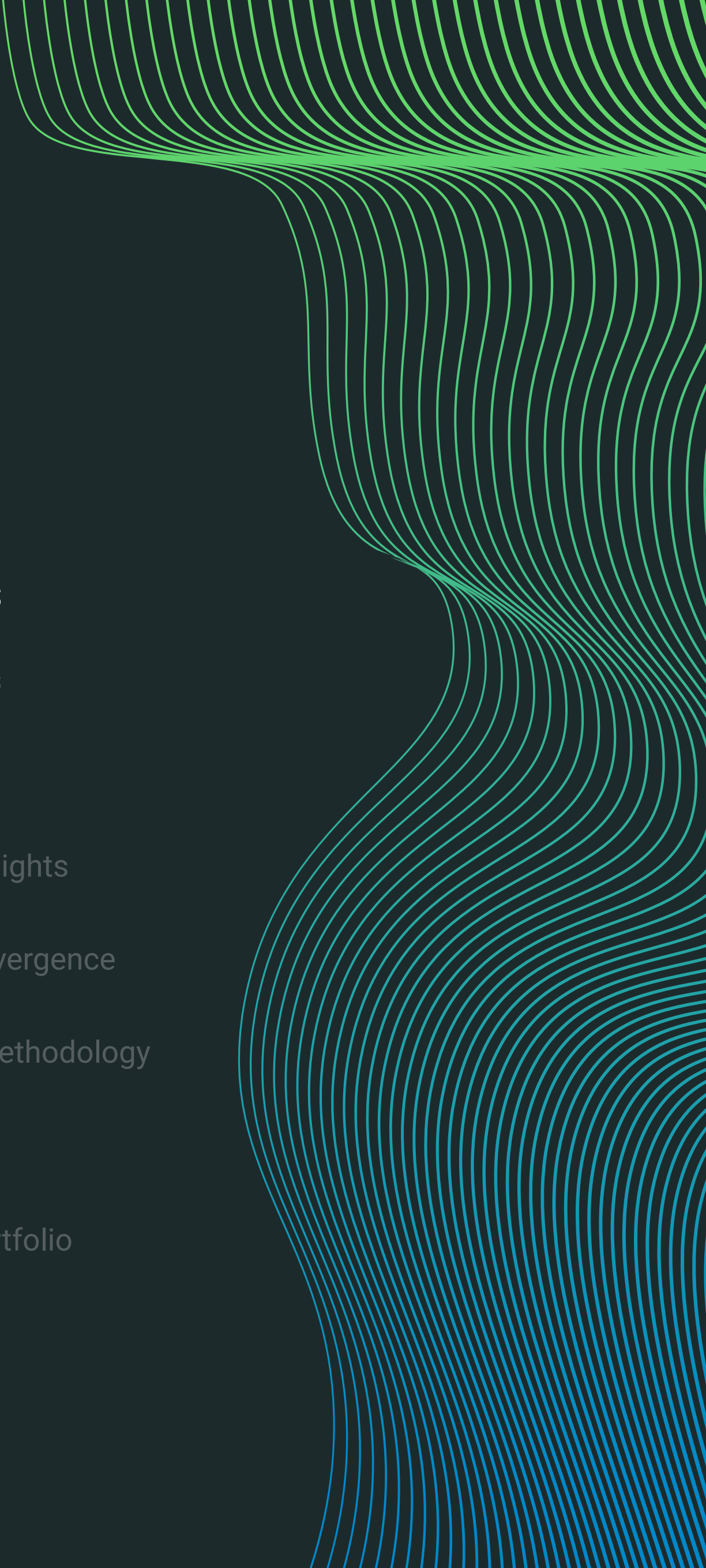
AI-assisted development is the most consequential shift our industry has seen in decades. Respondents estimate that 53% of their organization's code on average is AI-generated or AI-assisted. That means teams can move faster and build more than ever before, but it also introduces new risks many existing QA and security processes were not designed to handle. The opportunity is extraordinary, yet so is the responsibility.

That brings me to what I believe is the defining opportunity (and challenge) for our industry: scaling AI development through a shift toward test-driven development practices. For too long, quality processes were a checkpoint, now they need to advance earlier in the lifecycle. To maximize this advancement we need to include security alongside functional testing as test-driven development takes hold in the age of AI. The path forward is a unified model of quality driving development, not responding to it: testing and security both advancing together and guiding AI coding to enable the speed of AI, not checkpoint it.

That is what Sembi is built for. And it is what this report is designed to help the industry navigate.

I hope you find it as valuable as we found it humbling to produce.

Mark Haney,
President, Sembi



● 03 - Key Insights

- 04 - Global Quality Trends
- 05 - QA Testing Insights
- 06 - Software Security Insights
- 07 - QA and Security Convergence
- 08 - Survey Details and Methodology
- 09 - Conclusion
- 10 - About Sembi and Portfolio

Key Insights

The First Edition Sembi Software Quality Pulse Report draws on 3,800 responses from QA engineers, security professionals, developers, and engineering leaders to surface the trends defining software quality today. What emerges is a picture of an industry in transition—accelerating in some areas, struggling in others, and searching for a more unified path forward. The seven insights below capture the most important signals from this year's data.

Speed Is Outpacing the Processes Built to Support It

Release cycles are accelerating, but quality and security practices aren't keeping up. Teams are shipping faster while simultaneously reporting more defects reaching production, more reactive security postures, and more pressure on QA to do more with less.

- **43% of teams** are releasing new code weekly or faster
- **37% of teams** report that they prioritize speed of delivery over risk
- **59%** say they are only somewhat prepared or not prepared at all for emerging QualityOps trends in 2026

Automation Is Widespread, but Far From Optimized

Most teams have made significant [investments in automation](#), yet execution gaps remain. Skill shortages, tooling complexity, and high false positive rates are limiting the return on those investments across both QA and security workflows.

- **57% of QA tests are currently automated**, leaving nearly half still manual
- Security automation sits at an **average of 55.7%**
- Respondents estimate that only **51% of detected security issues are true positives**, creating significant noise

Integration Remains the Industry's Biggest Performance Gap

Across QA and security alike, integration with DevOps pipelines is the clearest differentiator between high- and low-performing teams—and most organizations are still falling short of full integration.

- **Only about 26% of QA teams** report being mostly or fully integrated with DevOps
- **Just 9% of security teams** report fully integrated toolchains
- Fragmented tools create **delays, blind spots, and duplicated effort**

AI Is Transforming Development Faster Than Quality Can Follow

More than half of all code is now AI-generated or AI-assisted, fundamentally changing the volume, velocity, and risk profile of software output. QA and security processes built for human-written code are struggling to adapt.

- Respondents estimate that, on average, **53% of their organization's code is now AI-generated or AI-assisted**
- **61% report moderate to dramatic increases** in QA testing demand due to AI-generated code
- **Only 17% describe the impact of AI-driven testing as significant**, with most gains remaining incremental

Security Is Still Reactive—and the Industry Knows It

Despite growing awareness, security practices remain largely defensive. Tools are fragmented, involvement comes late in the development process, and many teams lack the dedicated resources needed to shift their posture from reactive to preventative.

- Data breaches (**38.8%**), cloud misconfigurations (**36.9%**), and AI/LLM threats (**32%**) top security priority lists
- Only **half of security pros** are satisfied with current processes
- Nearly 1 in 6 security teams cite **resource constraints** as their top challenge

People and Skills Are the Constraint No Tool Can Fully Solve

Talent shortages are slowing automation adoption, limiting AI integration, and stretching teams thin across both QA and security. Until organizations close the skills gap, even the best tooling will underperform.

- **44.7% of QA teams** report being understaffed
- **38.8% of security teams** report being understaffed
- Data privacy, integration complexity, and cost are the **top barriers to AI integration in QA**

The Convergence of QA and Security Is the Industry's Next Major Shift

Teams overwhelmingly recognize the value of [aligning QA and security](#)—but organizational silos, misaligned KPIs, and fragmented tooling are preventing most from achieving it. Closing this gap will define competitive advantage in the years ahead.

- **68%** say stronger QA and security alignment would be very or extremely valuable
- Conflicting KPIs (**31%**), siloed teams (**30.5%**), and late security involvement (**28.9%**) are the top obstacles
- **Only ~33% are confident** in their organization's ability to scale QA and security together

The sections that follow examine each of these themes in depth, beginning with a broad look at the macro trends shaping quality across the industry. The data is detailed, but the message is consistent: the organizations that will lead in 2026 are those investing now in integration, automation, and a unified approach to quality and security—not as separate initiatives, but as a single, strategic discipline.

○ 03 - Key Insights

● **04 - Global Quality Trends**

○ 05 - QA Testing Insights

○ 06 - Software Security Insights

○ 07 - QA and Security Convergence

○ 08 - Survey Details and Methodology

○ 09 - Conclusion

○ 10 - About Sembi and Portfolio

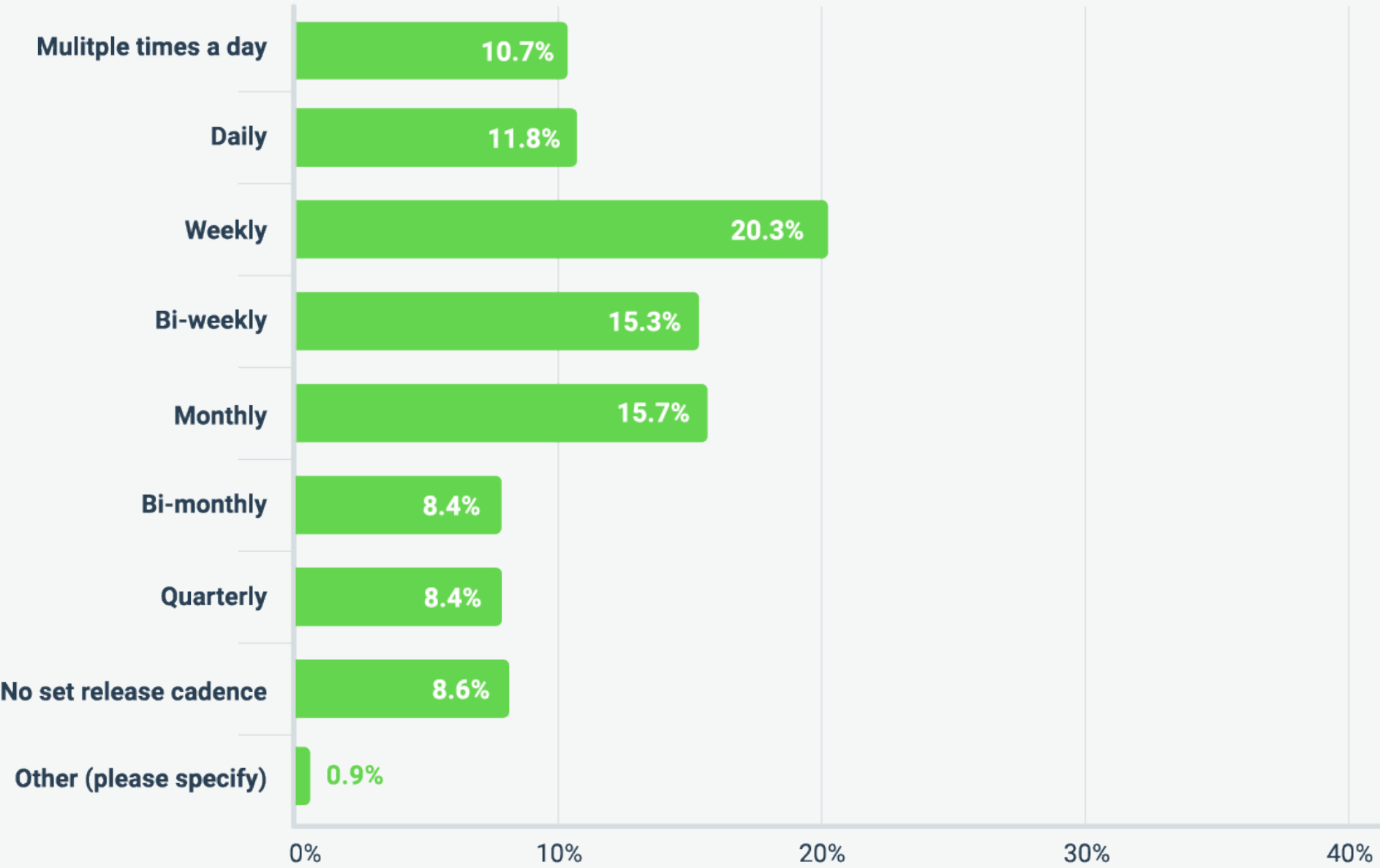
Global Quality Trends Introduction

Software quality is no longer a final checkpoint before release, it has become a continuous, organization-wide discipline. The industry has undergone a fundamental shift from isolated testing phases to integrated practices embedded throughout the entire software development lifecycle, and that evolution is still accelerating.

At the same time, the systems teams build and maintain have grown dramatically more complex. Distributed architectures, cloud-native infrastructure, microservices, and third-party dependencies have widened the surface area QA must cover, and AI is accelerating the pressure further.

Respondents estimate that, on average, 53% of their organization's code is now AI-generated or AI-assisted. That level of AI involvement is changing the volume, velocity, and risk profile of software output, forcing quality practices to adapt quickly. The traditional boundary between QA and security is eroding as a result, making the case for a unified approach clearer than ever. The data that follows reflects where 3,800 practitioners stand in the middle of this transformation.

How often does your organization release new code?



While weekly releases are the most common cadence, a significant portion of teams report releasing monthly or even quarterly. Many teams are pursuing faster delivery models, but the data shows that weekly, monthly, and quarterly release cycles remain common.

QA and security processes and tools need to be able to scale from high-frequency CI/CD pipelines to slower, more risk-averse traditional releases. This vast array of release cadence representation increases complexity in standardizing quality and security practices, and demands that teams be prepared to support multiple velocity models simultaneously.

- Release cadence varies wildly across teams
- About as many teams are releasing quarterly as are releasing daily
- Teams and tools need to be able to scale across this divide

“Release at the speed of testing. Test and tool adoption at the speed of compliance—that’s the thing for this year.”

- Chris Faraglia, Sembi Lead Solution Architect (Testing Brands)

What does your organization prioritize most and least when releasing software?



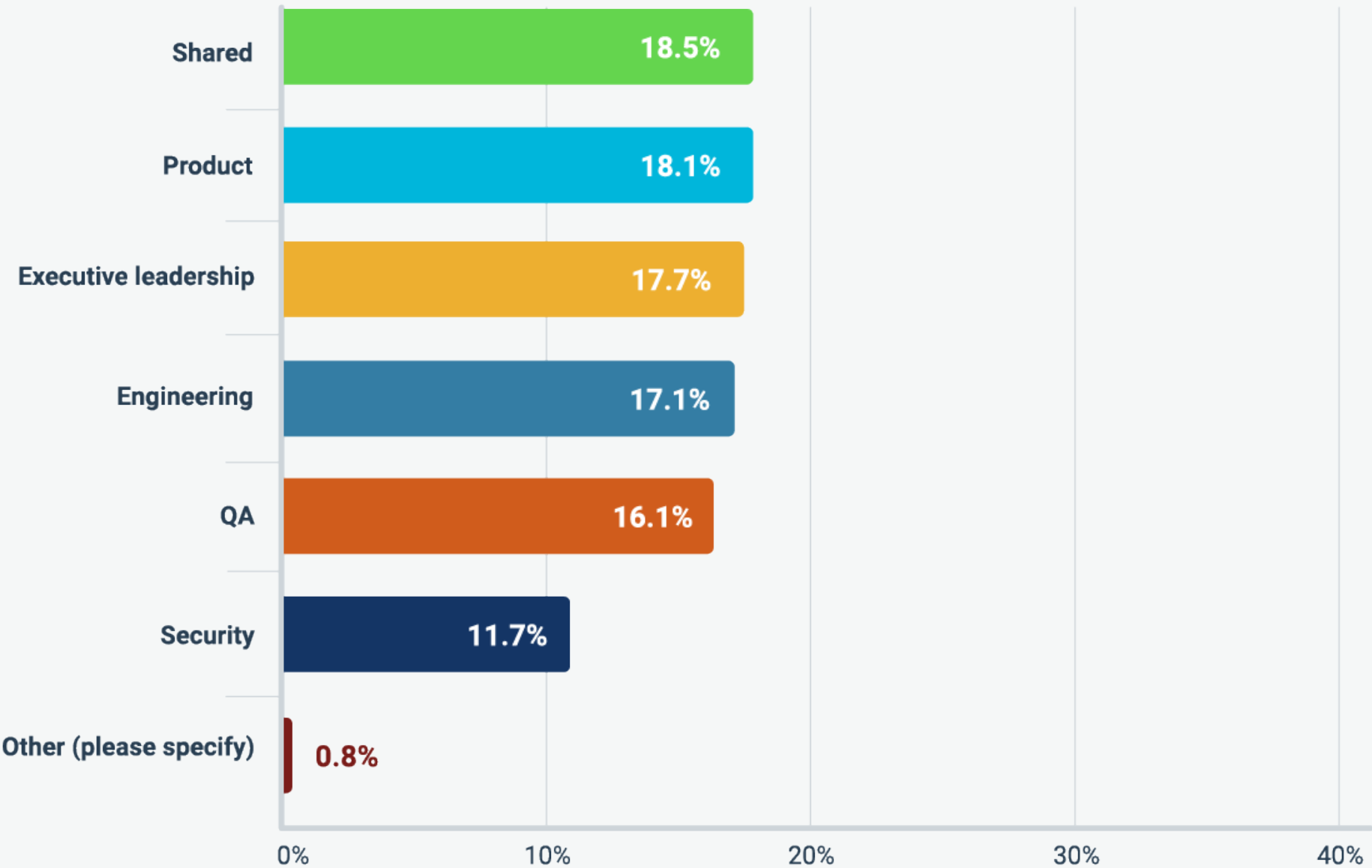
Quality, security, and compliance rank much higher than cost control

Teams are prioritizing quality, security, compliance, and stability in roughly equal measure, with only one factor appearing as a notable standout: cost control. Low prioritization of cost control suggests many teams are willing to prioritize resilience, compliance, and trust over short-term efficiency.

This indicates a massive opportunity for companies that optimize quality, security, and efficiency simultaneously. Teams, tools, and processes that support an integrated quality operations approach, shifting both quality and security left to embed high standards from the start, will rise to the top in 2026.

- **Cost control ranks as the lowest priority for over 90% of teams**
- **Quality and security are rising to board-level concerns**
- **Release speed is gated by quality and security**

Who owns final release risk decisions?



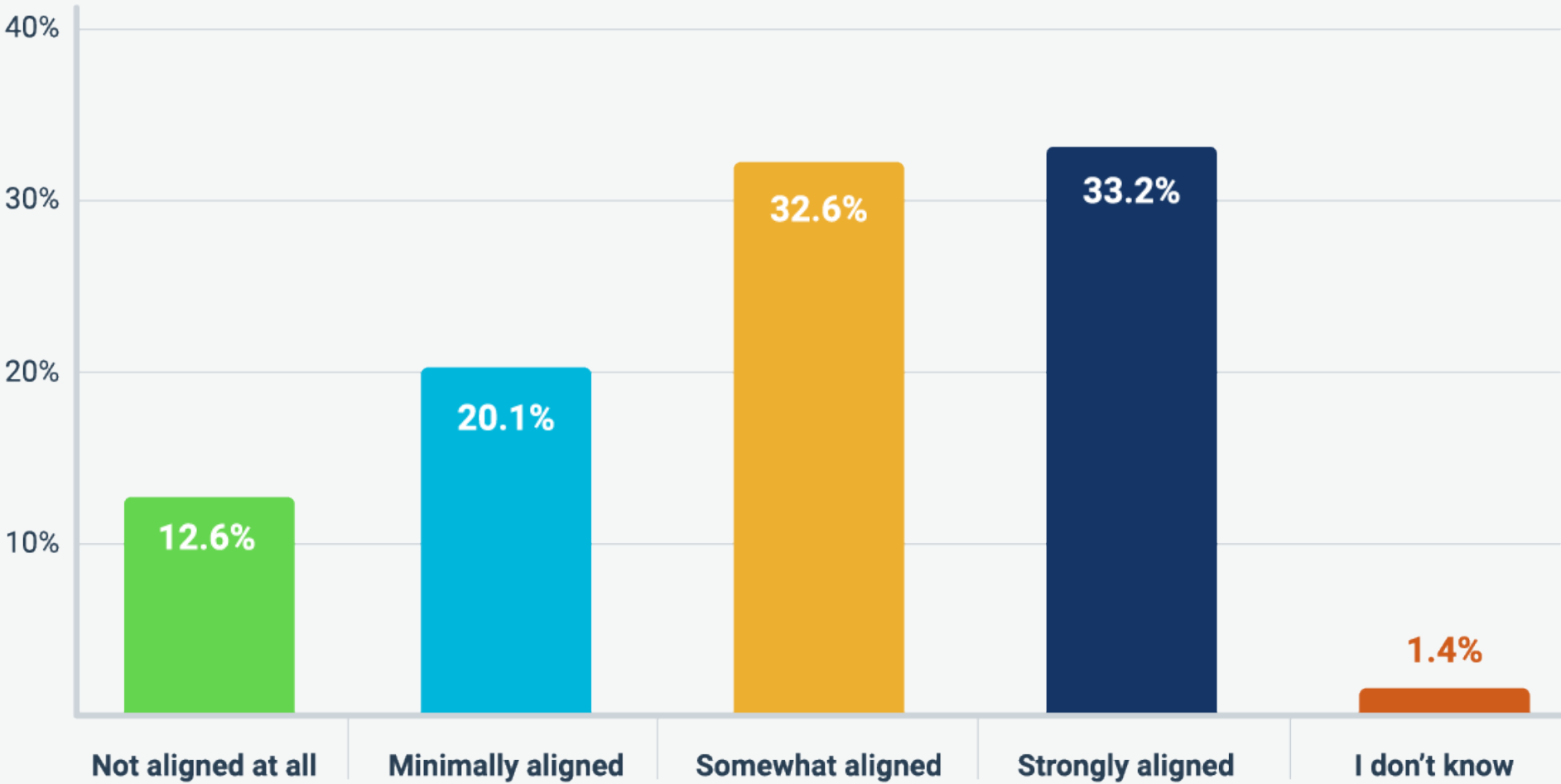
Release risk is claimed by everyone, but owned by no one

When asked to select the one function that owns release risk, teams report highly distributed results: with “Shared” ranking the highest at 18.5%, and most other options hovering between the 16–18% range. This indicates that in many teams, risk responsibility is distributed within the internal team itself, and in the industry as a whole, no clear consensus has emerged on who should own release risk.

While it is natural that several functions would want input into releases, a lack of clear ownership leads to diffused accountability and slower decisions. It also opens the opportunity for cross-functional friction, which can delay releases or introduce unmanaged risk. These findings illustrate a clear need for centralized risk scoring and decision frameworks, as well as platforms that aggregate QA and security signals into a single view.

- **Shared or unclear release risk ownership is prevalent across the industry**
- **Unclear ownership delays releases and increases risk**
- **Teams need centralized decision-making and aggregate quality and security dashboards**

How aligned are QA and security priorities?



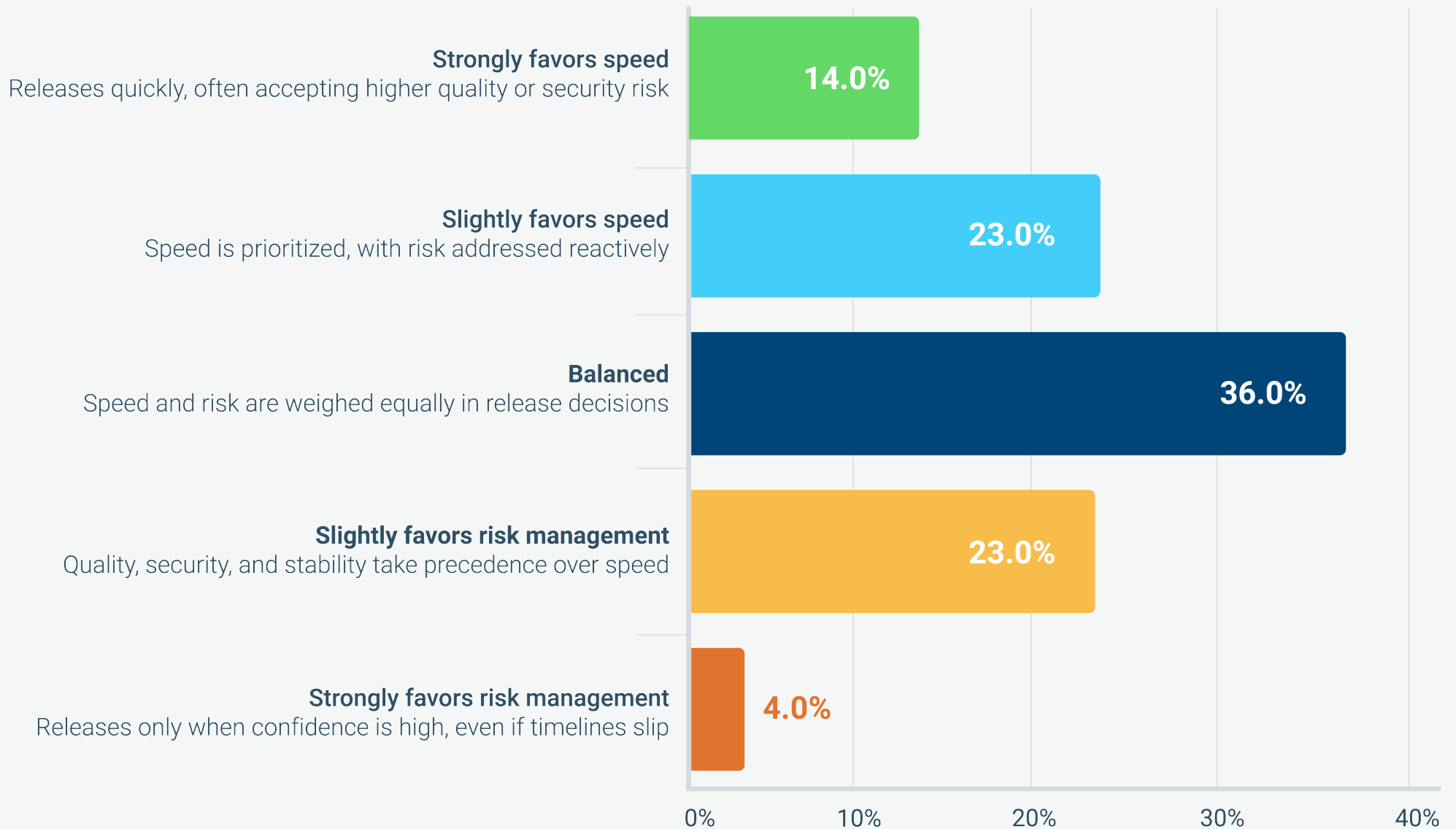
Most teams show some, but not strong, alignment between quality and security

Perceived alignment of quality and security priorities among teams remains uneven. While about a third consider them to be “somewhat aligned” and another third feel “strongly aligned,” the final third reports minimal-to-no alignment or a lack of insight altogether. Despite the largely positive response here, only a third reporting “strongly aligned” still indicates that most teams have a long way to go for such a critical alliance of priorities.

Misalignment between quality and security priorities creates gaps in risk visibility and release readiness decisions, leading to delayed releases and increased defect slippage. Additionally, teams with poor alignment between quality and security risk duplicating efforts or missing critical vulnerabilities due to a lack of cross-functional coordination and visibility. While much of the work in breaking down quality and security silos is done through internal culture shifts, unified quality and security platforms and workflows also help break through those walls.

- One-third of teams report minimal to no quality and security alignment
- Misalignment here leads to risky releases and wasted time
- Unified tools and workflows can bridge the gap

How well does your organization balance speed vs. risk today?



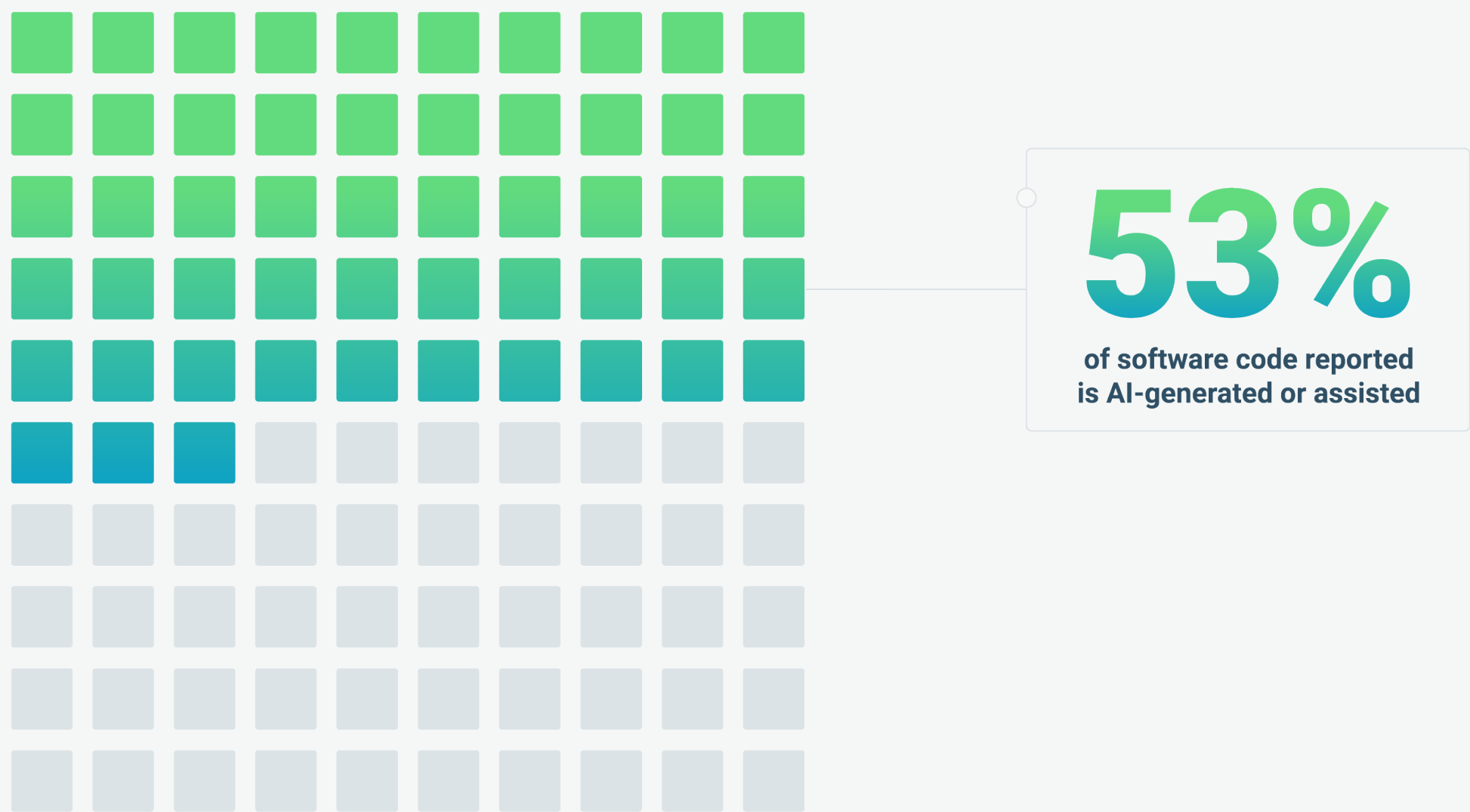
Organizations broadly balance speed and risk, but aggressive teams favor speed

When asked to rate how their organization balances speed vs. risk, the results were competitive. 36% report an even balance between the two, and “slightly favors speed/slightly favors risk management” are neck-and-neck with 23% reporting each. The “strongly favors” are where speed pulls ahead, with 14% vs. the 4% who strongly favor risk management. This indicates that teams are largely taking a balanced approach, but for those who move aggressively, speed is the priority.

Accepting risk to maintain delivery velocity and taking a reactive vs. proactive approach to risk management only harms organizations in the long run, giving them releases that might be faster but are also more likely to contain defects and vulnerabilities. Organizations can ease the speed/risk friction by [shifting security left](#), embedding throughout the SDLC rather than positioning it as a last step (and perceived bottleneck) before release.

- Trends are fairly balanced overall, but aggressive teams favor speed
- Speed-first releases increase downstream defect costs
- Indicates a need for real-time risk assessment embedded in CI/CD

What percentage of your software code is AI-generated or assisted?



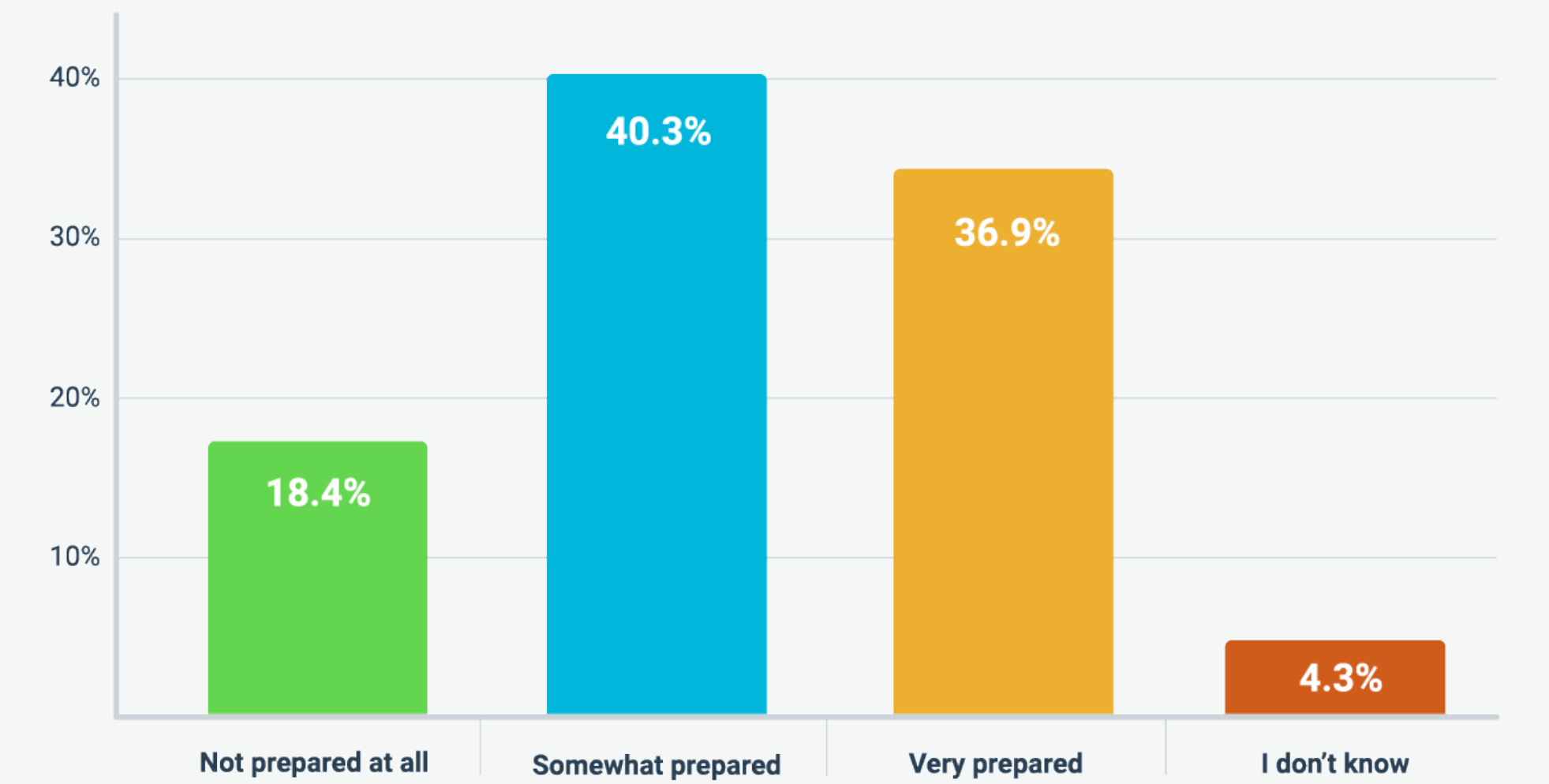
For the first time, respondents report that on average more than half of their code is touched by AI

In one of the biggest shifts in software development in recent years, respondents estimate that, on average, 53% of their organization’s code is now AI-generated or AI-assisted. This has major implications across the SDLC, reshaping how teams think about quality, security, and release readiness.

AI-generated code can often be produced faster, more consistently, with better documentation than its overburdened human counterparts, but it also introduces new risks—like hallucinations, insecure patterns, and code inconsistencies—that the QA and security processes that worked in 2023 may not be equipped to handle now. Additionally, the dramatically higher code volume and velocity enabled by AI are increasing demands on QA, and traditional test coverage strategies may not scale to AI-driven output.

- Among respondents’ companies, AI now touches an average of 53% of all code written
- Existing QA processes weren’t built for AI-generated code
- AI code volume is outpacing QA team capacity

How prepared is your organization to face emerging quality operations trends in 2026?



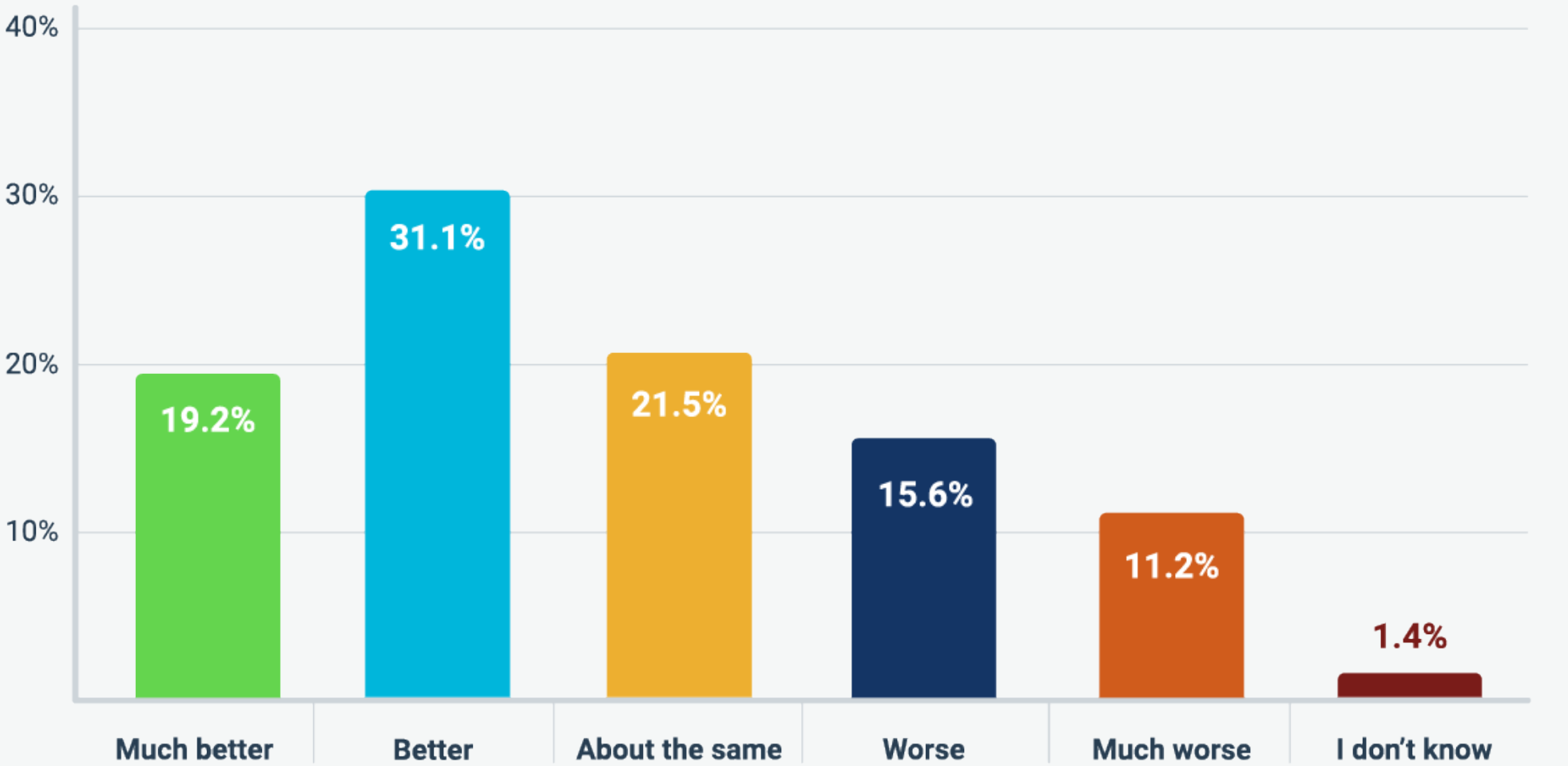
Nearly 1 in 4 organizations feel unprepared or uncertain

As evidenced in the previous question, the future is coming fast, and a significant number of teams aren't optimistic about their ability to keep up. When asked to rate how prepared they feel to face quality operations in light of emerging trends this year, nearly a quarter of teams don't feel prepared at all, or do not know.

On the other end of the spectrum, 37% feel very prepared—but while high confidence in handling the demands of AI and associated trends is promising on the surface, reported confidence levels may not align with actual technical readiness for AI-driven development. Many teams are underestimating the operational impact of emerging trends, failing to account for how quickly coding velocity and AI-induced code vulnerabilities may accumulate across the SDLC. This indicates a gap in future-proofing QA and security processes, and a clear demand for guidance, frameworks, and modern tooling built specifically for the AI era.

- Nearly a quarter of teams feel unprepared or unsure for 2026
- Confidence may mask the underestimation of AI's impact
- All teams need modern guidance, frameworks, and tooling

Do you think that overall software quality across the industry will get better or worse in 2026?



Despite uneven 2026 readiness, over half of teams expect software quality to improve

With results that closely mirror those of the previous question, about a quarter of respondents predict that overall software quality will worsen in 2026, while half predict it will improve. This further underscores the industry split between confidence and concern in future quality outcomes and the industry’s collective ability to keep up with the uncertainty of the modern era.

Mixed levels of confidence across the board in these areas of rising complexity are creating an undercurrent of uncertainty that’s rippling through the industry, reinforcing the urgency to adopt modern quality engineering approaches and fast. Periods of rapid change like these act as a pressure cooker, and organizations that take a proactive role in modernizing quality and security will stand to outperform their peers.

- Quality outlook mirrors team preparedness trends
- The industry is split between confidence and concern
- Teams that don’t modernize risk falling behind proactive peers

Global Quality Trends Summary

The data across this section tells a clear story: the industry is accelerating, but not always in the right direction. Release cadence varies widely across teams of all sizes, quality and security alignment remains inconsistent, and the speed-versus-risk balance is tipping toward speed in ways that create compounding risk downstream.

At the same time, there are clear signals of a maturing mindset. Teams are deprioritizing cost control in favor of quality, security, and compliance. AI is being taken seriously as both an opportunity and a source of new risk. And the majority of practitioners, while cautious, are heading into 2026 with a degree of optimism about the industry's direction.

For engineering teams, the implication is straightforward: the macro conditions are not getting simpler. Complexity is rising, AI is accelerating development, and the margin for reactive quality practices is shrinking. Organizations that build proactive, integrated approaches to quality and security will be the ones best positioned to lead.

- 03 - Key Insights
- 04 - Global Quality Trends
- **05 - QA Testing Insights**
- 06 - Software Security Insights
- 07 - QA and Security Convergence
- 08 - Survey Details and Methodology
- 09 - Conclusion
- 10 - About Sembi and Portfolio

QA Testing Insights Introduction

Testing has always been the backbone of software quality, but the backbone is under more strain than ever. As development accelerates, architectures grow more complex, and AI generates code at a scale no human team could match, QA teams are being asked to do more, faster, with resources that haven't kept pace with demand.

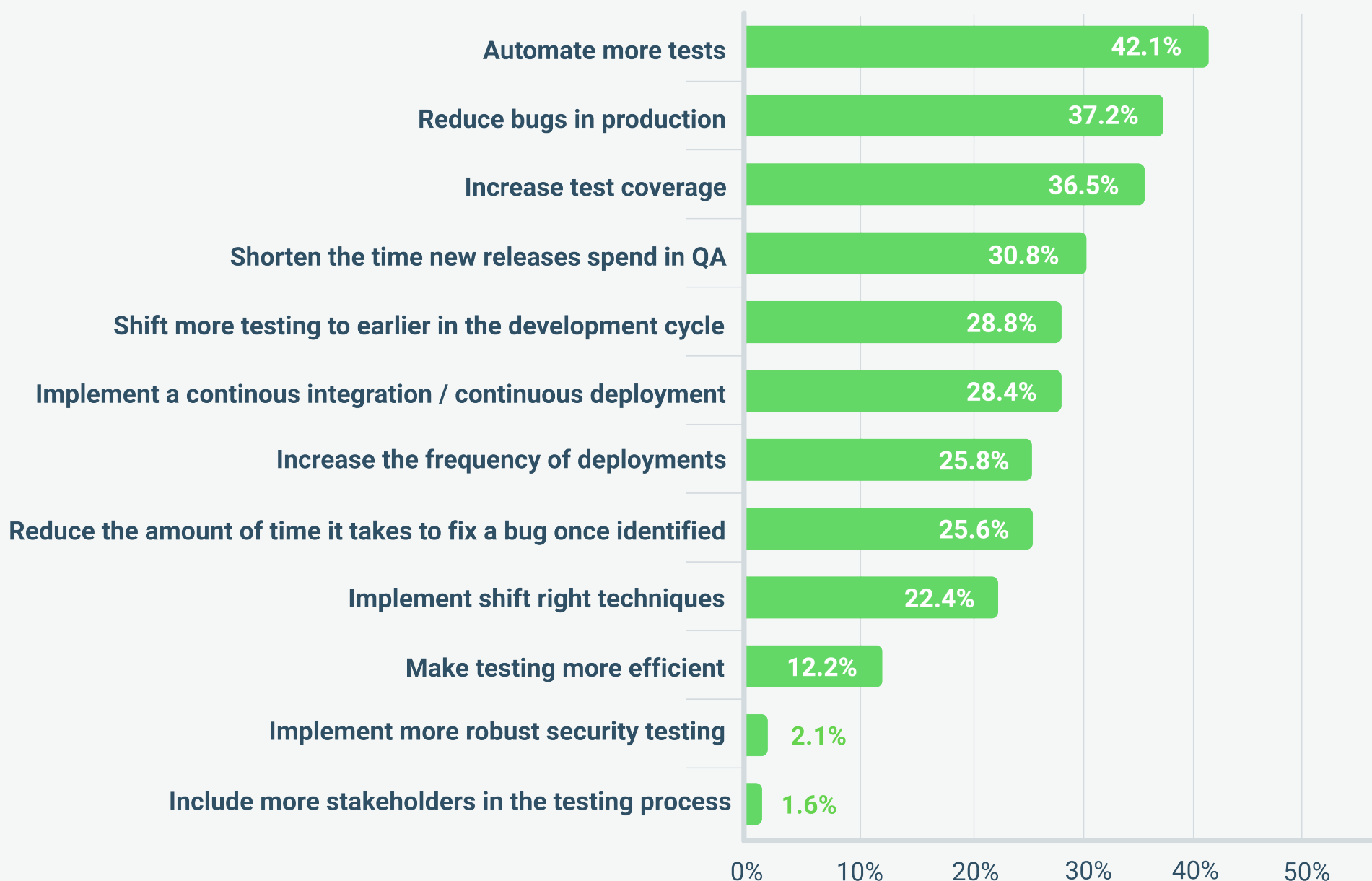
This section takes a ground-level look at the state of QA testing in 2026. It examines five interconnected dimensions of QA health:

- **QA Strategy, KPIs, and Priorities:** Where teams are focusing their energy and how they are measuring success
- **QA Practices and Maturity:** The testing types, processes, and satisfaction levels that define the current state of quality
- **QA Tools, Automation, and Integration:** How well teams are equipped, automated, and connected to DevOps pipelines
- **AI in QA:** How artificial intelligence is being adopted, what impact it is having, and what is getting in the way
- **QA Team Staffing and Resources:** The people behind the process and whether organizations have enough support to meet growing demand

What emerges is not a story of failure, but of friction. QA teams are motivated, becoming more sophisticated, and broadly aligned on the right priorities, but skill gaps, tooling fragmentation, CI/CD integration challenges, and the sheer volume of AI-generated code are all working against them. The gap between ambition and execution is where the most important opportunities lie.

As systems grow more complex and delivery cycles accelerate, QA teams are making deliberate choices about where to invest—automation, coverage, or defect prevention—while navigating persistent challenges that slow progress. This section examines the top QA priorities for the next 12 months alongside the obstacles teams face today, including end-to-end testing complexity, automation maturity, and gaps in organizational alignment.

What are your team's top 3 QA testing priorities for the next 12 months?



Automation leads QA priorities, followed by reducing production bugs and increasing coverage

Teams are prioritizing investments that support consistency and broader quality control. Automating more tests is the top priority, closely followed by efforts to reduce production bugs and increase test coverage.

This signals a move away from reactive or heavily manual processes toward more proactive and scalable quality strategies. Rather than focusing on speed, teams are placing greater emphasis on prevention and consistency. While goals like faster deployments and shorter QA cycles remain relevant, they appear to be secondary to quality, coverage, and prevention. Organizations prioritize the foundations needed for sustainable velocity and release confidence without compromising product quality.

- **Automation is a foundational priority**
- **Preventative efforts outweigh reactive ones**
- **Scale and consistency drive QA goals**

What are your top 3 QA testing challenges today?



End-to-end testing, time constraints, and automation maturity emerge as top challenges

Challenges are spread across multiple areas, with [end-to-end testing](#), time constraints, and automation maturity emerging as the most frequently cited, though not by a wide margin. Together, these issues reflect the growing strain that modern development environments place on traditional testing approaches.

As architectures become more complex, end-to-end testing is harder to maintain and scale effectively. At the same time, teams are under pressure to keep up with increasing delivery velocity, limiting the time available for thorough validation. Automation plays a dual role: it is recognized as essential, yet remains difficult to mature and implement consistently. Layered on top of this, gaps in organizational alignment, such as unclear requirements or ownership, continue to create friction, making it harder for teams to execute.

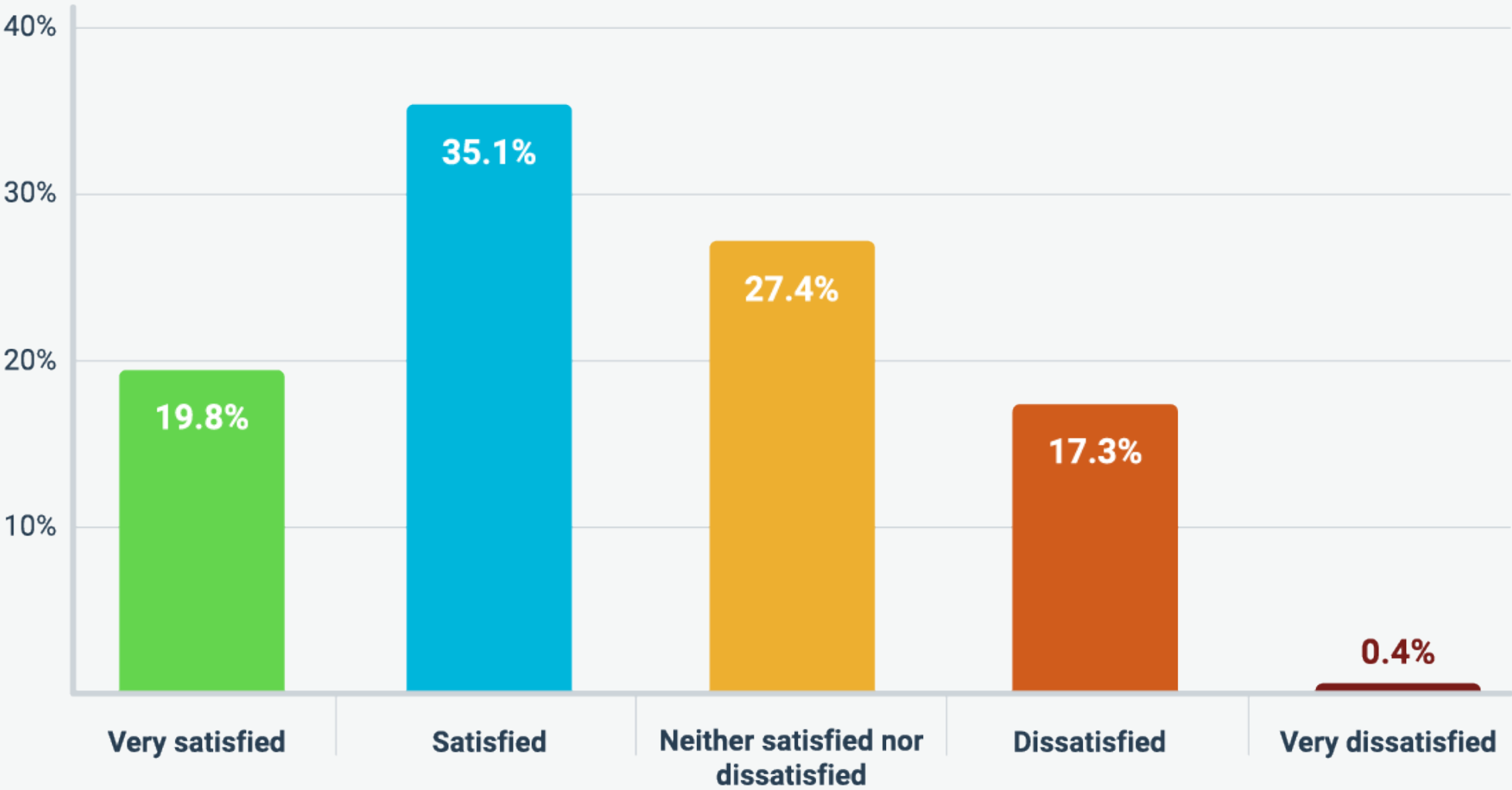
- Complexity challenges traditional testing models
- Automation is critical, but not yet mature
- Alignment gaps create execution friction

QA Practices and Maturity

QA practices continue to evolve as teams work to improve how testing is executed and scaled across the development lifecycle. While many organizations have established core processes, the level of maturity can vary widely, impacting everything from efficiency and consistency to the overall effectiveness of quality efforts. As demands on QA grow, understanding how teams assess their current processes and what practices they rely on becomes increasingly important.

This section explores how satisfied teams are with their existing QA processes and what that reveals about overall maturity. It also examines which types of testing are most commonly performed, highlighting where teams are focusing their efforts today. Together, these insights provide a clearer picture of how QA is being implemented in practice, from foundational testing approaches to more advanced and specialized methods. The data sheds light on where processes are working, where gaps remain, and how teams are progressing toward more integrated and mature quality strategies.

How satisfied are you with your current QA process?



Nearly half of QA teams are neutral or dissatisfied with their current QA process

While a slight majority of teams report satisfaction with their current QA processes, a substantial portion remain neutral or dissatisfied. Some organizations may be operating in a “good enough” state where processes function, but are not fully optimized. Neutral responses, in particular, may indicate underlying friction that has not yet translated into explicit dissatisfaction. These gaps stem from fragmented toolchains and inefficient workflows.

As QA becomes more complex and integrated into the broader development lifecycle, these inefficiencies become harder to ignore. Overall, the findings point to a clear opportunity for teams to improve how QA is structured, connected, and executed—and for platforms that streamline and unify QA workflows.

- **Many teams are operating below their optimal level of QA satisfaction**
- **Neutral sentiment signals underlying friction**
- **Fragmented workflows compound as QA gets more deeply embedded in the dev lifecycle**

Which testing types do you currently perform?

% share of selected responses

Functional testing	10.6%
Regression testing	9.7%
End-to-end testing	9.3%
Smoke testing	8.6%
Integration testing	8.5%
Exploratory testing	8.0%
User acceptance testing (UAT)	7.3%
Load/performance testing	6.0%
Security testing (SAST, DAST, IAST)	5.0%
Ad-hoc testing	5.5%
Risk-based testing	4.9%
Accessibility testing	4.3%
FDA GxP testing	4.3%
Context-driven testing	3.7%
Usability testing	2.7%
Compliance testing	1.7%
Other	0.1%

Core testing processes dominate, with functional, regression, and end-to-end testing in the lead

Core testing types, like [functional](#), [regression](#), and [end-to-end testing](#), remain the most widely adopted across teams, with more specialized areas like performance, security, and compliance trailing behind. This distribution suggests that QA efforts are still largely centered on validating functionality, rather than addressing broader dimensions of quality such as performance, security, or compliance. The relatively low adoption of security and compliance testing also highlights a persistent gap between QA and security workflows.

At the same time, more advanced approaches like risk-based or context-driven testing appear to be less established, pointing to varying levels of maturity across teams. Overall, teams seem to be optimizing for breadth of coverage, but not necessarily for depth, prioritization, or integration across different quality domains. There is a clear opportunity to unify testing strategies across functional, performance, and security domains into a more integrated model.

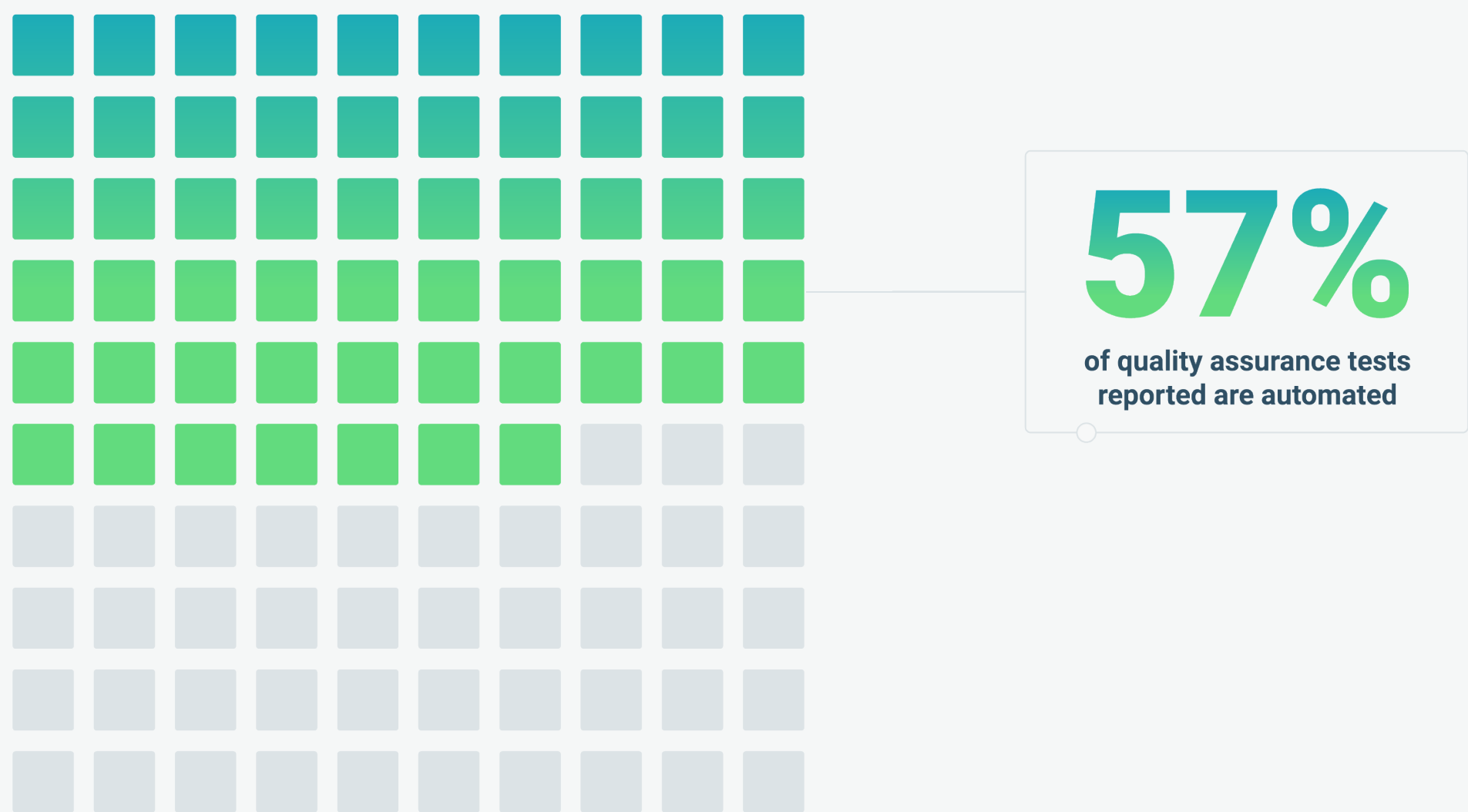
- Functional testing remains the primary focus
- Security and compliance lag behind
- Depth and integration remain limited

QA Tools, Automation, and Integration

QA tooling and automation are critical enablers of modern software delivery, but their effectiveness depends heavily on how well they are implemented and connected. While many teams have invested in automation and a growing set of tools, the level of maturity and integration varies widely—shaping how efficiently QA can support fast, reliable releases. As systems and pipelines become more complex, adding tools is no longer enough; how they work together is just as important as what they do individually.

This section explores the current state of automation and integration within QA environments. It looks at how much of the testing process teams have automated, providing insight into overall automation maturity and remaining gaps. It also examines how well QA tools are integrated into DevOps workflows, highlighting the extent to which teams can achieve end-to-end visibility and streamlined feedback loops. Together, these insights reveal how effectively teams are using tooling to scale quality, where fragmentation is holding them back, and what it takes to build more connected and efficient QA ecosystems.

What percent of your QA tests are automated?



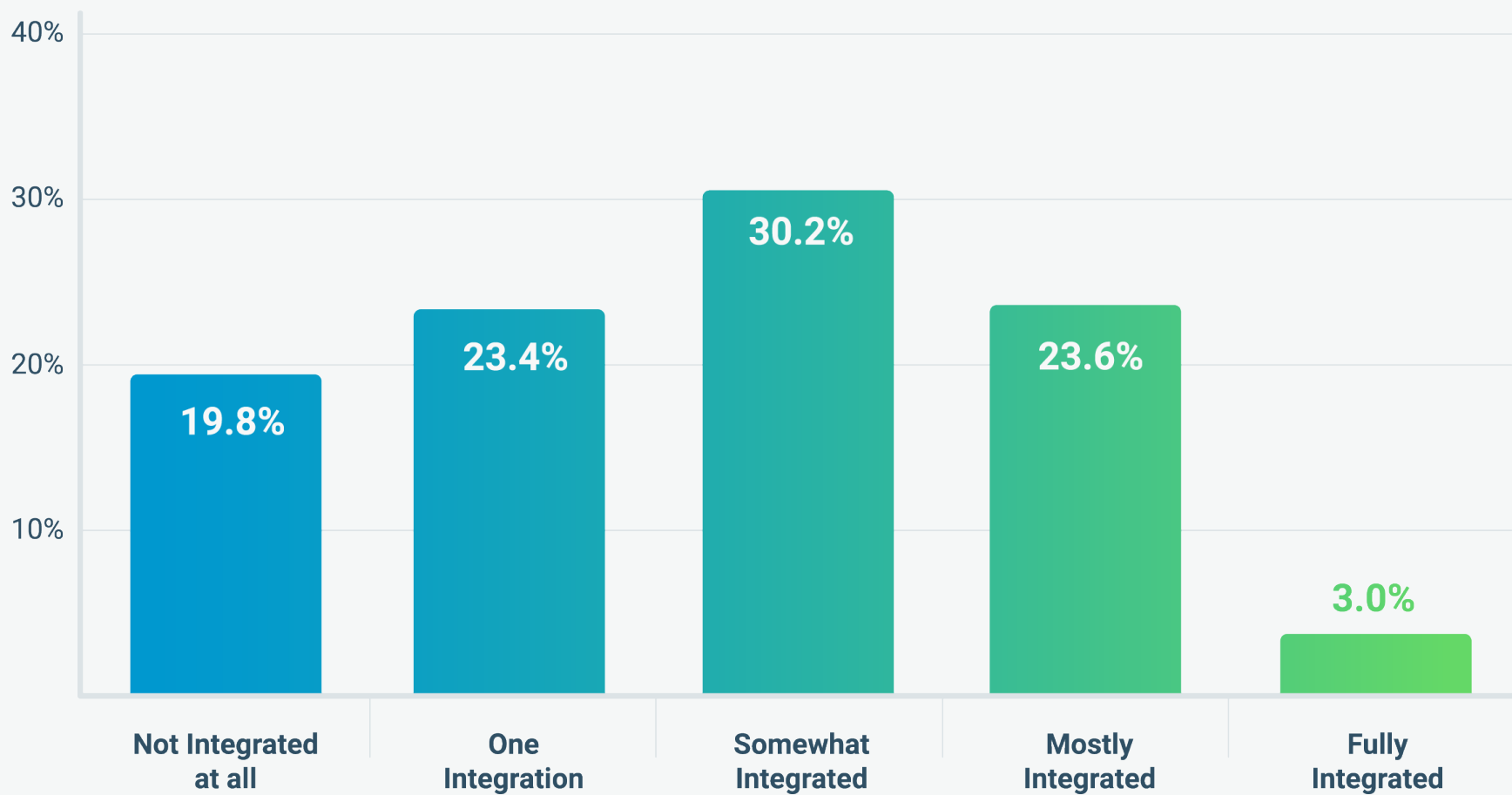
Teams report that a little over half of tests are automated, signaling progress without full maturity

On average, 57% of tests are automated, indicating that most teams have made meaningful progress in adopting automation. However, this also suggests that many are operating in a middle stage of maturity, where automation is established but not yet comprehensive.

The remaining reliance on [manual testing](#) can introduce challenges in scalability, consistency, and speed—particularly as systems grow more complex and release cycles accelerate. At the same time, the data reinforces that increasing automation alone is not enough. To deliver real impact, automation strategies need to be applied thoughtfully, with a focus on risk, prioritization, and maintainability. Overall, teams appear to be advancing in automation, but still have work to do in making it more effective and scalable.

- **Automation maturity remains at a mid-level**
- **Manual testing can limit scalability and consistency**
- **Smarter and risk-based automation is needed**

How integrated are your QA tools with DevOps?



Only a quarter of teams report mostly or fully integrated QA tools within DevOps pipelines

Just 26% report strong integration between their QA tools and DevOps workflows, while the majority remain partially integrated or siloed. This lack of connectivity limits end-to-end visibility and disrupts feedback loops, slowing decision-making and reducing the ability to respond quickly to issues. Fragmented tools also create delays and duplicated effort, with teams relying on manual handoffs between QA tools and pipeline systems.

As a result, integration gaps can impact both release speed and confidence in fast-paced delivery environments. Overall, the findings highlight a continued reliance on disconnected tools and point solutions, reinforcing the need for more unified, ecosystem-driven approaches to QA and DevOps integration.

- Integration remains limited across teams
- Tool fragmentation slows feedback loops
- More connected QA and DevOps ecosystems are needed

“Analytics, dashboards, deep integrations—all those things are building blocks for this thing we’ll call release agility.”

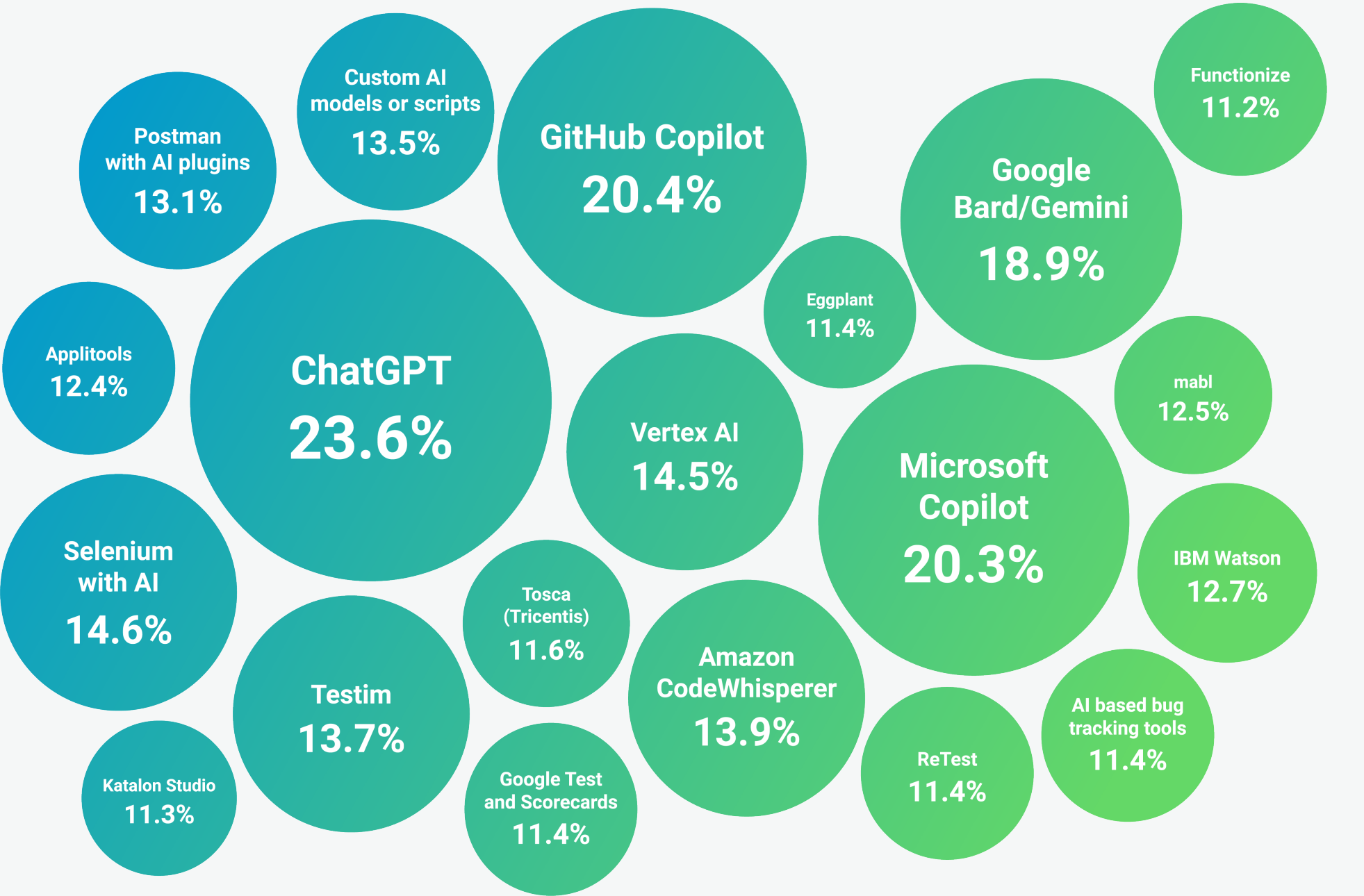
- Chris Faraglia, Sembi Lead Solution Architect (Testing Brands)

AI in QA

AI is becoming a bigger part of QA, but most teams are still working out how to use it effectively. Across the industry, organizations are trying a mix of AI tools to improve testing speed, coverage, and efficiency. While the opportunity is clear, the results are uneven. Some teams are seeing real benefits, while others are dealing with challenges like integration complexity, data privacy concerns, and rising costs.

This section examines how QA teams are using AI today, the impact it is having, and what is getting in the way of broader success. It looks at which tools are most common, how teams describe the results of AI-driven testing, how AI-generated code is increasing testing demand, and what barriers are slowing adoption. Overall, the data suggests that AI is becoming more common in QA, but consistent value still depends on how well it is implemented.

Do you currently leverage AI tools in your existing QA testing processes? If yes, which?



No single AI tool dominates QA, with ChatGPT leading adoption

The absence of a dominant AI tool in QA is itself a signal. Unlike other technology shifts where one or two platforms quickly captured most of the market, AI adoption in testing is sprawling and experimental. Teams are reaching for whatever is closest—general-purpose LLMs, IDE assistants, purpose-built testing plugins—rather than following an industry-wide playbook. That reflects how early this transition still is: organizations know AI belongs in their QA workflow, but most haven't yet determined where it creates the most durable value.

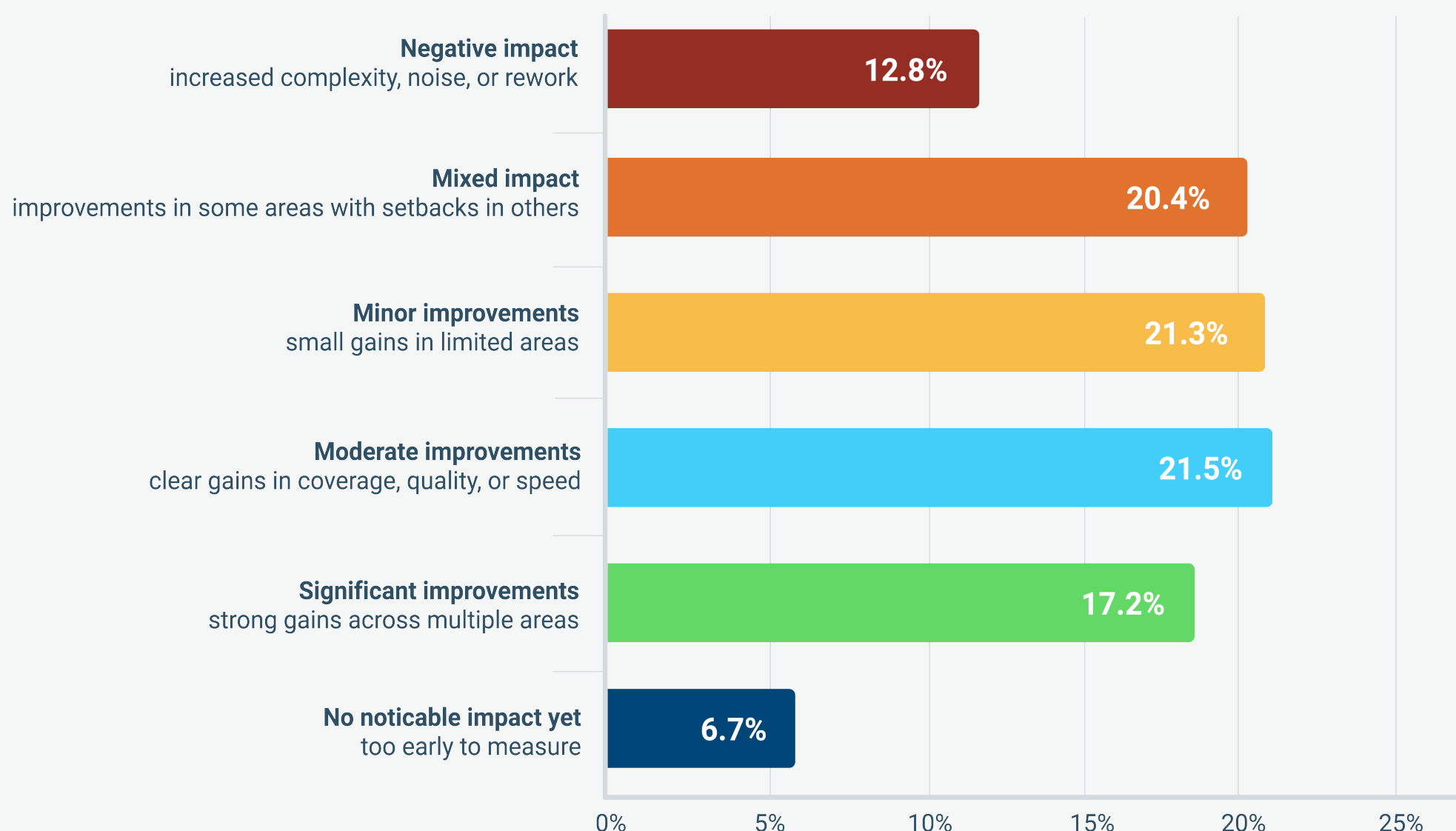
This matters because fragmented adoption can create new problems alongside new efficiencies. Different teams may use different tools for different tasks, which can lead to siloed workflows, inconsistent practices, and more complexity to manage. At the same time, the spread of adoption shows strong interest in AI across the testing lifecycle. The bigger opportunity now is not just using more AI, but using it in a way that is more connected, consistent, and easier to scale.

- **AI adoption is broad but fragmented**
- **Most teams are still experimenting**
- **Standardization has not yet emerged**

“The real breakthrough won't be ‘AI writes test cases’ or ‘AI replaces testers.’ It will be AI that understands the system, the risks, and the change impact, then tells QA what to test, when, and how deep.”

- Survey Respondent

How would you describe the impact of AI-driven testing on QA coverage, quality, and/or speed?



Most teams report minor, mixed, negative, or no noticeable impact from AI-driven testing

The results here push back on the more triumphant narratives around [AI in testing](#). Despite heavy investment and widespread adoption, teams are not reporting a step-change in quality outcomes. The distribution of responses—spread across incremental improvement, mixed results, and even negative impact—suggests that AI testing tools perform unevenly across environments, use cases, and levels of implementation maturity. For every team seeing meaningful gains, there is another absorbing the overhead of AI integration without a clear return.

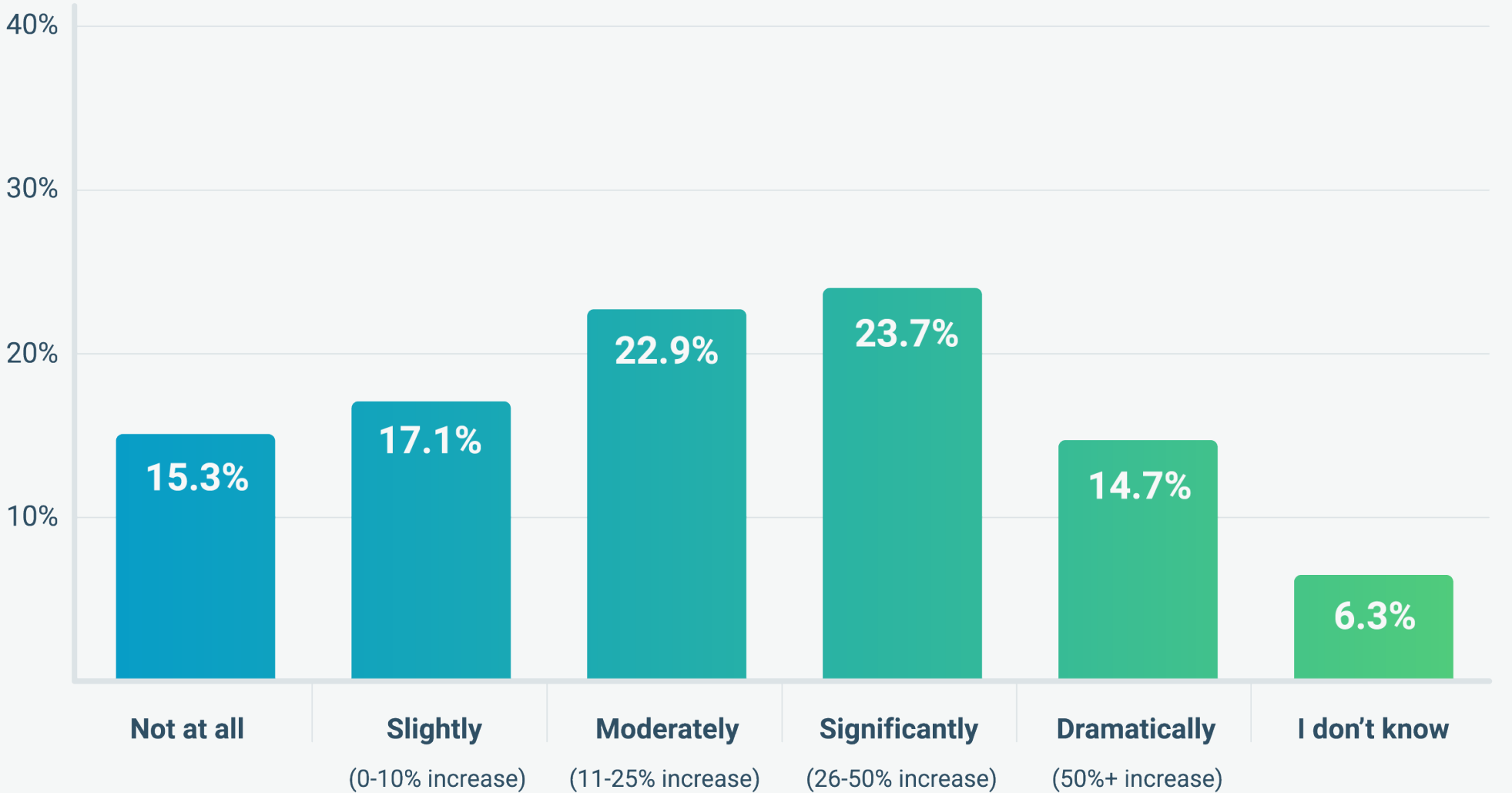
Adding AI to the testing process does not guarantee better outcomes alone. While 60% of teams report at least some improvement, only 17.2% say those gains have been significant. This suggests that AI is creating value in specific areas, but not yet in a way that is broad, consistent, or transformative across QA. The next opportunity is not just wider adoption, but a more thoughtful approach to applying AI in ways that deliver more reliable results.

- **Most gains are still incremental**
- **Outcomes remain uneven across teams**
- **Significant impact is still limited**

“AI is showing what is wrong within QA—implementing it on top of an unstable architecture, and AI really highlights the cracks in their systems.”

- Patrícia Duarte Mateus, TestRail Solution Architect

To what extent has AI-generated code increased QA testing demands?



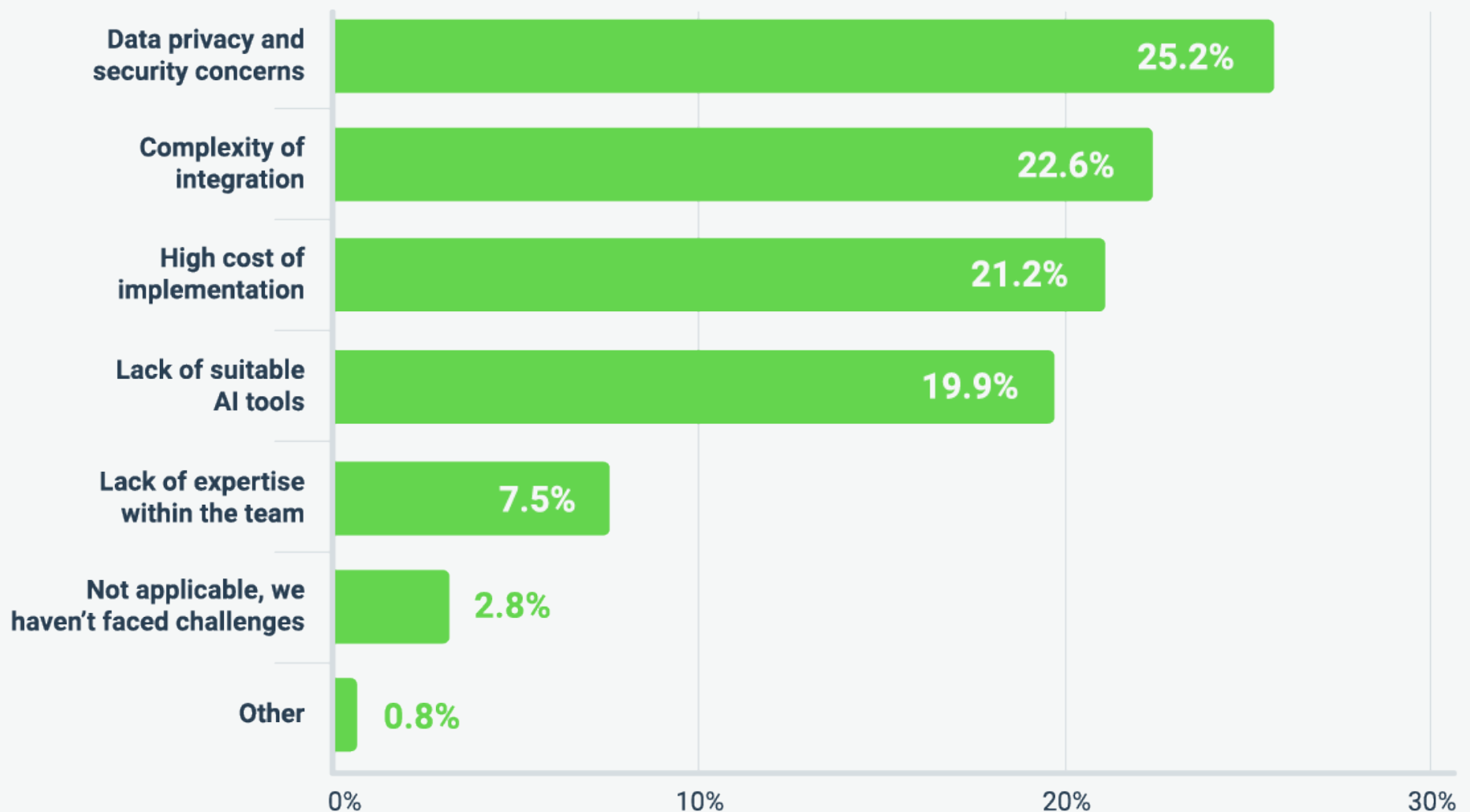
Most teams report moderate to dramatic increases in QA demand from AI-generated and AI-assisted code

There’s an uncomfortable math problem at the heart of AI-assisted development: the same technology that helps engineers write code faster is also generating more code that needs to be tested. QA teams didn’t sign up to become the safety net for AI-generated output, but that is the role they’re being asked to fill. And unlike development velocity, which AI can accelerate, testing capacity doesn’t scale automatically. The result is a widening gap between how fast code is produced and how thoroughly it can be validated.

This points to a growing gap between development speed and QA capacity. As teams use AI to produce more code, QA is left with more to validate, more edge cases to cover, and more potential issues to catch. That added workload is not matched by more time, more people, or more mature testing processes. This makes scalable automation, continuous testing, and earlier validation even more important if quality is going to keep up.

- **AI-generated code is raising QA demand**
- **QA must scale to match AI development speed**
- **Scalable automation is now a QA survival requirement**

What challenges have you faced in integrating AI into your QA testing process?



Data privacy and security concerns are the most common barriers to AI integration in QA

The barriers to AI integration in QA are more structural than technical. Privacy concerns reflect the reality that QA environments often touch sensitive data, production systems, and proprietary codebases that teams are understandably reluctant to route through external AI services. Integration complexity speaks to the fragmentation problem: most QA toolchains were not built with AI in mind, and retrofitting AI into existing pipelines requires effort that many teams can't absorb on top of their existing workload. Cost concerns, meanwhile, signal that AI's value proposition in QA has not yet been clearly demonstrated to the people controlling budgets.

These results show that interest in AI is not the problem. The bigger challenge is making AI work in a real-world QA environment. Teams need solutions that are secure, affordable, and easy enough to integrate into existing workflows. For many organizations, adoption is being slowed by the difficulty of finding tools that feel practical, scalable, and low-risk.

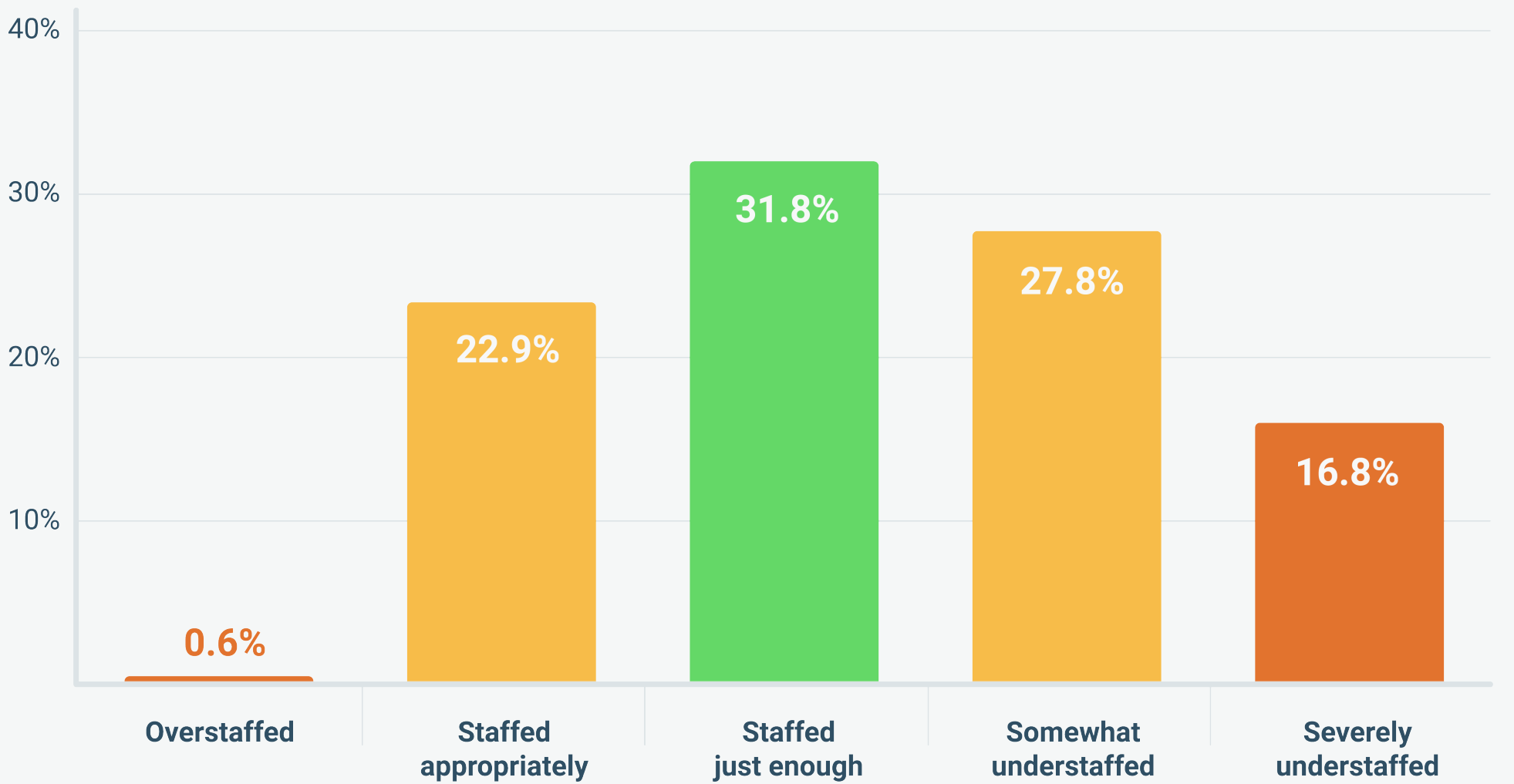
- Data privacy and security are the top barriers
- Integration remains a major hurdle
- High cost slows broader AI adoption

QA Team Staffing and Resources

QA teams are under pressure to do more than ever. They need to support faster release cycles, test increasingly complex software, and maintain quality, often without increasing their staff. At the same time, [automation and AI](#) are raising expectations for how much QA can do. That makes staffing and resources a key part of the picture. Do teams have the people they need, or are they expected to close the gap with tools and efficiency alone?

This section looks at how QA teams are staffed today and how organizations expect those teams to change over the next 12 months. It explores whether teams feel they have enough people to meet their goals, and whether leaders expect QA headcount to grow, remain the same, or shrink. Together, these answers show how organizations are thinking about the future of QA and whether that future will be built through more people, more technology, or a mix of both.

Please describe the staffing level of your QA team to accomplish your tasks and goals.



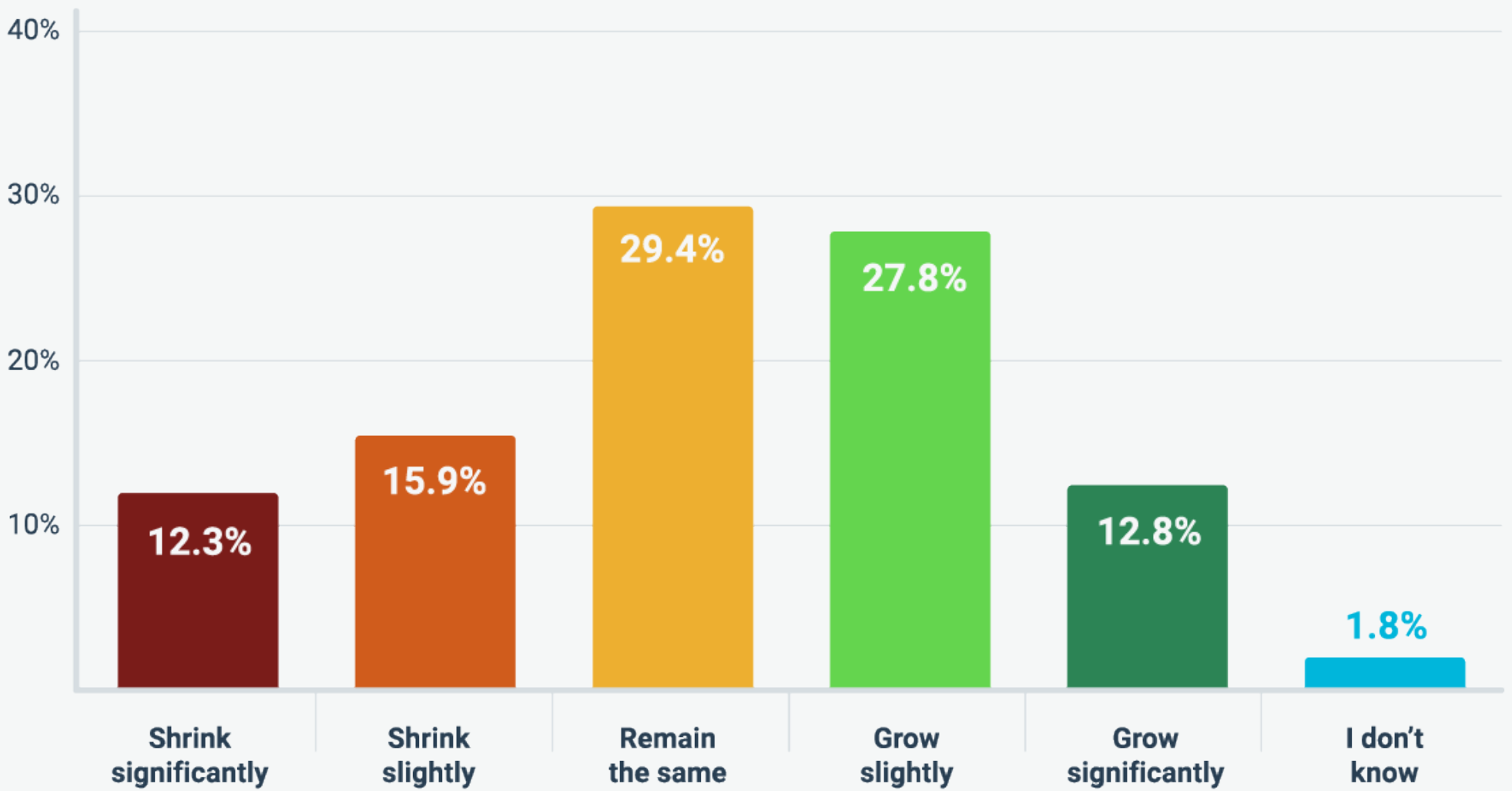
QA teams are struggling to secure adequate resources for rising demands

Staffing in QA has always been tight, but the data suggests it has reached a point where ‘tight’ is no longer sustainable. The combination of AI-generated code volumes, accelerating release cadences, and growing automation responsibilities has raised the baseline of what a QA team is expected to handle, without a corresponding increase in staffing. What’s particularly telling is how few teams sit on either extreme: almost no one is overstaffed, and only a small minority feel genuinely well-resourced. The overwhelming middle is a workforce stretched to capacity or beyond it.

This suggests that many QA teams are operating under pressure to do more with fewer resources. Even teams that are staffed just enough may not have the flexibility to absorb rising testing demands, tighter release timelines, or new responsibilities tied to automation and AI. That kind of pressure can increase the risk of bottlenecks, burnout, and missed defects, especially as organizations continue asking QA to do more with limited resources.

- **Almost half (44.6%) of QA teams report being understaffed**
- **Less than a quarter of teams feel fully staffed**
- **Staffing gaps make automation and AI augmentation essential**

How do you expect the size of your QA team to change over the next 12 months?



Despite resourcing struggles, most QA teams do not expect headcount to grow

The staffing forecast for QA in 2026 is less a consensus and more a three-way split, and that fragmentation is telling. Roughly equal shares of teams expect growth, stasis, and cuts, which suggests that organizations are making very different strategic bets about whether QA headcount is an investment worth making or a cost worth trimming. Given that most teams report being understaffed, the sizable share anticipating reductions isn't just a workforce planning data point; it's a warning signal.

It's clear that QA teams are heading into the next year with very different resourcing plans. Some organizations are investing in growth, while others appear to be relying on leaner teams and greater efficiency. That uncertainty reinforces a bigger shift visible across QA: future scale may depend less on headcount alone and more on how well teams combine skilled people with automation, AI, and smarter workflows.

- The outlook for QA hiring is mixed
- Some teams expect growth, while others expect contraction
- Future scale will rely on people and tools

QA Testing Insights Summary

QA Teams Know What Good Looks Like, and Are Fighting to Get There

Automation is the goal; integration is the gap.

Teams are broadly aligned on the need to automate, reduce production defects, and expand coverage, yet the majority report QA tools that are only partially integrated with DevOps pipelines, and nearly half remain understaffed. The gap between automation ambition and automation reality is the defining tension in QA today, and closing it will require more than investment in tools alone.

AI is raising the ceiling and the stakes simultaneously.

AI-generated and AI-assisted code is straining QA demand in ways most teams weren't prepared for, with 61% reporting moderate to dramatic increases in testing workload. At the same time, [AI-driven testing tools](#) are delivering incremental rather than transformational results, held back by data privacy concerns, integration complexity, and implementation immaturity. The opportunity is significant, but so is the work required to realize it.

People remain the variable that tooling cannot replace.

Nearly half of QA teams report being understaffed, and expectations for team growth over the next 12 months are deeply uncertain. Until organizations address the talent gap alongside the tooling gap, even the best [automation strategies](#) will underperform. Teams that prioritize both efficiency gains from automation and AI as well as scaling their headcount will have a meaningful advantage in the year ahead.

Taken together, the data from this section paints a portrait of a QA discipline that is more strategically aware than ever—but still navigating significant structural headwinds. Automation, AI, and integration are the right bets, and most teams know it. The challenge is execution: closing the gaps between intent and reality in an environment that keeps raising the bar. The organizations that solve for people, process, and tooling together will define what great QA looks like in 2026.

- 03 - Key Insights
- 04 - Global Quality Trends
- 05 - QA Testing Insights
- **06 - Software Security Insights**
- 07 - QA and Security Convergence
- 08 - Survey Details and Methodology
- 09 - Conclusion
- 10 - About Sembi and Portfolio

Software Security Insights Introduction

Security has never been more visible or more vulnerable. Software systems are growing more distributed, [AI-generated code is introducing new risk](#), and the threat landscape is evolving faster than most teams can track—forcing security out of its traditional role as a final-stage checkpoint and into every layer of the development lifecycle. The pressure is real, and the data shows that many teams are still catching up.

This section examines the state of security practices in 2026, drawing on responses from 3,800 software delivery professionals.

- **Security Strategy, KPIs, and Priorities:** Where security teams are directing their focus and what they are measuring to gauge success
- **Security Practices and Maturity:** The testing types, process satisfaction, and maturity levels that define how security is being practiced today
- **Security Tools, Automation, and Integration:** How well security toolchains are built out, automated, and connected to DevOps pipelines
- **AI in Security:** How artificial intelligence is being applied to security workflows, what it is delivering, and where it is falling short
- **Security Team Staffing and Resources:** The people responsible for keeping software secure, and whether organizations are giving them what they need

What the data reveals is a discipline that is becoming more aware of its own gaps. Security teams understand the stakes, broadly agree on the priorities, and are actively investing in tools and automation. But awareness alone is not enough. Fragmented toolchains, persistent staffing shortages, high false positive rates, and late involvement in the development process are all limiting effectiveness in ways that demand more than incremental improvement.

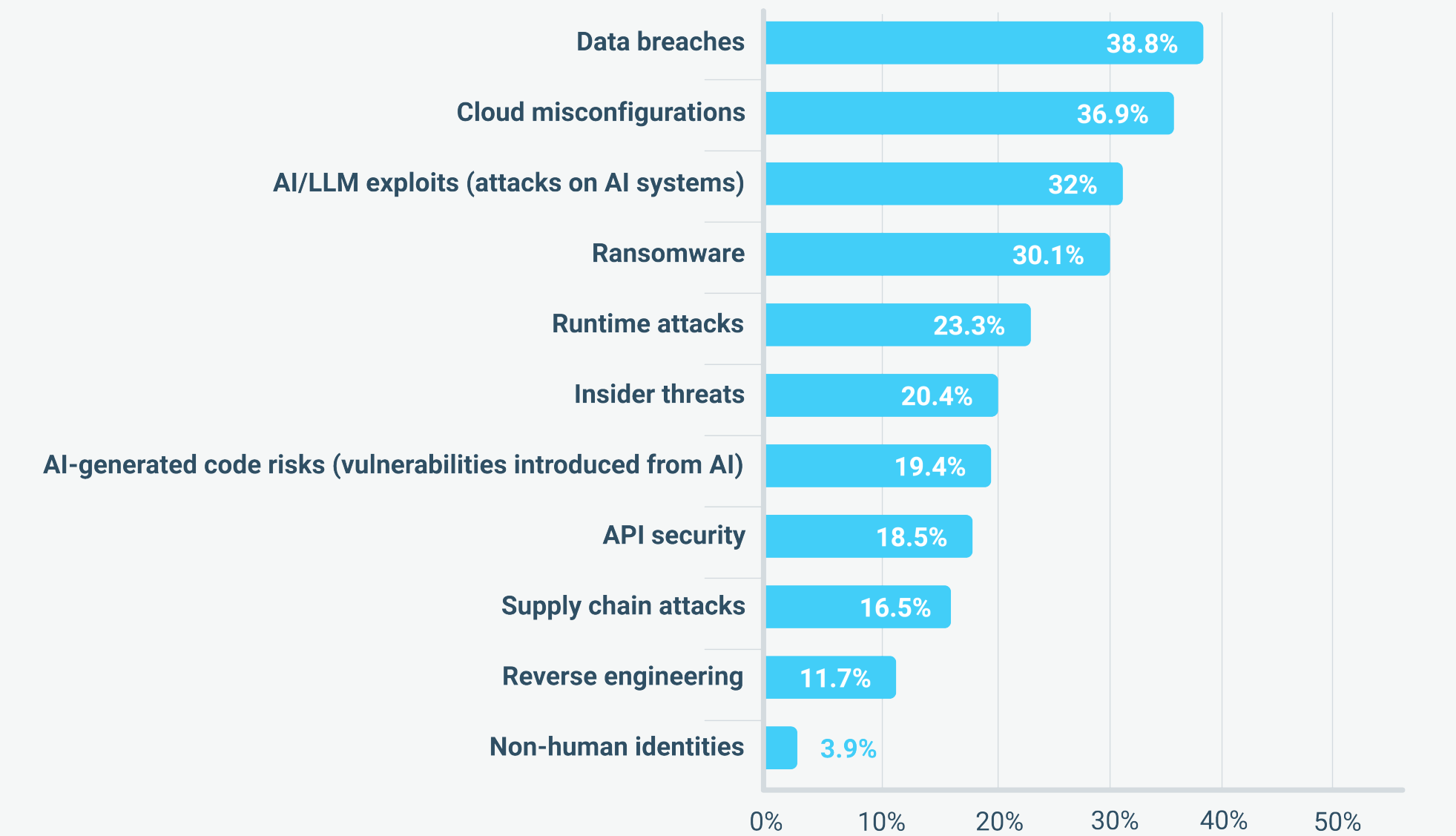
The following subsections break down where security stands today—and what it will take to move from reactive to resilient.

Security Strategy, KPIs, and Priorities

Security strategy is shifting as organizations respond to expanding attack surfaces and increasing system complexity. No longer confined to perimeter defense, efforts are now centered on risks introduced by cloud-native architectures, distributed systems, and AI-driven development. At the same time, teams are under pressure to embed security earlier in the lifecycle without slowing delivery.

This creates tension between intent and execution. Organizations recognize the need to prioritize prevention, accelerate remediation, and align more closely with development workflows, but operationalizing these priorities remains difficult. Complexity, capacity constraints, and integration gaps continue to limit progress. The data in this section highlights how security teams are navigating that gap, where they are focusing their efforts, what is holding them back, and which levers will most meaningfully improve their security posture.

What are your top 3 security priorities for the next 12 months?



Data breaches, cloud misconfigurations, and AI risks are top concerns

Security priorities demonstrate alignment with the realities of today's software environments. Data breaches (38.8%) are the top concern, closely followed by cloud misconfigurations (36.9%) and AI/LLM-related threats (32%). This signals a shift away from traditional perimeter-based risks toward vulnerabilities introduced by cloud infrastructure and AI-driven development. Rather than focusing on detection and reactive response, organizations are setting their sights on proactive prevention and posture management by securing dynamic environments, reducing configuration risk, and addressing emerging AI threats earlier.

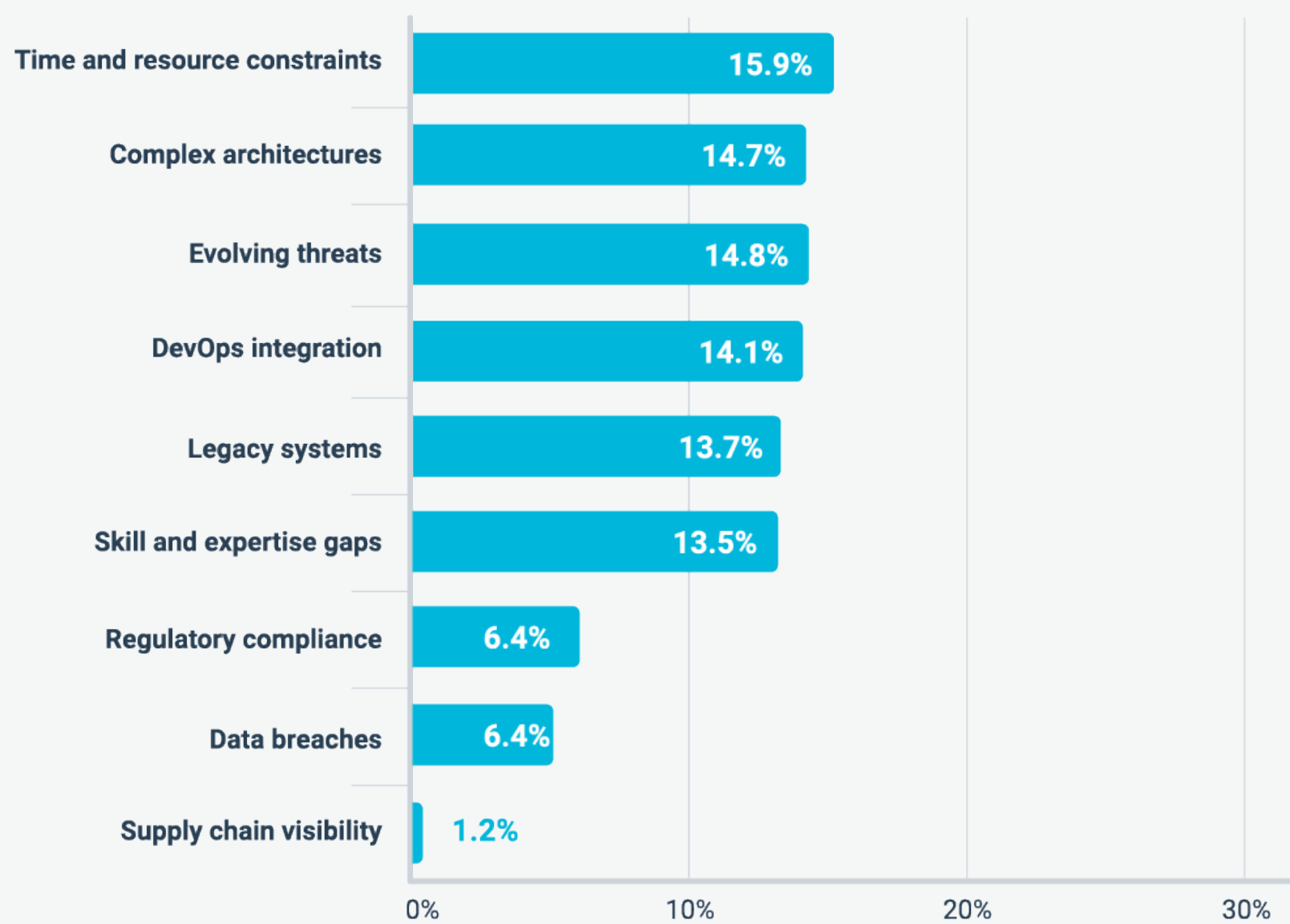
To support this shift at scale, security must be integrated into developer workflows, enabling earlier intervention and more consistent enforcement across the development lifecycle.

- **Security priorities are shifting toward cloud and AI risks**
- **AI threats are already a top-tier concern**
- **Prevention and posture management are gaining focus**

“Security is going to be big. People are using AI to create code they don't understand, can't debug, etc. There are security problems attributed back to AI.”

- Devon Jones, Ranorex Solution Architect

What are your top 3 security challenges today?



Multiple factors are compounding pressure on security teams, no one challenge stands out

Security teams are navigating a combination of capacity limitations and increasing technical complexity. The even distribution of responses suggests there is no single bottleneck; rather, multiple pressures are compounding at once.

Modern architectures and environments, including microservices architectures, cloud infrastructure, and APIs are expanding the attack surface faster than many teams can adapt. At the same time, incomplete DevSecOps integration makes it difficult to embed security into development workflows consistently. The result is a persistent gap between security intent and execution—and until organizations address capacity, architectural complexity, and integration maturity together, reactive postures will remain the default.

- Security teams are constrained by time and resources
- Architectural complexity is outpacing capabilities
- Incomplete DevSecOps integration widens security gaps

What would most improve your security posture?



Faster vulnerability remediation tops the list of ways to improve security posture

When asked what would most improve their security posture, organizations point first to faster vulnerability remediation (27.2%), followed by improved developer training (22.3%). This highlights a critical shift: the challenge is no longer just identifying issues, but resolving them quickly and consistently.

Detection capabilities may be in place, but remediation delays continue to introduce downstream risk. At the same time, the emphasis on developer training reflects the growing role engineers play in shaping security outcomes. As development and security become more closely connected, improving developer awareness and enabling earlier intervention become key levers for reducing vulnerabilities at the source.

Remediation speed is emerging as the primary constraint on improving security outcomes. As a result, developers play a central role in strengthening security, since they are closest to where issues are introduced and resolved. This makes shift-left practices critical, enabling teams to address risks earlier in the development lifecycle before they escalate. Adoption will depend on how well security integrates into the developer experience, with intuitive tools and actionable insights serving as key drivers of sustained improvement.

- **Remediation speed is a primary bottleneck**
- **Developers are central to security outcomes**
- **Shift-left practices are critical for improvement**

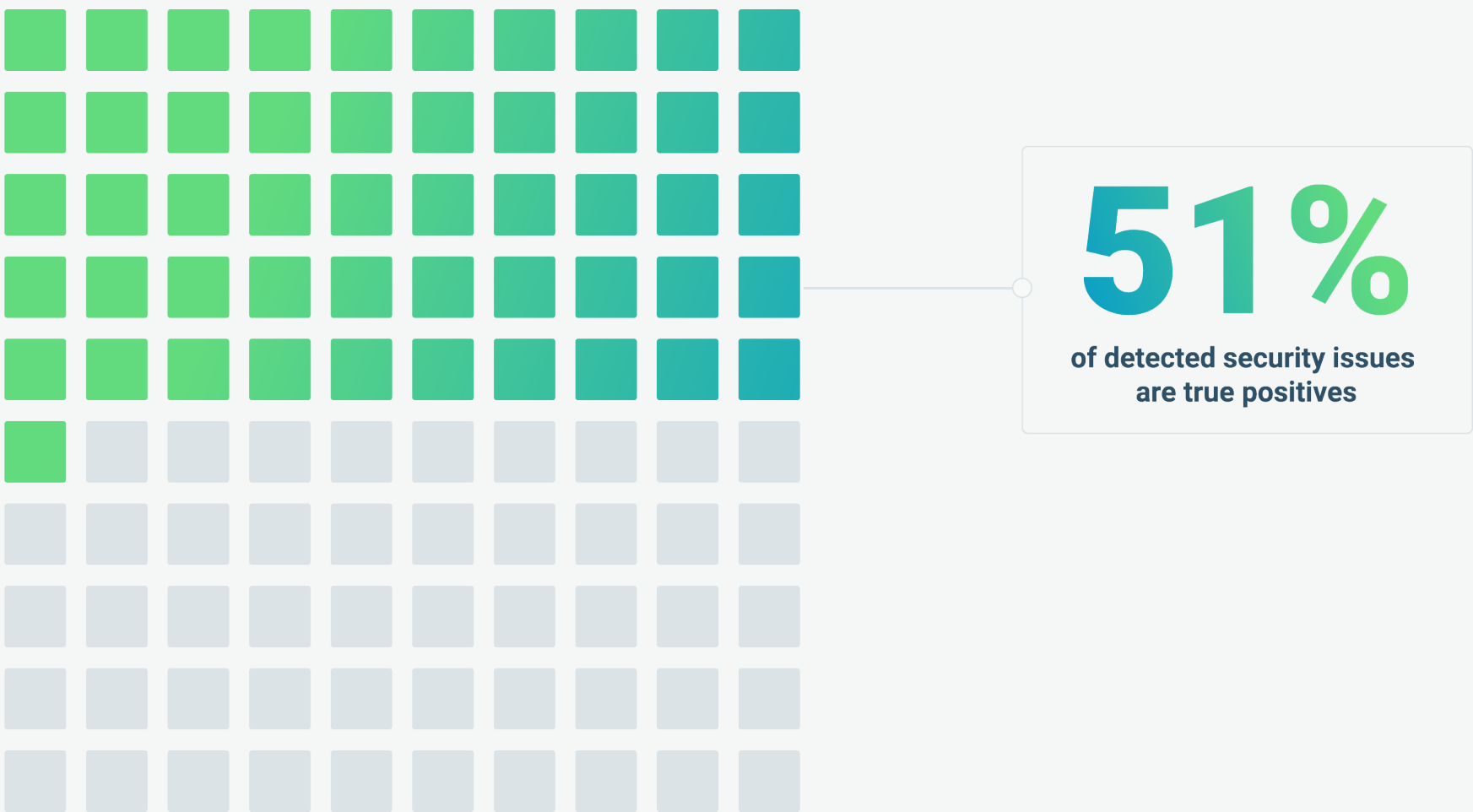
“Faster vulnerability remediation has to be more than just a process improvement—it requires developers to be more security-aware from the start.”

- John Brawner, Sembi Solution Architect (DevSecOps)

Security Practices and Maturity

Security has no one-size-fits-all solution. Teams operating across multiple environments face gaps that often go undetected, stemming from mismatched tools, overstretched teams, or an incomplete security strategy. Because security and QA are interdependent, when one lags, delays ripple across the entire lifecycle.

What % of detected security issues are true positives?



With only half of detected issues being true positives, alert noise is a critical problem

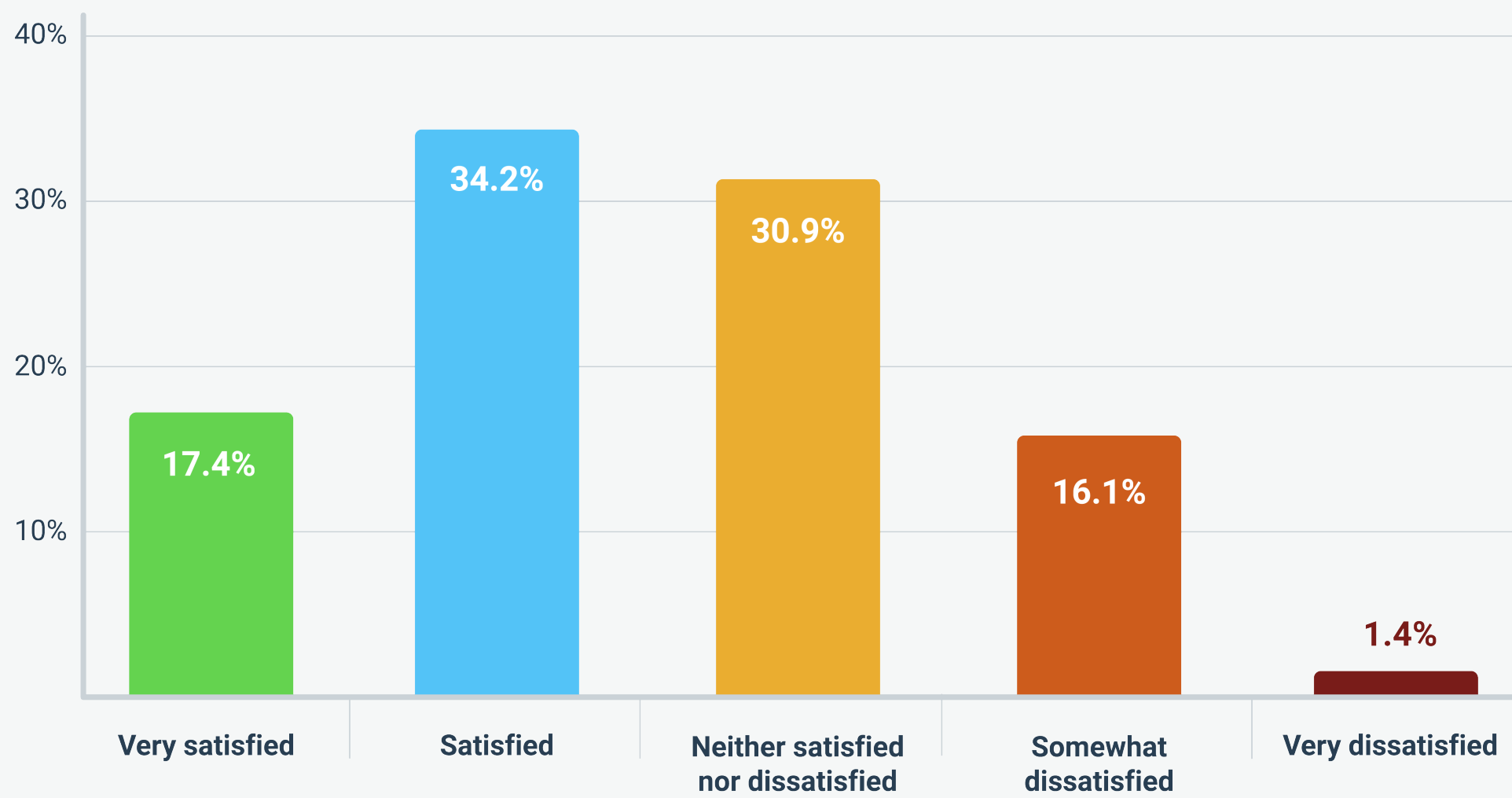
These findings demonstrate that teams are spending an enormous amount of time chasing noise rather than real threats. High false positive rates erode developer trust in tooling: when alerts can't be suppressed or tuned, the same irrelevant defects resurface on every scan. Over time, this creates alert fatigue—developers stop paying close attention, and genuine vulnerabilities slip through undetected. The operational cost compounds quickly: teams unable to mute false positives must manually triage thousands of flagged lines of code per day, diverting resources away from actual risk analysis and remediation. Trust in the tool erodes, morale drops, and the risk of a critical incident going unnoticed rises sharply. Selecting tools with flexible suppression settings and strong signal accuracy is not just a convenience; it's a security necessity.

- **High false positive rates create serious operational bottlenecks**
- **Inaccurate security threat reporting erodes developer trust and can lead to complacency**
- **Alert fatigue lets real threats slip through undetected**

“In today’s era of AI-driven attack automation, being hacked is almost inevitable. The focus of tracking should shift from ‘preventing’ to ‘restoring.’”

- Survey Respondent

How satisfied are you with your current security process?



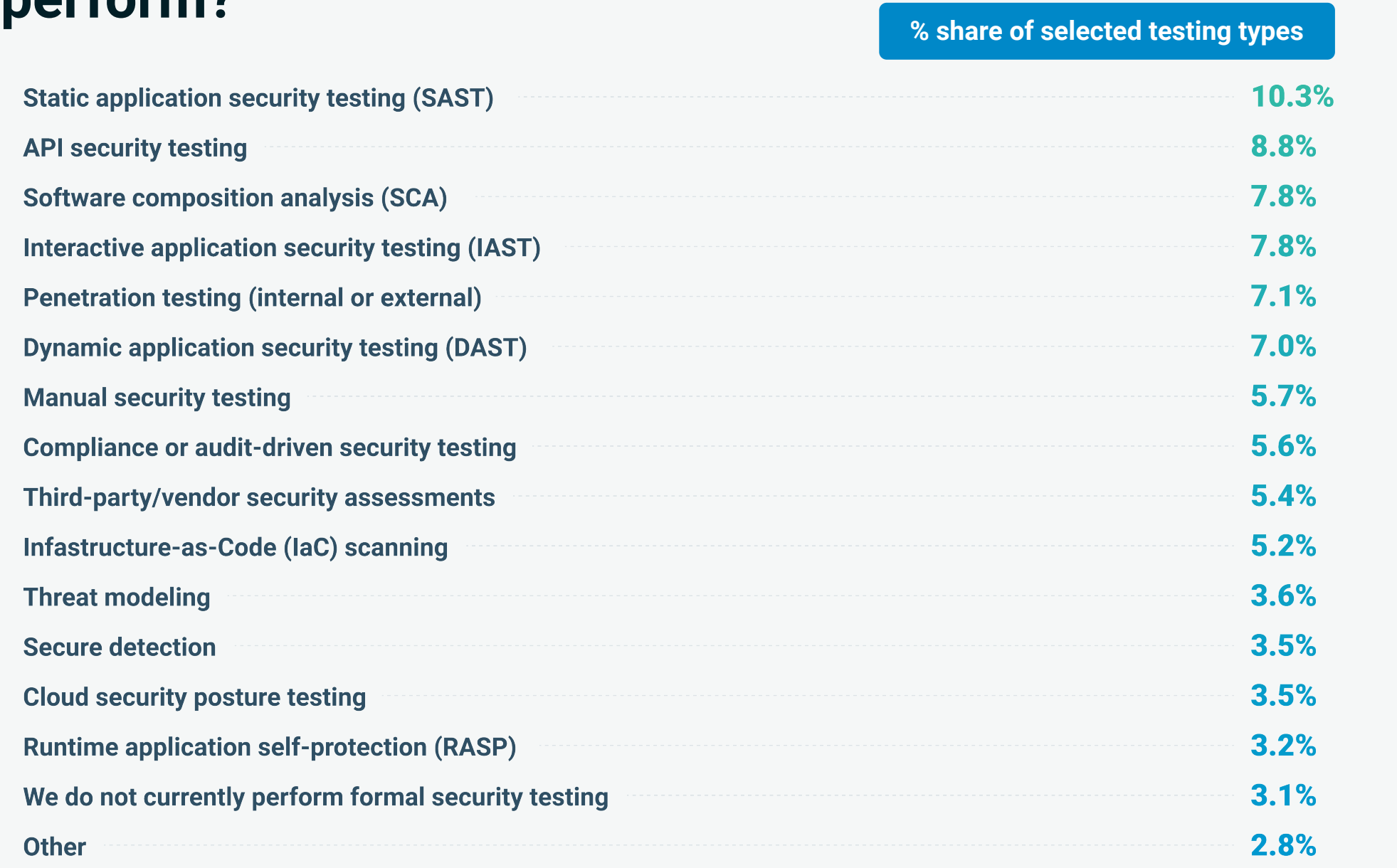
Only about half of security pros are truly satisfied with their process

Satisfaction with security processes tends to be a lagging indicator—teams often feel confident in what they’ve built until something breaks. With half of respondents reporting neutrality or dissatisfaction, the more important question isn’t who’s happy with their current setup, but whether the half who are satisfied have mature processes or have not yet encountered the failure that will change their minds.

While security teams are trying to keep pace with the threat landscape, significant process gaps remain. Some security procedures are only partially implemented, while other measures are applied inconsistently across teams or workflows. Security tools also tend to remain siloed, limiting their effectiveness within the broader security program. The data confirms that teams are aware of these gaps and working to close them, but the overall picture is one of a discipline still progressing toward genuine process maturity.

- Nearly half report neutrality or dissatisfaction with current processes
- 30% neutral response signals security education gaps may remain
- High satisfaction scores may overestimate true security maturity

What types of security testing do you currently perform?



Scattered responses indicate that security measures are still siloed

Security testing is widespread but highly fragmented; no single method dominates. [SAST](#), API security testing, and penetration testing lead the pack, reflecting where vulnerabilities are most concentrated.

The prominence of API testing is telling. Modern applications are deeply dependent on APIs that expose back-end databases and sensitive data. Unmonitored “zombie APIs” have become prime targets for attackers, a risk that only grows as GenAI integrations multiply. Despite this, there is no unified testing strategy across the industry. Security tools operate in silos, rarely communicating with one another, forcing developers to manually coordinate multiple testing methods.

Threat modeling—the most proactive security practice—ranks near the bottom in adoption, revealing a persistent gap between reactive testing and strategic risk prevention. Organizations need a centralized, cohesive security approach that provides clear visibility and closes the gaps that fragmented toolchains inevitably create.

- SAST, API testing, and pen testing lead adoption
- APIs are a growing, high-value attack target
- No unified strategy exists, and fragmentation leaves critical gaps

“Application Security Testing has evolved and become fragmented across SAST, SCA, DAST, IAST, etc, with no dominant standard. With this in mind, organizations should adopt an overall security strategy that takes into account risk appetite, compliance, and governance objectives.”

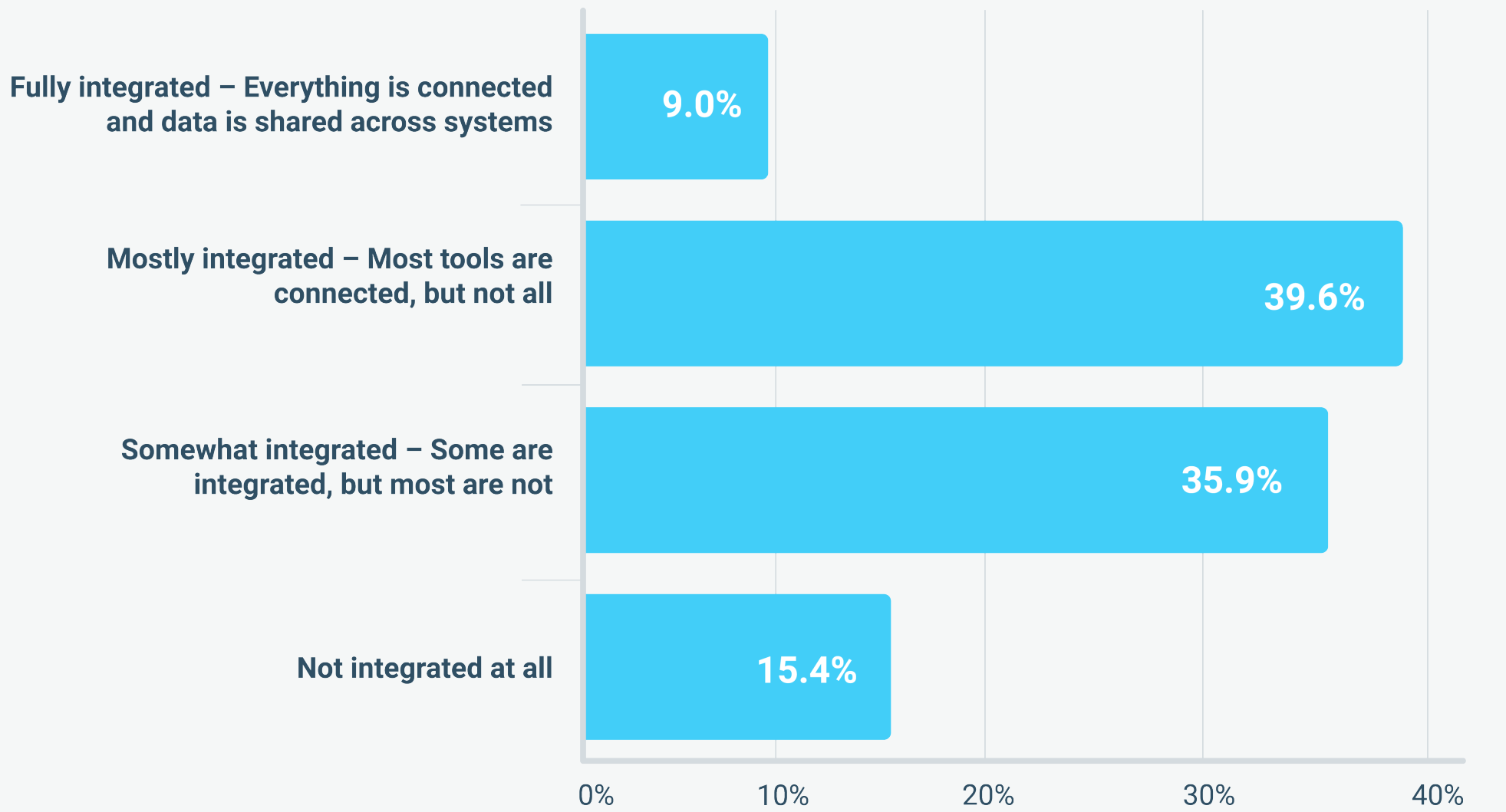
- John Brawner, Sembi Solution Architect (DevSecOps)

Security Tools, Automation, and Integration

Although the need for integrated security has been clear for some time, execution is still lagging behind expectations. As organizations push toward faster delivery and more complex architectures, security can no longer operate as a separate function—it must be embedded into the development process. DevSecOps has become the aspirational model, aiming for seamless integration between development, operations, and security. In practice, however, most teams are still working toward that ideal.

While many organizations report some level of integration and automation, few have achieved a [unified security workflow](#). Gaps in tooling integration and incomplete automation continue to introduce friction, slow pipelines, and create blind spots in risk visibility. At the same time, rising code volume, AI-assisted development, and expanding attack surfaces are making manual processes and loosely connected tools even more unsustainable. The data in this section examines how far teams have progressed and where the biggest opportunities remain to close the gap.

How integrated are your security tools with DevOps?



Only 9% of teams report fully integrated security tools within DevOps pipelines

Despite widespread adoption of DevOps, full security integration remains rare. Just over 9% of teams report that their security tools are fully integrated into DevOps workflows, while the majority operate in environments that are “mostly” (39.6%) or “somewhat” (35.9%) integrated. While this suggests progress, partial integration often still relies on manual steps, disconnected tooling, or delayed feedback loops.

Teams have moved beyond fully siloed security, but many have not yet achieved the seamless, real-time integration that DevSecOps promises. As a result, security is often present in the process but not fully embedded, limiting its ability to scale with development.

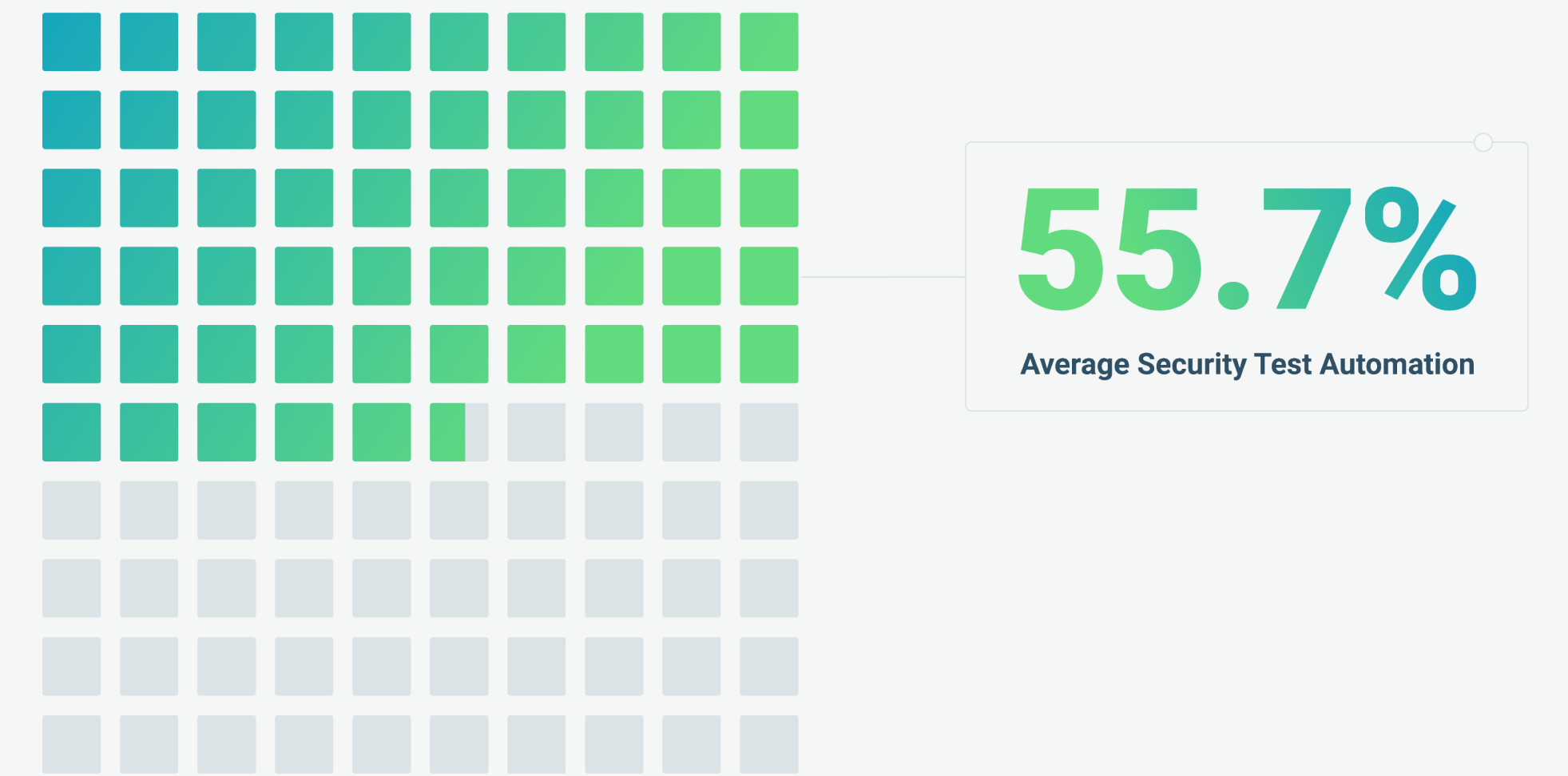
Partial integration introduces delays, manual handoffs, and inconsistent enforcement, making it difficult to maintain a reliable security posture at scale. Security signals are often not delivered in real time within developer workflows, limiting teams’ ability to act quickly when issues arise. At the same time, fragmented tooling reduces visibility across the pipeline and increases the likelihood of missed vulnerabilities. To address this, deep, native integration is required in order for security to operate continuously and scale alongside modern development practices.

- **Full DevSecOps integration remains uncommon**
- **Partial integration still creates workflow friction**
- **Deeper integration is key to scalable security**

“Testing and security teams are usually the ones who pump the brakes. You want code faster? Cool. But then security is the group saying, wait, did you think about this, or that, or compliance requirements? Testing and security should be advocates for taking the time to test and secure code properly.”

- John Brawner, Sembi Solution Architect (DevSecOps)

What percent of your security testing is automated?



Nearly half of security testing is still performed manually

Security automation is well underway, but far from complete. On average, teams report that 55.7% of security testing is automated, indicating that automation is now a standard part of security practices. However, a significant portion of testing still relies on manual processes, limiting scalability and consistency.

This reflects a transitional stage of maturity. Teams have automated repeatable tasks such as scanning and basic validation, but more complex workflows remain partially manual. As development velocity increases, this gap becomes harder to sustain. Automation is about coverage, yes, but it must also evolve to support speed, accuracy, and real-time feedback across the pipeline.

Manual processes continue to limit scalability while introducing inconsistency across security workflows. As delivery accelerates through CI/CD, gaps in automation make it difficult for teams to keep pace without sacrificing coverage or quality. Reliance on human-driven processes further increases the likelihood of errors and delays, particularly as complexity grows. To keep up, organizations need intelligent, integrated automation that enables real-time validation.

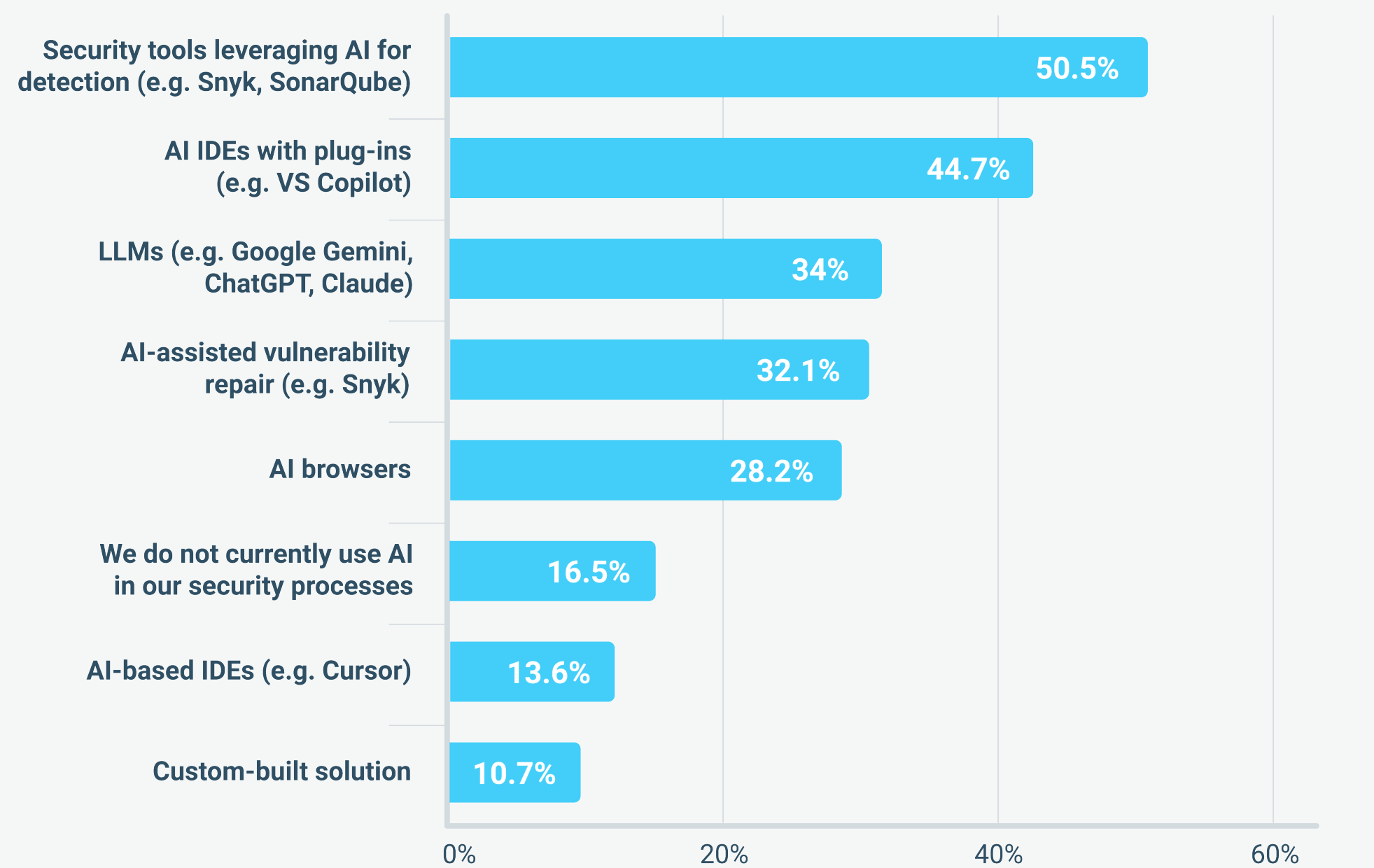
- Security automation is established but incomplete
- Manual processes limit speed and scalability
- AI-driven automation is a big opportunity

AI in Security

AI is reshaping how software is built, and teams are feeling the impact just as directly. What began as experimentation with AI-assisted coding has expanded into broader adoption across detection, analysis, and testing workflows. For many organizations, AI is no longer a future consideration; it is already embedded in day-to-day security operations.

At the same time, this shift introduces a new layer of complexity. AI is increasing both the volume of code and the range of potential vulnerabilities, while raising new concerns around trust, data privacy, and governance. Security teams are being asked to move faster, analyze more, and adapt to risks that did not exist just a few years ago. The result is a clear duality: AI is unlocking new efficiencies, but also raising the stakes. This section explores how teams are applying AI today, how it is reshaping security posture, and what is limiting broader, more effective adoption.

Do you currently leverage AI tools in your existing security processes? If yes, which?



Over half of teams use AI for threat detection, signaling mainstream adoption

AI is already embedded in security workflows. Over half of respondents (50.5%) report using AI for detection, alongside strong adoption of AI-assisted IDEs (44.7%) and large language models (34%). This shows that adoption spans both security operations and development environments, extending AI's role across the entire software lifecycle.

Rather than being confined to isolated use cases, AI is becoming a foundational layer in how teams identify and respond to threats. It enhances developer productivity, accelerates analysis, and expands detection capabilities. As adoption scales, organizations are operating under an AI-augmented security model that balances speed and efficiency with oversight, validation, and control.

AI is now embedded across both development and security workflows, fundamentally changing how teams build and protect software. As a result, security operations are shifting toward AI-augmented models that rely on automation and intelligent analysis. However, increased dependence on AI introduces new requirements for validation, oversight, and trust. While these workflows expand what teams are capable of, they also broaden the risk surface and make it critical to balance efficiency gains with stronger controls and visibility.

- **AI is now becoming mainstream in security workflows**
- **Adoption spans both development and security layers**
- **AI introduces new risks alongside efficiencies**

How has AI code generation impacted your security posture today?



Some security pros say AI improves security, but others say it's creating new risks

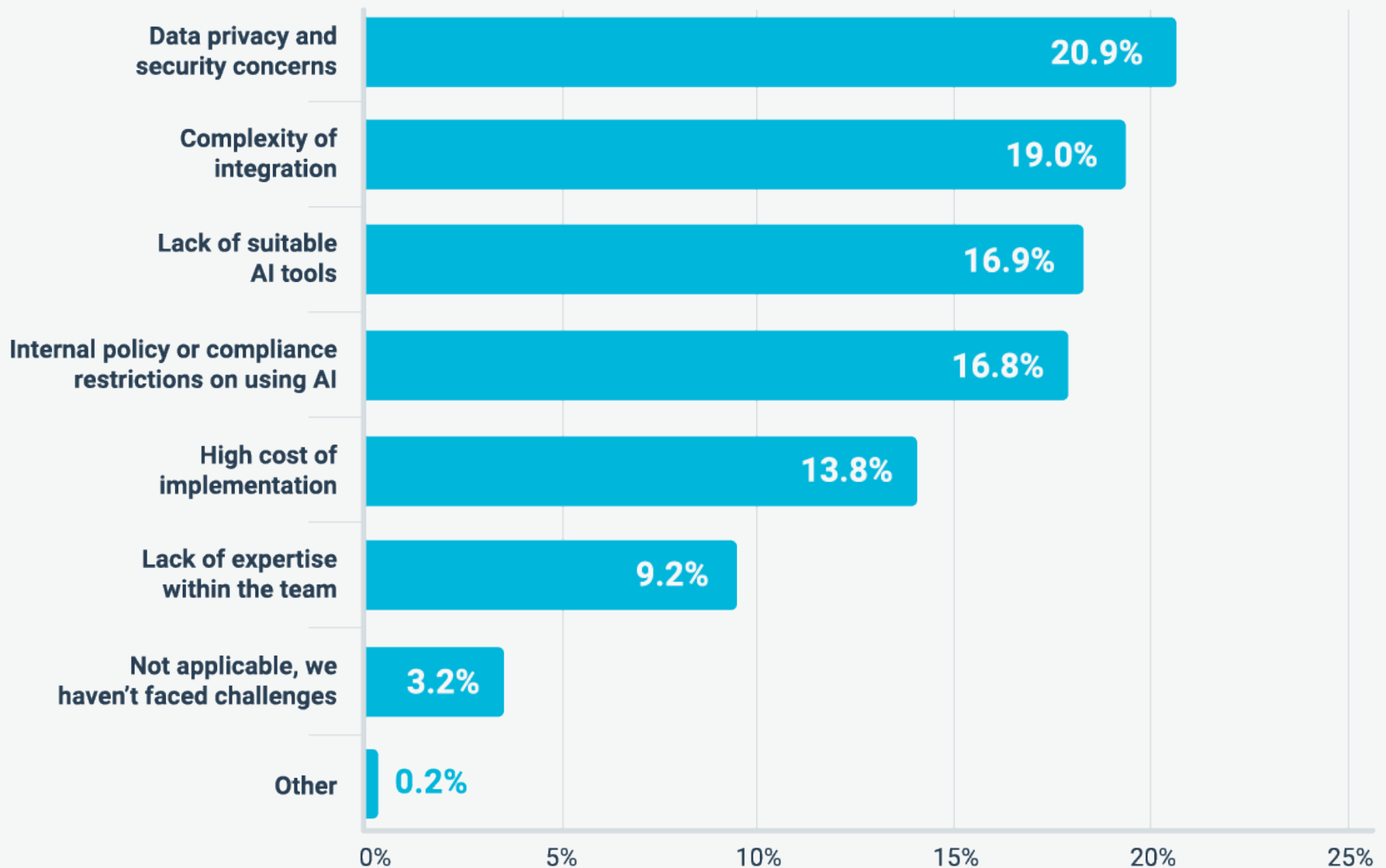
The impact of [AI-generated code](#) on security posture is both measurable and mixed. Some teams report clear benefits: 20.6% cite more advanced testing capabilities, and 14.3% report overall improvements. At the same time, 12.9% report new security concerns, reflecting uncertainty around the risks AI introduces.

This split highlights the dual nature of AI in security. It enables faster testing, broader coverage, and new approaches to detection, but also introduces unfamiliar vulnerability patterns, inconsistencies, and blind spots in traditional validation methods. The result is not a net-positive or net-negative shift, but a more complex and dynamic security landscape that requires new approaches to risk management.

AI is increasing what teams can do while raising the baseline level of risk they must manage. Traditional validation methods are not always equipped to detect AI-specific vulnerabilities, leaving potential gaps in coverage. At the same time, the surge in code volume amplifies overall exposure, making it easier for issues to slip through unnoticed. To keep pace, security strategies must evolve and adapt to new risk patterns introduced by AI-driven development.

- AI delivers both improvements and new risks
- Traditional security approaches may fall short
- AI-aware testing strategies are now essential

What challenges have you faced in integrating AI into your security process?



Organizations cite data privacy as the top barrier, with AI integration complexity close behind

While AI adoption is widespread, integrating it effectively into security processes remains a challenge. The most commonly cited barriers are data privacy concerns (20.9%) and integration complexity (19%), pointing to issues of trust and implementation rather than technical capability.

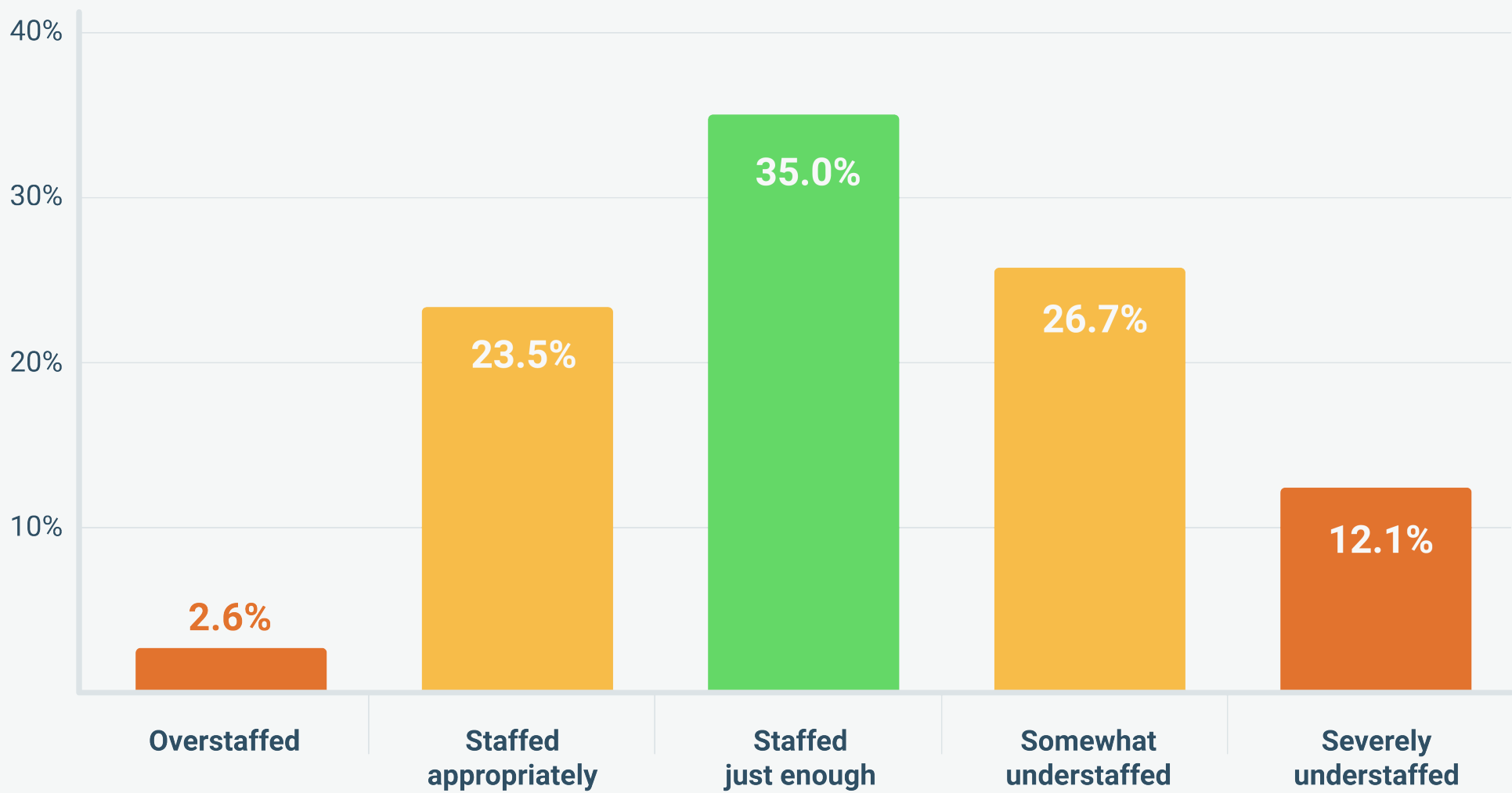
These challenges highlight a gap between experimentation and operationalization. Many teams are exploring AI tools, but embedding them securely and seamlessly into existing workflows is proving difficult. Concerns around [sensitive data exposure](#), regulatory compliance, and tool interoperability are slowing adoption and limiting how far organizations are willing to go.

- Trust and complexity limit AI adoption
- Privacy and compliance are major concerns
- Secure, integrated AI solutions are needed

Security Team Staffing and Resources

As AI accelerates the threat landscape, security teams are being asked to move faster than ever—but staffing hasn’t kept pace. [IBM’s 2024 and 2025 security reports](#) found that shortages increased 26% year-over-year, 50% of breached organizations linked the incident to understaffing, and the global cybersecurity workforce gap now stands at 4.8 million unfilled positions. To gauge where organizations stand today, we surveyed security and QA professionals on current staffing levels and 12-month hiring expectations.

Please describe the staffing level of your security team to accomplish your tasks and goals.



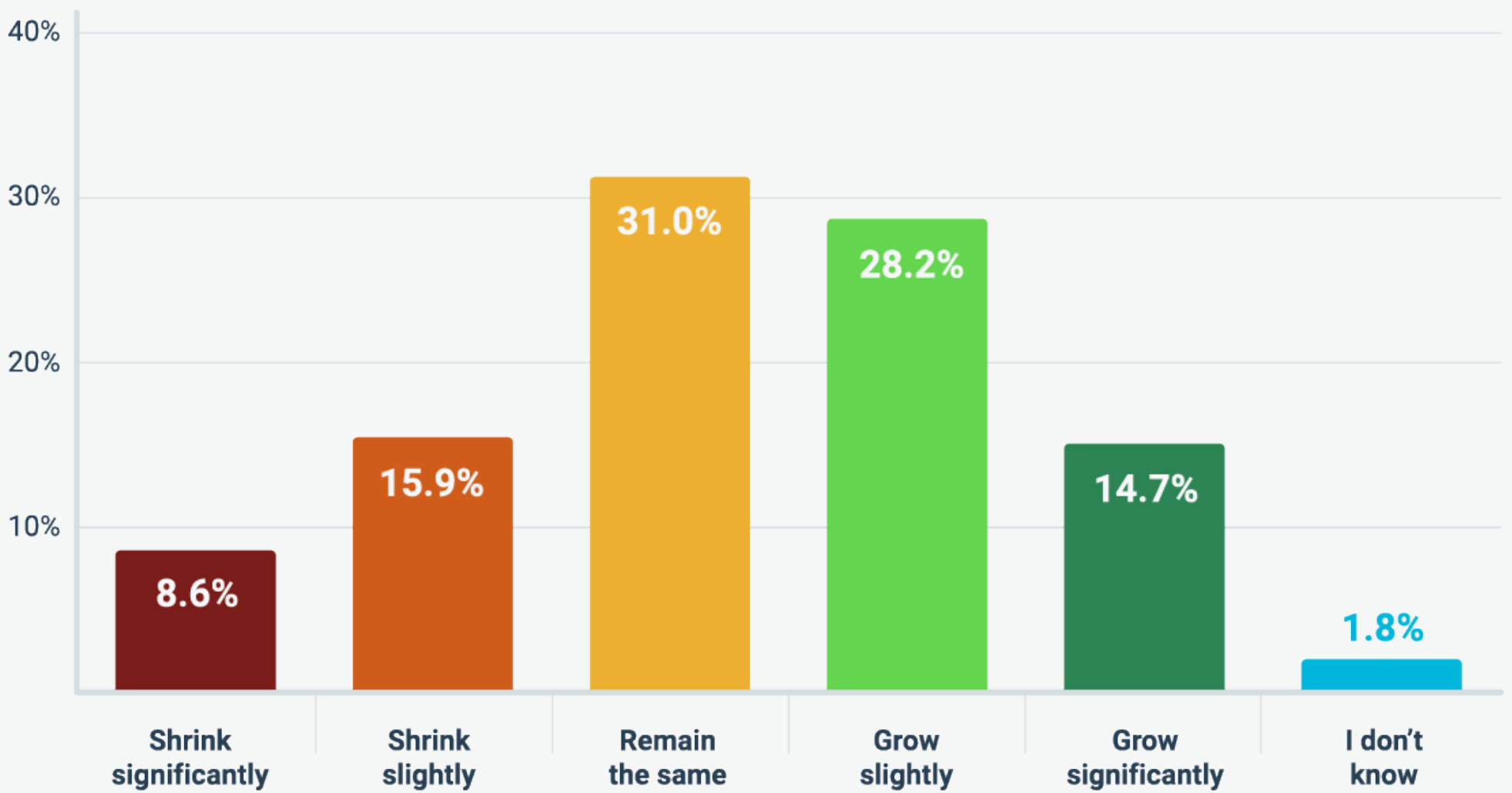
Nearly 3 in 4 security teams feel stretched thin

With only about a quarter of teams feeling appropriately resourced, these gaps are driven by compounding factors such as budget constraints, economic pressures, employee burnout, heavy dependency on tools, and niche talent shortage. Budget pressure is the leading driver. Organizations that consolidate security for financial reasons face compounding consequences: IBM reports that understaffed companies incur an average of \$1.76 million in additional data breach costs compared to those that are adequately staffed.

Thin teams also lead to burnout and turnover, deepening the shortage cycle. Specialized roles, such as cloud security, threat intelligence, incident management, and AI monitoring, are the hardest to fill, with an [average of 99 days to recruit from start to finish](#). While organizations search, teams lean on automation to fill the gap, which introduces its own bottlenecks when tools aren't properly tuned.

- **Budget, burnout, and niche talent gaps drive understaffing**
- **Specialized security roles take an average of 99 days to fill**
- **Cloud, AI, and threat intelligence roles are the hardest to hire for**

How do you expect the size of your security team to change over the next 12 months?



Despite increased demand for security, less than half of teams expect growth in 2026

The staffing outlook for security teams in 2026 is mixed and cautious. Just 28% anticipate slight growth, 31% expect headcount to stay the same, and 24% expect reductions. Despite rising threats and expanding workloads, most organizations are not investing in meaningful team expansion.

The dominant driver is a “run lean” mentality. CISOs face pressure from boards to [pursue AI-powered automation](#) rather than hire additional staff, as AI agents demonstrate the ability to handle monitoring, triage, and low-level vulnerability management. The result is a gradual shift in the human security role: rather than expanding headcount for routine coverage, organizations are redirecting existing professionals toward oversight of AI-driven systems and complex remediation—prioritizing efficiency over scale.

- Security headcount outlook is mixed and cautious for 2026
- AI tools are replacing headcount growth, not supplementing it
- “Run lean” mentality is reshaping security team structure

Software Security Summary

Security Is Aware of Its Gaps—and Is Running Out of Time To Close Them

The threat landscape is evolving faster than security practices are.

[Data breaches](#), [cloud misconfigurations](#), and [AI-driven threats](#) dominate security priority lists, yet the practices designed to prevent them remain inconsistently implemented. SAST and API testing are gaining traction, but adoption is fragmented, and coverage is uneven. Security teams are aware of what they're up against; the challenge is building the processes and infrastructure to match that awareness with action.

Automation and integration are progressing, but true DevSecOps remains out of reach.

Security teams are making progress with automation, but DevSecOps maturity is still uneven. Security automation averages 55.7%, yet only 9% of teams report fully integrated toolchains. And with just 51% of detected issues proving to be true positives, teams are still losing time to noise. Better integration and more trusted signals are needed to help security keep pace.

Staffing constraints are limiting everything else.

Nearly 39% of security teams report being understaffed, and most organizations are not planning significant headcount growth in the next 12 months. With complex architectures, evolving threats, and AI-generated code all expanding the attack surface, the math doesn't add up. Security teams are being asked to do more with less—and without automation, AI augmentation, and smarter workflows, that equation will only get harder.

Altogether, the responses in this section reveal a discipline that is clear-eyed about its challenges but still early in the work of addressing them. The priorities are right, the awareness is growing, and investment in tools and automation is real. But fragmented toolchains, persistent talent shortages, and a largely reactive posture are holding the industry back. The shift from reactive to resilient security won't happen overnight, but the teams that start building toward it now will be the ones setting the standard in the years ahead.

- 03 - Key Insights
- 04 - Global Quality Trends
- 05 - QA Testing Insights
- 06 - Software Security Insights
- **07 - QA and Security Convergence**
- 08 - Survey Details and Methodology
- 09 - Conclusion
- 10 - About Sembi and Portfolio

QA and Security Convergence Introduction

Software quality is becoming harder to separate from software security.

As AI-driven development speeds up code creation and release cycles continue to shrink, teams are being asked to validate more change, more often, with less room for error. A functional release that introduces security risk is no longer good enough, and a secure release that creates performance, reliability, or usability issues creates business risk.

That is why the relationship between QA and security is changing. What used to be treated as adjacent functions or sequential checkpoints, now needs to operate as a more connected system.

DevSecOps helped move security earlier into the software development lifecycle, but the next shift is even broader. Organizations now need unified quality and security processes that share context, surface risk earlier, and support continuous validation across development and delivery.

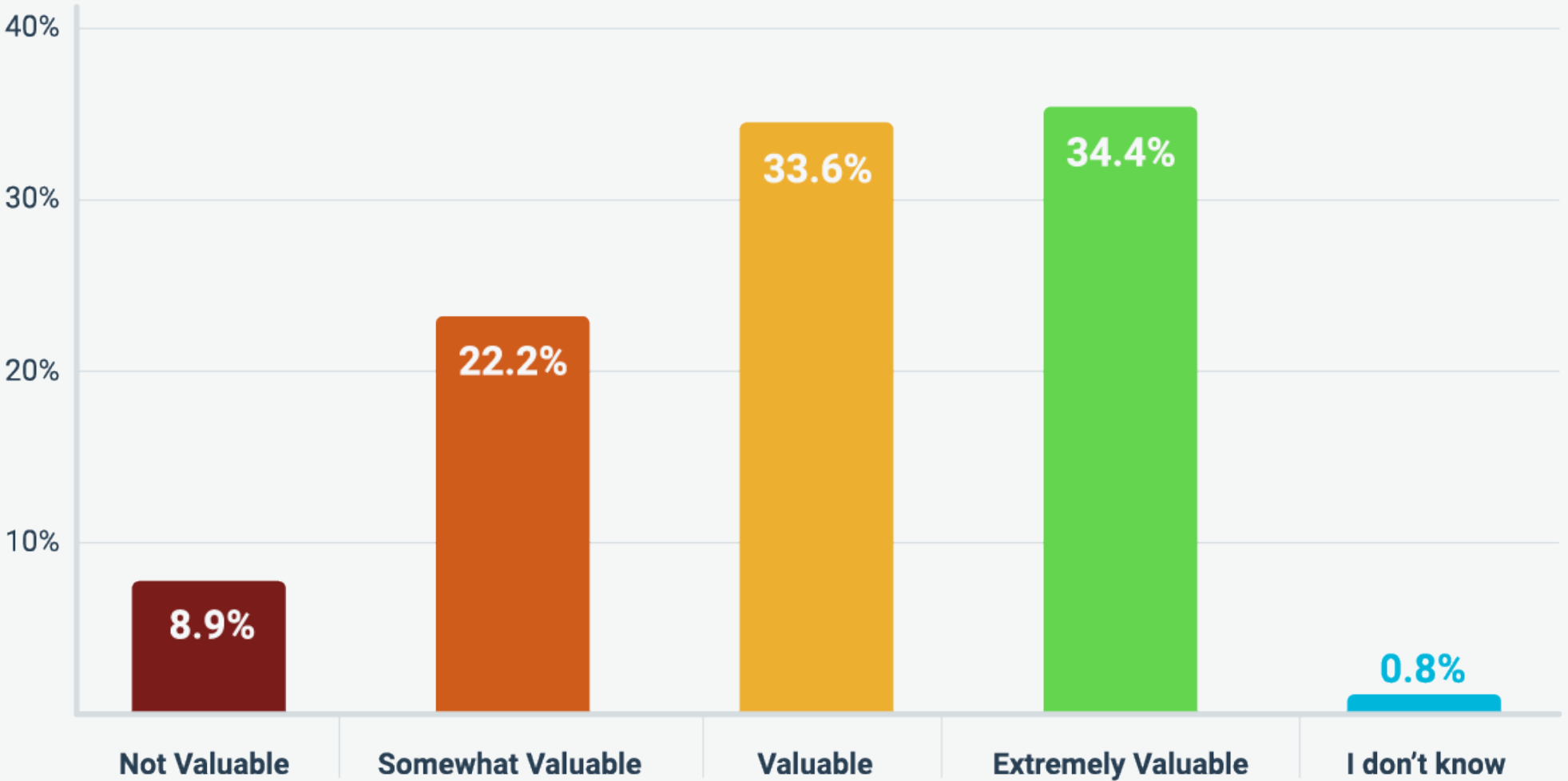
The urgency is growing for three reasons:

- **AI is increasing change velocity** across code, testing, and delivery
- **Risk is becoming more amplified** as releases move faster and systems grow more complex
- **Disconnected workflows create blind spots** between functionality, security, and release readiness

The data in this section shows that organizations recognize the value of stronger QA and security alignment. But it also makes clear that most have not yet built the processes, ownership models, and connected workflows needed to scale that alignment effectively.

The result is a growing gap between intent and execution. Teams understand where the market is headed. Many are not yet fully equipped to get there.

How valuable would it be to have better alignment between QA testing and security?

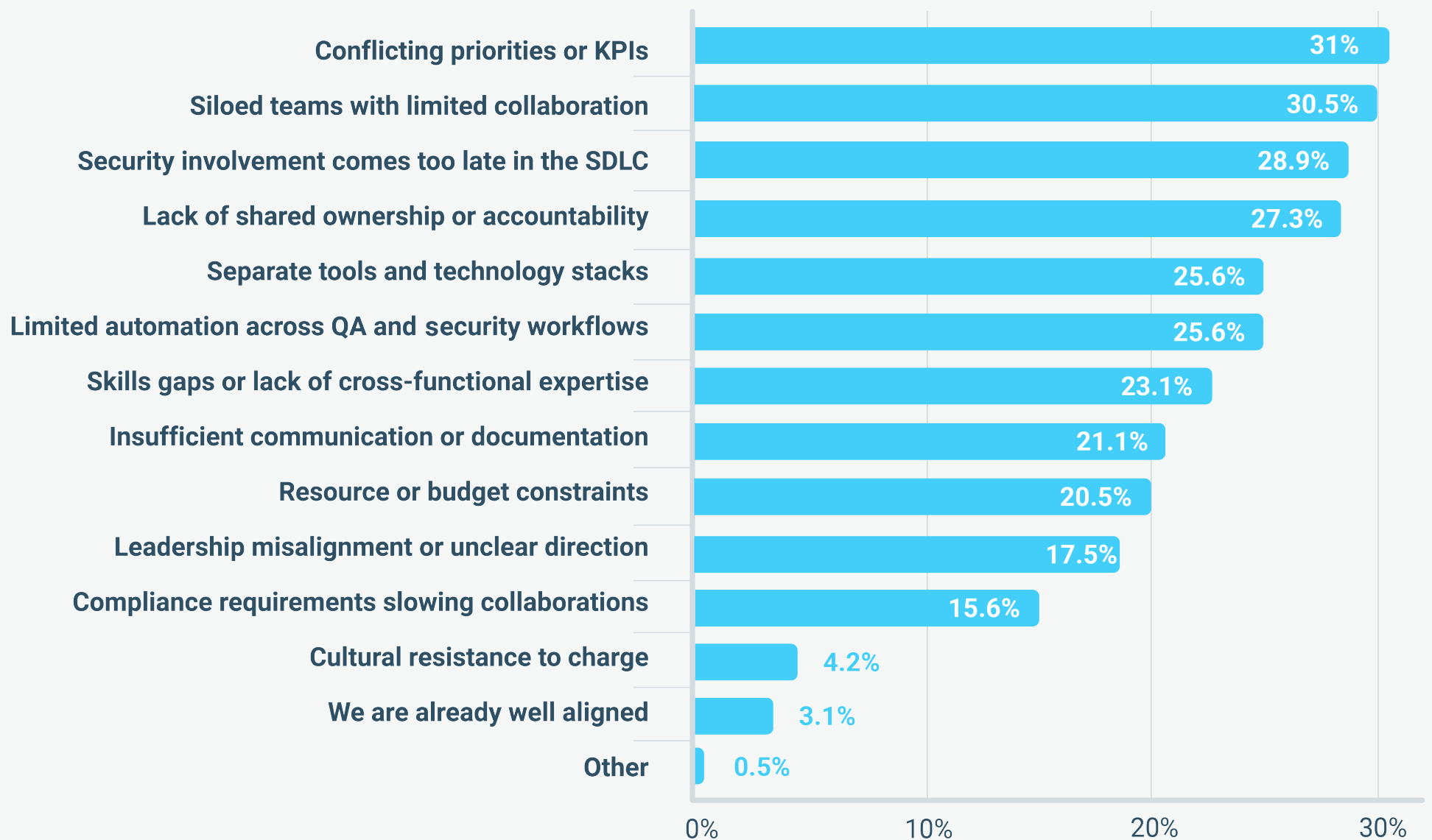


68% say stronger QA and security alignment would be valuable or extremely valuable

The signal here is clear: organizations do not see QA and security alignment as a niche improvement or aspirational idea; they see it as a business need. More than two-thirds of respondents say stronger alignment would be valuable, while fewer than 9% see little or no value in it. That level of consensus matters—it shows the market has largely moved past debating whether convergence matters and is now focused on how to make it operational. As delivery cycles accelerate and AI increases the speed and volume of code changes, release confidence depends more on combining functional quality, risk visibility, and shared decision-making. The opportunity is no longer theoretical. Better alignment is becoming an expected capability for teams trying to improve software quality without increasing exposure.

- **Convergence is now market validated**
- **Alignment is becoming an expected capability**
- **Release confidence depends on shared context**

What are the biggest obstacles preventing stronger alignment between QA testing and security?

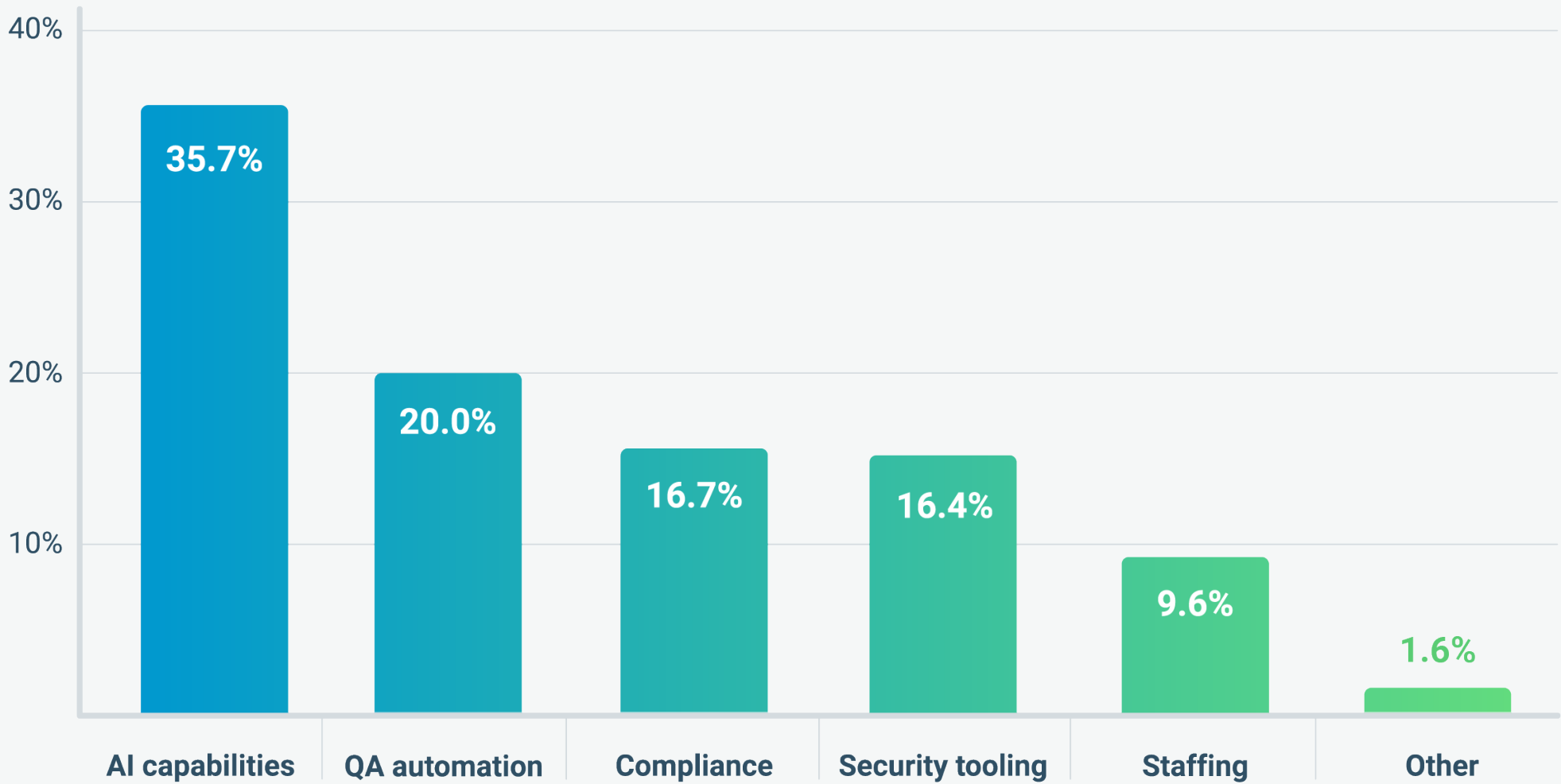


The biggest blockers are structural: KPIs, silos, and late security involvement

The obstacles to convergence are not technical; they are organizational. Respondents point first to conflicting priorities or KPIs (31%), siloed teams with limited collaboration (30.5%), and security involvement arriving too late in the SDLC (28.9%). Lack of shared ownership, separate tools, and isolated workflows also rank highly. Together, these responses show that most teams are not failing because they lack awareness of the issue. They are failing because their operating model still reinforces separation. QA and security may both influence release quality, but in many organizations, they are measured differently, work in different systems, and engage at different points in the lifecycle. That makes alignment difficult to sustain. The real challenge is not adding more tools, it is creating shared accountability, better coordination, and connected workflows across both disciplines.

- **Misaligned incentives block collaboration**
- **Silos still shape daily execution**
- **Operating models lag strategic intent**

Where will investment increase most in the next 12 months?

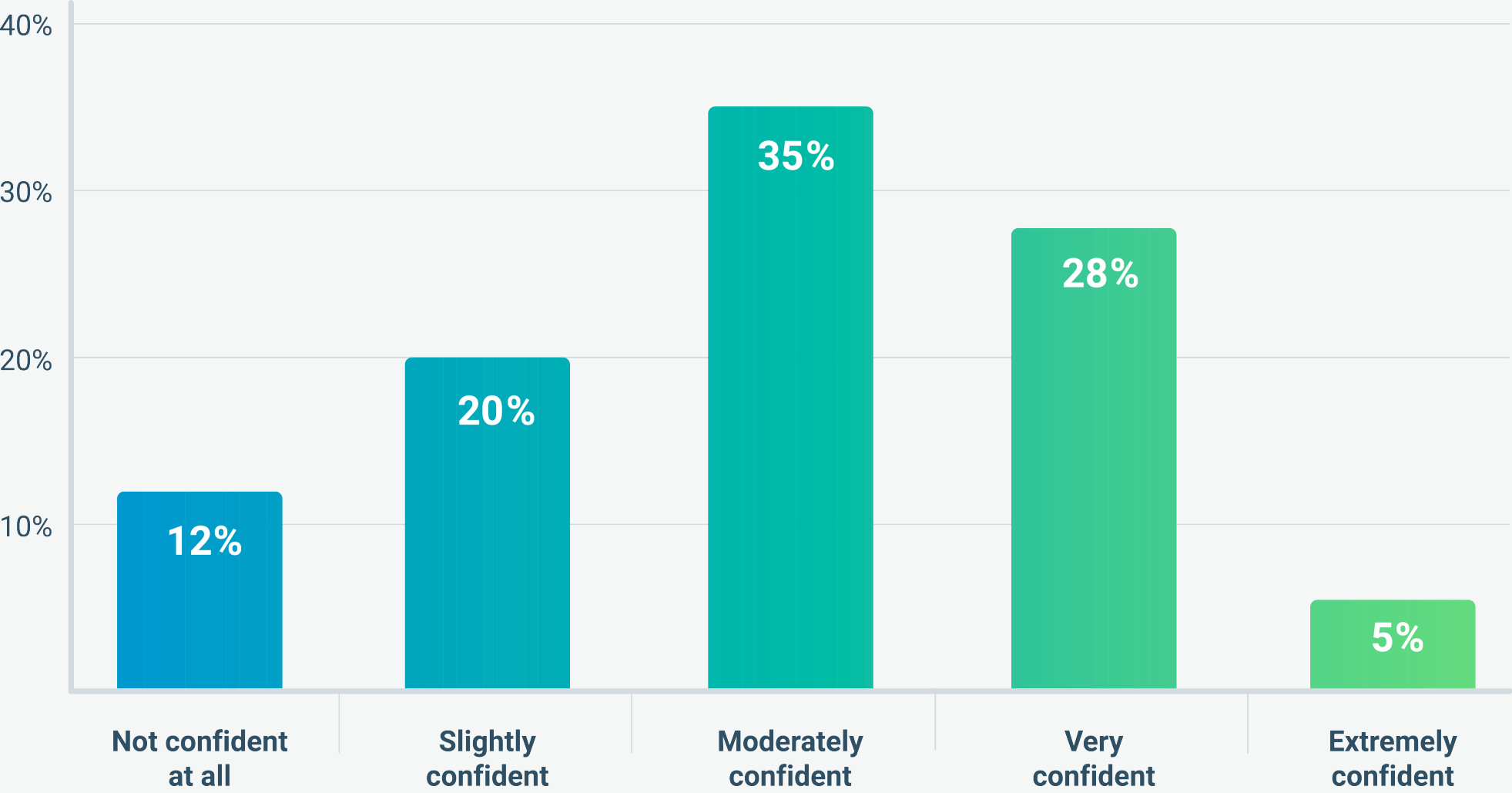


AI leads investment priorities at 35.7%, well ahead of every other category

AI is now the top investment priority for organizations looking to improve QA and security outcomes, outpacing QA automation, compliance, security tooling, and staffing. That suggests leaders see AI as the fastest path to scale, efficiency, and broader coverage, but the result also introduces tension. If AI investments are made separately within QA and security, organizations risk accelerating fragmentation rather than solving it. QA may invest in [AI for test generation](#), coverage optimization, and faster execution, and security may focus on detection, remediation, and vulnerability analysis. Both are valid, but without shared signals, common workflows, and connected decision-making, they can create parallel intelligence systems instead of a unified quality model. The data shows strong momentum behind AI. The bigger question is whether organizations will use that momentum to create convergence or deepen the divide.

- AI is the dominant investment bet
- Automation remains important, but secondary
- More tooling alone will not align teams

How confident are you in your org’s ability to scale QA testing and security together in the coming year?



Only 33% are very or extremely confident that they can scale QA and security together

This is where the execution gap becomes visible. While organizations clearly value alignment between QA and security, relatively few feel highly confident in their ability to scale it. Just one-third report high confidence, while roughly the same share reports low confidence. Most fall in the middle, suggesting cautious optimism without real operational maturity. This gap between intent and capability is one of the most important findings in the section. Teams understand where the market is headed, but many still lack the process integration, tooling coordination, and cross-functional ownership required to get there. As software complexity increases and AI speeds up change velocity, scaling disconnected QA and security practices becomes even more difficult. Confidence remains constrained not by ambition, but by the absence of a unified model for how these functions should work together.

- Intent is stronger than execution
- Scale remains a maturity challenge
- Confidence is limited by fragmentation

Key Insights and Takeaways

Insight 1: Alignment is recognized, but not yet operationalized

Organizations agree that QA and security should work more closely together. The demand side of the equation is strong: Teams see the value, leadership is investing, and the language of cross-functional alignment is well established. But the operating reality has not caught up. The data shows that alignment often exists at the level of intent, not execution. Teams may collaborate in meetings, share broad goals, or align around release milestones, but they still work through separate metrics, disconnected systems, and fragmented workflows. That creates a gap between what organizations believe is important and what they are actually built to do. The implication is significant: convergence will not be achieved through mindset alone. It requires systemic changes in ownership, orchestration, and workflow design. In many cases, the challenge is no longer convincing teams that alignment matters—it is giving them the structure to act on it consistently.

- **68%** say stronger alignment would be valuable or extremely valuable
- **31%** report struggling with conflicting KPIs as a top obstacle
- **30.5%** report struggling with siloed teams with limited collaboration

Insight 2: AI may accelerate divergence as much as convergence

AI is emerging as the biggest area of future investment across the software quality landscape, but investment alone does not guarantee integration. In fact, the data suggests AI could widen the gap between QA and security if it is deployed independently within each function. QA teams may use AI to generate tests, improve coverage, and optimize execution, and security teams may apply AI to threat detection, vulnerability analysis, and remediation guidance. These are powerful use cases, but they can reinforce different priorities, signals, and workflows if they are not connected through a shared context layer. As AI becomes more embedded in the SDLC, organizations will need to decide whether it serves as a unifying intelligence layer or a faster way to scale existing silos. The long-term winners will be those who use AI to connect quality and risk decisions, not just automate them separately.

- **35.7%** say AI capabilities will receive the most increased investment
- QA automation trails AI in priorities, with **20%** reporting increased investment
- Security tooling and compliance trail even further, with only **16%** indicating they are an investment priority

Key Insights and Takeaways

Insight 3: The real bottleneck is orchestration, not more tools

Most organizations already have the basic building blocks of modern software quality: test management systems, automation frameworks, CI/CD pipelines, and a growing stack of security tools. Yet the data shows alignment still breaks down because these systems do not operate as part of a shared model. Instead, data is fragmented, workflows are isolated, and decisions are often made without complete context. This is why respondents point to structural blockers like KPIs, ownership, and late-stage involvement rather than a simple lack of tooling. The market does not just need more products; it needs better coordination across the systems teams already use. That indicates a broader shift from point solutions to orchestration layers that connect signals, workflows, and decision-making across QA and security. The future of QualityOps will depend less on individual tools and more on how well organizations unify them into one operational framework.

- **27.3%** cite lack of shared ownership or accountability
- **25.7%** cite separate tools and technology stacks
- **25.7%** cite limited automation across QA and security workflows

Insight 4: The future of quality is unified and continuous

Taken together, the data points to a larger market transition. QA and security are no longer separate checkpoints that can operate independently and still keep pace with modern software delivery. As release cycles shorten, AI-generated code increases change velocity, and risk becomes harder to isolate, organizations need a more continuous and integrated model. That means moving away from sequential workflows and toward ongoing validation across functionality, security, and release readiness. It also means shifting from disconnected toolchains to more unified ecosystems that preserve context from development through production. This is the core promise behind convergence: not just more collaboration between teams, but a new operating model for software quality. The organizations that adapt fastest will be better positioned to reduce risk, improve release confidence, and respond to change with more speed and clarity.

- **68%** see stronger alignment as valuable
- **Only 33%** are highly confident in their ability to scale it
- **28.9%** say security still enters too late in the SDLC

QA and Security Convergence Summary

The future of software quality will be shaped by how well organizations bring QA and security together. The data shows the market already understands the value of convergence, but most teams are still early in building the operational model required to support it. The biggest barriers are not awareness or intent. They are structural misalignment, disconnected workflows, fragmented tooling, and limited shared accountability.

At the same time, AI is raising the stakes. It is accelerating development and creating new opportunities to improve testing, analysis, and risk detection. But without stronger integration, AI will scale disconnected processes faster.

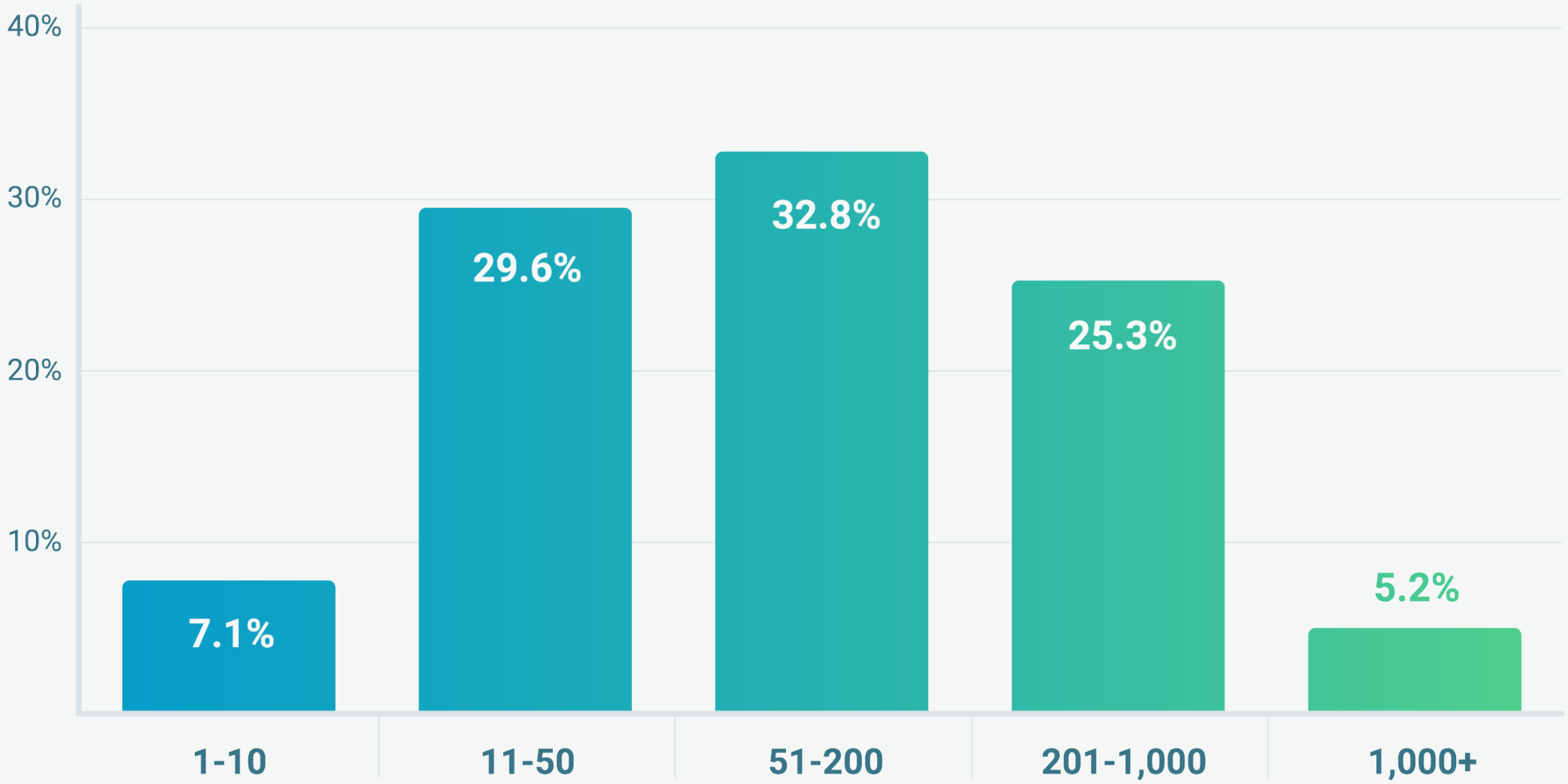
That is why convergence matters now. The next era of software quality will not be defined by isolated testing or standalone security checks. It will be defined by unified, continuous, context-aware systems that connect functionality, risk, and delivery outcomes. Organizations that make that shift will be better equipped to build faster, release with confidence, and manage risk as software complexity continues to grow.

- 03 - Key Insights
- 04 - Global Quality Trends
- 05 - QA Testing Insights
- 06 - Software Security Insights
- 07 - QA and Security Convergence
- **08 - Survey Details and Methodology**
- 09 - Conclusion
- 10 - About Sembi and Portfolio

Survey Details and Methodology

The 2026 Sembi Software Quality Pulse Report is based on survey responses collected from 3,825 software quality and security professionals worldwide. Respondents were drawn from across the Sembi product portfolio—including users and communities associated with [TestRail](#), [Xray](#), [Ranorex](#), [Testmo](#), [DesignWise](#), [PreEmptive](#), and [Kiuwan](#)—ensuring broad representation across QA, security, development, and engineering disciplines. The following section outlines the composition of this year’s respondent pool across organization size, geography, department, seniority level, and industry vertical.

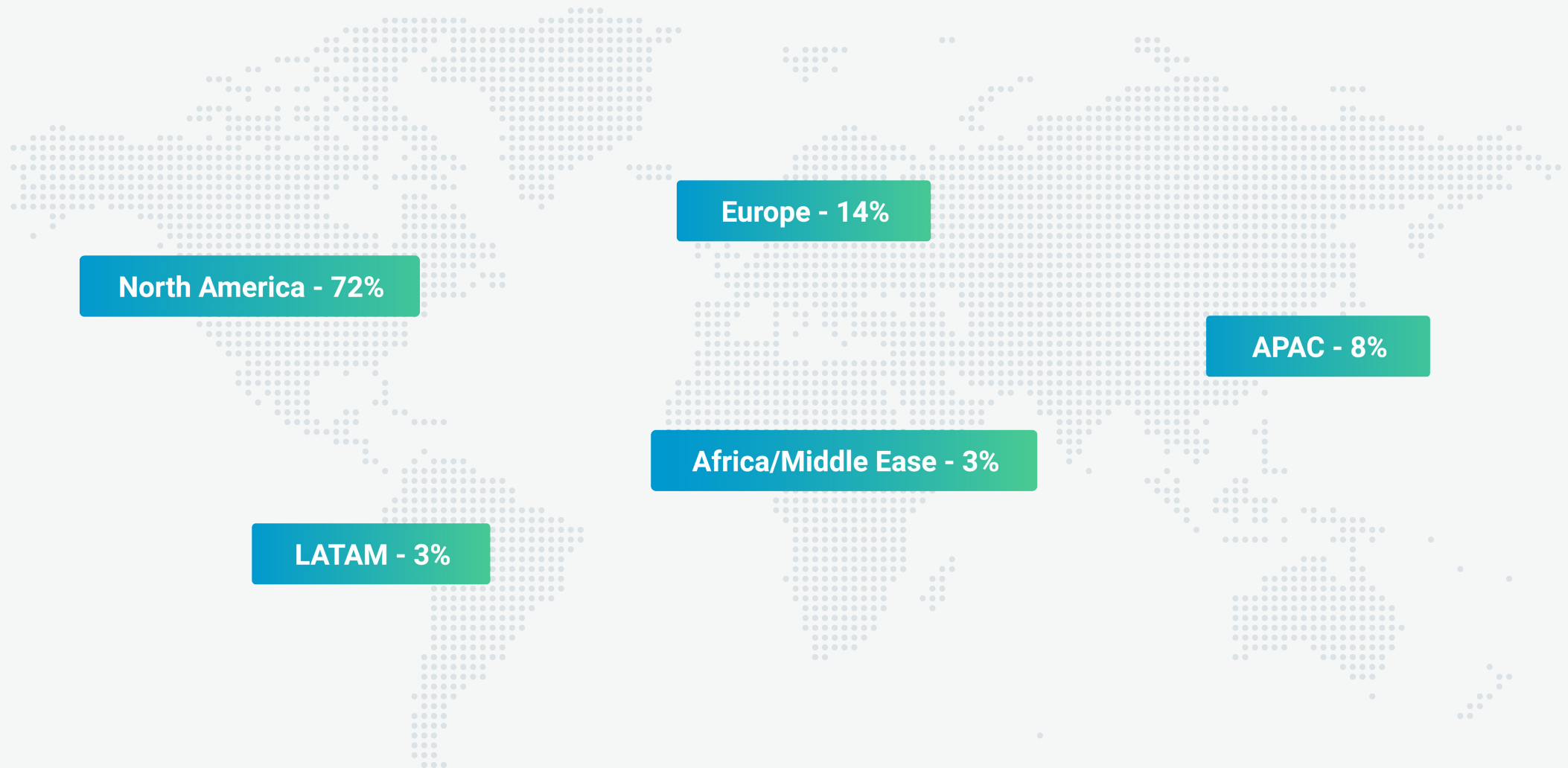
How large is your organization’s software development and engineering team?



The vast majority of survey respondents (87.7%) reported operating in mid-to-large-sized engineering environments, defined as more than 11 but less than 1,000 developers, with the largest cohort (32.8%) reporting a team size of 51–200.

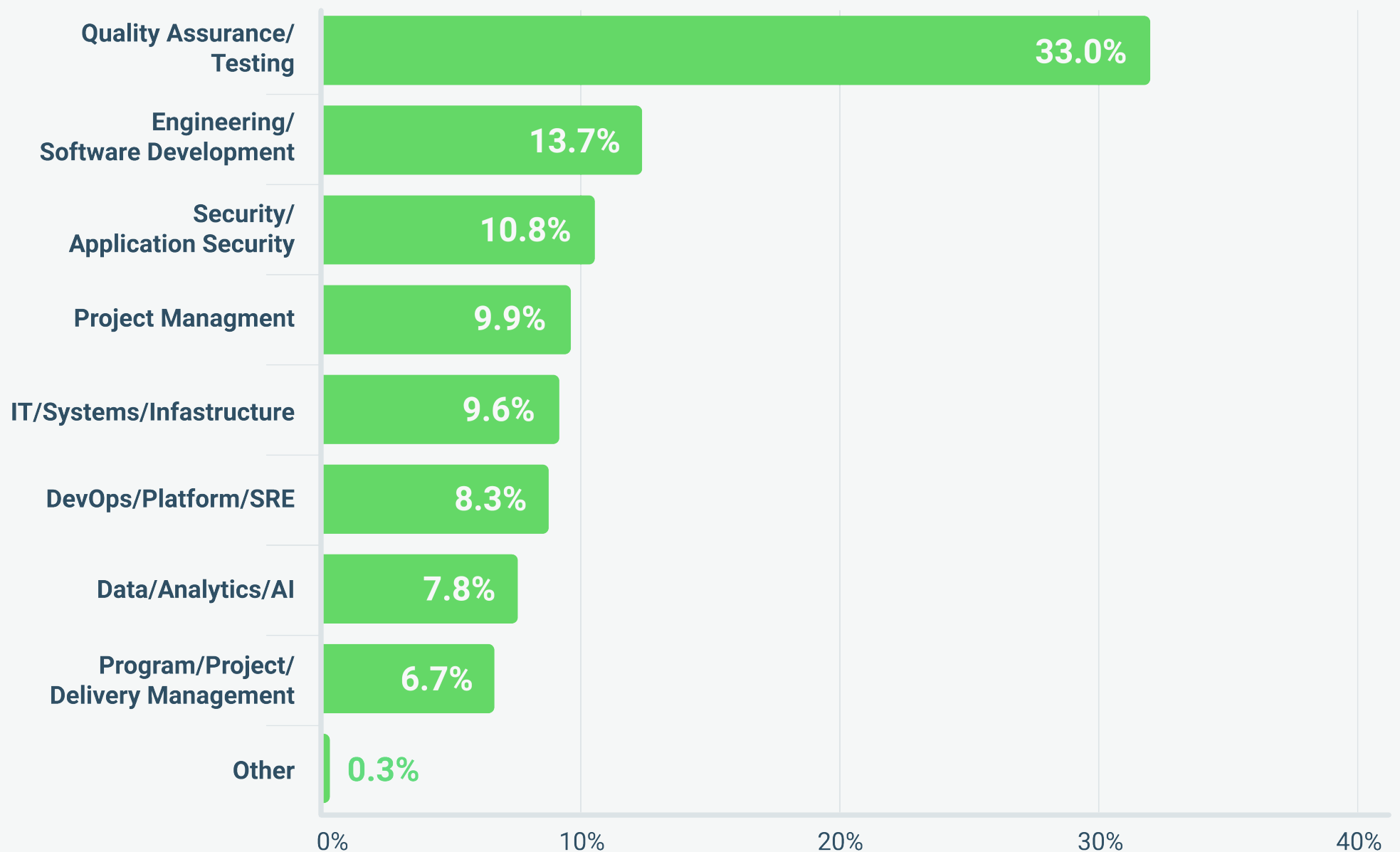
Structured QA and security processes are more often in place at organizations with mid-to-large-sized development and engineering teams. Although it looks and feels different to be on an 11-developer team versus a 1,000-developer team, the findings of this year’s report make it apparent that teams of all sizes are feeling the effects of technology advancement, increased complexity, and resource strain.

What region are you primarily located in?



This year's survey had heavy representation from North America, at 72% of respondents. Europe (14%) and the Asia-Pacific region (8%) took up second and third place.

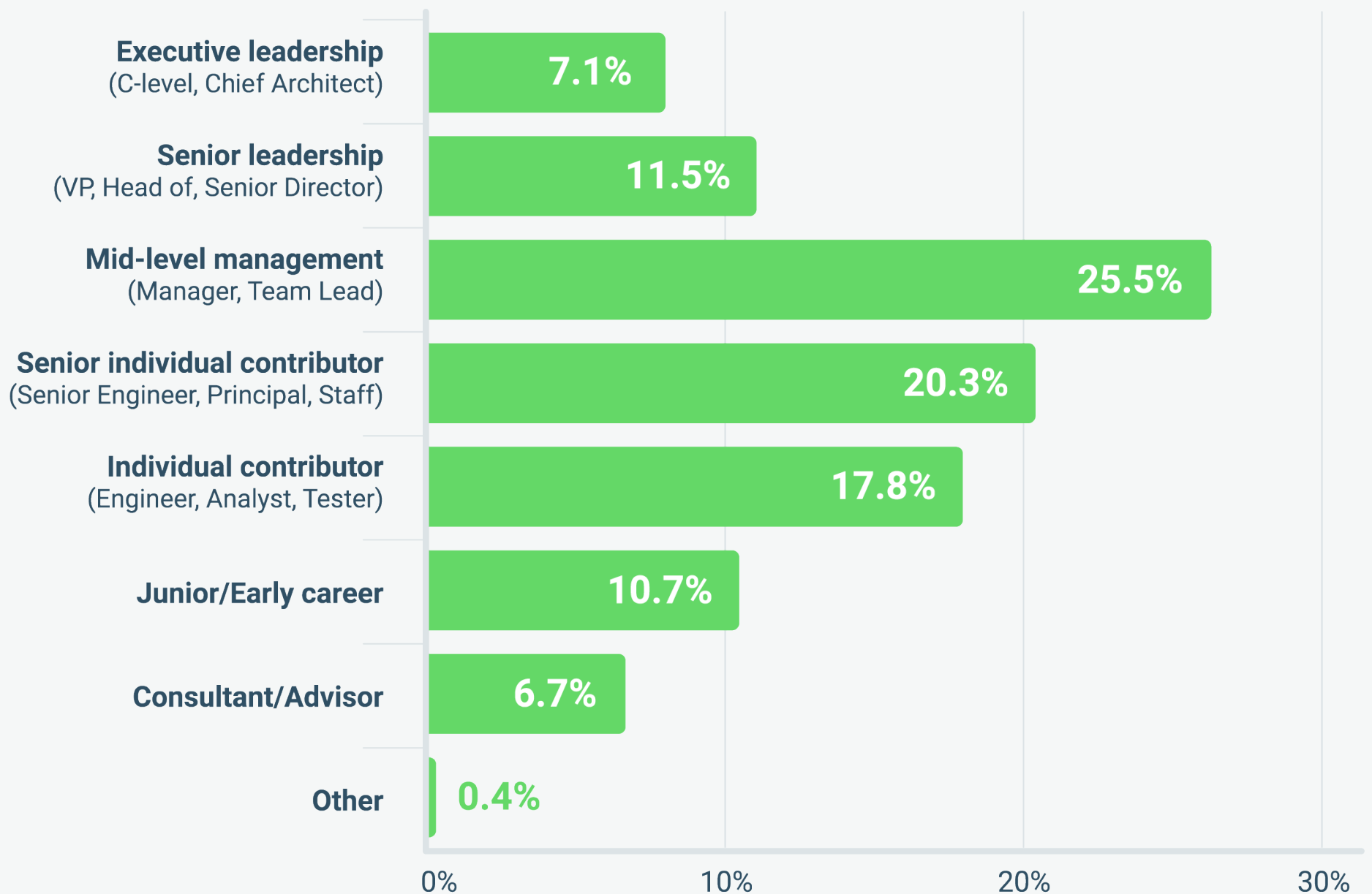
Which department or functional area do you primarily work in?



This year's survey invitation was extended to users, followers, and fans of the entire Sembi portfolio of products (TestRail, Xray, Ranorex, Testmo, DesignWise, PreEmptive, and Kiuwan). Responses had significant representation from those in quality assurance and testing (33%), and the remaining participants were a largely even split between engineering/development, security, DevOps, product management, IT, data/AI, and program management.

This demonstrates that the Sembi Software Quality Pulse report is a well-represented snapshot of insights from every aspect of software quality and security, from development to delivery. Additionally, the number of users who indicated within the survey that they work with both QA and security indicates that a large cohort of those with "QA" titles are also hands-on with application security.

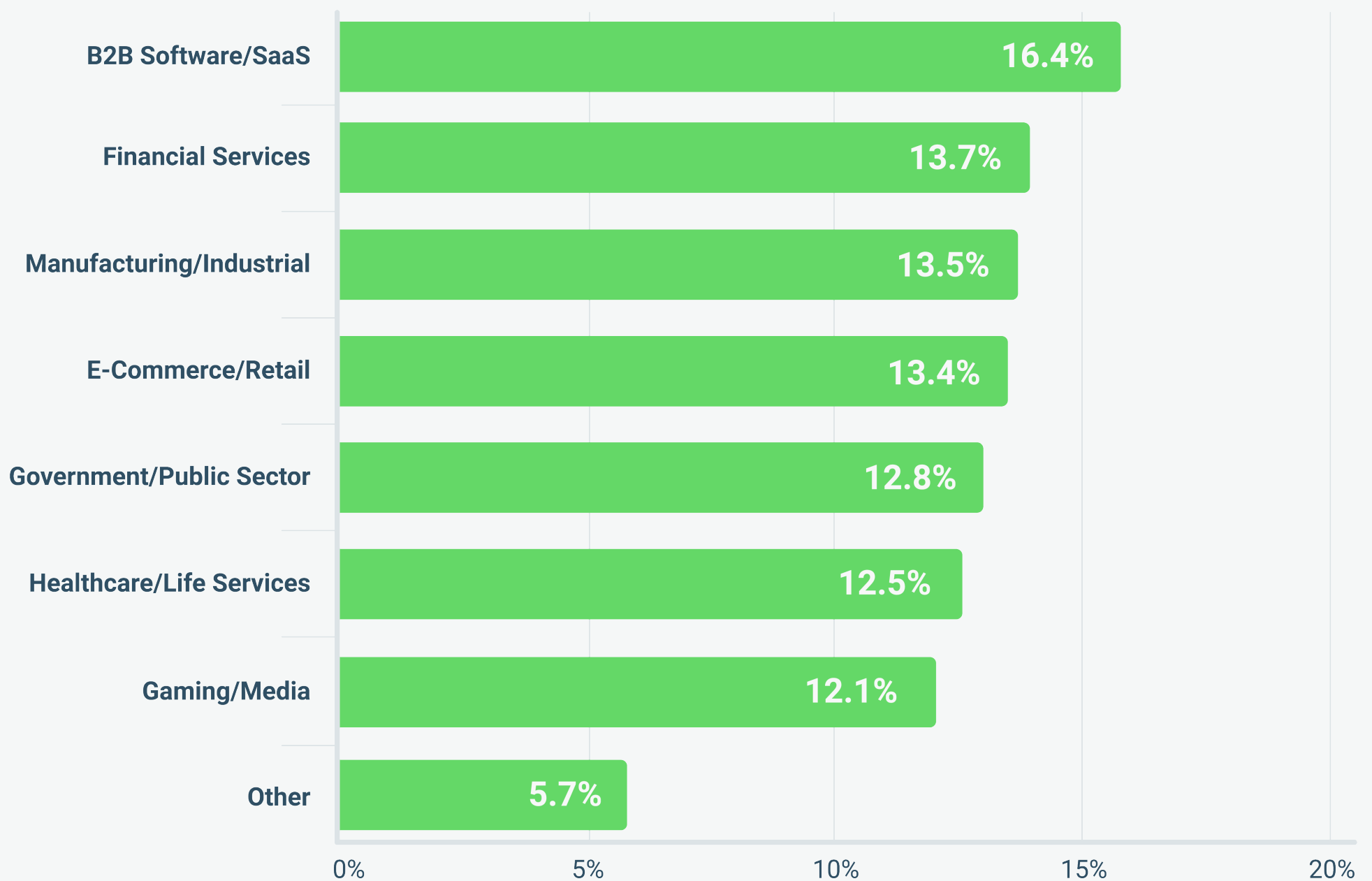
Which best describes your role level?



Mid-career professionals made up the majority of survey respondents, with a combined total of 63.6% reporting across individual contributors, senior individual contributors, and mid-level management.

This demonstrates that the insights found in this report are largely coming from quality and security professionals who both have the context and expertise that comes from years in their fields, while also still being the ones out there getting the work done hands-on every day.

What industry does your organization operate in?



The industry representation among this year's survey respondents was a remarkably even split, with B2B software pulling ahead with a slight edge at 16.4% and the remaining industries coming in with 12–14% each.

Representation from industries such as healthcare, financial services, government, and manufacturing is significant, as those industries are highly regulated. Regulated industries are subject to strict compliance requirements that can complicate things like AI adoption and systems integration, and are also more likely to be targeted by attackers due to the sensitive nature of the data they store and process.

- 03 - Key Insights
- 04 - Global Quality Trends
- 05 - QA Testing Insights
- 06 - Software Security Insights
- 07 - QA and Security Convergence
- 08 - Survey Details and Methodology
- **09 - Conclusion**
- 10 - About Sembi and Portfolio

Conclusion

The software industry is in the middle of a transformation that has no clear endpoint. Release cycles are accelerating, systems are growing more complex, and the expectations placed on the teams responsible for quality and security have never been higher. What this year's data makes clear is that the old model of siloed teams, sequential workflows, and reactive practices is no longer built for the demands of modern software development. The transformation is not coming; it is already here.

At the center of that transformation is AI. Respondents estimate that, on average, over half of their organization's code is now AI-generated or AI-assisted. That is changing the volume, velocity, and risk profile of software output, while putting new pressure on the processes designed to validate it. AI is no longer a future consideration for QA and security teams. It is an operational reality that demands thoughtful, strategic adaptation.

The teams that treat AI as a tactical add-on will struggle to keep pace. Those that evolve their quality and security practices around it will be better positioned to move faster, manage risk, and release with confidence.

The convergence of QA and security is the natural conclusion of everything this report describes. When development moves this fast, and AI introduces this much complexity, the cost of maintaining separate quality and security disciplines becomes too high. The industry is already moving toward a unified model built on shared workflows, integrated tooling, and continuous validation across the entire development lifecycle. The only question is how quickly organizations will make that shift.

The nearly 4,000 voices behind this report represent an industry that is awake, adaptive, and moving forward. The challenges are significant, but so is the collective will to meet them. The future of software quality belongs to those who build it together.

- 03 - Key Insights
- 04 - Global Quality Trends
- 05 - QA Testing Insights
- 06 - Software Security Insights
- 07 - QA and Security Convergence
- 08 - Survey Details and Methodology
- 09 - Final Conclusion
- **10 - About Sembi and Portfolio**



About Sembi

Sembi is a testing and security management leader, bringing together the industry's premier tools for software quality. Formed by Idera, Inc., Sembi unites top software delivery brands into a powerful, cohesive suite of solutions designed to support every phase of the software development lifecycle.

From automated testing to advanced security management, Sembi's portfolio empowers organizations to build software the world can rely on. With a focus on ease of use, adaptability, and proven reliability, Sembi's tools deliver immediate value for teams of every size. More than a collection of tools, Sembi is a collaborative ecosystem built to maximize efficiency, simplify the testing process, and drive continuous innovation—backed by dedicated, award-winning support every step of the way.

[Learn More](#)

About Sembi IQ

Sembi IQ is the Intelligent Quality AI Engine powering the next generation of quality operations. As the centralized AI platform driving next-generation features across Sembi's industry-leading brands, Sembi IQ is built on one of the largest operational datasets in software quality—enabling precise, context-aware AI outputs grounded in the realities of modern development.

Leveraging state-of-the-art AI models, including Agentic AI, GenAI, and machine learning, Sembi IQ solves real-world problems with real data securely, scalably, and seamlessly. Embedded directly within the tools teams already use, Sembi IQ requires no new systems to learn. Its centralized architecture accelerates innovation across the entire Sembi portfolio, turning breakthroughs into practical benefits at scale.

[Learn More](#)

The Sembi Portfolio

Sembi brings together the industry's most trusted tools for software testing and security: a unified portfolio designed to streamline quality operations and protect applications at every stage of development.

Software Testing

TestRail  — The standard in AI-driven test management

 **Xray** — #1 AI-enabled Jira-native test management

 **testmo** — Modern, agile test management

 **Ranorex** — All-in-one cross-platform UI test automation

 **XPORTER** — Enhanced Jira exporting and reporting

 **designwise** — AI-powered test optimization

Software Security

 **PreEmptive** — Advanced in-app obfuscation protection

 **kiuwan** — SAST, SCA, and code quality insights