A CHECKMARX REPORT

APPSEC: THE VIEW FROM SECURITY AND SOFTWARE DEVELOPMENT EXPERTS

Checkmarx

INTRODUCTION

AppSec managers and software developers are at the leading edge of building defenses against online attacks.

Hackers' attempts to break through the walls of businesses are becoming more sophisticated, so Checkmarx was keen to discover experts' impressions about the current and future security landscape.

In a wide-ranging survey, AppSec managers and developers told us their hopes and fears, including:

» Many lack confidence about the security of their software, especially around their digital transformation efforts.

» More than two-thirds of respondents claim their organization has suffered multiple online security breaches in the past year.

» They recognize a growing need to tighten supply-chain security.

» Despite all this, they're certain better security tooling, training, and testing can help teams collaborate to keep out bad actors.

This report highlights our survey's findings in more detail, pointing the way to better AppSec.

METHODOLOGY

Independent research consultancy Censuswide conducted research on behalf of Checkmarx, the global leader in developer-centric Application Security Testing solutions, in August and September 2021, with two separate panels as well as combined data.

» **Sample one:** 754 AppSec managers in companies with 1,000+ employees with in-house software development in the US, UK, France, Germany, Switzerland, Austria, Australia, and New Zealand between August 10 and 27, 2021.

» **Sample two:** 770 software developers in companies with 1,000+ employees with in-house software development in the US, UK, France, APAC, and DACH between August 10 and 31, 2021.

» **Combined sample:** 1,524 AppSec managers and software developers across the US, UK, France, APAC, and DACH between August 10 and September 13, 2021.

Censuswide abides by and employs members of the Market Research Society, which abides by the principles of the ICC/ESOMAR code.

All percentages shown in this report are rounded to the nearest whole number.

CENSUSWIDE
THE SURVEY CONSULTANTS

CONTENTS

CURRENT STATE OF APPLICATION SECURITY

First, let's look at the current state of application security from the perspective of software developers and AppSec managers who participated in the survey.

MULTIPLE BREACHES

We asked respondents how many times in the 12 months leading up to the survey their organization had been breached as a direct result of a vulnerable application, and the findings are eye-opening:

- Firms suffered an average of nearly two successful attacks in the past 12 months, with only 9% stating they didn't know of any successful attacks.
- A full 69% claim their organization suffered multiple such breaches in that period—44% of them were breached twice, and the rest three or more times.

This indicates that a staggering 91% of respondent organizations were breached at least once in that period. Any breach can be catastrophic, so strengthening the security of software applications is increasingly vital.

NEGATIVE OUTCOMES

Respondents laid bare the consequences of a breach, saying the top four negative outcomes for their firms after a successful attack were:

- Customer trust decline (39%)
- System failure (38%)
- IP theft/loss (37%)
- Loss of customers (36%)

CAUSES OF BREACHES

Meanwhile, AppSec managers and software developers are closely aligned on the causes of their security breaches. Their combined top four responses were:

- Insider data leak (41%)
- Software supply chain attack (41%)
- Known, unpatched vulnerability (40%)
- Cloud application misconfigurations (39%)

Although insider data leaks are normally outside of the AppSec domain, there are ways organizations can protect themselves against software supply chain attacks, unpatched vulnerabilities, and cloud application misconfigurations.

This data tells us that organizations can directly influence the likelihood of breaches by taking care of what's in their control. It also tells us that open source supply chain security is lacking in many cases, with the likely outcome of releasing vulnerable software into production. All this brings us to the question: What are organizations doing to better manage their software-related risk? According to this data, not enough. Letting known, unpatched vulnerabilities make their way into production is like setting sail in a leaky boat. Should anyone be surprised when it eventually sinks?

BREACH RESPONSES/FALLOUT

We went on to ask what internal steps organization took to prevent similar breaches in the future. The top three responses were:

- Penetration testing exercises (38%)
- Mandatory AppSec training (38%)
- Investments in improved security tooling (37%)

These organizations likely had to spend a lot of money post-breach. Overall, how valuable is this after-the-fact investment? Most organizations would choose to spend that same money to build a better AppSec program before a breach, rather than flailing in response to board-level pressure to shore up security quickly after a breach. In the end, an organization can have a better AppSec program and save money by taking a proactive approach instead of a reactive one.

CONFIDENCE IN SECURE SOFTWARE DEVELOPMENT

Respondents aren't particularly confident that their development teams are building increasingly secure applications:

- While 77% of AppSec managers say they're "confident" in their developers' ability to write secure code, just 28% say they're "very confident."
- Meanwhile, 81% of software developers are "confident" in the security of their code, but only 30% are "very confident."
- A further 22% of AppSec managers and 18% of software developers say they're "not confident" about the overall security of the applications their firm deploys.

This confidence is surprisingly high based on the number of breaches. Is this overconfidence in the face of evidence to the contrary? Does overconfidence make organizations more susceptible to breaches? The answer is probably yes.

The amount of "not confident" responses here suggests that security is underprioritized in those organizations. What developers do or don't do is a direct result of the priorities their leadership hands down, as well as how they're measured and rewarded. Typically, developers are asked to meet deadlines. If security is a second-class priority, it comes as no surprise that security is often neglected.

91%

of respondent organizations were breached at least once in that period.

WIDENING CHALLENGES

Cybersecurity concerns keep professionals up at night, and life doesn't seem to be getting any easier. The race to digital transformation and cloud native application development is stretching everyone's ability to keep up, and respondents cited this as one of the most prominent challenges to application development. In fact, 54% declared that the shift to the cloud has made them more concerned about secure application development. The biggest challenges cited when shifting application development to the cloud were:

- Adopting cloud native security testing (37%)
- Hybrid deployment (36%)
- Upskilling developers (35%)

To reinforce these challenges, when we asked about emerging areas that should be a bigger priority for AppSec, the top responses were:

- Serverless technologies (32%)
- Containers (31%)
- Infrastructure as code (28%)
- Hybrid cloud (28%)

Organizations are clearly nervous about cloud development, and in DevOps, it's all about continuous everything. Continuous integration and continuous deployment are common terms, and cloud engineering is forcing teams to face a continuous knowledge gap. Organizations need to invest in tooling to support AppSec and developers, in addition to educational resources to help them understand the threats, challenges, and risks.

WIDENING CONCERNS

AppSec managers and software developers in combination have become more concerned about numerous factors, including attacks on their web applications (41%), security misconfigurations caused by developers (40%), and secrets unknowingly being leaked online (38%). All of these are controllable using improved secure coding practices and better AppSec awareness and training.

We delved into what might be behind organizations' widening concerns. Among AppSec managers who admitted uncertainty about their development team's ability to build secure applications, the top three reasons are:

- Lack of time to work with development teams in the context of security (40%)
- Unavailability of appropriate application security tooling (38%)
- Sophisticated nature of attacker methods (37%)

There isn't a siege mentality among software developers about these issues. They know the risks, though they have their own opinions about widening concerns:

- Growing sophisticated nature of attacker methods (39%)
- Under-prioritization of application security by the organization (37%)
- Unavailability of adequate AppSec training (35%)

TARGETED ATTACKS: A GROWING THREAT

There's a lot of chatter in cybersecurity about targeted attacks. Although the SolarWinds breach in 2020 is a particularly notorious example, many security teams remain spooked about their own software supply chains and how best to secure the open source that's becoming more prevalent in their codebases.

In comparison, respondents feel as uncertain about their organizations' ability to withstand a targeted attack as they feel about secure software development overall. Combining AppSec manager and software developer responses, 80% are at least "confident," but only 33% are "very confident" in their ability to defend against a targeted attack.

There's little doubt that the software supply chain is increasingly at risk, showcasing the need for improved visibility on what open source is being pulled into an organization's codebase. As further proof, 37% of AppSec managers and software developers are more worried about supply chain attacks than they were 12 months before the survey, telling us that open source management is a glaring concern.

To help combat the growing problem of targeted supply chain attacks, Checkmarx has acquired Dustico, a software as a service (SaaS) solution that detects malicious attacks and backdoors in open source software supply chains. Checkmarx will combine its Application Security Testing (AST) capabilities with Dustico's behavioral analysis technology to give customers a unified view into the risk, reputation, and behavior of open source packages, resulting in a more comprehensive approach to preventing supply chain attacks.

BARRIERS LAID BARE

Adequately managing cyber risk involves many moving parts, and teams encounter multiple obstacles when trying to manage and reduce the risks their organizations face. For starters, 35% of AppSec managers and 32% of software developers agree that AST slowing down development pipelines is a major stumbling block.

When asked about barriers to quick, secure app development, 35% of combined respondents cited collaboration as their top challenge around remote/hybrid work arrangements in the COVID era. Some reports indicate that individual productivity has gone up during the pandemic, but it's no surprise that it has hindered some forms of collaboration. The "new normal" situation won't be going away any time soon, so what can be done to improve things in the meantime?

For one, teams need the telemetry that modern AST solutions offer to inform them of what issues to prioritize for remediation. This telemetry can also inform teams of any repetitive coding errors so they can focus their time together on more secure coding knowledge and skills. Integrated secure coding education combined with AST results can go a long way toward improving security overall.

There is also consensus on the most common challenge to better AppSec within organizations, with 38% of AppSec managers and software developers saying developer adoption of AST solutions into their pipelines hinders progress. This is a key indicator that their current AST approach lacks integration and automation into the tools developers use daily. The ability to perform fast, accurate code analysis at the source code level will vastly improve developer adoption of AST solutions.

Other key difficulties to overcome are keeping pace with new threats (38%) and a lack of appropriate or effective AST solutions (34%). As organizations move software development to the cloud, their AST solutions need to move there, too. Source code analysis, software composition analysis, and infrastructure as code (IaC) analysis, performed directly from the cloud-based tools developers are using, become a great foundation for securing applications designed to be deployed in the cloud.

With views on everything from security breach type and frequency to vulnerabilities and internal process frustrations out in the open, it's time to tackle the challenges head-on.

Next, we'll explore how AppSec managers, software developers, and their wider organizations can approach many of the AppSec issues they face.

TACKLING APPSEC ISSUES

In today's challenging cybersecurity environment, it's a matter of when, not if, an organization will fall victim to an attack. We've touched on the security challenges firms face today, so now let's focus on what actions organizations are taking—and more pertinently, what our respondents believe they should be doing—to learn from incidents and improve security.

It seems there's already much mitigation afoot.

TRAINING

AppSec managers say their developers receive periodic secure coding education and awareness training, and software developers echo this. Comparatively, AppSec managers say they also receive some form of periodic training themselves.

Regardless of how frequent this training is, there is a disconnect on how effective it is. Only about a quarter of developers (23%) say their AppSec training is "very effective," compared to less than a fifth of AppSec managers (17%). Similarly, more AppSec managers than developers (13% vs. 9%) deem the training completely "ineffective." Most likely, the training developers are getting is completely out of context relative to their daily duties, and effectiveness KPIs and measurements are deficient.

Secure coding education remains an obvious area for improvement, and the lack of lesson availability while coding is evident. Studies show that return on investment (ROI) for proper training can be enormous. These studies also confirm that training solutions need to be integrated directly into developers' IDEs, with lessons relevant to the task in front of them. The whole point of this approach is enabling the process of launching a scan, seeing scan results, jumping to a fast and pertinent training lesson, and quickly remediating the issues at hand.

22% of AppSec managers

18% of software developers

say they're "not confident" about the overall security of the applications their firm deploys.

TESTING

Since security is a key indicator of success for both job roles, it's concerning that less than 1% of both AppSec managers and developers continuously test their applications during development to detect exploitable vulnerabilities.

Among AppSec managers surveyed, the tools and strategies they most commonly use to test and secure applications and software are:

- Infrastructure as code (IaC) scanning (28%)
- Developer AppSec training and awareness (28%)
- Software composition analysis (SCA) (27%)
- Container scanning (30%)

For software developers, the three most common tools and strategies—SCA, penetration testing, and container scanning—all stand at 30% of respondents.

Both sets of responses show a broad toolset in use for application testing, demonstrating that computer-based scanners are likely gaining ground against manual code reviews, which require considerable expertise and more time to complete. SCA, IaC, and container scanning were high on the list, which is reflective of the technologies needed to secure cloud native applications.

We went on to ask respondents about the biggest challenges they encounter while shifting application development to the cloud. In both groups, those who have shifted toward the cloud responded that cloud native testing (37%) was high on the list of challenges. The demonstrates the need for comprehensive AST solutions that integrate well with cloud-based application development.

INCENTIVES

AppSec managers have a sense of what would do the most to make training more effective:

- A greater sense of competition (37%)
- More incentives like rewards and prizes (35%)
- Having more relevant topics and content for current secure software initiatives (34%)

Software developers, on the other hand, believe training would be best improved with:

- A bigger and/or better library of content and courses (38%)
- More real-life scenarios and situations (37%)
- More relevant topics and content for current initiatives (36%)

It's not surprising that incentives are high on the list, and there's a call from both groups for more relevant topics and content for current training initiatives. This may also be about the need for more training on expanding risks in the context of cloud native initiatives. That's not surprising, either—most organizations struggle with training and security tooling that lag behind developer initiatives.

CLOSING SUPPLY CHAIN LOOPHOLES

According to Gartner, "by 2025, 45% of organizations worldwide will have experienced attacks on their software supply chains, a three-fold increase from 2021."[1] This makes the security of open source software paramount.

Collectively, respondents believe the top challenges they face in visualizing and securing the software supply chain are:

- Gaining visibility into open source packages being utilized in custom applications (26%)
- Holding third-party software suppliers accountable (25%)
- Gaining visibility into software modules being utilized (25%)

Savvy organizations are already taking steps to reduce software supply chain risk by becoming more aware of the risks they face—though there is still much more to be done.

Approaching half of respondents (44%) require a software bill of materials (SBOM) from third-party suppliers, and 86% say their organization either already has, or is completing, an SBOM internally.

Meanwhile, respondents have other weapons in their supply chain armory:

- Adopting a DevSecOps model with supply chain security as a top priority (45%)
- Applying a Zero Trust policy to open source packages incorporated into a codebase (45%)
- Requiring relevant AppSec training sessions for developers and security teams (45%)

Now that we can see what it's going to take to tackle many AppSec issues, let's look at how AppSec managers and developers can work together to make it happen.

TEAMING UP FOR SUCCESS

Both AppSec managers and software developers feel their employers could be taking a more proactive approach. Developers appear dissatisfied: 36% believe a lack of clear roles and responsibilities hinders security improvements, and 35% think C-suite buy-in (in the context of more security budget) could be stronger. AppSec managers profess these same two points, but their top issue (at 39%) is with developer adoption of security processes during software development. That's not unusual—anything that slows developers down tends to result in pushback.

There's also a disconnect when it comes to prioritizing security for emerging technologies. AppSec managers want to focus on:

- Container security (31%)
- Data privacy (30%)
- Encryption priorities (29%)
- Software supply chains (29%)

However, software developers think the priorities should be:

- Data privacy (32%)
- Technologies such as microservices (32%)
- Software supply chains (31%)
- Secure application containerization (31%)

Despite some common ground here, there is a disconnect in priorities, which can increase organizational risk if it leads to teams working in silos rather than pursuing common goals. There's clearly ground to cover before everyone is aligned on the best way forward to secure emerging technologies in the context of modern application development (MAD).

CLOSER ALIGNMENT

Still, if we delve deeper into the results, there's closer alignment than you might expect between AppSec managers and developers in some respects.

Considering who has overall responsibility for security, 56% of AppSec managers and 60% of software developers think that other groups are most responsible for the security of the applications their organization builds and deploys. Security needs to be everyone's job. Collaboration and mutual accountability are a clear path to more secure software.

Underscoring this, when asked if they had a clear understanding of their role in AppSec, 73% of AppSec managers and 74% of developers said they did. How can that be? Worse, when asked what areas were ripe for better collaboration, AppSec managers overwhelmingly asked for more testing and prioritization of vulnerability fixes, whereas developers suggested they needed more time to scope and document the impact of vulnerabilities.

None of these responses speak to collaboration. Instead, they point to the "us vs. them" culture rampant between developers and AppSec. Security needs to help developers understand and fix the vulnerabilities, and developers need to spend more time fixing the mistakes instead of documenting them. These two teams need to stop pointing fingers at each other so they can begin to recognize that they're both responsible for the security of their applications.

GETTING THE RIGHT TOOLS

Having appropriate AST tools in place empowers AppSec teams and developers to be confident that the code they deploy is as secure as possible.

However, the AST market is flooded with disparate tools that don't fit well into MAD approaches. These tools end up producing vast amounts of uncorrelated risk data, which makes it more difficult for teams to quickly remediate vulnerabilities, ultimately slowing down the development pipeline.

This is reflected in what AppSec managers say are the biggest challenges with their current AST tools and strategies:

» Too many disparate tools (37%)

» Lack of correlated risk data (37%)

» Difficult or limited CI/CD integrations (37%)

Meanwhile, the biggest challenges for software developers are:

» Lack of correlated risk data (42%)

» High number of false positives (41%)

» Difficulty of integration with existing development tools (40%)

What we can see here is that teams need a comprehensive AppSec platform that easily integrates with the development tooling in use and works automatically. It also needs to provide correlated risk data and centralized scan findings, adjustable queries to reduce false positives, a broad set of integration capabilities, and access to the tools developers and AppSec teams both want and need.

CONCLUSION

Based on our findings, there's a lack of security confidence across organizations as they continue to suffer breaches. The threat landscape is widening and evolving, and targeted attacks on organizations and their software supply chains are becoming more frequent and intense. Better training, advanced integration of AST solutions, developer incentives, straightforward roles and responsibilities, and closer alignment between security and development will all help remedy the issues software-driven organizations face.

However, as organizations begin to adopt MAD practices to accelerate their digital transformation, they will face these same issues with greater intensity. MAD spans a panorama of technologies in which critical, revenue-generating applications are no longer designed to operate in an antiquated, monolithic way.

Today's modern applications, designed primarily for cloud native deployments, are composed of microservices, containers, APIs, infrastructure as code, and lots of open source libraries and components. The legacy AST systems and approaches many organizations take can't keep up—they need a new approach to building in security during software development.

At Checkmarx, we're using our forethought, expertise, and years of experience securing the world's most influential organizations to bring the most advanced and innovative AST platform—perfectly designed for cloud native and MAD approaches—to a market that desperately needs it.

Imagine a single AST platform that integrates perfectly with modern, cloud-based tooling and automates all the necessary functionality to secure applicative code (e.g., microservices, APIs, open source), container code, infrastructure as code, and the software supply chain. This same solution elevates confidence, radically improves security, promotes knowledge and awareness, and speeds up development and deployment. It saves time, reduces costs, and enables modernization to flourish.

That solution is available now. Welcome to the new age of application security!

Click here to learn more about the Checkmarx Application Security Platform™.

ABOUT CHECKMARX

Checkmarx is constantly pushing the boundaries of Application Security Testing to make security seamless and simple for the world's developers while giving CISOs the confidence and control they need. As the AppSec testing leader, we provide the industry's most comprehensive solutions, giving development and security teams unparalleled accuracy, coverage, visibility, and guidance to reduce risk across all components of modern software—including proprietary code, open source, APIs, and infrastructure as code. Over 1,600 customers, including half of the Fortune 50, trust our security technology, expert research, and global services to securely optimize development at speed and scale. For more information, visit our website, check out our blog, or follow us on LinkedIn.

[1] "How Software Engineering Leaders Can Mitigate Software Supply Chain Risks," Gartner, 15 July 2021.