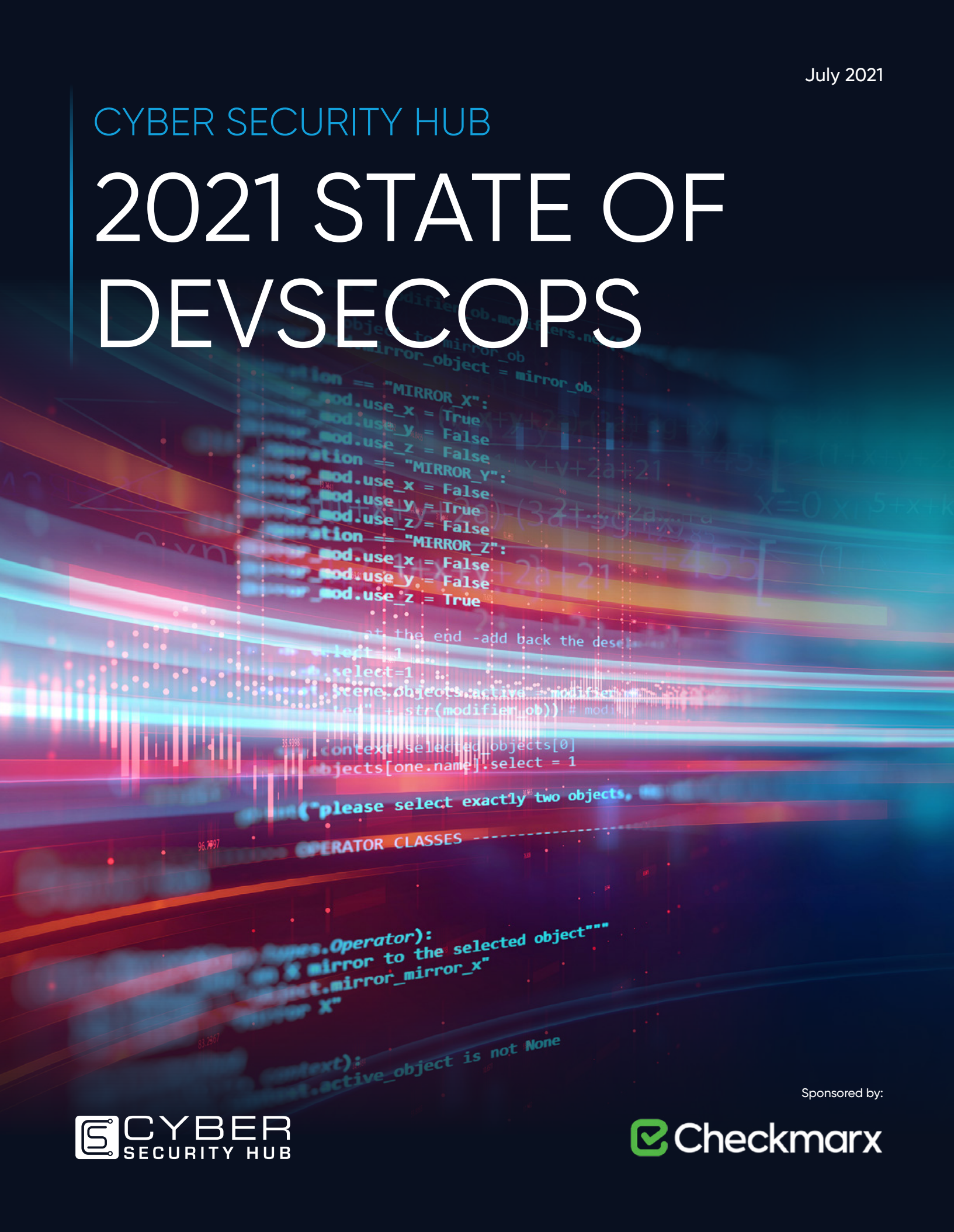
July 2021

CYBER SECURITY HUB

2021 STATE OF DEVSECOPS



Sponsored by:

CYBER SECURITY HUB

Checkmarx

Introduction

Today, DevOps teams are embracing new approaches of embedding application security testing (AST) into their development pipelines to truly achieve DevSecOps. Developer and security teams agree that a single vulnerability scan late in the software development life cycle (SDLC) is insufficient, can result in delayed software releases, and often becomes unnecessarily expensive. As with any type of security testing, discovering vulnerabilities late in the pipeline means it takes more time and effort to remediate them. Hence, it's highly desirable to shift AST as far left as possible by integrating security into the tooling developers use.

However, DevSecOps is still in its early stages, as evidenced by our 2021 survey, which indicates that the security portion of DevSecOps is maturing in pieces. Part of the problem is budget, and part of it has to do with a talent shortage. Smaller companies especially lack the security expertise and tools larger companies have.

Some of the key findings from our 2021 survey are:

- Half of respondents plan to add more application security staff in the next year.
- 43% say their biggest frustration is security testing done late in the SDLC because it delays launches.
- Of all the application security types, API testing is the most popular among two-thirds of respondents.
- Two-thirds lack a security champion.

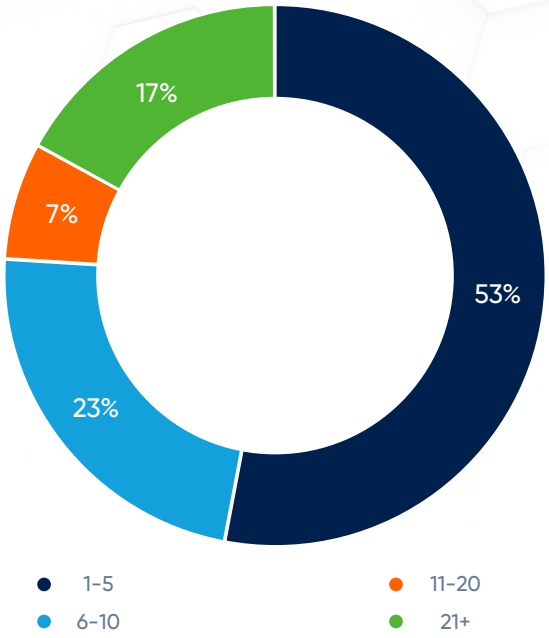
Methodology

This report is based on a global online survey of application security professionals that took place during late May and June 2021. Sixty percent are located in EMEA, followed by APAC and the Americas. Forty-five percent hold director, VP, or C-level titles.

How big is your application security team?

There are always far fewer application security team members than developers. Developers outnumber security professionals 100 to 1 in large organizations. More than half of respondents (53%) said they employ one to five application security team members, and generally speaking, the lower number is generally associated with small and medium businesses (SMBs).

However, typically, as the number of application security team members rise, so does the size of the organization. The number of companies employing 21 or more such professionals outnumbers those that employ 11 to 20.



Do you plan on adding additional staff to your application security team within the next year?

Half of respondents plan to add more security staff, with the majority (38%) adding just one to five more. A non-trivial factor is the shortage of security talent.

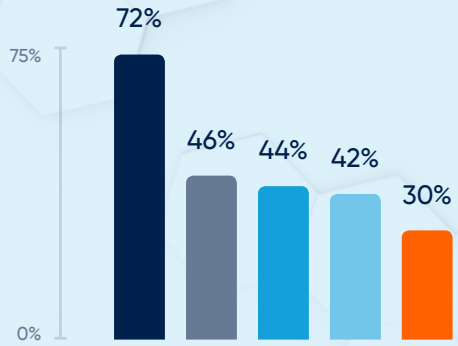
"There's always a shortage of experienced people, and that's the challenge," said Stephen Gates, security evangelist and senior solutions specialist at Checkmarx. "That's a hurdle organizations are going to have to deal with even though they want to double the size of their staff."

Checkmarx | CYBER SECURITY HUB

How do you measure a successful application security testing program?

*Respondents could select more than one option

- Overall reduction of all types of vulnerabilities
- Specific reduction of vulnerabilities (such as SQL injection and/or XSS)
- Number of vulnerabilities identified (but not fixed)
- Increased velocity of releases with application security testing integrated and automated
- Decrease in bug bounty efforts



Nearly three-quarters (72%) view vulnerability reduction as the key measure of success. Fewer (46%) are focused on particular vulnerabilities they've identified previously.

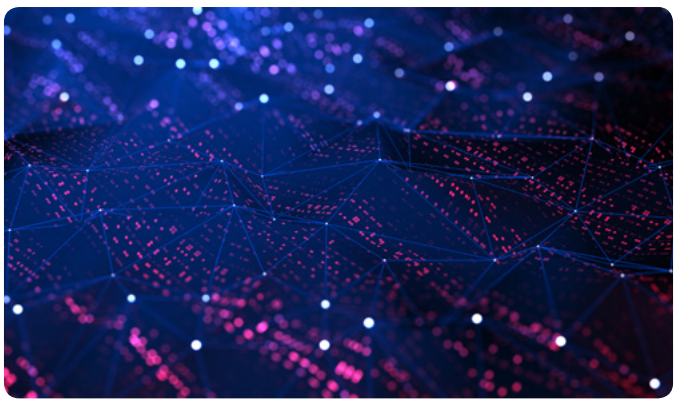
The scary statistic is that 44% measure success based on security vulnerability identification, not remediation. About the same number (42%) are achieving release speed and quality by integrating security testing and automation, typically in a CI/CD pipeline that is highly automated otherwise.

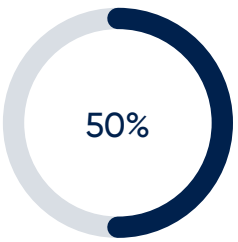
Three in 10 have reduced bug bounties because they likely don't want to advertise the fact that they have software vulnerabilities, and they don't want people hacking their systems to find bugs. On an industry level, companies generally are decreasing bug bounties.

What kind of reporting is given to the CISO?

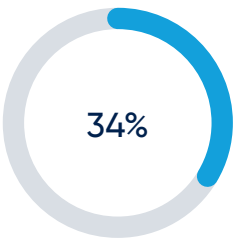
*Respondents could select more than one option

Half of respondents are providing the CISO with a PDF report of security scan findings, some of which include custom charts. The people who are not providing the CISO with any report may lack the necessary tools or time to do it, or the company may not have a CISO in the first place.

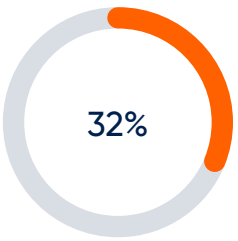




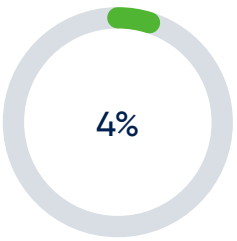
PDF report with findings



Custom charts created by the AppSec team



No report given

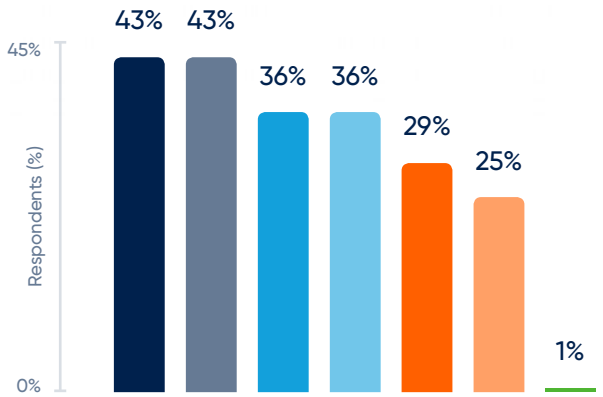


Other

What is your biggest frustration with application security testing?

*Respondents could select more than one option

- Too late in life cycle - delays launch
- Understanding the vulnerability findings - what was found, how risky, how to fix it?
- Too many false positives
- Tracking vulnerability status - verified? Scheduled for remediation? Fixed?
- Prioritizing vulnerability remediation
- Finding who can perform the remediation
- Other



There are several frustrations with application security testing, and one of the most prominent is testing too late in the life cycle (43%). A quick vulnerability scan is not enough to address all application vulnerabilities, and it often results in interpersonal conflicts between developers and security since it requires developers to revisit code after they've moved onto another project. From an overhead perspective, it's more time-consuming and expensive.

"When people say a vulnerability is being identified too late, is a clear indication that integration and automation of AST solutions within their development pipelines is less than perfect," said Gates. "Then, just before deployment, they launch a scan and then say, 'Oh wow, we have lots more work to remediate the vulnerabilities we've discovered'."

An equal number (43%) are frustrated by their inability to interpret the scan findings, how risky a suspected vulnerability may be, and how to quickly fix it. There's a gap between understanding a vulnerability and its remediation.

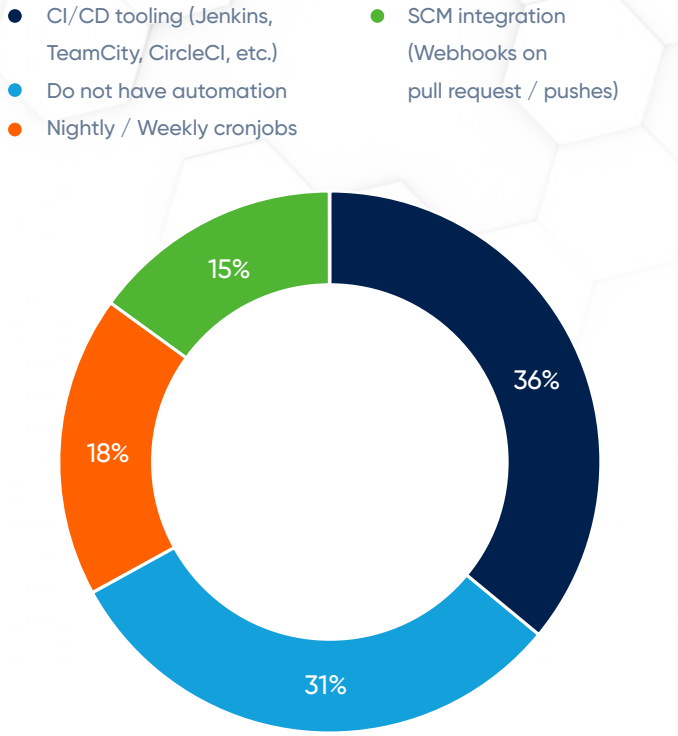
A related issue is tracking the vulnerability status to make sure it's actually being fixed (36%). Teams often lack the tools or tool integration they need to track the status of vulnerability remediation in a quick and easy way.
Also at 36% are false positives, which waste precious time. Alternatively, when false positives are too frequent, people tend to ignore them (which is called "alert fatigue"). Alert fatigue is not only very real, but also potentially dangerous because some true positives may be ignored.

How are you currently (or planning) automating application security testing?

Time to market pressure necessitates automated AST, but it also helps ensure code quality. Those with CI/CD pipelines (36%) are automating or plan to automate security testing as part of a larger automation strategy since CI/CD and automation go hand in hand. Another 15% are relying on SCM integration.

"Source code management integration is something people either don't know about or they don't have the correct application security testing tools that can integrate directly within their SCM tools of choice," said Gates. "Scan early and scan often is the way to work. Within the SCM is the best location to launch scans since it's the often the farthest left you can go."

Automation is a maturity issue that requires AST integration within the dev tools in use, which some organizations clearly lack (31%). However, modern AST solutions can easily be intergrated and fully automated.

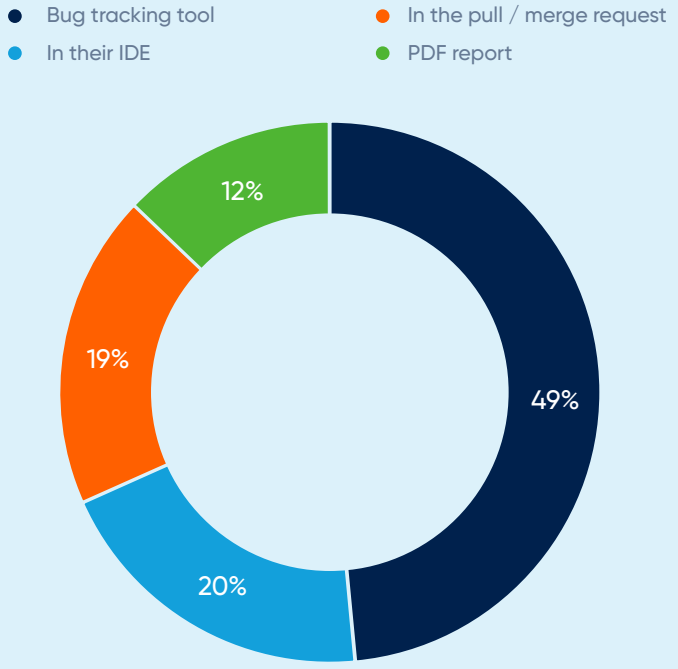


Where is the best place developers consume application security results?

Half the respondent base considers a bug tracking tool like Jira the best place to consume application security results because they already have the tool in place, with both develops and security already using it. A distant second (20%) is in an IDE, because modern IDEs show coding errors in real time, though they won't catch everything because they were not designed specifically to identify application vulnerabilities.

About the same number (19%) prefer a pull/merge request.

"If you automate the pull request, then you can actually get the results right back in the SCM tool you're working on, and what's nice about that is it's very fast," said Gates. "Developers have the opportunity to remediate quickly when they're sitting in a branch of code or working on it. It's just expected."



What type of security testing are you performing?

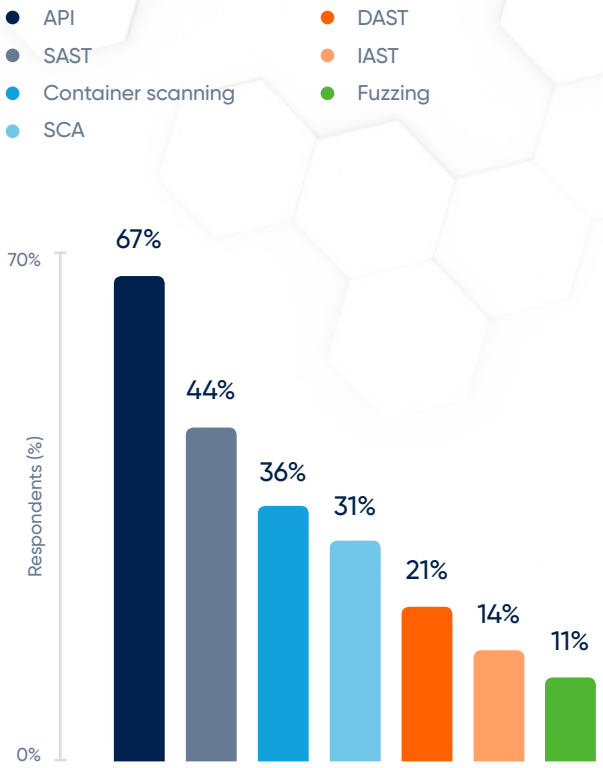
*Respondents could select more than one option

Two-thirds of survey participants (67%) are doing API testing because it's becoming an increasing threat. In fact, Gartner predicts that APIs will become the most popular attack vector by 2022. Another reason API testing is popular is because modern applications have a lot of dependencies. Today's applications rely on a number of outside functions and data to operate as intended. Those dependencies, if they contain security flaws, may infect an application. In addition, cloud native development and the use of microservices are growing exponentially, so API vulnerabilities are of particular concern.

Static application security testing (SAST) is more than twice as popular (44%) as dynamic application security testing (DAST, at 21%) because DAST doesn't always fit well into modern application development practices, which emphasize speed.

When organizations start decomposing applications into microservices and using containers—or building new applications with containers—they eventually realize they need special tooling and updated practices since traditional application security tooling and practices won't work. Each container represents an individual attack surface, unlike a monolithic application. The 36% using container scanning realize this.

Finally, the use of software composition analysis (SCA) tools is on the rise because of open source security concerns. Today's applications use more third-party commercial and open source components and libraries than ever before because of time to market mandates and an increase in application complexity. Without SCA, it can be more difficult to pinpoint open source vulnerabilities and license risks.

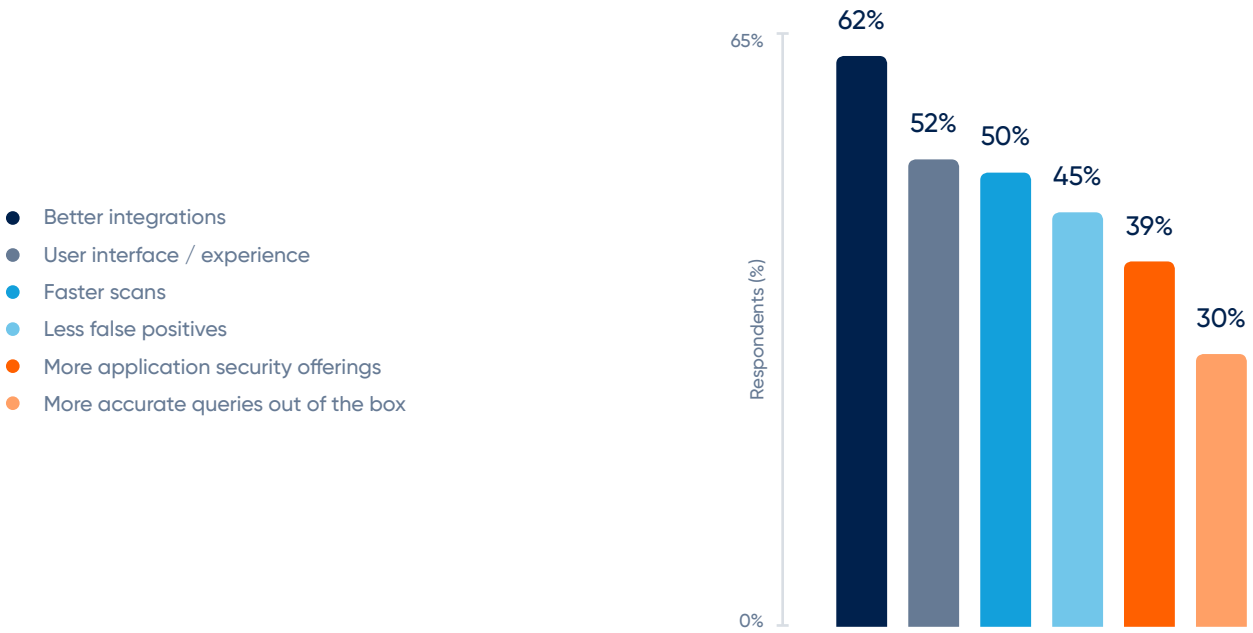


Do you use any open source tooling specifically for finding security vulnerabilities?

Despite the increased use of open source code, 58% said they're not using open source security testing tools. "That's a concern because of the increase in the amount of open source modules, packages, and libraries these teams are using," said Gates.

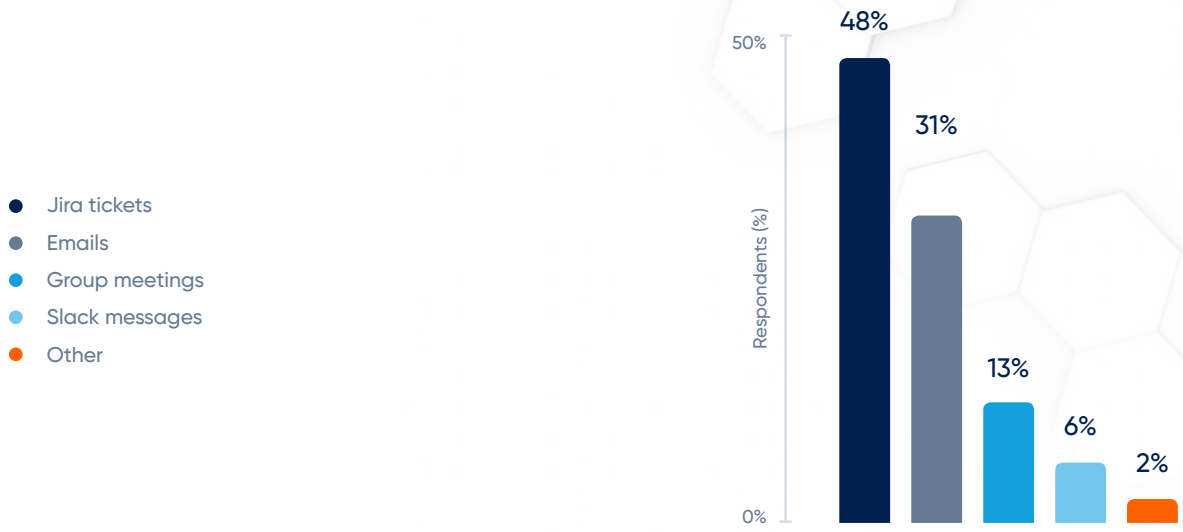
What improvements in application security testing would you like to see?

*Respondents could select more than one option



Most respondents (62%) chose better integrations as the top improvement they'd like to see because of all the friction a lack of integration causes. Insufficient integration precludes visibility across tooling and it prevents teams from achieving the level of automation they need to improve release velocity and product quality. Also causing friction are less-than-optimal user experiences (52%). Faster scans (50%) and fewer false positives (45%) are also about speed.

How do you communicate with developers on bugs found?



Jira tickets are the top means of communicating with developers (48%) since so many DevOps teams already use it.

"Integration and automation into Jira systems is probably vastly desired because of the need for speed, which extends to triage, troubleshooting, and remediation," said Gates. "There's a desire to have automation and integration with the actual scans to open and close tickets automatically."

Email is an antiquated way of communicating information about code issues, but it's still popular among 31% of respondents since everyone uses email but not everyone uses Jira. While email can accommodate all the information a developer will need, unlike Slack, emails are not a real-time communication medium, and they can get lost easily in a universe of other emails. It can also be difficult to search through emails to find out what the status of remediation is.

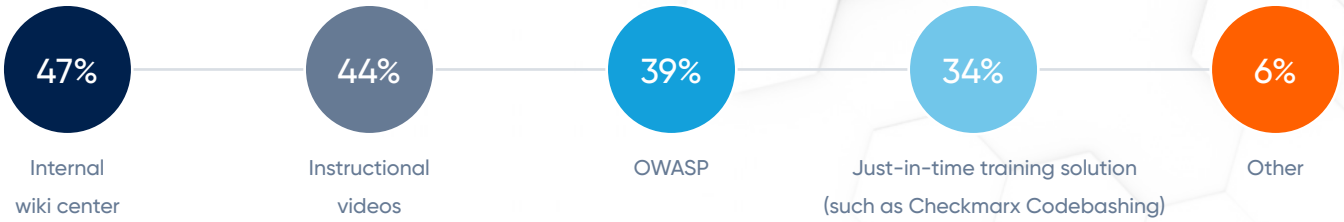
Group meetings have the benefit of getting everyone on the same page (13%), but they take time away from other day-to-day tasks.

Do you currently use a Security Champion type of model within your organization?

Another scary statistic is that two-thirds of respondents (66%) lack a security champion, when that's such an important role. The security champion evangelizes security within the organization, which helps create a cyber-aware culture. Since they're security experts and also good communicators, they're a good resource for mentoring developers. The security champion is the designated go-to person developers and others can tap when they have a security-related question.

What do you currently do for security awareness?

*Respondents could select more than one option



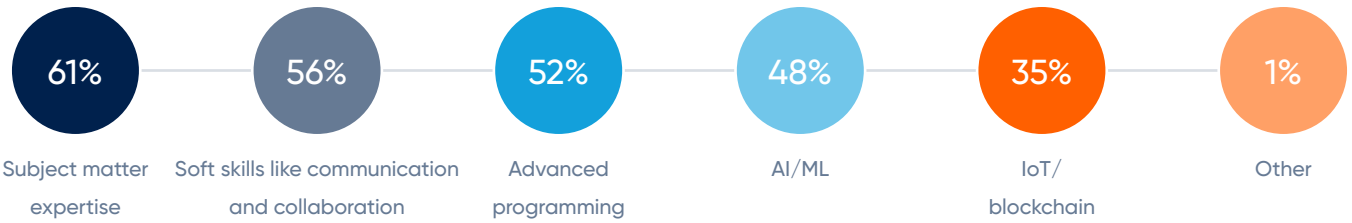
Wikis are alive and well in a security context. Nearly half of survey participants (47%) are using a wiki as the primary vehicle for security awareness, with just slightly fewer (44%) saying they use instructional videos.

Fewer still are those using the Open Web Application Security Project (OWASP), an online community providing freely available web application security articles, methodologies, documentation, tools, and technology. Then again, OWASP is specific to web application security, and the guidelines tend to be quite general.

About one-third have integrated real-time training, which helps accelerate application release speed and quality. Unlike videos, which take time away from coding, real-time training teaches developers application security within the context of writing code. If a developer doesn't know how to remediate an issue, a five-minute tutorial will often solve the problem.

What skills will be important for security professionals to have in the future?

*Respondents could select more than one option



Subject matter expertise tops the list (61%) of what respondents think security professionals will need to have in the future. Security professionals tend to be driven to upskill themselves because they want to advance their careers, often into the role of a security champion or CISO.

However, to become an effective security champion or CISO, they need to have technical expertise and soft skills (56%), such as communication and collaboration, because the role interacts with other roles in the company.

Advanced programming skills (52%) does not mean that security professionals are morphing into developers (though some may choose to do so). It's more about understanding the application side of application security as much as the security side of application security and being able to tell a developer what to do.

AI and machine learning are also considered important knowledge to have (48%), given their growing popularity throughout enterprises. At first, enterprises raced to implement AI and ML to reap the benefits, but more realize now that they must also manage the potential risks not only with the help of data scientists, but security professionals as well.

Meanwhile, IoT and blockchain use is emerging, as reflected by the 35%.

Is the speed of an application security testing scan more valuable than the quality of results?

It's encouraging that 64% of respondents consider the quality of application testing more important than speed, because it speaks to the desire to provide higher quality software and avoid the headaches of a customer or bad actor discovering security issues in production.

The result may seem counterintuitive based on some of the other survey responses that demonstrate a desire to remove obstacles to speed. However, given the choice between speed and quality, quality prevails because it equates to more secure code.

Conclusion

Application security continues to rise in importance because it represents legal, regulatory, and brand issues. End users expect applications to be of high quality, which includes security. It's not enough to simply build and deliver applications faster. Speed and quality must go hand in hand.

As always, individual respondent companies are at different stages of maturity because speeding delivering and increasing code security simultaneously is a journey. Moving from DevOps to DevSecOps helps.

About Cyber Security Hub

The Cyber Security Hub is an online news source for global cyber security professionals and business leaders who leverage technology and services to secure the entire perimeter in their enterprise.

We're dedicated to providing the latest industry news, thought leadership and analysis in the cyber security space. Cyber Security Hub's expert commentary, tools and resources are developed through obtaining data and interviewing end users and analysts throughout the industry to deliver practical and strategic advice.

Our editorial team surveys and monitors the latest trends in cyber security and creates news articles, market reports, case studies and in-depth analysis for a captive audience consisting of C-Level executives, VPs and directors of cyber security and information technology.

Cyber Security Hub Team

Dorene Rettas
Managing Director
Dorene.Rettas@CSHub.com

Seth Adler
Editor-in-Chief
Seth.Adler@iqpc.co.uk

Joshua Snead
North America Sales Director
Joshua.Snead@iqpc.com

Tilak Antony
Director of IQPC
Digital Partnerships
Tilak.Antony@iqpc.com

Imran Shafi
Sales Director
Imran.shafi@iqpc.com

Desiree Santiago
Marketing Manager
Desiree.Santiago@cshub.com

Social Media Information

Facebook:
CSHubIQPC

Twitter:
CSHubUSA

LinkedIn:
CSHub – Enterprise Security Professionals

Visit CSHub.com for more information from cyber security leaders for the cyber security community

The world runs on code.
We secure it.

Checkmarx is constantly pushing the boundaries of Application Security Testing to make security seamless and simple for the world's developers and security teams. As the AppSec testing leader, we deliver the unparalleled accuracy, coverage, visibility, and guidance our customers need to build tomorrow's software securely and at speed.