NowSecure™ DevOps.com

The State of Secure Mobile & Web App Development

SURVEYING THE JOURNEY TO DEVSECOPS • SPRING 2019

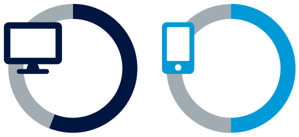
Executive Summary

While the industry has conducted numerous studies on DevSecOps practices and application security readiness across a broad portfolio of apps, until now there's been very little insight into the state of DevSecOps specifically within mobile apps.

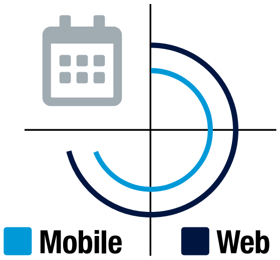
In order to better understand the differences and similarities between mobile and web apps in DevSecOps maturity, NowSecure and DevOps.com teamed up to ask more than 200 IT practitioners about their software delivery and application security testing practices across their entire software portfolio spanning web and mobile apps. The survey offered some illuminating highlights:

FOR BOTH MOBILE & WEB APPS, ORGANIZATIONS ON THE DEVSECOPS JOURNEY ARE AT OR PAST THE TIPPING POINT OF PRODUCTION DEPLOYMENTS:

- **56% of web apps and 50% of mobile apps** are developed through DevOps or DevSecOps practices today.

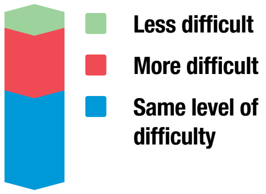


- **71% of web apps and 69% of mobile apps** are on a monthly or faster release cycle.



MANY ORGANIZATIONS SUFFER VISIBILITY GAPS IN THEIR MOBILE DEVSECOPS PRACTICES AND FEW CAN DELIVER MOBILE APPS MORE SECURELY THAN WEB:

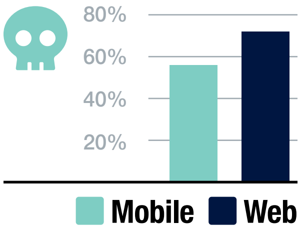
- **Almost 90 percent of organizations** say they have the same level or more difficulty securely delivering mobile apps compared to web apps.



- Of substantial concern, **42% of respondents don't know how often they test their mobile apps for security problems.**

MOBILE DEVSECOPS PRESENTS UNIQUE CHALLENGES FOR ORGANIZATIONS:

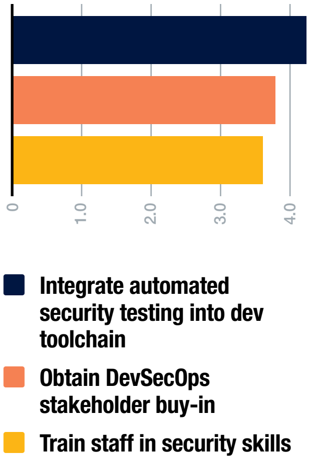
- **Only 56% of organizations can remediate a high severity security vulnerability** in mobile apps within one month, compared to 72% for web.



- **Lack of automated tooling** was consistently named as one of the biggest challenges for implementing mobile DevSecOps as well as for security testing mobile apps.

AUTOMATED TESTING IS THE NUMBER ONE SUCCESS FACTOR WHEN COUPLED WITH PEOPLE AND PROCESS CHANGES:

- **Integrating automated security testing into the dev toolchain** is the top named success factor for DevSecOps



- **Obtaining stakeholder buy-in,** and **training staff in security skills** are critical as well

© 2019 NowSecure & DevOps

State of Mobile DevSecOps

NUMBER OF APPS

In order to set the stage for the rest of our findings, we asked respondents to tell us how many web and mobile apps they develop and maintain. Our sample showed a diverse mix of application portfolio volumes. For web apps we saw that the highest percentages were bookended on either side of the scale, with the bulk of organizations either maintaining under five apps or more than 60. For mobile, the largest segment is in one to five apps, but interestingly 13% have 60 or more mobile apps under development or maintenance. (*see fig. 1*)

Meanwhile, there's clearly a severe lack of visibility into the mobile application portfolio for many organizations, where one in four don't know per this survey data set. This is a consistent theme that we'll explore throughout the survey results, which points to a troubling lack of visibility into the mobile space and could indicate potential governance and process issues if organizations were to dig further to understand their unknowns.

For the sake of transparency and discussion, it's important to note the high prevalence of these 'don't know' answers, but it's helpful to also look at the data with those answers removed to better understand ratios of those apps which are known. As shown below, adjusting for the outliers of no mobile app at 10% and 29% of web is more than 40 apps, most bands are comparable distributions. (*see fig. 2*)

FIGURE 1. How many apps does your organization develop and/or maintain?

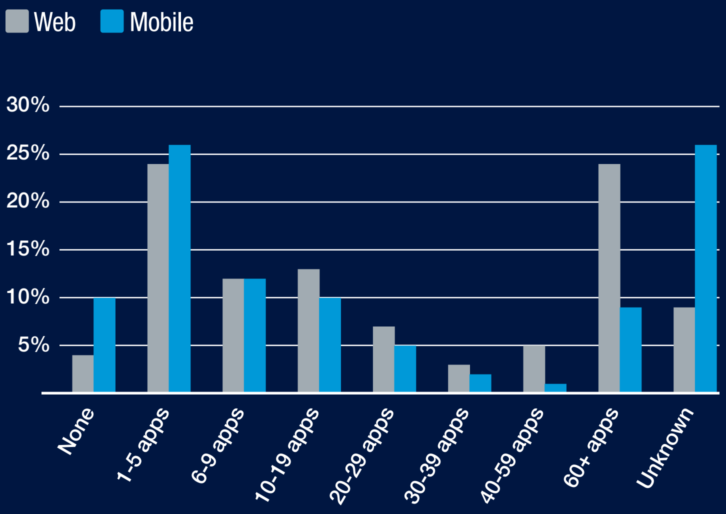
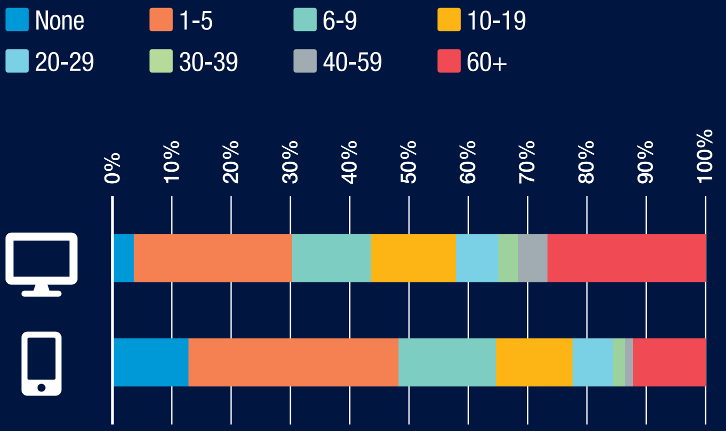


FIGURE 2. Number of apps



© 2019 NowSecure & DevOps

DEVSECOPS MATURITY

In asking about DevSecOps plans at respondents' organizations, it is clear that DevSecOps is in a nascent stage for both application types. Our first glance at the data shows that many organizations still have little visibility into where their organizations are with regard to DevSecOps maturity when it comes to mobile apps. Nearly one in three organizations reported they didn't know what the plans were to adopt DevSecOps. There are a couple of theories for this lack of visibility. Because DevSecOps requires a high degree of commitment across teams, it likely indicates a low degree DevSecOps maturity at these organizations. But it could also be a reflection of the respondent's role in the organization or perhaps that mobile teams are segregated from web application teams. (*see fig. 3*)

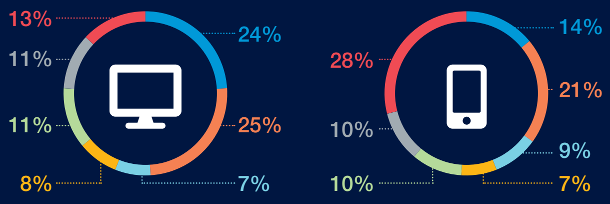
Overall including the "don't know" answers, we find that nearly one in six organizations operate mobile development in full DevSecOps mode vs. web at a rate of nearly one in four organizations. While this is a significant population, it's smaller than one would expect of mobile teams who may have started more recently than older web projects and may have initiated their program with DevOps due to speed to market pressures. This may just reflect the general lack of security embedded in mobile development because if you add those in full DevOps mode that are still adding security the result jumps up to one in three organizations engaged in these practices.

We thought it'd also be useful to take another look at the data with 'don't know' answers removed to get a clear look at where organizations are along the maturity curve among those with full visibility in to their organizations' processes. (*see fig. 4*)

This look offers a higher fidelity comparison between web and mobile apps' progression in DevSecOps. It shows that while web apps have definitely had a head start in the DevSecOps maturation process—they lead by about 7% in full DevSecOps mode—all of the other categories are within one point of one another.

It's also interesting to note that you can see that those organizations knowingly on a DevSecOps journey are at or past a 50% tipping point when it comes to being beyond the pilot stage. That's a striking statistic pointing to the momentum building within organizations across their application portfolios.

FIGURE 3. Which best describes your organization's plans to adopt DevSecOps for app development?



🟦 We have already invested and operate in full DevSecOps mode

🟧 We are engaged in DevSecOps but have not fully folded security into all aspects of the process

🟦 We are currently in DevSecOps pilot project(s) and plan to scale across our organization within 12 months

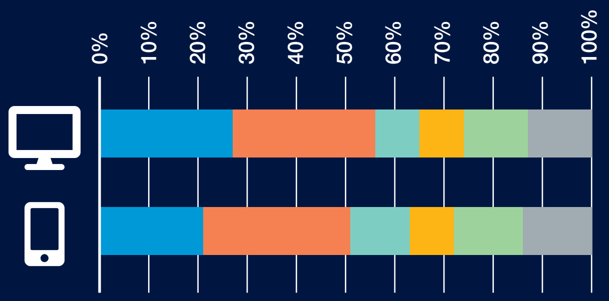
🟨 We are currently in limited DevSecOps pilot project(s) and plan to scale across our organization within 12-24 months

🟩 We are still in planning and are 24 months or more away

⬜ No plans

🟥 Unknown

FIGURE 4. State of DevSecOps removing unknowns (see legend above)



© 2019 NowSecure & DevOps

FREQUENCY OF RELEASE

So given the headway many organizations are making in DevSecOps maturity, how is that impacting release cadence? In fielding answers about frequency of release, we again saw a fairly even distribution of release practices. Clearly, continuous delivery practices are present in only a small advanced group of organizations. A slim number of organizations release more than once per day or even on a daily basis, particularly within mobile apps. As before, we note a serious lack of visibility into mobile release practices (*see fig. 4*).

Nearly two in five organizations reported they didn't know the cadence of their mobile delivery, which is a notable gap.

In order to get a clearer picture of the comparative practices we again stripped this data set of the 'don't know' answers (*see fig. 5*).

What we see here is a natural bell curve, with the most prevalent cadence being a monthly release cycle, common among both web and mobile applications – with weekly and quarterly runners up. As you can see, web applications are most likely to be released at a weekly or faster rate than mobile, with about 42% of web apps making this benchmark compared to 35% of mobile apps. But interestingly if we slice the data to monthly or faster cadences, web and mobile apps are neck and neck, at 71% and 69% respectively.

This speaks to the success of DevOps adoption overall, though we do need to note our sample was predisposed to be interested in DevOps based on how we surveyed vs. a broader market survey panel.

These offer some interesting results in aggregate, but that doesn't necessarily tell us how speeds vary within any given organization. To gain visibility, we examined individual results to find out how often mobile and web release speed varied within a given organization. We compared respondents' answers when the release cadence of both web and mobile was known. Interestingly, respondents indicated that just shy of two-thirds of organizations release mobile and web applications at the same frequency (*see fig. 6*).

When there was a difference, though, web applications were almost three times as likely to be released more frequently than mobile applications. This could indicate that mobile app development isn't necessarily a faster, leaner or more nimble process than web app development. It's hard to say definitively because the number of 'don't knows' made the sample size relatively small here (just shy of 100 responses), but it's definitely a conversation starter.

FIGURE 4. Frequency of app release cycle

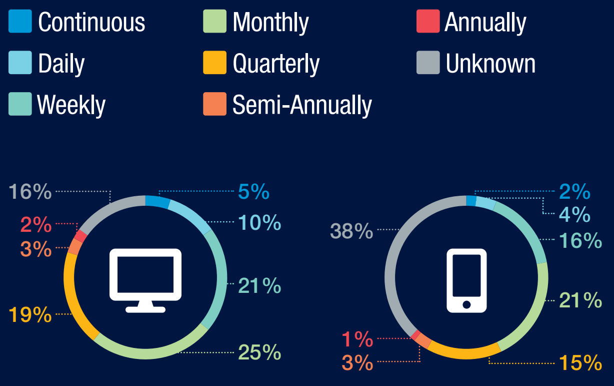


FIGURE 5. Frequency of app release cycle removing unknowns (see legend above)

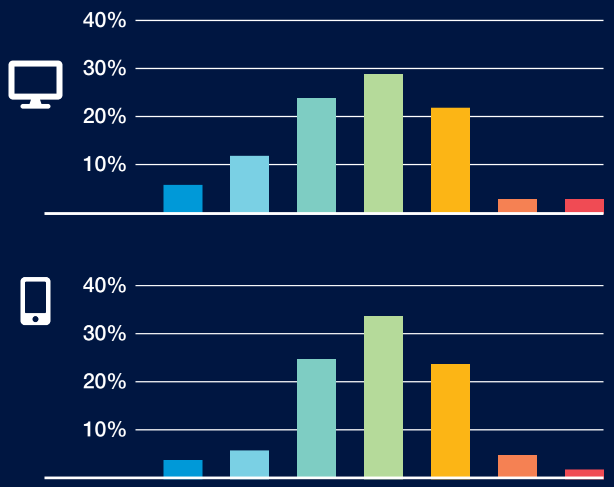
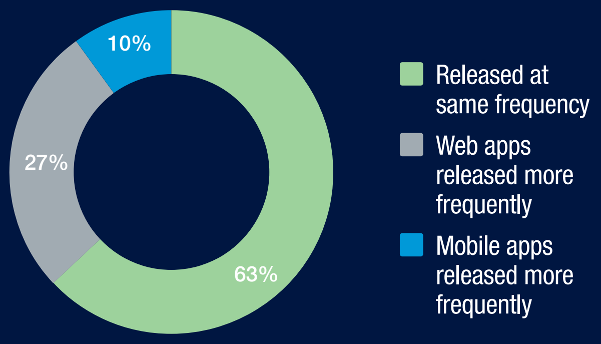


FIGURE 6. When cadence of both web and mobile is known



© 2019 NowSecure & DevOps

DEVSECOPS ENABLERS AND BLOCKERS

DevSecOps requires a cultural and technological transformation that touches on a mix of people, processes, and technology. In order to understand the importance of these different levers in affecting real change and successfully achieving DevSecOps maturity, we asked a few questions about the top enablers and blockers for DevSecOps transformation.

First up, we had organizations rank a list of common DevSecOps enabling activities based on their perceived impact for successfully implementing DevSecOps (*see fig. 7*).

The answers show that integrated, automated security testing clearly stands out, but gaining buy-in, training people effectively, and identifying champions are crucial, people-centric activities.

Meanwhile, when we looked at barriers to success (*see fig. 8*), we saw that people and process issues top the list, making up the first half of ranked issues. Tooling and management support issues were further down list. So, while tooling is a crucial part of success, when people and process issues aren't resolved the barriers may be insurmountable even with the best toolchain at an organization's disposal. Bringing the two sets of data together, this indicates overall that a mix of people, process, technology are crucial for success.

Additionally, for both of these answer sets the rankings were relatively close with one another, adding another indicator that there's no silver bullet when it comes to DevSecOps. At the end of the day, change is hard. Successfully implementing DevSecOps requires a transformative journey, not a product in a box, so it takes balance on all fronts.

FIGURE 7. Activities for successfully implementing DevSecOps (ranked 1 - 6)

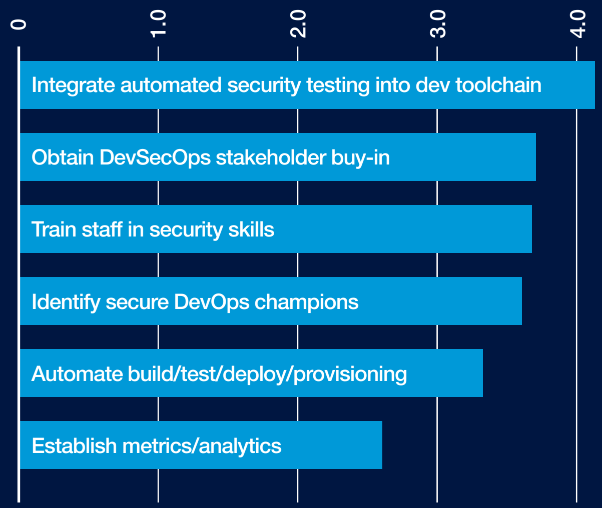
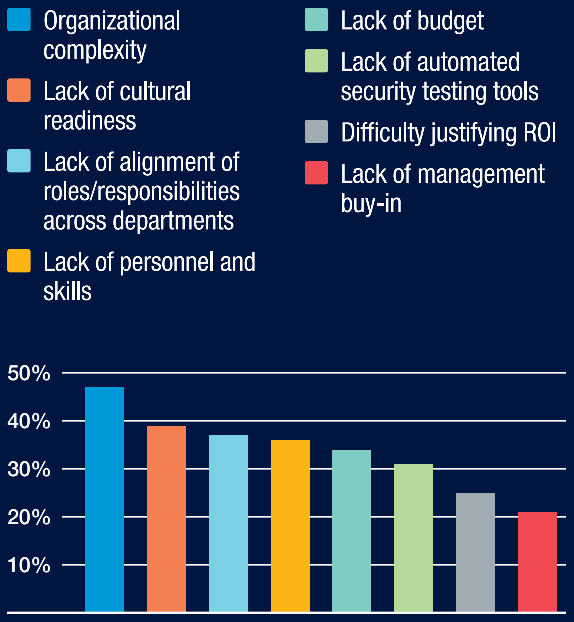


FIGURE 8. Primary inhibitors of DevSecOps within organizations



© 2019 NowSecure & DevOps

State of Mobile App Security

Now that we've framed the state of Mobile DevSecOps, let's dive specifically into the state of security in both web and mobile environments.

REASONS TO INTEGRATE APPSEC TESTING WITH DEVSECOPS WORKFLOWS

In examining the perceived benefits of integrating application security testing within any given software development lifecycle, risk reduction and quality improvements unsurprisingly top the list, but there are plenty of secondary benefits. (*see fig. 9*)

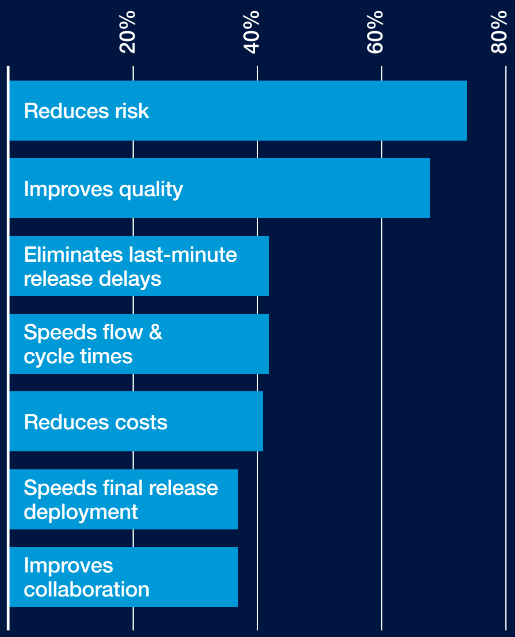
Though security can frequently be viewed as a speedbump by developers and business stakeholders, the fact is that many are seeing that integrated security testing can actually speed DevOps delivery. This flips the notion of security being a blocker of faster cadence on its head. In many cases security acts as a delivery accelerator when security testing is well integrated.

For example, 42% of respondents said that integrated testing can eliminate last-minute release delays and 37% say it speeds the overall final release deployment. This is because you'll see far fewer 'break the build' and 'pull-the-plug' security emergencies found by the security team prior to deployment when you're testing earlier and more frequently in the process.

It's beneficial to think about this in the context of the business objectives driving software delivery. Organizations release mobile apps to grow revenue, serve customers, beat competitors, attract new customers, and so on. Release slowdowns and delays can cause millions of dollars of losses in business value no matter what the delays are or why they've occurred. So reducing security friction can have huge economic and business benefits on this front.

Getting back to risks and quality, though, the fact that these two issues pop out of the data so clearly goes to show that they are two sides of the same coin. At the end of the day, security vulnerabilities are bugs and organizations need to treat them as such. When organizations integrate application security testing and closed-loop security bug tracking/remediation into the overall software delivery process, they are thereby raising the bar on software quality as a result.

FIGURE 9. Benefits of integrating appsec testing within SDLC workflows



© 2019 NowSecure & DevOps

TESTING FREQUENCY: COMPARING MOBILE TO WEB

Security testing practices still vary widely within organizations, as we'll see within the next few data sets. Once again, when queried about the frequency of testing we see a big visibility gap when it comes to mobile app security testing practices. Twice as many organizations don't know the frequency of security testing for mobile apps as those who don't know their testing cadence for web apps. More than likely, this indicates little to no testing maturity at these organizations (*see fig. 10*).

When removing the 'don't know' answers, we can see a fairly similar distribution between web and mobile, though it's notable to see that web is visibly more likely to be testing continuously or daily than mobile. Testing mobile apps can be harder than web for a variety of reasons such as complex underlying technology, difficulty instrumenting mobile OS and apps, lack of mobile-specific security skills and limited set of specialized mobile tools (*see fig. 11*).

We can see some of the aggregate differences with a different view of the data.

Although mobile trails web, the general pattern of distribution is similar here with heavier volume for both in the less frequently tested ranges. While 41% of web apps are tested at least monthly, only 31% of mobile apps are tested at that frequency.

FIGURE 10. Comparing web & mobile testing frequency, security and privacy

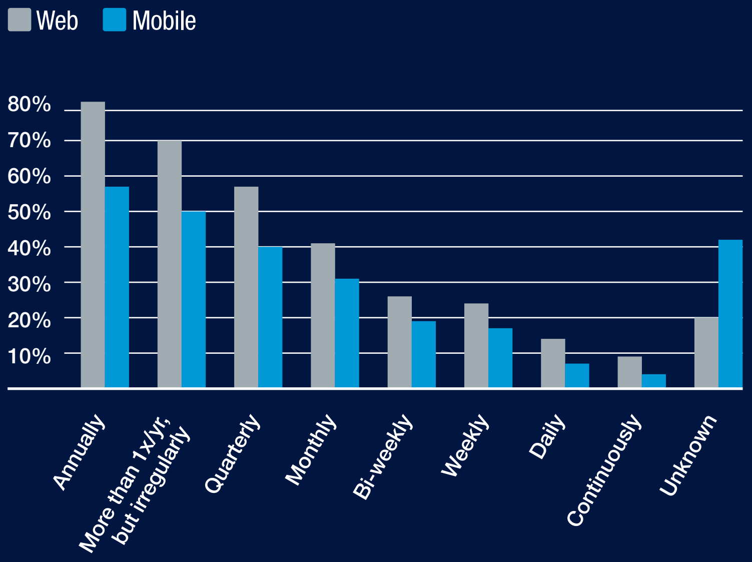
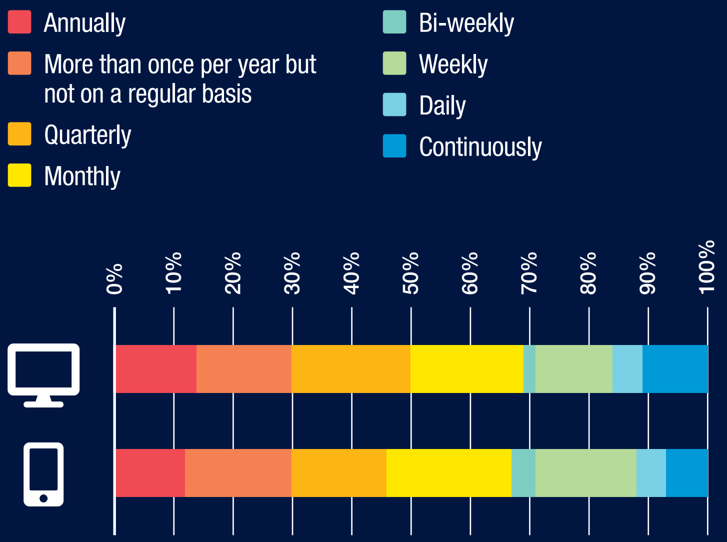


FIGURE 11. Comparing web & mobile testing frequency, security and privacy (removing unknowns)



© 2019 NowSecure & DevOps

WHEN SECURITY TESTING FITS INTO THE WORKFLOW

Security testing practices still vary widely within organizations, as we'll see in the next few data sets. The data indicates that organizations that do test their apps frequently do so at numerous checkpoints throughout the SDLC (*see fig. 12*).

While the overall distribution for web and mobile are roughly similar, it is of great concern that most testing across these stages is below 30%. Security best practices would recommend a higher rate at many different stages to minimize risks and ensure low security bug escape defect rates. For the most part, testing is light at each stage.

The overview shows that mobile apps are less likely to be tested at every stage of development. At each stage, mobile is some five to 10 percentage points behind web. In spite of this discrepancy, it's interesting to note that the distribution pattern of where testing occurs is very similar when comparing mobile to web. In both cases the most likely place to see a security test is when developers commit code, during unit testing, or at central build. Troublingly, approximately one in five organizations only test pre-release or at staging for either mobile or web, which indicates that plenty of laggards still exist.

TYPE OF TESTING

When engaged about the type of testing organizations do, our answers showed that for almost all categories, fewer than half are engaged in some form of automated testing for web apps. That ratio dips down considerably for mobile apps (*see fig. 13*).

Respondents were asked to choose as many as applied because many organizations use a layered approach to testing types, and the charting shows a fairly even distribution. This kind of question makes it difficult to show the global coverage for application testing, but with few categories reaching above a 50% penetration rate, it's clear that no single organization is adopting a truly layered approach to security testing coverage.

Interestingly, while internal penetration testing seems to be the most favored form of testing for web applications, SAST is the favorite among mobile apps. This might speak to the technical complexities of pen testing and lack of expertise and tools, and also reflect wide availability of SAST for both web and mobile. Ultimately SAST provides partial coverage of all mobile app risk vectors, but organizations often rely on periodic pen testing to achieve deeper coverage.

When comparing results of DAST and SAST, it's interesting to note that their prevalence was similar within both web and mobile categories. This is unexpected because DAST in a DevSecOps framework is hard to do in an automated fashion, is frequently outsourced, and many organizations traditionally see DAST as a speedbump.

FIGURE 12. When does your organization conduct security testing within agile or DevOps workflows for apps?

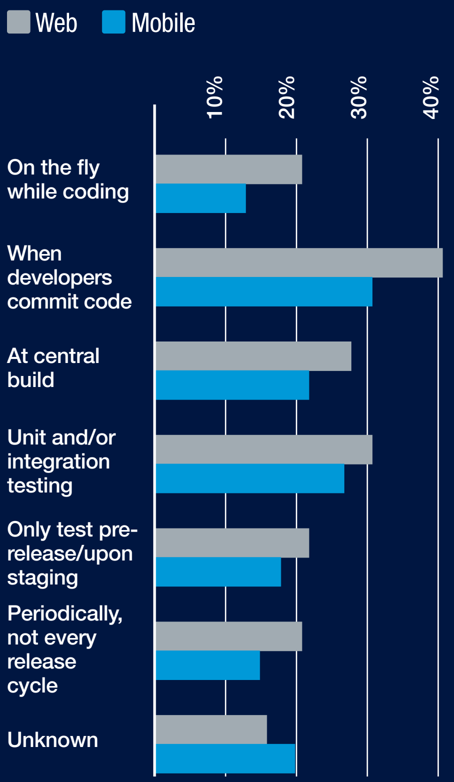
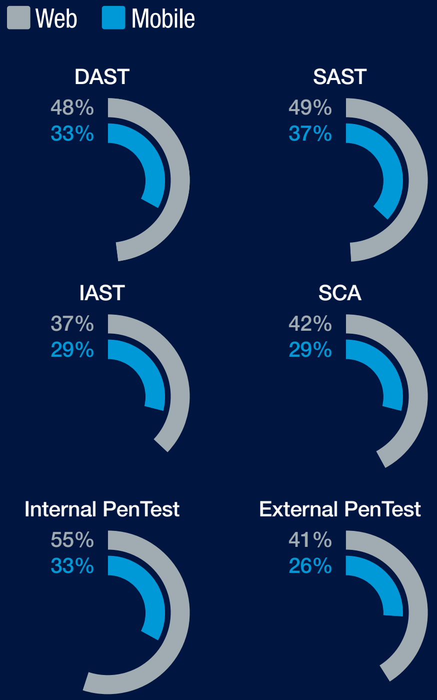


FIGURE 13. Types of application security testing performed



CONFIDENCE IN SECURITY OF APP

It's surprising how confident the bulk of respondents are about the security of their applications, given the lack of consistency in testing frequency and methods and the relatively small number of organizations operating in full DevSecOps delivery mode. Over two in three respondents say they're extremely confident to confident in their app security programs, which is not consistent with the narrative that only 14% to 24% have fully shifted left on security with DevSecOps practices across the board and the lack of frequency of security testing. This could indicate a false sense of security (*see fig. 14*).

INDUSTRY VIEW OF CONFIDENCE

Looking at the weighted average, you can see that healthcare and government tend to be more confident than the average, while finance, insurance and telecom are less confident (*see fig. 15*).

MOBILE APP SECURITY CONFIDENCE

Drilling down specifically into mobile confidence, this is where you see some wavering as well as a degree of uncertainty creeping in, with a statistically significant number of respondents answering 'don't know' (*see fig. 16*).

Still, more than half of organizations — 54% — are as confident or more confident in mobile app security as compared with web. This is somewhat surprising due to less frequent mobile testing across the board but this is a good indicator that confidence doesn't always necessarily correlate to actual security readiness.

FIGURE 14. How confident are you in the security of your applications?

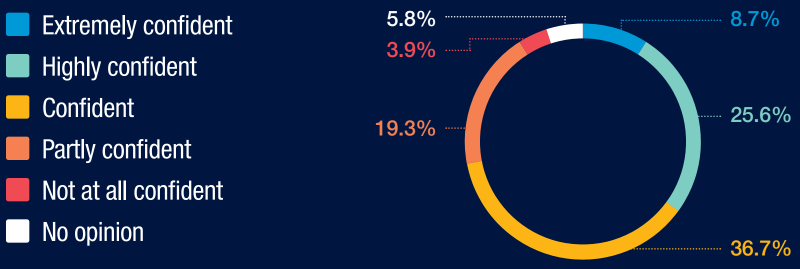


FIGURE 15. Application security confidence by major industry

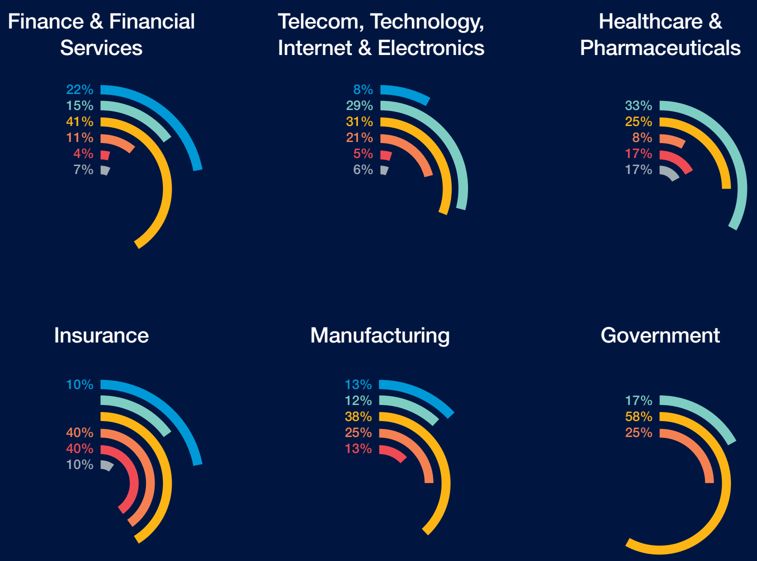
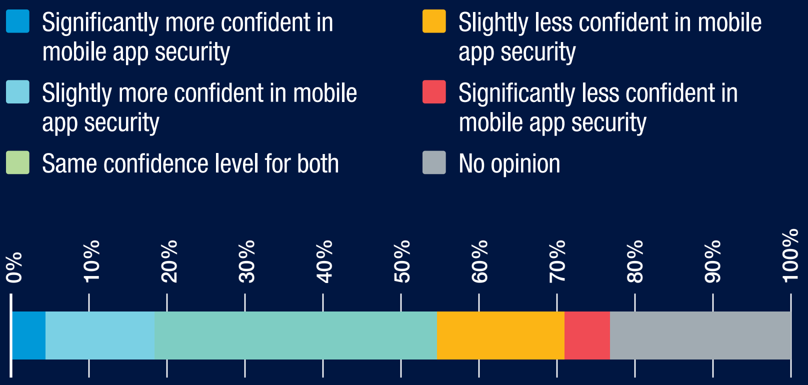


FIGURE 16. What is your level of confidence in the security of your mobile apps compared to the security of your web apps?



© 2019 NowSecure & DevOps

The Challenges of Mobile DevSecOps

SPEED OF REMEDIATION

When you put the pedal to the metal, the question is how well the entire DevSecOps team can respond to critical security issues. For the most part, organizations still struggle to remediate even the highest severity security vulnerabilities in their apps in a timely fashion—regardless of the platform. Indeed the curves below show very consistent pattern spreads across all remediation times for web and mobile (*see fig. 17*).

However, mobile app remediation in particular has some troubling sticking points, as *fig. 18* more clearly illustrates.

Most troubling is the fact that while 72% of organizations can get to web app critical flaws within a month, only a little over half can say the same for mobile. Additionally, the percentage of organizations that take a year or longer or don't know their remediation times are 2x more prevalent for mobile apps over web.

FIGURE 17. How quickly is your organization able to remediate high-severity security vulnerabilities in apps?

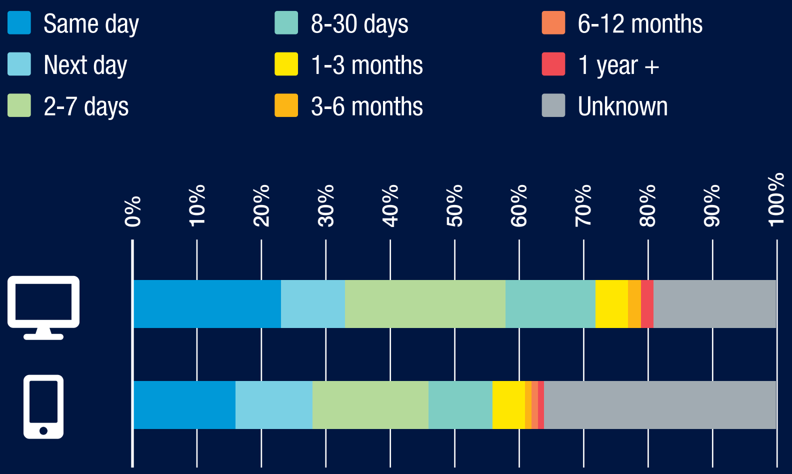
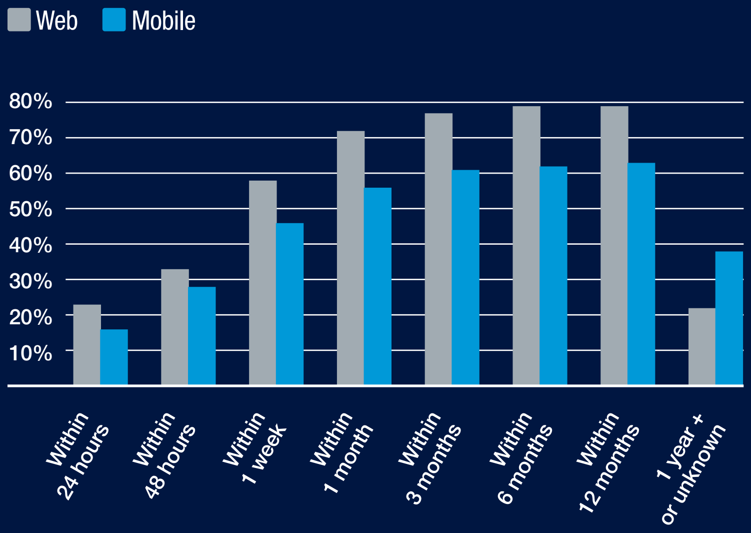


FIGURE 18. How quickly is your organization able to remediate high-severity security vulnerabilities in apps (cumulative)?



© 2019 NowSecure & DevOps

SECURE DELIVERY OF MOBILE SOFTWARE VERSUS WEB

One thing we can say conclusively is that very few organizations report that securing mobile app development is easier than web.

On the whole, 63% of respondents said it is at least the same level of difficulty or more difficult to securely deliver mobile apps, with at least one in three reporting more difficulties (*see fig. 19*).

If you look at the pie chart breakdown by the extreme ends of company size — companies with more than 10,000 employees (*fig. 20*) and those with fewer than 100 employees (*fig. 21*) — it becomes clear that securely delivering mobile software is easier for smaller organizations and harder for larger ones. This may correlate to smaller teams acting and adapting faster to critical needs.

FIGURE 19. Perceived difficulty of implementing secure delivery of mobile apps vs. web apps (all companies)

🟩 Less difficult
🟥 More difficult
🟦 Same level of difficulty

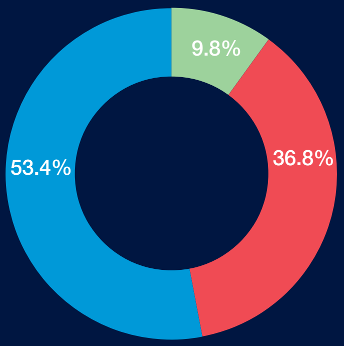


FIGURE 20. Perceived difficulty of implementing secure delivery of mobile apps vs. web apps (large 10,000+ employees)

🟩 Less difficult
🟥 More difficult
🟦 Same level of difficulty

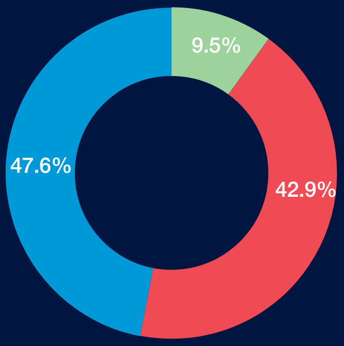
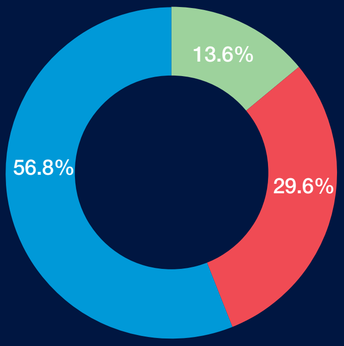


FIGURE 21. Perceived difficulty of implementing secure delivery of mobile apps vs. web apps (small under 100 employees)

🟩 Less difficult
🟥 More difficult
🟦 Same level of difficulty



GREATEST CHALLENGES IN TESTING MOBILE APPS

When examining the major challenges of testing mobile apps for security, without a doubt the lack of automated tooling rises to the top, followed closely by false positives. This mirrors overall DevSecOps challenges described in previous sections (*see fig. 22*).

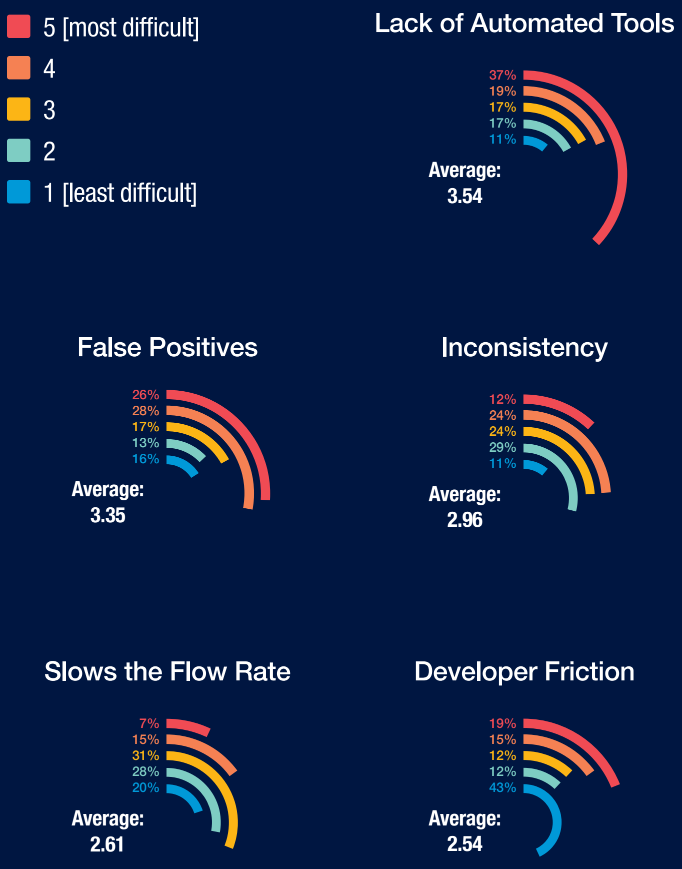
These results also reflect the realities of the market, where there are very few automated tools available for mobile app security testing compared to those that test web apps. In many cases app security tools purpose built for web apps generate a lot of frustrating false positive noise and leave testing coverage gaps in mobile app environments.

This is in large part because most web apps are written in HTML Java interpretive language which has a standard consistent syntax that is easy to run through emulators. Thus, the scope of complexity has a lower variability. As a result, organizations have a much higher ability to control the environment in order to instrument and effectively test it.

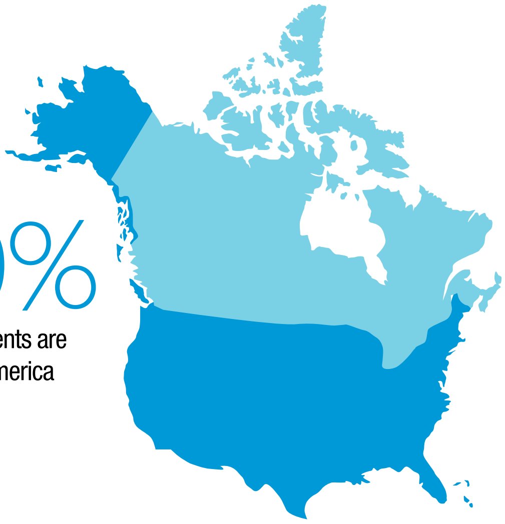
Mobile environments, on the other hand, are running apps on devices with a full mobile operating system (not just a browser) that are so locked down that it is virtually impossible to get instrumentation into the mobile devices in order to test the mobile apps effectively.

In the real world, experienced security testers must jailbreak iPhones or root Android devices, manage complex configurations, plug in open-source instrumentation or use third-party tools to achieve the same kind of test coverage as web. This is much harder than establishing a predictable web test environment. As a result, fewer vendors make technology to test mobile on mobile devices. That's why in many instances, mobile apps are only partially tested for security. It's a bear to do dynamic runtime testing, which is likely why SAST dominates in mobile apps as results in previous sections show.

FIGURE 22. How would you rank the biggest challenge your organization encounters when testing mobile apps for security within CI/CD workflows (scale of 1 - 5)?



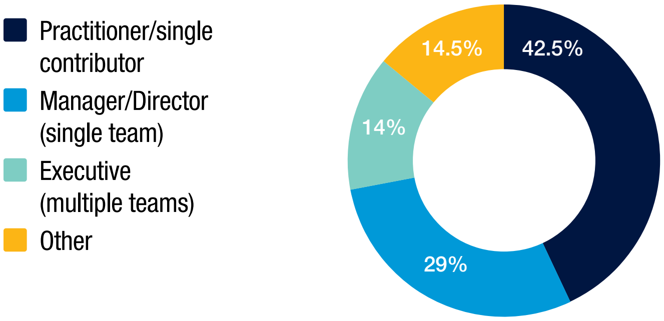
Study Demographics



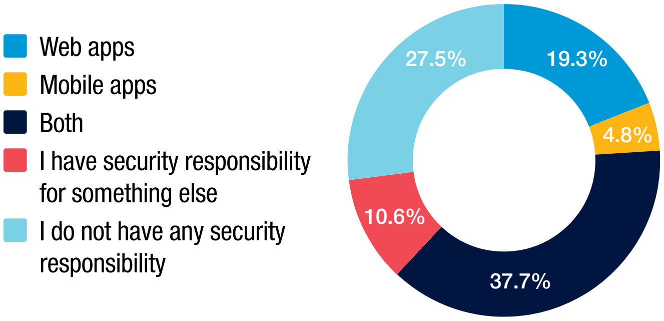
100%

of survey respondents are located in North America

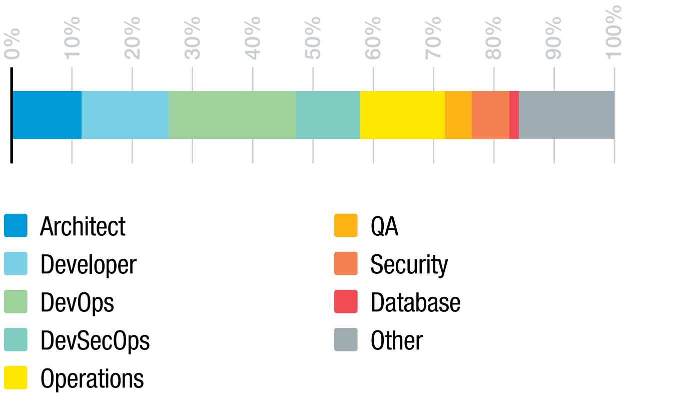
What is your role at your organization?



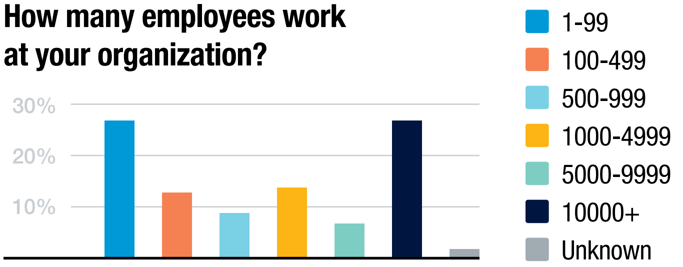
Do you have responsibility for securing either of the following?



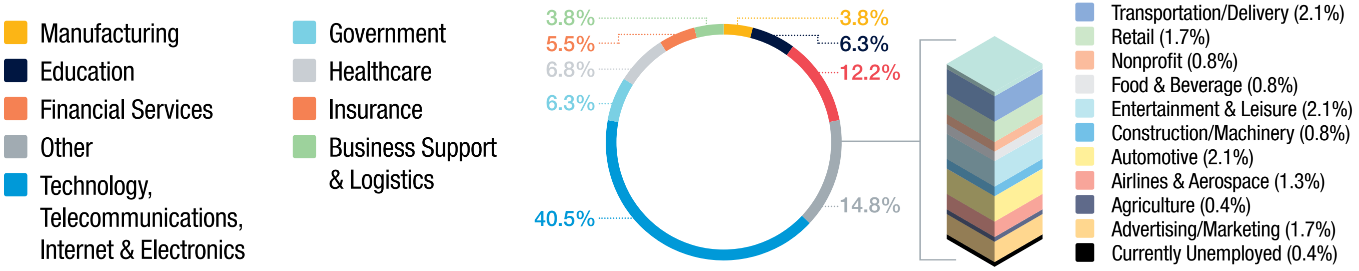
What area is your primary responsibility?



How many employees work at your organization?



Which of the following best describes the principal industry of your organization?



© 2019 NowSecure & DevOps

Conclusion

While confidence in DevSecOps practices is high, digging into respondents' answers shows that there's clearly a lot of work to go as organizations fold in more frequent security testing into DevOps practices across both web and mobile app development.

To learn more about how NowSecure can help your organization automate mobile app security testing, sign up for a free trial of our solution at *https://www.nowsecure.com/go/trial/*.

ABOUT US

Only **NowSecure** delivers fully automated mobile app security testing software with speed, accuracy, and efficiency for Agile and DevOps initiatives. Through static, dynamic, behavioral and interactive mobile app security testing on real Android and iOS devices, NowSecure identifies the broadest array of security threats, compliance gaps, and privacy risks. NowSecure customers can choose automated software on-premises or in the cloud, expert professional penetration testing and managed services, or a combination of all as needed.

DevOps.com, the flagship site of MediaOps, features the largest and most diverse original content related to DevOps. DevOps.com is one of the top destinations for DevOps influencers, buyers, practitioners and leaders.

© 2019 NowSecure & DevOps

NowSecure™

nowsecure.com
@NowSecure
@NowSecureMobile

DevOps.com

devops.com
@devopsdotcom
facebook.com/devopscom