Hewlett Packard Enterprise

Report

Application Security and DevOps

What is the true state of security in DevOps?

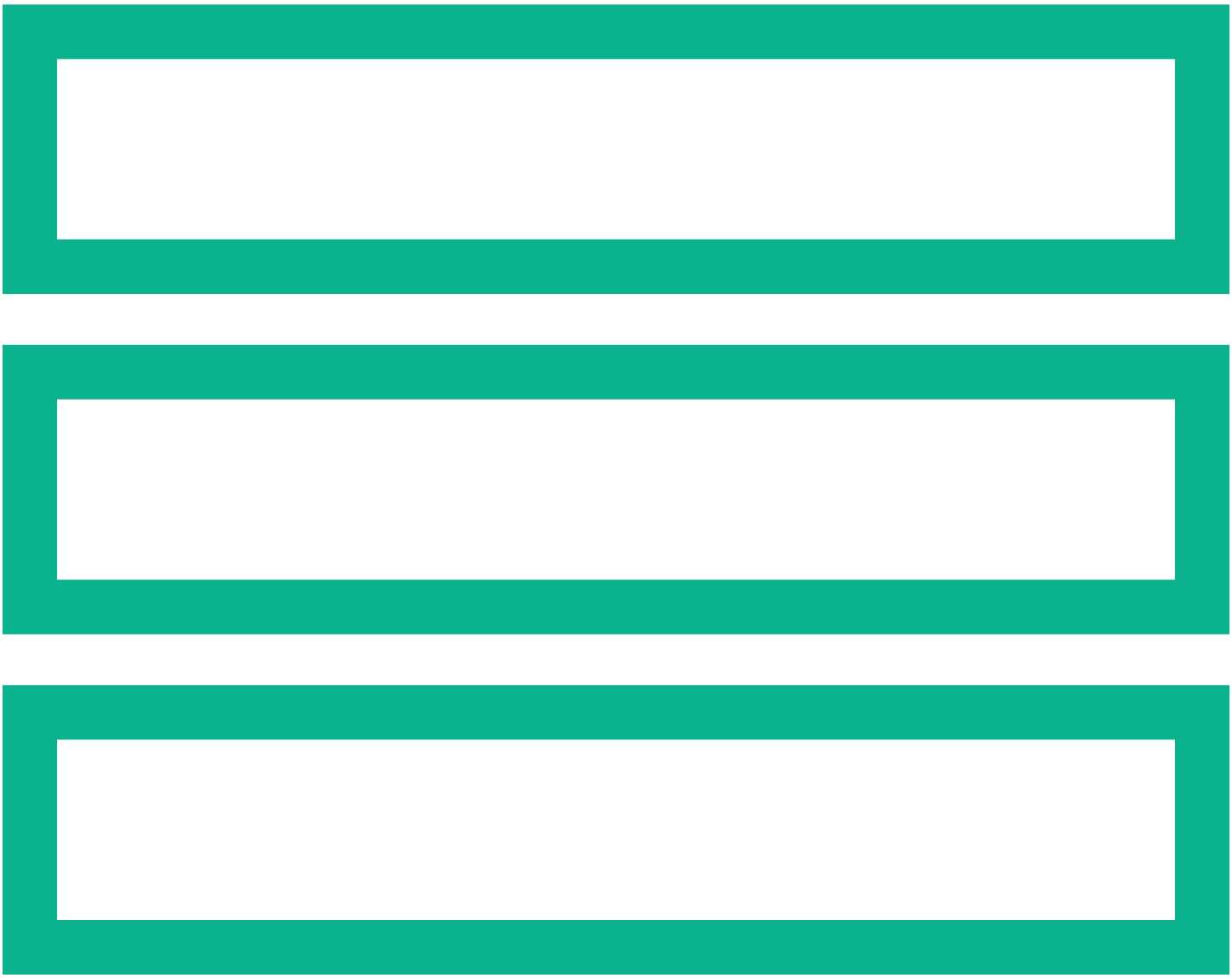


Table of contents

3 Introduction
3 DevOps makes security better
3 Missing an opportunity
3 Security remains siloed

4 DevOps undefined
4 DevOps—an amorphous definition
5 Frequent deployment
6 Automation
6 Teams
7 DevOps adoption

8 DevOps and application security
8 The promise
9 The reality

11 Identifying the barriers

11 Development organization
11 Lack of security awareness in development
12 Time-to-market pressure

12 IT operations organization

13 Security organization
13 Security leaders are not developers
13 Lack of application security talent
13 Integrating into DevOps is difficult

14 Conclusion

15 Research methodology

Introduction

Rapid application delivery is dramatically transforming how software is created and delivered, pushing the limits on the speed and innovation required of development teams. With the rise of DevOps, there is a new opportunity to improve the software development lifecycle (SDLC) in tandem with the moves being made toward agility and continuous delivery, but making the transition securely is not automatic. The **HPE Security Fortify** team surveyed a wide range of industry leaders, security practitioners, and developers to:

- Determine where organizations are in their transition to DevOps
- Gauge how security efforts are included in those efforts
- Identify the obstacles and opportunities in improving security practices in a DevOps environment

The results provide insight into current DevOps security practices at both large and mid-sized enterprises, and highlight multiple gaps that still exist between the opportunity to have security as a natural part of DevOps and the reality of current implementations. Key findings include:

1. Everybody believes that security should be an integral part of DevOps and that their DevOps transformations will actually make them more secure.
2. However, very few DevOps programs actually have included security as part of the process since it's a much lower priority than speed and innovation.
3. This problem persists and could worsen in DevOps environments because silos still exist between development and security.

DevOps makes security better

In theory, most people agree that application security and DevOps go hand-in-hand. DevOps is an opportunity to make security an integral part of development and truly builds secure coding practices into the early stages of the SDLC.

Missing an opportunity

While automation and team integration could lead to greater adoption of application security in the future, the current state is that most organizations are not implementing security within their DevOps programs. In mature security organizations, where application security is already an integral part of development, it continues to be prioritized as a critical DevOps component. If **a secure SDLC** was not a disciplined practice before, it is often left behind in the rush to DevOps.

Security remains siloed

The promise of DevOps bringing down organizational barriers hasn't yet materialized for security. Building security into the development tool chain and strategically implementing security automation are requirements for breaking down these walls. In addition, shared organizational responsibility between application teams and security with executive sponsorship is necessary—without everyone having a stake in **the secure SDLC** initiatives will fail to deliver.

DevOps undefined

DevOps—an amorphous definition
Similar to "the cloud" a few years ago, DevOps is somewhat of a buzzword that is seen as the next big thing, but most enterprises do not know exactly what it means to their organization. It is very common to selectively pick and choose processes that align to the distinct needs of the organization. In fact, 30 percent of respondents who said that their organization was not practicing DevOps were actually deploying some capabilities that are considered part of the DevOps process.

Within DevOps organizations, convergence is happening and commonalities are starting to emerge. Within the surveyed audience, the most commonly adopted practices of DevOps include frequent deployment, automated testing, and integrated teams.

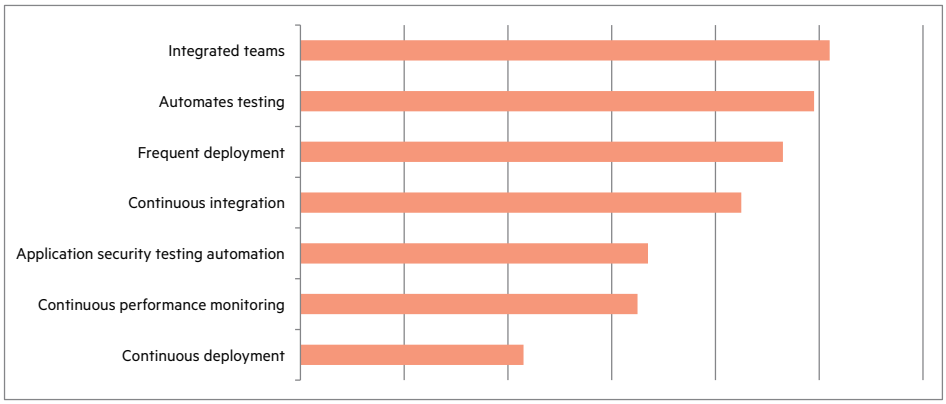


Figure 1. Defining characteristics of DevOps

Frequent deployment

Driven by increasing requirements to get features out to market before the competition, frequent deployment is the first characteristic associated with DevOps. Whether frequent deployment is a characteristic of DevOps or a key driver for DevOps, there's no doubt that the speed of deployment is increasing. Forrester reports that they are seeing organizations go from four application releases per year in 2010 to a whopping 120 releases per year by 2020.[1] This is a 30x increase. Based on **HPE Security Fortify** research, key drivers for more frequent delivery include:

- Mobile applications
- Web- or cloud-based application usage
- Market needs from customers
- Competition (e.g. if insurance companies do not deliver simpler ways for their customers to get quotes, view rates, pay bills, etc., online insurance companies will take their business.)

Our data shows that most DevOps organizations are at least on monthly release cycles now.

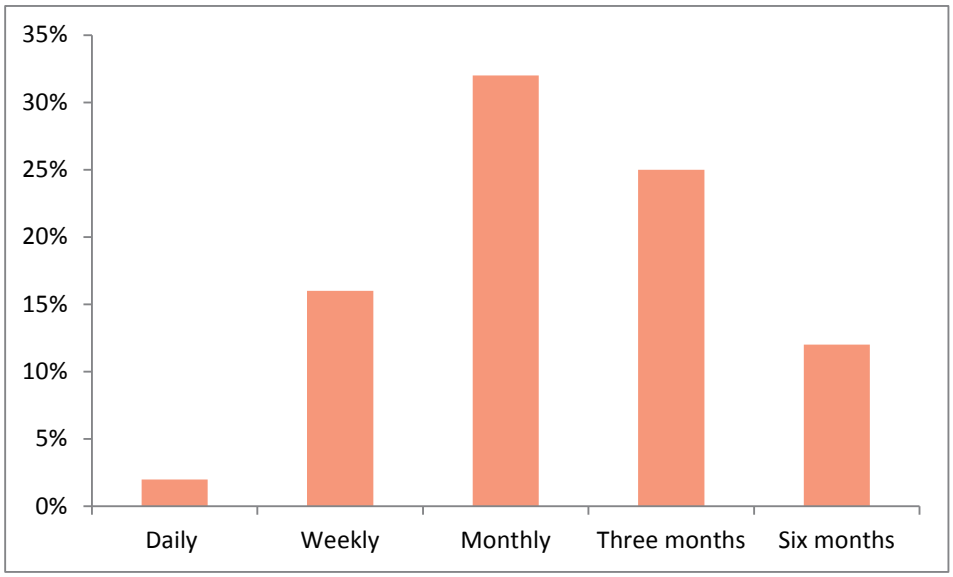


Figure 2. Release cycle frequency

[1] Better outcomes, faster results. Continuous delivery and the race for better business performance," Forrester Thought Leader Paper commissioned by HP (now Hewlett Packard Enterprise) **ssl.www8.hp.com/us/en/ssl/leadgen/document_download.html?objid=4AA5-1119ENW&siebelid=7404&sectionid=pdf&returnurl=%2Fus%2Fen%2Fsecure%2Fpdf%2F4aa5-1119enw.pdf&subbu=TSG.Software&simpletitle=ALM%20pillar%20page&parentUrl=https%3A%2F%2Fwww.google.com%2F&autoSubmit=true**

Automation

Automation is a key enabler to faster releases and was the #2 key characteristic in a DevOps rollout. Tool proliferation (often leading to tool exhaustion) was a common thread when discussing automation. When asked what tools are being used in support of a DevOps initiative, more than 50 were mentioned, and most of them play a role in enabling speed and collaboration.

Figure 3. DevOps toolbox

Teams

Some define DevOps in a very literal and simplistic sense: the joining of developers and operations into cohesive teams to enable faster delivery. It's often seen as a culture, movement, or practice with the main purpose of improving communications between software developers and IT operations (IT Ops). DevOps teams often form as the result of an initiative in a development organization to connect directly into the release pipeline and operational environment. The DevOps team now pulls in both developers and operations personnel alike to foster low latency, closed-loop communication.

> "I guess in a nutshell the way we use DevOps would be the intersection of development, IT operations, and QA. One of the things we do is we have rapid release cycles. There is just this constant movement of things through the whole SDLC from conception to release."
>
> – App manager

DevOps adoption

Recent Gartner research indicates that 38 percent of enterprises are now using DevOps and 50 percent will be actively using it by the end of 2016.[2] Our research got a bit more granular in assessing specific development teams, rather than organizations as a whole. We found that most organizations—90 percent of those surveyed—have at least 5 percent of their development teams practicing DevOps, typically with small pilot programs in progress. Despite all the hype and attention, advanced and mature DevOps programs are proving to be rare.

The common DevOps maturity progression is similar to what software development teams experienced in the transition from Waterfall to Agile. What is common with the Agile journey is the focus on evolving processes and teams to achieve a new way of working that enables better outcomes in terms of speed, innovation, and quality. However, the DevOps transformation is a bit more complex than just adopting Agile practices. It's both a process evolution and widespread adoption of automation and cultural change—all at once.

Agile brought a new way of organizing teams and release schedules but did not have as dramatic an impact on the tools used in the development process. DevOps is as much about heavily automating repetitive, error-prone tasks to improve overall development effectiveness, as it is about a new process or organizational principle. In this way, DevOps practices and Agile methods are inextricably linked toward a common goal—speed.

As shown in Figure 4, the evolution from Waterfall to DevOps is not exactly a continuum. Most of the participants were still struggling to become more Agile when the DevOps trend came on the scene. Ideally, most will work to implement both, rather than one over the other, as they go hand-in-hand and are meant to work together.

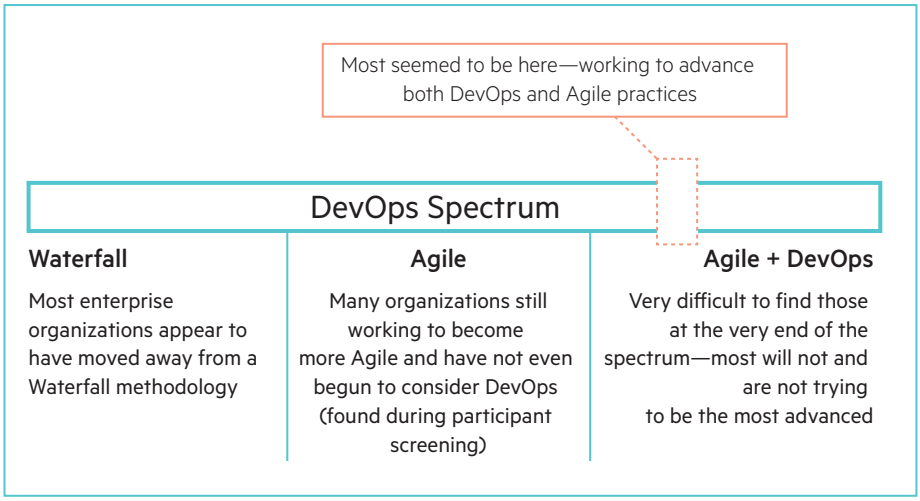


Figure 4. The DevOps spectrum

Those practicing more advanced DevOps appeared to be organizations that were "born" into it, for example, Facebook, Google™, and AirBnB. They have not faced the barriers most enterprise organizations have had to overcome, such as legacy applications and compliance mandates that do not allow for automated deployment.

[2] Gartner Enterprise DevOps Survey Study, May 2016

DevOps and application security

As overall security awareness has increased across all of IT, the question of how security fits into DevOps is a fast growing conversation. Terms such as DevSecOps and Rugged DevOps have been proposed to try to capture what is unique about integrating security into the DevOps process, but they treat security as an (optional) addition to DevOps rather than an integral part of it. Numerous articles and studies have been published predicting that DevOps will enable security adoption within the SDLC, but with little substance as to how teams can actually achieve that successfully. In our research, we looked at both the promise and the reality of the influence of DevOps on **the secure SDLC**.

The promise

The promise of DevOps increasing application security adoption is centered on the belief that security must be part of the DevOps process and not a separate function. In other words, it has to be built in. The study found that 99 percent of all respondents agree that adopting a DevOps culture has the opportunity to improve application security. Further, the expectation is that applications will be released with a level of security that meets the goals of the organization to ensure the protection of not only the software and customers but also the organization itself.

> "Adopting a DevOps process can help make applications more secure, since the development and production environment are built the same way and to the same security standards and testing. However, it requires a commitment across the organization to prioritize security, and incorporate more automated testing solutions that make it easier to gather real-time feedback and remediate vulnerabilities throughout the development process."
>
> John Meakin—Burberry, Group Information Security Officer

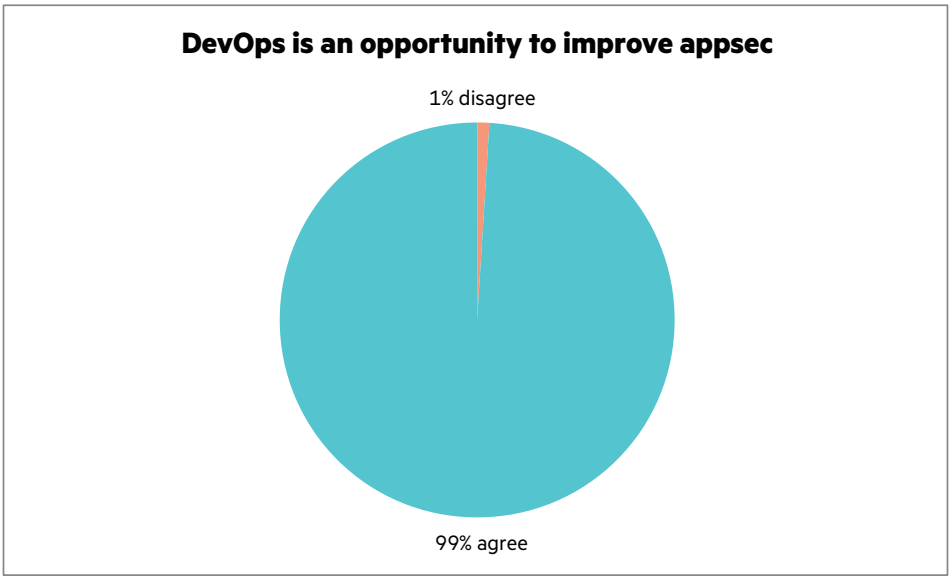


Figure 5. DevOps is an opportunity to improve application security

The reality

The reality of adoption has proven to be different from the promise. When asked how organizations adopting DevOps are currently protecting applications, the overwhelming majority cited security practices or controls downstream of the SDLC, with only 20 percent stating that **secure SDLC** testing is done throughout development. Most organizations are relying on the technologies downstream, such as pre-production penetration testing and network security. A shocking 17 percent stated that they are not using any technologies to protect their applications.

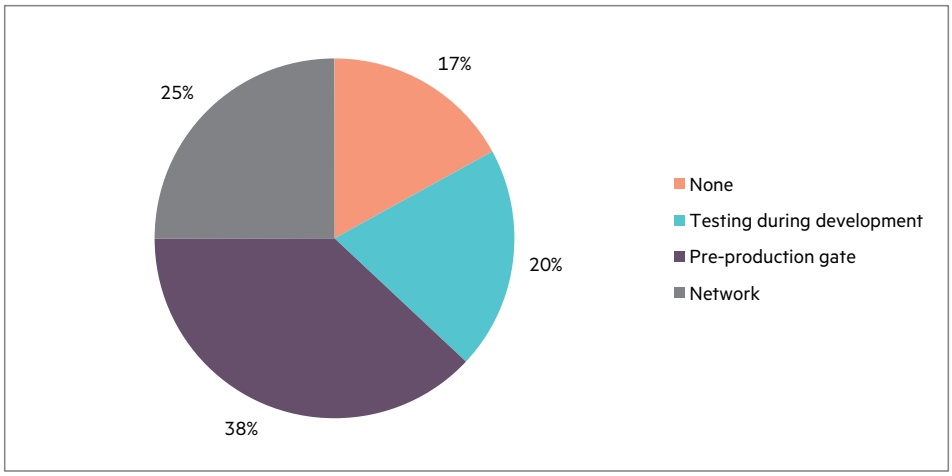


Figure 6. How organizations adopting DevOps are securing applications

When asked how DevOps has affected creating a **secure SDLC**, the overwhelming sentiment emerged that DevOps in itself has had little to no impact on application security adoption or effectiveness in their organizations. When asked about the effect of DevOps on key application security use cases such as frequency and thoroughness of testing, the result on a scale of 1 to 5 was 3.38, a neutral (no effect) finding.

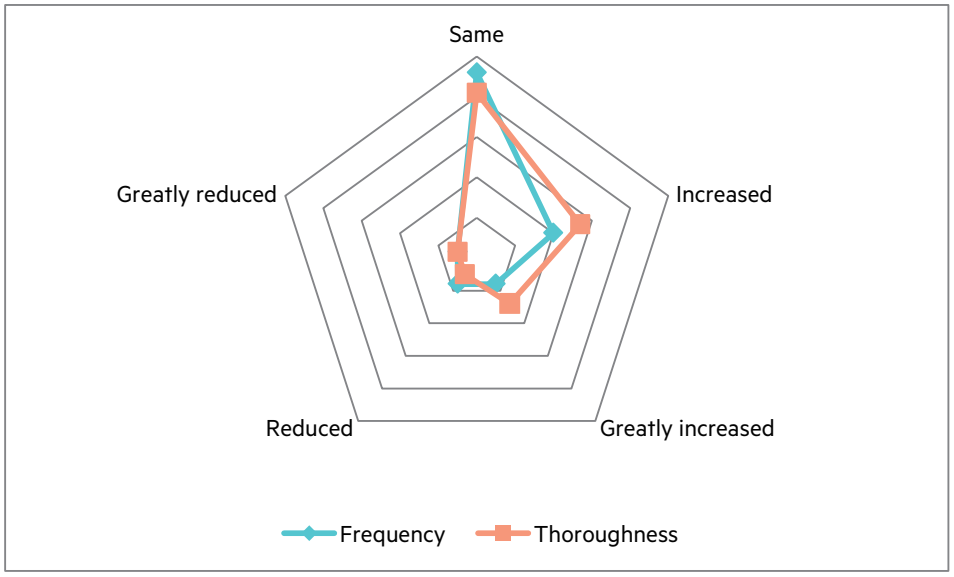


Figure 7. How DevOps has affected application security

During qualitative interviews, it became clear that mature security organizations, where application security is historically an integrated part of development, continue to prioritize security and include it as a critical component of DevOps. If organizations were doing it well before, they are most likely doing it well now as part of DevSecOps. But if they weren't very good at securing the SDLC before, the move to DevOps alone is not going to solve the problem.

All of these findings lead to the question of why is this the case given the impact that security breaches continue to have. What is preventing organizations from ensuring that application security is part of DevOps?

Identifying the barriers

To identify the barriers to more widespread **secure SDLC** adoption within DevOps organizations, we focused mainly on qualitative research. We wanted to hear from practitioners in real-world organizations. We found that the key factor hindering security adoption within DevOps is organizational barriers. While one of the main promises of DevOps is the collaboration between development, operations, and quality assurance (QA), security teams are often nowhere to be found in the DevOps conversation or team.

Overall, developers and IT Ops care about security but feel it is already under control or that it's someone else's issue (such as security, InfoSec, and compliance departments). Security feels disconnected from both development and operations, and in some cases, respondents admitted to not even knowing their security teams. Reporting lines within organizations do not help break down organizational silos and most development, operations, and security groups have completely separate reporting structures. These dynamics can lead to a divide between security organizations and development with differing metrics and misaligned priorities.

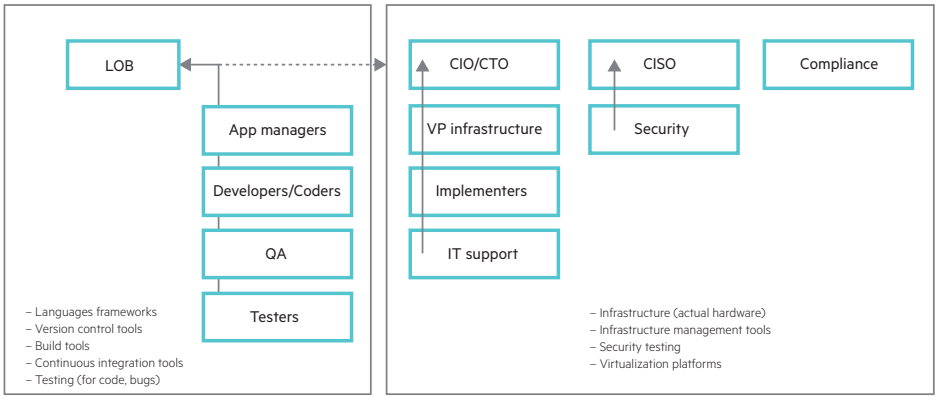


Figure 8. Development, operations, and security groups have separate reporting structures

Development organization

Most developers today do care about security and many are starting to learn and incorporate security practices in their work, but they are still primarily measured and motivated to focus on timely delivery, features, and quality. They usually agree that there is an overall lack of security training and pressure to release quickly compromises their ability to place more emphasis on software vulnerabilities until it is too late in the process.

Lack of security awareness in development

There are several factors leading to the lack of awareness regarding the importance of secure coding practices within development organizations.

1. Security is not part of computer science programs according to a 2016 CloudPassage report—out of the top 10 U.S. Bachelor's Computer Science programs, none require a security class to graduate.
2. Secure coding practices are not part of job requirements—in looking at more than 100 job postings for software developers at Fortune 1000 companies, none specified security, secure coding experience, or knowledge as part of skills required.

Time-to-market pressure

Developers are under immense pressure to get features out to market as fast as possible. A 2015 Forrester study[3] concluded, "customer expectations in competitive markets rise in response to attractive alternatives. Organizations in these markets are scrambling to keep pace. 73% of IT decision-makers surveyed said that business leaders demand more-frequent delivery." This business requirement is forcing development to prioritize features, functionality, and performance and eliminate anything that is considered to slow down the development process.

"I would say that if we proposed that hey, we should take some time and do some security testing, that management would rather we spend that time working on some other feature for some other product that some business unit is looking for."

– App manager

IT operations organization

IT Ops groups more often value and focus on security, since they are organizationally closer to security or more often share part of the responsibility. However, given the significant over-reliance on perimeter-based security, IT Ops focus on protecting the infrastructure instead of the application. In the eyes of most IT Ops, application security is mostly thought of as manual penetration testing and a responsibility of the security team. They have little understanding of the role that security should play in development.

"Security testing is hand-done by our InfoSec department, not us."

– IT Ops manager

[3] Application Delivery Speed Drives Success. How Mastering DevOps Enables Speed With Quality And Low Cost. A commissioned study conducted by Forrester Consulting on behalf of HP (now Hewlett Packard Enterprise), April 2015

There is another trend within IT Operations that has a significant impact on the role of IT Ops—the movement of infrastructure to the cloud. There are many questions on whether IT Ops will even be a critical function in five years. According to James Quin, senior director at business-to-business (B2B) marketing firm CDM Media:

"The IT department isn't going away, and the role of the CIO isn't going to be marginalized. But as more workloads shift to the cloud, the construction of the IT department, by necessity, must change away from traditional roles to those more focused on vendor, business, security, and service management. This doesn't mean that development and administration jobs go away, just that there are fewer of them."

This trend suggests that an even greater importance needs to be put on embedding security into the actual process before all those processes disappear into the cloud.

Security organization

In addition to the challenges with having stakeholders in development and operations, who have not fully bought into **the secure SDLC** initiatives, there are also internal factors that impede application security teams, such as the lack of understanding of development by security leadership, increasing lack of application security talent, and lack of integration of security requirements and practices into the development process.

Security leaders are not developers
Security roles have become very specialized with most security professionals having a background in IT. In our findings, only 15 percent of chief security officers (CISOs) have a background in development. This can lead to a misunderstanding of challenges faced by development teams.

Lack of application security talent
For organizations that have put a focus on **the secure SDLC**, there is a significant shortage of application security talent. There was an average of 900 developers in the organizations surveyed, in comparison to the average of 11 application security professionals in those organizations. This ratio, in combination with the increasing velocity of development, is leaving application security professionals unable to keep up.

Integrating into DevOps is difficult
Ninety percent of security professionals surveyed state that since their organization has started deploying DevOps methodologies, integrating application security into the development process has become more difficult. Interestingly enough though, 100 percent cite that integration is a key requirement to the success of an application security program.

> "With the primary goal of DevOps to reduce friction and increase velocity, I am concerned that applications will be less secure until automation and supporting processes are mature."
>
> – CISO from Fortune 100 Energy company

Key reasons for this difficulty include:

- Complexity of the DevOps development environment
- Rapid change in emerging technologies for automation, continuous integration, and continuous deployment
- Lack of involvement in security in the initial planning processes for application and development processes

Proper planning came out as a major theme in our interviews. Application security professionals overwhelmingly agree that security requirements have to be an integral part of planning for both the application itself as well as the engineering of the development process. This is truly the point of conversion as non-security participants did see value in having automated security tools for vulnerability testing tightly integrated into the SDLC.

> "With no proper planning for security, DevOps is a nightmare for every security engineer."
>
> – Application security manager from a Fortune 500 financial institution

Lowering security risk and meeting compliance standards allows for increased automation, but many felt it would be the security team that would implement it.

Broken down by each group, it becomes clear that in addition to high levels of automation and integration into the DevOps SDLC, every DevOps team must have a security function embedded in it. Additionally, executive sponsorship is a must have—without everyone having a stake in security, initiatives will fail.

Conclusion

While securing the SDLC remains a key challenge in most organizations, there are those that have overcome the organizational and process barriers to embed application security into their development processes. Best practices for secure application deployment, and better integration between application security and DevOps teams include:

- **Security should be a shared responsibility across the organization to eliminate barriers.** Security must be embedded throughout every stage of the development process, with executive support and metrics to hold teams accountable for secure development.
- **Bridge awareness, emphasis, and training gaps by making it seamless and more intuitive for developers to practice secure development.** Organizations should integrate security tools into the development ecosystem to allow developers to find and fix vulnerabilities in real-time as they write code. This makes it easy and efficient to develop securely, and educates the developer on secure coding in the process.
- **Leverage automation and analytics as application security force multipliers.** Organizations should leverage enterprise-grade application security automation with analytics built in to automate the application security testing audit process and allow their application security professionals to focus only on the highest priority risks. This reduces the number of security issues that require manual review, saving both time and resources, while lowering overall risk exposure.

Microsoft® IT and ServiceMaster are excellent examples of large- and medium-sized organizations that have invested and focused on application security. More information on ServiceMaster and Microsoft can be found on the HPE Fortify **secure SDLC** page.

Research methodology

The Application Security and DevOps Report 2016 leverages data and analysis from HPE Security teams, industry leaders, enterprises, and developers to deliver key insights on the multiple gaps and barriers between the promise and reality of secure DevOps.

In gathering this data, we performed both quantitative and qualitative studies.

For the quantitative research, 10 questions were asked anonymously, participation in DevOps was on a selection criteria for the survey, thus creating a sample that had DevOps within their organization and did not. Demographics of the survey population include:

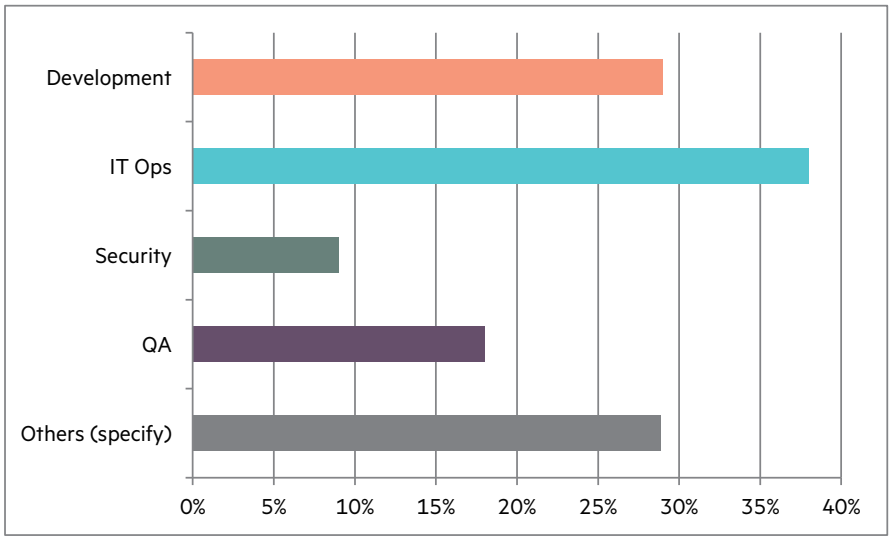
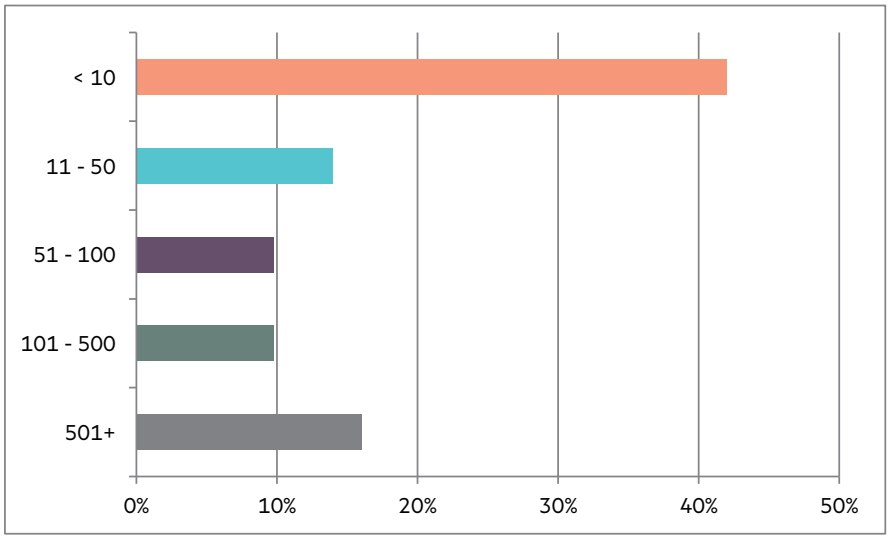
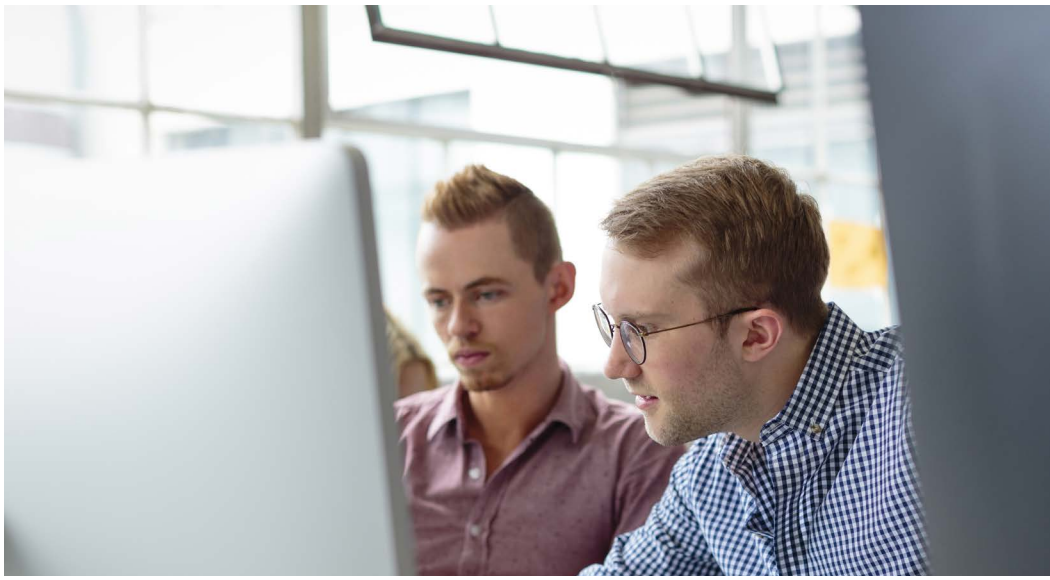


Figure 9. Job Role





Qualitative surveys were conducted in a blind study by a third party. All interviews were over the phone and 1-hour in length. Participation in the survey included several selection criteria such as organizational involvement in DevOps and role in the organization. Developers, QA, and IT Operations roles that were practicing some level of DevOps were selected for the interview.

Additionally, we surveyed a population of security executives in 1:1 interviews and email questionnaires.

Learn more about application security and securing the SDLC at:

hpe.com/software/fortify

Sign up for updates

4AA6-8302ENW, October 2016

Hewlett Packard Enterprise