FORRESTER®

FOR SECURITY & RISK PROFESSIONALS

Secure Applications At The Speed Of DevOps

How To Leverage DevOps Trends To Strengthen Applications

by Amy DeMartine

December 16, 2016

Why Read This Report

DevOps methodologies are increasing the pace of application releases, straining security teams that protect those applications at a time when security expertise is already scarce. Security pros must join developers and operations pros in engaging in DevOps practices, or they will lose an opportunity to systematically improve application security. This will require security pros to change their approach to people, process, technology, and oversight.

Key Takeaways

DevOps Can Increase Security But Only When Security Pros Engage
Underlying automation of the development life cycle means that security pros can insert automated tests and quality gates progressively earlier. Smaller releases also mean that security pros can change security testing on the fly, responding to changing application attack surfaces and continuously improving the security posture.

Get Developers and Operations Pros Involved On Their Terms
Get developers and operations pros involved by framing security gaps in terms of quality defects and production environment glitches and outages. Build a common understanding of business impact by using real-life examples of security breaches.

Create A Vision, And Lead Gradual Changes To Increase Application Security
DevOps is not a big bang improvement to application security. Just like developers and operations pros, security pros must learn how to improve application security incrementally and continuously.

Secure Applications At The Speed Of DevOps

How To Leverage DevOps Trends To Strengthen Applications

by Amy DeMartine
with Christopher McClean, Trevor Lyness, and Kara Hartig
December 16, 2016

Table Of Contents

Notes & Resources

Forrester interviewed five industry thought leaders for this report — Joshua Corman, Jez Humble, Gene Kim, Matt Konda, and James Wicket — as well as representatives from Blackboard, Dell, Disney, Intuit, Northwestern Mutual, nVisium, Orbitz, and ProQuest.

Related Research Documents

Boost Application Delivery Speed And Quality With Agile DevOps Practices

The State Of Application Security: 2016 And Beyond

Use DevOps Practices To Create A Lean And Responsive Application Delivery Organization

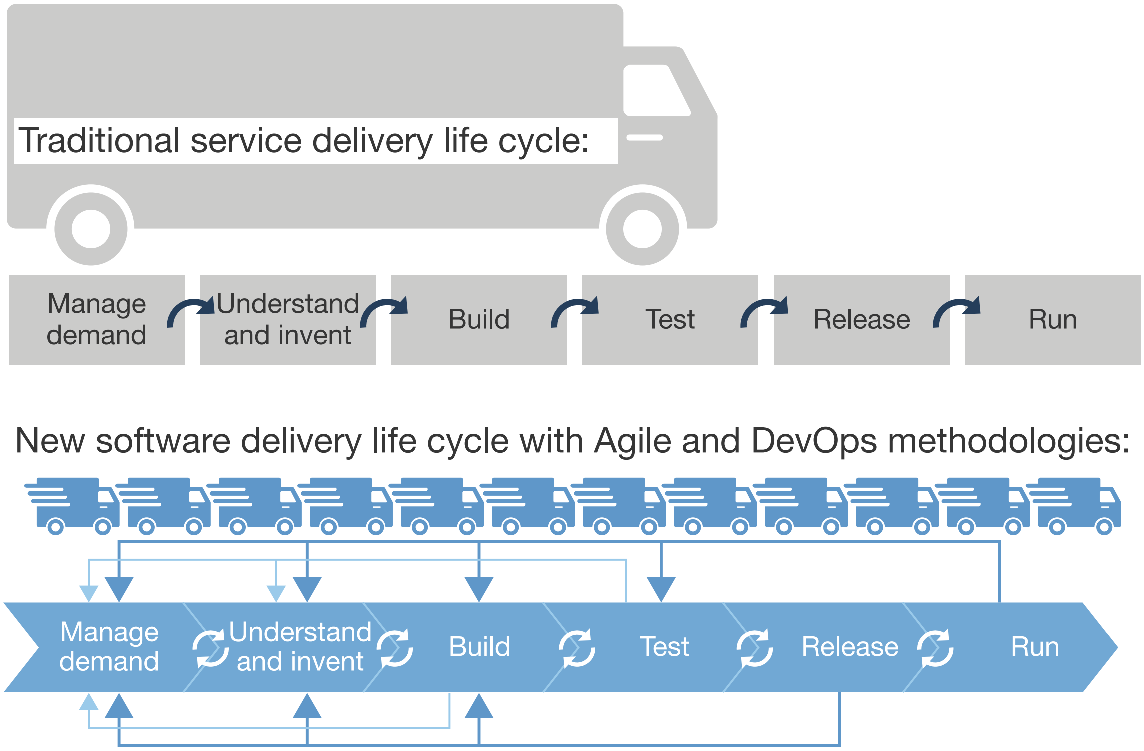
FORRESTER®

Forrester Research, Inc., 60 Acorn Park Drive, Cambridge, MA 02140 USA
+1 617-613-6000 | Fax: +1 617-613-5000 | forrester.com

Application Releases Are More Frequent And Vulnerable Than Ever

Applications are getting a lot of buzz and rightfully so. eCommerce websites, mobile apps, smart internet of things (IoT) interfaces, and other applications are how most firms engage with their finicky customers, so they need to be well designed and well engineered. To keep up with aggressive demand, Agile and DevOps methodologies allow developers and operations pros to break down traditionally large releases into a continuous development pipeline that Lean processes and software delivery life-cycle (SDLC) automation support (see Figure 1).[1] Cycles can now release daily or faster.[2] But as applications get more attention, they are increasingly becoming the focus of malicious attackers. The percentage of data breaches from a web application attack was 7% in 2015 and grew to 40% in 2016.[3] Worse, security resources are scarce, making manually securing these fast-paced application releases difficult if not impossible. In a recent study, 82% of respondents report a shortage of cybersecurity skills.[4] These conditions create the perfect storm for security pros who will have to support their organization's demand for rapid development with scarce resources.

FIGURE 1 Agile And DevOps Methodologies Break Down Previously Large Deliveries Into Continual Releases


Traditional service delivery life cycle:
Manage demand
Understand and invent
Build
Test
Release
Run
New software delivery life cycle with Agile and DevOps methodologies:
Manage demand
Understand and invent
Build
Test
Release
Run

Security Pros Must Join With Dev And Ops Pros To Harden Apps

Best practices dictate that security pros are on integrated DevOps product teams. But these teams usually struggle enough just to break down barriers between developers and operations pros to try and then break down the security barrier as well.[5] Security pros must therefore make the effort to reach out if they want to achieve effective application security changes. This will require new approaches to the relevant people, processes, technologies, and oversight.

People: Change Your Language To Match Concerns Of Developers And Operations Pros

Too often security pros use industry jargon to talk about malicious attackers and vulnerable applications. Instead, they should build trust with developers and operations pros by empathizing with them in their challenges and using their language. When talking about his efforts to get security more involved in application development, John Allspaw, senior vice president (SVP) of technical operations at Etsy, said, "Getting people to feel empathy for the pains or challenges that another group has to face is a good place to start."[6] To connect with colleagues and make effective changes, security pros need to:

- **Talk with developers about unplanned and unscheduled work.** Your firm incentivizes developers on delivery of new features, and fixing security defects can represent unplanned, unscheduled work that gets in the way of performance. This is especially true when these defects are found late in the SDLC. As David Mortman, SVP and cloud security architect at Bank of America, said, "Security issues are product quality issues, and no one wants to write buggy code."
- **Talk with operations pros about outages and performance glitches.** Operations pros pride themselves on understanding the interconnections between hardware and software, and firms rate them on their ability to improve the performance and stability of the production environment. Outages or performance issues due to security issues represent an interruption of service, and operations pros need your help to understand what environmental conditions create vulnerabilities and how to remediate them.
- **Impart knowledge about specific risks.** While developers and operations pros do not need to become security experts, they need to know how cybercriminals exploit application vulnerabilities, how they can work with security pros to decrease the attack surface, and how to respond quickly to attacks. For example, third-party and open source components are a particular source of risk, as attackers exploiting a single vulnerable component can affect all applications that use it.[7]
- **Host internal reviews of real-life breaches.** Too often the vault of security knowledge is locked tight in security pros' minds. However, examples of application security failures abound, including the breaches making use of SQL injection such as the vBulletin flaw, i-Dressup, and TalkTalk.[8] Use such examples to discuss real-life threats and vulnerabilities as well as prevention and detection methods that your colleagues can take to mitigate similar cases. Ken Johnson, chief technology officer at nVisium, told us that after discussing such scenarios with colleagues at his firm, "Everybody cares about security because they know that the threats are real. Developers and operations know what they didn't know before."

Process: Break Down Improvements And Improve Continuously

The current approach to application security puts security pros in the role of nag who only points out vulnerabilities but doesn't have the bandwidth to resolve them. To change this paradigm, security pros need to embed application security into existing continuous delivery processes. DevOps practices emphasize small incremental changes that development teams can release and test quickly. Likewise, security pros need to propose small improvements and experiment with new testing methods. There are several process changes that support these goals:

› **Discard detailed security road maps in favor of shared goals.** Adherence to militant processes or detailed improvement plans does not align with new application release speeds. Security pros must create a long-term vision or goal and then match rapid improvements to application security with those goals.

"We are going to solve a good number of issues, but we don't always know which ones or at which times. Security in DevOps needs to embrace change." (Matt Konda, founder of Jemurai and OWASP and global board member)

› **Prioritize incremental improvements.** You can't fix everything at once, and DevOps is all about quick wins. So instead of antiquated multiyear application security improvement plans, start with changes that reduce the most risk exposure for the least amount of effort, or automate the slowest security testing process in the SDLC. For example, an interactive application security testing (IAST) tool could reduce false positives and correctly identify unsafe coding practices, which could lead to long-term gains.

"We started with a multiyear security plan to align with our multiyear company strategy. However, after a year, the plans weren't meeting our needs as market dynamics changed. Even a year plan was too long. Now, we do quarterly plans to align with our Agile development cycles that reflect business needs at that time. As development cycles shrink, we will adjust our planning accordingly." (Daniel Ayala, director of global information security at ProQuest)

› **Use the plan, do, check, act (PDCA) model to identify the best solutions.** Once you've created a prioritized list of incremental improvements, the PDCA model will help you methodically execute improvement experiments.[9] For example, you might plan to insert static application security testing (SAST) during developer check-in. PDCA would recommend that you do a quick proof of concept for your top three tool choices and then check to see whether coding practices improve. If the tools are successful, the next step might be to pick the best one or to expand the test to other applications. If not, try other tools or new rule sets and test again.

› **Fix issues and gauge preparedness with security games.** Penetration tools and destructive testing can help you identify flaws in applications and supporting systems, such as the network, database, and OS. Some common packaged application penetration tools are Rapid7 Metasploit and Burp Suite, while common open source destructive testing tools include Chaos Monkey, Chaos Gorilla, and Chaos Kong.[10] To get the full value from these penetration and destructive testing tools, conduct red-team, blue-team (red-teaming) games, and involve developers, operations pros, and security pros.

Red teaming has roots in military exercises to test preparedness; the red team attacks something while the blue team tries to defend it.[11] For application security purposes, split operations pros, developers, and security pros into a red and a blue team, and eventually rotate all members of an integrated product team into the game to get equal exposure. You may use this practice intermittently during development sprints or schedule it regularly.

Technology: Insert Security Tests Early In The Software Delivery Life Cycle

Antiquated application security testing has historically been left until late in the SDLC or until after the product has been exposed in the production environment to customers — and malicious attackers. Organizations need security pros to constantly adapt to address new threats and to keep pace with them, and operations pros and developers need security testing results inside the continuous delivery pipeline.

Fortunately, the continuous delivery pipeline allows security pros to insert automated scanning capabilities at various stages of the delivery process. It gives them unprecedented opportunities to test early for security flaws, while enabling developers and operations pros to incorporate feedback as they go. As Andy Bustle, director of information risk management at Northwestern Mutual, explained to Forrester, “The continuous delivery pipeline is the main street where everybody does business, so that’s where we need to live to keep up with our development teams and eliminate any latency.”

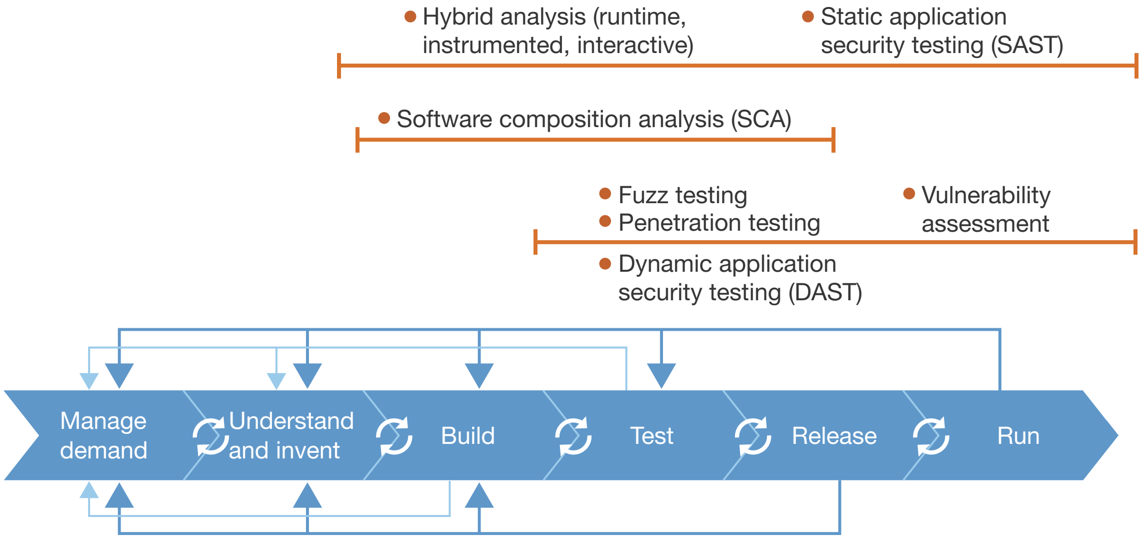
To take advantage of DevOps trends with technology advancements, security pros must:

› **Augment the existing continuous delivery pipeline.** Many tools are available to test applications throughout the SDLC (see Figure 2). For example, when developers check in code, static analysis automatic tests check for vulnerabilities while the code change is still fresh in their minds. Start new testing tools in the left-most spot in the SDLC and migrate existing tests to their left-most spot either incrementally in phases or directly to give the maximum amount of remediation time to developers and operations pros. For example, firms that start with scanning for vulnerable open source components in the production environment can slowly migrate to the testing environment and then developer desktop. Or they can simply skip directly to the developer desktop, while those firms just starting their scans can immediately start with the developer desktop.

› **Incrementally expand automated security tests.** As hackers get smarter and faster, organizations have to find and fix vulnerabilities faster too. Security pros need to respond by continuously changing the testing and tool mix. Advances across the application security ecosystem are fortunately moving fast. However, that means that tools that were previously considered feature-rich can easily lose their novelty. For example, with web application firewalls (WAFs), new attack vectors from bots, and DDoS layer 7 attacks, modern application delivery teams should already be looking at machine learning capabilities to identify and block this new malicious activity.

- **Pay close attention to open source software.** Approximately 80% to 90% of the code in modern applications is from open source components, and open source components that are at least two years old have three times the number of vulnerabilities.[12] Even when developers are diligent about using newer third-party libraries, these libraries often use other libraries of their own, resulting in latent vulnerabilities that expose themselves at a later date. Insert a software composition analysis (SCA) tool as early in the SDLC as possible, and continue to scan applications, including older applications with inconsistent or long release cycles, to ferret out newly discovered vulnerabilities.[13] Remember that the application stack includes more than just the application; all software must be scanned such as middleware, OS, database, and performance and security management tools.[14]

FIGURE 2 Apply Security Tools Across The Application Life Cycle


● Hybrid analysis (runtime, instrumented, interactive)
● Static application security testing (SAST)
● Software composition analysis (SCA)
● Fuzz testing
● Penetration testing
● Vulnerability assessment
● Dynamic application security testing (DAST)
Manage demand
Understand and invent
Build
Test
Release
Run

Oversight: Use Automated Audits To Monitor And Enforce Policy

With speedy releases, security pros clinging to manual governance of application security processes will quickly find that they must change. Fully understanding the inner workings of a single application in a production environment is hard enough, let alone understanding several applications changing swiftly. Furthermore, release decisions often require separate reviews from security, architecture, and operations. With each stakeholder having different objectives, developers and operations pros will more likely deviate from these cumbersome processes.[15] To avoid the risk of inconsistent oversight:

- **Govern application development with automated audit trails.** Automated tools throughout the SDLC create logs that show which person is responsible for each specific change. Security pros should learn how to gather the log data from these tools and identify potential policy violations. For example, if a developer or development team repeatedly fails quality gates based on results from an SCA tool, security pros can offer additional training about the dangers of including vulnerable third-party software.

- **Use the continuous delivery pipeline for high-risk changes.** Security pros cannot monitor all changes in all applications, especially when these releases occur quickly. Security pros should teach developers and operations pros to flag only high-risk security changes in the understand-and-invent SDLC stage. By reviewing these flagged changes early, security pros can familiarize themselves with the ramifications of the change and make corresponding changes to automatic testing and release quality gates to ensure that the changes are secure.

- **Enable proper authentication and authorization on all systems.** With DevOps, applications become consistent between development, testing, and production environments and tools rather than people creating and changing the environments. Once this has happened, security pros must work with operations pros to lock out all but critical personnel on application systems and supporting systems in test and production environments. Furthermore, work with developers to remove all direct calls from applications with user names and passwords, and use a service that can be queried to provide access. Ensure that proper logging of any authentication or authorization changes happen automatically with a judicious role-based access policy to the test and production systems including full auditing and accountability.[16]

- **Track drift across development, testing, and production environments.** Any out-of-process changes to an application and its supporting systems create differences in test and production environments, negating security scan results that start in the test environment. Work with operations pros to create alerts on drift of the application and supporting systems. For any drift you find, work with operations pros to either modify the model — using configuration management or application release automation tools, for example — or remove the change. Investigate repeated drifts as well, as they might indicate an intrusion.

- **Define quality gates as a part of the continuous delivery pipeline.** Use output data from security tools along with other testing tools as inputs to automate quality gates in the SDLC. Define what developers should be looking for with the security tools that will indicate a vulnerability and define which of those vulnerabilities operations pros or developers need to fix before release.

What It Means

The Future Is Security Tools That Self-Heal Applications

Layering security testing during the SDLC and then adding additional security in the production environment — such as WAFs or runtime application self-protection (RASP) tools — is like constructing a building to be as fireproof as possible but adding sprinkler systems in case fire occurs. However, in the future, tools will be able to identify vulnerabilities and then automatically make suggested fixes, basically becoming fully self-healing under certain conditions. The continuous delivery pipeline will enable this self-healing by allowing these security tools to automatically create, build, test, and deploy fixes to the application and its supporting systems. This future will not come quickly but will rather be a slow evolution of tools and a consolidation of vendors across a 10-year horizon. To prepare, security pros should look for vendors embracing machine learning and combining data from across different tools in the application security ecosystem to generate more intelligent information.

Supplemental Material

Companies And Thought Leaders Interviewed For This Report

We would like to thank the following individuals and companies who generously gave their time during the research for this report.

Andy Bustle	Blackboard

Daniel Ayala

Dell

Disney

Gene Kim

James Wickett

Jez Humble

Joshua Corman

Intuit

Matt Konda

Northwestern Mutual

nVisium

Orbitz

ProQuest

Endnotes

[1] For more information on how applications are being broken down into smaller and smaller releases via Agile and DevOps methodologies, see the Forrester report "DevOps Makes Modern Service Delivery Modern."

[2] For more information about release speeds, see the Forrester report "Boost Application Delivery Speed And Quality With Agile DevOps Practices."

[3] Source: "Verizon DBIR 2016: Web Application Attacks Are The #1 Source Of Data Breaches," Verizon Digital Media Services, June 21, 2016 (https://www.verizondigitalmedia.com/blog/2016/06/verizon-dbir-2016-web-application-attacks-are-the-1-source-of-data-breaches/).

[4] Source: "Hacking the Skills Shortage," Intel Security (http://www.mcafee.com/us/security-awareness/articles/hacking-skills-shortage.aspx).

[5] For more information about best practices around how to create an integrated product team that supports DevOps methodologies, see the Forrester report "Use DevOps Practices To Create A Lean And Responsive Application Delivery Organization."

[6] Source: "John Allspaw Discusses Devops and Continuous Delivery," Continuous Delivery, September 25, 2012 (http://continuousdelivery.com/2012/09/john-allspaw-discusses-devops/).

[7] For more on the threat of open source vulnerabilities, see the Forrester report "Vendor Landscape: Software Composition Analysis."

[8] Source: Jeremy Seth Davis, "Hackers exploit vBulletin flaw to access 27M accounts on 11 websites," SC Magazine, August 25, 2016 (https://www.scmagazine.com/hackers-exploit-vbulletin-flaw-to-access-27m-accounts-on-11-websites/article/530194/); Dan Goodin, "As we speak, teen social site is leaking millions of plaintext passwords," Ars Technica, September 27, 2016 (http://arstechnica.com/security/2016/09/social-hangout-site-for-teens-leaks-millions-of-plaintext-passwords/); and Phil Muncaster, "TalkTalk Breach: 17-year-old Confesses," info security, November 16, 2016 (http://www.infosecurity-magazine.com/news/talktalk-breach-17yearold-confesses/).

[9] For more information on how DevOps teams use PDCA to conduct experiments to continuously improve, see the Forrester report "Embrace Deming's PDCA Cycle To Continuously Optimize Modern Service Delivery."

[10] For more information on penetration testing tools and services as well as other security testing tools that can be used in the SDLC, see the Forrester report "TechRadar™: Application Security, Q2 2015."

[11] Source: Robin Mejia, "Red Team Versus Blue Team: How to Run an Effective Simulation," CSO, March 25, 2008 (http://www.csoonline.com/article/2122440/emergency-preparedness/red-team-versus-blue-team--how-to-run-an-effective-simulation.html).

[12] Source: “2014 Open Source Development and Application Security Survey Analysis,” Securosis, July 9, 2014 (https://securosis.com/assets/library/reports/Securosis_OpenSourceSurvey_Analysis.pdf) and “2016 State of the Software Supply Chain,” Sonatype, July 11, 2016 (https://www.sonatype.com/software-supply-chain).

[13] For more information on the SCA tool vendor landscape, see the Forrester report “Vendor Landscape: Software Composition Analysis.”

[14] For more information about ensuring security of security tools, see the Forrester report “Your Security Products Aren’t Secure.”

[15] For more information on how to govern with DevOps methodologies, see the Forrester report “Use DevOps And Supply Chain Principles To Automate Application Delivery Governance.”

[16] For a list of the most influential PIM vendors and how they stack up, see the Forrester report “The Forrester Wave™: Privileged Identity Management, Q3 2016.”

FORRESTER®

CHALLENGE THINKING. LEAD CHANGE.

We work with business and technology leaders to develop customer-obsessed strategies that drive growth.

PRODUCTS AND SERVICES

- › Core research and tools
- › Data and analytics
- › Peer collaboration
- › Analyst engagement
- › Consulting
- › Events

Forrester's research and insights are tailored to your role and critical business initiatives.

ROLES WE SERVE

Marketing & Strategy Professionals

CMO

B2B Marketing

B2C Marketing

Customer Experience

Customer Insights

eBusiness & Channel Strategy

Technology Management Professionals

CIO

Application Development & Delivery

Enterprise Architecture

Infrastructure & Operations

› Security & Risk

Sourcing & Vendor Management

Technology Industry Professionals

Analyst Relations

CLIENT SUPPORT

For information on hard-copy or electronic reprints, please contact Client Support at +1 866-367-7378, +1 617-613-5730, or clientsupport@forrester.com. We offer quantity discounts and special pricing for academic and nonprofit institutions.

Forrester Research (Nasdaq: FORR) is one of the most influential research and advisory firms in the world. We work with business and technology leaders to develop customer-obsessed strategies that drive growth. Through proprietary research, data, custom consulting, exclusive executive peer groups, and events, the Forrester experience is about a singular and powerful purpose: to challenge the thinking of our clients to help them lead change in their organizations. For more information, visit forrester.com.