EBOOK

Contrast SECURITY

Developers Need Integrated Application Security Tooling

Executive Overview

Traditional approaches for application security testing create problems for DevOps. Co-dependent workflows and poor communications support put development and security teams at odds with one another. The resulting inefficiencies slow down delivery cycles while leaving applications riddled with unaddressed vulnerabilities. Security must become a shared, collaborative concern that unites development, operations, and security teams without inhibiting aggressive delivery schedules. In response, organizations need to embrace modern application security that integrates testing with existing DevOps tools.

> 55% Of security professionals said it is difficult to get development teams to prioritize remediation of vulnerabilities—even if it's a performance metric for developers.[1]

[1] "2019 Global Developer Report: DevSecOps," GitLab, July 2019.

Table of contents

01 Applications Need Better Collaboration Between Development and Security

02 Security Integration Across Devops Tooling

03 Cross-Team Visibility

04 Automation and Verification

05 Realizing the Devsecops Ideal

01

Applications Need Better Collaboration Between Development and Security

The vast majority (79%) of DevOps teams are under increasing pressure to shorten release cycles.[2] But traditional approaches to application security remain a critical problem for developers due to valuable time being spent identifying, analyzing, and remediating vulnerabilities. Organizations pay a heavy price for lack of workflow orchestration across development, operations, and security—and this lack of coordination impacts delivery dates. Two-thirds (67%) of enterprise IT leaders say the lack of application workflow orchestration leads to services not being completed in time for the business.[3]

Developers are not empowered to incorporate security needs at the pace of business because they usually get security feedback outside their own workflow and completely out of context. When an application security analyst sends a code vulnerability back for remediation, the developer must stop working on their current project and go back to the previous code to troubleshoot—a problem known as "context switching."[4] Security becomes a disruptive sub-loop that halts forward progress and jeopardizes delivery schedules. Security and development teams work at cross purposes rather than toward common shared goals of fast delivery and high-quality code.

As a result of this tension, many companies forego thorough security testing in order to accelerate time to market—which can leave code with vulnerabilities, including those that are serious. More than half (55%) of organizations sometimes skip security scans to meet deadlines.[5] Skipping security processes for speed only pushes remediation downstream in the software development life cycle (SDLC)—where repairs become significantly more expensive. This also exposes customers to cyberattacks. And this is a significant concern with applications remaining a top attack surface for hackers looking to exploit widespread vulnerabilities in critical business systems. The percentage of data breaches tied to application vulnerabilities doubled over the

[2] "The State of DevSecOps Report," Contrast Security, December 2020.

[3] Louis Columbus, "How To Accelerate DevOps Cycles And Keep Customers First," Forbes, October 30, 2020.

[4] "Modernize your CI/CD," GitLab, accessed November 25, 2020.

[5] "The State of DevSecOps Report," Contrast Security, December 2020.

past year to account for 43% of all reported incidents,[6] and 95% of organizations surveyed indicate they experienced at least one successful application exploit in the past year.[7]

While security is important to developers, it traditionally hasn't been included in their responsibilities. Nearly three-quarters of DevOps teams report being inadequately prepared to deal with application security requirements.[8] But this is starting to change as organizations look to embrace more collaborative DevSecOps models for their environments.

> *Nearly 90% of developers report that the largest hindrance to productivity is a disconnect between development and security workflows.*[9]

[6] "2020 Data Breach Investigations Report," Verizon, June 2020.

[7] "The State of DevSecOps Report," Contrast Security, December 2020.

[8] Derek Rogerson, "State-Of-The-Art AppSec Goes Beyond Perimeter Into Application Runtimes," DZone, June 7, 2020.

[9] Brandon Vigliarolo, "Developers agree: Application security processes have a negative impact on productivity," TechRepublic, June 30, 2020.

02

Security Integration Across Devops Tooling

Developers need remediation clarity and orchestration in order to make accurate and actionable software fixes.[10] Integrating modern application security with existing tools, workflows, and processes can help streamline and automate vulnerability management while enabling teams to collaborate more effectively and efficiently. These DevOps tools include:

- Integrated development environment (IDE) systems
- Chat tools
- Ticketing systems
- Continuous integration/continuous deployment (CI/CD) tooling
- Security information and event management (SIEM) solutions
- Repositories (e.g., Git services)

Leading companies have started adopting tooling that aligns security and development throughout the development pipeline because integrating security into the software creation and delivery process unlocks rapid remediation capabilities. Developers have more context into how specific vulnerabilities are being pulled into their application when they can relate the vulnerability back to the dependency.

> *The goal is to make security part of the software development workflow, with secure coding best practices and testing automation, rather than bolting it on later in the cycle.*[11]

[10] Derek Rogerson, "State-Of-The-Art AppSec Goes Beyond Perimeter Into Application Runtimes," DZone, June 7, 2020.

[11] Jaikumar Vijayan, "6 DevSecOps best practices: Automate early and often," TechBeacon, accessed December 17, 2020.

03

Cross–Team Visibility

Deep, high-quality integration fosters remediation without disrupting DevOps workflows. Organizations can custom configure policies per vulnerability type and send automatic status notifications to track issue status, clarify responsibilities, and facilitate transparent communications across all teams.

IDE. Co-dependent and asynchronous workflow processes between developers and security teams create tension and delays. The limitations of de facto security testing procedures obstruct developers from insights that could help them write better, more secure code in the context of it being written. Application security integration with IDE tools (such as Visual Studio) can help developers write code with fewer vulnerabilities by giving them immediate access to remediation guidelines and security recommendations during coding. This obviates the need for cumbersome back-and-forth remediation cycles with security teams later in the SDLC.

TICKETING. Many ticketing systems require manual alert triaging and assigning of task ownership for vulnerability remediation, which is slow and can create cross-team tensions. Lack of automation capabilities makes it difficult to facilitate status awareness and resolution of issues as well. This inhibits rapid notifications and ultimately slows down time to resolution.

Application security integration with ticketing systems (e.g., Jira, Azure Boards) can streamline resolution processes by assigning ownership and publicizing resolution status to help motivate remediation. It can also provide proof of remediation to verify a fix has been made. In terms of capabilities, effective integration can:

- Automatically generate tickets, synchronize comments, and push notifications about applications • Allow multiple vulnerabilities to be sent as a single issue
- Assist with downstream reporting processes for compliance auditing requirements
- Offer the ability to centrally manage and verify issues have been viewed, worked on, and closed (unified management from submission to resolution)

Integrating security with ticketing also offers benefits such as:

- Enable productivity and efficiency by improving communication and transparency in and between teams
- Help prioritize the most critical vulnerabilities
- Enable cross-team/department management and remediation of issues

CHAT TOOLS. Tools like Slack and Microsoft Teams help developers communicate and collaborate with other teams in real time—such as sending a notification that a ticket has changed. Application security integration with the existing chat tools can give developers as well as operational and security teams instant access to automatic security updates based on the severity of a vulnerability (as defined by DevOps policies). Chat tools help reduce direct communication tensions between different teams and provide transparency about an application's vulnerability and remediation status.

When application security is integrated with chat tools, positive business outcomes can include:

- Real-time notification of vulnerabilities and their priority for remediation, as well as build/test cycle completion status
- Reduced lag time for all team members learning about an issue
- Real-time communications that help foster better collaboration across teams

> “*It is essential in devsecops to communicate the responsibilities of security processes and product ownership. Only then can developers and engineers become process owners and take responsibility for their work.*[12]

[12] "DevSecOps," IBM, July 30, 2020.

04 Automation and Verification

Effective verification and automation accelerates development processes—including teams making sure there are no new issues introduced in a build. Security processes cannot benefit from policy-based automation within DevOps workflows unless they are first integrated with CI/CD tools such as Jenkins and Azure Pipeline.

CI/CD. Application security integration with CI/CD incorporates automatic testing (based on predefined policies) within the development pipeline for greater agility. This automation helps teams accelerate cycles while verifying that there are no new critical issues introduced into the build.

SIEM. Slow or error-prone manual processes increase the likelihood that a critical issue will end up costing significantly more to repair at a later stage in the SDLC. Application security events and known vulnerabilities can be integrated into operations tools to centralize tracking, collection, analysis, and notification of events.

> *“Devsecops is defined as automating cybersecurity processes and controls via integration with the CI/CD toolchain that orchestrates the application lifecycle.*[13]

[13] “Oracle and KPMG Cloud Threat Report 2020,” Oracle/KPMG, May 2020.

05

Realizing the Devsecops Ideal

Integrating application security into DevOps can be a challenge. But the urgency of going to production without proper security testing practices will only lead to a buildup of defects that will cost more in the long run.[14]

From IDE platforms, to ticketing and ChatOps systems, to the broader CI/CD pipeline, security must be built into the existing workflows without adding additional churn to delivery cycles. Effective security integration with the common tools used by developers and operations teams helps organizations accelerate the process of realizing a true DevSecOps model. As a result, they gain the benefits of greater agility, more secure products in production, and fewer problems later in the SDLC.

> “
> *Devsecops makes application security an integrated, automated part of the software development life cycle, which means that security vulnerabilities just go into the issue tracker like anything else.*[15]

[14] Ruslan Desyatnikov, “Nine Best Practices For Integrating Application Security Testing Into DevOps,” Forbes, July 5, 2019.

[15] Jonathan Knudsen, “New survey shows integrating application security testing gaining traction in DevOps,” Security Boulevard, November 2, 2020.

Contrast Security provides the industry's most modern and comprehensive Application Security Platform, removing security roadblocks inefficiencies and empowering enterprises to write and release secure application code faster. Embedding code analysis and attack prevention directly into software with instrumentation, the Contrast platform automatically detects vulnerabilities while developers write code, eliminates false positives, and provides context-specific how-to-fix guidance for easy and fast vulnerability remediation. Doing so enables application and development teams to collaborate more effectively and to innovate faster while accelerating digital transformation initiatives. This is why a growing number of the world's largest private and public sector organizations rely on Contrast to secure their applications in development and extend protection in production.

240 3rd Street
2nd Floor
Los Altos, CA 94022
Phone: 888.371.1333
Fax: 650.397.4133

Contrast SECURITY

contrastsecurity.com