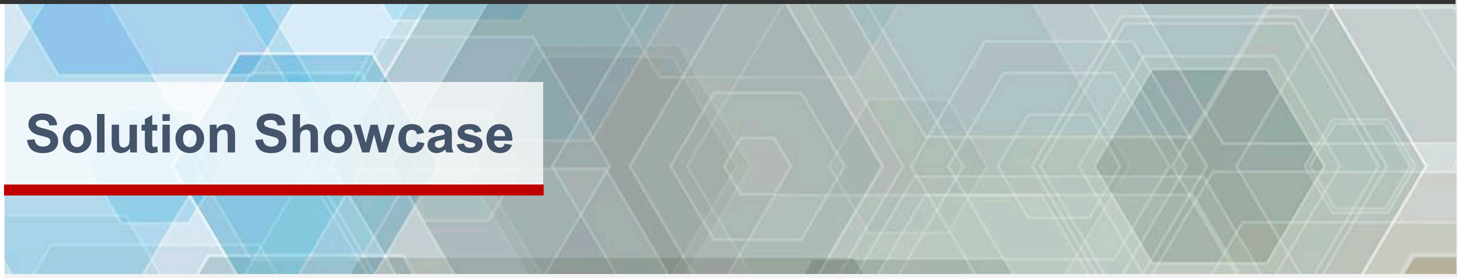


Solution Showcase

Automating Security for DevOps: Best Practices for Securing the Continuous Integration and Continuous Delivery (CI/CD) Pipeline

Date: November 2018 **Author:** Doug Cahill, Senior Analyst and Group Director

Abstract: Fully leveraging the agility of public cloud infrastructure-as-a-service (IaaS) platforms requires embracing DevOps processes that enable businesses to bring applications to market quickly and efficiently. Many companies are now realizing these benefits by automating the continuous integration and continuous delivery (CI/CD) of their applications. Cybersecurity, however, is too often not part of the shift to DevOps, a missed opportunity to efficiently improve an organization's cybersecurity posture by building security into the CI/CD pipeline. While organizations are interested in integrating security with DevOps processes (i.e., "DevSecOps"), they find it difficult to find specifics for how to get started. Following are a set of best practices that will help cross-functional scrum teams at organizations of any size, inclusive of application owners, developers, operations teams, and cybersecurity champions, to leverage automation to assure the right cybersecurity measures are applied at each step of the CI/CD pipeline.

The DevOps/Cybersecurity Schism

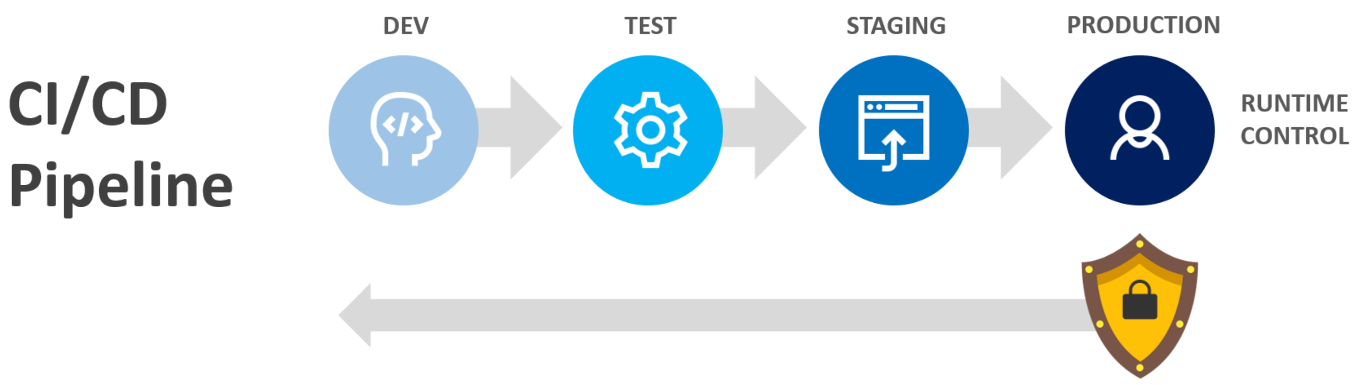
The current divide between development, operations, and cybersecurity teams is rooted in competing objectives. Developers are charged with delivering new applications to production as quickly as possible, while security professionals are tasked with protecting their organizations against a range of cybersecurity threats. The misalignment of speed and diligence is also due to a misunderstanding—DevOps can be perceived by security practitioners as simply moving too fast to be secure while security implies traditional waterfall processes of phases and gates that developers perceive as slowing things down.

The resulting lack of collaboration results in developers often using default settings when standing up new cloud services. For example, developers may not scan new server workloads for software and configuration vulnerabilities before deploying them externally, making them susceptible to port scanning. Similarly, not changing default settings on object stores can leave them open to public access.

The call to "shift security left" to the development phase needs to be augmented to also shift security right to automate the deployment of runtime controls in production environments.

As DevOps adoption expands into the enterprise, organizations need to shift their cultures and mindsets to view these two domains as complementary and symbiotic. The call to "shift security left" to the development phase needs to be augmented to also shift security right to automate the deployment of runtime controls in production environments. But how do organizations start down the path of internalizing DevSecOps and putting such use cases into play? Bridging the disconnect requires a focus on the three fundamental elements of a cybersecurity program—people, process, and technology (see Figure 1).

Figure 1. Shift Left Security

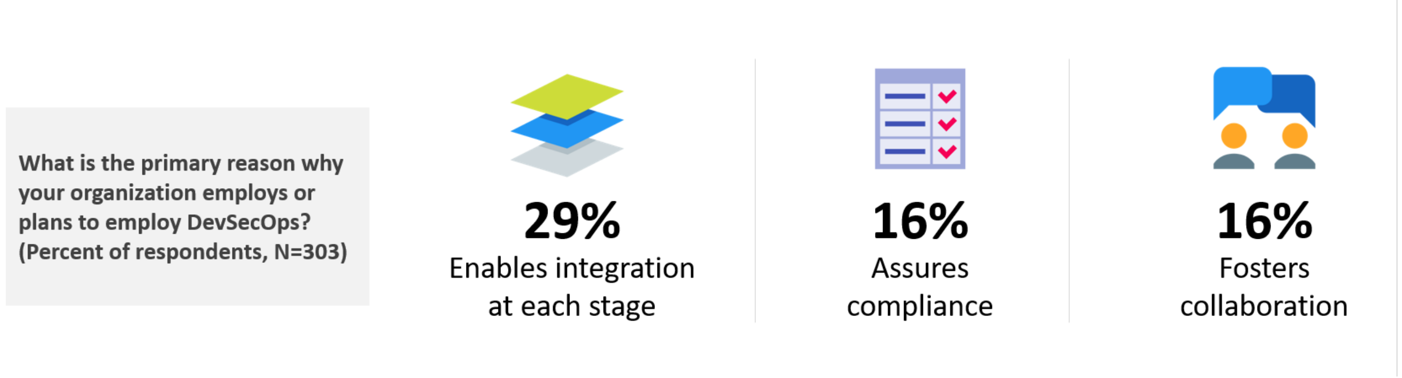


Source: Enterprise Strategy Group

Making the Case for Automating Security via DevOps

Security is emerging as a suitable DevOps use case, with 34% of organizations who participated in ESG research sharing that they have incorporated some level of security into their DevOps processes (“DevSecOps”) or plan to. There are multiple reasons behind the interest in DevSecOps, including the benefit of improving organizations’ security postures by integrating security at every stage of their CI/CD tool chain (see Figure 2).[1]

Figure 2. Reasons to Employ DevSecOps



Source: Enterprise Strategy Group

Another reason for integrating security processes and controls with DevOps is to assure ongoing compliance with industry regulations via configuration and auditing capabilities. To address the DevOps/cybersecurity schism, participants in ESG research respondents also cited fostering a high level of collaboration between various stakeholders and DevSecOps as a means to think more proactively about cybersecurity as a benefit. However, while the benefits of DevSecOps are generally

[1] Source: ESG Master Survey Results, *Trends in Hybrid Cloud Security*, March 2018. All ESG research references and charts in this solution showcase have been taken from this master survey results set.

understood, the same research study revealed that 40% of organizations are currently evaluating security use cases that leverage their DevOps process, highlighting a need for a prescriptive set of measures for how to get started.

Putting DevSecOps Into Action

Automating security via integration in the CI/CD pipeline starts with gaining organizational alignment, defining stage-specific use cases, and utilizing purposeful controls.

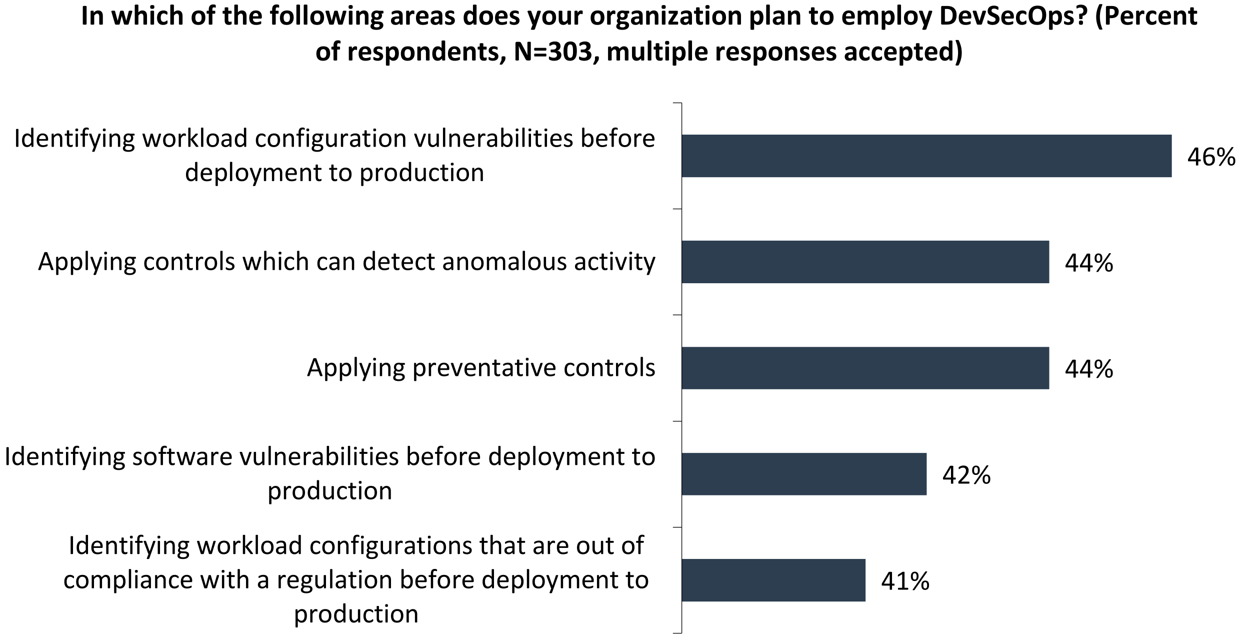
Securing the CI/CD Pipeline Is a Risk-based Shared Responsibility

The shared responsibility security model is a cloud security framework that outlines the obligations between a cloud service provider (CSP) and the consumer of the service(s) for securing cloud infrastructure, applications, and data. This concept, the notion of shared responsibility, also serves to convey how security and development teams should view securing application stacks. The core guiding principle is to agree that different applications have different risk profiles by virtue of their relative criticality to the business and their own specific exposure to cyber threats, based on where they're deployed, who has access, and the tiers of the stack. Using risk and threat modelling as a starting point allows cross-functional teams to define the appropriate cybersecurity policies from which the right processes and technologies can be applied. A risk-based approach allows for a practical means to prioritize security measures.

Define User Stories by Stage and Environment

Continuous integration and continuous delivery processes automate the building, testing, and deployment of applications. Development and cybersecurity teams should collaborate with product owners to author agile user stories for the development and build stages of pre-deployment environments and the runtime stage of production environments. ESG research reflects such an approach, with participants citing plans to employ a set of pre-deployment and runtime DevSecOps use cases (see Figure 3).

Figure 3. Top Five Plans for How to Employ DevSecOps



Source: Enterprise Strategy Group

Research participants cited plans to vet server workloads pre-deployment by identifying configuration and software vulnerabilities. For runtime controls, 44% of respondents stated plans to apply preventative controls and 44% also reported plans to apply controls that detect anomalous activity. Further along the attack chain, 39% of organizations also indicate plans to capture system activity to ready themselves for incident response investigations. Bolting in security via DevOps means aligning with the stages of the CI/CD pipeline and their associated environments to automate these use cases cited by participants in ESG's research.

Pre-deployment via SDLC Integrated Controls

Development Stage

Integrating security controls within the software development lifecycle (SDLC) should include automating the following security practices:

- **Static analysis** to identify inadvertently introduced vulnerabilities within the IDE (integrated development environment). Successful code scans should be a requirement for checking code into a source code repository.

- **Composition analysis** to establish a bill-of-materials inventory of all the components of a build tree, including the inclusion of open source software (OSS) that may include vulnerabilities.

Testing Stage

As part of the build process and testing phase, server workloads, inclusive of application code, should be verified for system integrity and hardened before deployment to production by vetting compliance with industry-standard configuration benchmarks such as those defined by the Center for Internet Security. Because many organizations treat production server workloads as immutable infrastructure in that they are not patched once deployed, known software vulnerabilities should be identified and applicable patches installed pre-deployment. These steps should also be applied for application containers by scanning registry-resident container images. Organizations should not trust images sourced from a public registry and should also assess their trustworthiness via these steps. Successful automated configuration assessments and vulnerability scanning should serve as the green light to ship containers and workloads to production environments.

Successful automated configuration assessments and vulnerability scanning should serve as the green light to ship containers and workloads to production environments.

Runtime via Automation/Orchestration Integrated Controls

Having "shifted security left" by automating security checks before deployment, scrum teams should also implement user stories that define the automated introduction of runtime cybersecurity controls via integration with the orchestration tools that ship code to production. Runtime security for server workloads includes anti-malware, anti-exploit, drift-prevention (e.g., file integrity monitoring), application control, and access segmentation controls. Central to automatically applying the right policy for a specific server workload is the use of name:value pair tags that denote the role and environment of a workload. Automating the application of runtime controls via policy-driven tags assures that server workloads are protected upon deployment to production. And this also applies to securing the automation environment itself, which represents a vector for adversaries. Integrating runtime controls in the staging phase to automate securing production workloads is essential to keep pace at scale.

Automating the application of runtime controls via policy-driven tags assures that server workloads are protected upon deployment to production.

Employ Security Controls Designed for CI/CD Integration

Implementing pre-deployment and runtime DevSecOps use cases requires utilizing cybersecurity controls such as cloud workload protection platforms that have been designed for CI/CD toolchain integration. That is, cybersecurity teams need to provide development and operations teams with security controls that work with the IDE, build, source code management, and orchestration tools they already use so that security can be truly built into the CI/CD pipeline. Such purposeful cybersecurity controls will support and provide the following:

- **Out-of-the-box scripts** for integration into Chef, Puppet, etc.
- **Support for tags** that convey roles and automate the assignment of policy.
- **Awareness to temporal instances** in auto-scaling groups.
- **API-driven and API-aware implementation** so they can both be instrumented and provide visibility and control into the use of IaaS services.

The Bigger Truth

As organizations continue to use more cloud-services as business-critical additions to their arsenals, they need to protect their modern infrastructures from cyber-threats. Getting started securing today's application development and delivery environments requires a change in approach, one that starts with the same DevOps cultural shift which brought development and operations teams closer together. Agile software development methodologies that also foster a high level of collaboration are essential for cross-functional scrum teams to prioritize the implementation of stage- and environment-specific cybersecurity user stories. Implementing these user stories by integrating security into the CI/CD toolchain requires cybersecurity controls designed for the job. Introducing cybersecurity process and controls into the DevOps processes that manage cloud-delivered applications should not be viewed as posing a risk to the agility of the cloud, but as an opportunity to gain operational efficiencies via automation.

Enterprise Strategy Group is an IT analyst, research, validation, and strategy firm that provides market intelligence and actionable insight to the global IT community.