cloudbees

Amazic
WHITEPAPER

# DevSecOps: Speed and Security Together, at Last

cloudbees

# Contents

INTRODUCTION

## If you're like most organizations these days, DevOps is a term that seems to find its way into practically every business and technical conversation. The vast majority have programs at least in the planning stage that bring together functions of software development (Dev) and software operations (Ops). While estimates still put the percentage of firms practicing DevOps in its purest form at under 20 percent, the concepts integral to the practice – automation, collaboration, short rapid development cycles and increased deployment frequency – are taking hold in the enterprise.

Even as DevOps practices are being increasingly adopted, security has often remained a siloed function, applied at the end of the process, independent of the continuous, collaborative work being accomplished by the DevOps team.

In recent years, following a rash of software breaches, security is at the forefront of IT priorities, and naturally being brought to the table with DevOps discussions. Organizations are pursuing new DevSecOps practices that integrate security with Dev and Ops, putting a new emphasis on incorporating security during the continuous integration (CI) and continuous delivery (CD) process. It's the emerging model for continuous security.

DevSecOps represents a new perspective on how to best approach security. What drove this change? What does the change mean for organizations trying to implement it? And how do you move forward to where DevSecOps is headed? This whitepaper dives into the details of DevSecOps and offers guidance on how to implement it in your organization.

# The Market — Demand for Speed and Innovation

## EVERY BUSINESS IS A SOFTWARE BUSINESS

It's long been accepted that advancements in software delivery are helping companies define who they are and how they compete. This holds true from poultry farms in Jamaica to SpaceX and everybody in between. Every business is under pressure to innovate. If you're not growing and gaining market share, the market perceives your company as failing. And if you're not innovating fast enough you run the risk of losing market share to those who are. To gain and ultimately retain market share, you need to innovate fast.

## INCREASED VELOCITY INTRODUCES NEW CHALLENGES AND RISK

Beating out competitors requires organizations to deliver software at higher speeds. But that comes at a price. As we transition from three-, six- and 12-month delivery cycles to a world of continuous delivery and DevOps, where organizations are expected to innovate and deliver new features weekly, daily or by the minute, we encounter new challenges and risks.

Businesses that are used to building software according to careful, regimented processes now have to make sure they're crossing their t's and dotting their i's.

How do you maintain audits that traditionally took weeks or months and do them in weeks, days or even hours? How do we ensure we're performing critical regression tests on business-critical software when we have even more limited time to run tests? How do we ensure all of our gates – our checks and balances – are honored and not skipped? When we require a manager's approval to deploy, such as frequently seen in financial industries, how do we ensure we have that approval? How do we deal with the fact that we need speed when we have a wide-ranging portfolio including Java, mainframe, mobile, Cobol or Haskell?

## TAKING GOVERNANCE INTO ACCOUNT

As software delivery speeds up, introducing more risk into the system, organizations have had to wrestle with a series of underlying questions: Who's responsible for ensuring everything goes smoothly? And what rules and policies does everybody need to follow? There's no single, clear cut answer – especially when change happens on a regular basis. In an environment defined by the new concept of continuous everything, organizations are integrating, testing, deploying and analyzing on a continuous loop, which requires businesses to set up continuous governance procedures to keep everything in line.

# The Problem: Speed vs. Security

## ADDRESSING VULNERABILITIES

Moving at high speeds, organizations have to ensure they are identifying and addressing security vulnerabilities. Recent incidents have shown us just how important security is, not only for the well being of customers and users, but for the entire company. Security breaches at Yahoo!, Equifax and Facebook have exposed users' personal information, costing the companies millions of dollars and damaging their reputations. Hackers have spread open source software bugs such Heartbleed and WannaCry throughout the world, causing disruptions in businesses of all sizes. The losses from the WannaCry ransomware attack alone are expected to cross $4 billion.

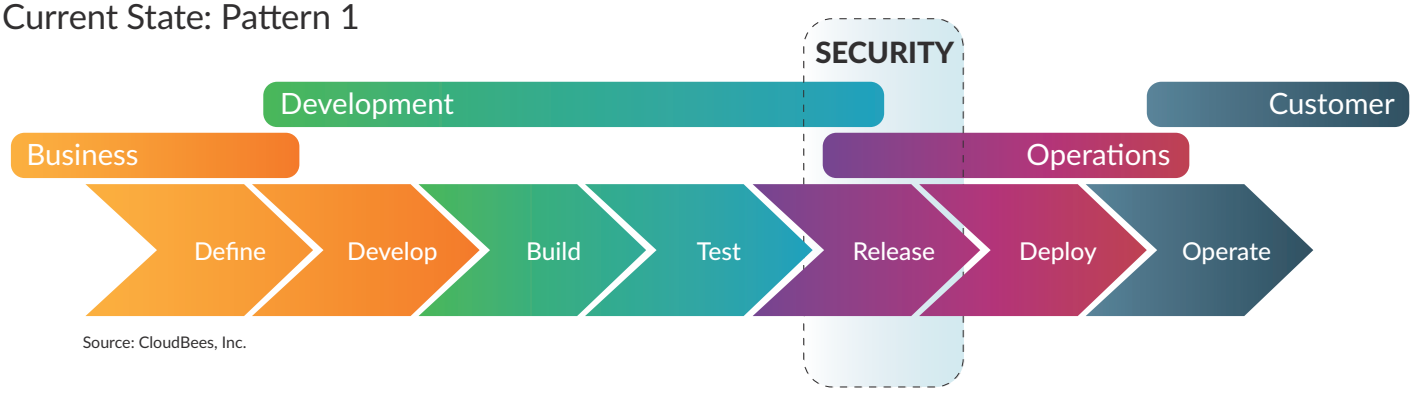
## IT'S SECURITY'S PROBLEM

In software development organizations, the mindset has traditionally been that the security organization is responsible for any and all security issues. A recent Sans Institute survey shows that organizations still assign security testing responsibilities to a handful of job functions – the internal security team, followed by security consultants, then quality assurance, then cross-functional teams or DevSecOps. A few steps farther along the track, the development team, the system architect and the business owner – the ones who are defining and creating the functionality – enter the security process.

According to the 2017 State of Application Security: Balancing Speed and Risk report by the Sans Institute, the development team is most responsible for corrective action. This means the developer has the job of addressing issues if they come up, but vulnerability testing is security's job. This creates a mindset where the security team is perceived as blocking the development team's progress and its ability to innovate. Organizations realize they need to make a move to implement DevOps across the entire lifecycle, from concept to deployment, but they haven't figured out how to effectively integrate security and compliance in a mature DevOps lifecycle. In Figures 1 and 2, the illustrations represent our current state of security in the software development lifecycle (SDLC).

*Figure 1*

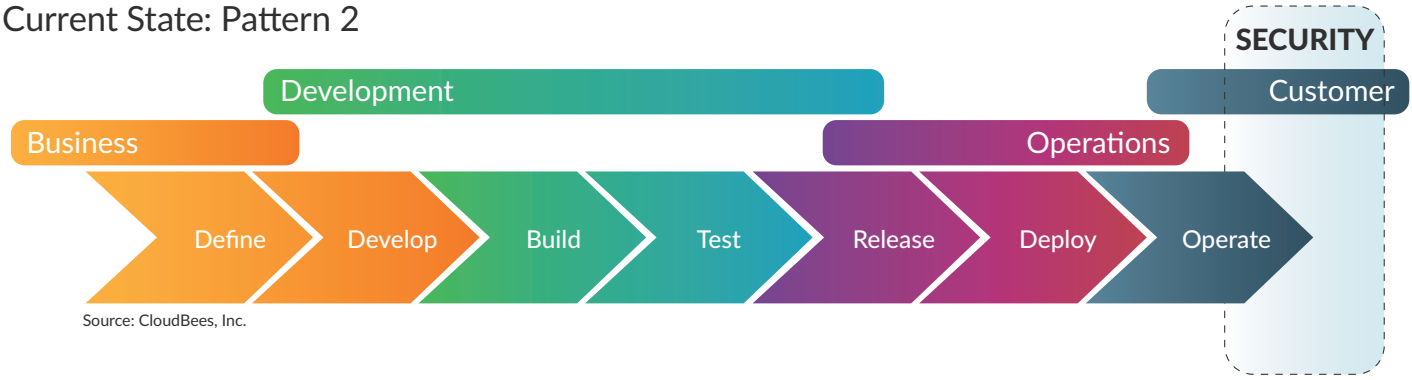
Current State: Pattern 1




Source: CloudBees, Inc.

*Figure 2*

Current State: Pattern 2




Source: CloudBees, Inc.

### OBSTACLES TO SECURITY WITH SPEED

What are the three top challenges in implementing application security for production systems? We like to call them the ABC challenges for DevSecOps at CloudBees®:

- **A** lack of an integrated and automated workflow
- **B**ridging the gap between software development security and compliance
- **C**lear lack of application security skills, tools and methods

Automation across silos and engaging stakeholders to define that automation can help break down silos between teams and business units as well as bridge the gap. Everybody needs to be responsible for security, but your average developer doesn't truly understand what security is. You want to automate your security process, but you don't understand how.

## The Solution: Dev + Sec + Ops

### WHAT IS DEVSECOPS?

DevSecOps is the answer to issues that have challenged DevOps organizations. It puts a renewed emphasis on security to detect and eradicate vulnerabilities early and often. It makes security everybody's responsibility, from the business, through development, QA and operations. It automates security functions, removing obstacles to the goal of achieving security with speed.

DevSecOps is like DevOps in the sense that it aims to drive more efficiency and productivity through team collaboration. But DevSecOps incorporates security principles in the overall process. Security leaders work with developers in every step of the process, not just at the end, the way traditional security approaches

work. While DevSecOps relies heavily on security leaders to offer expertise, it becomes the province of all technology workers involved in software delivery.

DevSecOps represents a mentality as much as a list of best practices. Like DevOps, it represents a cultural shift where stakeholders are aligned on delivering software not just with speed, but also equally with security. In fact, DevSecOps proposes that moving faster can improve security. It involves baking security into the process, continuously anticipating and checking for problems rather than applying security after the fact, when it may be too late.

Understanding and implementing DevSecOps can be aided by focusing on six key concepts.

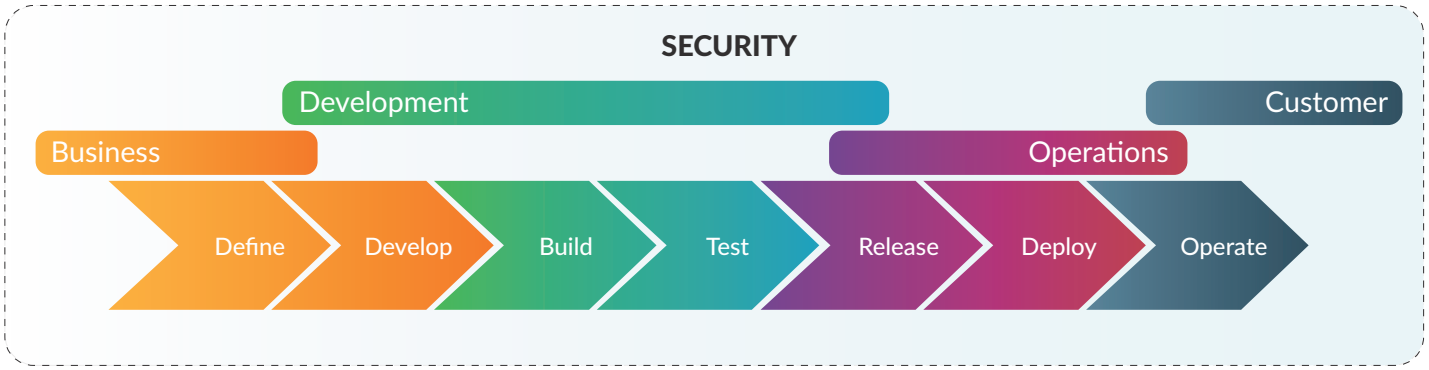
- **Security as code.** Security activities, have traditionally been performed as a manual step late in the process, slowing down the delivery of features, and making fixes more costly. DevSecOps integrates scans, tests and vulnerabilities as code or scripts, which can be automated and replicated, enabling security verification continuously throughout the software lifecycle.
- **Shift left.** Shifting security left starts security activities earlier, during development, and extends them throughout software development, deployment and production, incorporating continuous feedback at every stage. This not only means automating security checks, but also involving security experts as part of an agile team at inception of a project or story and ensuring others (engineering, QA and others) are educated and informed on security practices and issues.
- **Empower teams.** Security needs to be everybody's responsibility – and not just in the hands of security experts. The experts can serve as the overseers, recommending tools, processes and best practices. But the business, developers and quality assurance have to step forward to learn the discipline and integrate security activities into their daily tasks. Organizations have to empower them with the tools and knowledge to better embrace and enforce security.
- **Security visibility.** Security has to come out of the shadows. DevOps teams have tended to circumvent security practices in favor of moving fast and perfecting their code. What they don't understand, they tend to avoid. They have had their own procedures to follow, their own metrics to meet. DevSecOps integrates security metrics onto the DevOps list of priorities, allowing security to be tracked and measured like other tasks.
- **Continuous security.** Integrating security throughout the process sets up triggers to help an organization respond to threats at any stage with security activities that are performed continuously. Developers play an active role in planning security functions, teams enter projects understanding the risks involved and security leaders work with coders to institute safe practices. Checkpoints are set up to track changes, test for security flaws and activate remediation measures on a continuous basis.

## PURSUE SOUND SECURITY STRATEGIES

Bugs happen. They're part of the process. DevSecOps recognizes that organizations will never keep vulnerabilities out of the software life cycle – or stop hackers altogether. Integrating security principles throughout the life cycle and tying them to overall organizational risk strategies will minimize issues going forward. In order to do this, the industry needs to think about the shift left concept and ensure security is incorporated earlier, not later. As a result, security will be included throughout the SDLC process as in Figure 3.

*Figure 3*

Ideal State: Pattern 3




Source: CloudBees, Inc.

## DEVSECOPS VS. DEVOPS

How is DevSecOps different than DevOps? The reality is DevOps should address not just the handoff from development to operations but the entire software development life cycle which includes security. By definition, DevOps should include security.

However, due to the legacy security procedures and the siloed nature of organizations, it's not always clear that security is an integral part of the DevOps process. Including the word security signals to everyone involved, especially the security team, that the organization views security as a core part of the continuous delivery process. As organizations mature, we may no longer specifically call out the term DevSecOps. It will be assumed. If you're a DevOps organization, you'll be practicing security.

## SPEED HAS ITS BENEFITS

CD and DevOps doesn't just provide a path to balance speed and security. What organizations are realizing is that with CD and DevSecOps, working at higher speeds can actually improve security. When you put processes in motion to move faster, teams are forced to collaborate, automate and standardize their software delivery workflows. As software delivery life cycles and CD pipelines become more standardized, teams

inherently get more visibility into their procedures and can exercise more control and governance. By reducing the variation in your CD pipelines and automating them, you ultimately reduce your risk of having unwanted changes introduced which can result in security vulnerabilities.

Also, with CD, you tend to make smaller changes and deliver updates more frequently. This makes it easier to test if a change introduces a security vulnerability, rapidly identify any security flaws and correct it quickly. Moving faster with more incremental changes speaks back to the CI mantra of *fail fast* and this applies to security too.

Making changes incrementally, rather than all at once, provides more security protection for the system as a whole by changing the attack surface on a continual basis. This makes it more difficult for hackers not only to exploit vulnerabilities but also to identify them in the first place.

### AUTOMATING THE INFRASTRUCTURE

More often than not organizations will automate their infrastructure through infrastructure as code or container technology. With DevSecOps, we can do a better job protecting the attack surface by controlling access to our environments and having our environmental configurations well codified and documented. This makes it easier to track changes and test for vulnerabilities.

Another aspect of container technology is that they don't stay static for long, often running on ephemeral, cloud-based platforms. This makes it harder for an attacker, inside or outside, to leverage the build and test infrastructure and attack that application.

## Instituting DevSecOps

Organizations will face three key challenges as they seek to move to DevSecOps. They will need to change their *process* – instituting new governance models, workflows and overall processes. They will need to adopt new *technology* – adding tools that automate steps in the testing process. And they'll need to overhaul their *culture* – setting up new chains of communication and encouraging buy-in on the changes in process and technology.

### PROCESS OPTIMIZATION

To succeed, a DevSecOps effort needs to open up new lines of communication and collaboration. Organizations need to put mechanisms in place to unify the teams across functional silos, using communication and collaboration tools, reporting systems and metrics. They need to create feedback loops that promote process improvement and embrace a continuous improvement approach.

## TECHNOLOGY TRANSFORMATION

Adding tools to automate security testing removes a significant chunk of the guesswork and frustration from the software delivery process. Integrating DevSecOps tools that perform scripting, static and dynamic analysis and composition analysis with existing tools and processes will detect vulnerabilities earlier and facilitate better overall workflows.

## CULTURE SHIFT

To give security its seat at the table, organizations need to create an atmosphere of trust and cooperation. They need to make a commitment to training and learning, establish feedback loops, empower security champions and promote decision-making across teams.

## BRIDGING THE GAP

Bridging the gap between software development, operations, security and compliance requires collaboration and commitment, in addition to establishing new workflows and processes. Organizations that have succeeded in balancing speed and security report to have adopted more effective testing methods across the software development life cycle, building cross-functional teams and encouraging communications across teams and silos.

Moves to integrate technologies more fully are also playing important roles in breaking down silos and reducing risks. Studies show teams that perform end-to-end testing, automate end-to-end workflows and integrate IT tools, make more progress toward their goal of achieving fully functioning DevSecOps.

## DEVSECOPS' BENEFITS

DevSecOps offers many benefits to organizations that achieve maturity in the process. Today's market is moving fast, and every company surely has competitors that are implementing DevSecOps correctly. They're innovating and gaining a market advantage.

Here are a few benefits DevSecOps brings:

- Greater speed and agility for security teams
- An ability to respond to change and needs rapidly
- Better collaboration and communication among teams
- More opportunities for automated builds and quality assurance testing
- Early identification of vulnerabilities in code
- Team member freed for higher-value work

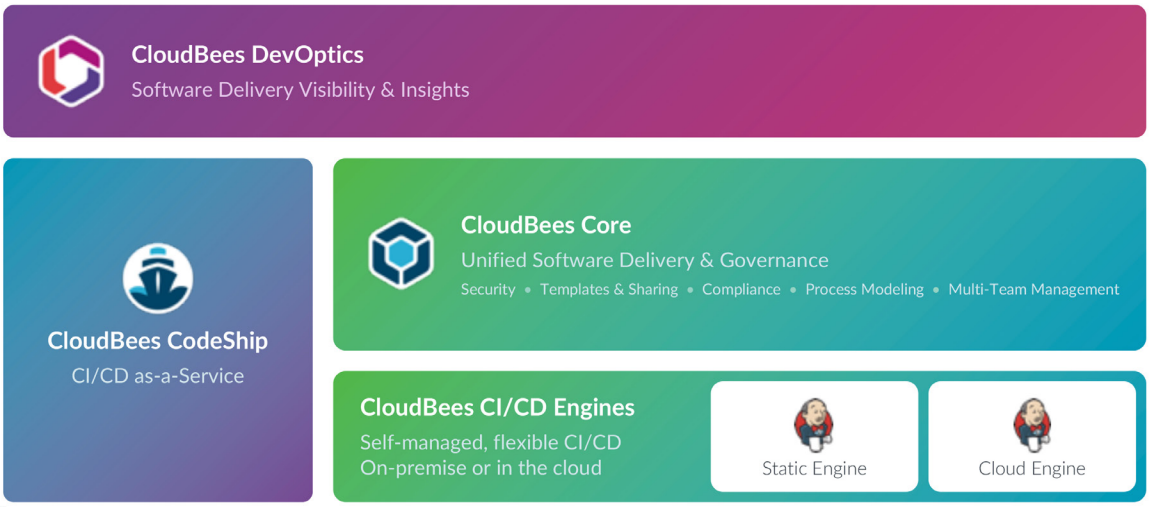
## CloudBees and Continuous Security

At CloudBees, we are building the world's first end-to-end continuous software delivery system -- powering the continuous economy and solving DecSecOps challenges.

The CloudBees Suite offers core and extended capabilities that enable organizations to scale CD and DevOps across all teams. Supporting a wide range of tools and technologies, components of the CloudBees Suite enable organizations to deliver business critical, modern and legacy apps. See Figure 4.

*Figure 4*

CloudBees Suite




Source: CloudBees, Inc.

CloudBees solutions meets compliance needs – integrating seamlessly with other solutions and enabling automation of compliance checks and enforcement. They enable quick and continuous monitoring by providing visualization of the software pipeline through the CloudBees Core™ team's user interface. This ensures that software is secure in two ways: through full end-to-end orchestration and automation of the software development life cycle with Jenkins Pipeline as code and integration into security scanning and testing solutions.

The CloudBees Suite meets the needs of enterprises embarking on DevSecOps journeys, as seen in Table 1.

*Table 1*

## Security Needs, Issues and Solutions

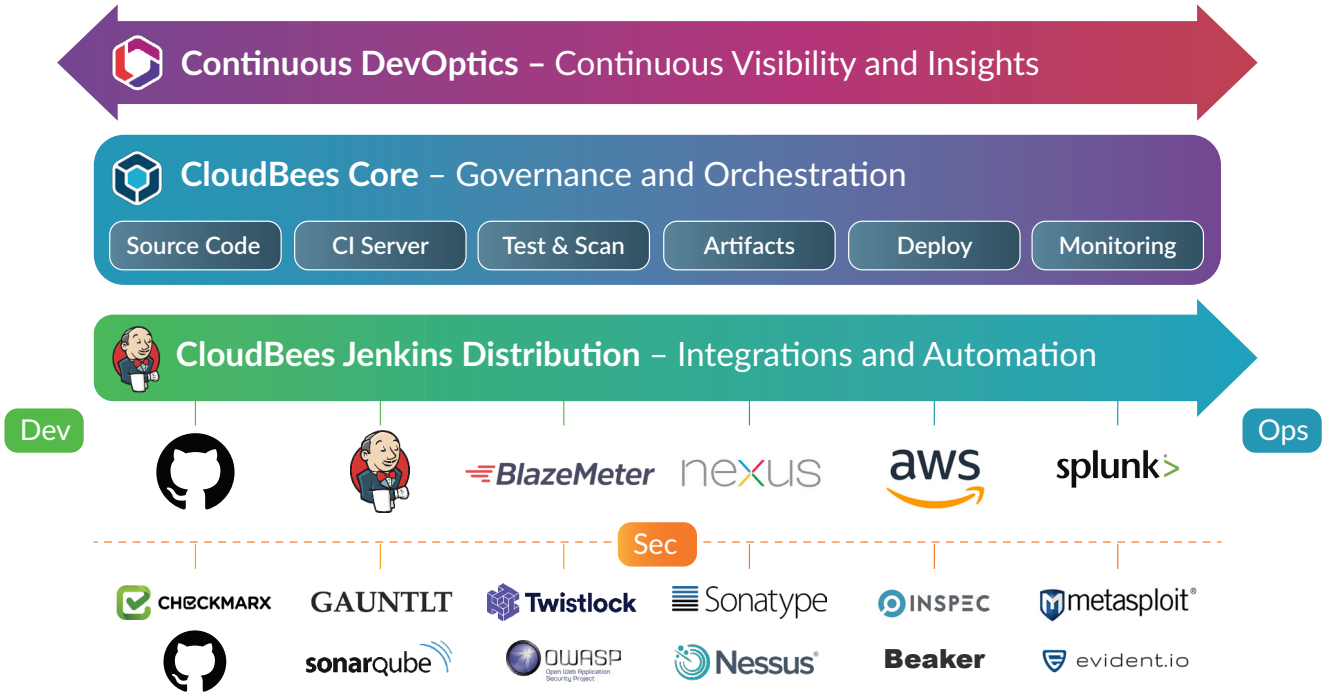
| ISSUE/NEEDS | CLOUDBEES SOLUTION |
|---|---|
| **Compliance with standards (NIST, ACAS, FedRamp, etc.)**<br>» Integrate tools to enforce compliance | Integrate with over 1,400+ plugins and is easily extended to integrate with additional solutions, enabling automation of compliance checks and enforcement.<br>Standardized, templated workflows enable compliance across the application portfolio |
| **Enable speed with change management (CM)**<br>» Provide confidence that checks have been completed<br>» Provide visibility into pipeline | Visualization of the software pipeline through CloudBees Core teams' UI enable quick and continuous monitoring.<br>Automatic and manual gates, ensure that insecure or non-compliant software does not move downstream.<br>All builds (job runs) are recorded and stored including logs and who/what originated the action. |
| **Ensure software is secure**<br>» Automate security checks<br>» Educate and enable developer to continuously test security | Full end-to-end orchestration and automation of the software development life cycle with Jenkins Pipeline as code and integration into security scanning and testing solutions.<br>Implement infrastructure as code for your environments and CD pipelines with integrations to CM solutions like Chef and Puppet, and deep support for containers and Kubernetes to support scalable immutable infrastructure. |

Source: CloudBees, Inc.

In addition, CloudBees brings all the elements together to form a fully functioning continuous delivery and security pipeline, as shown in Figure 5.

*Figure 5*

Example of Continuous Delivery + Security Pipeline




Source: CloudBees, Inc.

## Conclusion

DevSecOps won't be easy to implement. It will require a commitment to a new way of doing business. But it's necessary. Software is redefining today's business climate, and organizations that don't innovate quickly and securely will fall behind competitors.

To compete – and win – organizations need to adopt a flexible DevSecOps paradigm that will seamlessly embed security tests and compliance scans into their DevOps processes. And they need to detect and fix security flaws at a rapid pace, to keep up with DevOps. Security needs to be seen as being integral to the process, and not a drag on productivity. Done right, DevSecOps can deliver a sustainable competitive advantage.


## Learn More

**Download the eBook**
*Continuous Governance: The Guardrails for Continuous Everything*

**Watch the webinar**
*Orchestrating DevSecOps for Government: Automating Security as an On-Ramp to Making DevOps a Reality*

**Read the whitepaper**
*What is DevOps?*

CloudBees, Inc.
2001 Gateway Place, Suite 670W | San Jose, CA 95110 | United States
www.cloudbees.com | info@cloudbees.com



1218v00