Embedding Security into Cloud DevOps on AWS

Faster,
smarter,
and more secure

Checkmarx

aws partner network
Advanced Technology Partner
DevOps Competency

aws partner network
Advanced Technology Partner
Security Competency

aws partner network
Advanced Technology Partner
Public Sector Partner

Contents

A world that is now utterly dependent on code

Introduction

Faster, smarter, and more secure: these are the demands driving application development today as the cloud delivers exciting opportunities to create and deploy advanced software at a previously unimaginable pace. Free from the shackles of on-premises infrastructure, the potential is limitless, and providers like Amazon Web Services (AWS) are on hand to deliver an advanced development environment with unprecedented scalability and scope.

With this wealth of opportunity at their fingertips, organizations are under more pressure than ever to innovate. Fiercely competitive markets demand continuous improvement in dynamic, advanced applications that deliver new features, critical data insights, and exceptional user experience.

At the same time, in a world that is now utterly dependent on code, security risk has grown exponentially. Attack surfaces have expanded, and the criticality of applications, together with the data they handle, makes them a prime target for disruption, infiltration, and exploitation. Cybercriminals are deploying persistent, sophisticated attacks through multiple vectors in a bid to capitalize on software vulnerabilities.

Organizations need a streamlined approach that can scale with them

Change is Everywhere

In this fast-paced, constantly changing environment, agile and DevOps methodologies have gained ascendency as approaches capable of meeting the first two demands for faster, smarter software. However, the new pressures of cloud native application development mean that the third component—security—shifts to the cloud.

Responsibility for infrastructure security used to rest on the shoulders of the operations team, but now that the infrastructure itself is code, accountability has become blurred. Organizations are asking development teams to take on more responsibilities of software security. This additional workload potentially limits the rate at which organizations can develop secure software, contrary to the ambitions of agile, cloud-powered businesses on AWS.

To truly reap the benefits of AWS cloud native modern application development, organizations need a streamlined application security (AppSec) approach that can scale with them in the cloud environment.

This ebook will examine the challenges of embedding security into cloud DevOps on AWS and explore solutions that will deliver an environment where security across application and infrastructure scales at the pace of cloud application development.

The Cloud Application Development Shift

IDC estimates that by 2023, more than 500 million digital apps and services will be developed and deployed using cloud native approaches.[1]

Forrester predicts that at the end of 2021, 60% of companies will leverage containers on public cloud platforms and 25% of developers will leverage serverless.[2]

IaC errors are the second most common breach source, according to Verizon's 2020 Data Breach Investigations Report.[3]

1. "IDC FutureScape: Worldwide IT Industry 2020 Predictions," IDC, October 2019, https://www.idc.com/getdocjsp?containerId=US45599219.

2. "Predictions 2021," Forrester, October 2020, https://go.forrester.com/blogs/predictions-2021-cloud-computing-powers-pandemic-recovery.

3. "2020 Data Breach Investigations Report," Verizon, May 2020, https://enterprise.verizon.com/content/verizonenterprise/us/en/index/resources/reports/2020-data-breach-investigations-report.pdf.

Empowering developers to write more secure code

The Cloud DevOps Security Challenge

No developer sets out to write potentially vulnerable code, whether the development environment is on-premises or in the cloud. Nevertheless, as more and more organizations turn to the cloud to achieve scale, flexibility, and cost benefits, the elevated expectations around cloud native software development mean that even the most diligent developer might struggle to manage their security obligations. Ultimately, whatever environment developers are working in, they are only human and susceptible to making mistakes, especially when under pressure.

The consequence is that software vulnerabilities emerge, and malicious actors take the opportunity to weaponize them.

At Checkmarx, we empower developers to write more secure code with the right tactics, tools, and integrations that embed security throughout the software development life cycle (SDLC) on AWS. This way, teams can ensure that the applications they release are better protected from attacks.

Adopting true DevSecOps initiatives using the cloud requires organizations to think carefully about integrating and embedding security in the AWS-powered SDLC and, importantly, how they leverage automation to deliver secure code at the pace of cloud development demands. They also need to understand the key features that separate cloud native and on-premises security methodologies.

In the AWS shared responsibility model, AWS is responsible for securing the cloud infrastructure, and customers are responsible for securing their workloads deployed in AWS.

Who Is Responsible for Cloud Application Security? The AWS Shared Responsibility Model

In an on-premises environment, infrastructure, applications, and workload security is the domain of the operations team and the clear responsibility of the organization. However, when an organization uses AWS, security is managed on a shared responsibility basis.

Shared responsibility relieves the customers from operational burden as AWS operates, manages, and controls the components, from the host operating system and virtualization layer down to the physical security of the data centers in which the service operates. AWS cloud security gives customers access to a far higher level of built-in cybersecurity than they could achieve using their own resources and is one of the fundamental benefits of working with AWS.

AWS also supports cloud native application development and speeds up provisioning with templates and building blocks such as AWS CloudFormation and AWS Lambda, all of which have baked-in security that fulfills AWS's part of the bargain.

However, this does not mean customers abdicate responsibility for security–far from it. While AWS takes on responsibility for the security "of" the cloud, the customer retains responsibility for security "in" the cloud. In other words, data, hosts, containers, serverless, networks, users and their credentials, and resource configurations stay firmly within the customer's sphere of accountability. See figure 1 for more details about the AWS shared responsibility model.

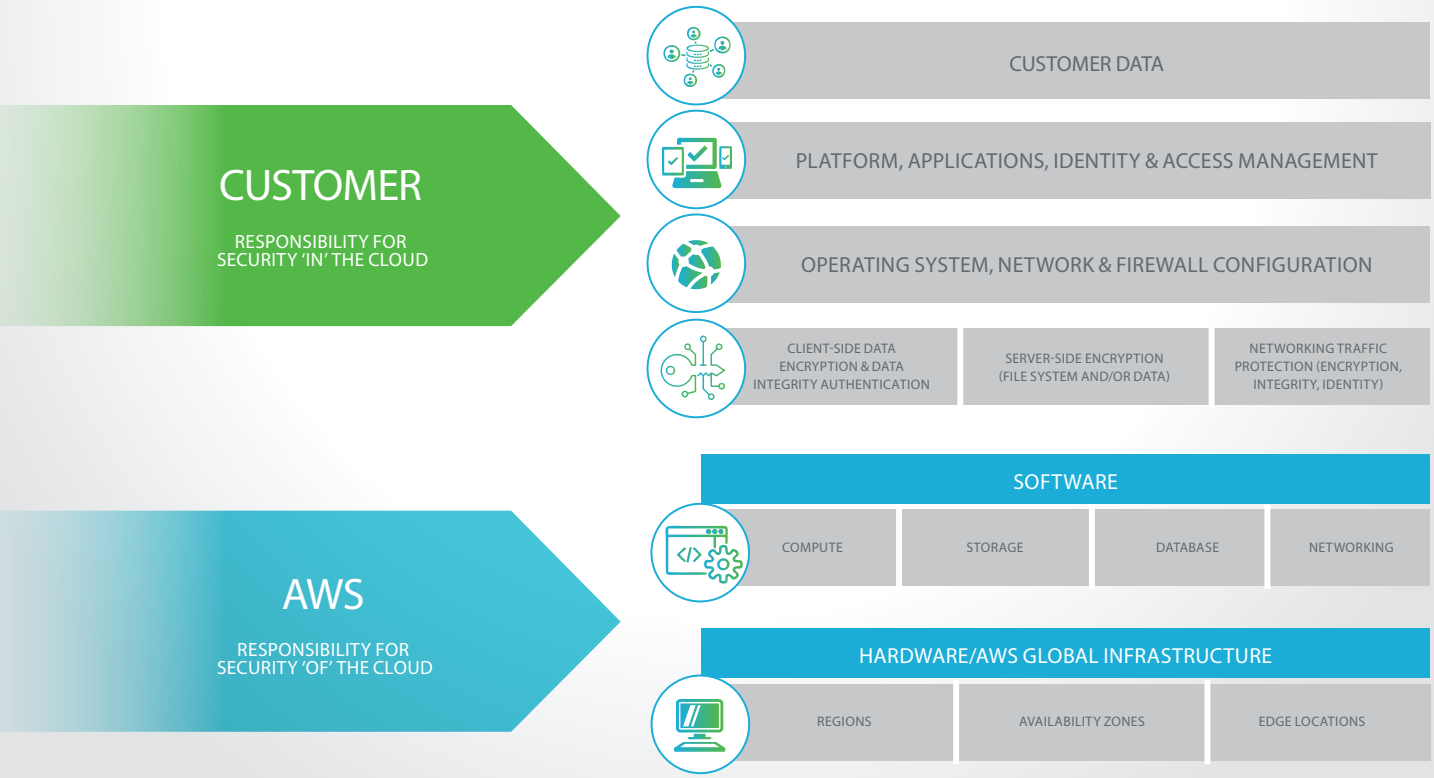


Figure 1: AWS shared responsibility model

> Given this expanded attack surface, companies must build security into the SDLC to have a strong security stance

What Else Is New with Cloud Native vs. On-Premises Development?

On top of the shared responsibility model, there are several other novel aspects when application development shifts from on-premises to cloud native.

First, there is *what* developers are building. Legacy applications incorporated binary code and scripts that were designed as single-tiered, monolithic applications. For cloud native applications, developers are building APIs, container images, serverless packages, infrastructure as code, and microservices.

In regard to *where* the runtime is hosted, customers formerly hosted the runtime applications on private or public cloud virtual machines or a physical server. Now applications leverage serverless, containers as a service, and container orchestration platforms.

The *how* is also rapidly evolving. Runtime and infrastructure deployment are also changing as installation scripts, ops teams, and existing infrastructure give way to infrastructure as code (IaC) orchestration tools.

Together, these changes make for dynamic applications that are scalable, flexible, easier to manage, and lower cost, but there are also new risks as the attack surface becomes broader and more nuanced. In traditional application security, teams might have focused on preventing cybercriminals from targeting databases, applications, and controlled environments. However, cloud native apps have many more building blocks interacting with each other, which bad actors can target.

Given this expanded attack surface, companies must build security into the SDLC to have a strong security stance—but at which stage?

AWS DEVELOPER TOOLS

To facilitate cloud native application development, AWS provides a powerful range of developer tools.

AWS CODEPIPELINE

Builds, tests, and deploys code every time there is a code change, enabling developers to rapidly and reliably deliver features and updates.

AWS CODEBUILD

Compiles source code, runs tests, and produces software packages that are ready to deploy.

AWS CODEDEPLOY

Automates code deployments to any instance, making it easier to rapidly release new features and avoid downtime.

AWS CODESTAR

Enables the fast development, building, and deployment of applications on AWS by providing a unified user interface, allowing easy management of software development activities in one place.

The Security Shift: Left, Right, or Center?

Since the linear approach of waterfall methodologies gave way to the continuously iterative principles of agile development and the emergence of DevOps, identifying the optimal point at which AppSec should sit in the SDLC has been under constant debate.

The DevOps movement established a culture wherein developing, testing, and delivering software was intended to take place quickly, regularly, and with more dependability. This cultural shift drove the adoption of continuous integration (CI) and continuous delivery (CD), which are fundamental components of DevOps.

DevOps is about processes, connections, automation, and tools throughout the development, test, and delivery stages. However, DevOps fundamentals have not fully addressed how to embed software security throughout the entire software development ecosystem. Security testing is still typically performed in the test/QA stage toward the end of the development cycle, at the point when development transitions to operations. Any vulnerabilities identified at this stage can still threaten to derail delivery schedules in the same way they did under the waterfall approach. As a result, tension remains. On one hand, there is a powerful commercial desire to deliver high-performing software quickly; on the other, it's imperative to guarantee that applications are secure in an escalating threat landscape with an evolving regulatory environment to match.

Security can become integral to the development process

In a bid to relieve this tension, the concept of "shifting left" has gained traction. The whole point of DevSecOps initiatives is to integrate and automate security testing earlier in the development pipeline as developers are being tasked with identifying and remediating vulnerabilities sooner—before they clash with delivery deadlines. However, there are limitations to this approach. Depending on the application security testing (AST) solutions used, some may not be able to scan code at the repository level, which is the farthest left security testing/scanning can shift.

The alternative, shifting right, can induce delays as organizations wait for full code scans to complete. Also, at this point, there is a greater likelihood that delivery deadlines will be impacted by the need to remediate a security problem just before the code is about to be deployed. Instead of shunting AST back and forth around the SDLC, the more innovative approach is to scan different types of code at various times, using integrated and automated tools that trigger scans based upon actions (e.g., pull requests) within developer workflows. This approach has been proven to speed up development since developers can quickly remediate errors in the branch of code they're currently working on.

Making AST intrinsic to the development process and testing throughout the SDLC creates an iterative cycle of continuous improvement that elevates the quality and security of code produced, even under time pressure. By shifting left and extending right (or "shifting center"), security can become integral to the development process and avoid being a brake on delivery speed.

However, cloud native development adds a new dimension to the challenges of AST: infrastructure as code.

> Developers and operators don't need to manually provision and manage servers

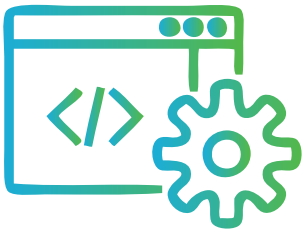
A New Risk: Infrastructure as Code

Infrastructure as code (IaC) makes the provisioning of infrastructure faster and more scalable for organizations operating in the cloud. It means developers or operations teams don't need to manually provision and manage servers, operating systems, storage, and other components each time they develop or deploy an application. It ensures the environment is stable and has become a critical DevOps practice that supports continuous delivery.

However, IaC brings its own security concerns. It can generate significant compliance and configuration risks, and these can weigh heavily on the shoulders of developers as they take on operational responsibilities.

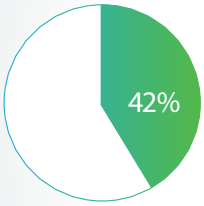
Previously, developers would build and test their application before passing it to operations to manually install, run, and monitor. Today, developers write infrastructure code in minutes and deploy an application with no other intervention, increasing speed but introducing new risks. When developers inadvertently make mistakes in configuring IaC, particularly when using templates, it can be catastrophic.

One example of IaC misconfiguration includes secrets storage (e.g., authentication tokens, SSH keys, passwords). If someone were to try to leave these in the IaC manually with the AWS console, AWS native security would flag it as contrary to best practice and put measures in place to try to prevent it. However, if a developer or cloud architect creates a valid but potentially vulnerable configuration in an AWS CloudFormation template, nothing stops that issue from making it into production. This reality poses immense compliance and security risk for organizations.



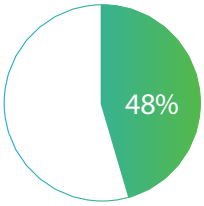
Cloud misconfigurations are a common problem

Research by Palo Alto Networks[4] found that:



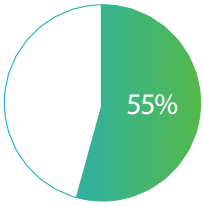
42%

of AWS CloudFormation templates contain at least one insecure configuration



48%

of Amazon S3 buckets don't have server-side encryption enabled



55%

of cloud user-configured S3 buckets don't have logging enabled

4. "The State of Cloud Native Security 2020" Palo Alto Networks, June 2020, https://www.paloaltonetworks.com/state-of-cloud-native-security.

Therefore, it is not surprising that Gartner predicts "through 2025, 99% of cloud security failures will be the customer's fault"[5]. The pressure is clearly on DevOps teams to close this gap urgently. However, this is a challenging task for traditional security testing solutions and will present one of the biggest obstacles to date in application security: making the connection between code, infrastructure, and configurations.

Despite its challenges, IaC, as part of the cloud native infrastructure, is an opportunity to finally have a single holistic platform that handles all layers similarly. Instead of having one dedicated tool for network security, another for the operating system, and yet another for application security, developers and organizations alike can now benefit from all these technologies being coded and scanned in a single streamlined process. Seizing this opportunity is crucial if organizations are to uphold their side of the shared responsibility model.

> "Technologies coded and scanned in a single, streamlined process

Keeping Infrastructure as Code Secure

To make this ambition reality, Checkmarx has launched KICS (Keeping Infrastructure as Code Secure), an open source solution for static analysis of IaC designed to help developers detect and fix configuration issues. KICS automatically detects vulnerabilities, hardcoded keys and passwords, compliance issues, and misconfigurations from the start of the IaC build cycle so developers can fix bugs before they reach production.

KICS supports all major IaC technologies, including AWS CloudFormation, allowing developers to check for vulnerabilities in IaC without resorting to manual, time-consuming reviews.

5. "Is the Cloud Secure?" Gartner, October 10, 2019, https://www.gartner.com/smarterwithgartner/is-the-cloud-secure.

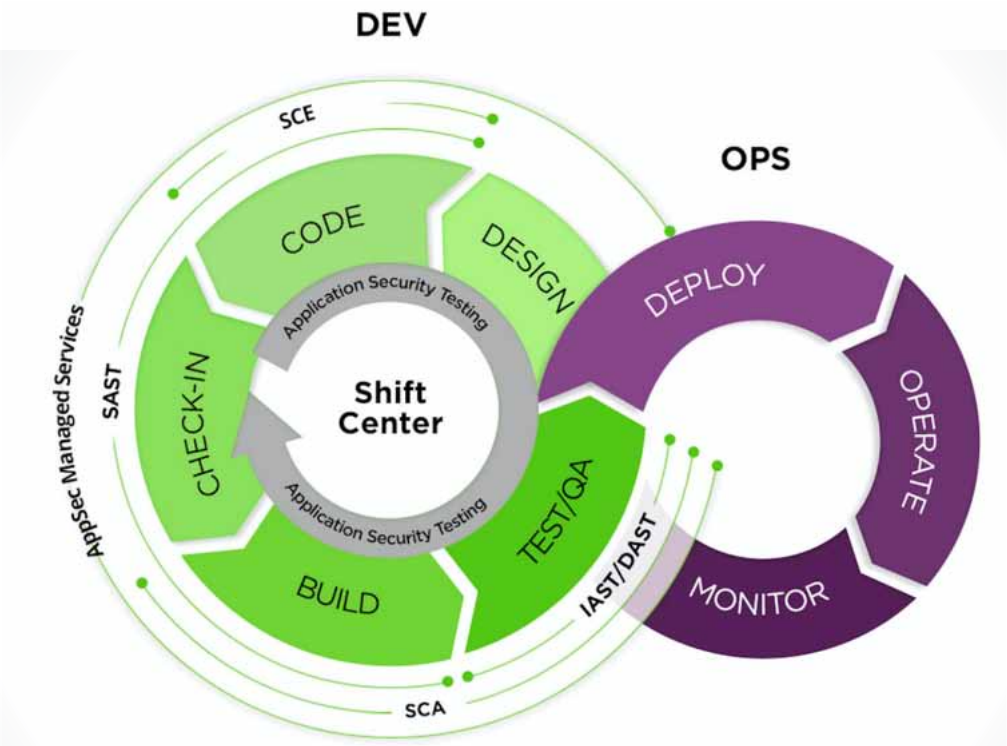


Figure 2: Points where AST solutions are often embedded in the different stages of development, extending into operations

Where AST Solutions Are Commonly Embedded in DevOps

Various facets of AST serve particular areas of the DevOps cycle:

- Static application security testing, such as CxSAST, incrementally scans at the source code level during the Code, Check-in, and Build stages for flaws such as SQL injection and cross-site scripting, and then provides guidance on where and how to fix source code vulnerabilities.
- Interactive application security testing, such as CxIAST, detects deployment configuration flaws in running applications found during functional testing in the Test/QA stage, before application deployment.
- Software composition analysis, such as CxSCA, triggered in the Build and Test/QA stages allows teams to detect vulnerable libraries or packages in order to address the vulnerability and license risks associated with open source software being pulled into a codebase.
- Secure coding education, such as CxCodebashing, is part of a comprehensive approach to secure software initiatives and should be available to developers during the Code stage.
- Infrastructure as code scanning, such as KICS (Keeping Infrastructure as Code Secure), should be run regularly from the start point within the integrated development environment (IDE) and at every pull request in the CI/CD pipeline to promptly identify misconfigurations and compliance issues. This ensures that exploitable application vulnerabilities and IaC issues never reach production and facilitates secure handoff to the AWS cloud.

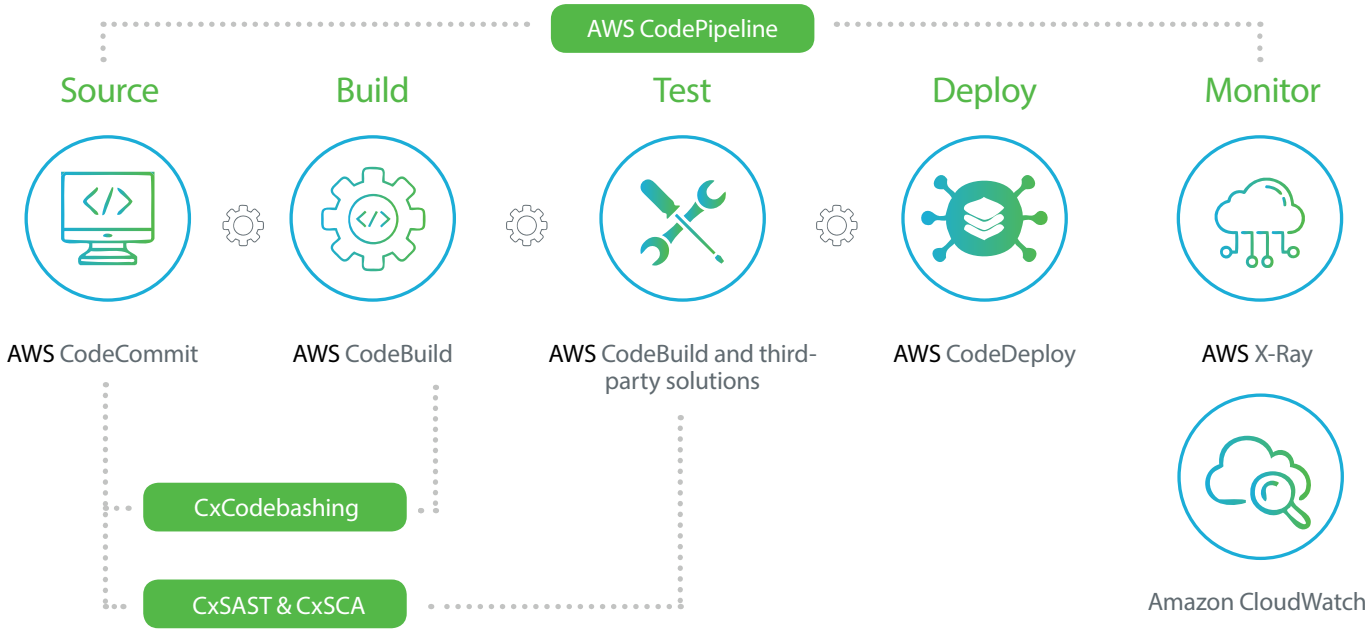


Figure 3: Integration between Checkmarx AST solutions and AWS

An AppSec Toolkit That Understands the AWS Cloud Environment

Although we discussed the differences between on-premises and cloud native software development, the fundamental security challenge remains the same. In a hostile cybersecurity environment, and facing increasing regulatory pressures, organizations' developer teams must create more secure applications regardless of where they code.

Developers need AST solutions that integrate with the coding journey, wherever it is taking place, to deliver fast and accurate scan results. A vital element of this is a management and orchestration layer that provides easy integration and automation capabilities with the development tools in use. A holistic picture of the software vulnerabilities across proprietary, third-party, and IaC code, through a single pane of glass, is highly desired.

Organizations using AWS can benefit from a comprehensive AppSec toolkit that understands the AWS environment and integrates with specific CI/CD pipeline stages. In figure 3, Checkmarx CxSAST, CxSCA, and CxCodebashing are highlighted to demonstrate where they integrate well within AWS CodePipeline.

In addition, Checkmarx is an AWS Security Competency and DevOps Competency partner with demonstrated expertise in AWS. Checkmarx provides integrated support for AWS CodeStar services, allowing customers to initiate automated AST scans from AWS CodeBuild and AWS CodePipeline for code stored in AWS CodeCommit. In contrast, KICS supports AWS CloudFormation to complete the circle and verify IaC with the same rigor.

In a world where code is everywhere, and everything is code, security should be everywhere, too.

Budgeting Benefit

An additional benefit for AWS Marketplace customers choosing Checkmarx is the ability to retire their Enterprise Discount Program commitment against their Checkmarx investment, making implementing a comprehensive cloud native AppSec program even more cost-effective.

Empowering Developers in the Cloud—Enabling the Human Layer

With all the speed, power, scalability, and flexibility delivered by the cloud, the human layer is the final limiting factor in achieving cloud DevSecOps. To truly embed security into the development cycle in the way described, developers need tools that empower rather than obstruct their work. We've already underlined the fact that no developer wants to write potentially vulnerable code. Still, the pressure to deliver can mean they can't always resolve every issue, resulting in less-than-optimum code security and the introduction of avoidable risk.

Software-driven organizations have grown accustomed to unrelentingly high-speed development and release frequency. Nonetheless, they can't afford to compromise on security either, as the proliferation of code delivers more opportunities into the hands of malicious actors.

At the same time, developers themselves are in short supply, and AppSec competency is even harder to find. Hence, developer teams are typically under-resourced, leading to developer burnout and employee churn, a genuine concern for organizations that depend on rapid deployment schedules to remain competitive and adequately serve their clientele.

Consequently, any tools designed to help these teams scale their security efforts to match delivery demands must offer as much automation and integration as possible to mitigate the increased security burden developers face.

Building in-house developer security skills and awareness is another foundational tactic to putting the "Sec" in DevSecOps. However, a lack of time and resources can make formal education and training programs impractical. A better option is for organizations to diversify the skillset of their in-house developers, making them better coders by adopting an application security solution that both informs developers of code vulnerabilities and provides guidance on the best way to fix the issue.

Checkmarx designed the Codebashing secure code training platform to inject AppSec awareness across the SDLC through just-in-time, bite-sized lessons that relate directly to the issues developers face in their code. The learning-while-doing approach rapidly accelerates developers' AppSec understanding and encourages a security mindset by helping them solve live problems.

Altogether, this approach of automation and integration combined with timely on-the-spot education allows developers to own software security and functionality. Agile and cloud native modern application development is all about this fully integrated security program to solve the last limitation.

> A lack of time and resources can make formal education and training programs impractical

Scaling In-House Capabilities with AppSec Accelerator

Many businesses have the ambition but not the immediate resources to scale up AppSec in their AWS-powered software development programs. Fast-moving organizations have rapid release cycles and large developer teams, but only relatively small security teams. The skills squeeze means it can be difficult to hire in-house support, even if budget is available.

Checkmarx AppSec Accelerator™ is a managed service which includes CxSAST, CxSCA, CxIAST, and KICS, together with expert consultancy to manage triage, remediation, guidance, uptime, integration, automation, and scalability.

Find out more…

DevSecOps Is the Future of Modern App Development

Cloud DevSecOps is the future of modern application development. But while many organizations are successfully scaling the speed and functionality of their application delivery cycle, too often, security can't keep pace. Businesses need a solution that leverages automation and intelligent orchestration to integrate seamlessly with the cloud-based DevOps environments to scale application security in the cloud. The right solution empowers developers to own application security and address the new challenges and responsibilities introduced by infrastructure as code.

8 Tips for Integrating Security into Cloud DevOps on AWS

Look for an AppSec solution provider that:

- Has approved relationships, competencies, and certifications with AWS.
- Prioritizes accuracy and relevance in scan results to avoid creating alert fatigue and devaluing your investment in AppSec. Make sure the tools you choose empower your developers and don't obstruct them.
- Provides a full suite of AST solutions with low TCO: the ability to automate scans and orchestrate results is foundational to DevSecOps success. Make sure your approach is unified rather than based on multiple point solutions.
- Includes an integrated developer secure coding education and awareness platform to help broaden your developer skill base, improve productivity, and enhance employee satisfaction.
- Supports static code analysis, software composition analysis, and interactive (runtime) code analysis across IaC, APIs, and conventional source code.
  The provider should also supply remediation guidance and best-fix location to speed vulnerability triage.
- Integrates into the IDEs, code repositories, CI/CD pipelines, and feedback channels developers prefer to use.
- Supports all major development languages and frameworks.

AWS and Checkmarx: Better Together

Checkmarx

Checkmarx is constantly pushing the boundaries of Application Security Testing to make security seamless and simple for the world's developers while giving CISOs the confidence and control they need. As the AppSec testing leader, we provide the industry's most comprehensive solutions, giving development and security teams unparalleled accuracy, coverage, visibility, and guidance to reduce risk across all components of modern software – including proprietary code, open source, APIs, and Infrastructure as Code. Over 1,600 customers, including half of the Fortune 50, trust our security technology, expert research, and global services to securely optimize development, at both speed and scale. For more information, visit our website, check out our blog, or follow us on LinkedIn.

www.checkmarx.com

aws partner network Advanced Technology Partner DevOps Competency

aws partner network Advanced Technology Partner Security Competency

aws partner network Advanced Technology Partner Public Sector Partner