ESG SHOWCASE

Developers: Own Your Security Destiny

Date: December 2021 **Author:** Melinda Marks, Senior Analyst

ABSTRACT: World class product development teams build security into development processes to deliver high quality products with consistent uptime and no major outages. Modern software development leveraging cloud services brings greater scale and speed with continuous releases and updates but opens the door for failure with a higher chance of security misconfigurations. Waiting for the security team to check code or insert processes isn't an option because it works against developer workflows, slowing the process down.

Instead, it's time for developers to own their destiny by securing their development practices with the right security tools built into their workflows, reducing the need to interact with security teams. It's not about developers having to learn about security or do more work. It's about using the right tools for shorter feedback loops on security coding issues, fostering a culture of secure development, and reducing work across teams. With this modern approach, developers can take ownership to deliver code that meets high standards for functionality, quality, performance, and security.

Working Security Into Development Processes

Businesses have been modernizing software development leveraging cloud services to speed up software delivery. Instead of waiting for other teams, software developers are empowered to provision their own infrastructure and applications. They also use continuous integration/continuous deployment (CI/CD) pipelines and processes to enable team collaboration for rapid, continuous releases and updates.

As teams grow, there is a higher chance for mistakes. Even simple mistakes can leave applications vulnerable to attack, exposing valuable company or customer data or bringing systems down. But when developers have to interact with security teams for security scans or for help setting things up, like configuration settings or checking for vulnerabilities, they are slowed down.

Developers often default to a "looks good to me (LGTM)" approach instead of testing their code for security or reliability issues. Slowing down for security processes, using a separate security tool or dashboard, and filing a ticket with the security team are too disruptive to the velocity of modern software development.

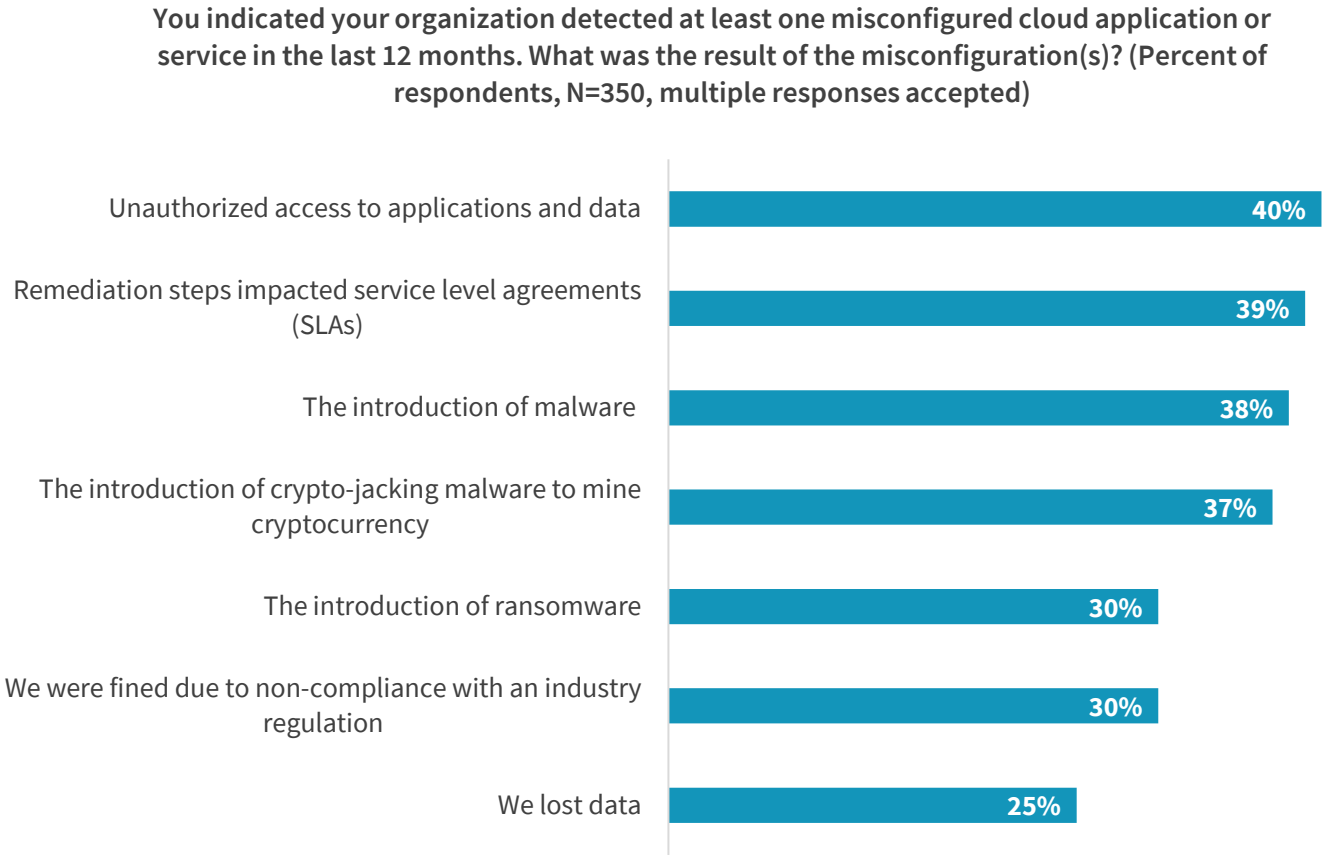
It's better to implement security tools that do the work for developers, without requiring help from other teams. When security tools run continuously and automatically throughout the software development lifecycle, developers can efficiently find and fix coding issues in their workflows without context switching. This provides the most efficient way to ensure that code is tested, secure, and reliable at every development phase.

Automate Security Testing in Development

It can be tempting for developers to skip security processes if it takes too much time or adds useless work. After all, development teams want to spend their time building and releasing application code. But most security issues are

preventable mistakes, such as misconfigurations that could expose an open database. ESG research shows that businesses face significant consequences from misconfigurations.[1]

Figure 1. Misconfigured Cloud Services Have Significant Consequences



Source: Enterprise Strategy Group

There are many security testing tools available, including free and/or open-source tools. Using these types of tools in development are a good starting point to identify problems. It is also a good practice for developers to introduce this step of security testing early in development processes, just as you would do unit testing.

But these testing tools vary in capabilities and setup requirements. Enterprise solutions are better designed to scale across the entire organization, standardizing processes across teams. These solutions can non-disruptively test code across an inventory of assets and test any code changes. They can also provide assistive guidance on how to fix the identified issues within your integrated development environment (IDE).

This automates security testing throughout the software development lifecycle (SDLC) so developers can fix issues efficiently without having the security team intervene to identify and fix security bugs.

The Importance of Accuracy

Many solutions promise alerts to developers so they can fix security issues but an overabundance of alerts can slow development down. It's important to look for an accurate solution that will only deliver the alerts that matter:

[1] Source: ESG Research Report, *The Maturation of Cloud-native Security*, May 2021.

- Better quality tools have more information on applications and context to only surface real security issues that need to be fixed.
- Getting too many alerts or alerts that don't matter can create useless work.
- Solutions that lack context on application usage, behavior, or networking patterns result in erroneous findings.

Surfacing Real Coding Issues

Fix security issues in your workflow without chasing false positives.

- Look for a solution that uses contextual information across the SDLC, including mapping application data flows, such as databases, webservers, and libraries, to assess exploitability so it can more accurately surface issues that need to be fixed.

Efficient Remediation and Code Delivery Instead of Painful Rework

With the right security solutions in place, developers can efficiently address actual security issues in their workflows at the right time, with shorter feedback loops. The key is to deescalate security issues and align them with other code defects so they can be addressed like any other bug or code defect at any point in the SDLC.

When security becomes an attribute of software quality instead of a competing process, developers can focus on the rapid, agile delivery of features without worrying about out-of-band security processes slowing them down. For example, without the right tools in place, if a preventable coding mistake is deployed in runtime, by the time the issue is discovered and triaged by the security team, it becomes a project to fix.

Instead, when organizations are able to build security processes into the full SDLC, fewer defects will materialize in production, and then, if issues do come up in production, developers can efficiently get the information they need to quickly fix any problems.

Let Security Pave the Road

Building security into the development process changes the role of an organization's security team. Instead of having to find issues, the security team's role moves to laying the groundwork for developers to secure their own code.

Security can set up the automated security processes throughout the SDLC, staying focused on deep alignment within each stage of design, build, test, and deploy. This includes codifying security rules as guardrails in the development process to prevent vulnerabilities—for example, setting policies to ensure that databases are not exposed or that S3 buckets aren't open. These policies need to be available in an organization's IDE and integrated in their CI/CD pipeline so that it is codified in development processes.

Developers gain a line of sight into the real code issues that exist along with assistive tools to help them quickly and easily resolve issues without friction at any point in the software lifecycle.

Introducing Contrast Security

Contrast Security provides a modernized approach to application security that works across the SDLC to empower developers to efficiently secure their code. It provides:

- Automated code testing and analysis embedded in SDLC processes, including automated software composition analysis (SCA), pipeline-native static application security testing (SAST), and continuous vulnerability assessments, for near real-time vulnerability remediation.
- Near instant feedback on developer code, providing developers with full context about each vulnerability, user input, line of code, query, library, etc., with clear how-to-fix guidance assistance delivered directly in developer workflows.
- Full scale integration across the entire SDLC from code to build to test to production. Backed by an industry-leading secure coding platform, Contrast Security embeds security into every code commit, pull request (PR), and functional test before pushing code to production.

Instrumentation for Accuracy

Contrast Security's platform embeds security throughout the SDLC by using instrumentation with sensors embedded in the application code. The sensors continuously monitor the applications with visibility into custom and third-party libraries, including transitive dependencies.

Contrast removes much of the tedious overhead and re-work that comes with sifting through false positives. It highlights your true application attack surface by showcasing the third-party libraries that are actually used during runtime and validates vulnerability findings and fixes in custom code with real-time route intelligence.

By emulating the runtime behavior of the application, the Contrast Platform can determine the vulnerabilities that can actually be exploited, reducing false positives and noise, so developers can focus on vulnerabilities that can be exploited, instead of chasing alerts for potential threats. The platform applies this same methodology towards emerging cloud-native environments to monitor for vulnerabilities in serverless applications with the ability to block threats in production.

The Bigger Truth

Modernized application security solutions give us the ability to build the right security processes and tools into development as an attribute of building quality applications instead of using bolted-on, misaligned security approaches that add friction to agile development.

With the right processes and tools in place, including policy control, automated testing, and shortened feedback loops to efficiently remediate security issues, developers gain the confidence of shipping secure, tested code and the ability to rapidly remediate issues if they are found. Solutions such as the Contrast Security Platform give developers the tools and assistance they need to build security into their own workflows, enabling them to efficiently deliver products meeting the highest standards of security, performance, and reliability.

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

www.esg-global.com contact@esg-global.com 508.482.0188