RAPID7 | insightAppSec

WHITEPAPER

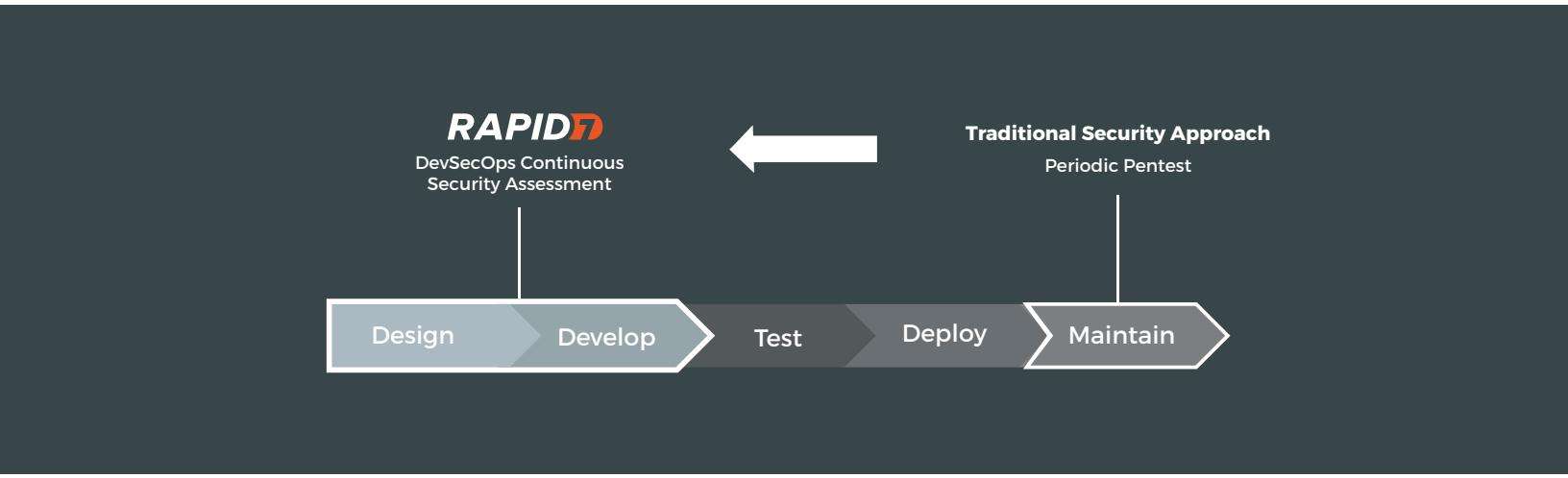
A Step-by-Step Guide to Shifting Left and Embracing a True DevSecOps Mentality

TABLE OF CONTENTS

A Major Shift is Underway

There is a major shift happening right now. It's not just affecting security teams, but IT operations and development teams, too. And it's something no company can afford to ignore today. It's the shift from web application security being the job of the security team to it being a shared initiative across many teams.

Responsible for the creation, deployment, and uptime of a company's products and services, development and operations teams are looking for ways to accelerate the software development lifecycle (SDLC) to stay competitive by releasing features faster. With changes being made so quickly, however, it's often difficult for security teams to review them before they get pushed to production, where they're more challenging to fix. The solution? Shifting the responsibility of security left, or much earlier in the SDLC. This can be done by bringing development and operations teams into the security equation so that as the development environment scales, security testing can, too.



Although it's a similar process to implementing a DevOps (or agile) culture, the path to DevSecOps isn't always so clear. In fact, bridging the gap between development, operations, and security is the ***number one challenge*** in application security today, according to a SANS report. Perhaps that's why only 24 percent of companies conduct application security testing just once a year, but considering that public-facing web applications have become the biggest source of breaches, *now* is the time for all companies to embrace the shift left.

In this whitepaper, we will walk you through the process to make this shift possible so that you can soon run fast *and* secure. Let's get started.

From DevOps to DevSecOps: Why Now is the Time to Shift Left

Similar to how DevOps enables companies to move fast, DevSecOps enables companies to move fast and secure. A simple concept at first glance, it's one that requires companies to rethink the way they approach and integrate security with the rest of the company. Considering just how important security is today, however, the earlier a company can adopt DevSecOps, the better off they will be. Especially considering that over 20 percent of companies don't know if they experienced a breach where an application was the source, now is the time to close the gap and gain full visibility. This can be accomplished with DevSecOps.

Whereas traditionally development teams would build a product or feature and then hand it over to the security team to ensure it meets certain requirements, today, the wall that separates these teams is being broken down. Not only is the elimination of these silos accelerating the SDLC, it's also removing security as a barrier to speed and progress—and this is a good thing for everyone involved.

To make this change possible, companies are bringing in dynamic application security testing (or DAST) to scan for issues more often and to better integrate security into their tools so that bugs and vulnerabilities can be spotted ***as the product is being built***. Failing fast and early is a good thing in application security, and shifting security left can make this happen. In practice, fixing security issues along the way is far less time intensive and expensive as they can be rapidly addressed. Not only that, but when security becomes a part of the normal development workflow, it can be built into the product from the very beginning, not treated as an afterthought.

More secure products equal more secure businesses, and that's a good thing for your bottom line, your partners and customers, and your reputation.

Implemented correctly, DAST can actually speed up development time, as it enables developers to address vulnerabilities at the same time they're fixing bugs, a rapidly iterative process that keeps the SDLC rolling along and without any surprises at the end. Designed to integrate with existing tools, scan for issues at every step in the SDLC, and enable seamless workflows between development, operations, and security teams, DAST is the way forward for companies looking to shift left. In this way, security no longer becomes an obstacle, but an enabler.

So how do you begin the process of shifting left? Let's show you…

A Step-by-Step Guide to Shifting Left

If accelerating the SDLC process and building security into your product or service sounds like the perfect remedy to your security challenges, below are the four steps to take to shift security left and adopt DevSecOps.

1. Create Common Goals

Building a Cross-Organizational Partnership

As with anything, team leaders need to be bought in and on board—before any tool or process is introduced—with this change in order for it to be effective. The best way to begin is to find common ground with other teams. This can be done by explaining that the goal you're looking to accomplish is to better secure what they're building—and in a way that won't slow them down, but perhaps even speed them up. A mutually beneficial goal, it won't be long before your colleagues start to see why this is the right move for the organization.

It's also important at this phase to talk about the real-world impacts and risks, both of continuing on the current path and adopting DevSecOps. It may help to whiteboard the company's existing challenges and goals with regards to web application security, outline how DevSecOps can help, and weigh the benefits and risks of taking this new approach.

Shifting left enables companies to rearrange processes in a way that benefited the entire organization. Helping your colleagues understand how it can be done so you can effectively roll it out is key to getting buy-in so that the shift can begin.

2. Integrate Tools

Embed Security with Existing Tools

Naturally, your team will want to know how this can happen. Explain to them that this can be accomplished by adding a layer of automation to the tools they are already using so that web application security testing can become a natural part of the SDLC. Integrated into your team's daily workflow, security can be built into the product from the start, making it easier to address along the way and no longer a roadblock at the very end. Done right, it won't even require your team to learn a new tool or add yet more work to their plate—it can be a seamless process designed to save both time and effort.

Dynamic application security testing is the most common method of scanning web applications in their running state for potential security vulnerabilities. DAST solutions, when integrated into the software build pipeline, will scan applications for security defects at regular, automated, intervals during development in order to catch issues early and often. DAST tools like Rapid7 InsightAppSec can be easily embedded into your existing development workflow and play nicely with existing development tools, which can go a long way towards ensuring a seamless rollout of DevSecOps.

The following tools are the ones your DAST solution should be able to connect with:

Continuous Integration Tools

If your company has a DevOps team, chances are they're using a continuous integration (CI) tool, whether it be Hudson, Jenkins, or the like. Using a CI tool, code can be pushed to production on a regular basis, but it should also be tested for security issues at every step of the development process. Doing so manually can be time-consuming and error-prone, especially if code is pushed to production multiple times per day. Instead, you can integrate a DAST solution with your continuous integration tool to automatically scan nightly builds of the application, enabling you to spot issues and fix them much earlier in the process.

InsightAppSec is a DAST solution designed for companies who are shifting left, helping companies scan applications as part of a dynamic testing schedule. With a well documented, open API, InsightAppSec can connect to virtually any continuous integration solution, as well as any other application, and isn't limited by the quantity or type of integrations. This is especially key for organizations who manage a large number of development and operations tools and have a fast-growing development environment.

Test Automation Tools

Often used by development and QA teams, tools like Selenium are designed to automate web application functionality testing, but they also enable teams to bring security into the mix. By layering on a DAST solution like InsightAppSec with Selenium, for example, you can repurpose unit tests from your QA team.

This means that not only can a Selenium script be used to test for functional issues within the web application, it can also be used to check for potential security vulnerabilities and bugs.

Issue Tracking Tools

Once real security bugs or vulnerabilities are detected, they need to be sent to your development team and prioritized alongside other issues. The best way to do this is by integrating your DAST with the issue tracking tool(s) your QA and development team already uses, such as Jira. This way, the moment an issue is detected in the code, it can be sent directly to Jira, giving it the same level of visibility as any other issue that needs to be fixed.

A DAST solution like InsightAppSec can power this entire process start to finish by feeding issues found in Jenkins or Hudson into Jira so that your team doesn't have to oversee the handoff or risk an issue falling through the cracks. In this way, security becomes a natural part of the SDLC and is easy to fold into the mix with the rest of your processes.

3. Streamline Processes

Automate Functional Tests

Shifting security assessment left in the SDLC requires collaboration between security, IT operations, and development teams facilitated with an automated end-to-end testing workflow. By integrating a DAST solution like InsightAppSec with your continuous integration, testing, and reporting solutions, you can begin automatically scanning web applications in their running state to find vulnerabilities that require remediation.

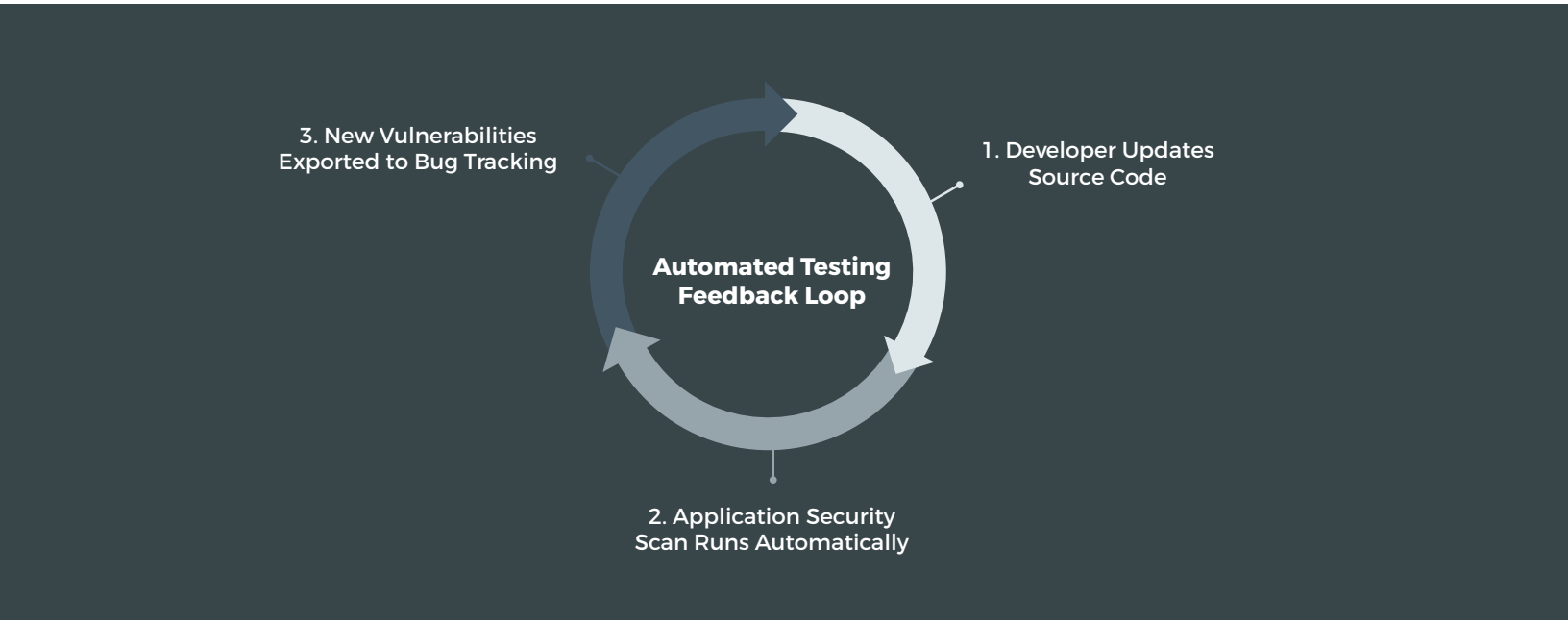
In practice, once the development team commits source code changes, Jenkins will compile code and run functional tests, Selenium

Building Bridges with InsightAppSec and Jenkins

The InsightAppSec plugin for Jenkins enables you to set up customizable pass/fail build conditions in your developers' existing CI/CD workflow. By sharing these tools and processes with your DevOps teams, you can shift security left in the SDLC and deliver more secure experiences for users.

To download the plugin or to learn more, visit the Jenkins plugin site.

will check for web browser functionality, and your DAST solution will run a dynamic scan to detect new vulnerabilities that may have been introduced. Then, when issues are detected at any point along the way, your DAST solution should send them immediately to Jira to be flagged for remediation. Done right, all of this will happen in parallel to other development work and run continuously in the background with no impact on performance or speed.



Solutions like InsightAppSec are specifically built to integrate with today's most common continuous development, integration, and ticketing solutions so that vulnerabilities are detected, reported, and prioritized in the same workflow and view your team is already accustomed to. In this way, security visibility can be rolled out organization-wide without any interruption of service, and security can become a natural part of the development process.

4. Real-Time Feedback

Bi-Directional Communication Between Dev, Sec, and Ops Teams

Traditionally, development, security, and operations teams worked independently, only coming together at the end of the SDLC for testing purposes. However, this void in communication is the opposite of collaboration, and can come to a head when issues are found at the very end of the process. True DevSecOps brings teams together to communicate about issues or challenges early and often so that the entire process can run smoothly. With a DAST solution like InsightAppSec serving as the connective layer between development, operations, and security tools, teams gain better visibility into anomalous or risky activities and can proactively work together to find a solution.

InsightAppSec can also help you put in place what are called quality gates, or policies that can halt a build or raise an alert when a threshold of vulnerabilities has been detected by a DAST scan. This gives the development team not just an early warning, but also real motivation to fix the issue so that the integration and testing process completes without an error. This is another key way companies can shift left and build more secure products.

Shifting Left: The Bottom Line

So, you may now be wondering how, if at all, shifting left will impact the bottom line. Shifting left is not only appealing to companies who want to run fast and secure, but also to those wanted to cut down on expenses. When security issues are caught earlier in the development cycle, they are far cheaper to fix. Think about it: if a web application is deployed and then a serious bug is found, the entire application needs to be taken down for days, if not an entire week, to remediate the issue. After that, it may take another week to re-deploy it. The result? Two whole weeks of downtime, fourteen days of salaries spent on bug fixes (not feature development), and measurable revenue loss (Gartner cites up to $5,600 per minute!). Alongside this, you may lose customer trust, cause partners to question the reliability of the service, and create a headache for the support team who is having to deal with all of this.

All of this can be avoided if testing can be done before the initial deploy. When issues are addressed along the way and before code is pushed to production, downtime becomes a thing of the past and these smaller issues can be addressed quicker. When it comes time to deploy, security has already been addressed and the application can be brought live without any hiccups or surprises. This means you can onboard new customers sooner and eliminate any tail-between-the-legs rollback of the app to fix a security issue that was caught after-the-fact.

Put simply, a secure product instills more trust with customers, costs you less money, and accelerates the entire development process. Being able to release applications and new features quickly and securely can also be a major competitive advantage, which can have a tremendous impact on the bottom line.

Automation Enables the Big Shift: A Case Study

Microsoft is one such company that saw the benefit of and adopted the "shift left" early on. In fact, they were one of the early pioneers of integrated web application security testing when they introduced the Secure Development Lifecycle (SDL), which was one of the first well-known models representing this shift.

Following the SDL, security was baked into their software development life cycle, not bolted on at the end. When Microsoft first took on this initiative, they developed a homegrown solution, but as the technology and threat landscape continued to proliferate, they knew it wouldn't be able to keep pace with modern applications with rich, dynamic clients and numerous APIs on the back-end. So, they began to look for a solution that was already built to meet the dynamic and growing needs of enterprises. After months of testing and extensive evaluation, Microsoft chose Rapid7 application security solutions to help them accomplish this.

Rapid7 is designed to handle complex application ecosystems like Microsoft's that have rich clients and RESTful APIs. This in and of itself met a broad range of their requirements, but above all, Microsoft required a DAST that could be flexible and extensible, able to scan for security issues in every part of their product and grow with them as they continued to expand their product portfolio.

Another important element they needed was the ability to develop custom attacks on their own, via an API. With so many endpoints and a diverse range of products, the company needed to be ready for anything, and Rapid7's robust and extensible API allowed them to do exactly this. Equally important was being able to handle complex authentication schemes with high accuracy, thus reducing false positive rates. Working at such a large scale, the team doesn't have time to sift through non-issues, and when testing Rapid7, they saw how the tool was able to analyze complex events and raise to the surface only the ones that were true issues.

With all of this taken into account, Microsoft knew Rapid7 would be the perfect fit for their large-scale application security needs, able to handle their requirements both now and down the road.

Ready to Shift Left?

Shifting security testing left in the SDLC and embracing DevSecOps is not only an operational change, but a cultural one as well. As organizations continue to expand their development environments, they also need to fundamentally change how they think about and approach application security. Faster product delivery is great, but it can't be at the sake of security. By sparking the conversation now and bringing DevOps and security closer together, companies of all sizes can better manage risk, gain security visibility, and create more secure products and services. While it does push the security responsibility, in large part, onto DevOps teams, automation can ensure that this shift isn't a burden or a nuisance, but instead a motivator to build more secure products and a bridge that brings together two traditionally separate teams.

By approaching security iteratively, incrementally, and rapidly, companies can become smarter about security, test early and often, and solve security problems together.

A natural extension of DevOps, DevSecOps stands to be the next big culture shift. Using the framework in this whitepaper, we hope you can begin the conversation sooner and roll out an effective and transparent web application security program that brings your teams closer together.

To bridge the gap between your teams and tools, try InsightAppSec for free today!

www.rapid7.com/insightappsec