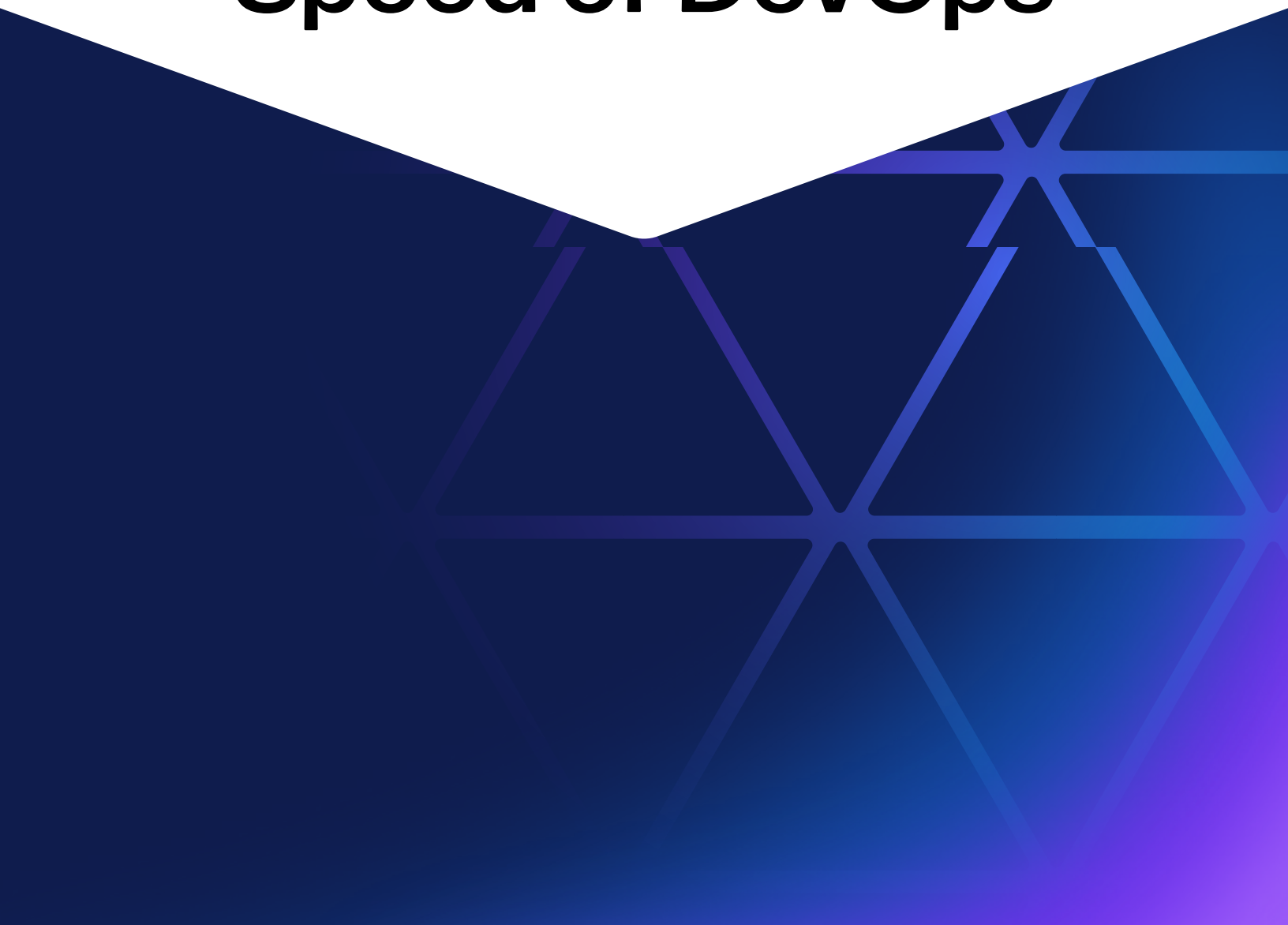
sonatype

Four Strategies for Securing Federal Applications at the Speed of DevOps



Introduction

The Need for (Secure) Speed

Federal CIO Tony Scott once said that his "measure of success is speed to market. In today's world, speed means everything."

Indeed, federal agencies are trying to move faster through the use of DevOps and open source software. Both can help agencies achieve greater levels of development velocity by encouraging collaboration and freeing developers from the need to build custom code, which can prove time-consuming and costly.

However, while open source and third-party components can help facilitate innovation and efficiency, they can also introduce significant risks. As referenced in Sonatype's 2020 State of the Software Supply Chain Report, 1 in 10 components used by development organizations to build applications contain at least one known security vulnerability. Worse yet, faulty components are increasingly becoming the preferred attack surface in today's applications. Known vulnerabilities in open source components led to the Apache Commons Collections Java library deserialization of 2015 (identified at the root of several high-profile ransomware attacks) and the notorious "Heartbleed" virus — two examples of bad actors using open source code vulnerabilities for malicious intent.

It doesn't help that security is often seen as a hindrance to DevOps agility. Teams that are focused on developing applications at a furious pace do not want to be encumbered by bolt-on security procedures at the end of a development lifecycle. They want to be creators, not gatekeepers. While guidelines like the Risk Management Framework (RMF) are ideal for laying out steps and processes necessary to integrate better security procedures, they must be complemented by automation, which allows federal agencies to establish a solid security posture without compromising their need for speed.

In short, agencies need security protocols that can keep pace with development practices — without holding them back.

Software supply chain automation (SSCA) addresses this need. It supports DevOps initiatives and accelerates software innovation by allowing agencies to ensure that the third party and open source components they are using are secure and reliable.

In this white paper, we'll take a look at how SSCA can help agencies achieve greater agility through DevOps while ensuring the code they're using is free of vulnerabilities. It will address:

- The benefits and risks of DevOps and use of open source and third-party components
- Mitigating those risks through control over software supply chains and comprehensive vulnerability scanning

- Applying and automating software supply chain guidelines and practices into the RMF process to advance DevOps and circumvent security roadblocks
- Setting up a repository of secure and reliable components that can safely be used and reused
- Empowering developers so that they are the first line of defense
- The need for ongoing risk management
- All of these are important considerations for federal agencies that want to accelerate innovation and application development without sacrificing quality, reliability, and security.

DevOps and Open Source: Benefit and Risk

DevOps and other agile methodologies have quickly begun to replace waterfall development and delivery approaches as agencies strive for more rapid application development. DevOps and similar approaches reduce the cycle time between the inception of an idea and delivery of that idea in reliable software.

As part of this effort, use of open source components — which can be easily procured, quickly deployed, and work in any environment — has become the norm. Federal software development organizations like U.S. Digital Services and the 18F group at GSA are encouraging agencies to adopt more open source components in their development activities (even to save as few as 20 lines of custom code) to speed up their processes and rollouts.

Today 90% of a typical application is comprised of open source code, and research shows that a single agency development organization may consume 200,000 open source and third-party components each year.

Today 90% of a typical application is comprised of open source code, and research shows that a single agency development organization may consume 200,000 open source and third-party components each year.

Most of these components are downloaded from public repositories like the (Maven) Central Repository, npms.org, rubygems.org, or PyPI.python.org. After all, why would agency developers create their own web application frameworks (e.g., Struts, Spring, etc.), crypto library (e.g., Bouncy Castle), or logging mechanism (e.g., Log4j) when they can quickly turn to proven components from open source projects and accelerate their development cycles?

However, not all components are created equal. As the need for agility rises, millions of developers have been flooding the software supply chain with components across many different development languages. Some of these components are more reliable and safe than others, and it's difficult for developers to separate the good from the bad.

As such, there's a good chance that the code that agency developers are downloading contains serious known vulnerabilities — and they could be unknowingly opening the door to

potential attackers. For example, for our State of Software Supply Chain report, we observed 12,000 engineering teams to document their consumption of open source and third party libraries, and we saw a 55% reduction in the use of vulnerable open source component releases within managed software supply chains. “Gartner estimates that through 2020, 99% of vulnerabilities exploited will continue to be ones known by security and IT professionals for at least one year,” wrote analysts Neil MacDonald and Ian Head in the September 2016 report DevSecOps: How to Seamlessly Integrate Security into DevOps.”

Let’s take a look at some strategies federal IT professionals can implement to ensure that their efforts toward acceleration remain unencumbered, yet secure.

1. Mitigate Risks by Proactively Managing Software Supply Chains

It may seem counterintuitive, but the more control federal developers have over their software supply chains, the more freedom they’ll have to develop applications that are vulnerability-free. That’s because proactively controlling and managing their software supply chains will enable them to better detect potential code vulnerabilities before they are put into development, saving them potentially enormous amounts of time and headaches later in the process.

Federal agencies should adopt SSCA best practices to fortify themselves against potentially unreliable and harmful code while being able to maintain the speed and agility afforded by DevOps. They must proactively gain control over their software supply chains by:

- Procuring components from fewer and better suppliers
- Procuring only the best parts from those suppliers
- Continuously tracking and tracing the precise location of every component across the supply chain

This final point can be achieved by creating a software bill of materials (BOM). A BOM helps agencies identify the open source and third party components used in the development of their applications. It includes information pertaining to number of components, component age, vulnerability risks, popularity, release history, and more. With a BOM, agencies can ensure that the open source and third party components they are using in their applications are safe and reliable. They can then adopt an automated set of governance policies to ensure the best components can be used safely over and over as necessary. This reduces the need to discover and download new components, allowing agencies to further accelerate development, since the necessary components will already be on-hand and deemed safe.

As DevOps and the use of open source and third party code has flourished, SSCA practices have begun to solidly take root in the federal government. Federal regulators and industry associations like FDA, FTC, UL, and FS-ISAC are taking action to build awareness and establish guidelines for sound SSCA best practices.

2. Scan for Vulnerabilities Across the Entire Application

One of the reasons attackers zero in on open source code is that it can be a highly effective gateway for exploiting many different applications. A single open source component may be

embedded in thousands of applications. This makes it easy for bad actors to propagate a virus that can spread and do significant damage.

Lack of effective vulnerability analysis tools make it difficult to address knowing what software components may be in use that will leave an organization at higher risk of exploit. Most of the applications being developed today are comprised of 90% open source code and only 10% custom code. However, most application vulnerability scanning tools that agencies employ in the RMF process, such as Static Application Security Testing and Dynamic Application Security Testing, inspect only the custom code, not the majority of the application that is comprised of open source or third-party components.

3. Improve Security Without Compromising Agility Through Automation

Many agencies have attempted to implement manual security measures to monitor the entirety of the application and define which components are and are not acceptable. Unfortunately this process has several drawbacks:

It's extremely time and resource consuming. Manual processes devoted to analyzing every aspect of an application's components can take an extraordinary amount of time and pull resources away from the development cycle. This undermines the entire basis for establishing a DevOps environment.

> "10 lines of code = 10 issues. 500 lines of code = looks fine."
>
> — @IAMDEVELOPER

Manual efforts are error prone. Put simply, people make mistakes, especially when they're working quickly. Despite having the best intentions, a federal developer trying to focus on delivering applications at a faster pace is probably more likely to miss something during analysis.

It's not scalable. There are thousands of pieces of open source code being introduced into government supply chains every day. The volume is massive, and will undoubtedly continue to grow. Manual process simply cannot keep up with this growth.

"There simply are not enough people with enough time for manual processes to adequately oversee application delivery. Governance processes that depend on manual inspection are guaranteed to fail," wrote analysts Diego Lo Giudice, Christopher Mines and Amy Homan in the November 2016 Forrester report entitled Use DevOps and Supply Chain Principles to Automate Application Delivery Governance. "Automating processes eliminates variability, reduces cost, and makes the remaining manual processes more visible."

Even if one believes that every developer in every instance is using only "approved" components, the manual effort to keep that "white list" up to date is enormous and endless. That is a big reason why RMF guidelines to federal agencies encourage the use of security control tools that are automated, consistent, and repeatable. Software supply chain automation solutions can help minimize those security blind spots that appear throughout the RMF process, dramatically improving cyber security controls, accelerating remediation efforts, and tracking improvements.

Automation allows federal developers to:

Define policies that are then automated along the entire development lifecycle. This is kind of like a "build once, deploy many times" scenario. By implementing automated processes, developers can establish policies at the very start of development that determine which components should or should not be used. Teams can be automatically alerted if one of these policies is violated. The end result is that application development proceeds without the need for manual interference, allowing the process to move more quickly and at the pace established and favored by the DevOps teams.

Be notified of recommended "fixes" in case a vulnerability is detected. In addition to being alerted if a bad piece of code enters the software supply chain, developers can also receive recommendations on how to address the vulnerability. The automated system can recommend alternative, safer component versions and remediate issues in real-time.

Both of these benefits serve to unshackle DevOps from the constraints of manual, bolt-on security practices. Agency personnel can continue their agile development efforts without having to worry about spending time trying to validate and approve every piece of code. Through automation, they will be able to successfully and efficiently monitor their software supply chains without impeding their development processes.

4. Create an Environment Built on Continuous Integration

SSCA must be an ongoing initiative that starts at the very beginning of the application design process and continues throughout final deployment.

Start by downloading safe components. Agencies should automatically evaluate any components that are being downloaded by developers and the tools they're using before they enter the application design or build processes.

Then, empower developers by supplying them with information on components they have selected in the design phase. Risky components must be immediately flagged, and developers should be alerted about both known issues and safer, alternative compo- nents. Information that will help them make choices about "good" or "bad" components should be made available immediately; they should not have to wait weeks or months for a security team to assess and provide feedback.

At the Continuous Integration (CI) stage, automated governance allows developers to be informed of potential issues in each build. Automated governance policies can be refined at this stage to prevent use of components with the most serious vulnerabilities. This makes it easier for devel- opers to address and fix these issues as they go along, enhancing productivity and supporting accelerated development.

This process effectively makes developers the first line of defense against potential vulnerabilities. While it alleviates the need for manual analysis, it empowers them to proactively and easily rectify problems before applications go into production. Further, it enables them to make smart decisions regarding application security. It also gives much more time to do what they really enjoy doing — creating — quickly, and efficiently.

It's important to remember that risk management needs to be ongoing. Components age more like milk than wine, and must be constantly monitored, updated, and patched. New vulnerabilities are frequently discovered in components previously thought to be safe. To keep applications from going sour, agency developers should deploy solutions that continue to monitor applications even after they've gone into production. Improved traceability of components over time with accelerate discovery of new vulnerabilities and compress mean time to repair.

Summary

Agencies are focused on accelerating application development. As such, they're deploying processes, such as DevOps, that support that mission and using open source code and third-party components to create applications more quickly and cost-effectively.

However, many are doing so at the expense of — knowingly or unknowingly — introducing vulnerabilities into their applications. Today, 90% of applications are comprised of open source code, but most tools only scan the 10% of the application that is custom built.

Still, developers do not want to be impeded by anything that will slow them down — including security measures designed to ensure they're using good code. Manual process do not work, as they consume too many resources and too much time.

Therefore, it's important to integrate automated SSCA processes into agencies' RMFs. These processes help minimize risk and catch potential vulnerabilities before, during, and after applications are developed. Through automation, developers can:

- Proactively manage their software supply chains to ensure components are trusted
- Analyze 100% of the application, rather than just 10%
- Be alerted to potential vulnerabilities so developers can quickly react and/or opt for safer, more reliable alternatives
- Be empowered to make smart security decisions regarding their applications
- Have more time to create and develop

The final point in particular illustrates the importance of SSCA for agencies with DevOps environments. Automation takes security out of the way of development and integrates it directly into each phase of the software development lifecycle and the RMF. This leaves developers with an unimpeded, automatically governed pathway that allows them to continuously develop applications at speed — without having to worry about potential vulnerabilities along the way.

Sonatype Platform

The Sonatype Platform can help agency developers achieve velocity. This platform applies proven supply chain best practices so federal agencies can use fewer and better components, reduce variety and variability, and track known vulnerabilities. Developers are guided and empowered to choose better components from the start to avoid unnecessary risk as well as the costs associated with downstream fixes. Policies can be automated throughout the entire SDLC. When new vulnerabilities are discovered, developers will be alerted so they know applications will be impacted and which components will become preferred.

For more information on how the Sonatype Platform can help your agency balance its DevOps and application security needs, visit www.sonatype.com/government.

sonatype

Sonatype is the software supply chain management company. We enable organizations to innovate faster in a highly competitive market. Our industry-leading platform empowers engineers to develop software fearlessly and focus on building products that power businesses. Sonatype researchers have analyzed more than 120 million open source components – 40x more than its competitors – and the Sonatype platform has automatically blocked over 115,000 malicious components from attacking software development pipelines. Enabling high-quality, secure software helps organizations meet their business needs and those of their customers and partners. More than 2,000 organizations, including 70% of the Fortune 100 and 15 million software developers, rely on our tools and guidance to be ambitious, move fast and do it securely. To learn more about Sonatype, please visit www.sonatype.com.

Headquarters
8161 Maple Lawn Blvd, Suite 250
Fulton, MD 20759
USA • 1.877.866.2836

European Office
168 Shoreditch High St, 5th Fl
London E1 6JE
United Kingdom

APAC Office
60 Martin Place, Level 1
Sydney 2000, NSW
Australia

Sonatype Inc.
www.sonatype.com