FOSSA

TL;DR

The movement away from the Waterfall Model of software development to one of agile development and continuous integration and continuous delivery/deployment (CI/CD) enable application development teams to deliver code changes more frequently and reliably.

DevOps combines software development practices and information technology operations to shorten the software development life cycle and align development with business objectives. DevOps has become increasingly important as software continues to become significant in every kind of business, even outside of "traditional" software companies.

Tasked with automating developer workflow, the DevOps team has a growing amount of responsibilities—especially when it comes to ensuring that the open source software used at a company is secure and compliant.

Legacy tools for compliance scanning destroy developer workflow by slowing down or freezing processes and requiring manual audits. By having an issue resolution workflow that can be integrated with CI/CD you don't hinder the development process — or halt it altogether when a one-off manual scan is required. It allows everything to be continuous and compatible with DevOps best practices such as agile method, CI/CD, automation, and security.

DevOps and Open Source 101: The DevOps Role in Modernizing Open Source Best Practices

Introduction

Combining software development practices and information technology operations can be key for shortening the software development life cycle and aligning development with business objectives. Because of this, DevOps has become a key part of the software development team at any company. DevOps has become increasingly important as software continues to become significant in every kind of business, as software development becomes a necessity across almost every industry. As of 2018, 72% of companies have started or fully adopted DevOps[1], as these teams are integral to embracing modern development patterns such as agile, CI/CD, and leveraging open source for building better and more reliable software. Tasked with automating developer workflow, the DevOps team has a growing amount of responsibilities—especially when it comes to ensuring that the open source software used at a company is secure and compliant.

91%

OF ORGANIZATIONS REPORT ADOPTING THE AGILE DEVELOPMENT METHODOLOGY.[2]

Modern Development Practices

As developers continue to shift towards modern practices, the Agile Method has supplanted the more static and rigid Waterfall Model of software development as the best practice. In fact, 91% of organizations report that they have adopted the Agile Development Methodology[2].

While the Waterfall Method allowed for more control and predictability on the surface, lack of flexibility restricts the benefits of iterative improvements and more frequent testing. In addition to

1 Belagatti, Pavan. "What Did We Learn about DevOps in 2018? - DZone DevOps." *Dzone.com*, Statista, 11 Jan. 2019, https://dzone.com/articles/what-did-we-learn-about-devops-in-2018

2 "Sauce Labs Releases Fourth Annual 'State of Testing' Survey." Sauce Labs, Dimensional Research, https://saucelabs.com/news/sauce-labs-releases-fourth-annual-state-of-testing-survey

leveraging new technology and a movement to the cloud, this shift from Waterfall to agile has become the standard in the evolution of development processes. The Agile Method requires continuous software development and automated processes to deliver frequent and stable software releases.

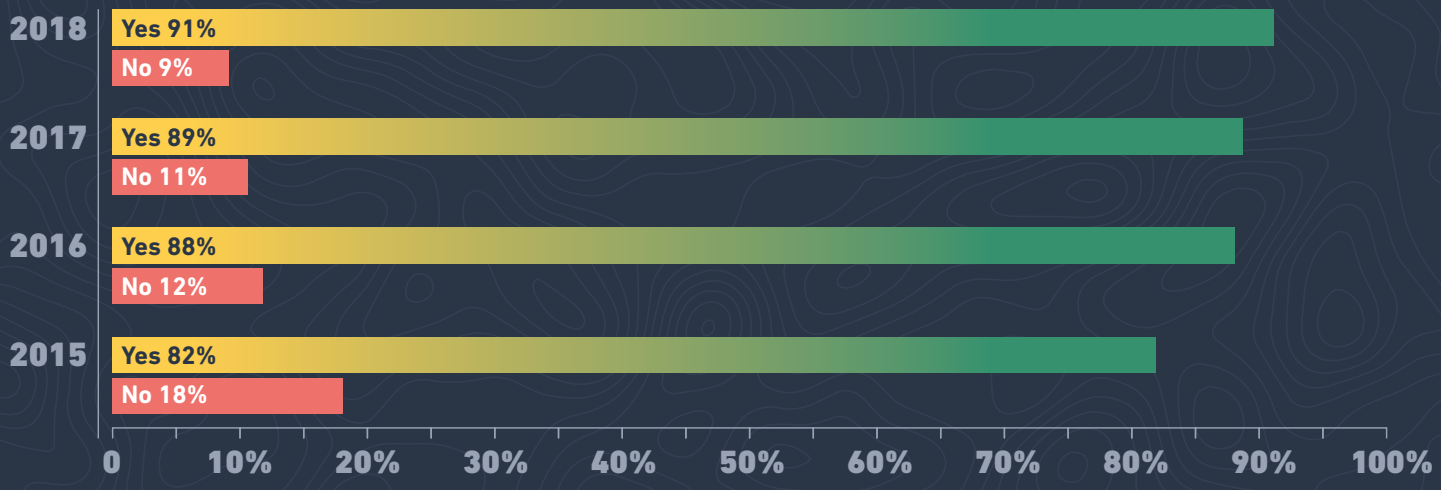
Coupled with a more continuous deployment cycle, engineers are taking advantage of the shift towards the cloud by pushing out more frequent releases. A large part of moving to agile is creating a culture of CI/CD, also known as continuous integration and continuous delivery/deployment. Continuous integration (CI) and continuous delivery/deployment (CD) is a culture, a set of operating principles, and a collection of practices that enable application development teams to deliver code changes more frequently and reliably. CI/CD, governed by DevOps and paired with Agile Method provides a dependable, efficient, and high-quality method for the software development process.

Modern Development and Open Source

As companies continue to modernize their software, engineers are more likely than not using open source software. As the Red Hat report titled *The State of Enterprise Open Source* states, "The question is no longer whether your enterprise should adopt open technologies. The question is when—and how." Engineers are constantly incorporating open source components to complete their projects to help with digital transformation, application integration, application modernization, application development, and DevOps. Engineers can easily pull open source components into their codebase to enrich their proprietary code.

> **THE QUESTION IS NO LONGER WHETHER YOUR ENTERPRISE SHOULD ADOPT OPEN TECHNOLOGIES. THE QUESTION IS WHEN—AND HOW.**
>
> **— "The State of Enterprise Open Source" Red Hat Software Report**

» HAS YOUR ORGANIZATION ADOPTED AN AGILE DEVELOPMENT METHODOLOGY?



Source:
"Sauce Labs Releases Fourth Annual 'State of Testing' Survey." Sauce Labs, Dimensional Research, https://saucelabs.com/news/sauce-labs-releases-fourth-annual-state-of-testing-survey

> **THE ENVIRONMENT OF FREQUENT RELEASES (IF NOT MONITORED) CAN PUSH LICENSING AND COMPLIANCE ISSUES OR SECURITY VULNERABILITIES INTO THE CODESET WITHOUT PROPER REVIEW.**

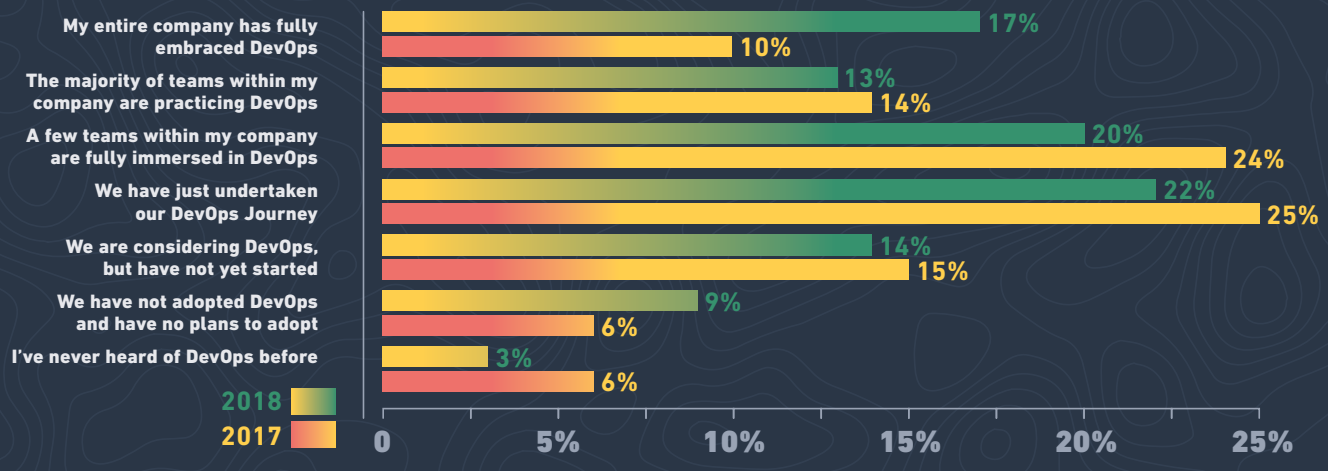
Using open source effectively and safely does require some governance, as the licenses associated with the open source components need to be approved for use and any vulnerabilities should be patched. You can't always plan what open source you need up front, which means the approval process can bring development projects to a standstill. And because the engineering team is constantly iterating and pulling in new open source components, a manual process of review can be very painful—especially when engineers are also using a large amount of open source.

With each open source project, security and compliance need to be vetted. Security is an obvious concern, but open source license compliance is also essential because a lack of compliance to otherwise "free" open source tools can have a revenue impact when proof is required for enterprise sales, M&A, IPO or distribution on application platforms. Maintaining a continuous level of visibility and governance over your software's composition in agile development and CI/CD is especially important, because the environment of frequent releases (if not monitored) can push licensing and compliance issues or security vulnerabilities into the codeset without proper review.

Speed + Security = DevOps

A key role as a member of the DevOps team is to help move product innovation and software development at a faster pace than organizations that use traditional software development and infrastructure management processes. Parts of this role include automating processes, implementing CI/CD into build pipelines, and ensuring

» EXTENT OF DEVOPS ADOPTION BY SOFTWARE DEVELOPERS WORLDWIDE IN 2017 AND 2018



Source:
Belagatti, Pavan. "What Did We Learn about DevOps in 2018? - DZone DevOps." Dzone.com, Statista, 11 Jan. 2019, https://dzone.com/articles/what-did-we-learn-about-devops-in-2018

software is compliant and secure. This is accomplished by using automated build tools so that development moves faster (while making everyone's job easier and less painful). Automating these processes removes manual errors, is more accurate, and increases speed. By automating each step of the software development process, teams are able to deliver frequent releases at a high quality.

As a part of the transition to Agile Method, DevOps implement CI/CD in their company's pipeline. Fully moving your company over to CI/CD is an iterative process that is never truly over. The primary goal with this transformation is to automate software development process from build to release. A CI/CD pipeline enables developers to be flexible and able to adjust to different development environments without slowing down. This automation removes repetitive manual processes that take away from engineering productivity. Automation also improves efficiency so that software can be delivered more quickly.

Part of ensuring high-quality deploys is ensuring that the software being built is stable and secure. As engineers continue to pull open source packages from online repositories, the DevOps frequently helps to ensure that production codebase remains compliant and secure. For many companies, security checks are built into every part of the DevOps lifecycle, from inception to release. In fact, security is so vital that a new role within DevOps, known as DevSecOps, is increasing in popularity. DevOps security usually entails vulnerability management to help engineering teams with identifying and solving any security issues. This team helps integrate security tools into the CI/CD process to ensure that the speed of safeguarding a secure code base matches the speed of development while reducing risk and providing efficiency.

Open Source & the Value of DevOps

All this automation can come to a standstill when it comes to software audits validating third-party licenses and vulnerabilities. The most common process for auditing software is extremely manual and tedious. It starts with scanning (or manually listing) and identifying the open source dependencies, which is just the first step in a lengthy auditing process. Most legacy software composition tools give you a list of only the first layer of dependencies and the declared licenses, but these layers can actually be several levels deep (dependencies within dependencies). And this list only helps you identify the top layer of dependencies used, as you still need to compare that list against your standards and practices to determine if any component violates your internal policies. To resolve the problems, you would have to take the compliance and security issues to a lawyer, engineer, or security team to see if you have to

"FOR MANY COMPANIES, SECURITY CHECKS ARE BUILT INTO EVERY PART OF THE DEVOPS LIFECYCLE, FROM INCEPTION TO RELEASE. IN FACT, SECURITY IS SO VITAL THAT A NEW ROLE WITHIN DEVOPS, KNOWN AS DEVSECOPS, IS INCREASING IN POPULARITY."

> **WHEN LEGACY TOOLS GIVE YOU A LIST OF DEPENDENCIES WITHOUT SUGGESTING RESOLUTIONS, THE LAWYER AND ENGINEER HAVE TO GO BACK AND FORTH TO DETERMINE HOW THE OPEN SOURCE DEPENDENCY IS USED AND WHETHER IT IS ACCEPTABLE BASED ON THE LICENSE.**

change the open source package or if the one you are using is okay — and it still only digs into that first level of dependencies, leaving deeper layers of dependencies unexamined.

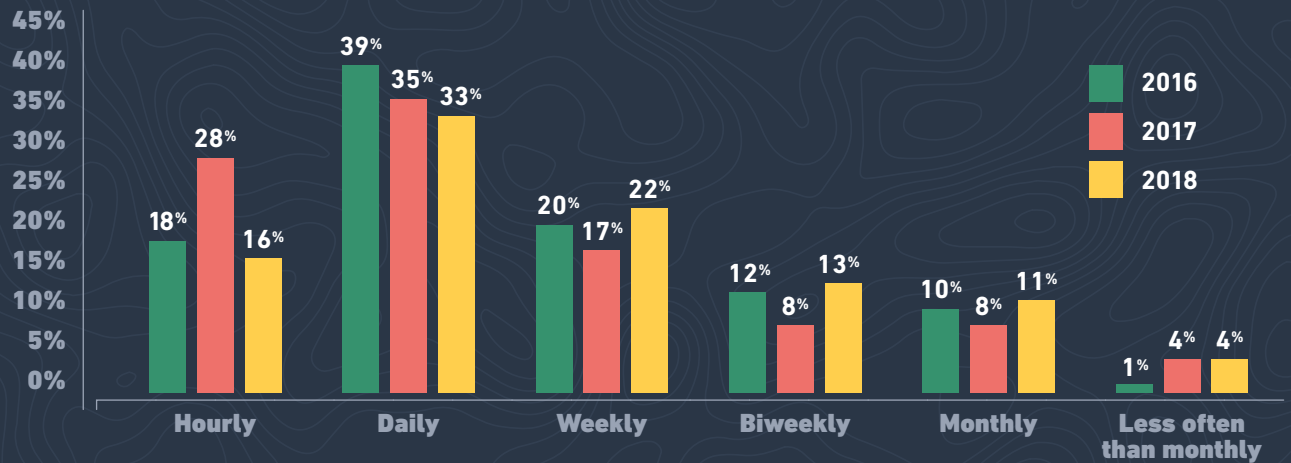
Additionally, deciding whether or not there is a compliance issue is tricky. Unlike most security issues, licensing is nuanced because it depends on how the open source package is being used in your codebase or how the software is being distributed. When legacy tools give you a list of dependencies without suggesting resolutions, the lawyer and engineer have to go back and forth to determine how the open source dependency is used and whether it is acceptable based on the license. This not only creates a difficult and lengthy resolution process, it also temporarily short-circuits the advantages of agile and CI/CD, breaking the automated deployment processes the DevOps team has implemented.

Deployment - Compliance = Risk

When auditing, a public-facing report needs to be generated for the entities that need the audit. However, the most common auditing process does not work with DevOps best practices such as CI/CD implementation, automation, and compliance because of a reliance on manual processes. Because Agile Method is continuous, oftentimes new code is being committed before a previous audit can be finished. This delay leads to a code freeze in order to finish the audit, which slows down development and destroys DevOps workflow.

The most common auditing process can either lead to a delay in code being committed because of audits not being completed in time, or it can lead to risk when code is exposed because it has been continuously deploying before an audit has been finished.

» "IDEALLY, HOW OFTEN WOULD YOUR TEAM LIKE TO DEPLOY A NEW BUILD?



Source:
"Sauce Labs Releases Fourth Annual 'State of Testing' Survey." Sauce Labs, Dimensional Research, https://saucelabs.com/news/sauce-labs-releases-fourth-annual-state-of-testing-survey

Incorporating Open Source Management Into CI/CD

Legacy tools destroy developer workflow by slowing down or freezing processes and requiring manual audits. By having an issue resolution workflow that can be integrated with CI/CD, you don't have to stop developers from working or do one-off scans. It allows everything to be continuous and compatible with DevOps best practices such as Agile Method, CI/CD, automation, and security.

That's where FOSSA comes in to the picture. FOSSA is a modern open source management tool that integrates directly with your developer workflow so you can audit your software without having to stop development.

ABOUT FOSSA:

FOSSA is the world's first Modern Open Source Management platform. Designed for development and legal teams alike, FOSSA provides component intelligence, continuous compliance, and cross-team collaboration solutions that enable engineering excellence and accelerate market capture while mitigating business risk.

FOSSA

FOSSA.com | Sign up with Github | tldrlegal.com

FOSSA, Inc. | Modern Open Source Management
950 Howard Street, San Francisco, CA 94103