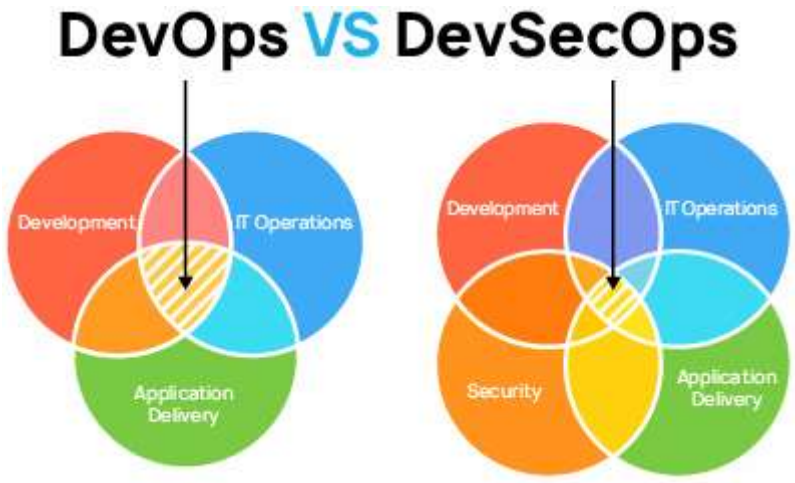
Ekaterina Nikiforova
Feb 04 2020

Tags:
#DevOps #Security

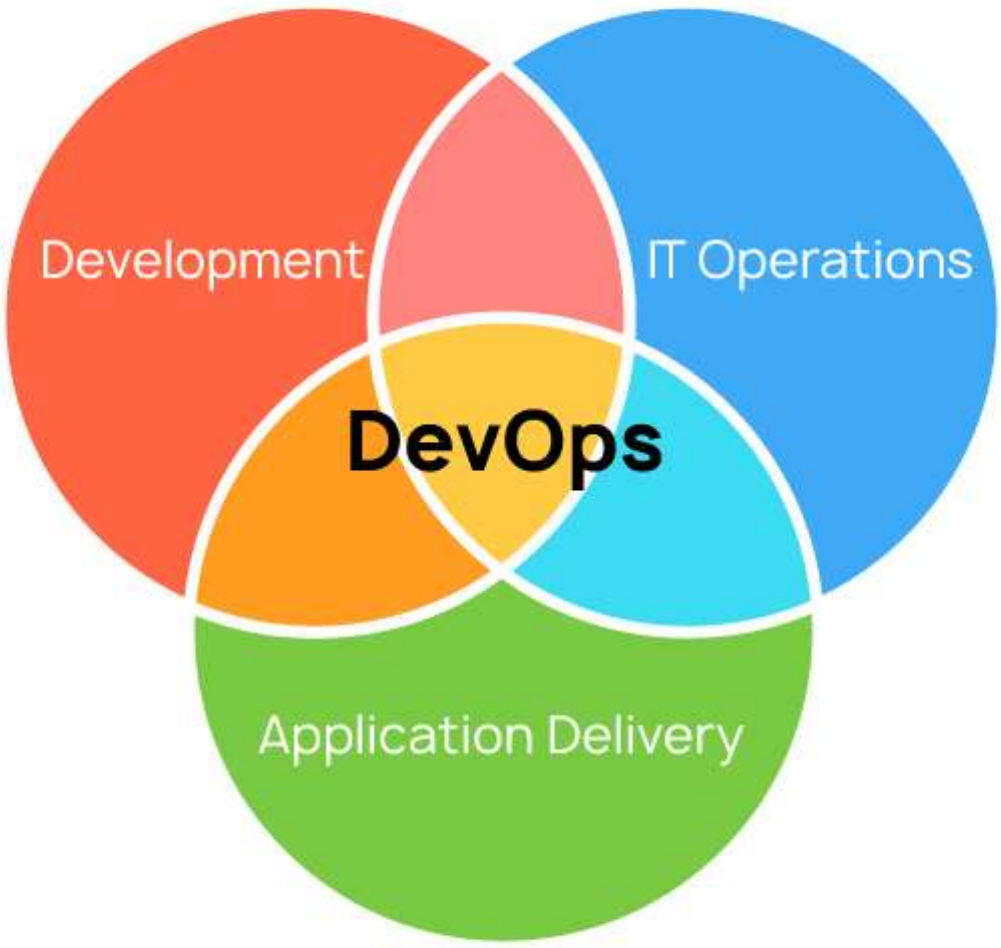
What Is the Difference between DevOps and DevSecOps?

DevOps is a methodology aiming at establishing closer collaboration between programmers and system administrators in the software development process. A DevOps engineer is a specialist working on the intersection of these two fields. A DevOps engineer's primary objective is to take the predictability, efficiency, and security of software development to the highest level possible. DevSecOps is a further development of the DevOps concept that, besides automation, addresses the issues of code quality and reliability assurance.



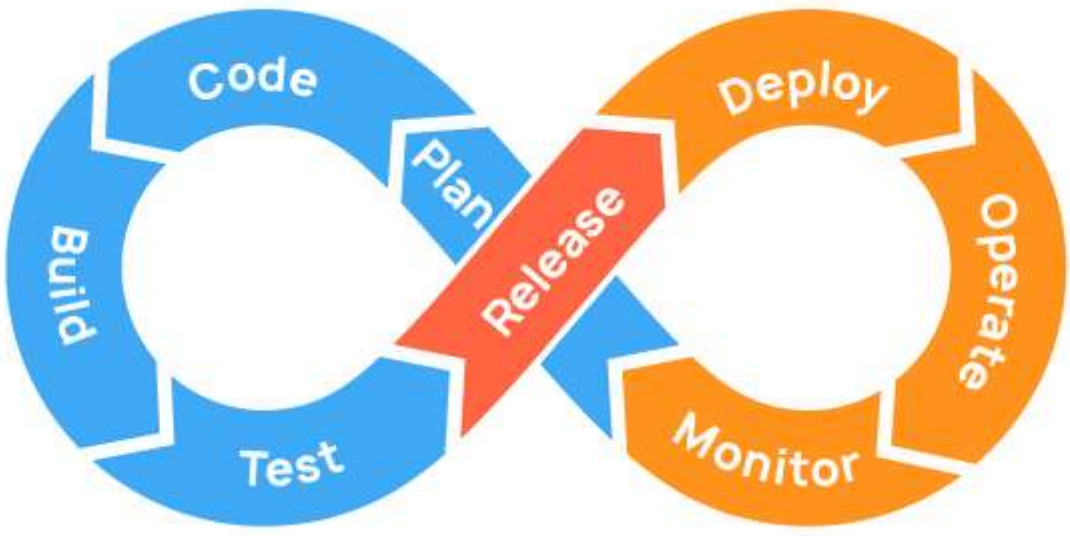
DevOps methodology

Let's talk about the DevOps methodology first. Its objective is to unite the activities related to development, quality assurance, deployment, and integration. DevOps can be called a philosophy that aims at building up a culture of collaboration between originally isolated teams. Software development and deployment operations have been traditionally carried out by two different persons or departments. DevOps aims at improving the efficiency by eliminating the boundaries between these two phases of software development. For any technical specialist, DevOps is a key factor in optimizing time and resources for better productivity, understanding, and training.



What is a DevOps engineer?

Since this job title emerged quite recently, at the end of the '00s, it has no precise definition yet. We suggest the following definition: a DevOps engineer is a person who helps an organization adopt the DevOps methodology.



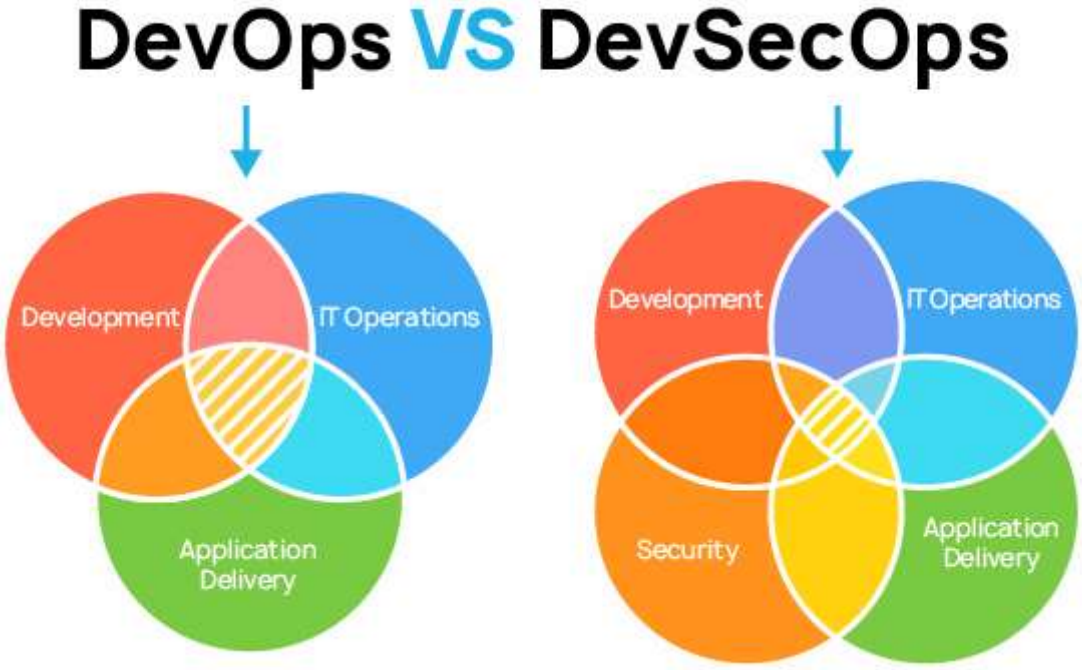
There is a set of areas that a DevOps engineer must be competent in:

- Version control systems. Any DevOps engineer's resume has a line mentioning Git, SVN, Mercurial, etc.
- Continuous integration (CI). As for this one, you should look for Jenkins and TeamCity. It's important to note, however, that there are so many DevOps tools out there that it is impossible to cover them all in this article. So don't be too picky about the exact names of CI tools an applicant mentions in his or her resume.
- Containers such as Docker or Vagrant.
- Framework automation tools. What counts here is good knowledge of Python, Shell, or Bash. Besides providing the obvious benefits of automation, it also helps DevOps engineers save a lot of time.
- Cloud services. This is one of the most essential skills a DevOps engineer must possess. Microsoft Azure, Google Cloud, Amazon Web Services are some examples.
- Testing. One of the duties of a DevOps engineer is to speed up the delivery of software to the clients. Since most companies care about the quality of their software, testing has become an essential part of DevOps engineers' job.
- Communication. A DevOps specialist must have well-developed empathy because their job implies a lot of communication with other people. Conflicts aren't going to do them much good.

Next, we'll answer the question asked at the very beginning of this post.

DevSecOps: what's that and how is it different from DevOps?

Similarly to DevOps, DevSecOps can also be called a culture of its own. DevSecOps is a philosophy of integrating security methods into a DevOps process. Team work is as crucial to a DevSecOps engineer as it is to a DevOps engineer: their ability to resolve conflicts and conduct productive negotiations plays a crucial role in creating secure applications. From the very start of a SDLC, DevSecOps works to make the application secure by introducing a variety of security techniques.



DevOps heavily relies on automation. The same is true for DevSecOps, which aims at automating every aspect, including security audit.

To stimulate the adoption of DevSecOps, the Cloud Security Alliance company has defined six fundamental aspects, or pillars:

- Collective Responsibility: Security is not something ephemeral whose progress and contribution cannot be measured. Each person in the organization has their own security responsibility and must be aware of their own contribution to the organization's security stance.
- Collaboration and Integration: Security can only be achieved through collaboration, not confrontation.

Automation: Software quality can be bettered by improving the thoroughness, timeliness and frequency of testing. Processes that can be automated should be automated, and those that can't should be considered for elimination.

Measurement, Monitoring, Report and Action: The results during software development as well as post-delivery must be continuously controlled by the qualified people at the right time for DevSecOps to succeed.

Practical steps to transitioning from DevOps to DevSecOps

The transition from DevOps to DevSecOps requires understanding the particular techniques and practices ensuring software security. Let's discuss this aspect in more detail and find out what technologies exactly will be needed.

First, I suggest adopting Dynamic Application Security Testing (DAST) tools. As tools performing the so called black-box testing, dynamic analyzers can identify program vulnerabilities such as SQL injections, buffer overflows, and the like. Building dynamic analyzers into your software development process is one of the steps towards the DevSecOps practices.

Runtime Application Self-Protection (RASP) is one of the security technologies used at runtime. RASP analyzes the application's behavior, thus implementing continuous security analysis.

Interactive Application Security Testing (IAST). The IAST approach analyzes the application from the inside at runtime and keeps track of code execution in memory, looking for specific events that could lead to a vulnerability. These events are further analyzed to see if they are clean or pose a risk of causing a vulnerability.

Static Application Security Testing (SAST) is used to check the code without actually executing it. SAST helps find potential vulnerabilities in the source code, thus preventing multiple possible zero-day vulnerabilities. Common Weakness Enumeration (CWE) is one of the most popular classifications of warnings produced by SAST tools. CWE is an official list or dictionary of common security weaknesses exploitable by intruders to obtain unauthorized access to the system. Using a static analyzer as part of the development process will help prevent software bugs from getting to the next level, CVE. CVE (Common Vulnerabilities and Exposures), is a database of widely known information security vulnerabilities, which was worked out as an attempt to make an ordered list of known software defects.

Software Composition Analysis, SCA. SCA can identify vulnerabilities in open-source components and analyze applications to see if they include components that are known to contain vulnerabilities.

Conclusion

Let's recap.

DevSecOps is a practice of integrating security objectives into the DevOps methodology. Security automation in DevOps is an aspect that requires new approaches, technologies, and tools. DevSecOps can be viewed as an extension of the DevOps methodology since DevSecOps builds on it. If you want to learn more on the topic, the following resources should be helpful:

Rudolf Groetz. DevSecOps – How to Add Security Testing to your Delivery Pipeline
Red Hat. What is DevSecOps?
Quora. What is the difference between DevOps and DevSecOps?
Sergey Vasiliev. How Can PVS-Studio Help in the Detection of Vulnerabilities?
Wikipedia. DevOps
Atlassian. DevOps: Breaking the Development-Operations barrier
Graeme Messina. Transitioning from DevOps to DevSecOps

Tags:

#DevOps #Security

Popular related articles

The Ultimate Question of Programming, Refactoring, and Everything

Date: Apr 14 2016
Author: Andrey Karpov

Yes, you've guessed correctly - the answer is "42". In this article you will find 42 recommendations about coding in C++ that can help a programmer avoid a lot of errors, save time and effort. The au...

Characteristics of PVS-Studio Analyzer by the Exam of EFL Core Libraries, 10-15% of False Positives

Date: Jul 31 2017
Author: Andrey Karpov

After I wrote quite a big article about the analysis of the Tizen OS I received a large number of questions concerning the percentag false positives and the density of errors (how many erro...

Comments (0)