

Raw data (DevSecOps Definitions)

1. Definitions quoted or paraphrased from Mohan & Othmane (12):

IEEE_01: DevSecOps or SecDevOps is seen as a necessary expansion to DevOps, refers to incorporating security practices in the DevOps processes by promoting collaboration between the development teams, the operations teams, and the security teams.

IEEE_08: DevSecOps or SecDevOps is seen as a necessary expansion to DevOps, refers to incorporating security practices in the DevOps processes by promoting collaboration between the development teams, the operations teams, and the security teams.

IEEE_26: DevSecOps can be defined as a cultural approach to improve and accelerate the delivery of business value by making dev/sec/ops teams' collaboration effective.

ACM_68: SecDevOps has emerged as a new popular practice where the development, operations and security teams collaborate in a more unified process.

SC_02: This concept is an attempt at creating and including modern security practices that can be incorporated in the fast and agile world of DevOps. It promotes an extension to DevOps' goal of promoting collaboration between developers and operators by involving security experts from the start as well.

SC_04: DevSecOps is seen as a necessary expansion of DevOps that aims to integrate security controls and processes into the DevOps software development cycle by promoting the collaboration among security teams, development teams and operations teams.

SC_09: The integration of security with development and operation is called DevSecOps.

SC_10: It refers to the incorporation of security practices in a DevOps environment through the collaboration between development, operation, and security teams.

SC_11: SecDevOps and DevSecOps have appeared, and they incorporate security practices in DevOps environments.

SC_21: DevSecOps or SecDevOps is seen as a necessary expansion to DevOps, refers to incorporating security practices in the DevOps processes by promoting collaboration between the development teams, the operations teams, and the security teams.

ACM_45: DevSecOps has a reliance on operational tools, established agile engineering practices and the right culture of collaboration.

2. Definitions quoted or paraphrased from Rahman & Williams (4):

IEEE_12: DevSecOps to refer to the concept of integrating security principles through increased collaboration between the development teams, operations teams, and security teams of a DevOps organization.

IEEE_08: Rahman and Williams also emphasizes on the importance of security, by breaking silos of security, sharing that knowledge with various teams of software development process in order to build the relationship between them.

IEEE_44: SecDevOps “(also known as DevSecOps and DevOpsSec) is the process of integrating secure development best practices and methodologies into development and deployment processes which DevOps makes possible.

SC_22: Ur Rahman and Williams (2016), leading to the introduction of more security-oriented processes, named SecDevOps or DevSecOps Mohan and Othmane (2016) .

3. Definitions quoted or paraphrased from Myrbakken & Colomo-Palacios (4):

SC_02: DevSecOps is seen as a necessary expansion to DevOps, where the purpose is to integrate security controls and processes into the DevOps software development cycle and that it is done by promoting the collaboration between security teams, development teams and operations teams.

SC_03: DevSecOps is defined as the integration of security practices into DevOps.

IEEE_10: DevSecOps is defined as the integration of security processes and practices that are meant to shift the mindset of all participants in the SDLC to get everyone to do what they can to ensure security of a system.

ACM_53: DevSecOps has been introduced in order to integrate security practices within the DevOps process.

4. Definitions quoted or paraphrased from Carter (2):

IEEE_24: DevSecOps is about using the DevOps methodology for security. It is about breaking the silos of security, giving that knowledge to the different teams, and ensuring that security is implemented at the right level and at right time. DevSecOps puts security at the forefront of requirements to avoid the costly mistakes that come from treating security as an afterthought.

IEEE_26: DevSecOps is about “breaking the silos of security, giving that knowledge to the different teams, and ensuring that security is implemented at the right level and at right time”.

5. Definitions quoted or paraphrased from Carturan & Goya (2):

ACM_04: DevSecOps is a set of practices about IT processes with security approach. This aims the time reduction of development and of transferring into production environment, without lost quality of the software in terms of code, security and delivery mechanisms.

IEEE_21: DevSecOps is a set of practices about IT processes with security approach. This aims the time reduction of development and of transferring into production environment, without lost quality of the software in terms of code, security and delivery mechanisms.

6. Definitions quoted or paraphrased from Mohan, Othmane, & Kres (1)

SC_11: SecDevOps is a paradigm for integrating the software development and operation processes considering security and compliance requirements.

Definitions without clear references or derived by authors themselves:

07. DevSecOps is an agile based software development process which integrates continuous security into the original DevOps development process. (S2_IEEE_03, 2019)

08. DevSecOps is a combination of development, security and operations. It makes the team responsible for application security by implementing security activities and decisions at the same measure and speed as development and operations tasks. (S2_IEEE_05, 2019)

09. DevSecOps is a term that promotes shifting security to the early stages of a project. (S2_IEEE_06, 2019)

10. The broad purpose of DevSecOps is to break down the barrier between software development and operations practices, increase deployment rates, and increase the rate of testing feedback while maintaining a secure operational atmosphere. (S2_IEEE_22, 2020)

11. Inclusion of security into DevOps workflow has resulted in an extension to DevOps, called DevSecOps. In simple term, goal of DevSecOps is to safeguard application from any potential threats by including security mechanisms into all phases DevOps workflow. (S2_SC_01, 2021)

12. In some DevOps models, quality affirmation and security groups may turn out to be more firmly coordinated with improvement and activities and all through the application lifecycle. At this point, when security is the main emphasis of everybody's on a DevOps group, this is to be alluded to as DevSecOps. (S2_SC_14, 2020)

13. The term "Secure DevOps" indicates security is given high priority throughout the SDLC and general software development security practices are implemented at each stage of the cycle. (S2_ACM_07, 2018)

14. DevSecOps refers to the injection of security principles and controls into the DevOps model that again integrates development and operational work while focusing on cultural shifts to empower the success of the new development process. (S2_ACM_50, 2019)

15. Transforming security from being treated as just another non-functional requirement to a key concern throughout all phases of the development lifecycle and even post deployment, supported by a smart and lightweight approach to identifying security vulnerabilities. (S2_SC_31, 2014)

01 DevSecOps is a term that emphasizes the importance of sound information security practices in the pursuit of continuous delivery. Because the origins of DevOps did not explicitly include security as a top-level concern (as it did for development and operations) DevSecOps has emerged as a popular label that avoids any risk of security being an afterthought. S2_GL_01: <https://www.scaledagileframework.com/devops/>

02 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. S2_GL_02:

<https://pvs-studio.com/en/blog/posts/0710/>

03 DevSecOps means thinking about application and infrastructure security from the start. It also means automating some security gates to keep the DevOps workflow from slowing down. DevSecOps is about built-in security, not security that functions as a perimeter around apps and data. S2_GL_04:

<https://www.redhat.com/en/topics/devops/what-is-devsecops>

04 DevSecOps simply means placing security practices early during the SDLC (Software Development Life Cycle) processes within an agile framework. S2_GL_05:

<https://securityboulevard.com/2020/08/devops-vs-devsecops-what-is-the-difference/>

05 DevSecOps refers to the integration of security practices into a DevOps software delivery model. Its foundation is a culture where development and operations are enabled through process and tooling to take part in a shared responsibility for delivering secure software. The definition of DevSecOps Model, at a high-functioning level, is to integrate security objectives as early as possible in the lifecycle of software development. While security is “everyone’s responsibility,” DevOps teams are uniquely positioned at the intersection of development and operations, empowered to apply security in both breadth and depth. S2_GL_10: <https://snyk.io/devsecops/>

06 DevSecOps is a culture shift in the software industry that aims to bake security into the rapid-release cycles that are typical of modern application development and deployment, also known as the DevOps movement.

S2_GL_11:

<https://www.csoonline.com/article/3245748/what-is-devsecops-developing-more-secure-applications.html>

07 DevSecOps is about creating a culture where security is a part of everyone’s job, not just the people specifically working in security roles. Security needs to be at the top of every developer’s mind as they build, test, and release features to production. S2_GL_12: <https://www.bmc.com/blogs/devops-devsecops/>

08 DevSecOps is a combination of both DevOps and SecOps, fusing both methodologies together to create a cyclical system that brings in information and practices from software development, cyber security, and technology operations focuses. S2_GL_13:

<https://www.clouddefense.ai/blog/devops-vs-devsecops-the-differences>

09 DevSecOps is a joint effort by development, security and operations personnel to ensure that products are released efficiently and securely from the start. S2_GL_15:

<https://www.imperva.com/learn/application-security/devsecops-devops-security/>

10 DevSecOps is the practice of integrating security into a continuous integration, continuous delivery, and continuous deployment pipeline. S2_GL_16:

<https://www.atlassian.com/devops/devops-tools/devsecops-tools>

11 DevSecOps, also known as SecDevOps, is a software development philosophy that promotes the adoption of security through the entire software development lifecycle. DevSecOps goes beyond a single tool or specific practice, but generally speaking, it favors security automation, communication, and scalability. This development philosophy scope includes all teams involved in the SDLC, such as development, operations, and security.

S2_GL_19:

<https://hdivsecurity.com/bornsecure/devsecops-the-7-key-factors-to-secure-your-devops-practice/>

12 DevSecOps is a set of practices of adding security components to each step of the DevOps process. It aims to shorten the systems development life cycle and provide continuous delivery with high software quality while taking care of the security aspect. S2_GL_23:

<https://blog.pentesteracademy.com/devsecops-learning-path-integrating-security-with-devops-1cc03670552f>

13 DevSecOps is the philosophy of integrating security practices within the DevOps process. DevSecOps involves creating a ‘Security as Code’ culture with ongoing, flexible collaboration between release engineers and security teams. S2_GL_26:

<https://www.sumologic.com/insight/devsecops-rugged-devops/>

14 If you want a simple DevSecOps definition, it is short for development, security and operations. Its mantra is to make everyone accountable for security with the objective of implementing security decisions and actions at the same scale and speed as development and operations decisions and actions. S2_GL_27:

<https://www.forcepoint.com/cyber-edu/devsecops>

15 DevSecOps is an extension of the DevOps approach that considers security as a shared responsibility that has to be integrated into the development process from the beginning. S2_GL_33:

<https://www.padok.fr/en/blog/devsecops-security>

Codes of DevSecOps Definitions from WL

expansion to DevOps: 5

extension to DevOps: 2

incorporating security practices in the DevOps processes: 3

incorporation of security practices in a DevOps environment: 2

integrate security controls and processes into DevOps: 2

integration of security practices into DevOps: 2

including modern security practices that can be incorporated in the fast and agile world of DevOps

integration of security with development and operation

integrating security principles

integrating secure development best practices and methodologies into development and deployment processes

integration of security processes and practices

introduction of more security-oriented processes

integrating the software development and operation processes considering security and compliance requirements

integrates continuous security into the original DevOps development process

inclusion of security into DevOps

including security mechanisms into all phases DevOps workflow

injection of security principles and controls into the DevOps

IT processes with security approach: 2

development, operations and security teams: 9

dev/sec/ops

developers and operators by involving security experts

collaboration/collaborate: 11

culture

cultural approach

cultural shifts

shift the mindset

agile: 3

smart and lightweight approach

breaking silos of security: 3

break down the barrier

sharing that knowledge

giving that knowledge to the different teams: 2

security is the main emphasis

security is given high priority throughout the SDLC

security is a key concern throughout all phases of the development lifecycle and even post deployment

security practices are implemented at each stage of the cycle

security is implemented at the right level and at right time: 3

puts security at the forefront of requirements

shifting security to the early stages

time reduction: 2

increase deployment rates

increase the rate of testing feedback

without lost quality: 2

quality affirmation

responsible for application security

maintaining a secure operational atmosphere

safeguard application from any potential threats

identifying security vulnerabilities

reliance on operational tools

These codes are some names of authors who presented common DevOps definitions and their definitions were quoted or paraphrased by selected papers.

Mohan & Othmane (12)

Rahman & Williams (4)

Myrbakken & Colomo-Palacios (4)

Carter (2)

Carturan & Goya (2)

Mohan, Othmane, & Kres (1)

Codes of DevSecOps Definitions from GL

philosophy: 4

culture: 3

culture shift

tooling

agile

extension of the DevOps

combination of both DevOps and SecOps

integrating security methods into a DevOps process

integrating security practices within the DevOps process

integration of security practices into a DevOps

integrating security into a continuous integration, continuous delivery, and continuous deployment pipeline

adding security components to each step of the DevOps

development, operations, and security: 3

bake security into the rapid-release cycles

collaboration

team work

communication

shared responsibility: 2

everyone's responsibility

security is a part of everyone's job

make everyone accountable for security

at the top of every developer's mind

emphasizes the importance of sound information security practices

security from the start: 2

from the beginning

placing security practices early during the SDLC

integrate security objectives as early as possible

avoids any risk of security being an afterthought

adoption of security through the entire software development lifecycle

built-in security

automation/automating: 2

Security as Code

scalability

shorten the systems development life cycle

high software quality

Codes of DevSecOps Challenges

Resistance to change in the organization's culture and people mindset

Developer resistance to integrate security protocols 2

Developers lose autonomy

Teams working in isolation

no communication, collaboration and sharing

synchronization and transparency issues

Challenge of unrestricted collaboration 2

Coordination of security team and DevOps team 2

Untrusted inputs causing isolation 2

Conflict between security and development

Lack of clarity and transparency in strategy

Lack of commitment of leadership and senior management for DevSecOps adoption

Management does not prioritize security

Low or no confidence in DevSecOps principles, practices and potential benefits

lack of trust and skepticism

Lack of understanding and awareness of DevSecOps principles and practices

improving security awareness

Nobody is responsible for security

Security in the team and to the left

Security push-pull

Cross team dependencies and some domains don't allow or cause difficulties in adopting DevSecOps

A need for new standards for security prevention, detection and response

Lack of secure coding standards

Lacking security education

Lacking knowledge and training

Lack of skills

lack of knowledge

Restructuring organization and implementing DevSecOps practices can lead to high cost such as salaries for security experts, costs on new tools

Insufficient number of resources to align with DevSecOps practices

Abundance of information is a serious threat to secure data 2

Risk and cost battle

Neglecting change control in security

The boundary between a specialist and generalist

Customer readiness for applying frequent releases to production setup

Codes of DevSecOps Practices

Be reactive and responsive; 32

Cultural shift; 41

Learn from each other; 32

Cross-functional collaboration; 30

Foster collaboration between security and development; 25

Open contribution and collaboration; 24

Collaboration and integration; 02

Communicate and collaborate; 32

Improving empathy and cooperation; 10

Establishing a security champions program; 10

Security evaluation and learning; 14

Change the security mindset; 32

Get buy-in from stakeholders (make security a priority); 32

Collective responsibility; 02

Assign security responsibility to one person from your DevOps team; 28

Provide training 06, Get training 10, Enabling through training 32, Cross-training 35;

Shared threat intelligence; 24

Educate developers to gain buy-in; 25

Invite InfoSec to demos; 18

Reinforcing success; 10

Pragmatic implementation; 02

Reducing delivery friction; 10

Data and security science vs. fear, uncertainty, and doubt; 24

Define security requirements; 06

Define metrics 06, 19, Measurement 02

Bridging the divide between compliance and development; 02

Separation of duties 14, 17;

Report 2; better reporting 19;

Version control, metadata, and orchestration; 10

Compliance; 10

Compliance operations over clipboards and checklists; 24

Identify compliance requirements beforehand; 28

Security architecture; 10

Proactive security assessments; 10

Security requirements and design; 14

Incident management 08, 10; Vulnerability management 23, 30; Vulnerability and incident management 14;

Availability and business continuity management; 14

Integrate security into CI/CD practices; 17

Integrate security during the planning phase; 35

Secure by design – embed security into each release 31;

Embedded security; 19

Take a proactive approach to security; 17

Include security early in the life cycle; 28

Moving security to the left (08, 09, 13, 15, 18, 31, 35, 36)

Conduct security reviews; 18

Integrate security review into every phase; 18

Continuous Feedback Loop (09, 13, 15, 22, 35)

Enhance visibility; 41

Privileged access management; 30

Secure Access via secrets management; 41

Common weaknesses enumeration (CWE); 08

impose policy and governance 41;

Implement security policies; 30

Business-driven security; 24

Keep credentials safe; 06

Operational controls validation and improvement; 14

Use Software Composition Analysis (SCA) and Governance; 06

Software composition analysis; 23

Automation 02, 04, Use tools and automation 06, Automate protection of business logic flaws 19, Automated code review 23,

Automate as much as possible 25,28, Automate tools and security processes 30, Automate security tests 35, Automated security testing 08,11,13,15,35, Use automated security tools 41, Automate security processes 17

Threat modeling 06, 10, 14, 25, 28.

Monitoring 02; continuous monitoring 06; 24x7 proactive monitoring 24; monitor and scale 25; own the security monitoring 31;

Source code repository; 10

Configuration management; 10

Host hardening; 10

Application-level assessment; 10

CI/CD for patching; 10

Secure coding practices; 10

Source code scanning; 10

Secure coding; 14

Implement secure coding practice; 28

Conduct code dependency checks regularly; 25

Build preapproved code; 18

Container security-Docker (Minimize container images, Limit user privilege, Sign and verify container images, Regularly monitor images for open source vulnerabilities, Protect images from information leakage, Use fixed tags for immutability, Use COPY instead of ADD, Use labels for metadata, Use multi-stage builds to minimize image size, Use a linter);10

Orchestration (Kubernetes); 10

Leverage containerization; 28

Harden the container; 41

Secure deployment and operations to ensure that cloud platform, runtimes and applications are deployed securely, checked regularly for security configuration and hygiene, tested for security vulnerabilities, and updated with software patches and security fixes; 14

Verify cloud infrastructure; 28

Consumable security services with APIs; 24

Red and Blue Team exploit testing; 24

Integrate security issues within your general bug tracker; 19

Linear scalability and affordable cost; 19

Test for security; 18

Security testing; 14

Sensitive information scan; 23

Static Application Security Testing (SAST); 02, 08, 23, 25

Dynamic Application Security Testing (DAST); 02, 08, 23, 25

Runtime Application Self-Protection (RASP); 02, 08, 25

Interactive Application Security Testing (IAST); 02, 08, 19, 25

Security-as-Code; 32

Compliance-as-Code; 23

Policy-as-Code; 17