Review Article

Features of the organization of the software development process: analysis of the effectiveness of the implementation of DevSecOps practices

ABSTRACT

Aims: DevOps generate a revolution in the software development lifecycle by providing agile and fast-paced solutions. Since DevOps focuses only on increasing development and development speed, it ignores security prospects. DevSecOp is a notion of implanting security into DevOps operation without distressing its agile nature by discovering contemporary security practices. This research aims to reveal a comprehensive overview of DevSecOp.
Methodology: Here we will present a brief overview of the research methodology. Afterward, it presented the method of gathering the required information. This research paper is distributed in the following section. Section II will present the research methodology, while section III will provide results from this study. In the end, we conclude our research

Results: In this study, we discover essential DevSecOp concepts, leverages of DevSecOp and potential research challenges in implementing it. We used a Multivocal literature review to explore the aforementioned subjects. For this Multivocal literature review, we searched grey data and, after processed that data, found answers to our research questions. This review concluded that DevSecOp, although challenging to implement, can be a constructive addition to the DevOps paradigm.

Conclusion: DevSecOp is a relatively new concept that is not even fully concise in its name and definition. The key idea of DevSecOp is to implant security into DevOps procedures to make them more secure. We presented MLR on DevSecOp, keeping in mind pre-designed research questions. Since DevSecOp is not as popular and does not contain enough academic literature, we had to include grey data for our literature review. This MLR concluded that DevSecOp is mostly defined as integrating security into DevOps

Keywords: DevSecOp, security, DevOps, Multivocal Literature Review.

1. INTRODUCTION

Software development lifecycle has experienced drastically changed in the past decade. Numerous companies prefer to develop software as a product as it can be delivered to the end-user and run locally hence helping to use the software as a product or service. In SaaS (Software as a Service) scenario, the software is developed in the cloud environment and

delivered to the user by a web browser[1]. Services are provided by subscriptions and licensing[2]. In a SaaS environment, users cannot control basic infrastructure and activation of applications. In this way, the software provider does not need to deliver updates in software to all users. Rather they need to update their software, and all users automatically get an updated version. Continuous Integration(CI) is a complicated process in which software is uninterruptedly unified while distributed to the users. Continuous delivery refers to the delivery of software developed by different developers and bug fixing[3]. Continuous Delivery (CD) refers to the deployment of software in a production environment that is a different process than traditional deployment. It can be repeated even two or three times a day. Continuous delivery helps the business cycle process, which helps in continuous feedback from the end-user and reduces the risk of deployment cost[4].

DevOps is described as requirements of progress and operations and an imaginary system of technologies and teams[5]. The main purpose of DevOps was to synchronize operational and progress teams to work collaboratively in software development and deployment in the production process[6]. Due to the massive popularity of DevOps, numerous organizations are adopting its associated practices. Still, organizations seldom adopt DevOps security as its part. According to Gartner[8], only 20% of organizations implemented security steps in their DevOps process[9]. Most developers and management consider security options a barrier to the speed of CI and CD operations[10].

DevSecOp fulfills the necessity of security. DevSecOp is an effort to implement security techniques in modern DevOps operations without affecting its speed and efficiency. The main purpose of DevSecOp is to make a collaborative effort with security teams and DevOps professionals, increasing the main purpose of DevOps[11].

Since DevSecOp is a new trend, a comprehensive analysis of its methods and preferences is needed. Although there is not much literature review available on the DevSecOp paradigm, from its current literature review, it can be concluded that: it is a practice with collaborations if DevOps experts and staff are available on the internet to explore their practices[12]. In [14], the author presents a systematic mapping study on DevSecOp and comprehensively covers all research work on DevSecOp. In the result, the author presented a review on CI/CD techniques and the use of security in it[13]. In [14], the author collected literature related to CD and described the challenges and leverages of CD. In [15], the author presented different use cases of CI. None of the aforementioned studies comprehensively describe DevSecOp and for what purpose it is used. There is no single search work on DevSecOp that comprehensively describes its nature and its presented literature review to the author's best knowledge.

This research paper is distributed in the following section. Section II will present the research methodology, while section III will provide results from this study. In the end, we conclude our research.

2. METHODOLOGY

Here we will present a brief overview of the research methodology. Afterward, it presented the method of gathering the required information.

2.1 Multivocal literature review

As discussed in the introduction section, no solid framework exists to gather literature on DevSecOp comprehensively. Hence we choose Multivocal Literature Review (MRL) for our

research. A Multivocal literature review gathers data from all kinds of literature, including research papers, white papers, blogs, and articles. Although the voice and tone are different in all aforementioned literature, it leverages the collection of the opinion of researchers, practitioners, and all other experts on the chosen topic. Since MLR is a relatively less common review process, there exists some MLR on the following topics. In [17], the author presented MLR on software automated testing and practitioners' opinions about software testing. The MLR of [18] is related to DevOps practice and development. In [19], the author presented MLR on maturity assessment and practices. Hence to our best knowledge, there is no MLR in DevSecOp; this is not the first work in DevOps but the first one in DevSecOp.

2.2 Research questions

Since the main purpose of this MLR is to explore the basic concepts of DevSecOp, the challenges faced by DevSecOp and these challenges can be cooped. Following research questions are being formed to approach the goal of this research
Research Question1 (RQ1): How can DevSecOp be defined according to existing literature?
Research Question2 (RQ2): What are the major features of DevSecOp?
Research Question3 (RQ3): The key benefits and potential challenges of adopting DevSecOp.
Research Question4(RQ4): Evaluation of DevSecOp.

2.3 Study procedure:

This section will describe the study procedure or, in other words, the study method. This protocol describes how we find our targeted literature and which data sources are used for finding literature and inclusion and exclusion criteria. The last study protocol will discuss the process of cataloging literature. Databases, Table 1 shows databases used for search

Table 1. Databases used for collecting data

Data source	URL	Type of literature
Google search engine	www.google.com	Grey data. i.e., technical article, BlogSpot white paper.
Google Scholar	www.scholar.google.com	Academic literature that includes conference proceedings, journal articles

Although more precise data sources are available such as the IEEE digital library, Springer Link, and web of science, this topic is very new and very rare research papers are available; hence we use Google scholar only.

Search Terms:
As mentioned earlier, DevSecOp is a relatively new term; it developed by integrating "SECurity" in the current term of "DEVelopment" and "OPeration.". there does not exist any particular order in a combination of the aforementioned terms; hence we had to define our query string by combining all of the following terms

("DevSecOps" OR "SecDevOps" OR "DevOpsSec") AND("definition" OR "characteristics" OR "challenges" OR "benefits"OR "evolution").

Study selection:
After preliminary findings, we formed an inclusion/exclusion criteria to filter the initial result and find the best-matched literature.

Inclusion criteria:

- Research Paper that is published in any conference or general or symposium
- Research Papers and grey literature that is focused on the domain of DevSecOp
- All literature related to our research questions, such as The introduction of DevSecOp, advantages and applications.
- We include only the literature that was published after 2016.

Exclusion criteria:

- We exclude all research papers and grey material that can not be accessed legally—for instance, non-open access journal articles.
- Literature that was not available in a language other than English
- Marketing and advertising data were also excluded.

Search procedure:
The search procedure is illustrated in Fig 1. In the first step, both data sources are queried by pre-defined search terms. Since there exists four RQ,s hence different term was placed for different RQ in focus. The initial query string was distributed into five different parts. The initial finding was reviewed according to the given procedure. Only title and abstract was observed for academic research, while grey data was observed by title, metadata and bird's eye view. The result was filtered out after the inclusion/exclusion criteria were read fully and carefully, leading to the preliminary study.

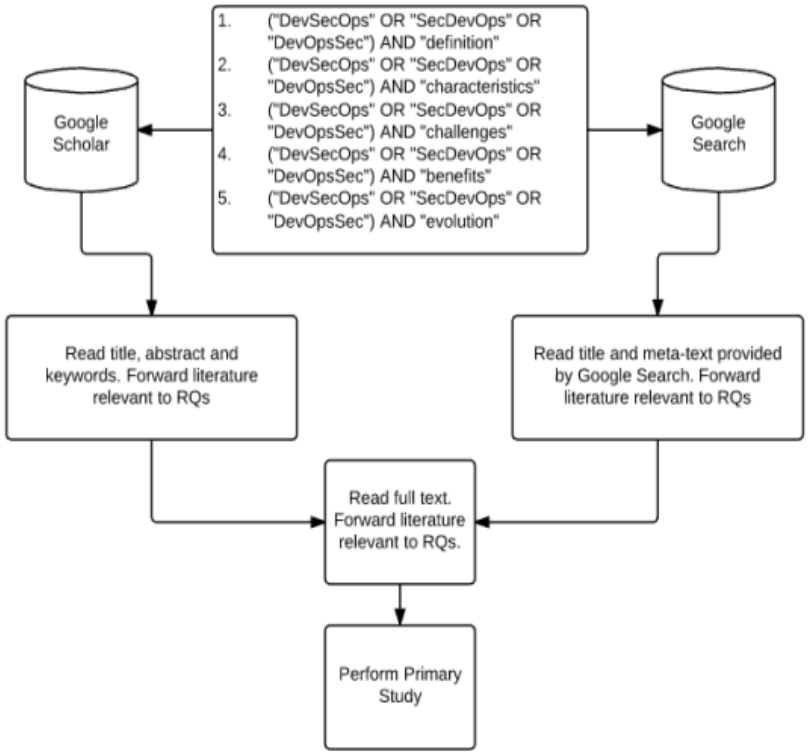

Fig 1: MLR procedure overview

3. RESULTS AND DISCUSSION

The following section will discuss the result by executing our research in conjunction with our research question.

Table 2. Summary of primary search result

Search engine	Initial result	Title, abstract and meta text	Full text
Google Scholar	34	4	2
Google search	250	62	50

3.1 RQ 1: Definition of DevSecOp

As mentioned in section 2, our first and foremost research question was to investigate the proper definition of DevSecOp. After carefully reviewing the selected literature, we conclude that no hard and fast definition of DevSecOp is presented in any literature[20]. DevSecOp can be defined as the "integration of DevOps with security protocols to enhance its security requirement without decreasing its speed." The targeted DevSecOp procedure can be possible only by the collaborative effort of development teams, security specialists, and operational teams.

3.2 RQ2: characteristics of DevSecOp:

Our second research question was related to investigating the basic characteristics of DevSecOp. Literature review leads us to characterize DevSecOp by its basic principles and best practices. Basic principles deal with finding reasoning to implement DevSecOp[21]. In other words, practices are best to implement security protocols with DevOps while keeping its speed unaffected. We find the following principles and reasoning to answer the characteristics of DevSecOp.

3.2.1 Principles:

Principles of DevSecOp are inherited from the basic characteristics of DevOps, with the addition of security in each characteristic. As mentioned earlier, the basic principles of DevOps in use are also called CAMS[22]. Following, we will present each principle in conjunction with security requirements that forms DevSeOp's basic structure.

Culture

When discussing the culture of DevOps, it is very easy to understand that DevOps culture consists of the collaborative effort of the development team and operational teams [23]. Both teams work together for a united goal of delivering a successful end product to the end user [24]. While discussing the culture of DevSecOp, the security team also participates with the development and operational teams. The security team works to implement security protocols in both development and operational procedures [25]. Here, one of the challenging parts of the security team is to implant security without affecting their speed[26].

Automation:

In particular, DevOps environment automation is a key feature that ensures rapid development and deployment. With automation, timely feedback from the end-user can be possible [27]. In DevSecOp automation principle is altered by embedding security automation without decelerating the existing automation process. Since it can be understood that the automation process needs to be as fast as possible and can be bearded any overhead, security teams have to do special care in that phase. Implementing security in the DevOps procedure should not become friction [28].

Measurements:

In DevOps, paradigm measurements refer to monitoring particular business metrics. These metrics can span from key level performance indicators, and these indicators can be used to measure the need for a new release and the effect on an existing one. DevSecOp measurement identifies threats, risks, and vulnerabilities attached to the DevOps procedure. Measurements in DevSecOp should be designed not to slow down operation efficiency, deployment, or development.

Sharing:

In the DevOps environment, all the key stockholders of development and operation teams share their knowledge and experiences needed in the operational and development process. DevSecOp is an augmentation of that sharing with security teams[29]. Security teams should share their security practices with other members to focus their minds on the security perspective of deployment and development[30].

3.2.2 Practices:

From our literature review, our findings related to best practices in DevSecOps are the following:

Threat modeling:

Threat modeling can also be assumed as risk assessment is an essential DevSecOps practice[31]. Threat assessment and modeling deal with organizations' practice of designing and developing security measures to encounter potential threats and vulneratabilities[32]. Risk assessment should be designed before time; in other words, during each development and deployment phase, security persons should prepare a risk analysis and assessment report[33]. Threat modeling can be considered a way of documenting threats during the development phase[34].

Continuous testing:

As its name suggests, continuous testing is the practice in the DevSecOps environment to continuously test security measures during each development and design life cycle phase[35]. Continuously testing the potential threat and anomalies can help remove anomalies[36].

Monitoring and logging:

After routine security testing, it is best to monitor those security standards continuously. Since monitoring can be the best option for finding the success of these security measurements, these security measurements can be updated[38].

Security as Code:

Security as code refers to the design of security policies implemented as part of the development code. It can also be a good practice to write a script based on suggested security steps, and this script can be run from the start of the execution of code.

3.2.3 RQ3: Advantageous of DevSecOps

Our research question 3 is related to finding the advantages of DevSecOp in the light of selected literature. From our literature review, we find the following benefits of DevSecOps and its practices:

Security on the left:

As discussed earlier, the primary goal of DevSecOp is to involve security in DevSec operations. From our literature review, we find that the key benefit of DevSecOp is that they involve security practitioners throughout the deployment and development process, hence ensuring the security of the complete process.

Automating security:

Since DevSecOps promises to optimize security controls fast, scalable and fully controlled, this is advantageous for DevSecOp. Potential threats and vulnerabilities can be automatically copied and mitigated in time[37]. With the help of DevSecOp, risk can be confined to its lowest level. Furthermore, the analysis of risk help to understand the cause of risk and vulnerabilities.

Values:
In [39], the author describes that if security in a DevOps environment is ignored, it can cause potential problems. The whole process can be made more secure and function more efficiently with the involvement.

3.2.4 RQ 3 Challenges in DevSecOps:

Since our research question, three continued two-part; the first part has been described in the previous section, and the next part, that is, challenges of DevSecOp, will present in this section. From our literature review, we find the following potential challenges in implementing of DevSecOps

Merging with DevOps:

One major challenge of DevSecOp is integrating existing security technologies into DevOp. Along with this, these security methods are sometimes overhead in the development process. DevSecOps security experts should adopt agile and fast-paced security techniques so that they do not hinder existing DevOps operations[40].

Organizational:

To implement DeveSecOps in any organization, this organization needs to adopt new skills, changes, new culture, cutting edge tools and technologies, required processes and policies and DevSecOps practices[41]. A new skill is needed in the area of cryptography. That is assumed to be beyond existing DevOps skills. The developers and the manager can suffer from frustration caused by adopting new security practices[42]. Another organizational challenge in shifting to DevSecOps is that security teams must learn development practices[43]. Most organizations assume security as costly activity, yet they have to realize that their cost can be reduced by avoiding threats by implanting security practices.

Tools and practices:

in the existing DevOps environment, all tools are designed to achieve speed, while on the other hand, security tools are made by keeping in mind the security requirements[44]. Hence, there is a need to develop new tools that fulfill both demands, as mentioned earlier.

3.2.5 RQ 4: history of DevSecOps:

Our research question 4 was related to the evaluation or history of DevSecOps. From our literature review, we find that concept of DevSecOps was first discussed by Gartner s analyst Neil Mcdonald in his blogpost "DevOps needs to become DevOpSec" in 2012[45]. Since the birth of this concept, its popularity is gradually increasing. Table 3 discusses the number of publications in the domain of DevSecOps

Table 3. Number of research publications of DevSecOp per year

Years	Number of research publications
2014	2
2015	8
2016	27
2017	46
2018	107
2019	238
2020	409
2021	573

Limitation of results

In this section, we will fairly discuss the limitation of our results. Since this research presented Multivocal Research, the authenticity of the literature did not keep in mind. For instance, it is not noticed whether either research paper is taken from peer-reviewed journals or not in academic literature. Another limitation of our research is that DevSecOp is relatively a new term, and there are no final consciences of this term. Different authors used various terms such as DevOps, SecDevOps, DevSecOps, Secure DevOps and Rugged DevOps.

4. CONCLUSION

DevSecOp is a relatively new concept that is not even fully concise in its name and definition. The key idea of DevSecOp is to implant security into DevOps procedures to make them more secure. We presented MLR on DevSecOp, keeping in mind pre-designed research questions. Since DevSecOp is not as popular and does not contain enough academic literature, we had to include grey data for our literature review. This MLR concluded that DevSecOp is mostly defined as integrating security into DevOps. We found numerous challenges in its implementation, such as organizational challenges, it is merging with DevOps, and the need to develop new security tools for its proper working. Our literature review explored plenty of advantages that can make DevSecOp an emerging field of the future.

CONSENT (WHERE EVER APPLICABLE)

All author declare that 'written informed consent was obtained from the patient (or other approved parties) for publication of this case report and accompanying images. A copy of the written consent is available for review by the Editorial office/Chief Editor/Editorial Board members of this journal.

ETHICAL APPROVAL (WHERE EVER APPLICABLE)

All author hereby declare that all experiments have been examined and approved by the appropriate ethics committee and have therefore been performed in accordance with the ethical standards laid down in the 1964 Declaration of Helsinki."

REFERENCES

1. Mell PM, Grance T. The nist definition of cloud computing. Special Publications; (NIST SP)-800-145, (7), 9 2011. NIST Definitions on Cloud Computing.
2. Fitzgerald B, Stol K-J. Continuous software engineering: A roadmap and agenda. J Syst Softw [Internet]. 2017;123:176–89. Available from: http://dx.doi.org/10.1016/j.jss.2015.06.063
3. Claps GG, Berntsson Svensson R, Aurum A. On the journey to continuous deployment: Technical and social challenges along the way. Inf Softw Technol [Internet]. 2015;57:21–31. Available from: http://dx.doi.org/10.1016/j.infsof.2014.07.009
4. Humble J, Joanne M. Why enterprises must adopt devops to enable continuous delivery. The Journal of Information Technology Management. 2011;(24): 7

5. Ebert C, Gallardo G, Hernantes J, Serrano N. DevOps. IEEE Softw [Internet]. 2016;33(3):94–100. Available from: http://dx.doi.org/10.1109/ms.2016.68

6. Cois CA, Yankel J, Connell A. Modern DevOps: Optimizing software development through effective system interactions. In: 2014 IEEE International Professional Communication Conference (IPCC). IEEE; 2014.

7. Callanan M, Spillane A. DevOps: Making it easy to do the right thing. IEEE Softw [Internet]. 2016;33(3):53–9. Available from: http://dx.doi.org/10.1109/ms.2016.66

8. Spinellis D. Being a DevOps Developer. IEEE Softw [Internet]. 2016;33(3):4–5. Available from: http://dx.doi.org/10.1109/ms.2016.76

9. Hewlett Packard Enterprise. Application security and devops. Technical report, Hewlett Packard Enterprise. 2016.

10. MacDonald N, Head I. DevSecOps: How to Seamlessly Integrate Security Into DevOps. Technical report, Gartner. 2016.

11. Mohan V, Othmane LB. SecDevOps: Is it a marketing buzzword? - mapping research on security in DevOps. In: 2016 11th International Conference on Availability, Reliability and Security (ARES). IEEE; 2016.

12. Ur Rahman AA, Williams L. Software security in DevOps: Synthesizing practitioners' perceptions and practices. In: Proceedings of the International Workshop on Continuous Software Evolution and Delivery. New York, NY, USA: ACM; 2016.

13. Karvonen T, Behutiye W, Oivo M, Kuvaja P. Systematic literature review on the impacts of agile release engineering practices. Inf Softw Technol [Internet]. 2017;86:87–100. Available from: http://dx.doi.org/10.1016/j.infsof.2017.01.009

14. Rodríguez P, Haghighatkhah A, Lwakatare LE, Teppola S, Suomalainen T, Eskeli J, et al. Continuous deployment of software intensive products and services: A systematic mapping study. J Syst Softw [Internet]. 2017;123:263–91. Available from: http://dx.doi.org/10.1016/j.jss.2015.12.015

15. Ståhl D, Bosch J. Modeling continuous integration practice differences in industry software development. J Syst Softw [Internet]. 2014;87:48–59. Available from: http://dx.doi.org/10.1016/j.jss.2013.08.032

16. Ogawa RT, Malen B. Towards rigor in reviews of multivocal literatures: Applying the exploratory case study method. Rev Educ Res [Internet]. 1991;61(3):265–86. Available from: http://dx.doi.org/10.3102/00346543061003265

17. Garousi V, Mäntylä MV. When and what to automate in software testing? A multi-vocal literature review. Inf Softw Technol [Internet]. 2016;76:92–117. Available from: http://dx.doi.org/10.1016/j.infsof.2016.04.015

18. de França BBN, Jeronimo H Junior, Travassos GH. Characterizing DevOps by hearing multiple voices. In: Proceedings of the 30th Brazilian Symposium on Software Engineering - SBES '16. New York, New York, USA: ACM Press; [Internet]. 2016. Available from: https://doi.org/10.1145/2973839.2973845

19. Garousi V, Felderer M, Hacaloğlu T. Software test maturity assessment and test process improvement: A multivocal literature review. Inf Softw Technol [Internet]. 2017;85:16–42. Available from: http://dx.doi.org/10.1016/j.infsof.2017.01.001

20. Garousi V, Felderer M, Mäntylä MV. The need for multivocal literature reviews in software engineering: Complementing systematic literature reviews with grey literature. In: Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering. New York, NY, USA: ACM; 2016.

21. Shackleford D. A devsecops playbook. SANS Institute InfoSec Reading Room, 3

[Internet]. 2016. A DevSecOps playbook. Available from: https://pages.cloudpassage.com/rs/857-FXQ-213/images/sans-a-devsecops-playbook.pdf
22. 4 tips for integrating security into your DevOps practices [Internet]. Fastly.com. [cited 2022 Jun 8]. Available from: https://www.fastly.com/blog/4-tips-for-integrating-security-into-your-devops-practices
23. Lietz S. <— shifting security to the left — [Internet]. DevSecOps. 2016 [cited 2022 Jun 8]. Available from: https://www.devsecops.org/blog/2016/5/20/-security
24. Bledsoe G. Getting to DevSecOps: 5 best practices for integrating security into your DevOps [Internet]. TechBeacon. 2016 [cited 2022 Jun 8]. Available from: https://techbeacon.com/app-dev-testing/getting-devsecops-5-best-practices-integrating-security-your-devops

25. Lim F. DevSecOps is the Krav Maga of Security — [Internet]. DevSecOps. 2016 [cited 2022 Jun 8]. Available from: https://www.devsecops.org/blog/2016/9/8/devsecops-is-the-krav-maga-of-security
26. Lietz S. Principles of DevSecOps — [Internet]. DevSecOps. 2015 [cited 2022 Jun 8]. Available from: https://www.devsecops.org/blog/2015/2/21/principles-of-devsecops
27. Greene T. What security teams need to know about DevOps [Internet]. Securityweek.com. [cited 2022 Jun 8]. Available from: https://www.securityweek.com/what-security-teams-need-know-about-devops
28. Jennings R, Park B, Jeffers J, Ranganathan K, Shalom N, Pontarelli B, et al. Security breaks DevOps – here's how to fix it [Internet]. DevOps.com. 2015 [cited 2022 Jun 8]. Available from: https://devops.com/security-breaks-devops-heres-how-to-fix-it/

29. Jennings R, Park B, Jeffers J, Ranganathan K, Shalom N, Pontarelli B, et al. The DevSecOps approach to securing your code and your cloud [Internet]. DevOps.com. 2017 [cited 2022 Jun 8]. Available from: https://devops.com/downloads/devsecops-approach-securing-code-cloud/

30. Ibrahim A, Yousef AH, Medhat W. DevSecOps: A security model for infrastructure as code over the cloud. In: 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC). IEEE; 2022.

31. DevOps invites security to "join the party" [Internet]. White Hat Security. [cited 2022 Jun 8]. Available from: https://www.whitehatsec.com/trending/content/devops-invites-security-join-party

32. Hornbeek M. DevOps makes security assurance affordable [Internet]. DevOps.com. 2015 [cited 2022 Jun 8]. Available from: https://devops.com/devops-makes-security-assurance-affordable/

33. Anderson E. How to build an effective DevSecOps culture [Internet]. The GitHub Blog. GitHub; 2020 [cited 2022 Jun 8]. Available from: https://github.blog/2020-04-28-how-to-build-an-effective-devsecops-culture/

34. Romeo C. The 3 most crucial security behaviors in DevSecOps [Internet]. TechBeacon. 2017 [cited 2022 Jun 8]. Available from: https://techbeacon.com/app-dev-testing/3-most-crucial-security-behaviors-devsecops

35. Cureton A. Building Security into DevOps: Is DevSecOps the beginning of the future? [Internet]. Linkedin.com. LinkedIn; 2017 [cited 2022 Jun 8]. Available from: https://www.linkedin.com/pulse/building-security-devops-devsecops-beginning-future-andy-cureton

36. McKay J, SVP, Chief Technology Officer, Logicworks. How to use DevSecOps to smooth cloud deployment [Internet]. Network World. 2016 [cited 2022 Jun 8]. Available from: https://www.networkworld.com/article/3041640/how-to-use-devsecops-to-smooth-cloud-deployment.html

37. Amazon Web Services. Introduction to devsecops on aws. Amazon.com. [cited 2022 Jun 8]. Available from: https://aws.amazon.com/about-aws/whats-new/2022/04/devsecops-solutions-competency-partners/

38. Francis R. 7 ways DevOps benefits CISOs and their security programs [Internet]. CSO Online. 2016 [cited 2022 Jun 8]. Available from: https://www.csoonline.com/article/3125604/7-ways-devops-benefits-cisos-and-their-security-programs.html

39. 9 Ways DevOps and Automation Bolster Security and compliance [Internet]. CloudBees. [cited 2022 Jun 8]. Available from: https://www.cloudbees.com/whitepapers/9-ways-devops-automation-bolster-security-compliance

40. WhiteSource. 7 essential steps to DevSecOps success - WhiteSource [Internet]. Medium. 2016 [cited 2022 Jun 8]. Available from: https://medium.com/@WhiteSourceSoft/7-essential-steps-to-devsecops-success-a6d7a12537c3

41. Eldridge I. SecDevOps: Injecting security into DevOps processes [Internet]. New Relic. 2018 [cited 2022 Jun 8]. Available from: https://newrelic.com/blog/best-practices/what-is-secdevops

42. Rohr M. Agile security & SecDevOps touch points [Internet]. Pragmatic Application Security. 2017 [cited 2022 Jun 8]. Available from: https://blog.secodis.com/2017/01/02/pillars-agile-security-secdevops/

43. Myrbakken H, Colomo-Palacios R. DevSecOps: A Multivocal Literature Review. In: Communications in Computer and Information Science. Cham: Springer International Publishing; 2017. p. 17–29.

44. DevOps security challenges and how to overcome them [Internet]. CCSI. 2019 [cited 2022 Jun 8]. Available from: https://www.ccsinet.com/blog/devops-security-challenges/

45. O'Connor RV, Elger P, Clarke PM. Continuous software engineering-A microservices architecture perspective. J Softw (Malden) [Internet]. 2017;29(11):e1866. Available from: http://dx.doi.org/10.1002/smr.1866