Review Article

Multivocal Literature Review on the Security of DevSecOp

ABSTRACT

DevOps revolutionize the software development lifecycle by providing agile and fast-paced solutions. DevOps ignores security prospects since it focuses only on increasing development and speed. 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. Here we presented 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 presented the research methodology, while section III provided results from this study. In the end, we conclude our research

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.

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 mainly 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, 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. Services are provided by subscriptions and licensing [1, 2]. In a SaaS environment, users cannot control basic infrastructure and activation of applications[3]. In this way, the software provider does not need to deliver updates in software to all users. Instead, 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[4]. 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 cos[5].
DevOps is described as requirements of progress and operations and an imaginary system of technologies and teams[6]. The primary purpose of DevOps was to synchronize operational and progress teams to work collaboratively in software development and deployment in the production process[1, 3]. 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[7], only 20% of organizations implemented security steps in their DevOps process[9]. Security and privacy are critical aspects in almost every technological paradigm [8-12]. Most developers and management consider security options a barrier to the speed of CI and CD operations[9].
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 primary purpose of DevSecOp is to collaborate with security teams and DevOps professionals, increasing the primary purpose of DevOps[13].
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[14]. In [15], 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[16]. The author collected CD literature and described CD's challenges and leverages. In [17], the author presented different use cases of CI. None of those mentioned above 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. Our research aims to fill the study gap by providing a multivocal literature review. Our planned multivocal literature review is comprehensive enough to cover technical detail of all aspects of the security of DevSecOp hence investigating the areas of research that need attention. The results of our study provided a cushion on Culture, Automation, Sharing, and Measurement, which are the basic principles of DevSecOp.
This research paper is distributed in the following section. Section II presented the research methodology, while section III provided results from this study. In the end, we conclude our research.

2. METHODOLOGY

Here we presented 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[18]. Although the voice and tone are

different in all literature as mentioned above, 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 [19], the author presented MLR on software automated testing and practitioners' opinions about software testing. The MLR is related to DevOps practice and development. In [20], the author presented MLR on maturit[21]y 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 primary 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 significant 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 described the study procedure or, in other words, the study method. This protocol describes how we find our targeted literature, which data sources are used for finding literature, and inclusion and exclusion criteria. The last study protocol discussed 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 such as the IEEE digital library, Springer Link, and web of science, this topic is very new, and scarce 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 terms as mentioned earlier; 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 four RQ,s hence different terms were 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 thoroughly and carefully, leading to the preliminary study.

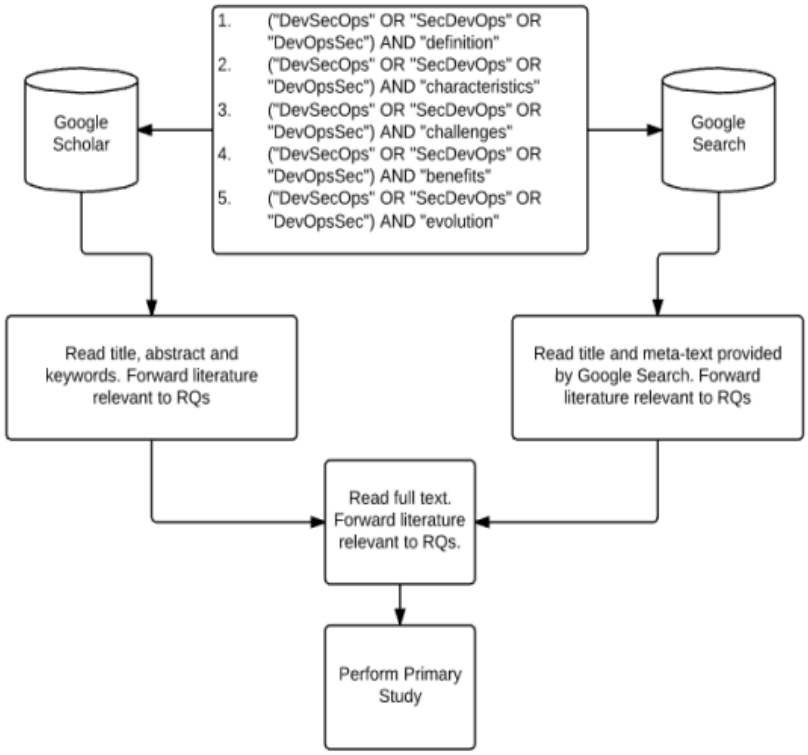

Fig 1: MLR procedure overview

3. RESULTS AND DISCUSSION

The following section discussed 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	1580	372	2
Google search	30100	653	50

3.1 RQ 1: Definition of DevSecOp

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[13, 22]. 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 only be possible through the collaboration of development teams, security specialists, and operational teams.

3.2 RQ2: characteristics of DevSecOp:

Our second research question was related to investigating the essential characteristics of DevSecOp. The literature review characterizes DevSecOp by its basic principles and best practices. Basic principles deal with finding reasoning to implement DevSecOp[23]. In other words, practices best 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 essential characteristics of DevOps, with the addition of security in each characteristic. As mentioned earlier, the basic principles of DevOps are also called CAMS[24]. Following, we presented each principle in conjunction with security requirements that forms DevSeOp's basic structure.

Culture

When discussing the culture of DevOps, it is straightforward to understand that DevOps culture consists of the collaborative effort of the development team and operational teams. Both teams work together for a united goal of delivering a successful end product to the end-user. 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 operation. Here, one of the challenging parts of the security team is to implant security without affecting their speed[15, 25].

Automation:

In particular, DevOps environment automation is a critical feature that ensures rapid development and deployment. With automation, timely feedback from the end-user can be possible [26]. 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 [27].

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 critical 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[28]. Security teams should share their security practices with other members to focus their minds on the security perspective of deployment and development.

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[29]. Threat assessment and modeling deal with organizations' practice of designing and developing security measures to encounter potential threats and vulneratabilities[30]. 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. Threat modeling can be considered a way of documenting threats during the development phase[17].

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 [31]. 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[32].

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, which can be run from the start of executing 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, ensuring the complete process's security.

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[33]. 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:

The author describes that if security in a DevOps environment is ignored, it can cause potential problems. The involvement can make the whole process more secure and function more efficiently.

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, challenges of DevSecOp, presented 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. These security methods are also 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[34].

Organizational:

To implement DeveSecOps in any organization, this organization needs to adopt new skills, changes, culture, cutting-edge tools and technologies, required processes and policies, and DevSecOps practices[35]. 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[36]. Another organizational challenge in shifting to DevSecOps is that security teams must learn development practices[37]. Most organizations assume security as costly activity, yet they must 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, as mentioned earlier, there is a need to develop new tools that fulfill both demands.

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[6]. 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

Table 3 shows the result of the google scholar search result related to DevSecOps year-wise. From this table, it is easy to understand that research in the area of DevSecOps is growing proportionally. This propositional trend indicates the future potential growth of this domain.

Limitation of results

In this section, we reasonably discuss the limitation of our results. Since this research presented Multivocal Research, the literature's authenticity was not kept 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

new, 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 mainly defined as integrating Security into DevOps. We found numerous challenges in its implementation, such as organizational challenges, 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.

ETHICAL APPROVAL

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

REFERENCES

[1] N. Mazher, F. Riaz, M. Tariq, F. Ishaque, and H. S. Shakir, "Performance Comparison of Load Balancing Algorithms using Cloud Analyst in Cloud Computing," 2018.

[2] A. Ahmed, "DevSecOps: Enabling Security by Design in Rapid Software Development," 2019.

[3] J. Asad and N. Mazher, "Load Balancing Protocol for dynamic resource allocation in cloud computing," 2018.

[4] M. Sánchez-Gordón and R. Colomo-Palacios, "Security as culture: a systematic literature review of DevSecOps," in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, 2020, pp. 266-269.

[5] J. C. Akujobi, "A Model For Measuring Improvement Of Security In Continuous Integration pipelines: Metrics and Four-Axis Maturity Driven DevSecOps (MFAM)," University of Twente, 2021.

[6] A. Koskinen, "DevSecOps: building security into the core of DevOps," 2019.

[7] J. Brodkin, "Gartner: Seven cloud-computing security risks," ed, 2008.

[8] M. Ahmadi, "Hidden fear: Evaluating the effectiveness of messages on social media," Arizona State University, 2020.

[9] M. Ahmadi and B. S. Ghahfarokhi, "Preserving privacy in location based mobile coupon systems using anonymous authentication scheme," in *2016 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC)*, 2016: IEEE, pp. 60-65.

[10] P. Kiaei, C.-B. Breunesse, M. Ahmadi, P. Schaumont, and J. Van Woudenberg, "Rewrite to reinforce: Rewriting the binary to apply countermeasures against fault injection," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*, 2021: IEEE, pp. 319-324.

[11] M. Ahmadi, K. Leach, R. Dougherty, S. Forrest, and W. Weimer, "Mimosa: Reducing malware analysis overhead with coverings," *arXiv preprint arXiv:2101.07328,* 2021.
[12] M. Ahmadi, P. Kiaei, and N. Emamdoost, "SN4KE: Practical Mutation Testing at Binary Level," *arXiv preprint arXiv:2102.05709,* 2021.
[13] P. Bitra and C. S. Achanta, "Development and Evaluation of an Artefact Model to Support Security Compliance for DevSecOps," ed, 2021.
[14] R. Kumar and R. Goyal, "When Security Meets Velocity: Modeling Continuous Security for Cloud Applications using DevSecOps," in *Innovative Data Communication Technologies and Application*: Springer, 2021, pp. 415-432.
[15] W. F. Santoso and D. S. S. Sahid, "Implementation And Performance Analysis Development Security Operations (DevSecOps) using Static Analysis and Security Testing (SAST)," *International ABEC,* pp. 17-19, 2021.
[16] M. Zhou, "Towards a poetics of immersion in lyric translation: Aesthetic illusion and the translator as immersive reader in English translations of classical Chinese ci poetry," *Target. International Journal of Translation Studies,* vol. 30, no. 3, pp. 383-407, 2018.
[17] R. Kumar and R. Goyal, "Modeling continuous security: A conceptual model for automated DevSecOps using open-source software over cloud (ADOC)," *Computers & Security,* vol. 97, p. 101967, 2020.
[18] B.-J. Butijn, D. A. Tamburri, and W.-J. v. d. Heuvel, "Blockchains: a systematic multivocal literature review," *ACM Computing Surveys (CSUR),* vol. 53, no. 3, pp. 1-37, 2020.
[19] N. Mazher and I. Ashraf, "A Survey on data security models in cloud computing," *International Journal of Engineering Research and Applications (IJERA),* vol. 3, no. 6, pp. 413-417, 2013.
[20] A. Pereira-Vale, E. B. Fernandez, R. Monge, H. Astudillo, and G. Márquez, "Security in microservice-based systems: A multivocal literature review," *Computers & Security,* vol. 103, p. 102200, 2021.
[21] J. Scheuner and P. Leitner, "Function-as-a-service performance evaluation: A multivocal literature review," *Journal of Systems and Software,* vol. 170, p. 110708, 2020.
[22] Z. Ahmed and S. C. Francis, "Integrating security with devsecops: Techniques and challenges," in *2019 International Conference on Digitization (ICD)*, 2019: IEEE, pp. 178-182.
[23] S. A. Shah and N. Mazher, "A review on security on internet of things," in *November 2018 Conference: 1st International Multi-Disciplinary Research Conference (IMDRC 2017)*.
[24] A. Ibrahim, A. H. Yousef, and W. Medhat, "DevSecOps: A Security Model for Infrastructure as Code Over the Cloud," in *2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, 2022: IEEE, pp. 284-288.
[25] B. Yadav, G. Choudhary, S. K. Shandilya, and N. Dragoni, "AI Empowered DevSecOps Security for Next Generation Development," in *International Conference on Frontiers in Software Engineering*, 2021: Springer, pp. 32-46.
[26] X. Sun, Y. Cheng, X. Qu, and H. Li, "Design and Implementation of Security Test Pipeline based on DevSecOps," in *2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2021, vol. 4: IEEE, pp. 532-535.
[27] I. Saidani, A. Ouni, M. W. Mkaouer, and F. Palomba, "On the impact of continuous integration on refactoring practice: An exploratory study on travistorrent," *Information and Software Technology,* vol. 138, p. 106618, 2021.
[28] I. R. a. Kelley, "Data management in dynamic distributed computing environments," Thesis (Ph.D.), Cardiff University, 2012. [Online]. Available: http://orca.cf.ac.uk/44477/

[29] B. J. Mills, Y. Donnadieu, and Y. Godderis, "Spatial continuous integration of Phanerozoic global biogeochemistry and climate," *Gondwana Research,* vol. 100, pp. 73-86, 2021.
[30] R. Ahmad and I. Alsmadi, "Machine learning approaches to IoT security: A systematic literature review," *Internet of Things,* vol. 14, p. 100365, 2021.
[31] I. Saidani, A. Ouni, and M. W. Mkaouer, "Improving the prediction of continuous integration build failures using deep learning," *Automated Software Engineering,* vol. 29, no. 1, pp. 1-61, 2022.
[32] F. Vega *et al.*, "An IoT-based open platform for monitoring non-ionizing radiation levels in Colombia," in *Communications and Computing (COLCOM), 2016 IEEE Colombian Conference on*, 2016: IEEE, pp. 1-4.
[33] I. Foster, J. Vockler, M. Wilde, and Y. Zhao, "Chimera: A virtual data system for representing, querying, and automating data derivation," in *Scientific and Statistical Database Management, 2002. Proceedings. 14th International Conference on*, 2002: IEEE, pp. 37-46.
[34] G. K. Sriram, "A Novel Approach for Cloud Exchanger Problem Using Blockchain Based Solution," *International Research Journal of Modernization in Engineering Technology,* 2022.
[35] G. K. Sriram, "Edge Computing vs. Cloud Computing an Overview of Big Data Challenges and Opportunities for Large Enterprises," *International Research Journal of Modernization in Engineering Technology,* 2022.
[36] F. Zampetti, C. Vassallo, S. Panichella, G. Canfora, H. Gall, and M. Di Penta, "An empirical characterization of bad practices in continuous integration," *Empirical Software Engineering,* vol. 25, no. 2, pp. 1095-1135, 2020.
[37] G. K. Sriram, "Security Challenges of Vehicular Cloud Computing," *International Research Journal of Modernization in Engineering Technology …,* 2022.