**International Journal of Management, IT & Engineering**
Vol. 14 Issue 07, July 2024,
ISSN: 2249-0558 Impact Factor: 7.119
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gate as well as in Cabell's Directories of Publishing Opportunities, U.S.A

# DevSecOps: A Thematic literature review

**Suryaprakash Nalluri**
**Karanpreet Kaur**

Abstract

Most thriving software development exemplification is DevOps that focuses on the thematic study of the challenges faced while adopting DevSecOps, identifies gaps that require further future research. For this, an exhaustive literature survey is performed to demonstrate that Shift-left security approach and continuous security assessment are key recommendations for DevSecOps.

*Keywords:*

**Security;**
**DevSecOps;**
**SecDevOps;**
**Appsec Champion;**
**Shift left initiative.;**

*Author correspondence:*

**Suryaprakash Nalluri,**
**Email: spnalluri@gmsil.com**

## 1. Introduction

The constant growth rate of sophisticated, high-speed cyber-attacks brings new challenges to the people working in the cyber security domain. There is a need for all the vulnerabilities to be fixed, detect attacks in real time and respond to security incidents effectively. At the same time further down the development pipeline, another challenge of time is faced by software developers. Business needs are constantly pressurized for faster software release cycles. At the same time, with the rising number of attacks and open-source dependency, the security of software is critical in today's context, particularly in a cloud environment [1]. DevOps is like a significant inroad into a range of IT organizations. Most organizations adopt DevOps that focuses on rapid software development and delivery through agile practices to enhance the collaboration between development and operation teams to reduce inconsistencies between development, operations and release. DevOps (Development and Operations) has led to a paradigm shift aimed at removing the traditional boundaries (or "silos") of the software development and software operations teams. However, this practice of rapid delivery has presented new challenges to organizations. One such challenge is ensuring the security of software outputs to stakeholders while maintaining the agility of DevOps. Traditionally, security is treated as a non-functional requirement, which is handled at a later stage of the software development life cycle. Accordingly, a set of standard application security tests or activities are conducted on a software release. These activities often need substantial manual effort i.e. Dynamic Application Security Testing (DAST). Therefore, applying the same security tests in the context of DevOps would hinder the speed of deployments. Recently, DevSecOps evolved from the DevOps model as software development teams realized the importance of addressing security concerns early in the development cycle. DevSecOps integrates security management throughout the development process to coordinate activities among the trio of development, operations, and security teams. As the interest in DevSecOps continues to rise in the industry, it is valuable for practitioners to be aware of such adoption challenges and the solutions available to address them. Many companies are behind in achieving their DevOps and DevSecOps goals: 76% of organizations acknowledge they need to be more strategic about how they manage DevSecOps, and 17% still consider themselves at an exploratory and proof-of-concept stage [2]. Security is the number one driver behind most DevOps and DevSecOps implementations. Yet only 30% feel confident in the level of collaboration between security and development, 86% experience challenges in their current approaches to security and 51% admit that they don't fully understand how security fits into DevSecOps. Culture is the biggest barrier to DevOps and DevSecOps success. The main contribution of this paper is as follows- Firstly, this paper is intended for practitioners who are planning or in the process of adopting DevSecOps to be aware of the frequently reported problems in this domain. Identifying the adoption challenges at a very early stage of a project would be beneficial in addressing them early. Secondly, the aim is to provide practitioners with a critical review of the proposed solutions related to DevSecOps adoption, reported in the peer reviewed studies. Thirdly, this study can be a starting point for further research in the research community, as the gap areas in DevSecOps are identified that are based on the current literature. Accordingly, the main aim in this study is to systematically select, thematically analyze and present the challenges, solutions, and gaps for further research on DevSecOps. To achieve this aim, a TLR is conducted to evaluate a selected set of peer-reviewed [3]. Based on the results, this paper makes the following three specific contributions:

- A thematic classification of the main security-related challenges an organization could face in adopting DevSecOps is presented.
- The current solutions proposed in the literature, which address these challenges in terms of guidelines are described, best practice, methodologies or frameworks, and tools. Afterwards, thematically map the challenges to the proposed solutions.
- The potential gaps for future research or the areas for technological development (e.g., tools) or framework support is identified by combining the findings of the above two contributions.

The rest of the paper is organized as follows: Section 2 describes the Related work; section 3 explains the research methodology, section 4 describes the results and discussion, section 5 describes the limitation of the proposed research study, and section 6 describes the conclusion and future analysis.

## 2. Related work

In modern software development, competition, and rapid technological advancement and automation, technological progress is often identified in a new digital device model that provides a speedy software release to end-users. Researchers conducted a mapping study of secure DevOps research. Only five parts of applicable academic research are highlighted. Those five studies and three industry parts are mapped to the secure DevOps state and elaborated the phenomenon of SecDevOps or DevSecOps (the merging of Development, Security and Operations) not just as a "marketing buzzwords" but as a significant subject for future research including 11 implementations of DevOps. This review paper deepens the knowledge of this study, enabling the creation of more in-depth TLR. Authors enlightens a multivocal literary review based on the definitions, benefits and challenges of DevSecOps [4]. Review focused on analyzing 52 reviewed papers including Internet artifacts that aimed at DevSecOps. Four principles of DevOps, Culture, Automation, Measurement and Sharing, (CAMS) are highlighted to have a deeper understanding of DevSecOps [5]. This paper continues their work by looking at later academic articles to study the subject of security practices in DevOps and DevSecOps. Further, Mao et al. carried out a Grey Literature Review (GLR) on DevSecOps [6]. Our research differs from GLR as the study is based on the research questions addressed, while GLR research questions investigate the impact and the key practices of the DevOps paradigm. Some of the practices captured by this study are reported as solutions in our study too. We map these practices to the challenges identified in our thematic analysis. We have also identified studies that have used a systematic review of the literature as part of the study. The study conducted by Ramaj et al. contained SLR to identify the security challenges in DevOps as part of the research [7]. The authors then evaluated the identified challenges using DevOps experts. DevSecOps strives to ensure that software applications are secure before being delivered to the customers and are continuously secure during application updates. Various such as Security challenges in DevOps Ensuring pipeline security , Balancing security and fast deliveries, Increased insider access, Balancing automated security activities with manual activities, Getting the security requirements right, Getting developers' security knowledge to the required level, Finding the right security activities and tools that fit DevOps development style and technologies, Faster deliveries require constant monitoring and faster bug-fix processes, Including the security team in the development life cycle. Challenges with DevSecOps can be broadly categorized into four areas: tools, practices, infrastructure, and people. In the following sections, we discuss each of these four areas and the most common challenges facing them. What kinds of solutions have been suggested by previous studies to mitigate these challenges are also investigated.

## 2.1. Tools

Automation plays a crucial role in DevSecOps, such that tools which perform these automated tasks are very important to any DevSecOps implementation. Different tools are designed to be used in different phases of DevSecOps to manage the various security tasks. These tools can include, for example, automated security scanning tools, threat detection systems, and vulnerability management tools. The security tool market has many tools available, and selecting the right ones is important for the smooth adoption and usage of DevSecOps. It is not only about finding the most suitable tool for each task but also how these tools perform together. Secondly, no single automated security tool can catch all security flaws. Different tools are designed to spot different types of vulnerabilities, and so a combination of tools is used to ensure broader test coverage. Each tool produces its own testing report, so accessing the different security testing reports and consolidating the results becomes an issue. Thirdly, installing, configuring, and maintaining DevSecOps tools present major challenges for the security teams. Without meticulous configuration, the usefulness of security tools is severely hindered. Security tools used out of the box without careful configuring can result in major gaps in their security coverage, and so vulnerabilities can be missed. This is especially true with Static Application Security Testing (SAST) tools that usually find large amounts of false positives. In addition to it, various tools need to be integrated into the pipeline to identify static and runtime security issues. The inclusion of security tools in the pipeline would also alleviate the integration challenges. The security testing tools require fine-tuning or customization to reduce false positives.

## 2.2. Practices

In this part, the challenges related to security practices in DevSecOps are considered. Organizations implementing DevSecOps will have to initially do a big push before finding their version of DevSecOps which works for them. This initial recourse cost might also encourage companies to wait until DevSecOps is more mature. The fact that DevSecOps is not standardized will also affect this thesis since there might be significant differences in DevSecOps' implementations between companies. As for the issue of DevSecOps lacking standardization, though the answer would be to create these standards, this will take time. In the meanwhile, the solution would be to adopt things that are usually associated with practicing DevSecOps. The first is the practice of shifting security to the left, already mentioned previously. Teams should, for example, implement continuous security assessments and threat modeling in the planning phase. The second is to utilize known metrics to measure security, one example being the widely used and trusted OWASP proactive controls (Open Web Application Security Project [8]. Another widely used model for threat modeling is STRIDE which addresses the most common threat categories: spoofing, tampering, repudiation, information disclosure, denial of service, and elevation on privileges. The third is to have sufficient monitoring and quick feedback loops to detect any vulnerabilities in production. Security findings can then quickly be fixed by using DevOps methods, just like any other bug or feature change. Finally, good security assessments, monitoring, and feedback loops are enabled by sufficient documentation and good logging practices.

## 2.3. Infrastructure

Some challenges with DevSecOps have to do with the already existing infrastructure. DevSecOps can be difficult to implement in some infrastructures. Very resource constrained, distributed, heterogeneous, or complex cloud environments are especially difficult infrastructures for DevSecOps. Heavily regulated environments also pose major challenges. Reasons for DevSecOps infrastructure challenge are: In the case of embedded systems, DevSecOps as a methodology can be a poor choice. In embedded systems, compliance and security are more important than software development speed, so a more traditional approach to security could be the better choice. Similarly, highly restricted environments have policies and practices that fit poorly with DevSecOps' principles [9]. Such policies and practices include, for example, segregated environments, zero trust architectures, and restrictive communication policies. This only goes to show that DevSecOps is not suited for all occasions. It is found in their study that sub-optimally implementing DevSecOps in a highly regulated environment caused the security level of the client company to worsen. Authors in turn, created a secure abstraction framework using Kubernetes suitable for services consisting of both on-premises and cloud-based resources [10]. The issue of complex or restricted environments is currently an ongoing one for many companies. Research in implementing DevSecOps in challenging infrastructures is still young, and a prevailing solution for these infrastructures has yet to emerge.

### 2.4. People

DevSecOps needs the support of the whole organization, from developers to management. The DevSecOps activities need to be prioritized as a part of software development. Hence, DevSecOps requires a major cultural shift. The most common challenges related to the people of an organization are focused that specifies the interest towards the implementation of DevSecOps. First, Author highlighted the DevSecOps purpose, to create stronger communication between security and the development and operations teams, Author also showed in their study that it can also create an adversarial environment. One common solution would be to organize secure coding training to increase their knowledge of common security vulnerabilities. There could also be training on how to use automated security testing tools and how to manage found security vulnerabilities. Though the goal is to get developers involved in security, the author discovered that developers were not involved in all the intended security activities. This could be at least partially explained by the lack of security knowledge. With less participation from developers, a bigger burden is placed on the security teams to perform these tasks. However, there are currently not enough security people to fill all the open positions. Finally, using DevSecOps alone does not guarantee that security will be prioritized sufficiently in the organization.

Based on the SLR by [11], many studies states that DevOps is enabled by continuous practices. For example, CI, CD, and CDE are key practices that enable the rapid and continuous deployment cycles of the DevOps paradigm. Another component of DevOps [11] which is quite like continuous practices, is tools. Tools are a critical component in both continuous practices and the DevOps paradigm, as they enable automation. We note that there are many overlaps between what practitioners consider as DevOps tools and tools used in continuous practices (e.g., CI tools). Based on the above reasons, we argue that to cover security of DevOps, security of these continuous practices needs to be considered. To verify this argument, we conducted pilot searches in digital libraries using the search terms that captured the relevant studies (e.g., security AND "continuous integration"). By analyzing the results, we found a number of studies that can contribute to our research questions. However, none of the previous SLRs or MLRs (Table 1) with similar research questions have considered the security of these continuous practices, which we have captured in our study. Only the study by [12], which is not a core literature review study, has captured this aspect. When searching for internet artifacts in Google search to answer their research questions and then to prepare a survey on security in DevOps, they used "Security in Continuous Delivery" and "Security in Continuous Deployment" as search terms. Lastly, Stahl et al. state that the terms DevOps and continuous practices are widely used interchangeably [11]. Therefore, our decision to cover the security of continuous practices reduces the possibility of missing out on the relevant studies. Ultimately, this is why we managed to capture more peer-reviewed studies relevant to security in DevOps than the previous SLRs or MLRs. In summary, this review differs from the existing studies in the following ways.

- The combination of challenges related to adopting DevSecOps and proposed solutions have not been systematically reviewed using a substantial body of literature. By identifying the challenges, solutions, and the mapping between them, we were able to identify key gap areas in this domain.
- Security of the key continuous practices is considered which enable DevOps as part of our study. This resulted in capturing a large set of relevant studies, which were not included in the previous studies.

## 3. Research Methodology

An implementation of the qualitative approach-based study is performed for the extraction of the desired knowledge from the restricted quantity of analyzed literature. Therefore, TLR is followed for the identification of the key challenges and the identical methodologies [13]. TLR incorporates a significant interest in IT, and researchers adopt from this knowledge.TLR is different from the conventional literature survey and is an ideal approach to identify, calculate, analyze and highlight the most relevant research studies, deploying the default protocol of search strings. TLR provides much less biased, more reliable, and accurate results than the generic literature review. TLR constitutes three main stages, i.e., plan, organize, and report stage. In the plan stage, majorly two steps are involved:

- Find the purpose for review.
- Unfold and validate the TLR protocol.

In the organize stage, the following sequence of steps is followed:

- Perform preliminary investigation using search terminologies
- Final study selection based on predefined Incorporation /Elimination criteria
- Evaluation of research quality
- Data extraction from a final selection of research papers
- Extracted data synthesis from papers

In the report stage, the results are drafted.

3.1. Search String Construction

A. Trial Search - Trial search helps to search for the most relevant literature available about DevSecOps. It searches in online electronic databases which are IEEE Xplore, Springer Link, ACM Digital Library, Science Direct, and Google Scholar. (DevSecOps AND Challenges AND Solutions). Then trial search string is expanded for more details: ((DevSecOps OR SecDevOps OR ‘‘development security operations’’ OR “ security development operations’’ ‘‘software development’’ OR ‘‘IT operation’’ OR ‘‘product development’’ OR ‘‘continuous integration’’ OR ‘‘continuous development” OR ”CI/CD” OR ‘‘collaborative culture’’) AND ( problems OR obstacles OR challenges OR barriers OR issues)). 2) Search String Attributes Recognition Following is the search approach implemented for the search string constructing:

- Research questions are used for major terms extraction on the basis of population identification, its intervention and result outcome.

Results RQ: DevSecOps process, culture, challenges, Practices.

- Specifically for major terms, alternate spellings and synonyms are searched.

Results RQ: (DevSecOps AND Challenges AND Solutions). Then trial search string is expanded for more details: ((DevSecOps OR SecDevOps OR ‘‘development security operations’’ OR “ security development operations’’ ‘‘software development’’ OR ‘‘IT operation’’ OR ‘‘product development’’ OR ‘‘continuous integration’’ OR ‘‘continuous development” OR ”CI/CD” OR ‘‘collaborative culture’’) AND ( problems OR obstacles OR challenges OR barriers OR issues)).

- Keywords verification is performed in some relevant papers

Results DevSecOps, DevSecOps development, DevSecOps process, DevOSecps culture, culture, challenges, culture challenges.

- Boolean operators are used. For example, ‘AND’ for the concatenation for the major strings and ‘OR’ for the concatenation of alternate spellings and synonyms.

Results RQ: (DevSecOps AND Challenges AND Solutions). Then trial search string is expanded for more details: ((DevSecOps OR SecDevOps OR ‘‘development security operations’’ OR “ security development operations’’ ‘‘software development’’ OR ‘‘IT operation’’ OR ‘‘product development’’ OR ‘‘continuous integration’’ OR ‘‘continuous development” OR ”CI/CD” OR ‘‘collaborative culture’’) AND ( problems OR obstacles OR challenges OR barriers OR issues)). 3) Results of various databases using search string. It searches in five different digital libraries/search engines i.e., IEEE Xplore, Springer Link, ACM Digital Library, Science Direct, and Google Scholar. ((DevSecOps OR SecDevOps OR ‘‘development security operations’’ OR “security development operations’’ ‘‘software development’’ OR ‘‘IT operation’’ OR ‘‘product development’’ OR ‘‘continuous integration’’ OR ‘‘continuous development” OR” CI/CD” OR ‘‘collaborative culture’’) AND (problems OR obstacles OR challenges OR barriers OR issues)).

B. Incorporation Criteria The following Incorporation criteria are used in the proposed study for the filtration of relevant literature leading to desired data extraction. Papers based on the practices and challenges of DevSecOps culture are focused and are written in English and available electronically. The criteria for incorporation is defined as:

- Research articles are included that provide keywords mapping as described in the search string.
- DevSecOps culture relevant Research articles.

- DevSecOps cultural challenges-based Research articles.
- Solution for DevSecOps cultural challenges-oriented Research articles.
- Research articles that highlight DevSecOps culture in real-world practices.

C. Elimination criteria Elimination criteria are implemented when the research publications are irrelevant to the proposed review study and eradicate that literature to ease and relate the data extraction process. The following elimination criteria are defined:

- Irrelevant research question-based research articles.
- Eliminated unassociated DevSecOps Culture, its challenges and practices in the software development companies research articles.
- Excluded research articles that are not written in the English language and are redundant in more than one digital library

## 4. Results and Discussions

In this section, the results from executing TLR in conjunction with our research questions. Outlines of DevSecOps found through TLR study are enlightened and answered the defined research question. DevSecOps definition: In the literature, it is reviewed that DevSecOps is a significant extension of DevOps, for the integration of the security controls and processes into the DevOps SDLC. The collaboration between security teams, development teams and operations teams is promoted.

4.1. Principles of DevSecOps: DevSecOps is based on DevOps and the CAMS principles with the adoption of security from the beginning. Culture of DevSecOps: DevSecOps collaborate with the security team as well as promote a culture of security integration in DevOps work. For this certain set of security metrics are developed, promotes client focus by aligning business and security strategies of the organization. Now the security teams are involved in SDLC from the beginning and security included in each DevOps phase such as :

- Plan or Design :- Defining Security Requirements and Performing Threat Modeling
- Build : IDE Lint plugins, Perform code and library scans for vulnerabilities, open source
- Pipeline: Scan the code and libraries in the pipeline and enforce gating

4.2. Culture of DevSecOps: DevSecOps collaborates with the security team as well as promote a culture of security integration in DevOps work. DevSecOps team responsible for maintaining a system needs to have both the authority and responsibility for their system. The culture required for DevOps must promote shared ownership. The team has to be responsible for the application; teams need to be responsible, but they also have the authority to manage their service. There must be a culture that promotes learning from failure, which means that logs and monitoring data need to be visible and traceability of code needs to be put in, which improves compliance.

4.3. Orchestration of DevSecOps: DevSecOps focuses on automating security by keeping up the speed, scale and metrics. DevSecOps encourages the implementation and development of metrics for tracking threats and vulnerabilities throughout SDLC.

4.4. Shift security to the left: An overview of the challenges from DevSecOps is provided in the following section. The review study incorporates all the challenges, has been identified as the most critical challenge identified during TLR is lack of collaboration and communication. Development, security and operations teams lack behind in sharing common goals and plans. Hence, it becomes difficult to communicate, leading to timeline delays. Establishment of a cooperative environment is needed as it will rearrange and evaluate the team's perceptions. The lack of skill and knowledge is marked as another significant challenge as per TLR. Some organizations are not equipped with skilled employees following DevSecOps practices [14]. Lack of technical knowledge and key concepts understanding, and challenges of implementing DevSecOps are other challenges. Sometimes, required training and motivation to learn DevSecOps is missing. Teams are interested in expertise only in their domain, which leads to numerous challenges. Some of these challenges are solved by hiring a security person and leveraging them as security champions.

## 5. Limitations

The proposed review study implemented TLR for conducting a systematic review of the literature based on DevSecOps culture. The proposed review followed every step of the TLR process. This research is based on thematic literature, and material is not subject to the rigorous peer-review academic research. The literature includes blogs, industrial reports, white-papers, and academic research. DevSecOps is named as SecDevOps, DevSecOps, DevOpsSec, Secure DevOps, and Rugged DevOps [15]. The major limitation is that the TLR results can become outdated with time and as best practices change.

## 4. Conclusion

A grounded theory-based approach is used to present changes involved in DevSecOps that are both cultural and technical. The way organizations build software has to change, making significant use of automation. Any task related to security, compliance, and assurance can be automated and integrated into the CI/CD pipeline. Once the CI/CD pipeline and tools are standardized, other organizational changes follow; for example, it gives developers faster feedback, which gives better results due to the changes being tested in running systems. Monitoring of these systems also represents a challenge. It demands a complementary approach using both log data and survey data to get real-world feedback. The information generated through monitoring needs to be visible to all parties involved; doing so will provide visibility to all parties involved and might help align the incentives for the various teams in DevSecOps, which in turn helps collaboration. However, these practices increase the security exposure of the system, which can be mitigated by securing the underlying tooling, standardizing the tooling, and doing manual checks parallel to the build pipelines. Access control must also be managed using granular configuration in CI/CD security tools. DevSecOps will require cross-training to be implemented properly since security needs to shift left, which means security has to be integrated into the coding cycle. Leveraging security tools during coding in developer workstation, scanning code in repositories and in pipeline using SAST tools and deploying runtime tools such as DAST or IAST automated via the CI/CD pipeline would reduce the developer feedback loops and enable automation.

## References

[1] P. M. Mell and T. Grance,"The nist definition of cloud computing," *Special Publications (NIST SP)*,pp.800-145, vol.7, 2011.

[2] Forsgren, N.; Kersten, "M. DevOps Metrics," Commun. ACM 2018, vol. 61, pp. 44–48.

[3] D. St˚ahl and J. Bosch," Modeling continuous integration practice differences in industry software development," *Journal of Systems and Software*, vol. 87, pp. 48 – 59, 2014.

[4] Myrbakken, H. & Colomo-Palacios, R.," DevSecOps: A Multivocal Literature Review," *International Conference on Software Process Improvement and Capability Determination*, pp. 17−29, September 2017.

[5] Ravi Teja Yarlagadda ," DevOps and Its Practices", *SSRN Electronic Journal* , vol. 9, issue. 3, pp. 2320-2882, March 2021.

[6] Runfeng Mao; He Zhang; Qiming Dai; Huang Huang; Guoping Rong; Haifeng Shen; Lianping Ch , "Preliminary Findings about DevSecOps from Grey Literature*", 2020 IEEE 20th International Conference on Software Quality, Reliability and Security (QRS)*, 2020.

[7] Xhesika Ramaj, Mary Sánchez-Gordón, Vasileios Gkioulos,Sabarathinam Chockalingam," Holding on to Compliance While Adopting DevSecOps: An SLR," *Electronics*, vol. 11, issue. 22, pp. 3707, November 2022.

[8] TOMAS, N.; LI, J.; HUANG, H.," An Empirical Study on Culture, Automation, Measurement, and Sharing of DevSecOps," *International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1–8, 2019.

[9] AHMED, Z.; FRANCIS, S. C.," Integrating Security with DevSecOps: Techniques and Challenges," *International Conference on Digitization (ICD)*, pp. 178–182, 2019.

[10] PRATES, Luís; FAUSTINO, João; SILVA, Miguel; PEREIRA, Rúben, "DevSecOps Metrics.," *WRYCZA, Information Systems: Research, Development, Applications, Education. Cham: Springer International Publishing*, pp. 77–90, 201.

[11] Stahl, D., Martensson, T. and Bosch, J., "Continuous practices and devops: beyond the buzz, what does it all mean?", *Felderer, M., Holmström Olsson, H., Skavhaug, A. and Applications, E.C.o.S.E.a.A. (Eds.), 43rd Euromicro Conference on Software Engineering and Advanced Applications: SEAA 2017 proceedings*, IEEE, pp. 440–448, September 2017.

[12] Rahman and Williams," Software Security in DevOps: Synthesizing Practitioners' Perceptions and Practices", *IEEE/ACM International Workshop on Continuous Software Evolution and Delivery (CSED)*, 2017.

[13] J. Alonso, R. Piliszek and M. Cankar, "Embracing IaC through the DevSecOps philosophy: Concepts, challenges, and a reference framework", *IEEE Software*, vol. 40, issue. 1, Jan.-Feb. 2023

[14] Ullah, K.W.; Ahmed, A.S.; Ylitalo, J.," Towards Building an Automated Security Compliance Tool for the Cloud," *In Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 1587–1593, July 2013.

[15] Kitchenham, B.; Charters, S.,"Guidelines for Performing Systematic Literature Reviews",*Software Engineering*, 2007.