Computer Science and Information Technology 10(4): 37-51, 2022 http://www.hrpub.org
DOI: 10.13189/csit.2022.100401

Secure Change Management Process: On the Effectiveness of DevSecOps

Zalialetdzinau Kanstantsin

Software Engineer Brimit LLC, Office 25, Platonova Str. 49, Minsk, Republic of Belarus

Received September 29, 2022; Revised December 6, 2022; Accepted December 22, 2022

Cite This Paper in the Following Citation Styles

(a): *[1] Zalialetdzinau Kanstantsin , "Secure Change Management Process: On the Effectiveness of DevSecOps," Computer Science and Information Technology, Vol. 10, No. 4, pp. 37 - 51, 2022. DOI: 10.13189/csit.2022.100401.*

(b): *Zalialetdzinau Kanstantsin (2022). Secure Change Management Process: On the Effectiveness of DevSecOps. Computer Science and Information Technology, 10(4), 37 - 51. DOI: 10.13189/csit.2022.100401.*

Abstract DevSecOps is a modern organizational and management system of production and development processes in the field of software and digital production, which has significant potential for implementation due to the possibility of obtaining safe, stable software products with reduced resource and time costs, which has been formed as a result of the sector. Its architecture is modernized by deep integration of safe approaches at each stage of formation of the digital software body of developed digital products. The object of research is the system of modern digital software development. The subject of the research is the integration of Security-blocks (as a focus toolkit of the organizational and management system DevSecOps) into digital fabrication with an assessment of the effectiveness of the implementation of the concept of "secure software products". The main purpose of the research is to determine the vector of potential development of modernized software-digital production aimed at minimizing digital code body vulnerabilities of developed software products and optimizing resource and timing costs. In the framework of the present study, to achieve the set goal, methods of scientific search, analytics and determination of correlative convergence, by identifying empirical patterns obtained in the analysis of results of bibliometric search of leading scientometric databases regarding organizational and managerial schemes and development processes of software-digital development and manufacturing are applied. Based on the results of the research in a given horizon of scientific and empirical data, a probabilistic vector of software-digital production and manufacturing development in the field of deep architectural solutions to reduce the vulnerabilities of the digital code body will be obtained. The practical significance of the study is determined by the possibility of identifying the appropriate horizons of scientific search and the vector of digital development and production with a focus on the integration of security control solutions.

Keywords Organizational and Management, Security, Software and Digital, Production, Cyber, Cybersecurity, Implementation, Vulnerabilities, Manufacturing

1. Introduction

Creating of software that meets the requirements of cyber and cyberphysical security, providing potential customers with confidence in the protection of confidential data and client cyberphysical systems is a relevant trend with direct economic implications. Stable and secure software complexes attract a large customer base. According to S. Pooja [1], which is based on an analysis of The National Vulnerability Database (NVD) (USA), over the past three years, the number of digital code vulnerabilities has increased by 26.6% (2019 - 17.3 thousand records / 2021 - 21.9 thousand records), data from M. Fu [2] points to the rapid growth of software code vulnerabilities - in 5 times during the last decade, J. Zhou [3] points out that about 64% of the core software of the banks of the global financial community have digital code vulnerabilities, which according to 2021 is estimated in the losses from cybercrime of $ 6 trillion. Analysis of the data presented in the aforementioned publications, as well as in

other relevant publications on the specified research vector [4-13], allows us to come to the following conclusion: 90% of software vulnerabilities are caused by the violations and defects in the source code, 21% of incidents involving loss of confidential data are caused by vulnerabilities in the digital code of software products used, every third software application being implemented and used at present has a digital code body, and in 1000 lines of software code it is revealed up to 60 % of the vulnerabilities. Consequently, due to the presence of such a system error, which tends to multiply significantly, the need to find and develop adequate solutions that will significantly improve information and cyber-digital security, as well as to secure critical infrastructure objects operating under the control of cyber-physical systems, becomes urgent. Modernization of the software production architecture through the introduction of security-free solutions and system-organizational solutions to the organizational and management structure of DevOps is justified by the following factors. DevOps has proven to be effective in increasing productivity, optimizing resource and time costs compared to typical methods of digital software development. It minimizes digital body vulnerabilities of software products, provides security for the subsequent use of a software application, which results in obtaining a stable program. Its digital code is practically devoid of defects and vulnerabilities [14]. The development and implementation of DevSecOps digital product engineering systems is currently a feasible organizational structure, which allows to obtain a secure application with an optimized production process [15,16]. Given the current trends and the need to obtain competitive stable software products, it is advisable to explore the prospects for the development of software-digital development and production methodology by implementing the organizational and management system of digital production DevSecOps.

2. Materials and Methods

In order to study the implementation and development aspects of the modernized architecture of software-digital development and DevSecOps manufacuring, we will perform a profile study, which is based on the following steps and methods:

- Scientific search in a given horizon of information data (Multivocal Literature Review) regarding the problems of ensuring the security of developed software products at all stages of digital development;
- Correlative analytics and search for empirical patterns regarding the organization of digital production with a focus on security-sensitive methods and solutions aimed at minimizing and completely eliminating vulnerabilities, defects and critical bugs in the cyrocode body of software products being developed and implemented;
- Evaluating the impact of DevSecOps security tools on the overall digital development and manufacturing process;
- Bibliometric search and analysis of scientometric databases regarding the potential vector of software-digital production development with a correlative focus on ensuring the security of developed software products and complexes.
- Assessment of the prospects of implementation in digital development of DevSecOps architecture and organizational and management system with the identification of appropriate vectors and horizons of research activities that will determine the future of the software industry and digital components of cyber-physical systems.

Collection, analysis and systematization of disparate data presented in the leading scientometric databases will allow to form an appropriate factorial background, on the basis of which it is possible to obtain analytical and correlative conclusions regarding the goals and objectives of the research with the potential possibility to determine probabilistic vectors of development of diffusion interaction of intra-architectural security solutions and the overall system of digital development and production. The formed methodology, which includes elements of scientometric analysis, will reveal the level of scientific coverage of the issues of cyber-digital and cyber-physical security of developed software products, as well as determine the subsequent evolutionary milestones of a given research vector, which has practical results and interest for all related industries in the transformation process with the transition to secure automated digital-code-development operations. Development of optimized organizational and communication schemes of software-digital engineering affects not only the quality of the produced software product (ensuring its stability and security), but also allows to significantly reduce the time and resource costs (ensuring high productivity of the IT cluster).

The latest researches published after 2022 were used to identify methodological differences in iterative methods of digital software production, to substantiate the objectives of DevSecOps, to explain its management system and security block infrastructure, to obtain statistical data on the level DevSecOps application in business, to justify the necessity of implementing the technology in IT organizations and to assess the prospects for its development. All mentioned works were published in high impact journals or in conference proceedings.

3. Results

Modern digital software development and production has come to a closed-loop DevOps production culture, having passed through Waterfall, Agile, CI/CD

(Continuous Integration/ Continuous Delivery) stages [17-22]. At the same time, there are both methodological differences in these iterative methods of digital software production (Figure 1), and systemic-production-culture-wide differences in digital development and manufacturing (Figure 2), which have the following iterative-empirical features of the development of each stage:

- transition Waterfall → Agile is associated with the need for flexible software-digital production with the transition from long logical-production processes (which are low-variant and low-maneuverability, leading to the accumulation of errors and defects in the digital code body of developed software products, requiring the introduction of a long testing phase and bringing the produced software to the desired level of performance) to short successive iterative periods of code-programming and testing, which significantly reduces error accumulation, in the
- The transition Agile → CI/CD is connected with the necessity to support the full life cycle of the developed software products with the integration of the deep mechanisms of automatic software and digital production;
- Transition CI/CD → DevOps is connected with the formation of system-wide culture of digital development and manufacture, which provides the formation of a closed system of software production, consisting of the enlarged staged cycles of digital code development and operational and operational support, which provides control, update and continuous improvement of developed and delivered software products.

Currently, the outlined methodological approaches in digital manufacturing are found in the individual processes of modern organizational and management schemes of software development and manufacturing, which are either modernized with modern tools or have a combinatorial nature of functional operation.

Gartner introduced the term "DevSecOps" (originally coined as DcvOpsSec) in 2012, and it became the most discussed topic at the RSA conference in 2017. DevSecOps aims to bring synchronous cybersecurity thinking to development teams and make everyone accountable for security. "Information security architects must integrate security at multiple points into DevOps workflows in a collaborative way that is largely transparent to developers and preserves the teamwork, agility and speed of DevOps and agile development environments, providing "DevSecOps" - Gartner [3]. Zhou J. [3], the development and intensification of the implementation of production management system focused on the safety aspects of developed software products got a start in 2017. Currently, the studied organizational and management system DevSecOps is a priority of implementation for the leading production organizations, whose software products have been widely used and integrated into the global information and cyberphysical systems.

Organizational and management system DevSecOps is a logical continuation of the current scheme of software-digital production DevOps, with the above described system of safety testing (in various modes) integrated into the processes of software products production, which allows to obtain in conjunction with more complete (compared to DevOps) production communication management system between the key participants of software-digital engineering and production cycle - Figure 3 [23–28].

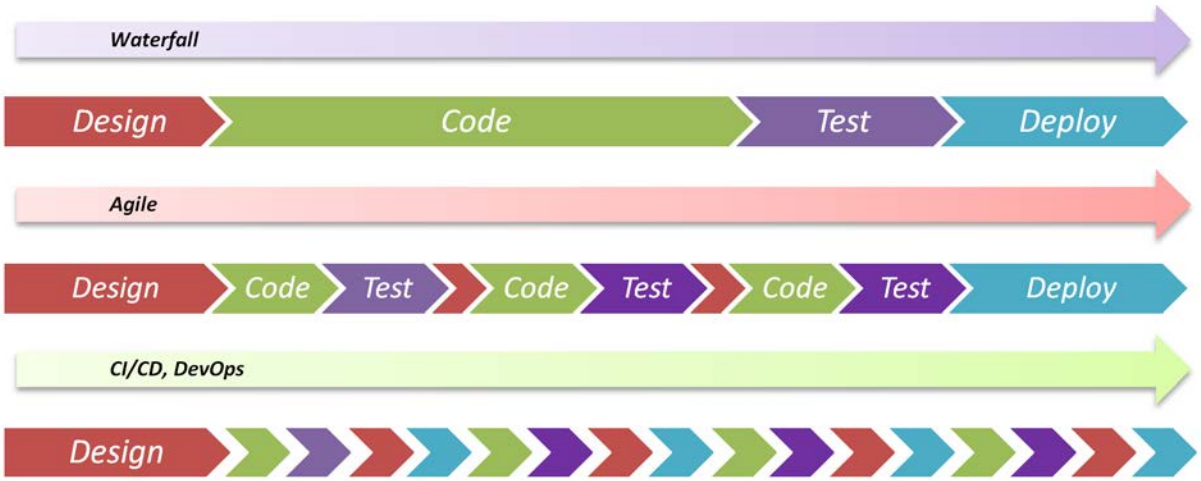

Figure 1. Visualizing the difference in methodological approaches of software-digital production in organizational-management systems of digital development, which has undergone evolutionary and iterative progress

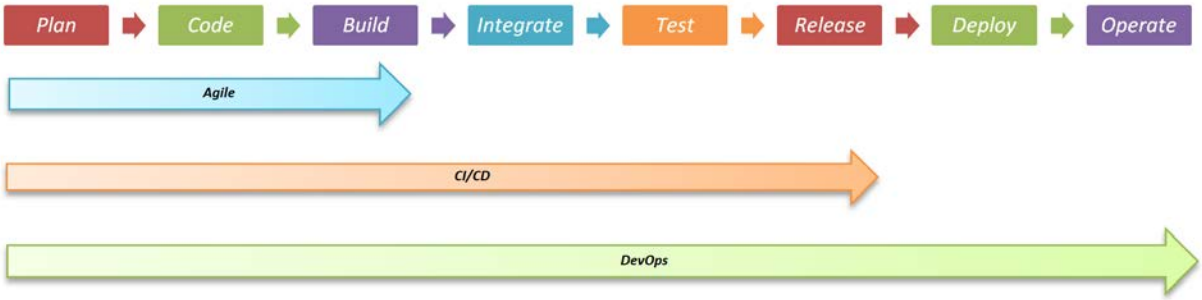

Figure 2. Conceptual scheme of the involvement of evolutionary iterations of the organizational and management system of software and digital development in the completeness of production processes of digital production

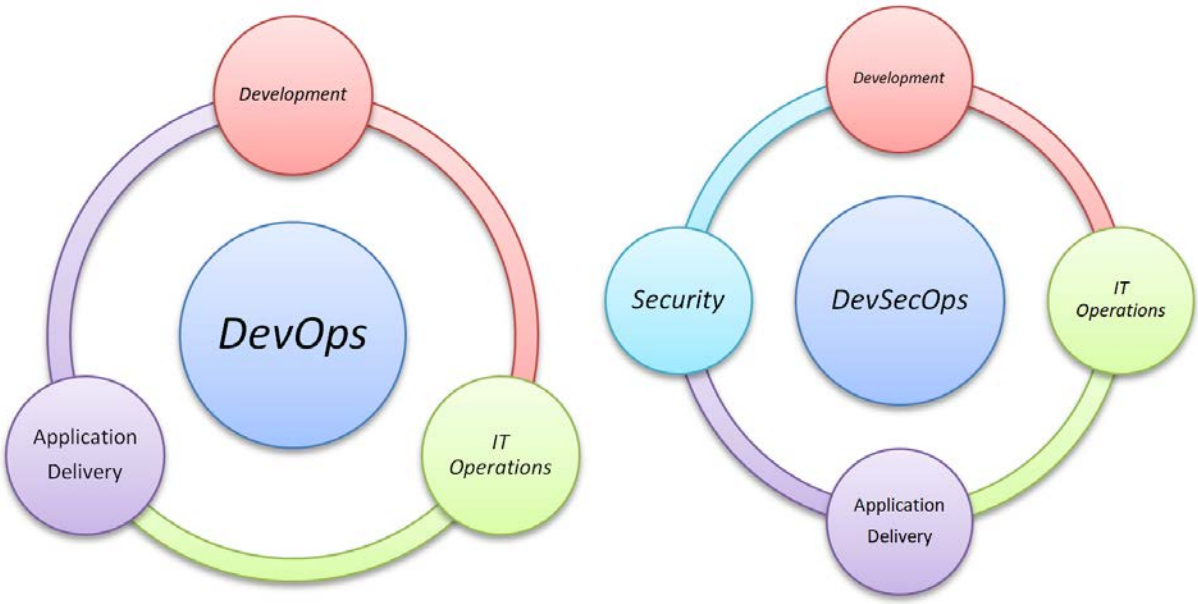

Figure 3. The difference in the structure of organizational and management systems of software and digital production, based on the management concepts of DevOps and DevSecOps

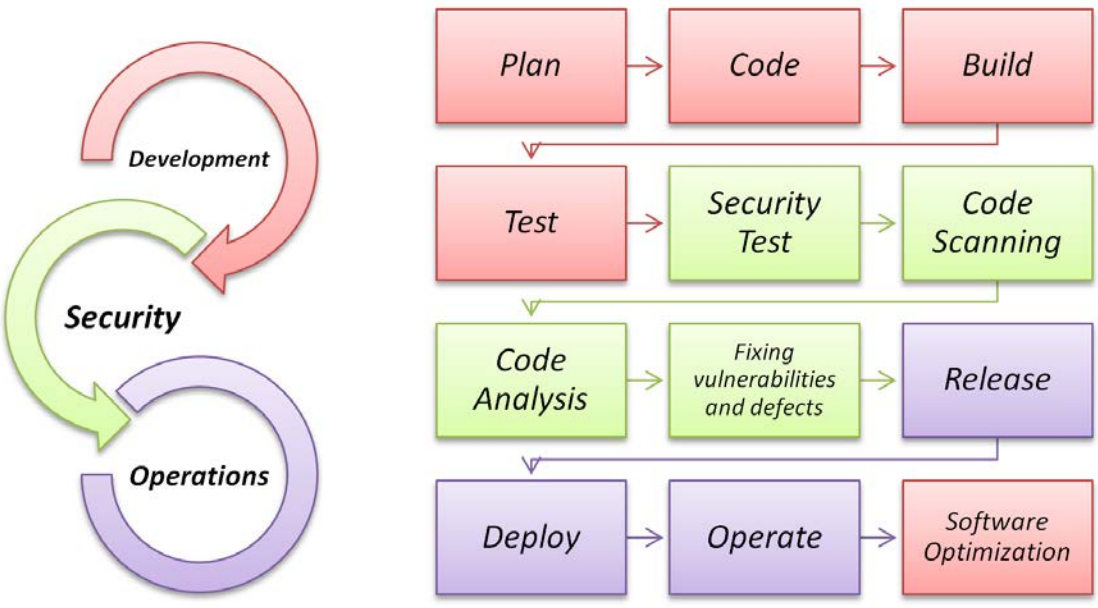

Figure 4. Algorithmization of DevSecOps concept functioning

Analysis of relevant scientific papers and publications on the device and system of DevSecOps methodology functioning, such as R. N. Rajapakse [29] (system information and analytical review on the device of DevSecOps methodology and the problems of software-digital production transition from DevOps to the integration of the logical security-free section in the generalized concept of transition to DevSecOps), A. Ibrahim [30] (research and development of proposals for implementing DevSecOps in a modular solution (using cloud services) in DevOps-based digital production process), A. Landry [31] (analyzing the experience of implementing DevSecOps-based security tools and building an internal secure data transmission system in service communications for the US Department of Defense - DARPA Secure Handhelds on Assured Resilient networks at the tactical Edge (SHARE)), N. Harshitha [32] (analyzing the integration of DevSecOps security tools in Cloud Computing technologies), M. Orosz [33] (application of DevSecOps organizational and management system in the space industry), A. Schwan-Gijima [34] (information and analytical review regarding the device and methods for implementing DevSecOps security tools and blocks in software-digital production processes), Y. Malhotra [35] (study of the problems of implementing IaC, DevSecOps and MLops security tools in hybrid cloud computing with zero-trust beyond "lift and carry"), M. Ekoramaradhya [36] (study of the possibility of applying security tools for implementation in digital Internet of Things (IoT) protection protocols), allows to formulate a generalized system-wide view of the structure of the studied organizational and management system. According to the results of the conducted multi-literature search, we conclude that the methodology DevSecOps is a generalized closed-loop scheme of sequential stages of software-digital engineering and production (from planning a software product to its release), in which the main logical competent groups (developers, operators, administrators and testers) are linked by an optimized communication network, allowing with lower resource and time costs to obtain a stable and secure software product. The general algorithm of the DevSecOps system under study is shown in Figure 4.

The DevSecOps security block infrastructure provides the use of a special set of tools of the general AST (Application Security Testing) group, which allows you to perform screening and analysis of the initial course of the tested software product in several characteristic modes [37-39]:

- Static Application Security Testing (SAST): statistical scanning of numerical code errors in the source code of the software under test;
- Dynamic Application Security Testing (DAST): a stress test of the software under development by emulating attacks using all known methods;
- Interactive Application Security Testing (IAST): testing the software under development during its normal operation (search for defects and vulnerabilities inside the running software code)).

A generalized algorithm of functioning of the Security Block of DevSecOps software production organization scheme is formed by the availability of the above described tools in the following median sequence [40-42]:

1. Structural screening of the code body architecture of the software product.
2. Using the tools for searching defects and vulnerabilities of AST group in the corresponding modes: SAST, DAST, IAST.
3. Screening library code inclusions and third-party code inclusions. 4.
4. Application software container code body screening.
5. Screening the code body of mobile applications.
6. Screening and analysis of functional Infrastructure-as-Code *(IaC).*

A general concept diagram of the tools used in the process of software-digital engineering and DevSecOps production is shown in Figure 5.

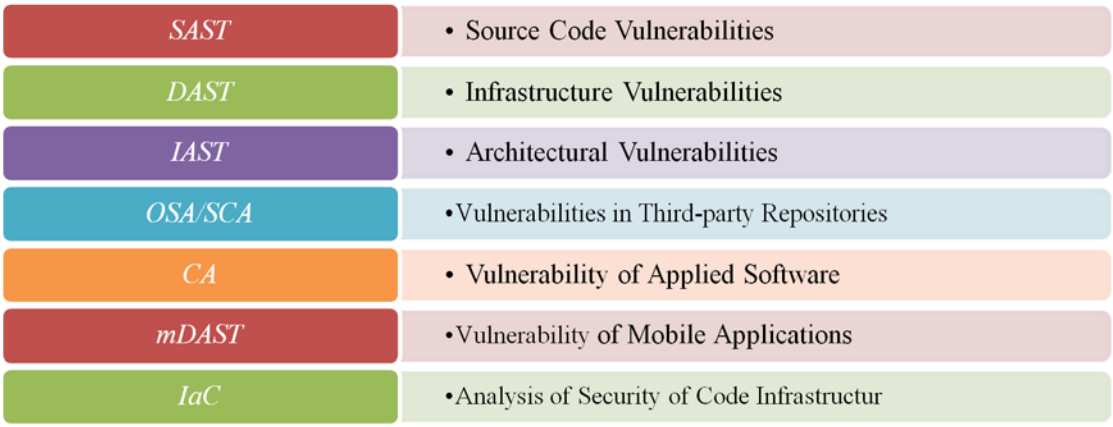

Figure 5. DevSecOps concept diagram of tools in the process of software-digital engineering and production

Multivocal Literature Review allowed to determine the iterative-evolutionary sequence of the emergence of the organizational-management system of digital development and manufacturing DevSecOps, as well as the structure and tools of the investigated software-digital corporate -production system of digital production. Tools of scientometric search and analysis VOSviewer [43] allow to define probabilistic vectors and horizons of potential development of the investigated organizational and managerial DevSecOps system by taxonometric analysis of bibliometric data of leading scientometric databases - Figures 6, 7.

Based on the scientometric data presented in Figures 6, 7, we conclude that the profile scientific research and publications regarding the specified vector of scientific search (in the direction of DevSecOps) in the current horizon of time constraints indicate that the main direction within the specified research is organizational and technical issues of DevSecOps system implementation in software and production processes of digital development and production - Figure 8.

The identified problems regarding the transformational-implementation process of integrating the organizational-management system DevSecOps are covered in the publications of the authors J. W. Beard [44], F. Almeida [45], O. H. Plant [46], R. Rose [47], T. N. Nisha [48] and the previously mentioned K. Scarfone [14], A. Landry [31], R. Chandramouli [36] highlight the issues of transition of IT-developing companies from the system management of inter-institutional communication DevOps to the organizational structure DevSecOps (with integration of the block of security testing of developed software for defects, errors and vulnerabilities of digital body), according to the analysis of which it is possible to determine the median strategy (road map) of migration to the updated structure-management system of organization of digital-production, presented in Fig. 9.

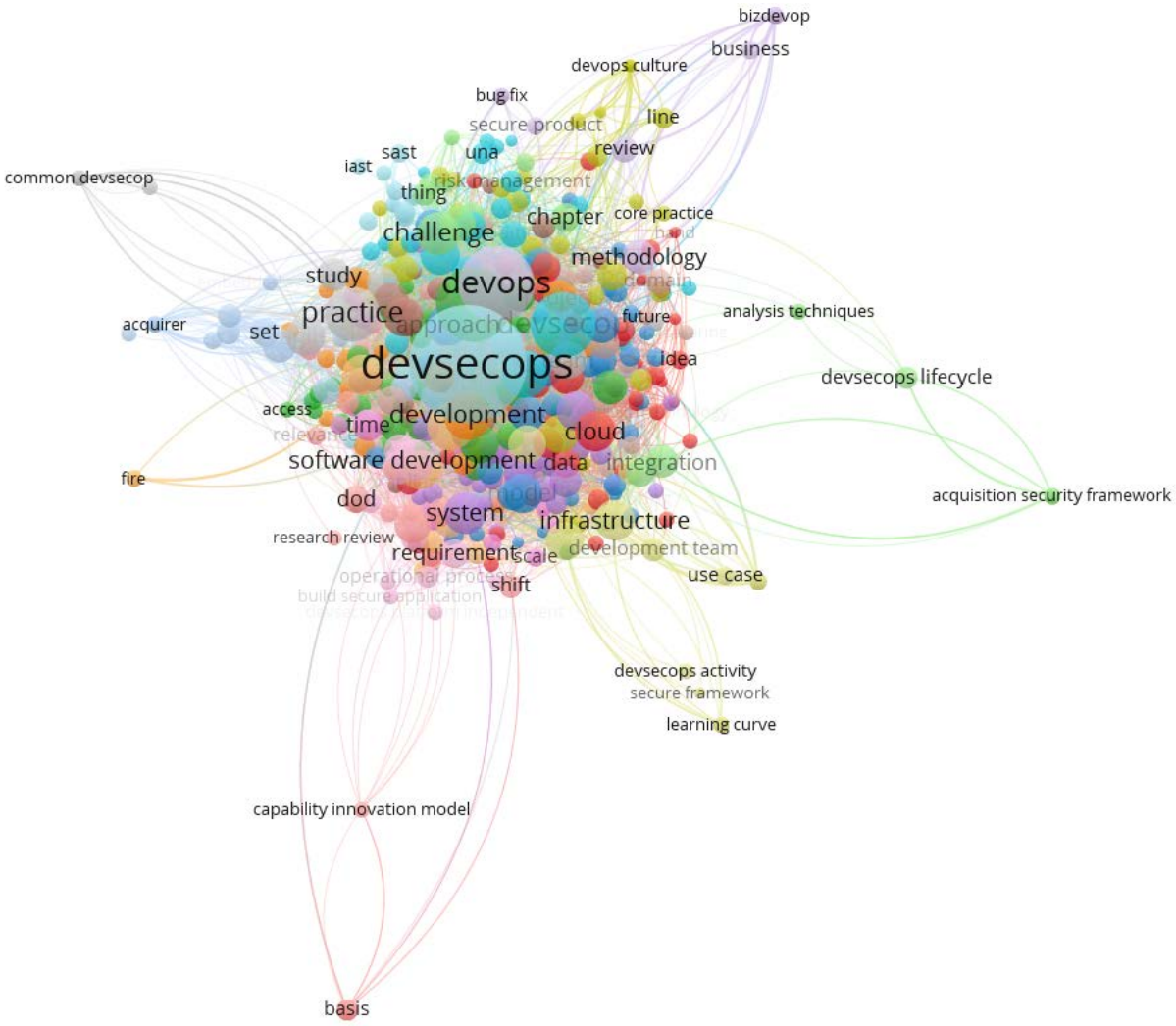

Figure 6. Taxonometric scheme of the results of bibliometric analysis of nacometric data regarding the study of organizational and management system DevSecOps (by means of VOSviewer [43])

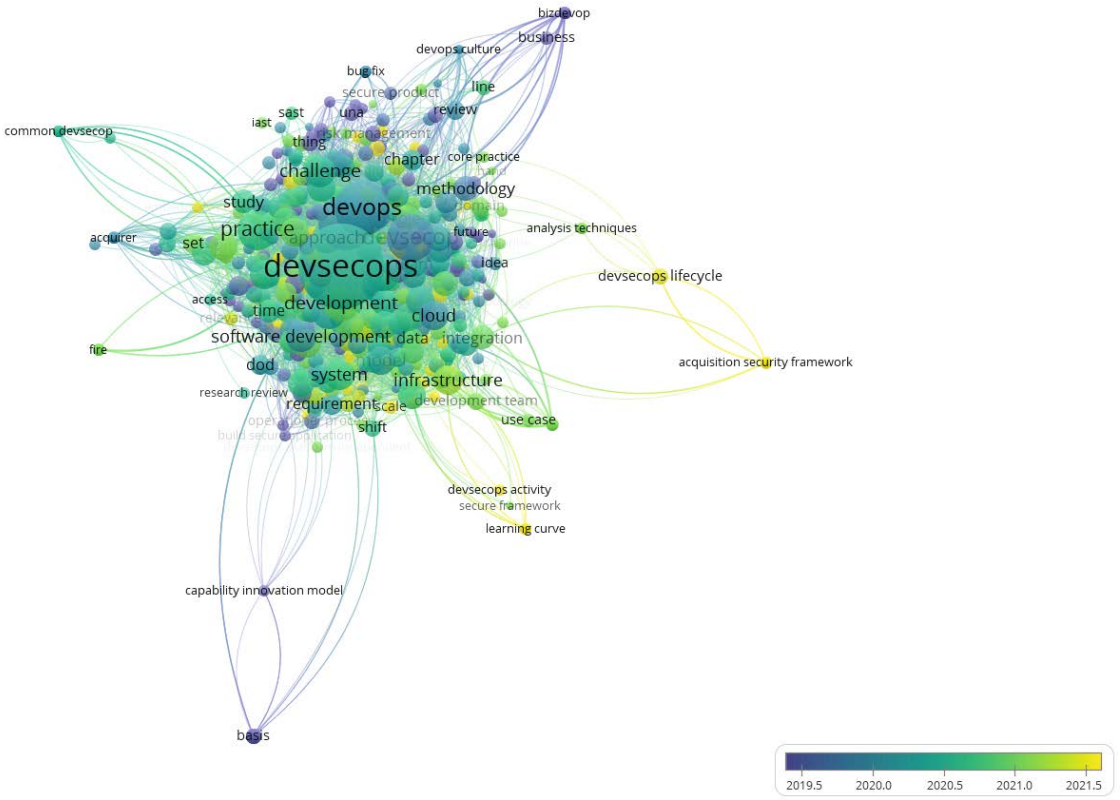

Figure 7. Taxonomic scheme of the results of bibliometric analysis of nacometric data regarding the study of organizational and management system DevSecOps by means of VOSviewer [43]

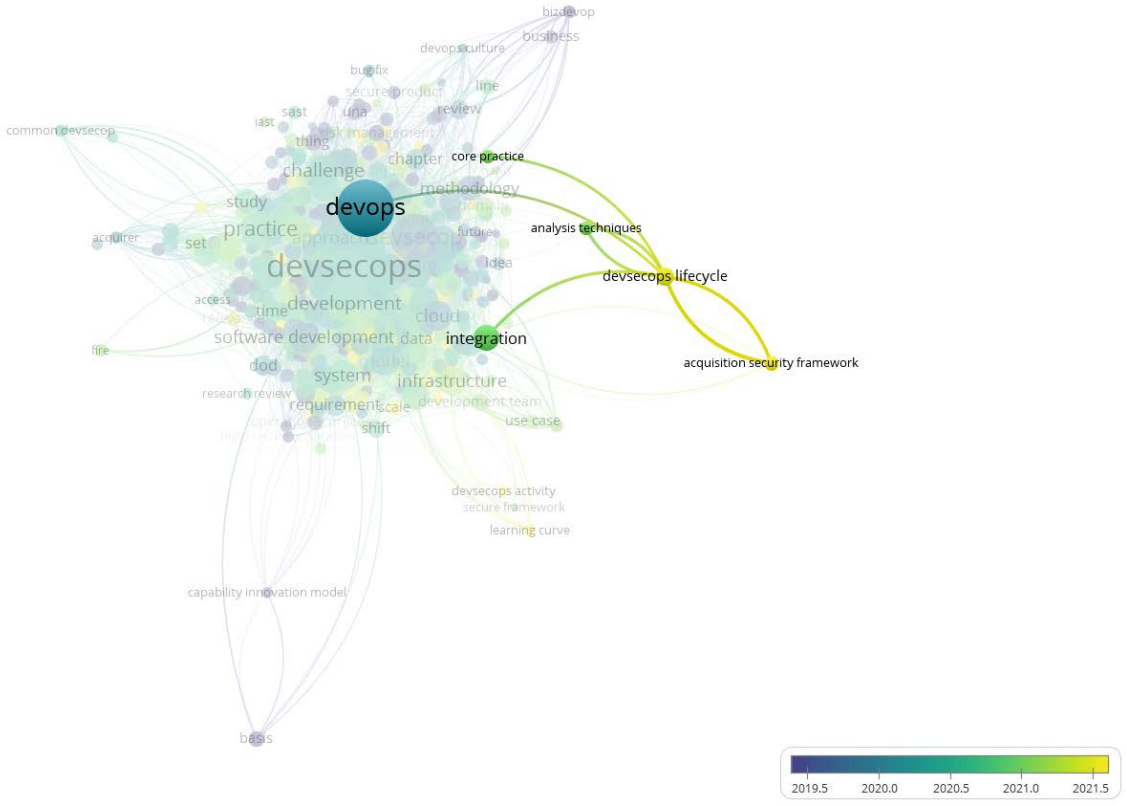

Figure 8. Taxonomic scheme of selected results of bibliometric analysis of nacometric data regarding the study of organizational and management system DevSecOps (by means of VOSviewer [43])

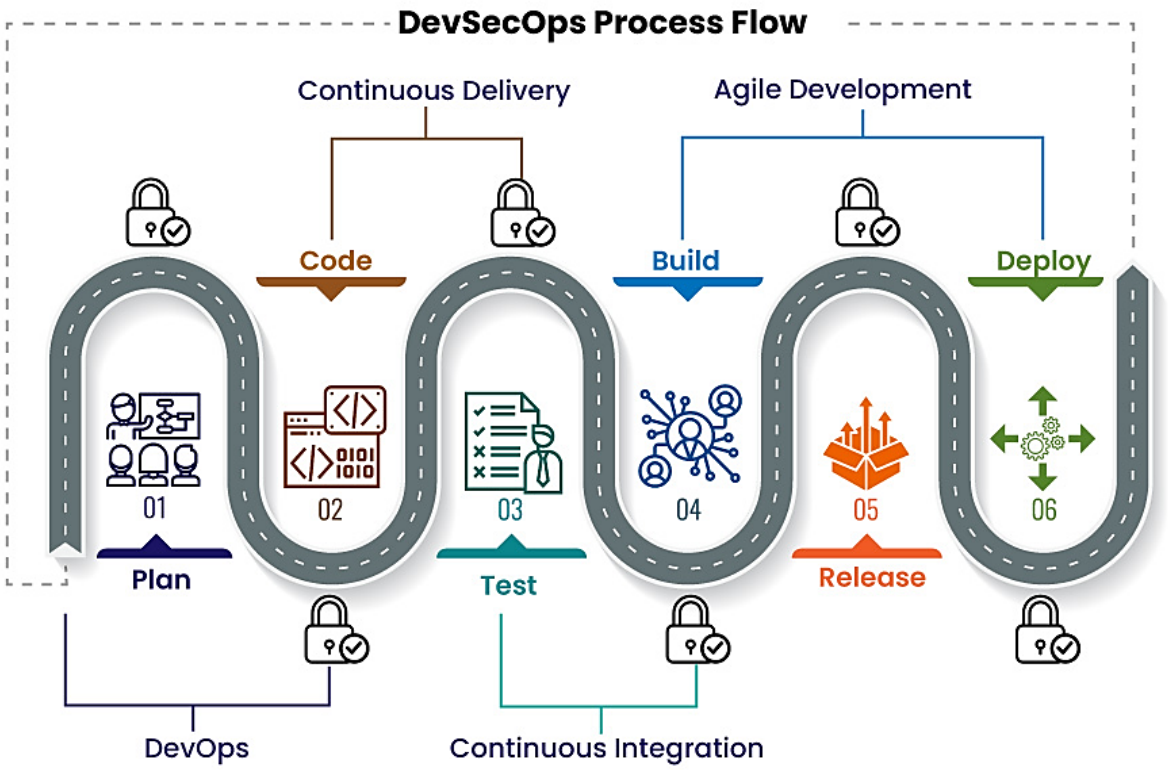

Figure 9. A median roadmap for IT development companies to transition from DevOps to DevSecOps methodology [49]

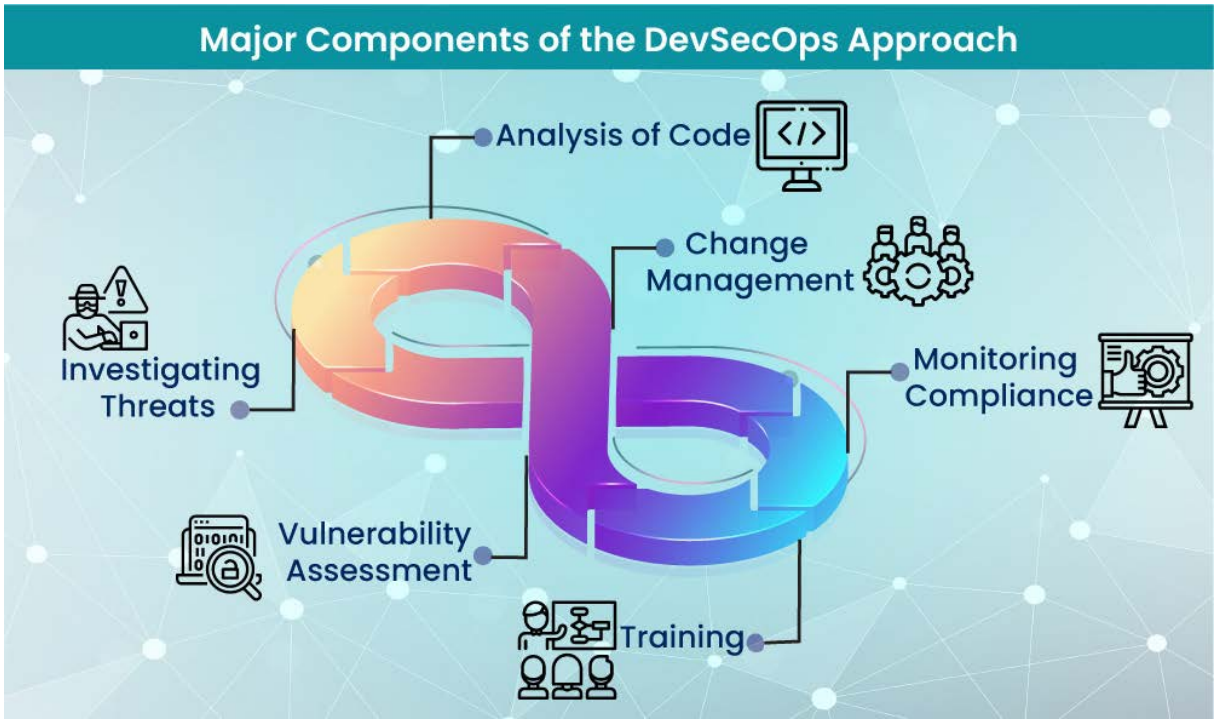

Figure 10. New corporate policy concept for implementing intra-system integration of Security Blocks as part of DevSecOps methodology [49]

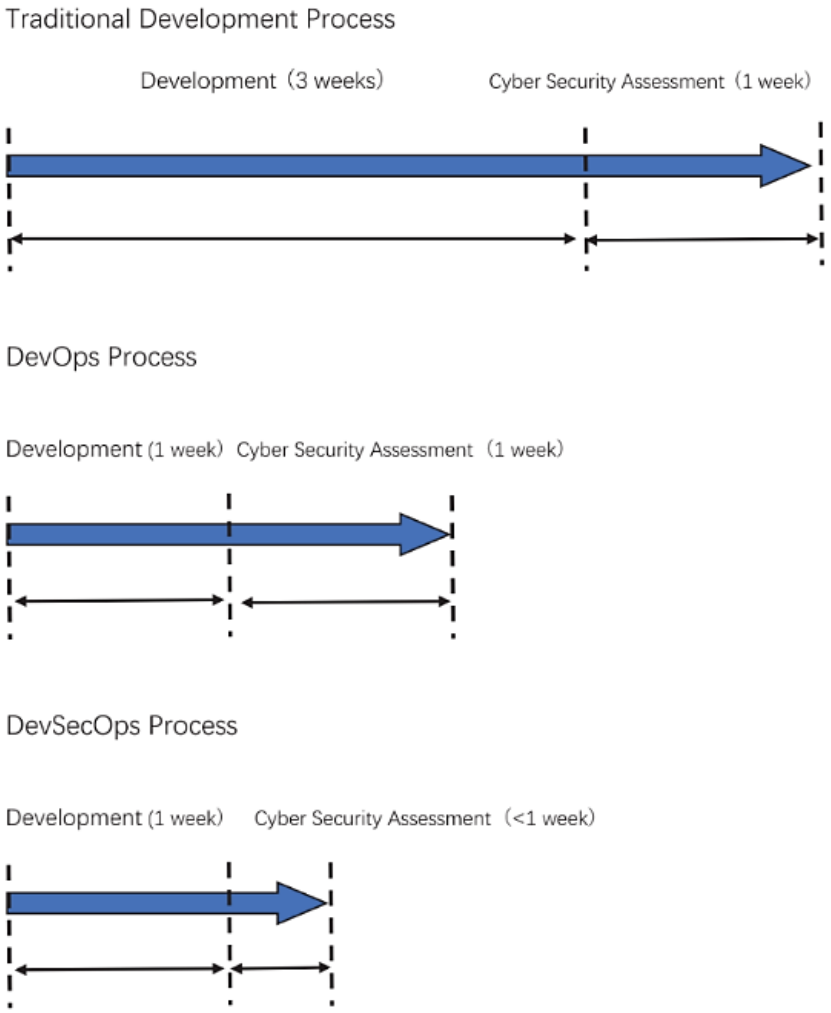

Figure 11. Comparison of timing costs under different organizational and management schemes for the management of digital production [3]

When implementing the median strategy of evolution to intra-system integration of Security-blocks in the organizational and production processes of software and digital production (Figure 9), the profile organizations get a new concept of corporate policy, which provides for a dynamic system of auto-renewal - Figure 10 [44-49].

According to the example presented in J. Zhou [3] (Figure 11), three different development processes are compared: traditional process, DevOps process, and DevSecOps process. When a development team works through a traditional development process, the first step takes three weeks to gather requirements, design the architecture, code, build, test, and finally everything is ready to be deployed in a working environment.

Before release, the cybersecurity team needs one week to assess potential risks and discover vulnerabilities in applications by conducting a cybersecurity assessment. When the DevOps process is implemented (e.g., automation, microservices, better team coordination, etc.), development efficiency is improved and development time is reduced from three weeks to one week. However, there is no change for the second phase, so it still takes one week to complete the security assessment. Now it's easy to see why cybersecurity becomes a DevOps bottleneck, especially if teams' DevOps levels are mature.

To overcome this bottleneck, the DevSecOps process was created to remove the barriers between DevOps and security. It switches security from reactive to proactive, identifying security issues early in the development process.

DevSecOps provides the strategies and technologies needed to ensure security at every stage or checkpoint in the DevOps process so that development teams can deliver better and safer products faster. To achieve this, development teams must change the traditional way of working and begin to develop a new mindset and culture that holds everyone accountable for security and develops software both with DevOps and in a secure way. The three main benefits offered by DevSecOps to the development team can be boiled down to the following three aspects:

- – speed - DevSecOps reduces time to market by moving security assessment work to the design, coding or testing stage by automating DevSecOps tools so teams can find and fix security issues early and finally reduce security assessment time;

- Controlled risk - DevSecOps provides basic security criteria for identifying and remediating vulnerabilities so that project teams can validate applications for deployment against acceptable risk criteria and therefore reduce overall business application and internal application risk;
- Resource savings - DevSecOps reduces the overall reliance on cybersecurity teams to manually perform code and infrastructure checks after release, resulting in an overall reduction in patching costs.

An extensive analysis of profile papers and publications by authors R. G. Lennon [50], S. R. Goniwada [51], J. Faustino [52], M. A. Shameem [53], W. R. Nichols [54], K. A. Rassmann [55], J. Díaz [56], allows us to identify the benefits of implementing DevSecOps tools in traditional software production processes (Figure 12):

- improvement of the software product quality at the customer (higher customer value), due to the optimization of the processes of andate and software kernel updates, taking into account the increase of security qualities;
- reduced time spent on cybersecurity testing procedures (reduced time for security checks), due to the high level of automation of test operations in the DevSecOps system;
- decreased time for release of the developed software products (increased delivery speed), due to the absence of pre-release testing stage to detect defects and vulnerabilities in the developed software;
- improved search for software vulnerabilities and cyberdigital threats (enhanced threat hunting) due to the introduction of systems for continuous monitoring of the digital code body of the delivered software product;
- Reduced time costs for recovery of exploited software (increased recovery speed) due to the system of early detection of software vulnerabilities and cyberdigital threats;
- Increased security of the developed software (improved software security) through the introduction of Security Blocks into the organizational and production processes of software and digital production on the basis of the DevSecOps production management methods investigated;
- Reduced costs of the final software product and digital production processes (reduced costs) through the introduction of a system for early detection of defects, errors and vulnerabilities in the digital code body of the software being developed.

Figure 12. Conceptual scheme of advantageous qualities of DevSecOps organizational and management system implementation in software and digital production processes [57]

Informational and analytical review of publications by M. A. Akbar [58], T. Scanlon [59], C. W. Carol Woody [60], D. W. Hallock [61], K. Siau [62] regarding the hindering reasons in the full implementation of DevSecOps organizational and management system in software-digital production processes, points out that although DevSecOps offers a new methodology to solve problems between DevOps and cybersecurity, and makes security a task for everyone in the development process, transition from DevOps to DevSecOps is not easy and requires a culture or mindset change, new management, work processes and process must be established, and new technology must be adopted. Here, the challenge of implementing DevSecOps breaks down into two major challenges:

- technical problem of DevSecOps implementation;
- the corporate-developer problem of DevSecOps implementation.

The technical side of DevSecOps implementation: software development is a complex process that may use different programming languages, large source code repositories and many open source libraries and packages. DevSecOps should be able to support them all. Unfortunately, there are not many DevSecOps tools on the market yet. In addition, many DevSecOps tools are underdeveloped and still have flaws that can slow down the promotion and widespread adoption of DevSecOps tools. Nevertheless, modern developers are developing and creating more and more quality software-digital products (including automated solutions and tools, tools based on artificial intelligence, machine learning and neural networks) to safely test developed software, which causes confidence in overcoming technical and technological factors that hinder full implementation of DevSecOps organizational and management system.

The corporate-developer side of DevSecOps organizational -management system implementation: compared to technical issues, the biggest challenges come from the people involved in software-digital production. Security is often thought of later, and has little to do with the software development lifecycle, so IT developers typically lack cybersecurity awareness. In addition, many developers believe that cybersecurity is the job of a dedicated cybersecurity department or group, and that DevSecOps is an extra job. Research [63-66] showed that 52% of companies cut their security budget to meet business deadlines or goals, indicating that the main reason DevSecOps has not become a reality in most organizations is due to a lack of support from senior management who are interested in business, not cybersecurity. Without top management involvement in pushing DevSecOps from top to bottom, it will be difficult and slow for teams to implement DevSecOps. However, the current state of affairs indicates that on the one hand potential and target client groups increasingly want stable and secure software, and on the other hand developing new solutions to automate and optimize production processes of software-digital production, allow to bring the organizational and management system DevSecOps to the first positions when choosing a system of organization of IT-development.

4. Discussion

The presented analyses regarding the prospects of full implementation of the organizational and management system DevSecOps in the production processes of software and digital engineering and production, indicate the development of the vector of potential widespread application of this system of organization of digital production in the IT-development environment.

Since 2017, DevSecOps has become more and more popular, and some digital producers have begun to adopt it. Data from researchers [63-66] shows that as of 2022, 33% of digital production companies have already implemented DevSecOps methods well for their business, and another 30% of teams are on track to partially or fully implement DevSecOps. This is a good sign that many development teams are becoming interested in DevSecOps and that DevSecOps adoption is growing rapidly.

The potential for adoption of the investigated system by the software and digital production and IT development organization is aided by the creation and development of DevSecOps cybersecurity profile tools: static application security testing (SAST), dynamic application security testing (DAST), interactive application security testing (IAST), software composition analysis (SCA), self protection tools (RASP), penetration testing to fully address security issues early in the development life cycle

SAST, DAST and IAST are the best DevSecOps tools and are well received by IT development teams. The SCA tool, also known as the Free Open Source Security Tool (FOSS) analyzes the application by detecting open source software (e.g., third-party libraries, dependencies, licenses, etc.). Materials (BOM) store all open source information in the code base (e.g., versions and potential vulnerabilities). Comprehensive SCA solutions monitor the specification to notify customers in advance of new vulnerabilities and even deliver updates or patch guides. In addition, as cloud and container adoption grows, container security will become more and more important, leading the IT team to migrate to cloud services and computing.

Regarding the response to a discovered critical open-source vulnerability, the researchers' data [63-66] indicate that more than half of IT teams need to spend two to three weeks to eliminate the discovered vulnerability in the source code of software under development, in other words, the process of creating the target software product stops (lags) by the specified period (two to three) weeks. Moreover, a quarter of teams ignore risks in their applications for up to a month, and only 16% of teams fix the vulnerability within a week. Because of this, most IT development teams do not respond quickly enough to a

critical vulnerability and therefore keep their applications at risk. A developer may not have the right mindset to know about vulnerabilities, may lack the knowledge to fix a vulnerability, may not have an automation solution to speed things up, or may have to go through complex and time-consuming processes to deal with security issues. Any of the reasons outlined could result in a slow response to critical vulnerabilities discovered and expose their applications to potential attacks.

The most common challenge IT organizations face is the cultural change required to implement DevSecOps. People who have used traditional or agile development approaches often have trouble adapting to a completely different approach. When moving to DevSecOps, the IT development team must learn a lot about cybersecurity, become more open about work issues, and make security practices part of their routine. Many software and digital production organizations underestimate how complex these changes can be and, as a result, fail to fully implement DevSecOps. Another problem is that some IT organizations try to completely replace a working agile production process (Agile) with DevSecOps. When that attempt fails, they decide that DevSecOps is not for them. The real challenge here is to combine Agile and DevSecOps in the most effective way for an organization in IT development. DevSecOps can complement Agile. While Agile introduces collaboration, iteration, and continuous feedback into the development process, DevSecOps can strengthen quality control and delivery processes while securing the digital body of the software product being developed.

DevSecOps implementation is more difficult for organizations in industries that must meet strict cybersecurity requirements: healthcare, manufacturing, financial services, etc. Regulation in these industries is not flexible enough to allow companies to fully implement DevSecOps practices. That's why organizations often have to combine agile and secure DevOps with traditional approaches to software development.

There is also a technical challenge in implementing DevSecOps. Integrating traditional security tools such as antivirus software and firewalls, DevOps and DevSecOps tools into one system requires significant changes to an organization's infrastructure. CI/CD pipelines, binary libraries, static application security testing, software composition analysis, and many other tools typically come from different vendors, but IT developers need to organize them to work together. The best way to solve this problem is to carefully plan the implementation of DevSecOps tools and deploy selected tools one by one. Deploying and configuring them all at once is faster and may seem more convenient, but it will actually create chaos and lead to security vulnerabilities. Learning about these issues and planning to address them is key to a smooth DevSecOps implementation.

The clearest assessment of the prospects for software development and protection using DevSecOps methodology is provided by current results and analytical assessments of leading industry speakers [63-66]:

- forecast for the global DevSecOps market: $5.9 billion by 2023, at a compound annual growth rate of 31.2%;
- median average ratio of annual DevSecOps application scans is 2:7;
- 91% of firms consider security integration in software development;
- 11.5 times the rate of DevSecOps programs compared to traditional flaw and vulnerability remediation practices;
- 38% of mature DevOps firms are likely to integrate automated security;
- 50% higher revenue growth due to DevSecOps;
- by 2023, 80% of development teams are likely to adopt DevSecOps practices;
- about 24% of IT companies apply some elements of DevSecOps as of 2022;
- 2.5 times the performance of companies with integrated DevSecOps outperforms competitors that have not implemented the organizational and management scheme under study.

Thus, let us establish that DevSecOps is the next evolutionary step of software-digital production, which currently has unlimited prospects for implementation, operation and development.

5. Conclusions

For the present study goals were set in determining the prospects of implementation and integration of modern evolutionary solution in organizational and management solutions based on DevOps Security-blocks, which are the structural elements of the methodological basis of IT DevSecOps development.

The main barrier factors for transition from DevOps software-digital engineering and production organization system to IT-development with security integration (starting from creation of a digital code body of a developed software product to full support of security issues throughout the software lifecycle) - DevSecOps:

- The need for changes in corporate policy and system approach to software product development, which require deep integration of cybersecurity methods at all stages of the production process of software and digital development. Which is solved by current trends: the need of target customers for a secure and stable software application, the benefits of DevSecOps development, system-wide incentives to switch to DevSecOps organizational and management system;
- The technical challenges in integrating cybersecurity tools into the software-digital engineering and production system. This is solved by design, development and implementation of tools such as

SAST, DAST, IAST, etc., that allow to automate the process of cyber-security testing of a developed software product from the creation of the digital body (source code) to the use of third-party repositories and applications to release and further service support.

The analysis of current perspectives of DevSecOps technology development as well as the technical and technological advantages of the investigated organizational and management system point to the significant potential of the new system securityless methodology, which is the next step in the evolution of IT-development progress.

REFERENCES

[1] Pooja, Chandrakala, and L. K. Raju, “Developer’s roadmap to design software vulnerability detection model using different AI approaches,” IEEE Access, vol. 10, pp. 75637–75656, 2022. DOI: 10.1109/ACCESS.2022.3191115.

[2] Fu, Michael, et al. "VulRepair: A T5-Based Automated Software Vulnerability Repair." ESEC/FSE ’22, November 14–18, 2022, Singapore, 2022. DOI: 10.1145/3540250.3549098

[3] J. Zhou, “DevSecOps: Integrating security in DevOps for financial applications,” in The Future and FinTech, WORLD SCIENTIFIC, 2022, pp. 423–450.

[4] Kalouptsoglou, D. Tsoukalas, M. Siavvas, D. Kehagias, A. Chatzigeorgiou, and A. Ampatzoglou, “Time series forecasting of software vulnerabilities using statistical and Deep Learning models,” Electronics (Basel), vol. 11, no. 18, p. 2820, 2022. DOI: 10.3390/electronics11182820

[5] E. Iannone, R. Guadagni, F. Ferrucci, A. De Lucia, and F. Palomba, “The secret life of software vulnerabilities: A large-scale empirical study,” IEEE trans. softw. eng., pp. 1–1, 2022. DOI: 10.1109/TSE.2022.3140868.

[6] S. Forootani, A. Di Sorbo, and C. A. Visaggio, "An Exploratory Study on Self-Fixed Software Vulnerabilities in OSS Projects," in *2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, Honolulu, HI, USA, Mar. 15–18, 2022. IEEE, 2022. Accessed: Oct. 12, 2022. DOI: 10.1109/saner53432.2022.00023

[7] Tyagi *et al.*, “TheHuzz: Instruction fuzzing of processors using golden-reference models for finding software-exploitable vulnerabilities,” 2022. arXiv preprint arXiv:2201.09941. DOI: 10.48550/arXiv.2201.09941.

[8] Alqahtani, S. S. (2022). A study on the use of vulnerabilities databases in software engineering domain. *Computers & Security*, 116, 102661. doi.org/10.1016/j.cose.2022.102661.

[9] J. Iqbal, T. Firdous, A. K. Shrivastava, and I. Saraf, “Modelling and predicting software vulnerabilities using a sigmoid function,” *Int. J. Inf. Technol.*, vol. 14, no. 2, pp. 649–655, 2022. DOI: 10.1007/s41870-021-00844-2.

[10] R. R. Althar, D. Samanta, M. Kaur, D. Singh, and H.-N. Lee, “Automated risk management based software security vulnerabilities management,” IEEE Access, vol. 10, pp. 90597–90608, 2022. DOI: 10.1109/ACCESS.2022.3185069.

[11] M. Khurana, “Secure coding and software vulnerabilities in implementation phase of software development,” *ECS Trans.*, vol. 107, no. 1, pp. 7037–7045, 2022.

[12] F. Lu, M. Tang, Y. Bao, and X. Wang, “A survey of detection methods for software use-after-free vulnerability,” in Communications in Computer and Information Science, Singapore: Springer Nature Singapore, 2022, pp. 272–297. DOI: 10.1007/978-981-19-5209-8_19.

[13] L. Liu, Z. Li, Y. Wen, and P. Chen, “Investigating the impact of vulnerability datasets on deep learning-based vulnerability detectors,” PeerJ Comput. Sci., vol. 8, p. e975, 2022. DOI: 10.7717/peerj-cs.975.

[14] K. Scarfone and M. Souppaya, “[project description] software supply chain and DevOps security practices: Implementing a risk-based approach to DevSecOps (draft),” pp. 19–19, 2022.

[15] C. E. Otwell, “DevSecOps: Design science research of the DevSecOps technology stack, definitions, concepts, and improvements identified thereof,” 2022.

[16] X. Ramaj, “A DevSecOps-enabled framework for risk management of critical infrastructures,” in 2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2022. DOI: 10.1109/ICSE-Companion55297.2022.9793812.

[17] J. G. Süß, S. Swift, and E. Escott, “Using DevOps toolchains in Agile model-driven engineering,” Softw. Syst. Model., vol. 21, no. 4, pp. 1495–1510, 2022. DOI: 10.1007/s10270-022-01003-2.

[18] G. Sriraman, S. Sehar, and Suganya, “Agile and Touchless Automation in the software industry,” in 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022. DOI: 10.1109/ICACCS54159.2022.9785260.

[19] S. Gupta, M. Bhatia, M. Memoria, and P. Manani, “Prevalence of GitOps, DevOps in fast CI/CD cycles,” in 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), 2022. DOI: 10.1109/COM-IT-CON54601.2022.9850786.

[20] K. Chandran and M. Das Aundhe, “Agile or waterfall development: The Clementon Company dilemma,” *J. Inf. Technol. Teach. Cases*, vol. 12, no. 1, pp. 8–15, 2022. DOI: 10.1177/204388691987054.

[21] M. K. Kodmelwar, P. R. Futane, S. D. Pawar, S. A. Lokhande, and S. P. Dhanure, “A comparative study of software development waterfall, spiral and agile methodology,” J’ Positive School Psychology, vol. 6, no. 3, pp. 7013–7017, 2022.

[22] K. R. Halani and K. Jhajharia, “A quantitative study of waterfall and agile methodologies with the perspective of project management,” in Contemporary Challenges for Agile Project Management, IGI Global, 2022, pp. 111–133. DOI: 10.4018/978-1-7998-7872-8.ch007.

[23] M. A. Akbar, K. Smolander, S. Mahmood, and A. Alsanad, “Toward successful DevSecOps in software development

organizations: A decision-making framework," Inf. Softw. Technol., vol. 147, no. 106894, p. 106894, 2022. DOI: 10.1016/j.infsof.2022.106894.

[24] H. Yasar and S. E. Teplov, "DevSecOps in embedded systems: An empirical study of past literature," in Proceedings of the 17th International Conference on Availability, Reliability and Security, 2022. DOI: 10.1145/3538969.3544451.

[25] Z. Kanstantsin, "Multivocal literature review on the security of DevSecOp," AJRCoS, pp. 1–9, 2022. DOI: 10.9734/AJRCOS/2022/v14i230329.

[26] Gupta, "An Integrated Framework for DevSecOps Adoption," Int. j. comput. trends technol., vol. 70, no. 6, pp. 19–23, 2022. DOI:10.14445/22312803/IJCTT-V70I6P102.

[27] R. Naidoo and N. Möller, "Building software applications securely with DevSecOps: A Socio-technical perspective," Proc. Eur. conf. inf. warf. secur., vol. 21, no. 1, pp. 198–205, 2022. DOI: 10.34190/eccws.21.1.295.

[28] V. Tortoriello, "Definition of a DevSecOps Operating Model for software development in a large Enterprise," Politecnico di Torino, 2022.

[29] R. N. Rajapakse, M. Zahedi, M. A. Babar, and H. Shen, "Challenges and solutions when adopting DevSecOps: A systematic review," Inf. Softw. Technol., vol. 141, no. 106700, p. 106700, 2022. DOI: 10.1016/j.infsof.2021.106700.

[30] 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. DOI: 10.1109/MIUCC55081.2022.9781709.

[31] Landry, J. Schuette, and M. R. Schurgot, "DevSecOps for the transition of secure data sharing technology," in Open Architecture/Open Business Model Net-Centric Systems and Defense Transformation 2022, 2022. DOI: 10.1117/12.2619423.

[32] N. Harshitha, "Systematic Review Of Security Challenges in Devsecops For Cloud", International Journal of Research Publication and Reviews, vol. 3, no.6, pp 150-155, June 2022.

[33] M. Orosz, Introducing Agile/DevSecOps into the Space Acquisition Environment. Acquisition Research Program, 2022.

[34] S. L. Hoe, "Agile and DevSecOps," in Digital Transformation. Boca Raton: Auerbach Publications, 2022, pp. 91–104. Accessed: Oct. 12, 2022. [Online]. Available: https://doi.org/10.1201/9781003311393-7

[35] Y. Malhotra, "How You Can Implement Well-Architected 'Zero Trust' Hybrid-Cloud Computing Beyond 'Lift and Shift': Cloud-Enabled Digital Innovation at Scale with Infrastructure as Code (IaC), DevSecOps and MLops," SSRN Electronic Journal, 2022. Accessed: Oct. 12, 2022. [Online]. Available: https://doi.org/10.2139/ssrn.4131044

[36] R. Chandramouli, "Implementation of DevSecOps for a Microservices-based Application with Service Mesh," National Institute of Standards and Technology, Mar. 2022. Accessed: Oct. 12, 2022. [Online]. Available: https://doi.org/10.6028/nist.sp.800-204c

[37] S. Rathee and A. Chobe, "Securing Open Systems," in Getting Started with Open Source Technologies, Berkeley, CA: Apress, 2022, pp. 57–73.

[38] L. Ma, H. Yang, J. Xu, Z. Yang, Q. Lao, and D. Yuan, "Code analysis with static application security testing for python program," J. Signal Process. Syst., 2022. DOI: 10.1007/s11265-022-01740-z.

[39] Shiang-Jiun, P. Yu-Chun, M. Yi-Wei, C. Cheng-Mou, and T. Chi-Chin, "Trustworthy Software Development — practical view of security processes through MVP methodology," in 2022 24th International Conference on Advanced Communication Technology (ICACT), 2022. DOI: 10.23919/ICACT53585.2022.9728811.

[40] K. Neharika and R. G. Lennon, "Investigations into secure IaC practices," in Proceedings of Seventh International Congress on Information and Communication Technology, Singapore: Springer Nature Singapore, 2023, pp. 289–303. DOI: 10.1007/978-981-19-1610-6_25.

[41] Martin, "Practical security in high-velocity environments," 2022.

[42] M. Thakur, S. Hitefield, M. Mcdonnell, M. Wolf, R. Archibald, and L. Drane, & Mintz, B, "Towards a Software Development Framework for Interconnected Science Ecosystems" in Towards INTERSECT-SDK, 2022.

[43] "VOSviewer - Visualizing scientific landscapes," VOSviewer. [Online]. Available: https://www.vosviewer.com/. [Accessed: 12-Oct-2022].

[44] J. W. Beard, K. L. Siau, and J. Erickson, "Agile and DevOps in Digital Transformation – A Comparative Analysis," in PACIS 2022 Proceedings. 299, 2022.

[45] F. Almeida, J. Simões, and S. Lopes, "Exploring the benefits of combining DevOps and Agile," Future internet, vol. 14, no. 2, p. 63, 2022. DOI: 10.3390/fi14020063.

[46] O. H. Plant, J. van Hillegersberg, and A. Aldea, "Rethinking IT governance: Designing a framework for mitigating risk and fostering internal control in a DevOps environment," Int. J. Acc. Inf. Syst., vol. 45, no. 100560, p. 100560, 2022. DOI: 10.1016/j.accinf.2022.100560.

[47] R. Rose, "The Evolution Stage," in Software Development Activity Cycles, Berkeley, CA: Apress, 2022, pp. 123–148. DOI: 10.1007/978-1-4842-8239-7_7.

[48] Nisha T. N. and A. Khandebharad, "Migration from DevOps to DevSecOps: A complete migration framework, challenges, and evaluation," Int. j. cloud appl. comput., vol. 12, no. 1, pp. 1–15, 2022. DOI: 10.4018/IJCAC.2022010102.

[49] "DevSecOps services," Veritis.com, 05-Apr-2018. [Online]. Available: https://www.veritis.com/solutions/devops/devsecops-services/. [Accessed: 12-Oct-2022].

[50] R. G. Lennon, "DevOps best practices in highly regulated industry," in Proceedings of Seventh International Congress on Information and Communication Technology, Singapore: Springer Nature Singapore, 2023, pp. 567–585. DOI: 10.1007/978-981-19-1607-6_51.

[51] S. R. Goniwada, "Enterprise Cloud Native Automation," in Cloud Native Architecture and Design, Berkeley, CA: Apress, 2022, pp. 523–553. DOI: 10.1007/978-1-4842-7226-8_14.

[52] J. Faustino, D. Adriano, R. Amaro, R. Pereira, and M. M. da Silva, "DevOps benefits: A systematic literature review," Softw. Pract. Exp., vol. 52, no. 9, pp. 1905–1926, 2022. DOI: 10.1002/spe.3096.

[53] M. Shameem, "A systematic literature review of challenges factors for implementing DevOps practices in software development organizations: A development and operation teams perspective," Evolving Software Processes. Wiley, pp. 187–199, 22-Jan-2022. DOI: 10.1002/9781119821779.ch9.

[54] W. R. Nichols and C. L. Miller, "Automated Data for DevSecOps Programs," Dtic.mil. [Online]. Available: https://apps.dtic.mil/sti/pdfs/AD1168421.pdf. [Accessed: 12-Oct-2022].

[55] K. A. Rassmann, "A Goal, Question, Metric approach to Coherent Use integration within the DevOps lifecycle." Digital Repository at the University of Maryland, 2022.

[56] J. Díaz et al., "Harmonizing DevOps taxonomies - A grounded theory study," Research Square, 2022.DOI: 10.21203/rs.3.rs-1946752/v1.

[57] Katrenko and A. Beliba, "Introduction to DevSecOps with AWS: How to integrate security into DevOps," Apriorit, 02-Dec-2021. [Online]. Available: https://www.apriorit.com/dev-blog/530-delivering-devsecops-aws. [Accessed: 12-Oct-2022].

[58] T. Scanlon and J. Morales, "Revelations from an agile and DevSecOps transformation in a large organization: An experiential case study," in Proceedings of the International Conference on Software and System Processes and International Conference on Global Software Engineering, 2022. DOI: 10.1145/3529320.3529329.

[59] Woody, C. Wallen, C. Alb, and M. Bandor, "SOFTWARE ENGINEERING INSTITUTE | CARNEGIE MELLON UNIVERSITY distribution statement A: Approved for public release; Distribution is unlimited acquisition security framework (asf): integration of supply chain risk management across the devsecops lifecycle," Dtic.mil, 2022. [Online]. Available: https://apps.dtic.mil/sti/pdfs/AD1168407.pdf. [Accessed: 12-Oct-2022].

[60] W. Hallock, "DevSecOps in Practice", 2022. *CrossTalk*. cutt.ly/iXPefPe. [Accessed: 12-Oct-2022].

[61] K. Siau et al., "Information systems analysis and design: Past revolutions, present challenges, and future research directions," Commun. Assoc. Inf. Syst., vol. 50, no. 1, pp. 835–856, 2022. DOI: 10.17705/1CAIS.05037.

[62] Sriraman, S. Sehar, and Suganya, "Agile and Touchless Automation in the software industry," in 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022. DOI: 10.1109/ICACCS54159.2022.9785260.

[63] O. Dzyad and D. Starodub, "Economic losses and international methods against cybercrimes and ways to oppose cybercrimes worldwide," Efektyvna ekonomika, no. 1, 2022. DOI: 10.32702/2307-2105-2022.1.91.

[64] E. Indriasari, Binus University, H. Prabowo, F. L. Gaol, B. Purwandari, and Jakarta 11530, Indonesia, "Adoption of design thinking, agile software development and co-creation: A qualitative study towards digital banking innovation success," International Journal of Emerging Technology and Advanced Engineering, vol. 12, no. 1, pp. 111–128, 2022. DOI: 10.46338/ijetae0122_11.

[65] Sahid, Y. Maleh, and S. Mounir, "Towards an agile itsm maturity framework for organizations : A case study," EDPACS, pp. 1–21, 2022. DOI:10.1080/07366981.2022.2045541.

[66] S. Sumitra, "An analysis of cybersecurity for business enterprises," University of Alberta Library, 2022. DOI: 10.7939/r3-tjyd-k045.