Journal of Information Security, 2024, 15, 218-231
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

A Study on the Challenges of Human-Centric Cyber-Security and the Guarantee of Information Quality

Mohammed Hussein Kurdi[1], Mohsen Denden[2,3], David Paul[4]

[1]Department of Cyber-Security, National Events Center, Riyadh, Saudi Arabia
[2]Department of Computer and Information Technologies, Technical College of Telecommunication and Information Riyadh TCTI, Technical and Vocational Training Corporation TVTC, Riyadh, Saudi Arabia
[3]Department of Computer Science, Higher Institute of Applied Sciences of Sousse, University of Sousse, Sousse, Tunisia
[4]School of Science & Technology, University of New England, Armidale, Australia
Email: mohsen@cti.edu.sa, mohkurdi4@gmail.com, mohsen.denden@isi.rnu.tn, dpaul4@une.edu.au

How to cite this paper: Kurdi, M.H., Denden, M. and Paul, D. (2024) A Study on the Challenges of Human-Centric Cyber-Security and the Guarantee of Information Quality. *Journal of Information Security*, **15**, 218-231.
https://doi.org/10.4236/jis.2024.152013

Received: March 23, 2024
Accepted: April 21, 2024
Published: April 24, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/
Open Access

Abstract

Information security and quality management are often considered two different fields. However, organizations must be mindful of how software security may affect quality control. This paper examines and promotes methods through which secure software development processes can be integrated into the Systems Software Development Life-cycle (SDLC) to improve system quality. Cyber-security and quality assurance are both involved in reducing risk. Software security teams work to reduce security risks, whereas quality assurance teams work to decrease risks to quality. There is a need for clear standards, frameworks, processes, and procedures to be followed by organizations to ensure high-level quality while reducing security risks. This research uses a survey of industry professionals to help identify best practices for developing software with fewer defects from the early stages of the SDLC to improve both the quality and security of software. Results show that there is a need for better security awareness among all members of software development teams.

Keywords

Cyber Security, Development Methodology, Information Quality, Human-Centric, SDLC, Quality Assurance

1. Introduction

Nowadays, secure IT is critical to any organization. A secure system performs

the functions for which it was designed and refrains from performing functions for which it was not designed. The primary criteria for software security are data confidentiality, integrity, and availability (CIA), though other criteria, such as authentication, authorization, privacy protection, security management, access control, and auditing are important too. There are few problems today that have generated more corporate concern than cyber-security risk [1]. With the potential to destroy a company, including the potential loss of customers and withdrawal of shareholders, complaints and lawsuits from the affected parties, and undesired media coverage, the threat of cyber risk is pervasive. No business or enterprise is resistant to cyber risks, which highlights the importance for organizations to address cyber-security risks from the top down. Presently, mitigating or addressing cyber-security risk has evolved such that it cannot only be confined to the realm of the IT department; rather, it is now an overall organization's responsibility to ensure that every department and all users adhere to the redefined security measures.

In this study, we examine the relationship between Quality Assurance (QA) and cyber-security by surveying people who work in either or both of these areas.

SDLC (Software Development Lifecycle) involves the integration of security processes throughout software development and planning. This process makes it possible to group security requirements as well as functional requirements. Quality control is a mechanism for reviewing the design, production, and code phases to ensure reliable quality. Security and quality control are two key elements for managing cyber-security risks. Researchers consider that enhanced security in the System Software Development Life-cycle (SDLC) can reduce the risks of a cyber-attack [2]. The findings are that frameworks for both cyber-security and quality assurance teams are available and already covered in some of the literature [3] [4], though are typically considered separately. Building a framework between the security team and the quality team should help an organization reduce its risks. Security engineering necessitates the development of a new mindset that is both preventive and reactive and takes into consideration risk calculation and experiment [5].

To improve the security awareness of the software development team, training sessions and workshops focused on security best practices, common vulnerabilities, and threat awareness should take place regularly. Moreover focusing on the following topics: secure coding, SSDLC, DevOpsSec, secure code review, can improve their awareness. Performing the testing development life cycle is also important to improve the development team by acknowledging the teasing findings and learning from the mistakes detected while developing software.

This research aims to identify existing software security best practices that are used to develop software with fewer defects from the early stages of the SDLC [6] which improves both the quality of the software and its security. This research also aims to identify whether Governance Risk and Compliance can positively help in assuring both security and quality of software and explores how security

can be incorporated into software and how secure software can be developed. A survey of current IT professionals is used to support this aim. The goal of this paper is to assess how security and quality assurance methods and procedures are integrated into software development phases to ensure the quality of the final software product. It does not provide precise enumerations or metrics of specific benefits and risks but concentrates on developing a detailed and comprehensive image of the software implementation security paradigms currently practiced by real-world organizations (both medium-sized organizations and start-ups). For this purpose, this research aims for the following objectives:

- Articulate the necessity of software security.
- Appreciate security from the perspective of software development and usage, and how software security varies from other varieties of security such as information security.
- Examine the present software development procedures used for software security.
- Investigate whether there are issues with the existing approach to software security. If yes, try to determine the underlying issues.

The remainder of this paper is structured as follows. Related works are illustrated in Section II. The problem description is elaborated in Section III. The methodology and tests are presented in Section IV, and details of the conducted survey are provided in Section V. Finally results and discussion, and conclusions are respectively presented in Sections VI and VII.

2. Related Works

Information security and quality management are often considered as two different fields. However, since business conduct is constantly evolving, organizations must be mindful of how information security may affect quality control problems. Studies suggest that many companies and enterprises have suffered huge losses because of inadequate security operations and governance [7]. It has also been observed that IT operations are often not in sync with the security or overall governance of the company, which creates problems in terms of decision-making, procurement, and implementation [8] [9]. On the other hand, research shows that human beings are typically the weak link in cyber-security; the psychology of an employee can affect their decisions in front of a machine [10] [11]. Therefore it is recommended that IT and security governance be more aligned in terms of planning and strategizing. There is a need for clear standards, frameworks, processes, and procedures [12] to be followed by organizations to ensure high-level quality while reducing security risks [13]. A common model of security includes Confidentiality, Integrity, and Availability (CIA) as fundamental building blocks [14]. Confidentiality relates to preserving data privacy, *i.e.*, preventing unwanted revelation of information. Integrity refers to the process of ensuring the accuracy and completeness of data, *i.e.*, preventing illegal modification of data. Availability refers to the process of ensuring that information is accessible to authorized individuals [14]. Any software that meets these three cri-

teria may be considered secure, though there are often other important security considerations.

One approach to help ensure both software security and quality is testing. Software testing is the process of determining whether a software implementation's features are reliable with the design [15]. Functional testing tests software against specified requirements or functions, while non-functional testing is concerned with non-functional aspects, such as performance, usability or reliability. Once a system has been deployed, maintenance testing (such as regression) can be used to ensure the system continues to work correctly as the system evolves and adapts. Security vulnerability testing and security functional testing are two types of software security testing [16]. However, organizations frequently view security as a post-development effort [17]. During the predevelopment and development phases, security is often not considered (or tested). Organizations are unaware (or ignore the fact) that "software security is an emergent attribute, not a feature, of a whole system" [17]. After software development is complete, organizations attempt to include security as a patch. Additionally, corporations invest significant resources in obtaining effective firewalls and antivirus software, often believing that this exterior layer is sufficient to keep software secure.

These methods are ineffective [18], and organizations continue to incur significant financial losses because of the exploitation of security vulnerabilities [19]. In literature, many studies have considered particular cases to consolidate security information for specific areas such as healthcare [20] [21] [22]. Research has been developed to minimize the risk of data transfer between organizations themselves and between companies and their cloud environment [23]. This work asserts that data protection can be better ensured by reducing the number of data migrations [24].

3. Problem Description

This study will try to answer the question: How can organizations effectively align 'IT governance' and 'Security' in practice for information security governance? To answer this question, we will explore three principles to obtain insights into current practices, namely the 'process' of 'governance' and 'security'. These concepts will be analyzed to obtain more understanding of how 'governance' is embedded in IT Security (IS). IS Governance (ISG) is defined as developing and maintaining a control system to ensure it supports CIA. ISG mainly covers three areas: IT governance, corporate governance, and information security. Researchers can classify ISG's area of coverage as risk management, implementing effective IT controls, and building a security culture in the organization through training and awareness building [25].

Many application providers are dangerously ignorant of the actual security problems that consumers face, leading to a false sense of security among users and a lack of urgency among vendors. Many consumers and vendors incorrectly think that protection is an issue with the operating system or with the network

perimeter and firewall, but this is not the case [26]. In other words, if we write applications that can be exploited by potentially malicious users, either inside or outside the firewall, our program could be targeted. Just adding a firewall should not be seen as a valid approach to ensure security in the SDLC.

Nowadays, quality is a critical issue in product development. If a business is facing stiff competition from other vendors, the quality of the software becomes a competitive factor. Further, software quality is critical when dealing with systems that must never cease operating or cannot fail, such as a car, an airplane, or a nuclear power plant. The expected high quality in these kinds of systems is critical, as the costs of failure can be unacceptable. There is a need to prioritize quality when producing software, as practically every machine is now controlled by software. CIOs and IT teams are under intense business pressure to modernize applications, enhance customer experiences, and automate routines.

Development methodologies such as Agile [27], Lean [28] and DevOps [29] support philosophies, practices, tools, and automation that enable application engineering teams to accomplish these goals and produce business value with a higher level of quality and speed [30]. Currently, expert and experienced software developer teams are continuously employing automatic testing and code-based deployment integration and continuous delivery (CI/CD) mechanisms to ensure fast deployment of the product [30]. Change management and incident management are often done using Agile development strategies [31] to expedite the process of determining the underlying cause of production issues.

Nonetheless, security concerns persist in software engineering. According to ESG's Modern Application Development Security Study [32], just 36 percent of respondents approved of their application security program by giving a rating of at least 8.5 out of 10, while 66 percent indicated their application security solutions protect less than 75 percent of their code base, and 48 percent admitted to regularly pushing vulnerable code into production. These security gaps are not due to a scarcity of technology, consultancy, or security service providers. Therefore, the key to achieving business value while avoiding security risks in software development is to properly define and communicate security principles to software development teams.

4. Methodology and Tests

Security covers various interests and activities. Thus, information security has no single methodology that is followed by researchers when performing research activities. The broad range of methods available to information security researchers may influence how "usable" data in a security research publication may be for other information security researchers. Methodologies have a key role in assuring the consistency of analysis, as well as the ability to incorporate findings meaningfully with one another. Without a reliable and reproducible methodology, it becomes difficult for the reader to determine the legitimacy and accuracy of the results. An exploratory research methodology was used in this

study. This is because this research aims to elicit relevant insights into how risk governance and the adoption of best security practices during software development can enable organizations to develop secure and quality software solutions. We have designed a survey questionnaire to capture the current practice and find the gaps between existing theory and practice. We have surveyed current IT professionals to obtain their opinions and suggestions. However, we would like to highlight that the objective of this research is not to generate any numerical data or conduct a detailed quantitative analysis. Rather, it presupposes conclusions based on subjective evaluation of opinions.

Table 1 represents the different abbreviations used in this paper.

5. Survey Design

The survey consisted of 4 sections containing a total of 42 questions for which participants had to provide a single answer, multiple answers, and/or a response to a 5-point Likert scale.

Section 1 contained 11 general questions about participant demographics, including information such as the participant's age, gender, years of work experience, educational level, position level, organization and team size, and the organization type which sector it is. This was to allow an understanding of each participant's level of experience, organization and team size and their knowledge about the topics related to this study. The study did not require any sensitive information.

Section 2 contained 12 questions for evaluating four factors including Software QA, testing, SDLC methodologies and awareness. There are many methodologies used in different organizations and those used depend on an individual organization's governance, security culture and business needs.

Section 3 contained 15 questions for reviewing five factors including software security, security testing, secure SDLC, security standards and awareness. This allows an understanding of the maturity level of the individual and the organization

Table 1. Nomenclatures.

Symbols	Description
ALM	Application Life Cycle Management
CIA	Confidentiality, Integrity, and Availability
CIO	Chief Information Officer
ESG	Enterprise Strategy Group
GRC	Governance, Risk management and Compliance
IS	IT Security
ISG	IT Security Governance
QA	Quality Assurance
RFT	IBM Rational Functional Tester
SDLC	System Software Development Lifecycle

as to whether security rules and other quality practices exist or are further implemented for areas such as IT assets, functionalities, software performance, change and configuration management, testing, and quality management.

Section 4 contained two questions for assessing two factors including procedure and frameworks.

With ethics approval from the University of New England, the aim was to survey a minimum of 50 participants. This number of participants was to ensure a comprehensive response that allows for an adequate understanding of current practices for at least a subset of the wider population. The inclusion criteria for the study were:

- Age 18+
- Works in IT
- Experience with cyber-security, quality assurance, and/or software project management/development/testing
- Has access to social media

Each participant recruited through a social media post completed a short structured online survey that was timed to take about 20 minutes to complete. While such a convenience sample does not necessarily represent the entire population, issues identified in this limited sample are likely to exist in wider parts of the industry.

6. Results and Discussion

The survey received 59 valid responses, with 57 of the respondents indicating they were male and 2 indicating they were female. This indicates that the field of information security is male-dominated, matching common results [33]. Most of the participants were from Saudi Arabia and Australia and belonged to international companies.

The majority of the respondents were between the ages of 28 and 47 inclusive. Only 10% of participants were aged between 18 - 27 and 48 - 57, and only 3% were aged 58 or above. These statistics suggest that currently established security professionals are aged between 48 - 57 years. This may be because older employees have moved to higher positions. Of the 59 respondents, 45.76% work with government organizations, with 25.42% work in semi-government organizations and another 25.42% in private organizations, with the remaining 3.39% working in other unspecified organizations. From this finding, we can deduce that the respondents are from diverse work backgrounds, and thus, the responses concerning the research topic are reasonably comprehensive. While we should be careful to not draw industry-wide conclusions from the collected data, any trends identified are likely to affect a larger population than just those surveyed.

6.1. Development Methodologies

From the survey results (see **Figure 1**), it was found that the largest percentage

of respondents (33.59%) were working in organizations that use an agile development methodology as their primary software development approach, with around 15% working with organizations that use more rigid approaches such as Waterfall [34] as their primary software development methodology.

6.2. SDLC Security Practices

Figure 2 represents the participants' organizations' testing practices during the application development process. Considering these responses, it is clear that virtually all organizations conduct some kind of software test during actual application development. Participant responses indicate that functional testing has the highest priority in organizations, followed by non-functional and maintenance testing respectively. This satisfies one of the aims of this research which seeks to determine the best practices that software developers can employ during software development phases to ensure secure software.

Moreover, the survey results presented in Figure 3 show that many organizations are utilizing tools to automate their software testing activities. The most

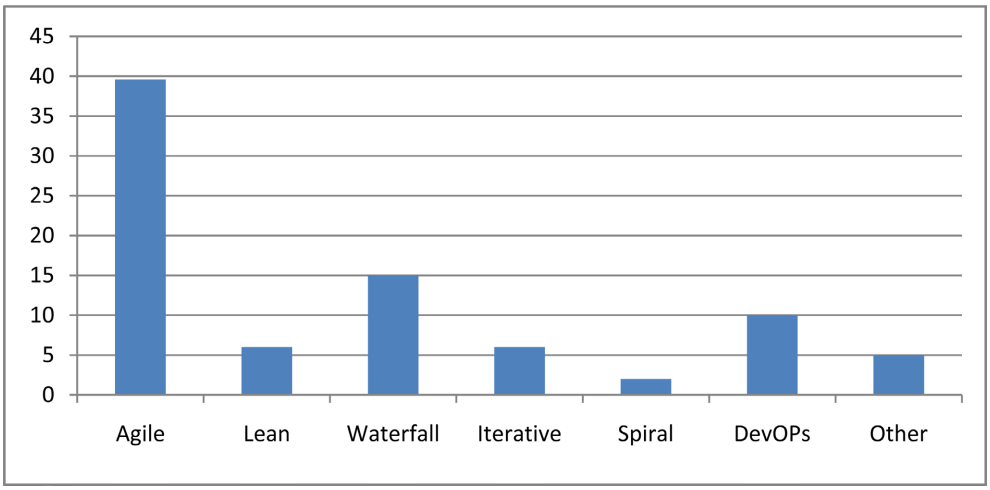

Figure 1. Organisation's usage of development methodologies.

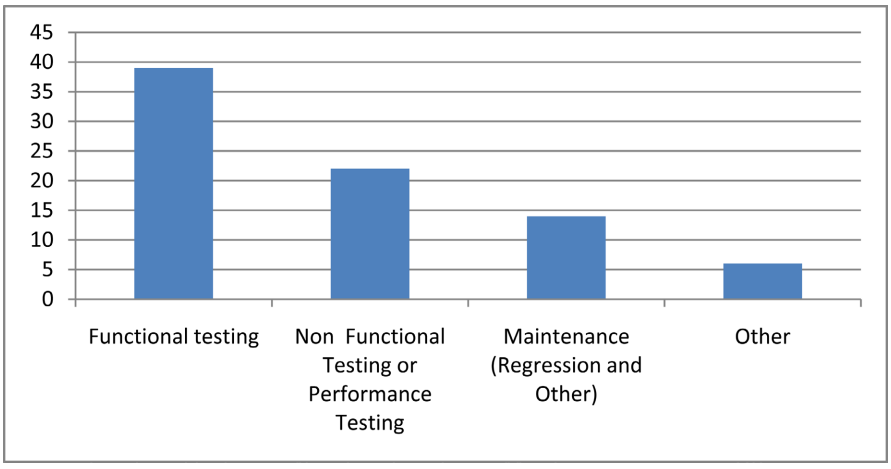

Figure 2. Organisations' software testing practices.

popular among those surveyed was Application Life Cycle Management (ALM), while other testing tools such as SoapUI, Selenium, Apache JMeter, and IBM Rotational Functional Tester (RFT) were also relatively popular.

6.3. Quality Assurance

Organizations always strive to ensure that their final product meets required standards, and more often than not there are processes in place for this purpose. Most of the respondents acknowledged that their corresponding organizations always carry out security awareness programs for their developers, though other stakeholders, such as testers and dedicated security teams, are less likely to receive dedicated training (see Figure 4).

This suggests that organizations believe that to develop secure software, the development team must be cyber-security aware with appropriate training.

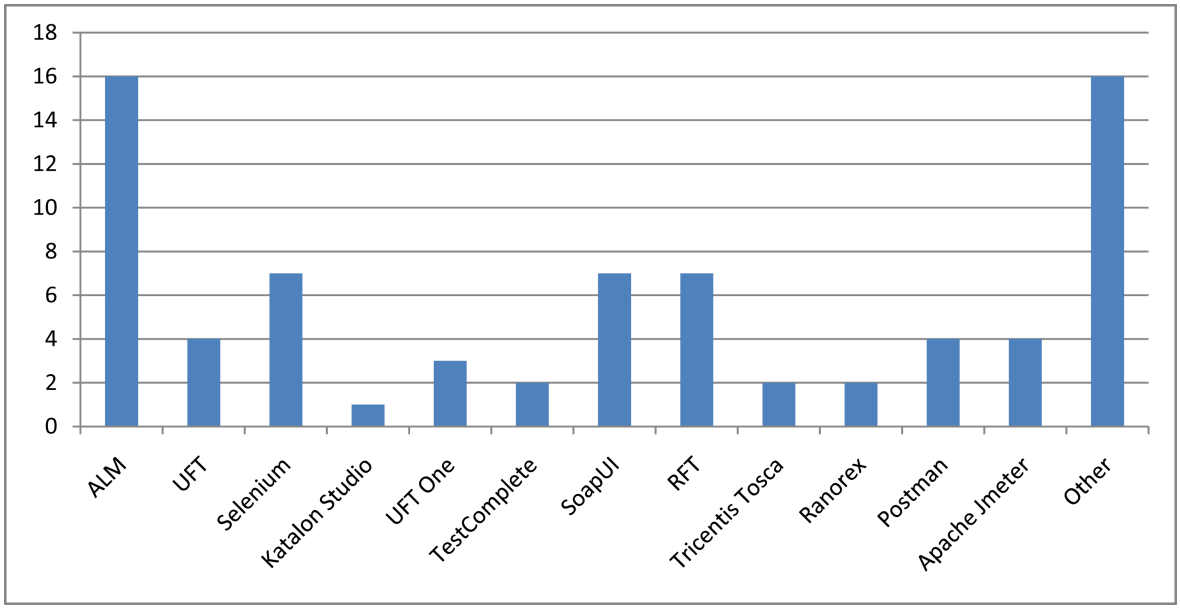

Figure 3. Testing tools usage.

Figure 4. Security awareness programs for quality assurance.

However, the fact that other stakeholders are offered security training at such high rates suggests that software quality is closely tied to the security maturity and security awareness of all involved parties (beyond just the development team).

The majority of businesses strive to provide the greatest product possible to the end user. Customer-centricity, however, is sometimes overlooked to stay up with market expectations and deliver the latest innovations as rapidly as feasible. When software is developed, the goal is for the development, design, distribution, and delivery processes to be seamless. However, this is an uncommon occurrence [35]. Software testing and quality assurance are two distinct processes. The former is concerned with locating faults, vulnerabilities, and other flaws, while the latter addresses non-technical usability difficulties.

Software quality assurance testing is focused on giving the best possible solution to the customer. In a QA context, a software defect is not limited to bugs; it can encompass any issue that negatively impacts the end-user experience, from bad navigation to slow page load times or unclear web copy. A promising approach to decrease defect risk while optimizing end-user experience is to incorporate software and quality assurance testing throughout the development process. For example, from **Figure 5**, respondents agree that to ensure a secure SDLC in the requirements phase, the security team must perform risk assessment, compliance analysis, and/or security requirement elicitation.

7. Conclusion

Results of the survey in this research suggest that developing high-quality software involves the participation of everyone. Anyone who has experienced the difficulty of designing a software system understands that the task of building high-quality software is more difficult than it first appears, and certainly more difficult than most clients believe. Security should be considered from the very

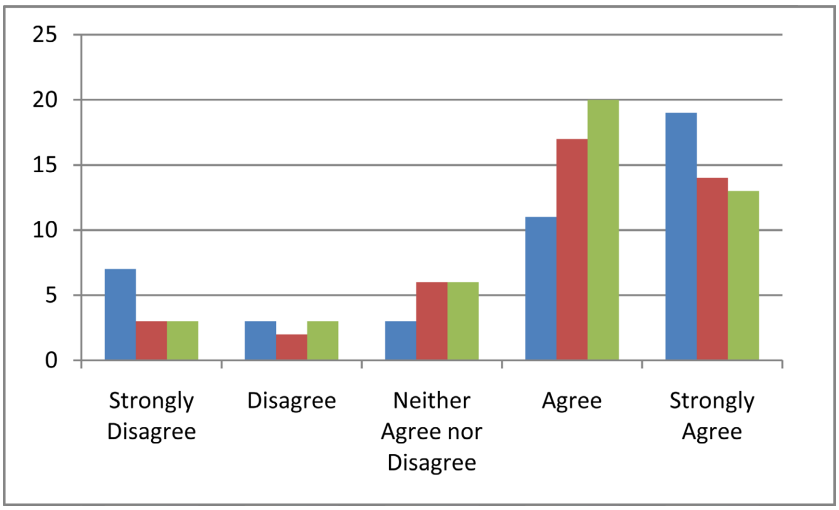

Figure 5. Security enhancing processes during the requirements phase of the software SDLC.

first stages of the SDLC and all people, from requirement engineers to software developers, should be aware of current software security challenges, including functional requirements and non-functional security issues. Security understanding should be more technical and in-depth among all team members.

To fulfill the software's security objectives, security must be an obligation tightly coupled throughout the SDLC. If security concerns are addressed and resolved appropriately by the requirement engineer, it enables the system software designer to create more secure software and the programmer to write secure code. By addressing security concerns earlier, an implementation engineer will be able to implement and configure software more safely. The deployment engineer will then be better able to safeguard the software deployments in open environments. This overall process will result in better-quality software.

The path to functional software is fraught with dangers, and the likelihood of failure is high. Much of the complexity and challenge inherent in building software stems from its intangibility; one cannot simply draw a design or define its physical properties. While the process of producing software is heavily influenced by known engineering disciplines, several aspects of the process remain unexplained.

Acknowledgements

This work was approved by the Human Research Ethics Committee of the University of New England under the No HE21-084. The authors extend their appreciation to all participants of this project.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Coburn, A., Leverett, E. and Woo, G. (2018) Solving Cyber Risk: Protecting Your Company and Society. John Wiley & Sons, Hoboken.

[2] Salin, H. and Lundgren, M. (2022) Towards Agile Cybersecurity Risk Management for Autonomous Software Engineering Teams. *Journal of Cybersecurity and Privacy*, **2**, 276-291. https://doi.org/10.3390/jcp2020015

[3] Cissé, M. (2019) An ISO 27001 Compliance Project for a Cyber Security Service Team. *Cyber Security*: *A Peer-Reviewed Journal*, **2**, 346-359.

[4] Harmer, G. (2014) Governance of Enterprise IT Based on COBIT®5. IT Governance Publishing, Ely.
https://www.itgovernance.co.uk/download/governance-of-enterprise-it-based-on-cobit-5-book-sample.pdf
https://doi.org/10.2307/j.ctt7zsxfv

[5] Blokdyk, G. (2017) Java Machine Learning Complete Self-Assessment Guide. CreateSpace Independent Publishing Platform, North Charleston.
https://dl.acm.org/doi/10.5555/3164673

[6] Alshammari, B., Fidge, C. and Corney, D. (2016) Developing Secure Systems: A

Comparative Study of Existing Methodologies. *Lecture Notes on Software Engineering*, **4**, 139-146.

[7] Bahl, S. and Wali, O.P. (2014) Perceived Significance of Information Security Governance to Predict the Information Security Service Quality in Software Service Industry: An Empirical Analysis. *Information Management & Computer Security*, **22**, 2-23. https://doi.org/10.1108/IMCS-01-2013-0002

[8] Bokhari, S.A.A. and Myeong, S. (2023) The Impact of AI Applications on Smart Decision-Making in Smart Cities as Mediated by the Internet of Things and Smart Governance. *IEEE Access*, **11**, 120827-120844. https://doi.org/10.1109/ACCESS.2023.3327174

[9] Abed-Alguni, B.H. and Paul, D. (2022) Island-Based Cuckoo Search with Elite Opposition-Based Learning and Multiple Mutation Methods for Solving Optimization Problems. *Soft Computing*, **26**, 3293-3312. https://doi.org/10.1007/s00500-021-06665-6

[10] Alkhalifah, A. and Denden, M. (2023) Investigating the Impact of Covid-19 on the Morale of Deaf and Hearing-Impaired Students in Saudi Arabia Technical Colleges: Lessons Learned and Future Implications. *Journal for Educators, Teachers and Trainers*, **14**, 420-428. https://doi.org/10.47750/jett.2023.14.03.051

[11] Denden, M. and Alkhalifah, A. (2023) Assessing the Impact of Covid-19 on the Psychology of Saudi Technical College Students: Lessons and Tips. *Creative Education*, **14**, 518-529. https://doi.org/10.4236/ce.2023.143036

[12] Jemmali, M., Denden, M., Boulila, W., Srivastava, G., Jhaveri, R.H. and Gadekallu, T.R. (2022) A Novel Model Based on Window-Pass Preferences for Data Emergency Aware Scheduling in Computer Networks. *IEEE Transactions on Industrial Informatics*, **18**, 7880-7888. https://doi.org/10.1109/TII.2022.3149896

[13] Alsmadi, I., Easttom, C., Tawalbeh, L. and Alsmadi, I. (2020) It Risk and Security Management. In: Alsmadi, I., Easttom, C. and Tawalbeh, L., Eds., *the NICE Cyber Security Framework*: *Cyber Security Management*, Springer, Cham, 55-78. https://doi.org/10.1007/978-3-030-41987-5

[14] Andress, J. (2014) The Basics of Information Security: Understanding the Fundamentals of InfoSec in Theory and Practice. Syngress, Rockland.

[15] Chakraborty, R.S., Zheng, Y. and Bhunia, S. (2016) Obfuscation-Based Secure Soc Design for Protection Against Piracy and Trojan Attacks. In: Chang, C.H. and Potkonjak, M., Eds., *Secure System Design and Trustable Computing*, Springer, Cham, 269-299. https://doi.org/10.1007/978-3-319-14971-4_8

[16] Fatima, A., Khan, T.A., Abdellatif, T.M., Zulfiqar, S., Asif, M., Safi, W., Al Hamadi, H. and Al-Kassem, A.H. (2023) Impact and Research Challenges of Penetrating Testing and Vulnerability Assessment on Network Threat. 2023 *International Conference on Business Analytics for Technology and Security* (*ICBATS*), Dubai, 7-8 March 2023, 1-8. https://doi.org/10.1109/ICBATS57792.2023.10111168

[17] Khan, K.M. (2012) Developing and Evaluating Security-Aware Software Systems. IGI Global, Hershey. https://doi.org/10.4018/978-1-4666-2482-5

[18] Dowd, M., McDonald, J. and Schuh, J. (2006) The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities. Pearson Education, Upper Saddle River.

[19] Wiltshire, I., Adapa, S. and Paul, D. (2023) Pandemic Speed: Accelerating Innovation in Cyber Security. In: Adapa, S., McKeown, T., Lazaris, M. and Jurado, T., Eds., *Small and Medium-Sized Enterprises, and Business Uncertainty. Palgrave Studies in Global Entrepreneurship*, Palgrave Macmillan, Singapore, 151-172.

https://doi.org/10.1007/978-981-99-4844-4_9

[20] Kim, L. (2022) Cybersecurity: Ensuring Confidentiality, Integrity, and Availability of Information. In: Hübner, U.H., Mustata Wilson, G., Morawski, T.S. and Ball, M.J., Eds., *Nursing Informatics. Health Informatics*, Springer, Cham, 391-410. https://doi.org/10.1007/978-3-030-91237-6_26

[21] Melhim, L.K.B. (2023) Intelligent Surveillance Drone System for Health Care Enhancement in a Smart City. *Communications in Mathematics and Applications*, **14**, 551-559. https://doi.org/10.26713/cma.v14i2.2153

[22] Eljack, S., Jemmali, M., Denden, M., Sadig, M.A., Algashami, A.M. and Turki, S. (2024) Intelligent Solution System for Cloud Security Based on Equity Distribution: Model and Algorithms. *Computers, Materials & Continua*, **78**, 1461-1479. https://doi.org/10.32604/cmc.2023.040919

[23] Mohsen, D., Ghannay, N. and Samet, A. (2009) A Half Hollow Cylindrical Antenna (HHCA) Analysis Using the CFDTD Algorithm. *Progress in Electromagnetics Research C*, **11**, 51-60. https://doi.org/10.2528/PIERC09090804

[24] Eljack, S., Jemmali, M., Denden, M., Turki, S., Khedr, W.M., Algashami, A.M. and ALsadig, M. (2023) A Secure Solution Based on Load-Balancing Algorithms Between Regions in the Cloud Environment. *PeerJ Computer Science*, **9**, e1513. https://doi.org/10.7717/peerj-cs.1513

[25] Ghannay, N., Denden, M., Romdhani, F. and Samet, A. (2008) A Novel Technique for Calculating Moment Method Impedance Matrix. *IEEE Mediterranean Microwave Symposium* 2008 *Symposium*, Damascus, 14-16 October 2008, 77-80.

[26] Ohki, E., Harada, Y., Kawaguchi, S., Shiozaki, T. and Kagaya, T. (2009) Information Security Governance Framework. *Proceedings of the First ACM Workshop on Information Security Governance*, Chicago, 13 November 2009, 1-6. https://doi.org/10.1145/1655168.1655170

[27] Tashtoush, Y.M., Darweesh, D.A., Husari, G., Darwish, O.A., Darwish, Y., Issa, L.B. and Ashqar, H.I. (2021) Agile Approaches for Cybersecurity Systems, IoT and Intelligent Transportation. *IEEE Access*, **10**, 1360-1375. https://doi.org/10.1109/ACCESS.2021.3136861

[28] Rossi, M., Taisch, M. and Terzi, S. (2012) Lean Product Development: A Five-Steps Methodology for Continuous Improvement. 2012 18*th International ICE Conference on Engineering, Technology and Innovation*, Munich, 18-20 June 2012, 1-10. https://doi.org/10.1109/ICE.2012.6297704

[29] Jabbari, R., Bin Ali, N., Petersen, K. and Tanveer, B. (2016) What Is Devops?: A Systematic Mapping Study on Definitions and Practices. *Proceedings of the Scientific Workshop Proceedings of XP*2016, Edinburgh, 24 May 2016, 1-11. https://doi.org/10.1145/2962695.2962707

[30] Grembi, J. (2008) Secure Software Development: A Security Programmer's Guide. Cengage Learning, Boston.

[31] Siddiqi, M.A. and Pak, W. (2021) An Agile Approach to Identify Single and Hybrid Normalization for Enhancing Machine Learning-Based Network Intrusion Detection. *IEEE Access*, **9**, 137494-137513. https://doi.org/10.1109/ACCESS.2021.3118361

[32] Gruber, D. (2020) Modern Application Development Security. Enterprise Strategy Group, Newton.

[33] Gallivan, M.J. (2006) Diversity in Studying Gender and IT. In: Trauth, E.M., Ed., *Encyclopedia of Gender and Information Technology*, IGI Global, Hershey, 216-223. https://doi.org/10.4018/978-1-59140-815-4.ch034

[34] McCormick, M. (2012) Waterfall vs. Agile Methodology. MPCS Inc., Newburgh.

[35] Brooks, F.P. (1995) The Mythical Man-Month. Anniversary Edition, Addison-Wesley Longman Publishing Co. Inc., Boston.