Contemporary Research Analysis Journal

e-ISSN: 3050-5909
p-ISSN: 3050-5895

Open Access

Volume 03 Issue 01 January 2026

CrossRef DOI: 10.55677/CRAJ/03-2026-Vol03I01

Page no: 28-43

Assessing the Methods and Difficulties of Incorporating Software Security Testing into System Development Process in Tanzania's Public Sector

Stephen M. Wangwe*[1], George S. Oreku[2]

[1]Personal Data Protection Commission (PDPC), Dodoma, Tanzania

[2]Open University of Tanzania (OUT), Dar Es Salaam, Tanzania

ABSTRACT: In recent years, the public sector in Tanzania, like many others globally, has continued to embrace digital transformation, facilitating the modernization of online public services and enhancing government operations. However, the complexity and interconnectedness of these systems raise concerns about their vulnerability to cyber threats, as security testing is often neglected during the development process. This oversight can lead to the deployment of systems with security flaws that compromise data, disrupt services, and erode public trust. While previous studies have focused on integrating security into the software development life cycle, none has specifically evaluated the challenges of embedding security testing into public sector systems. Therefore, this study aims to fill this gap by examining the current practices and challenges of integrating security testing early in the development lifecycle within Tanzania's public sector. Using a mixed-method approach, data were collected from 104 ICT managers, security officers, software developers, and systems administrators through surveys, with analysis focusing on practices and perceptions using descriptive statistics. The findings reveal that only 6.7% of public organizations have fully integrated security testing, while over 38% report little to no integration, highlighting significant gaps that leave systems vulnerable to cyber threats. Satisfaction with current security testing integration is low, with over 60% of respondents dissatisfied, indicating substantial challenges in implementing effective practices. Key obstacles identified include a lack of skilled personnel, inadequate resources, time constraints, and insufficient management support, indicating a need for targeted interventions. Hence, the study points out key recommendations to address this gap.

KEYWORDS: Software Security, Software Development, Software Testing, Public Sector.

Cite the Article: *Wangwe, S.M, Oreku, G.S. (2026). Assessing the Methods and Difficulties of Incorporating Software Security Testing into System Development Process in Tanzania's Public Sector. Contemporary Research Analysis Journal, 3(1), 28–43. https://doi.org/10.55677/CRAJ/03-2026-Vol03I01*

Publication Date: *January 15, 2026*

***Corresponding Author:** George S. Oreku*

INTRODUCTION

In recent years, the rapid advancement of technology has led to an increased reliance on computer systems in the public sector of Tanzania and worldwide. These systems facilitate government operations, from citizen services and financial management to data storage and information sharing. However, the growing complexity and interconnectedness of these systems have raised concerns about their vulnerability to cyber threats and attacks [1,2,3]. Ensuring the security of public sector systems is paramount to safeguarding sensitive information, maintaining operational continuity, and protecting citizens' privacy and trust. Nevertheless, there is a prevailing issue of the inadequate integration of software security testing within the public sector system development process in Tanzania and across the world [4,5,6,7]. Software security testing evaluates the security of a software application or system to identify vulnerabilities and potential threats and ensure that the system is protected against unauthorized access, data breaches, and other security-related issues [8,9]. Incorporating security testing early in the development process helps organizations identify and fix vulnerabilities before they are released into production to help prevent security breaches and protect sensitive data [10].

Conventionally, software security testing is often treated as an afterthought rather than being ingrained as an integral part of the system development process in many projects [11,12]. This lack of emphasis on software security testing has significant implications for the robustness and resilience of public sector systems, leaving them vulnerable to potential breaches and compromises. The consideration of software security as an afterthought can be attributed to several factors, including limited awareness of security best practices, resource constraints, competing priorities, and insufficient expertise in implementing security measures [12,4]. Notably, the integration of software security testing into the public sector system development process is often overlooked or given inadequate

attention, leading to potential vulnerabilities and loopholes that malicious actors can exploit [15]. There are several ways to integrate software security testing into the system development process. One approach is to use a secure software development lifecycle (SDLC) framework [16,28]. The SDLC frameworks provide a structured approach to security testing that can be integrated into the development process.

Another approach is using security testing tools and techniques throughout development. For example, static analysis tools can be used to scan source code for potential vulnerabilities [17]. Dynamic analysis tools can be used to execute software applications and monitor their behavior for suspicious activity [18]. Penetration testing can be used to simulate an attack on a software application to identify potential vulnerabilities [19]. It is also important to involve security experts in the software development process. Security experts can help to identify and fix vulnerabilities that may not be obvious to developers. They can also help to develop and implement security policies and procedures. Through these steps, organizations can integrate software security testing into the system development process and help ensure their software applications' security.

Previous studies have focused on developing models, tools, and frameworks for integrating security into the software development life cycle [20,21,22] . However, there are no studies that have focused on evaluating the practices and challenges of integrating software security testing into the public sector system development process across the world. Therefore, this research aimed to evaluate the current state of integration of software security testing into the public sector system development process in Tanzania. It also explored the challenges associated with this integration to shed light on the root causes of the problem and propose recommendations for improvement. The findings of this research not only contribute to enhancing the security posture of public sector systems in Tanzania but also provide valuable insights for other countries facing similar challenges.

Through an in-depth analysis of current practices, identification of challenges, and development of practical recommendations, this research bridges the gap between the perception of software security testing as an afterthought and the imperative need to integrate security testing as an integral part of the public sector system development process. Note that, addressing this significant problem, public sector in Tanzania can establish a solid foundation for secure and resilient public sector systems that protect personal data and sensitive information, foster public trust, and ensure the uninterrupted delivery of essential services.

RELATED WORKS

The research presented by [4] emphasizes the significance of security in in-house developed information systems, a growing trend in Tanzanian organizations. The paper raises awareness about the risks of inadequate security measures, including data breaches and system damage caused by hackers. Furthermore, it identifies shortcomings in the security practices of information system developers and offers guidance to organizations to mitigate these risks. The main focus is on the importance of prioritizing security during the early stages of information system development.

The study by [23] addresses the significance of secure software development in today's digital landscape. It sheds light on the challenges developers face regarding security integration and proposes an approach that considers security throughout all software development phases. The research aims to provide valuable insights into the issues faced by developers and discusses methodologies to enhance security and client relationships. The paper emphasizes adopting a software, attacker, and asset-centric approach to bolster security in the system development process.

The research by [24] focuses on the imperative of integrating security into software development to achieve global software security. It explores diverse practices for ensuring software security and recommends examining these practices to attain optimal outcomes based on the desired security level. The paper classifies security testing strategies and tools into technical and non-technical assessment approaches. Its primary objective is to offer a thorough analysis of security testing, aiding professionals and researchers in addressing software security testing challenges worldwide. The paper also suggests isolating security issues from other enforcement matters to address them autonomously and apply them universally.

The study in [25] conducted an extensive review of software security testing approaches and techniques proposed between 2000 and 2015. The primary aim of security testing is to assess the effectiveness of implemented security measures and identify system vulnerabilities, ensuring protection against intrusions. Security testing can be conducted before or after production, but it is advisable to integrate it into the early phases of the software development life cycle to prevent increased costs and rework. The paper presents a well-structured review, categorizing software security testing approaches and techniques proposed thus far.

The study conducted by [26] is a literature review addressing the challenges and solutions of integrating software security practices with agile software development. The authors emphasize the rising necessity for effective integration of security tasks in the software development process, driven by the growing number of software security threats and vulnerabilities. However, achieving this integration poses difficulties due to differences in process dynamics and the focus on functional vs. non-functional requirements. The paper presents ongoing efforts to integrate security practices in agile methods, but further research is needed to optimize and simplify the processes for developers.

The research presented by [27] introduces an integrated security testing framework designed for the secure software development life cycle (SSDLC). This framework incorporates security activities and practices specific to SSDLC to generate security guidelines. Additionally, it integrates various security testing tools into a unified platform to offer comprehensive testing services and consolidate

the results from different tools for enhanced test accuracy. The authors developed a prototype system based on the framework to validate the approach and applied it to over 50 software development projects. The outcomes demonstrate that the prototype system delivers dependable and high-quality testing services.

The research conducted by [28] delves into the incorporation of security testing techniques into Continuous Integration/Continuous Delivery (CI/CD) pipelines, commonly employed in DevOps for swift feature delivery. The paper focuses on dynamic testing methods, which entail automated techniques to simulate attacks on running applications and identify vulnerabilities. A case study is presented where three dynamic testing techniques - Web Application Security Scanning (WAST) using Zed Attack Proxy (ZAP), Security API Scanning (SAS) with JMeter, and Behaviour Driven Security Testing (BDST) via the SeleniumBase automation framework - are integrated into a CI/CD pipeline. Additionally, the paper addresses challenges and limitations that DevOps teams may encounter when automating security tests, offering potential solutions. The findings of this research can significantly influence the adoption of DevSecOps practices in agile enterprise applications engineering and enterprise security.

Research Gap

The identified research gap revolves around the limited focus on integrating software security testing into the public sector system development process. Although the significance of security in software development is widely acknowledged, there is a notable absence of comprehensive studies and analysis specifically investigating the integration of security testing practices within the public sector context. While the existing literature does cover software security testing in general and within specific industries, there is a scarcity of research that addresses the distinct challenges and practices encountered in the public sector of Tanzania [29]. This scarcity contributes to a lack of in-depth understanding of the current state of software security testing within the public sector system development in the country.

Furthermore, while the available research offers various recommendations for best practices in software security testing, there is a lack of consensus on which practices are most effective in the Tanzanian context. Thus, more research is needed to identify and establish the optimal practices for seamlessly integrating software security testing into Tanzania's public sector system development process. The significance of addressing this research gap cannot be overstated, as it holds the potential to significantly enhance the security of public sector systems in Tanzania. By providing valuable insights into the effectiveness of existing security testing practices, this research can pinpoint areas that require improvement and guide the development of targeted strategies to bolster the security of information systems within the public sector. Additionally, it can serve as a foundation for the formulation of policies and guidelines to ensure the systematic incorporation of security best practices throughout the development process of public sector software projects.

Conceptual Framework

The conceptual framework presented in Figure 1 shows how the integration of software security testing into the public sector system development process is influenced by a number of factors.

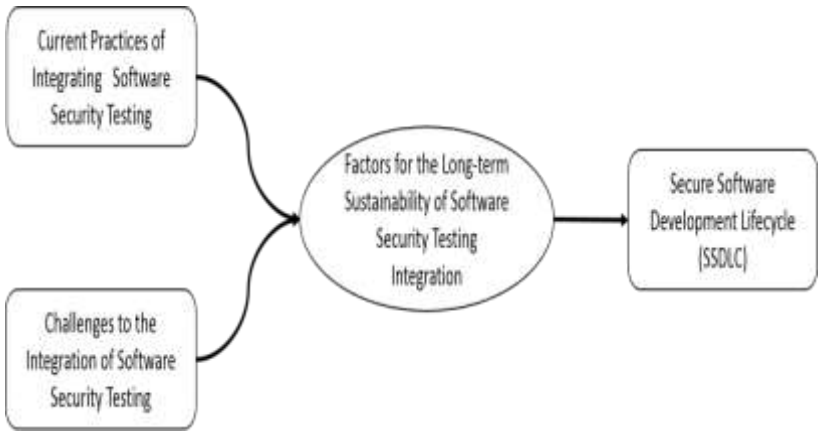

Figure 1: The conceptual framework

These are the current practices for software security testing, the challenges to integrating software security testing, and the factors influencing the integration of software security testing. The framework also demonstrates that integrating current practices and addressing the challenges of incorporating security into software testing in the public sector can identify key factors for the long-term sustainability of software security testing. This, in turn, can lead to a secure software development lifecycle (SSDLC).

The evaluation of the key factors for the long-term sustainability of software security testing integration was established based on the inputs gathered from the literature review, public sector system development practices, software security testing practices, and identified challenges. These factors act as the guiding principles for evaluating the integration of software security testing in the public sector context. The theoretical framework has led to a secure software development lifecycle based on the evaluation of the integration of software security testing and aim to address the identified challenges effectively.

METHODOLOGY

This chapter outlines the overall strategy for accomplishing the research objectives and addressing the research questions. Firstly, the selected research strategy is explained, and its justification is provided. Subsequently, the research methods employed in this study are detailed, along with the approach for validating and testing the reliability of the research instrument. Lastly, the ethical considerations are presented, highlighting the integration of ethical principles into the study.

Research Approach

This study used a mixed approach that entails qualitative and quantitative approaches. The study by [31] opined that the qualitative approach aims to explore and to discover issues because very little is known about the problem while it uses soft data and gets rich data. According to [32] qualitative approach is designed to help researchers understand people's social and cultural contexts within which they live and work. In addition, the quantitative approach uses questionnaires, surveys and experiments to gather data that is revised and tabulated in numbers, which allows the data to be characterized by statistical analysis [32]. Quantitative researchers measure variables on a sample of subjects and express the relationship between variables using effective statistics such as correlations, relative frequencies or differences between means. Their focus is to a large extent on the testing of theory [33].

Research Design

This sub-section presents an outline that guided the execution of this research study. It is an outline applied to display how answers to research objectives and questions (Hypothesis) will be achieved [34,35].

Study Area

This study will be done in the selected public organisations, including Ministries, Authorities, Agencies and Local Government Authorities. The identified research gap revolves around the limited focus on integrating software security testing into public sector system development process in Tanzania. Although the significance of security in software development is widely acknowledged, there is a notable absence of comprehensive studies and analyses specifically investigating the integration of security testing practices within the public sector context.

Population and Sample Size

This section provides a comprehensive overview of the population and sample size utilized in this study as detailed in the following subsections.

Population

The study population consists of individuals occupying managerial and operational roles within a carefully chosen set of Ministries, Authorities, Agencies, and Local Government Authorities across Tanzania. These organizations were selected based on their significance in the public sector and their involvement in system development processes. The participants represent a diverse range of departments and functions within these entities, including but not limited to information technology, administration, project management, and policy implementation. In total, the study engages with 266 participants, providing a comprehensive perspective on software security testing practices within the Tanzanian public sector.

Sample Size

The sample size is an essential feature of any empirical study whose goal is to make inferences about the population from a sample [33]. The sample size formula is applied to determine the appropriate respondents to represent the study population [36]. Where n is the sample size, N is the total target population, in this case, the 266-population size obtained, and e is the error rate, in this case 10%. The sample size for this study was calculated as shown below.

$$n = \frac{N}{1 + Ne^2}$$

whereby,

n= sample size

N= The Total population

e=standard error

1= constant

e = the margin of error (10% has been used to obtain the best sample given the population size)

$$n = \frac{266}{1 + 266(0.1)^{\ 2}}$$

n = 104

Therefore, the sample size is 104 respondents, which included 34 respondents from the management level and 70 respondents from the staff.

Sampling Procedure

The sampling procedure for this study employed a combination of purposive and stratified sampling techniques. Initially, key Ministries, Authorities, Agencies, and Local Government Authorities were purposively selected based on their relevance to the public sector system development process in Tanzania. Subsequently, within each selected entity, participants were stratified based on their

roles and responsibilities, ensuring representation from various levels of management and operational staff involved in software development and testing. The selection of the technique is based on the fact that the technique allows the capturing of only participants with vivid exposure and knowledge of a particular topic under investigation [37,38]. On the other hand, with a rigorous review of existing methods of determining the sample sizes viz; census applicable for the small population, benchmarking the size of samples for related studies, and formulas [39].

Data Collection Tools

The data collection tools for this study primarily consisted of structured questionnaires designed to gather quantitative data on software security testing practices. This choice was based on the fact that questionnaires allow data collection conveniently from large groups of people in a limited time, as justified. [40.41].

Data Analysis

For data analysis, the study utilized a combination of Microsoft Excel, Statistical Package for the Social Sciences (SPSS), and online survey platforms to process and interpret the collected data. Quantitative data from the surveys were analyzed using SPSS to generate descriptive statistics, such as frequencies, means, and standard deviations, providing insights into software security testing practices and trends. Additionally, qualitative data gathered from interviews were transcribed and coded using thematic analysis techniques to identify recurring themes and patterns in participants' responses. These analytical insights served as foundational input for determining the requirements and components essential for designing the reference architecture studied in the research, guiding subsequent design decisions and implementations.

Validity and Reliability of Data

This section presents a discussion on Validity and Reliability as detailed in the following subsections.

Validity

According to [42,43] data validity concerns whether the findings are really about what they appear to be. Validity encompasses the entire experimental concept and establishes whether the results obtained meet all of the requirements of the scientific research method. To ensure the validity of the data collected and the data collection instruments, efforts has been made to supplement qualitative data with some explanations to minimize biases and distortions. On the other hand, pilot testing was used to determine how far the instruments are measured correctly and accurately. Whenever possible, the data gathered from the study area was counter-checked for correctness and in case any error was identified, the respondent was requested to verify. This has helped to improve the validity and accuracy of data [44]

Reliability

Reliability is the extent to which the data collection process yields consistent results [44]. The idea behind reliability is that any significant results must be more one-off findings and inherently repeatable and that other researchers will be able to perform the same experiment under the same conditions and generate the same results. The instruments used must be reliable and able to keep true and accurate time. Therefore, reliability is necessary to determine a scientific experiment's overall validity and enhance the results' strength [44]. The Cronbach's alpha, which is the coefficient of internal consistency, has been used in study as an estimate of the reliability of the study. Reliability for the entire questionnaire and for all variables was checked and Cronbach's alpha was used to prove whether the findings are reliable. As used by (Noble and Smith 2015) the rule of thumb that applies to most situations is $0.9 \leq \alpha \leq 1.0$ excellent, $0.8 \leq \alpha < 0.9$ good, $0.7 \leq \alpha < 0.8$ acceptable, $0.6 \leq \alpha < 0.7$ questionable, $0.5 \leq \alpha < 0.6$ poor and $0.0 \leq \alpha < 0.5$ unacceptable. Therefore, the Cronbach's alpha exceeding 0.7 was accepted.

RESULTS AND DISCUSSION

Introduction

This paper presents the results and analysis of the study conducted to evaluate the integration of software security testing into the public sector system development process in Tanzania. The primary research goal was to assess the current practices, challenges, and potential improvements for effective security testing integration. The analysis considers the survey responses through the lens of the following specific research objectives:

- Investigating current software security testing practices in the public sector.
- Analyzing factors for the long-term sustainability of software security testing integration.
- Exploring challenges and barriers hindering successful software security testing integration.

The analysis draws upon quantitative and qualitative data collected through the survey questionnaire. It utilizes descriptive statistics to summarize demographic information and responses to Likert-scale questions as detailed in the following subsections.

Demographic Analysis

The section presents the distribution of respondents across gender, age, education level, area of specialization, experience, role in the development process, and public organization affiliation. The findings show that most (88%) respondents were male, and majority (79%) were aged between 26 and 44 years. More than three-thirds (79%) of the respondents had a bachelor degree or above. More than 99% of the respondents reported to perform security testing at least once a week during system development to detect security

vulnerabilities. Many (99%) of the respondents had more than one year's experience in security testing. These findings are shown in Table 3. Table 1:

TABLE 1: Demographic characteristics of the respondents

Characteristics	Value	Frequency	Percentage
Sex	Male	88	84.6
	Female	16	16.4
Age	15-25	4	3.8
	26-44	79	76
	45-64	21	20.2
Education Level	Secondary	-	0
	Certificate	2	1.9
	Diploma	3	2.9
	Degree	79	76
	Master's and above	21	19.2
Occupation	Government employee	96	92.3
	Private sector employee	8	7.7
Frequency of security testing	Once a week	6	5.8
	Multiple times a week	93	89.4
	Once a month	5	4.8
	Once a year	0	0
Experience in applications security testing	Less than a year	5	4.8
	1-2 years	20	19.2
	More than 2 years	79	76

The demographic data of the respondents indicates that the majority possess sufficient knowledge and skills, as evidenced by their experience, age, and education level. This enables them to provide relevant and substantial information, thereby justifying the significance of the study's findings.

Commonly Used Software Security Testing Techniques

Findings in Table 2 highlight commonly used software security testing approaches within the Tanzanian public organizations. Public organizations implement a range of techniques and methodologies in testing security risks and vulnerabilities for various systems and applications they develop. This study explored software security testing techniques and/or methodologies that are adopted and implemented in various public organizations.

TABLE 2: Commonly used software security testing techniques

Software Security Testing Techniques and/or Methodologies	Percentage of public organizations implementing it
Penetration testing	84.6
Vulnerability scanning	73.1
Burp Suite and OWASP ZAP	69
Code review/static analysis	46.2

The findings indicated that 84.6% of the public organizations implement penetration testing as software security testing techniques or methodologies. In implementing this technique, public organizations simulate real-world attacks so as to comprehensively identify vulnerabilities and assess security posture in the developed software systems. Findings also show that vulnerability scanning techniques was implemented by 73.1% of the public organizations in security testing. Those public organizations which implement vulnerability scanning engage automated scans in order to detect known vulnerabilities in software applications by aiding in identifying potential security weaknesses proactively. Moreover, a substantial proportion of respondents (about 69%) indicated that in their organization they use Burp Suite and OWASP ZAP techniques in software security testing. These techniques are used to conduct various security testing activities, including web application security testing, which is crucial given the increasing reliance on web-based systems in the public sector. Lastly, code review (other stated it as a static analysis) emerged also as one of the commonly used techniques with 46.2% of public organizations implementing it. While implementing Code review as a security testing measure, public organizations manually or automatically examine source code to detect security vulnerabilities so as to ensure software developed considers and comply to security issues.

Tools for Software Security Testing

The study also investigated the tools and technologies used for software testing in public organizations. Respondents were asked to select all applicable options from a list of tools and technologies utilized in software security testing within public organizations. The study findings indicate majority of respondents (73.1%) reported using Dynamic Application Security Testing (DAST) tools to assess software vulnerabilities in running applications. They adopt DAST as it allows organizations to identify security flaws in real-time, enhancing the overall security posture of their systems. Static code analysis tools also emerged as a one of commonly utilized technology (with 57.7% of respondents employing them) in analysing source code for potential vulnerabilities without executing the program. These tools provide insights to public organizations concerning security weaknesses during the development phase, thus facilitating proactive security measures.

Furthermore, nearly half of the respondents (46.2%) indicated the use of Web Application Firewalls (WAFs) on testing web-based applications from common cyber threats. They claim to adopt these tools because they act as barriers between web applications and the internet. Also, such tools monitor and filter HTTP traffic to prevent attacks such as SQL injection and cross-site scripting. Few respondents (30.8%) reported utilizing Security Information and Event Management (SIEM) systems in monitoring, detecting, and responding to security incidents by aggregating and analyzing security-related data from various sources. These findings imply that public organizations incorporate diverse tools and technologies to mitigate potential security risks throughout the system development lifecycle.

Dedicated Personnel for Software Security Testing

The study gathered information to ascertain whether public organizations establish dedicated team or personnel responsible for conducting software security testing. The findings reveal that 50.9% of the organizations involved in this study have dedicated teams or personnel responsible for software security testing. Conversely, a significant proportion of respondents (43.1%) reported the absence of dedicated teams or personnel responsible for software security testing within public organizations. Additionally, 6% of the respondents indicated being uncertain about the presence of dedicated teams or personnel for software security testing, within their organizations.

Based on the findings above the presence of dedicated teams or personnel responsible for software security testing within organizations in the Tanzanian public sector reveal notable insights into organizational structures and resource allocation to prioritize security measures. The existence of dedicated teams or personnel implies a proactive approach towards addressing security concerns within the system development process, potentially leading to more robust security postures and enhanced resilience against cyber threats. However, based on the fact that the percentage of organizations having the dedicated teams and personnel is relatively small, this raises concerns about potential gaps and challenges in addressing security considerations adequately throughout the system development lifecycle. The implication is that, many public organizations may face difficulties in identifying and mitigating security vulnerabilities effectively, increasing the risk of security breaches and data compromises, suggesting a lack of clarity or visibility regarding security testing responsibilities.

Furthermore, among organizations with dedicated teams or personnel responsible for software security testing, the majority of respondents (50.5%) identified to have their own security experts who are primarily responsible for this task. This implicate that most of the public organizations have personnel possessing adequate specialized skills required for identifying and mitigating security vulnerabilities effectively. The remaining organizations (49.5%) reported that the e- Government Authority (eGA) and other external experts conducts software security testing before systems go live, suggesting a centralized approach to security oversight within the public sector. This implies a multifaceted distribution of responsibilities for software security testing, involving various stakeholders to ensure comprehensive security coverage throughout the system development process in public organizations.

Importance of Software Security Testing

The findings from 104 respondents reveal a clear consensus among them from various organizations in Tanzania regarding the importance of software security testing. The majority (76%) of respondents responded that software security testing is very important, indicating a widespread acknowledgment of its crucial role in the public sector system development process. Additionally, 24% of respondents consider software security testing important, further recognizing its significance in safeguarding software systems against potential threats. None of the respondents indicated that software security testing was not important, or that they were not sure about its importance.

TABLE 3: Importance of software security testing

Importance level	Frequency (n=104)	Percentage
Very important	79	76
Important	25	24
Not sure	0	0
Not important	0	0
Unimportant at all	0	0

The findings on Table … stress the importance of prioritizing and improving software security testing within Tanzania's public sector, aligning with global best practices to mitigate vulnerabilities and enhance defenses against security breaches. These findings align with studies by other previous studies in the related works which reported security testing as an important aspect for efficient, effective and sustainable software. The implication is that, integrating security testing into the system development process is important for strengthen software resilience against cybersecurity threats. Also, security testing is vital in safeguarding organizational assets and ensuring public sector system integrity.

Satisfaction with Current Integration

The study also sought to weigh the respondents' satisfaction levels regarding the current integration of software security testing into the system development process. The findings indicate that minority of respondents (about 18%) expressed satisfaction with the current integration of software security testing into the system development process in public institutions. The findings further reveal that more than 60% of the respondents indicated being dissatisfied, highlighting substantial discontentment with the existing integration of software security testing efforts. Furthermore, almost one-fifth of the respondents (19.2%) were undecided and showed a lack of sentiment towards the current state of integration.

TABLE 4: Satisfaction with current integration

Importance level	Frequency (n=104)	Percentage
Very satisfied	8	7.7
Satisfied	11	10.6
Neutral	20	19.2
Dissatisfied	40	38.5
Very dissatisfied	25	24

This dissatisfaction signals potential gaps or challenges in implementing effective security testing practices within the system development process, which could compromise the overall security posture of public sector systems. This situation is attributed to a number of reasons among which

include uncertainty about the effectiveness of existing testing measures to align with global best practices in mitigating vulnerabilities and fortify defenses against potential security breaches.

Rating Integration of Software Security Testing

The study also valued the respondents' perceptions on the extent to which software security testing is integrated into the system development process within their respective organizations. Findings indicated that 11.5% of the respondents indicated that software security testing is not integrated at all within their organizations' system development processes. Moreover, 26.9% of the respondents reported that software security testing is only slightly integrated, indicating limited implementation of security testing measures in their organization. Conversely, a considerable number of respondents (34.6%) rated software security testing as integrated into their organization's system development process. Additionally, a noteworthy proportion (20.6%) reported that software security testing is highly integrated, indicating a significant commitment to incorporating robust security measures within the system development lifecycle. Only 6.7% of the respondents stated that software security testing is fully integrated, signaling an advanced level of maturity in security practices within their organizations.

TABLE 5: Rating Integration of software security testing

Extent of Integration	Frequency (n=104)	Percentage
Fully Integration	7	6.7
Highly Integrated	21	20.6
Integrated	36	34.3
Slightly Integrated	28	26.9
Not Integrated at all	12	11.5

The findings in Table 5 suggest a concerning gap in integrating security testing practices in systems development in public organization as only about 6% report a full integration. Overall, these findings highlight the varying degrees of integration of software security testing across Tanzanian public sector organizations and underscore the importance of further efforts to enhance security measures and mitigate potential risks effectively. This is a concern to address as the absence of full integrated formalized testing procedures leaves systems development in public institutions vulnerable to cybersecurity threats. This situation implicates a need for substantial improvements on procedures for system testing to bolster the resilience of software systems against potential vulnerabilities in public institutions. This also calls for a combined effort among all key stakeholders towards setting up and incorporating security testing measures mechanisms for enhancement and strengthening further security testing postures in public organizations.

Requirements Definition and Documentation

This study also enquired about prevailing practices of defining and document software security requirements in public organizations in Tanzania. A notable portion of respondents (57.7%) indicated that software security requirements are not defined or documented within their organizations. Additionally, a portion of respondents (23.1%) reported utilizing a separate set of security requirements for defining and documenting security requirements distinct from the overall system requirements. Conversely, a smaller percentage of respondents (19.2%) indicated that software security requirements are integrated and clearly defined and documented within the overall system requirements.

TABLE 6: Software Security requirement definition and documentation

Status of Integration and Documentation	Frequency	Percentage
Clearly defined and documented	20	19.2
Somewhat defined and documented	24	23.1
Not defined or documented	60	57.7

These findings highlight a concerning gap in security practices, suggesting a lack of formalized processes or protocols for identifying and documenting security requirements. The absence of defined security requirements could potentially leave software systems vulnerable to security threats, as security considerations may not be adequately addressed throughout the development lifecycle. As reported by (Chan and Kwok 2001), these findings suggest and raises questions about the integration and alignment of security requirements with broader system development objectives against potentially software systems vulnerable to security threats. Yet, for the few organizations that reported a clear mechanism for defining and documenting software security requirements portray an acknowledgment of the interconnectedness between security considerations and system functionality, potentially facilitating a more holistic approach to security within the development process. These findings imply a need for public organizations to establish clear and robust mechanisms for defining and documenting software security requirements, ensuring that security considerations are adequately addressed throughout the system development lifecycle.

Factors for Long-Term Sustainability of Integration

This section aims to explore the key factors that should be considered to ensure the long-term sustainability of software security testing integration within organizations operating in Tanzania's public sector. The following subsections details the findings under this section.

Key Factors

The respondents emphasized several key factors to ensure the long-term sustainability of software security testing integration in their organizations. Firstly, they stressed the importance of executive support and commitment, highlighting the need for ongoing leadership backing to prioritize security testing as a fundamental aspect of development. Secondly, respondents underscored the necessity of having dedicated security personnel involved in every software development project, advocating for their early integration rather than just at the end for testing purposes. Collaboration between development teams and security testers was also emphasized, with the recommendation to view security testers as essential contributors rather than adversaries.

The respondents emphasized the significance of day-to-day security assessment to proactively identify and address potential cyber threats, highlighting the importance of education and training for all team members involved in projects to ensure a comprehensive understanding and application of security measures. They also suggested creating a special unit within the organization responsible for software security testing to streamline and centralize security efforts. Additionally, the respondents stressed the importance of succession planning to ensure the continuity of security expertise and mitigate potential disruptions. Furthermore, they highlighted additional considerations such as defining metrics for measuring effectiveness, leveraging third- party assessments for fresh perspectives, and integrating security by design principles.

Moreover, respondents stressed the need for continuous capacity building and budget allocation to prioritize security matters effectively. They emphasized the importance of integrating security into the Software Development Life Cycle (SDLC) and developing a clear security governance structure with defined roles and responsibilities. Lastly, the respondents recommended establishing a comprehensive incident response plan, conducting regular audits and reviews, and ensuring compliance with relevant regulations and standards to maintain the long-term sustainability of software security testing integration. Overall, these key factors highlighted by the respondents provide a comprehensive framework for organizations to address and prioritize security concerns effectively.

Recommendations from Experience

The survey findings reveal several key recommendations to enhance the integration of software security testing into the system development process within Tanzanian organizations. One prominent challenge highlighted by respondents is the resistance to change. Overcoming this resistance necessitates a cultural shift towards embracing security as an integral part of the development process. To address this, organizations should prioritize security testing by design, integrating security considerations from the conceptualization

phase through requirement gathering. This proactive approach ensures that security measures are woven into the fabric of the software development lifecycle, fostering a security-conscious mindset among stakeholders.

Moreover, respondents emphasize the importance of involving security experts early in the project lifecycle, particularly during concept development and requirement gathering stages. This involvement ensures that security considerations are deeply embedded in the project's foundation. Additionally, to institutionalize these practices, respondents suggest establishing legal or regulatory frameworks mandating the integration of security into the development process. This would include allocating a consistent budget for capacity building and utilizing licensed software, ensuring sustained investment in security infrastructure. Furthermore, respondents advocate for the adoption of DevSecOps principles, advocating for the seamless integration of security tools and processes into the Continuous Integration/Continuous Deployment (CI/CD) pipeline. By automating security testing at each stage of code development, organizations can enhance efficiency and effectiveness while maintaining robust security standards.

Modifying the System Development Process

In response to the survey question regarding how organizations can modify the system development process to prioritize software security testing, the feedback from respondents provided valuable insights. One common suggestion highlighted by respondents is the integration of security teams into the development process from the early stages, including requirements gathering, testing, and piloting. This ensures that security considerations are embedded throughout the development lifecycle, addressing vulnerabilities proactively. Additionally, respondents emphasized the importance of conducting software testing for individual modules, allowing for comprehensive testing of each component to identify and remediate security flaws effectively.

Moreover, respondents stressed the necessity of establishing ICT security as an independent unit or section within the organization, separate from the ICT department. This structural change aims to enhance compliance with ICT security issues and ensure dedicated focus on security matters. Furthermore, respondents outlined the importance of raising awareness among management and providing training to programmers and security personnel. By educating stakeholders on security best practices and business process owners on system security, organizations can foster a culture of security awareness and accountability.

Respondents also provided a comprehensive list of measures and changes to improve software security testing integration. These include defining security standards and guidelines, integrating security requirements into the development process, adopting secure design principles, conducting regular security testing, implementing continuous monitoring, and developing an incident response plan, among others. By following these recommendations and making modifications to the system development process, organizations can effectively prioritize software security testing, mitigate vulnerabilities, and safeguard against potential cyber threats.

Software Security Testing Through Training and Education

In response to the survey question regarding beneficial training or educational programs to enhance software security testing capabilities, respondents provided a range of valuable suggestions. One prevalent recommendation was the pursuit of industry-recognized certifications such as Certified Information Systems Security Professional (CISSP), Certified Ethical Hacker (CEH), and Certified Secure Software Lifecycle Professional (CSSLP). These certifications cover various aspects of software security testing, including ethical hacking, secure software development principles, and secure coding practices, providing participants with a solid foundation for conducting effective security testing.

Moreover, respondents emphasized the importance of specialized training programs focused on specific areas of software security testing. These include courses on web application security testing, threat modeling workshops, and training on secure coding principles. By targeting these specific areas, organizations can ensure that their team members possess the necessary skills and expertise to identify and mitigate security vulnerabilities effectively. Additionally, hands-on training initiatives such as bug bounty programs and Capture, the Flag (CTF) challenges were highlighted as valuable opportunities for practical experience and skill development in real-world security testing scenarios.

Furthermore, respondents stressed the importance of continuous learning and staying updated on the latest security trends and threats. This can be achieved through participation in security conferences, seminars, and online courses offered by platforms like Coursera, edX, and Udemy. By providing access to these resources, organizations can empower their team members to stay abreast of emerging security challenges and technologies, enabling them to adapt their security testing practices accordingly.

Lastly, enhancing software security testing capabilities within an organization requires a multifaceted approach that combines formal education, specialized training programs, practical experience, and continuous learning initiatives. By investing in these programs and initiatives, organizations can build a highly competent and skilled software security testing team capable of effectively identifying and mitigating security vulnerabilities, ultimately strengthening the overall security posture of their software applications.

Best Practices from Local and Global Experiences

In evaluating the practices and challenges of integrating software security testing into public sector system development processes in Tanzania, respondents indicated a keen interest in learning from global experiences. They highlighted the importance of considering examples from developed countries such as the USA, Europe, and China, which are often pioneers in cybersecurity initiatives. These regions offer valuable insights into effective security testing integration strategies that could be adapted to the Tanzanian context. Additionally, respondents pointed to successful examples of organizations in Tanzania like the E-Government Agency (eGA), CERT

(Computer Emergency Response Team) and TCRA (Tanzania Communications Regulatory Authority) which have established dedicated units responsible for software security testing. Emulating such approaches could prove beneficial in Tanzania's public sector.

Respondents offered a range of recommendations and examples for improving software security testing integration based on global best practices. These included implementing a DevSecOps culture, leveraging automation for security testing processes, and adopting threat modeling early in the development process. Examples such as Microsoft's Security Development Lifecycle (SDL) and the US FedRAMP certification were cited as successful models that organizations could emulate. Additionally, bug bounty programs, penetration testing, and third-party risk assessments were highlighted as effective strategies for identifying and addressing security vulnerabilities.

To ensure the long-term sustainability of software security testing integration, respondents stressed the importance of various factors. Leadership commitment, clear security policies, skilled workforce, and integration with the software development lifecycle (SDLC) were identified as critical elements. Other factors included continuous improvement, compliance with regulations, budget allocation, and fostering a culture of security awareness. By addressing these factors, organizations can establish a robust and effective software security testing program that adapts to evolving threats and technologies.

Lastly, respondents emphasized the importance of learning opportunities and collaboration with other countries and organizations. They highlighted the success of countries like India, Estonia, and Singapore in cybersecurity initiatives and suggested that Tanzania could benefit from studying their approaches. Collaboration with global cybersecurity organizations, participation in industry- specific forums, and leveraging open-source tools and resources were also recommended. By embracing a culture of continuous learning and collaboration, organizations can stay informed about the latest trends and best practices in software security testing integration, ultimately enhancing their cybersecurity posture.

Challenges in Integrating Software Security Testing

This section aims to delve into the challenges which hinder the successful integration of software security testing into the public sector system development process in Tanzania as detailed in the following subsections.

Common Challenges

This subsection focuses on identifying the common challenges encountered in the process of integrating software security testing into the system development process within organizations operating in Tanzania's public sector. The survey findings on the main challenges faced in integrating software security testing into the system development process within organizations in Tanzania's public sector reveal several key obstacles that impede effective security testing practices. A significant majority of respondents (61.5%) identified the lack of skilled personnel as a primary challenge. This finding underscores the crucial role of skilled professionals in implementing and managing security testing processes effectively. Insufficient expertise in security testing methodologies and tools can hinder organizations' ability to identify and mitigate security vulnerabilities, leaving systems vulnerable to potential cyber threats.

Moreover, more than half of the respondents (53.8%) cited insufficient budget and resources as a major challenge. This highlights the critical importance of adequate financial investment and resource allocation for implementing robust security testing measures within the system development process. Insufficient resources can limit organizations' ability to acquire necessary tools, conduct comprehensive testing activities, and train personnel, thereby compromising the overall effectiveness of security testing efforts. Similarly, time constraints in the development process emerged as a significant challenge, with 50% of respondents expressing concerns about the limited time available for incorporating security testing activities. Tight deadlines and project schedules may lead to rushed development cycles, potentially overlooking critical security considerations and increasing the likelihood of introducing vulnerabilities into software systems.

Furthermore, the survey findings reveal a considerable lack of awareness about software security testing, with 46.25% of respondents citing this as a major challenge. This highlights a critical gap in knowledge and understanding among stakeholders regarding the importance and benefits of security testing practices. Additionally, management awareness was identified as a challenge by 54.3% of respondents, indicating a need for increased awareness and support from organizational leadership to prioritize and invest in security testing initiatives. The rush to bring applications to production was cited as a significant challenge by the majority of respondents (68.7%), underscoring the pressure to meet deployment deadlines and the potential trade-offs between speed and security. Overall, these findings highlight the multifaceted challenges faced by organizations in Tanzania's public sector in integrating software security testing into the system development process and underscore the need for targeted interventions and strategies to address these barriers effectively.

Prioritization of Software Security Testing

The survey findings regarding the extent to which respondents agree or disagree with the statement "Software security testing is given adequate priority in the system development process in your organization" provide valuable insights into the perception and prioritization of security testing within organizations in Tanzania's public sector. The responses indicate a varied perspective among respondents, with a mix of strongly agree, agree, neutral, and disagree responses. While a significant number of respondents expressed agreement, suggesting a recognition of the importance of software security testing, there were also instances of disagreement and

neutrality. This suggests potential discrepancies in the prioritization of security testing practices within organizations, with some stakeholders perceiving it as adequately prioritized while others may view it differently. These findings underscore the importance of ensuring consistent prioritization and allocation of resources towards software security testing to enhance the overall security posture of public sector systems in Tanzania.

Also, the survey findings regarding the extent of collaboration between developers and security testers within organizations in Tanzania's public sector reflect a varied perspective among respondents. While a significant number of respondents expressed agreement or strong agreement with the statement, indicating a perceived level of collaboration between these two crucial stakeholders, there were also instances of disagreement, neutrality, and even strong disagreement. This suggests potential discrepancies in collaboration practices across different organizations, with some stakeholders perceiving sufficient collaboration while others may not. These findings highlight the importance of fostering closer coordination and collaboration between developers and security testers to ensure effective security testing practices throughout the system development process. Enhancing collaboration between these roles can facilitate the identification and mitigation of security vulnerabilities more efficiently, ultimately leading to stronger security postures for public sector systems in Tanzania.

Moreover, the survey responses regarding the extent to which management actively supports the integration of software security testing within organizations in Tanzania's public sector demonstrate a mixed perspective among respondents. While a considerable number of respondents expressed agreement or strong agreement with the statement, indicating perceived support from management for security testing initiatives, there were also instances of disagreement, neutrality, and even strong disagreement. This suggests potential disparities in management support across different organizations, with some stakeholders perceiving active support while others may not. These findings underscore the importance of fostering greater awareness and advocacy among management regarding the significance of software security testing and its integration into the system development process. Ensuring robust support from management can facilitate the allocation of resources, implementation of security measures, and adoption of best practices, ultimately enhancing the overall security posture of public sector systems in Tanzania.

Additionally, the survey findings regarding the extent of training and education on software security testing for personnel involved in the development process within organizations in Tanzania's public sector demonstrate a varied perspective among respondents. While there were instances of agreement or strong agreement with the statement, suggesting perceived adequacy of training initiatives, there were also responses indicating disagreement, neutrality, and even strong disagreement. This suggests potential disparities in the availability and effectiveness of training and education programs across different organizations. Ensuring adequate training and education on software security testing is crucial for equipping personnel with the necessary skills and knowledge to effectively identify and mitigate security vulnerabilities throughout the development lifecycle. Therefore, these findings emphasize the need for organizations to prioritize and invest in comprehensive training initiatives to enhance the competency of personnel involved in software development processes, ultimately contributing to strengthened security practices within the public sector in Tanzania.

Lastly, the survey responses regarding the clarity and enforcement of legal and regulatory requirements for software security testing within organizations in Tanzania's public sector indicate a varied perspective among respondents. While there were instances of agreement or strong agreement with the statement, suggesting perceived clarity and enforcement of requirements, there were also responses indicating disagreement, neutrality, and even strong disagreement. This suggests potential discrepancies in the understanding and implementation of legal and regulatory frameworks across different organizations. Clear and enforced legal and regulatory requirements are essential for providing guidance and establishing standards for software security testing practices, ultimately contributing to the overall security posture of public sector systems. Therefore, these findings highlight the importance of ensuring consistent interpretation and enforcement of legal and regulatory requirements to foster a more robust security environment within the public sector in Tanzania.

DISCUSSION

The findings from this study underscore the critical need for enhanced software security testing practices within public sector organizations in Tanzania. Despite the growing reliance on digital systems, a significant portion of these organizations exhibit limited integration of security testing into their development processes. This gap poses a substantial threat to the security and resilience of public services. Moreover, this study revealed a notable disparity between the perceived importance of software security testing and the actual implementation of robust practices. While respondents universally acknowledged the significance of security testing, a significant portion of organizations lacked dedicated teams and faced challenges in aligning existing measures with global best practices. This discrepancy highlights the need for a more proactive and systematic approach to security testing.

Several key factors emerged as crucial for the successful integration of software security testing. Executive support, early involvement of security personnel, and fostering collaboration between development and security teams were identified as essential elements. Additionally, continuous security assessment, education, and training were emphasized as vital components for building a security-conscious culture within organizations. Furthermore, this study highlighted a number of challenges hindering the integration of software security testing. A lack of skilled personnel, insufficient budget and resources, time constraints, and inadequate management support were identified as primary obstacles. Addressing these challenges requires targeted interventions, such as

enhancing skills development, allocating sufficient resources, raising awareness, and fostering a supportive management environment.

Based on the findings, several recommendations can be made to improve the integration of software security testing in Tanzanian public sector organizations. Firstly, organizations should prioritize investments in ICT personnel training and resource allocation to enhance their cybersecurity capabilities. Secondly, fostering closer collaboration between development teams and security testers, coupled with proactive management support, can significantly strengthen security testing practices. Finally, aligning legal and regulatory frameworks with international standards will ensure clarity and consistency in enforcing security measures, thus bolstering the overall resilience of public sector systems against cyber threats. Lastly, this study provides valuable insights into the current state of software security testing in public sector organizations in Tanzania. The government organization need to address the identified challenges and implement the recommended strategies to significantly enhance their cybersecurity posture and protect the integrity of public services.

CONCLUSION

This study has evaluated the current practices and challenges in embedding security testing early in the software development lifecycle to promote a secure-by-design approach that ensures robust and resilient public sector information systems. This investigation was guided by three specific research objectives: first, to investigate current software security testing practices in the public sector; second, to analyze factors influencing the long-term sustainability of software security testing integration; and third, to explore challenges and barriers hindering successful integration. The study employed a mixed-method approach, collecting both quantitative and qualitative data from 104 respondents through a survey questionnaire from ICT managers and directors, ICT security officers, software developers, and systems administrators across the selected public organizations in Tanzania. The analysis utilized descriptive statistics to summarize demographic information and responses to Likert-scale questions, offering a comprehensive understanding of the prevailing practices and perceptions within the target population.

Findings from the survey revealed a varied landscape of software security testing practices across the public sector, highlighting the need for standardized approaches and enhanced awareness. The study found considerable variation in the extent of software security testing integration across Tanzanian public organizations, with only 6.7% reporting full integration and over 38% reporting little to no integration. This signals a concerning gap in the consistent adoption of robust security testing practices, leaving public sector systems vulnerable to potential cybersecurity threats. The findings underscore the importance of enhancing security measures and formalizing security testing procedures to mitigate risks more effectively. Addressing this issue will require concerted efforts by all stakeholders to establish and implement comprehensive security testing mechanisms across public institutions. Improving the integration of security testing into system development lifecycles is crucial for bolstering the resilience and safeguarding the integrity of public sector software systems.

Also, the study assessed respondents' satisfaction with the current integration of software security testing in public institutions' system development processes. Only 18% of respondents were satisfied with the current integration of software security testing in public institutions' system development processes, while over 60% expressed dissatisfaction, indicating significant discontent with current practices. Additionally, 19.2% of respondents were neutral, showing a lack of strong sentiment about the current state of integration. This widespread dissatisfaction suggests substantial gaps or challenges in implementing effective security testing practices, potentially compromising the security of public sector systems. The findings highlight concerns about the effectiveness of existing measures in aligning with global best practices, contrasting with previous studies that reported higher satisfaction levels in other contexts. Also, while there is a consensus (100%) on the importance of software security testing, only half (50.9%) of the organizations have dedicated teams for this task. This raises concerns about the ability of some organizations to effectively address security vulnerabilities. The study also found that a mix of internal security experts and external bodies like the eGovernment Authority contribute to software security testing in these organizations.

On the contrary, the study identified several key factors from the respondents crucial for the long- term sustainability of software security testing integration within Tanzanian public sector organizations. Respondents reported that, executive support and commitment, early involvement of dedicated security personnel, and fostering collaboration between development teams and security testers are essential. Also, continuous security assessment, education, training for all team members, specialized security units, and succession planning reported to be vital. Similarly, integrating security into the Software Development Life Cycle (SDLC), establishing clear governance structures, and implementing incident response plans, audits, and compliance measures reported to be necessary for robust security practices. Recommendations by the respondents for improving integration include overcoming resistance to change, involving security experts early in projects, establishing legal frameworks, and adopting DevSecOps principles (development, security, and operations). Leveraging best practices from global experiences, such as DevSecOps culture and threat modeling, and fostering continuous learning and collaboration with global cybersecurity organizations were also emphasized. Therefore, Tanzanian public sector organizations can enhance their software security testing integration and resilience against cyber threats by addressing these factors and recommendations,

Lastly, the study identified several key challenges hindering the integration of software security testing in public sector in Tanzania. A majority of respondents (61.5%) highlighted the lack of skilled personnel as a primary obstacle, followed by insufficient budget and resources (53.8%) and time constraints (50%). Additionally, a lack of awareness about software security testing (46.25%) and inadequate management support (54.3%) were significant barriers. The rush to bring applications to production (68.7%) also compromised security efforts. These challenges underline the need for targeted interventions, such as enhancing skills, allocating sufficient resources, raising awareness, and fostering management support to improve security testing practices in the public sector.

Based on the findings, it is recommended that public sector organizations in Tanzania prioritize the enhancement of cybersecurity capabilities through targeted investments in ICT personnel training and resource allocation. Additionally, fostering closer collaboration between development teams and security testers, coupled with proactive management support, can significantly strengthen software security testing practices. Lastly, aligning legal and regulatory frameworks with international standards will ensure clarity and consistency in enforcing security measures, thus bolstering the overall resilience of public sector systems against cyber threats.

REFERENCES

1. Pallangyo, Hakeem. (2022). “Cyber Security Challenges, Its Emerging Trends on Latest Information and Communication Technology and Cyber Crime in Mobile Money Transaction Services.” Tanzania Journal of Engineering and Technology 41 (2). https://doi.org/10.52339/tjet.v41i2.792.
2. Mahendra, N., & Ahmad, S. (2016). A categorized review on software security testing. *International Journal of Computer Applications*, *154*(1), 21–25. https://doi.org/10.5120/ijca2016912023
3. Riisom, Klaus Reche, Martin Slusarczyk Hubel, Hasan Mousa Alradhi, Niels Bonde Nielsen, Kati Kuusinen, and Ronald Jabangwe. (2018). “Software Security in Agile Software Development: A Literature Review of Challenges and Solutions.” In *ACM International Conference Proceeding Series*. Vol. Part F147763. https://doi.org/10.1145/3234152.3234189.
4. Mushi, Magreth, and Jabiri Bakari. (2012). “Security in In-House Developed Information Systems: The Case of Tanzania.” *Systemics, Cybernetics and Informatics* 10 (2): 1–5.
5. Lyimo, Benson James. (2022). “Information Security Vulnerabilities and Tanzania Ministry of Education.” *Olva Academy – School of Researchers* 4 (1).
6. Ally, Said. (2014). “Security Vulnerabilities of the Web Based Open-Source Information Systems: Adoption Process and Source Codes Screening.” *HURIA: Journal of The Open University of Tanzania* 17: 1–13.
7. Brucker, Achim D, Dimitar Yanev, and Stephen Hookings. (2015). “Bringing Security Testing to Development: How to Enable Developers to Act as Security Experts.” In. https://api.semanticscholar.org/CorpusID:58662775.
8. Abdul Rahman, Abdul Hadi Bin, Abdullah Nazir, Kim Tae Hyun, Tan Horng Yarng, and Fatima Tuz Zahra. (2020). “Software, Attacker and Asset-Centric Approach for Improving Security in System Development Process.” *ArXiv*, 1–12.
9. Memon, Muhammad Sulleman, Mairaj Nabi Bhatti, Manzoor Ahmed Hashmani, Muhammad Shafique Malik, and Naveed Murad Dahri. (2021). “Techniques and Trends Towards Various Dimensions of Robust Security Testing in Global Software Engineering.” In *Research Anthology on Agile Software, Software Development, and Testing*. Vol. 3. https://doi.org/10.4018/978-1-6684-3702-5.ch062.
10. Memon, Muhammad Sulleman, Mairaj Nabi Bhatti, Manzoor Ahmed Hashmani, Muhammad Shafique Malik, and Naveed Murad Dahri. (2021). “Techniques and Trends Towards Various Dimensions of Robust Security Testing in Global Software Engineering.” In *Research Anthology on Agile Software, Software Development, and Testing*. Vol. 3. https://doi.org/10.4018/978-1-6684-3702-5.ch062.
11. Riisom, Klaus Reche, Martin Slusarczyk Hubel, Hasan Mousa Alradhi, Niels Bonde Nielsen, Kati Kuusinen, and Ronald Jabangwe. (2018). “Software Security in Agile Software Development: A Literature Review of Challenges and Solutions.” In *ACM International Conference Proceeding Series*. Vol. Part F147763. https://doi.org/10.1145/3234152.3234189.
12. Memon, Muhammad Sulleman, Mairaj Nabi Bhatti, Manzoor Ahmed Hashmani, Muhammad Shafique Malik, and Naveed Murad Dahri. (2021). “Techniques and Trends Towards Various Dimensions of Robust Security Testing in Global Software Engineering.” In *Research Anthology on Agile Software, Software Development, and Testing*. Vol. 3. https://doi.org/10.4018/978-1-6684-3702-5.ch062.
13. Rajapakse, Roshan N., Mansooreh Zahedi, M. Ali Babar, and Haifeng Shen. (2022). “Challenges and Solutions When Adopting DevSecOps: A Systematic Review.” *Information and Software Technology*. https://doi.org/10.1016/j.infsof.2021.106700.
14. Mahendra, Neha, and Suhel Ahmad. (2016). “A Categorized Review on Software Security Testing.” *International Journal of Computer Applications* 154 (1). https://doi.org/10.5120/ijca2016912023.
15. Tung, Yuan Hsin, Sheng Chen Lo, Jen Feng Shih, and Hung Fu Lin. (2016). “An Integrated Security Testing Framework for Secure Software Development Life Cycle.” In *18th Asia- Pacific Network Operations and Management Symposium, APNOMS 2016: Management of Softwarized Infrastructure - Proceedings*.

https://doi.org/10.1109/APNOMS.2016.7737238.

16. Sosnytskyi, Sergii, Mykola Glybovets, and Olena Pyechkurova. (2020). "Statical and Dynamical Software Analysis." *NaUKMA Research Papers. Computer Science* 3 (0). https://doi.org/10.18523/2617-3808.2020.3.50-55.
17. Sosnytskyi, Sergii, Mykola Glybovets, and Olena Pyechkurova. (2020). "Statical and Dynamical Software Analysis." *NaUKMA Research Papers. Computer Science* 3 (0). https://doi.org/10.18523/2617-3808.2020.3.50-55.
18. Nigam, Divya, Vinita Malik, and Sarvagya Nigam. (2015). "Methods and Techniques of Security Testing: A Survey." *International Journal of Advanced Engineering and Global Technology* 3 (1).
19. Lee, Younghwa, Jintae Lee, and Zoonky Lee. (2002). "Integrating Software Lifecycle Process Standards with Security Engineering." *Computers and Security*. https://doi.org/10.1016/S0167-4048(02)00413-3.
20. Baldassarre, Maria Teresa, Vita Santa Barletta, Danilo Caivano, and Michele Scalera. (2020). "Integrating Security and Privacy in Software Development." *Software Quality Journal* 28 (3). https://doi.org/10.1007/s11219-020-09501-6.
21. Rangnau, Thorsten, Remco V. Buijtenen, Frank Fransen, and Fatih Turkmen. (2020). "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines." In *Proceedings – 2020 IEEE 24th International Enterprise Distributed Object Computing Conference, EDOC 2020*. https://doi.org/10.1109/EDOC49727.2020.00026.
22. Abdul Rahman, Abdul Hadi Bin, Abdullah Nazir, Kim Tae Hyun, Tan Horng Yarng, and Fatima Tuz Zahra. (2020). "Software, Attacker and Asset-Centric Approach for Improving Security in System Development Process." *ArXiv*, 1–12.
23. Memon, Muhammad Sulleman, Mairaj Nabi Bhatti, Manzoor Ahmed Hashmani, Muhammad Shafique Malik, and Naveed Murad Dahri. (2021). "Techniques and Trends Towards Various Dimensions of Robust Security Testing in Global Software Engineering." In *Research Anthology on Agile Software, Software Development, and Testing*. Vol. 3. https://doi.org/10.4018/978-1-6684-3702-5.ch062.
24. Mahendra, Neha, and Suhel Ahmad. (2016). "A Categorized Review on Software Security Testing." *International Journal of Computer Applications* 154 (1). https://doi.org/10.5120/ijca2016912023.
25. Riisom, Klaus Reche, Martin Slusarczyk Hubel, Hasan Mousa Alradhi, Niels Bonde Nielsen, Kati Kuusinen, and Ronald Jabangwe. (2018). "Software Security in Agile Software Development: A Literature Review of Challenges and Solutions." In *ACM International Conference Proceeding Series*. Vol. Part F147763. https://doi.org/10.1145/3234152.3234189.
26. Tung, Yuan Hsin, Sheng Chen Lo, Jen Feng Shih, and Hung Fu Lin. (2016). "An Integrated Security Testing Framework for Secure Software Development Life Cycle." In *18th Asia- Pacific Network Operations and Management Symposium, APNOMS 2016: Management of Softwarized Infrastructure - Proceedings*. https://doi.org/10.1109/APNOMS.2016.7737238.
27. Rangnau, Thorsten, Remco V. Buijtenen, Frank Fransen, and Fatih Turkmen. (2020). "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines." In *Proceedings - 2020 IEEE 24th International Enterprise Distributed Object Computing Conference, EDOC 2020*. https://doi.org/10.1109/EDOC49727.2020.00026.
28. Lyimo, Benson James. (2022). "Information Security Vulnerabilities and Tanzania Ministry of Education." *Olva Academy – School of Researchers* 4 (1).
29. Ally, Said. (2014). "Security Vulnerabilities of the Web Based Open Source Information Systems: Adoption Process and Source Codes Screening." *HURIA: Journal of The Open University of Tanzania* 17: 1–13.
30. Creswell, J W. (2003). "Research Design Qualitative Quantitative and Mixed Methods Approaches." *Research Design Qualitative Quantitative and Mixed Methods Approaches*. https://doi.org/10.3109/08941939.2012.723954.
31. Snyder, Hannah. (2019). "Literature Review as a Research Methodology: An Overview and Guidelines." *Journal of Business Research* 104. https://doi.org/10.1016/j.jbusres.2019.07.039.
32. Creswell, J W. (2003). "Research Design Qualitative Quantitative and Mixed Methods Approaches." *Research Design Qualitative Quantitative and Mixed Methods Approaches*. https://doi.org/10.3109/08941939.2012.723954.
33. Schoonenboom, Judith, and R. Burke Johnson. (2017). "How to Construct a Mixed Methods Research Design." *KZfSS Kölner Zeitschrift Für Soziologie Und Sozialpsychologie* 69 (S2). https://doi.org/10.1007/s11577-017-0454-1.
34. Ganesha, H. R., and P. S. Aithal. (2022). "How to Choose an Appropriate Research Data Collection Method and Method Choice Among Various Research Data Collection Methods and Method Choices During Ph.D. Program in India?" *International Journal of Management, Technology, and Social Sciences*. https://doi.org/10.47992/ijmts.2581.6012.0233.
35. Taherdoost, Hamed. (2018). "Sampling Methods in Research Methodology; How to Choose a Sampling Technique for Research." *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3205035.
36. Bodnar, Małgorzata, Jacek Namieśnik, and Piotr Konieczka. (2013). "Validation of a Sampling Procedure." *TrAC - Trends in Analytical Chemistry*. https://doi.org/10.1016/j.trac.2013.06.011.
37. Baltes, Sebastian, and Paul Ralph. (2022). "Sampling in Software Engineering Research: A Critical Review and Guidelines." *Empirical Software Engineering* 27 (4). https://doi.org/10.1007/s10664-021-10072-8.
38. Bodnar, Małgorzata, Jacek Namieśnik, and Piotr Konieczka. (2013). "Validation of a Sampling Procedure." *TrAC - Trends in Analytical Chemistry*. https://doi.org/10.1016/j.trac.2013.06.011.

39. Sandelowski, Margarete. (2000). "Focus on Research Methods: Combining Qualitative and Quantitative Sampling, Data Collection, and Analysis Techniques in Mixed-Method Studies." *Research in Nursing and Health* 23 (3). https://doi.org/10.1002/1098- 240x (200006) 23:3<246::aid-nur9>3.0.co;2-h.
40. Hamed, Omayma, Husain Hamza Jabbad, Omar I. Saadah, Mahmoud S. Al Ahwal, and Fatin M. Al-Sayes. (2018). "An Explanatory Mixed Methods Study on the Validity and Validation of Students' Assessment Results in the Undergraduate Surgery Course." *Medical Teacher* 40 (sup1). https://doi.org/10.1080/0142159X.2018.1465181.
41. Heale, Roberta, and Alison Twycross. (2015). "Validity and Reliability in Quantitative Studies. *"Evidence-Based Nursing*. https://doi.org/10.1136/eb-2015-102129.
42. Hayashi, Paulo, Gustavo Abib, and Norberto Hoppen. (2019). "Validity in Qualitative Research: Processual Approach." *Qualitative Report* 24 (1). https://doi.org/10.46743/2160- 3715/2019.3443.
43. Twycross, Alison, and Linda Shields. (2004). "Validity and Reliability--What's It All about? Part1. Validity in Quantitative Studies." *Paediatric Nursing* 16 (9). https://doi.org/10.7748/paed2004.11.16.9.28.c954