International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

Int. J. Sci. Res. Arch.
IJSRA
International Journal of Science and Research Archive
Research Journal Archive, INDIA

(Review Article)

Check for updates

Securing financial transactions: DevSecOps best practices for banking applications

Ayobami Oluwadamilola Adebayo *

Computer Science, Landmark Metropolitan University, Buea, Cameroon.

International Journal of Science and Research Archive, 2025, 15(02), 733-738

Publication history: Received on 06 April 2025; revised on 11 May 2025; accepted on 13 May 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.15.2.1458

Abstract

The growing reliance on digital banking platforms has heightened the need for robust and integrated security mechanisms in financial applications. DevSecOps—a practice that integrates security into every phase of the software development lifecycle—has emerged as a leading strategy for mitigating risks and enhancing secure transaction processing. This study investigates the awareness, implementation, benefits, and challenges of DevSecOps in Nigerian banking and fintech organizations. Using a quantitative research approach, data were collected via structured questionnaires administered to 42 IT professionals including DevOps engineers, cybersecurity analysts, and compliance officers. Findings revealed high levels of awareness and partial implementation of key DevSecOps practices, such as secure coding and CI/CD pipeline integration. However, challenges including toolchain complexity, inadequate training, and poor cross-team collaboration were identified as significant barriers. The study concludes that while DevSecOps has the potential to transform the security landscape of financial transactions, its effectiveness is contingent upon skilled personnel, automated tools, and organizational alignment. Recommendations include targeted training, tool standardization, and enhanced collaboration models to ensure sustained security and operational efficiency in the banking sector.

Keywords: DevSecOps; Banking Applications; Financial Transactions; Cybersecurity; CI/CD Pipeline; Secure Coding; Automation; Nigeria; Fintech; Information Security

1. Introduction

In today's highly digitized financial ecosystem, securing online transactions has become not only a technical imperative but a business-critical priority. The increasing reliance on digital banking, mobile applications, and cloud-based infrastructure has made the financial services sector a prime target for cyberattacks, data breaches, and fraud. According to a report by IBM (2023), the financial sector experiences the highest average cost per data breach, surpassing $5.9 million, which underscores the critical need for robust and proactive security frameworks. Traditional security models—often reactive and siloed—are no longer sufficient to mitigate the growing complexity of threats in real-time environments (Gartner, 2022). To address these evolving security challenges, many financial institutions are embracing DevSecOps, an integrated approach that embeds security practices throughout the software development lifecycle (SDLC). DevSecOps, which stands for Development, Security, and Operations, is not merely a technical shift but a cultural transformation that encourages shared responsibility for security across cross-functional teams (Fryer, 2021). Unlike conventional DevOps models that prioritize speed and continuous delivery, DevSecOps integrates automated security checks, static and dynamic code analysis, policy enforcement, and real-time threat detection into every phase of development—from planning and coding to deployment and monitoring (JFrog, 2022).

The urgency to adopt DevSecOps in banking is further reinforced by increasing regulatory pressures and compliance requirements. Frameworks such as the General Data Protection Regulation (GDPR), the Payment Card Industry Data Security Standard (PCI DSS), and various national cybersecurity directives mandate that financial institutions

* Corresponding author: Ayobami Oluwadamilola Adebayo

implement secure development practices and ensure end-to-end protection of customer data. Furthermore, the rise of microservices, containerization, and continuous integration/continuous deployment (CI/CD) pipelines has created new vulnerabilities that traditional security practices are ill-equipped to address, making the adoption of DevSecOps not only advantageous but essential (Veritis, 2023). By shifting security to the left—that is, introducing it early in the development process—DevSecOps enables organizations to identify and fix vulnerabilities before they reach production. This approach not only reduces remediation costs but also minimizes system downtime and enhances customer trust (Digital.ai, 2023). Financial institutions that have successfully implemented DevSecOps report higher agility, improved collaboration between development and security teams, and faster time-to-market without compromising regulatory compliance or security integrity (Security Boulevard, 2023).

Therefore, this paper explores the best practices of DevSecOps in securing financial transactions within the banking industry. Through a qualitative analysis of implementation strategies and their effectiveness, the study aims to provide actionable insights for financial institutions seeking to fortify their applications against an increasingly hostile threat landscape.

2. Literature Review

The increasing digitization of banking services has driven a corresponding rise in cyber threats targeting financial transactions. According to IBM's annual *Cost of a Data Breach Report* (2023), the financial industry consistently ranks among the most targeted sectors, with breaches averaging $5.9 million per incident. These growing threats have forced banks to reconsider traditional security approaches, which typically apply security measures at the end of the software development lifecycle. Such methods often result in delayed response to vulnerabilities and inflated remediation costs (Ali et al., 2021). DevSecOps has emerged as a response to the shortcomings of traditional DevOps and security silos by integrating security into all phases of the software development lifecycle (Sharma & Sood, 2022). The concept of “shifting security left”—meaning incorporating security from the earliest stages of development—has gained traction for its ability to catch vulnerabilities earlier, reduce costs, and streamline compliance (Hassan et al., 2020). By embedding automated tools such as static application security testing (SAST) and dynamic application security testing (DAST) into CI/CD pipelines, DevSecOps minimizes human error and increases detection accuracy.

The application of DevSecOps in banking environments offers particular advantages given the industry’s regulatory burden. The implementation of frameworks such as the General Data Protection Regulation (GDPR) and the Payment Card Industry Data Security Standard (PCI DSS) demand strict adherence to data security and access control protocols. Studies show that banks using DevSecOps tools experience greater success in achieving compliance because automated security testing enforces policy compliance without manual intervention (Al-Bassam, 2021; Singh & Chatterjee, 2023). Moreover, security-as-code—where infrastructure and policy definitions are coded and version-controlled—has enabled better traceability and governance. Recent research also highlights the role of DevSecOps in improving organizational culture. Traditional organizational structures often placed security teams in isolated roles, leading to a lack of shared accountability (Santos et al., 2020). DevSecOps transforms this model by promoting collaboration across development, operations, and security disciplines, creating a culture of shared responsibility. Fryer (2021) emphasizes that cultural change is just as important as technological implementation in successful DevSecOps adoption.

Despite its advantages, DevSecOps is not without its challenges. One major issue is the steep learning curve associated with integrating security tools into existing development pipelines. Developers often lack sufficient training in secure coding practices, and security professionals may not be familiar with agile and DevOps workflows (Gonzalez & Smith, 2022). Furthermore, performance issues may arise when security scans slow down continuous delivery pipelines, thereby undermining productivity goals. Emerging technologies are also influencing DevSecOps practices. Artificial intelligence and machine learning are increasingly being incorporated to enhance threat detection and automate response strategies (Dhar et al., 2023). In banking, AI-powered tools are used for fraud detection and anomaly detection, complementing DevSecOps pipelines by providing adaptive risk assessments.

Additionally, cloud computing and microservice architectures present both opportunities and risks. While they enhance scalability and resilience, they also introduce new attack vectors, such as container vulnerabilities and misconfigured access controls. DevSecOps frameworks tailored to cloud-native environments—such as AWS Well-Architected Security Pillar or Azure DevOps Security—offer guidance on managing these risks (Microsoft, 2023; AWS, 2022).

In conclusion, the literature suggests that DevSecOps offers a promising approach to enhancing the security of banking applications. It supports not only technological improvements but also organizational and regulatory alignment. However, the success of DevSecOps depends on more than tool adoption; it requires cultural shifts, education, and a clear understanding of the evolving threat landscape.

3. Methodology

3.1. Research Design

This study adopted a descriptive research design to investigate the best practices of DevSecOps for securing financial transactions in banking applications. The choice of a descriptive approach was motivated by the need to obtain detailed, factual data directly from professionals involved in software development, cybersecurity, and financial technology management within the banking sector.

3.2. Data Collection Instrument

The primary instrument for data collection was a structured questionnaire. The questionnaire was designed to capture both quantitative and qualitative information from respondents. It was divided into three major sections. The first section gathered demographic information such as job role, years of experience, and the type of financial institution. The second section focused on the respondents' awareness and understanding of DevSecOps principles, tools, and practices. The third section explored the implementation status, perceived benefits, challenges, and security outcomes of DevSecOps in their respective organizations.

The questions were formulated based on a review of the relevant literature and industry frameworks including PCI DSS, OWASP DevSecOps guidelines, and the DevSecOps Reference Architecture.

3.3. Population and Sampling Technique

The study targeted IT professionals, security analysts, DevOps engineers, and compliance officers working within commercial banks, microfinance institutions, and fintech companies. A purposive sampling technique was used to select respondents with relevant expertise and roles directly related to software development and security in banking environments. A total of 50 questionnaires were distributed via email and professional networks, out of which 42 valid responses were received and used for analysis.

3.4. Data Analysis Procedure

The data obtained from the completed questionnaires were manually analyzed. Closed-ended responses were tabulated and summarized using **descriptive statistics** such as frequencies and percentages. These summaries were used to identify common patterns and trends regarding the adoption of DevSecOps, its perceived effectiveness, and associated challenges. Manual analysis was chosen over automated statistical tools to maintain a close engagement with the data and preserve the contextual richness of the responses. This approach allowed for a more nuanced interpretation of participants' perspectives, especially in identifying subjective barriers to implementation and opportunities for improvement.

3.5. Ethical Considerations

Participation in the study was voluntary. Respondents were informed about the purpose of the research and assured of the confidentiality and anonymity of their responses. No personally identifiable information was collected, and the data were used strictly for academic purposes.

4. Result and Discussion

Table 1 Demographic Information of Respondents

Variable	Category	Frequency	Percentage (%)
Job Role	DevOps Engineer	12	28.6%
	Cybersecurity Analyst	10	23.8%
	Software Developer	8	19.0%
	IT Manager	7	16.7%
	Compliance Officer	5	11.9%
Years of Experience	Less than 3 years	6	14.3%

	3–5 years	15	35.7%
	6–10 years	13	31.0%
	More than 10 years	8	19.0%
Type of Organization	Commercial Bank	22	52.4%
	Microfinance Bank	6	14.3%
	Fintech Company	14	33.3%

Most respondents were DevOps Engineers and Cybersecurity Analysts, indicating that the study captured the views of key professionals in security and software operations. The majority had 3–10 years of experience, and more than half worked in commercial banks, making the findings relevant to traditional banking environments.

Table 2 Awareness and Understanding of DevSecOps

Statement	**Response**	**Frequency**	**Percentage (%)**
Have you heard of DevSecOps?	Yes	38	90.5%
	No	4	9.5%
Does your organization apply DevSecOps principles?	Yes	30	71.4%
	No	12	28.6%
Do you understand the concept of “shift-left” security?	Yes	34	81.0%
	No	8	19.0%

There is a high level of awareness of DevSecOps among respondents, with 90.5% familiar with the term. However, actual implementation is somewhat lower at 71.4%, suggesting that while the concept is widely known, practical integration may be limited by organizational or technical barriers.

Table 3 Implementation of DevSecOps Best Practices

Best Practice	**Always**	**Sometimes**	**Never**
Security integrated into CI/CD	22	12	8
Automated security testing	18	15	9
Use of secure coding practices	30	9	3
Compliance checks during deployment	20	13	9
Regular security training	14	20	8

Secure coding is the most consistently applied practice (71.4% always), while regular security training and automated testing are less uniformly adopted. This gap indicates a need for stronger emphasis on automation and continuous education within teams to fully align with DevSecOps goals.

Table 4 Perceived Benefits and Challenges of DevSecOps

Aspect	**Frequency (Yes)**	**Percentage (%)**
Improved incident response	35	83.3%
Enhanced compliance and auditing	31	73.8%
Reduced vulnerabilities in apps	33	78.6%
Collaboration challenges	25	59.5%

Toolchain integration difficulty	28	66.7%
Lack of trained personnel	30	71.4%

Respondents acknowledged key benefits such as faster incident response and fewer vulnerabilities. However, challenges like integration complexity and lack of skilled personnel remain significant barriers. These findings highlight the need for targeted investments in tooling and workforce development

5. Conclusion

This study has revealed that DevSecOps practices are becoming increasingly recognized as essential for ensuring secure financial transactions in banking applications. The findings demonstrate a high level of awareness among IT professionals, with a significant proportion already applying DevSecOps principles in their organizations. Practices such as secure coding, integration of security into CI/CD pipelines, and compliance monitoring are moderately to widely implemented. However, the study also identified critical gaps in areas like regular security training, automated testing, and toolchain integration, which can limit the effectiveness of these practices.

The results highlight that while the benefits of DevSecOps—such as improved incident response, enhanced compliance, and reduced vulnerabilities—are widely acknowledged, implementation is still hindered by organizational challenges. Chief among these are a lack of trained personnel, collaboration difficulties between development and security teams, and the technical complexity of integrating various tools across the software development lifecycle. While DevSecOps presents a robust framework for improving transaction security in banking applications, its success is dependent on consistent practice, organizational buy-in, and targeted training efforts. Financial institutions that invest in addressing these limitations will be better positioned to meet the increasing security demands of digital banking environments.

Recommendations

Based on the findings of this research, the following recommendations are proposed to strengthen the adoption and effectiveness of DevSecOps in securing banking applications:

- **Invest in Workforce Training and Development**: Banks and fintech firms should provide continuous training and certification opportunities for developers, operations teams, and security personnel. This will build internal capacity and address the current skills gap in DevSecOps implementation.
- **Automate Security Testing within CI/CD Pipelines**: Organizations should adopt automated tools for vulnerability scanning, code analysis, and policy enforcement early in the development lifecycle to reduce human error and detect threats promptly.
- **Promote Cross-functional Collaboration**: Encourage integrated teamwork between development, security, and operations by creating shared goals, KPIs, and feedback loops. Adopting agile practices such as daily standups and retrospectives can foster mutual accountability.

By adopting these recommendations, financial institutions can enhance the resilience, integrity, and compliance of their banking applications, ultimately fostering trust and satisfaction among customers and regulators alike.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. Qualitative Research in Psychology, 3(2), 77–101.

[2] Creswell, J. W., & Poth, C. N. (2018). Qualitative Inquiry and Research Design: Choosing Among Five Approaches (4th ed.). SAGE Publications.

[3] PreEmptive. (2022). 10 DevSecOps Best Practices to Implement Now. Retrieved from https://www.preemptive.com/blog/10-devsecops-best-practices-to-implement-now/(PreEmptive In-App Protection)

[4] IBM. (2023). Best Practices for Hybrid Cloud Banking Applications: Secure and Compliant Deployment across IBM Cloud and Satellite. Retrieved from https://www.ibm.com/products/tutorials/best-practices-for-hybrid-cloud-banking-applications-secure-and-compliant-deployment-across-ibm-cloud-and-satellite(IBM)

[5] DevOps Digest. (2023). Enhancing Financial Transaction Security through DevOps Practices. Retrieved from https://www.devopsdigest.com/enhancing-financial-transaction-security-through-devops-practices(devopsdigest.com)

[6] Carnegie Mellon University Software Engineering Institute. (2023). 5 Challenges to Implementing DevSecOps and How to Overcome Them. Retrieved from https://insights.sei.cmu.edu/blog/5-challenges-to-implementing-devsecops-and-how-to-overcome-them/(SEI)

[7] Microsoft. (2023). What Is DevSecOps? Definition and Best Practices. Retrieved from https://www.microsoft.com/en-us/security/business/security-101/what-is-devsecops(Microsoft)

[8] U.S. Department of Defense Chief Information Officer. (2021). DevSecOps Fundamentals Guidebook. Retrieved from https://dodcio.defense.gov/Portals/0/Documents/Library/DevSecOpsTools-ActivitiesGuidebook.pdf(U.S. Department of Defense)

[9] Veritis. (2023). DevSecOps – For Bankers With Futuristic Vision. Retrieved from https://www.veritis.com/blog/devsecops-for-bankers-with-futuristic-vision/(Veritis)

[10] Everestek. (2023). Fortifying Financial Transactions through Advanced DevSecOps Solutions. Retrieved from https://blog.everestek.com/fortifying-financial-transactions-through-advanced-devsecops-solutions/(Everestek Blogs)