JIPTI
JURNAL INOVASI PENDIDIKAN DAN TEKNOLOGI INFORMASI

Volume 7 No. 1, April (2026)
E-ISSN: 2720-9725
P-ISSN: 2987-8462

Cloud Native Zero Trust API Gateway Architecture for Digital Banking Systems

Riama Santy Sitorus[1*], B. Junedi Hutagaol[2], Raipa Triana[3]
Information Technology, Asa University, Indonesia[1]
Information System, Asa University, Indonesia[23]

E-mail: riama@asaindo.ac.id[1*], junedi@asaindo.ac.id[2], raipa@asaindo.ac.id[3]

Abstract

The increasing reliance on Application Programming Interfaces (APIs) in digital banking has expanded the attack surface of distributed systems, exposing limitations in traditional perimeter-based security models within cloud-native environments. This study aims to develop a Cloud-native Zero Trust API Gateway architecture that enables secure, scalable, and compliant API communication in digital banking ecosystems. A qualitative research approach was employed using a systematic literature review combined with architectural synthesis and expert validation to identify key security requirements and design principles. The findings demonstrate that integrating Zero Trust principles such as continuous authentication, identity-centric access control, least-privilege enforcement, and micro-segmentation into cloud-native components, including API Gateways, service meshes, and identity management systems, enables consistent, multi-layered security enforcement across distributed services. The proposed model further aligns security controls with international standards, including OWASP API Security Top 10, PCI DSS, and ISO/IEC 27001, reducing fragmentation between regulatory compliance and architectural implementation. This study concludes that embedding Zero Trust Architecture (ZTA) principles within a cloud-native API Gateway provides a unified and adaptive security framework capable of addressing the dynamic and complex security challenges of modern digital banking systems.

Keywords: *API; API Gateway; Cloud-native; Digital Banking; Zero Trust Architecture (ZTA).*

Submitted: *2026-03-09.* ***Revision:*** *2026-03-22.* ***Accepted:*** *2026-04-05.* ***Publish:*** *2026-04-13.*

INTRODUCTION

The accelerating digitalization of the banking sector has positioned Application Programming Interfaces (APIs) as critical infrastructure for delivering modern financial services. APIs enable real-time integration across internal systems, third-party platforms, and customer-facing applications, supporting key functionalities such as payments, identity verification, and

Sitorus, R, S., Hutagaol, B, J., & Triana, R. (2026). Cloud-Native Zero Trust API Gateway Architecture for Digital Banking Systems. *Jurnal Inovasi Pendidikan dan Teknologi Informasi (JIPTI), 7(1), 255-268.* https://doi.org/10.52060/jipti.v7i1.4065

http://ejournal.ummuba.ac.id/index.php/JIPTI/

open banking services. At the same time, the adoption of cloud-native architecture characterized by microservices, containerization, and orchestration platforms such as Kubernetes has transformed the operational landscape of banking systems into highly distributed and dynamic environments. While these advancements enhance scalability and agility, they simultaneously introduce significant security challenges, particularly in protecting API communications against increasingly sophisticated cyber threats (Kaur & Saukko, 2022; Utomo & Rahman, 2024).

Over the past decade, research has explored API Security, cloud-native architecture, and Zero Trust Architecture (ZTA) as key approaches to addressing modern cybersecurity risks. API Security studies have largely focused on mitigating vulnerabilities outlined in the OWASP API Security Top 10, including broken object-level authorization, authentication flaws, and excessive data exposure (Almeida et al., 2016). In parallel, cloud-native architecture research emphasizes scalability, resilience, and automation through microservices and container orchestration, yet often treats security as an external or supplementary concern rather than an embedded architectural principle (Ajay Varma Indukuri, 2025; Gannon et al., 2017). Meanwhile, Zero Trust Architecture, as formalized by NIST SP 800-207, introduces a paradigm shift from implicit trust to continuous verification, identity-centric access control, and least-privilege enforcement (Rose et al., n.d.). Although several studies have attempted to apply Zero Trust concepts to API management and microservices security, these efforts remain largely fragmented, typically focusing on isolated mechanisms such as authentication, identity management, or network-level controls without offering a comprehensive architectural integration (Manne, 2025; Zanasi et al., 2024).

Existing approaches frequently rely on perimeter-based security assumptions, static access control mechanisms, or partially implemented Zero Trust controls that fail to address the dynamic and context-aware nature of cloud-native environments (Bayya, 2025; Samira et al., 2024). Furthermore, there is limited research that systematically maps Zero Trust principles—such as continuous authentication, adaptive authorization, and micro-segmentation—into core cloud-native components, including API Gateways, service meshes, identity providers, and policy enforcement mechanisms. This limitation is especially critical in digital banking, where security architectures must not only mitigate sophisticated API-based threats but also comply with stringent regulatory frameworks such as PCI DSS, ISO/IEC 27001, and secure API standards (PCI Security Standards Council, 2013).

In response to these limitations, this study advances a conceptual architectural model that integrates Zero Trust principles into a cloud-native API Gateway framework for digital banking systems. The primary objective is to develop a unified and systematic mapping of Zero Trust security controls across cloud-native architectural components, enabling continuous

verification, context-aware access decisions, and secure service-to-service communication.

This study contributes to the existing body of knowledge by proposing a unified Cloud-native Zero Trust API Gateway architecture that systematically integrates identity, network, and application-level security controls within a single framework. In addition, this research provides a comprehensive mapping of Zero Trust principles to key cloud-native components, including API Gateway, service mesh, identity providers, and policy enforcement mechanisms. Furthermore, the study introduces an alignment framework that embeds international security standards, such as PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001, directly into the architectural design. By doing so, this research bridges the gap between compliance requirements and practical API security implementation in digital banking systems.

METHOD

This study uses a qualitative descriptive approach to design a Cloud-Native Zero Trust API Gateway model for digital banking. The focus is on developing the concept and architecture, not on technical implementation or testing. The method combines literature review, industry standards, and concept synthesis to create a relevant and practical security architecture, as shown in Figure 1.

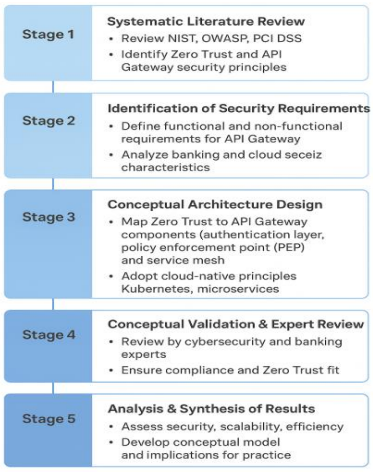

Figure 1 : Research method

A. Systematic literature analysis

This study adopts a structured Systematic Literature Review (SLR) to ensure methodological rigor and reproducibility. Literature was collected from major academic databases, including Scopus, IEEE Xplore, ScienceDirect, and Google Scholar, using keywords such as "API Security," "Zero Trust Architecture," and "Cloud-native Security."

Inclusion criteria covered peer-reviewed studies published between 2015–2025 focusing on API security, Zero Trust, or cloud-native architecture, while non-English, duplicate, and irrelevant studies were excluded. The selected literature was analyzed using thematic content analysis to identify key security principles, architectural components, and challenges. These findings were synthesized into design requirements that underpin the proposed Cloud-native Zero Trust API Gateway architecture

B. Conceptual Architecture Design

This stage includes mapping Zero Trust principles to key API Gateway

components such as the authentication layer, authorization server, service mesh, and policy enforcement point (PEP). The architecture is designed to operate within Cloud-Native environments, leveraging technologies like Kubernetes, container orchestration, and identity aware proxies. The design also emphasizes micro segmentation to isolate service interactions and the implementation of continuous verification mechanisms to ensure that each API request is validated based on identity, context, and applicable security policies.

C. Conceptual validation and Expert Review

The proposed Cloud-native Zero Trust API Gateway architecture was conceptually validated through an expert review process involving two senior experts with complementary professional backgrounds in banking cybersecurity and cloud-native system architecture. Both experts possess more than ten years of experience in designing, securing, and governing information systems within the financial sector, including extensive involvement in API Security, cloud-native banking platforms, and regulatory compliance initiatives.

To strengthen the validity of the proposed architecture, the expert evaluation instrument was developed based on key assessment dimensions derived from Zero Trust Architecture (NIST SP 800-207) and API Security best practices. The evaluation criteria include: (1) alignment with Zero Trust principles, (2) effectiveness in mitigating OWASP API Security risks, (3) feasibility in cloud-native environments, and (4) compliance with regulatory standards such as PCI DSS and ISO/IEC 27001.

The experts provided qualitative assessments and structured feedback using a semi-structured evaluation form. Their responses were analyzed using a qualitative coding approach, where key observations were categorized into themes such as architectural consistency, security coverage, and implementation feasibility. The feedback was systematically mapped to the architectural components, and necessary refinements were applied to improve policy enforcement placement, identity integration, and micro-segmentation mechanisms. This iterative validation process ensures that the proposed architecture is both conceptually sound and practically relevant for digital banking environments.

D. Analysis and Synthesis of Result

This stage analyzes and combines the research findings to identify the theoretical and practical impacts of the proposed model. It evaluates how the design meets security, efficiency, and scalability needs in digital banking. The result is a conceptual model showing how the architecture, data security, and supporting systems like identity, threat detection, and compliance tools work together.

The research relies on secondary data obtained from academic literature, industry reports, technical documentation, and security policies issued by international standard organizations. The literature selection criteria focus on publications from

the past ten years (2015–2025) addressing API security, Cloud-Native architectures, and Zero Trust applications in the financial sector. To ensure relevance and data quality, each source was analyzed using a content analysis approach to identify key themes, principles, and architectural components.

This combined analytical framework ensures that the resulting architecture is theoretically sound, practically relevant, and aligned with modern banking security needs. By embedding Zero Trust principles within a Cloud-Native framework, this research aims to contribute to the development of a sustainable, scalable, and resilient API security model capable of addressing current and emerging cyber threats in the digital banking landscape.

RESULT AND DISCUSSION

The literature review highlights the convergence of three fundamental concepts consists of Cloud-Native Architecture, Zero Trust Security, and API Gateway Technology in addressing the evolving cybersecurity needs of digital banking ecosystems. Studies emphasize that as financial institutions transition toward cloud environments, the complexity of managing distributed APIs and microservices introduces new attack surfaces (Kaur & Saukko, 2022; Utomo & Rahman, 2024). Rather than merely representing a technological paradigm shift, cloud-native architecture fundamentally redefines how security boundaries are established and enforced in distributed systems. As argued by (Gannon et al., 2017), the elasticity and decentralization of cloud-native systems dissolve traditional perimeter-based controls, thereby expanding the attack surface across microservices and APIs. This structural transformation explains why security can no longer rely on static, network-based trust assumptions, but must instead be dynamically enforced at the level of individual services and interactions.

The essential criteria for Cloud-Native systems include the ability to scale horizontally and adapt to changing workloads, the adoption of loosely coupled microservices that are independently deployable, and the use of automation in deployment, scaling, and recovery processes (Ajay Varma Indukuri, 2025). Moreover, containers play a critical role in encapsulating application components to ensure portability and consistency across environments. Collectively, these characteristics enable Cloud-Native applications to achieve greater operational efficiency, fault tolerance, and agility, aligning with modern digital transformation demands in sectors such as banking and financial services (Ajay Varma Indukuri, 2025; Gannon et al., 2017).

This convergence is not merely a technological trend but can be theoretically explained through the integration of Zero Trust Architecture (ZTA), cloud-native security principles, and API Gateway control frameworks. From a theoretical perspective, the increasing attack surface in distributed systems necessitates a shift from perimeter-based security toward identity-centric and context-aware security models (Rose et al., n.d.). From a Zero Trust theoretical perspective, this necessity

emerges from the foundational assumption that no implicit trust can be granted based on network location or system boundaries (Rose et al., n.d.).

Instead, trust must be continuously evaluated using identity, device posture, and contextual attributes. In cloud-native environments, where workloads are ephemeral and communication occurs across distributed services, this assumption becomes critical. The absence of fixed infrastructure boundaries invalidates traditional perimeter-based models, thereby explaining why continuous verification and dynamic policy enforcement are not optional enhancements but fundamental requirements.

From a regulatory and best practice perspective, three global frameworks PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001serve as foundational references for securing Cloud-Native API architectures in digital banking (PCI Security Standards Council, 2013; Rajgopal, 2025). The PCI DSS emphasizes protecting cardholder data through strong access control and encryption, while OWASP API Security Top 10 focuses on mitigating common API vulnerabilities that could expose sensitive financial information. Meanwhile, ISO/IEC 27001 provides a comprehensive information security management framework that ensures systematic implementation of controls, risk assessment, and continuous improvement. Rather than functioning as standalone compliance instruments, these frameworks implicitly reinforce the principles of Zero Trust by emphasizing continuous verification, strict access control, and data protection.

However, unlike traditional implementations where compliance is treated as an external layer, the findings of this study suggest that embedding these requirements directly into the architecture enables more consistent and enforceable security controls. This explains why the proposed model achieves not only regulatory alignment but also operational resilience, as security policies are inherently integrated into system design rather than retrofitted.

The correlation presented in Table 1 can be analytically interpreted as a convergence between compliance-driven security and architecture-driven security. Unlike traditional approaches where compliance frameworks are implemented as external controls, the proposed model embeds these requirements directly into the architecture through Zero Trust principles. This explains why the architecture is capable of achieving both regulatory compliance and operational security simultaneously.

From a Zero Trust perspective, the alignment with PCI DSS and ISO/IEC 27001 reflects the shift toward identity-based and continuously verified access control. Meanwhile, the integration with OWASP API Security Top 10 demonstrates that API Gateway security must operate not only at the infrastructure level but also at the application logic level. This finding is consistent with prior studies emphasizing that API vulnerabilities are often exploited

due to weak runtime enforcement rather than design flaws.

While prior studies (Zanasi et al., 2024) tend to treat compliance frameworks and architectural design as separate domains, this study demonstrates that such separation leads to fragmented policy enforcement and inconsistencies across distributed services. In contrast, the integrated mapping proposed in this research enables unified policy orchestration across identity, network, and application layers. However, this finding partially contrasts with (Samira et al., 2024), who argue that tight integration between compliance and architecture may reduce system flexibility. This study addresses this concern by leveraging cloud-native principles such as declarative configuration and container orchestration, which preserve adaptability while maintaining centralized control. This indicates that integration, when supported by cloud-native capabilities, does not necessarily compromise flexibility but instead enhances security consistency.

Table 1 : Correlation between Cloud-Native Zero Trust API Gateway principles and the three global standards

Security Standard / Framework	**Key Focus Areas**	**Correlation with Cloud-Native Zero Trust API Gateway**	**Implementation Relevance in Digital Banking**
PCI DSS (Payment Card Industry Data Security Standard)	Data protection, access control, network segmentation, encryption	Aligns with Zero Trust principles by enforcing strict authentication, encryption of API communication, and tokenbased access in API Gateway. Cloud-Native components enable continuous monitoring of payment API traffic.	Ensures secure handling of cardholder data in digital payment APIs, preventing unauthorized data access or leakage.
OWASP API Security Top 10	API vulnerability management, broken authentication, excessive data exposure	Guides API Gateway configuration to mitigate top API threats such as injection, broken authentication, and mass assignment through policy enforcement and request inspection.	Strengthens APIlevel defense against exploitation, ensuring compliance with secure coding and runtime validation practices.
ISO/IEC 27001	Information Security Management System (ISMS), risk assessment, continual improvement	Supports Zero Trust governance by institutionalizing access control, incident response, and	Provides enterpriselevel security governance framework for maintaining resilience and audit readiness in

	continuous monitoring financial APIs. policies across API Gateway environments.

This correlation underscores that implementing a Cloud-Native Zero Trust API Gateway in digital banking is not merely a technical enhancement but also a compliance driven architectural strategy. By mapping its core principles to PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001, financial institutions can achieve alignment between regulatory requirements, operational security, and architectural scalability. The synthesis of these frameworks enables proactive threat prevention, real time verification, and sustainable security governance critical pillars for the future of secure digital banking ecosystems.

A. Conceptual Architecture of the Cloud-Native Zero Trust API Gateway

This study proposes a Cloud-Native Zero Trust API Gateway Architecture designed to meet the comprehensive security and compliance requirements outlined in international standards such as PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001. The architecture aims to provide a secure, scalable, and compliant infrastructure for digital banking environments by integrating Zero Trust principles never trust, always verify, and assume breach into a Cloud-Native environment that supports microservices and distributed workloads.

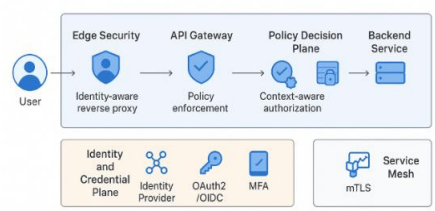

Figure 2 : High Level Architecture Cloud-Native ZTA API Gateway

The proposed architecture aims to establish a secure, scalable, and context aware API Gateway ecosystem for digital banking environments. It adopts Zero Trust principles namely never trust, always verify, least privilege access, and continuous validation while maintaining Cloud-Native agility through microservices, containers, and declarative automation. The architecture ensures that every transaction from the user to the backend service is verified, monitored, and enforced by contextual security controls at multiple layers. Each architectural component addresses specific compliance requirements and security objectives derived from PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001. Table 4.2 presents the correlation between the architectural components, Zero Trust principles, and the referenced standards (Jaiswal, 2025; Kumar, 2024; Oluomachi Eunice Ejiofor et al., 2025; Putra & Idrus, 2026; Sharanya Vasudev Prasad, 2025).

The mapping presented in Table 2 reveals that Zero Trust principles are not implemented as isolated mechanisms but are systematically distributed across

architectural layers. This layered enforcement model explains why the proposed architecture achieves defense-in-depth beyond traditional network security approaches. Each layer independently validates trust assumptions, thereby reducing the likelihood of single-point failure.

Compared to conventional architectures, where security controls are often centralized at the perimeter, this distributed enforcement aligns with cloud-native security principles that emphasize workload-level protection and decentralized policy execution (Ajay Varma Indukuri, 2025; Edo et al., 2022; Gambo & Almulhem, 2026; Zanasi et al., 2024).

Table 2 : Correlation of Architectural Components with Security Standards

Component	**Primary Function**	**Zero Trust Principle Addressed**	**Security Standard Alignment**	**Contribution to API Gateway Security**
Edge Security Layer (Reverse Proxy, WAF, Rate Limiter)	Protects entry point by authenticating users, filtering malicious requests, and enforcing rate limits.	Never trust, always verify	OWASP API 1–API10 (mitigates injection, data exposure, excessive requests); PCI DSS Req. 1 & 6 (network and app layer protection)	Prevents external threats, validates request integrity, enforces TLS encryption, and guards against DDoS and injection attacks.
API Gateway (Policy Enforcement Point – PEP)	Enforces policybased access control and request validation before routing.	Least privilege access	NIST 800207 (PEP enforcement); OWASP API 3, 4, 5; PCI DSS Req. 7 & 8 (restrict and authenticate access)	Ensures all traffic is inspected and authorized; provides runtime policy enforcement; prevents unauthorized API access.
Policy Decision Plane (PDP + Risk Engine)	Evaluates contextual and behavioral factors for adaptive authorization.	Continuous verification	NIST ZTA Framework (contextbased decisions); ISO/IEC 27001 A.9 & A.12	Implements riskbased, dynamic authorization; blocks highrisk transactions; supports ABAC and realtime trust

				scoring.
Identity & Credential Plane (IdP, OAuth2, OIDC, MFA)	Manages digital identities, authentication tokens, and credential validation.	Identitycentric trust	PCI DSS Req. 8 (user authentication); ISO/IEC 27001 A.9; OWASP API 2 (broken authentication)	Ensures verified user and service identity; supports shortlived tokens and MFA; enforces identity assurance.
Service Mesh Layer (mTLS, Micro segmentation)	Secures east–west traffic between microservices with encrypted and segmented communication.	Least privilege & Micro segmentation	NIST SP 800207; PCI DSS Req. 4; ISO/IEC 27001 A.13.1	Prevents lateral movement; ensures encrypted service to service communication; isolates workloads per trust zone.
Data Protection & Key Management Layer (KMS, Secrets Manager)	Manages cryptographic keys, secrets, and encryption policies.	Defenseindepth	PCI DSS Req. 3 & 4 (data protection); ISO/IEC 27001 A.10	Ensures data encryption at rest and in transit; secures keys via rotation and lifecycle management.
Observability & Threat Detection Layer (SIEM, UEBA, Anomaly Detection)	Provides visibility, event correlation, and anomaly monitoring.	Continuous monitoring and feedback	NIST CDM; ISO/IEC 27001 A.12.4; OWASP API 10 (insufficient logging)	Enables proactive threat detection; logs all access; supports behavioral analytics for anomalybased response.
Compliance & DevSecOps Layer	Embeds security checks into CI/CD through policyascode and complianceascode automation.	Automation and continuous assurance	PCI DSS Req. 6 & 11; ISO/IEC 27001 A.18	Ensures every build and deployment complies with security standards; maintains immutable, auditable configurations.
Governance & Continuous Improvement Layer	Manages policies, incident response, and continuous improvement of ISMS.	Adaptive trust and governance	ISO/IEC 27001 (ISMS framework); NIST ZTA Governance	Maintains ongoing compliance; governs incident handling, policy lifecycle, and periodic risk assessments.

B. Thematic Analysis of Research Findings

The thematic findings presented in Table 3 can be interpreted as core operational manifestations of Zero Trust principles within a cloud-native environment. Continuous authentication and context-aware authorization reflect the transition from static identity verification to dynamic trust evaluation, as emphasized in Zero Trust theory (Rose et al., n.d.).

Micro-segmentation and service isolation further demonstrate how cloud-native architectures operationalize Zero Trust at the infrastructure level. This explains why the system is more resilient to lateral movement attacks, as each microservice operates within a limited trust boundary. This finding is consistent with prior studies on service mesh security, which highlight the role of mTLS and traffic segmentation in securing east–west communication.

Observability and anomaly detection provide the necessary feedback loop for continuous monitoring, a key requirement in both Zero Trust and ISO/IEC 27001 frameworks. However, unlike traditional monitoring approaches, the integration with behavioral analytics enables proactive threat detection rather than reactive response.

While these findings align with existing research, some studies suggest that implementing such comprehensive controls may introduce operational complexity and performance overhead (Putra et al., 2023; Samira et al., 2024). Therefore, although the proposed architecture enhances security, it also highlights the need for future research on performance optimization and scalability trade-offs.

Table 3 : Thematic Analysis of Research Findings

Theme	Key Findings	Implications
Continuous Authentication	Zero Trust requires every entity (user, service, or device) to be verified at each interaction.	Reduces risks of session hijacking and credential theft in API access.
Context Aware Authorization	Access decisions should consider contextual attributes (location, device type, behaviour).	Enhances dynamic trust and supports adaptive access control in digital banking.
Micro segmentation and Service Isolation	Service mesh enables granular segmentation across microservices.	Prevents lateral movement and limits the blast radius of security breaches.
Observability and Anomaly Detection	Realtime traffic monitoring detects malicious or anomalous API behaviour.	Improves proactive incident response and threat hunting capabilities.
Compliance Automation	Integrating regulatory requirements into architectural design ensures continuous adherence.	Simplifies audits and reduces human error in compliance management.

These themes collectively demonstrate that the implementation of Zero Trust in API Gateway design extends beyond authentication mechanisms into a broader paradigm of continuous, context-aware decision-making. Unlike traditional security models that rely on static access control, the findings indicate that effective security in cloud-native environments requires adaptive policies driven by real-time contextual intelligence.

This reinforces the argument that Zero Trust is not merely a security framework but a dynamic operational model that fundamentally reshapes how trust is established, evaluated, and enforced in distributed systems.

C. Practical Implementation Scenario in Digital Banking

To illustrate the practical applicability of the proposed architecture, a typical digital banking API interaction scenario is considered. When a user initiates a mobile banking transaction, the request first passes through the Edge Security Layer, where initial validation such as TLS enforcement, IP filtering, and rate limiting is applied.

The request is then forwarded to the API Gateway, acting as the Policy Enforcement Point (PEP), which validates access tokens and enforces policy rules. Simultaneously, the Policy Decision Point (PDP) evaluates contextual attributes such as user behavior, device profile, and transaction risk level.

If the request meets the required trust level, it is securely routed through the service mesh, where mutual TLS (mTLS) ensures encrypted service-to-service communication. Each microservice verifies the request independently, enforcing micro-segmentation and least-privilege access.

Throughout the process, all activities are logged and monitored by the observability layer, enabling real-time anomaly detection and incident response. This scenario demonstrates how Zero Trust principles are continuously enforced across multiple layers, ensuring secure and adaptive API communication in digital banking environments.

CONCLUSSION

This study demonstrates that the integration of Zero Trust principles within a cloud-native API Gateway architecture enables consistent, multi-layered security enforcement across distributed digital banking systems. The findings reveal that security effectiveness is achieved not through isolated controls, but through the systematic alignment of identity-centric verification, micro-segmentation, and context-aware authorization across architectural components. By mapping these mechanisms to established standards such as PCI DSS, OWASP API Security Top 10, and ISO/IEC 27001, this study provides a structured approach for embedding compliance requirements directly into system architecture, reducing fragmentation between regulatory and operational security. The primary contribution lies in proposing a unified architectural model that integrates API Gateway, identity management, and service mesh into a coherent Zero Trust framework, addressing inconsistencies observed in prior fragmented

implementations. However, the absence of empirical validation highlights the need for future studies to evaluate performance trade-offs, scalability, and real-world applicability of the proposed model in production environments.

REFERENCES

Ajay Varma Indukuri. (2025). Cloud-native transformation: Architectural principles and organizational strategies for infrastructure modernization. *World Journal of Advanced Research and Reviews*, *26*(1), 3914–3926. https://doi.org/10.30574/wjarr.2025.26.1.1467

Almeida, C. S. de, Miccoli, L. S., Andhini, N. F., Aranha, S., Oliveira, L. C. de, Artigo, C. E., Em, A. A. R., Em, A. A. R., Bachman, L., Chick, K., Curtis, D., Peirce, B. N., Askey, D., Rubin, J., Egnatoff, D. W. J., Uhl Chamot, A., El-Dinary, P. B., Scott, J.; Marshall, G., Prensky, M., … Santa, U. F. De. (2016). No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title. *Revista Brasileira de Linguística Aplicada*, *5*(1), 1689–1699.

Bayya, A. K. (2025). *Cutting-Edge Practices for Securing APIs in FinTech : Implementing Adaptive Security Models and Zero Trust Architecture International Journal of Applied Engineering & Technology CUTTING-EDGE PRACTICES FOR SECURING APIS IN FINTECH : IMPLEMENTING ADAPTIVE S. January*.

Edo, O. C., Tenebe, T., Etu, E., Ayuwu, A., Emakhu, J., & Adebiyi, S. (2022). Zero Trust Architecture: Trend and Impacton Information Security. *International Journal of Emerging Technology and Advanced Engineering*, *12*(7), 140–147. https://doi.org/10.46338/ijetae0722_15

Gambo, M. L., & Almulhem, A. (2026). Zero Trust Architecture: A Systematic Literature Review. *Journal of Network and Systems Management*, *34*(1). https://doi.org/10.1007/s10922-025-09998-x

Gannon, D., Barga, R., & Sundaresan, N. (2017). Cloud-Native Applications. *IEEE Cloud Computing*, *4*(5), 16–21. https://doi.org/10.1109/MCC.2017.4250939

Jaiswal, D. (2025). Zero-Trust Architecture for Telecom API Security: A Framework for the Communications Economy. *European Modern Studies Journal*, *9*(4), 11–21. https://doi.org/10.59573/emsj.9(4).2025.2

Kaur, H., & Saukko, P. (2022). Social access: role of digital media in social relations of young people with disabilities. *New Media and Society*, *24*(2), 420–436. https://doi.org/10.1177/14614448211063177

Kumar, R. (2024). An Extensive Analysis on Zero Trust Architecture. *International Journal of Innovative Science and Research Technology (IJISRT)*, *9*(5), 1056–1061. https://doi.org/10.38124/ijisrt/ijisrt24may1225

Manne, T. A. K. (2025). Implementing Zero Trust Architecture in Multi-Cloud Environments. *International Journal of Computing and Engineering*, *7*(3), 74–82. https://doi.org/10.47941/ijce.2753

Oluomachi Eunice Ejiofor, Oluwafemi Olusoga, & Ahmed Akinsola. (2025). Zero trust architecture: A paradigm shift in network security. *Computer*

Science & IT Research Journal, *6*(3), 104–124. https://doi.org/10.51594/csitrj.v6i3.1871

PCI Security Standards Council. (2013). PCI DSS Cloud Computing Guidelines. *Security Standard Council*, *February*, 52.

Putra, Y. I., & Idrus, A. (2026). Determination of Technopreneurship, Work Motivation, Digital Literacy on the Work Readiness of Information Technology Students. *Jurnal Penelitian Pendidikan IPA*, *12*(3), 84–92. https://doi.org/10.29303/jppipa.v12i3.14407

Putra, Y. I., Kusmana, A., & Fitrah, Y. (2023). Falsifikasi sebagai pedoman Memahami Informasi di Media Sosial secara Objektif. *Jurnal Inovasi Pendidikan Dan Teknologi Informasi (JIPTI)*, *4*(2), 289–295. https://doi.org/10.52060/pti.v4i2.1515

Rajgopal, P. R. (2025). Secure Enterprise Browser - A Strategic Imperative for Modern Enterprises. *International Journal of Computer Applications*, *187*(33), 53–66. https://doi.org/10.5120/ijca2025925611

Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (n.d.). *Zero Trust Architecture NIST Special Publication 800-207*.

Samira, Z., Weldegeorgise, Y. W., Osundare, O. S., Ekpobimi, H. O., Kandekere, R. C., Researcher, I., & Texas, D. (2024). *API management and cloud integration model for SMEs*.

Sharanya Vasudev Prasad. (2025). Zero trust architecture: The future of enterprise security. *World Journal of Advanced Engineering Technology and Sciences*, *15*(1), 660–666. https://doi.org/10.30574/wjaets.2025.15.1.0247

Utomo, B. C., & Rahman, A. A. (2024). Analisis Kesadaran Keamanan Data Pribadi pada Pengguna E-Wallet DANA. *Jurnal Riset Sains Dan Teknologi*, *8*(2), 155–166.

Zanasi, C., Russo, S., & Colajanni, M. (2024). Flexible zero trust architecture for the cybersecurity of industrial IoT infrastructures. *Ad Hoc Networks*, *156*(May 2023). https://doi.org/10.1016/j.adhoc.2024.103414