Journal of Information Systems and Informatics

Vol. 7, No. 3, September 2025 e-ISSN: **2656-4882** p-ISSN: **2656-5935**

DOI: 10.51519/journalisi.v7i3.1219 Published By DRPM-UBD

Designing a Zero Trust Architecture for Securing API Gateways in Digital Banking Systems

Riama Santy Sitorus[1], B. Junedi Hutagaol[2]

[1]Information Technology, Asa Indonesia University, Jakarta, Indonesia
[2]Information System, Asa Indonesia University, Jakarta, Indonesia
Email: [1]riama@asaindo.ac.id, [2]junedi@asaindo.ac.id

Abstract

In the era of digital banking transformation, Application Programming Interfaces (APIs) are essential for system integration and customer-facing innovations but also increase exposure to cyber security risks such as credential theft, API abuse, data breaches, and unauthorized access. This research proposes a conceptual Zero Trust Architecture (ZTA) model specifically designed to secure API Gateways in digital banking systems. Adopting a conceptual design methodology comprising literature review, component identification, architectural modelling, standards-based evaluation, and recommendation development the study introduces a framework that integrates core Zero Trust principles. Strong identity verification counters credential misuse, dynamic access control mitigates unauthorized access, encryption protects sensitive financial data, continuous monitoring identifies abnormal traffic, and real-time behavioral analytics prevents API abuse. Each component is mapped to relevant industry standards, ensuring resilience and regulatory compliance. Beyond the conceptual design, the findings highlight practical implications: applying ZTA at the API Gateway strengthens cyber security defenses against modern API threats, supports regulatory readiness, and provides banks with a structured roadmap for secure digital services. The study concludes that the proposed model delivers a comprehensive foundation for secure API communication in digital banking and actionable guidance for future implementation and research.

Keywords: Access control; API Gateway; Cyber Security; Digital Banking; Zero Trust Architecture.

1. INTRODUCTION

In the era of digital transformation, banking services increasingly rely on Application Programming Interfaces (APIs) to enable fast, efficient, and integrated transactions. APIs play a critical role in connecting various banking systems with both internal applications and third-party platforms, fostering more flexible and innovative financial services [1], [2]. However, the growing dependency on APIs has also introduced new security challenges that demand more sophisticated protection approaches [3].

Currently, many banks secure their APIs through traditional mechanisms such as perimeter firewalls, static access tokens, and rule-based authentication. While these measures provide a baseline of protection, they often fall short in addressing modern attack vectors. For example, static credentials can be stolen or reused, perimeter-based defenses are ineffective against insider threats or compromised accounts, and limited monitoring capabilities hinder real-time detection of abnormal behaviors. As a result, API Gateways the central points for managing and routing API traffic remain vulnerable to threats such as credential theft, abuse, unauthorized access, and man-in-the-middle attacks. API-related threats include abuse, unauthorized access, data breaches, and man-in-the-middle attacks. For instance, API abuse occurs when attackers exploit vulnerabilities to gain unauthorized access to sensitive information. A study by [4] revealed that 51% of mobile banking users in Indonesia had experienced cybercrime attempts, with 21% reporting actual victimization. In the United States, API-related data breaches accounted for an estimated financial loss between $12 and $23 billion in 2022 alone. These realities underscore the urgency of implementing robust security strategies for managing API access and communication in digital banking operations.

One of the most effective approaches to secure APIs is through the deployment of an API Gateway. Modern API Gateways are not only designed to support scalability and performance, but also to offer strong security capabilities [3].Selecting the right API Gateway solution can significantly reduce security risks and improve operational efficiency [5]. [6] Identified several API Gateway technologies that demonstrate strong potential as cybersecurity solutions within the banking industry. However, the effectiveness of an API Gateway is not solely determined by its technical features [7], but also by the underlying security architecture applied [8].

A security architecture gaining increasing relevance for API protection is the Zero Trust Architecture (ZTA) [9]. Traditional perimeter-based security models—such as firewalls and VPNs—are no longer sufficient to safeguard modern, distributed banking systems. These models are vulnerable to lateral movement attacks due to their inherent “trust but verify” philosophy within internal networks [10], even though today’s threats may originate from inside the system itself. ZTA addresses this limitation by adopting a "never trust, always verify" principle, asserting that no entity—internal or external—should be trusted by default [11]. In the context of API security, this means every access request must be thoroughly validated, authenticated, and authorized before being granted. ZTA also incorporates encryption, real-time monitoring, and security analytics to detect and respond to suspicious activity [12].

Implementing Zero Trust within an API Gateway aligns with established industry standards such as NIST 800-207 (Syed and Shah 2022), OWASP API Security Top 10 [13], and PCI DSS [9], all of which emphasize strict access controls and proactive threat mitigation. These standards provide critical guidance for designing API security systems that can both withstand cyberattacks and ensure compliance with financial regulations.

Although prior studies on Zero Trust Architecture (ZTA) have primarily focused on general network security or endpoint protection, in-depth analyses specifically addressing ZTA design and implementation for API Gateways in banking environments remain scarce. To fill this gap, this study proposes a conceptual architectural model of a Zero Trust-based API Gateway tailored for digital banking systems. It aims to identify the essential components required for alignment with industry security standards and provide actionable recommendations for practical implementation.

This research is limited to conceptual and architectural design and does not include technical testing or performance evaluation. Nonetheless, it offers valuable insights for developing a more resilient API security architecture in the banking sector and serves as a reference for future research and implementation efforts.

2. METHODS

This study employs a conceptual and architectural design approach without involving technical testing or direct implementation. The steps undertaken in this study are as shown in Figure 1.

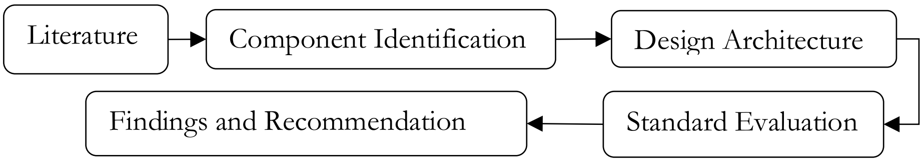

Figure 1. Research Method

The first step involves conducting an in-depth literature review on Zero Trust Architecture (ZTA), API Gateways, and the implementation of Zero Trust principles in digital banking security systems. The literature reviewed includes references covering the foundational principles of Zero Trust, key components of API Gateways, and industry standards such as NIST 800-207, OWASP API Security, and PCI DSS. The purpose of this step is to obtain a comprehensive understanding of Zero Trust architecture, explore challenges in its implementation within API Gateways, and examine existing practices described in previous research.

After building a theoretical foundation, the next step is to identify the key components necessary for designing Zero Trust architecture tailored to API Gateways in the banking sector. This includes authentication and authorization mechanisms to validate user identities and ensure that access is granted only to authorized entities. It also includes the application of encryption to safeguard data transmitted through APIs, and the use of monitoring and analytics tools to observe API activity in real time and detect threats or abnormal behavior.

Based on the identified components, the third step focuses on designing a conceptual and architectural model of Zero Trust for API Gateways in digital banking. This model illustrates how various components work together to form a resilient and adaptive API security system. The architecture ensures that every API access request is authenticated, verified, and monitored. It incorporates access control policies based on Zero Trust principles, emphasizes the development of dynamic and fine-grained authorization rules, integrates real-time monitoring and analytics tools for threat detection, and applies end-to-end encryption to ensure the confidentiality and integrity of API communication.

The fourth step is to evaluate the alignment of the proposed Zero Trust architecture with relevant industry standards, including NIST 800-207, OWASP API Security, and PCI DSS. This evaluation compares the components and mechanisms included in the design with the requirements and recommendations provided by these standards. The objective is to verify that architecture not only addresses a wide range of cyber threats but also fulfils regulatory compliance expectations in the financial services industry.

The final step is to formulate key findings and practical recommendations derived from the study. These recommendations aim to support the implementation of Zero Trust within API Gateways in digital banking systems. They include suggestions for step-by-step implementation, strategies to overcome potential challenges during integration with existing infrastructures such as costs, human resources, and technological change and guidance for risk management, particularly in the dynamic administration of user authorization and access control.

3. RESULTS AND DISCUSSION

This section presents the results of the conceptual and architectural design of a Zero Trust-based API Gateway for digital banking, followed by a comprehensive discussion.

3.1. Zero Trust Architecture Component Mapping

While NIST's ZTA framework [10] provides a generalized model, this study extends its application to API Gateway-specific threats in banking, such as credential stuffing and API abuse, which demand finer-grained controls than traditional network-level Zero Trust implementations. The first result of this study is the identification and classification of critical Zero Trust Architecture (ZTA) components relevant to securing API Gateways in the banking sector [14], [15]. These components are derived from existing literature and analysed in the context of digital banking operations.

Authentication and Authorization emerge as the cornerstone of ZTA [10], ensuring that only verified and authorized entities can access protected APIs [16]. The model emphasizes the use of Identity and Access Management (IAM) systems with multi-factor authentication (MFA), OAuth 2.0 tokens, and role-based or attribute-based access controls (RBAC/ABAC) [17]. This component mitigates risks related to impersonation, stolen credentials, and session hijacking.

Encryption, both in transit and at rest, is enforced throughout API communications [10]. The model recommends Transport Layer Security (TLS) 1.3 [10] and advanced key management mechanisms to protect sensitive financial data [16]. Encryption ensures that even if data is intercepted or exfiltrated, it remains unreadable without the appropriate decryption keys [18], [19]. Monitoring and Analytics form the continuous trust evaluation mechanism of ZTA. Real-time traffic monitoring, API call logging, and security information and event management (SIEM) systems are integrated into the API Gateway. Anomaly detection models using machine learning (ML) techniques are included to identify behavioural deviations, such as excessive requests or abnormal access patterns, which may signal credential abuse or bot activity [10], [20].

Policy Enforcement Engine serves as the dynamic decision layer. It enforces least-privilege access policies by evaluating real-time contextual attributes such as user identity, device posture, geolocation, and access time before granting or denying requests. Policies are updated continuously based on the evolving threat landscape and user behaviour [10], [17], [20]. Those components mapping provide a foundational blueprint for implementing ZTA in banking APIs and enables the formulation of a security posture that is identity-centric, data-driven, and continuously adaptive.

3.2. Conceptual Design Model of Zero Trust for API Gateway

Based on the identified components, a conceptual architectural model of a Zero Trust-based API Gateway has been constructed. The model follows a layered security approach and is designed to be integrated within existing banking system architectures with minimal disruption. At the core of the model is the API Gateway, which acts as the policy enforcement point. It intercepts all incoming API requests and interacts with an external Policy Decision Point (PDP) to evaluate access based on real-time identity verification and contextual attributes. The PDP communicates with external Identity Providers (IdPs), such as Active Directory or OAuth servers, to authenticate users or systems.

Surrounding the gateway are the supporting services such as Security Analytics Engine, which continuously monitors API traffic and leverages both signature-based and behaviour-based detection. Audit and Logging System, which records every API transaction, including metadata such as user ID, IP address, time, and action taken, to support forensic analysis and compliance. Encryption Layer, which secures both incoming and outgoing data flows using symmetric and asymmetric encryption protocols. Trust Evaluation Engine, which recalculates the trust score of every user or system based on behavioural history and contextual factors. The dsign also includes adaptive response mechanisms, such as rate-limiting, token revocation, dynamic re-authentication, or traffic blocking when anomalies are detected. These mechanisms allow the system to not only detect threats but also respond automatically before damage occurs.

3.3. Proposed Architecture: Zero Trust API Gateway for Digital Banking

The following sections provide a detailed explanation of each component in the architecture and how they integrate to form a unified Zero Trust model. This architecture consists of some components such as External Client which represents any external user or third-party service attempting to access banking APIs such as mobile banking users, fintech platforms, or corporate partners. In a Zero Trust model, no request from an external client is trusted by default. Addresses risks such as API abuse and unauthorized access from untrusted sources. The API Gateway acts as the primary entry point for all API traffic. It performs request validation (e.g., schema, tokens), Rate limiting and throttling. Initial authentication checks. Forwarding requests to the Policy Decision Point (PDP). Serves as the first line of defence and enforcement of Zero Trust policies for all API interactions.

The proposed architecture is designed to address the rising security challenges associated with API usage in digital banking. While previous research has discussed Zero Trust principles at the network or endpoint level, this study focuses specifically on their application within API Gateway environments. The architecture integrates Zero Trust components to ensure that every API request is authenticated, authorized, encrypted, and continuously monitored.

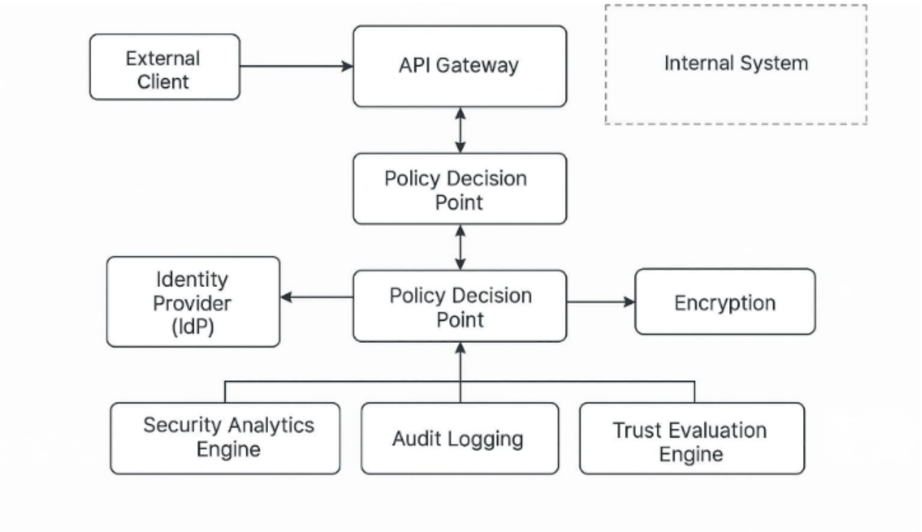

Figure 2. Zero Trust API Gateway Architecture

Policy Decision Point (PDP) is the core decision-making engine for access control. It evaluates incoming API requests by consulting Identity verification from the Identity Provider, Risk assessments from the Trust Evaluation Engine, Behavioural insights from the Security Analytics Engine, based on these inputs, the PDP either grants or denies access to the internal system. This is the main component to implements the Zero Trust principle of "never trust, always verify." Internal System which including core banking service such as transaction processing, account information, or KYC modules that the client ultimately seeks to access. Access to these services is only permitted after successful evaluation and approval by the PDP. The main objective of zero trust is protecting sensitive banking operations from exposure to unauthorized or potentially harmful requests. Identity Provider (IdP) handles authentication of all API users and clients through OAuth2 / OpenID Connect, Multi-Factor Authentication (MFA), Role-Based or Attribute-Based Access Control (RBAC/ABAC). Prevents impersonation and enforces identity verification before granting access.

Encryption module ensures end-to-end encryption of all data passing through the API Gateway using protocols like TLS 1.3 or mTLS (Mutual TLS) and Signed and encrypted JSON Web Tokens (JWT). It will Prevents data breaches and man-in-the-middle attacks, addressing OWASP and PCI DSS requirements. Trust Evaluation Engine enables adaptive security decisions based on contextual risk analysis. This component dynamically calculates a risk or trust score for every request based on user context (location, device, time), historical behavior patterns. External threat intelligence, the score influences the PDP's access control decision.

All actions and decisions taken by the Gateway and PDP are comprehensively logged which maintain by Audit log component. Important data is logged including access attempts, authentication results, policy decisions and outcomes. This component will support forensic analysis, auditing, and compliance with financial regulations. Utilizing artificial intelligence in this case will give an extra benefit, it will enable proactive threat detection and response. Security Analytics Engine applies machine learning and analytics to detect anomalies and potential threats by analyzing logs and real-time traffic. It can feed alerts into the PDP or notify security teams.

To ensure the proposed architecture effectively addresses the objectives outlined in this study, a detailed alignment analysis was conducted. This analysis evaluates how each architectural component contributes to solving the identified security challenges and fulfilling the goals of Zero Trust implementation within API Gateways in digital banking systems. Table 1 presents a structured mapping between the core research objectives and the architectural components designed to meet them. This alignment highlights the practical relevance of the proposed model and its potential to bridge the gap in current literature concerning Zero Trust applications at the API layer in banking environments.

Table 1. Proposed Model API Layer

Research Focus	Addressed by
API abuse and unauthorized access	API Gateway, Identity Provider, Policy Decision Point
Strict validation for each access request	PDP, Trust Evaluation Engine, IdP
Man-in-the-middle attacks and data leakage	Encryption Module
Real-time detection of anomalies	Security Analytics Engine
Compliance with industry standards (NIST, OWASP, PCI DSS)	Encryption, Audit Logging, PDP
Research gap in applying ZTA at the API Gateway level in banking	Contribution of a novel, banking focused Zero Trust API Gateway architecture

This architecture offers a conceptual and standards-aligned solution to enhance API security in digital banking. By integrating Zero Trust principles across the API lifecycle from authentication to analytics it provides a robust, scalable, and regulation-compliant security model that mitigates modern cyber threats.

3.4. Evaluation Against Industry Standard

The designed architecture was evaluated for its compliance with three key industry frameworks: NIST SP 800-207, OWASP API Security Top 10, and PCI DSS v4.0. The model satisfies the core tenets of NIST 800-207, such as strict identity verification, dynamic policy enforcement, and continuous monitoring. It also aligns with NIST's recommendation to decouple the control plane from the data plane—evident in the separation between the API Gateway (data path) and Policy Decision Point (control path). Regarding OWASP API Security Top 10, the architecture addresses multiple attack vectors, including:

1) Broken Object Level Authorization (BOLA) and Excessive Data Exposure through strict identity and payload validation.
2) Broken Authentication via MFA and token-based identity control.
3) Lack of Monitoring by integrating real-time analytics and logging mechanisms.

Concerning PCI DSS, the architecture complies with key requirements such as encrypted transmission of cardholder data, user access control, continuous logging, and real-time security monitoring. These capabilities are critical for ensuring regulatory compliance in banking and financial services. The architecture was evaluated against NIST SP 800-207, OWASP API Security Top 10, and PCI DSS v4.0, as shown in Table 2.

Table 2. Architecture Standard

Standard	Alignment Evidence
NIST 800-207	Identity-centric policies, continuous monitoring, separation of control and data plane.
OWASP API Top 10	Mitigates BOLA, broken authentication, and lack of monitoring through identity control and analytics.
PCI DSS	Ensures encrypted transmission, logging, and access control for financial compliance.

The evaluation demonstrates the model’s strong alignment with cybersecurity and regulatory best practices. The evaluation demonstrates that the proposed architecture is not only conceptually sound but also practically aligned with recognized security benchmarks.

3.5. Implementation Considerations and Strategic Implications

While architecture presents a robust model for API protection, its implementation in real-world banking systems introduces several technical and strategic considerations. Technical challenges include integration with legacy banking systems, ensuring high availability and performance of the API Gateway under Zero Trust constraints, and managing the complexity of policy configuration. Continuous policy tuning, token expiration, session handling, and encryption key rotation require advanced orchestration tools and automation frameworks. From a strategic perspective, the implementation of Zero Trust brings several advantages. It reduces the attack surface of digital banking APIs, improves organizational security posture, and builds customer trust by ensuring sensitive data protection. Furthermore, embedding Zero Trust in the API layer supports secure Open Banking initiatives, enabling partnerships with fintech providers without compromising core infrastructure. The adoption of ZTA also aligns with the trend toward DevSecOps in banking IT operations, embedding security earlier in the software development lifecycle and automating threat response mechanisms. However, decision-makers must consider costs, resource allocation, skill gaps, and organizational change management to achieve successful adoption.

The proposed conceptual architecture introduces a Zero Trust-based security framework specifically designed for API Gateways in digital banking environments. While Zero Trust Architecture (ZTA) has gained widespread attention in recent years, much of the existing literature concentrates on its application to general IT infrastructure, network perimeter defense, and endpoint protection. This study extends the discourse by focusing on the API layer, a critical and often vulnerable surface in the modern digital banking ecosystem. The proposed model demonstrates how core Zero Trust components—such as identity-centric access control, policy-based decision-making, continuous monitoring, and behavioral analytics—can be cohesively integrated to address domain-specific threats such as API abuse, credential theft, and unauthorized data access.

By aligning the architectural design with industry standards including NIST SP 800-207, OWASP API Security Top 10, and PCI DSS v4.0, this research ensures both technical soundness and regulatory compliance. This alignment validates the model's applicability in highly regulated financial sectors, where strict enforcement of access control, encryption, and monitoring is required. For example, the implementation of end-to-end encryption using TLS 1.3 addresses the requirements of PCI DSS for secure data transmission, while the integration of identity and behavior-based controls directly mitigates OWASP-identified vulnerabilities such as Broken Object Level Authorization (BOLA) and Broken Authentication.

The architecture's layered approach provides not only a conceptual foundation but also a strategic roadmap for financial institutions seeking to modernize their API security infrastructure. The inclusion of components like a Trust Evaluation Engine and Security Analytics Engine illustrates the importance of contextual and adaptive access control mechanisms. These capabilities support the shift from static, perimeter-based security models to dynamic, risk-aware decision-making processes that reflect the Zero Trust principle of "never trust, always verify." Furthermore, the ability to detect anomalies and respond automatically through rate-limiting, token revocation, or re-authentication adds a layer of resilience against advanced threats.

However, despite its strengths, the practical implementation of the proposed architecture may face several challenges. Integrating Zero Trust principles into legacy banking systems could introduce architectural complexity, performance concerns, and operational overhead. Real-time policy enforcement, identity verification, and behavioral analysis require scalable computing resources and automation tools to minimize latency and ensure high availability. Additionally, transitioning to a Zero Trust model requires more than just technical redesign; it involves organizational readiness, skill development, and change management to align people, processes, and technology.

The contribution of this study lies in its ability to translate abstract Zero Trust principles into an actionable API security architecture tailored to the specific requirements and constraints of digital banking. By proposing a standards-aligned model, it offers a structured reference for future implementation efforts while also laying the groundwork for further research. In particular, the integration of artificial intelligence to enhance real-time decision-making and the application of Zero Trust in hybrid or multi-cloud API ecosystems represent promising directions for continued investigation. Overall, the findings underscore the importance of adopting adaptive, identity-centric security strategies in protecting critical banking APIs from increasingly sophisticated cyber threats.

4. CONCLUSION

The increasing reliance on APIs in digital banking has introduced complex security challenges that demand a more rigorous and adaptive protection model. This study proposed a conceptual Zero Trust-based architecture for securing API Gateways in digital banking environments. The architecture integrates key Zero Trust components such as identity verification, context-aware policy decision points, encryption, real-time monitoring, and behavioural analytics to enforce strict access control and ensure continuous security assurance. Through literature-driven analysis and standards-based evaluation, the proposed model addresses core cybersecurity threats including unauthorized access, API abuse, and data breaches. The architecture is aligned with established frameworks such as NIST SP 800-207, OWASP API Security Top 10, and PCI DSS, ensuring its relevance for both technical and regulatory stakeholders. This research contributes to the body of knowledge by filling a gap in existing literature specifically the lack of architectural models that apply Zero Trust principles directly to API Gateways within the banking sector. While the study is conceptual in nature and does not include implementation or performance testing, it offers a foundational blueprint that can guide future research and real-world deployment efforts.

Future work may explore practical implementation scenarios, system performance under varying traffic loads, and advanced threat detection techniques using machine learning. Additionally, further studies can assess the human and organizational aspects of adopting Zero Trust, including change management, training, and policy governance. A well-designed Zero Trust API Gateway architecture can significantly enhance the cybersecurity posture of digital banking systems while supporting compliance, resilience, and operational agility. Banks and financial institutions are encouraged to actively adopt or pilot Zero Trust approaches for API security, both to strengthen their defenses against evolving threats and to align with emerging industry and regulatory expectations.

REFERENCES

[1] D. Dinçkol, P. Ozcan, and M. Zachariadis, "Regulatory standards and consequences for industry architecture: The case of UK Open Banking," *Res. Policy*, vol. 52, no. 6, p. 104760, 2023, doi: 10.1016/j.respol.2023.104760.

[2] P. Hanafizadeh and M. G. Amin, *The transformative potential of banking service domains with the emergence of FinTechs*, vol. 28, no. 3. Palgrave Macmillan UK, 2023. doi: 10.1057/s41264-022-00161-0.

[3] D. Cota, J. Martins, H. Mamede, and F. Branco, "BHiveSense: An integrated information system architecture for sustainable remote monitoring and management of apiaries based on IoT and microservices," *J. Open Innov. Technol. Mark. Complex.*, vol. 9, no. 3, 2023, doi: 10.1016/j.joitmc.2023.100110.

[4] B. J. Hutagaol, R. S. Sitorus, and N. Hutagaol, "Identifikasi tingkat kesadaran pengguna mobile banking terhadap ancaman cybercrime," *J. Teknol. Sist. Inf. dan Apl.*, vol. 7, no. 3, pp. 1043–1054, 2024, doi: 10.32493/jtsi.v7i3.41639.

[5] N. Subramanian and A. Jeyaraj, "Recent security challenges in cloud computing," *Comput. Electr. Eng.*, vol. 71, no. July 2017, pp. 28–42, 2018, doi: 10.1016/j.compeleceng.2018.06.006.

[6] R. S. Sitorus et al., "Capability-based API gateway technology selection analysis for banking cybersecurity solution using AHP method," *Sinkron: Jurnal dan Penelitian Teknik Informatika*, vol. 9, no. 1, pp. 338–347, 2025.

[7] H. Joshi, "Emerging technologies driving zero trust maturity across industries," *IEEE Open J. Comput. Soc.*, vol. 6, no. January, pp. 25–36, 2025, doi: 10.1109/OJCS.2024.3505056.

[8] T. Fadziso, "Evolution of the cyber security threat: An overview of the scale of cyber threat," *Digitalization & Sustainability Review*, vol. 3, no. 1, Sept. 2023, doi: 10.6084/m9.figshare.24189921.v1.

[9] H. Yerramsetty, "Zero Trust Architecture in cloud computing: A paradigm shift in platform engineering security," *World J. Adv. Res. Rev.*, vol. 6, no. 6, pp. 1–9, 2024.

[10] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture NIST Special Publication 800-207," *NIST*, 2020, doi: 10.6028/NIST.SP:800-207.

[11] Y. Kusnanto, M. A. Nugroho, and R. Kartadie, "Implementasi Zero Trust Architecture untuk meningkatkan keamanan jaringan: Pendekatan," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 4, pp. 2357–2364, 2024.

[12] C. Sample, C. Shelton, S. M. Loo, C. Justice, and L. Hornung, "ZTA: Never trust, always verify," in *European Conf. Cyber*, pp. 256–262, 2021.

[13] P. V. Bhat, S. Hg, M. Sujith, C. Ca, and B. Suhas, "Zero Trust Architecture (ZTA)," *NIST special publication 800*, no. 3, pp. 123–130, 2024.

[14] B. F. Rodrigues, "Zero Trust Applied to Digital Banking Platforms," *IET Blockchain*, no. June, 2025.

[15] A. K. Bayya, "Cutting-edge practices for securing APIs in FinTech: Implementing adaptive security models and Zero Trust Architecture," *Int. J. Appl. Eng. Technol.*, no. January, 2025.

[16] R. Chandramouli and Z. Butcher, *Guidelines for API Protection for Cloud-Native Systems*, no. NIST Special Publication (SP) 800-228 (Draft), National Institute of Standards and Technology, 2025.

[17] H. Omotunde and M. Ahmed, "A comprehensive review of security measures in database systems: Assessing authentication, access control, and beyond," *Mesopotamian J. CyberSecurity*, vol. 2023, pp. 115–133, 2023, doi: 10.58496/mjcsc/2023/016.

[18] E. Barker and A. Roginsky, *Transitioning the use of cryptographic algorithms and key lengths*, no. NIST Special Publication (SP) 800-131A Rev. 2 (Draft), National Institute of Standards and Technology, 2018.

[19] H. Omotunde and M. Ahmed, "A comprehensive review of security measures in database systems: Assessing authentication, access control, and beyond," *Mesopotamian J. CyberSecurity*, vol. 2023, pp. 115–133, 2023, doi: 10.58496/mjcsc/2023/016.

[20] Z. Wu, E. Feng, and Z. Zhang, "Temporal-contextual behavioral analytics for proactive cloud security threat detection," *Academia Nexus J.*, vol. 3, no. 2, 2024.