Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

A Low-Latency Zero Trust Framework for Cloud-Based FinTech Systems Using Risk-Based IAM and Selective MFA

Nishant Gaur
nishantgaur5959@gmail.com
Bachelor of Technology (Information Technology)
Noida Institute of Engineering and Technology, Greater Noida
Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India

Supervisor: Ms. Neetu Kumari Rajput
Assistant Professor
Department of Information Technology
Noida Institute of Engineering and Technology, Greater Noida
Affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India

***Abstract*—**Cloud-based FinTech systems need strong security mechanisms to protect confidential financial operations at the same time it has to maintain low-latency transaction processing. Zero Trust Architecture (ZTA) has emerged as an effective cloud security model based on continuous authentication, authorization, and least-privilege access control to users. However, ordinary Zero Trust implementations sometimes introduce authentication latency and computational overhead due to repeated token validation, mandatory Multi-Factor Authentication (MFA), and continuous access verification.

This paper showcases a low-latency framework based on Zero Trust architecture for cloud-based financial systems using risk-based Identity and Access Management (IAM) and selective Multi-Factor Authentication (MFA). The approach dynamically classifies user requests into different risk categories and applies MFA only to high-risk operations such as money transfer and password alteration and low-risk requests are handled using lightweight token-based authentication.

A prototype implementation was developed using Spring Boot, JWT, MySQL, and Role-Based Access Control (RBAC). Experimental evaluation was conducted by comparing the model with a standard Zero Trust model using authentication latency, computational overhead, and security performance as evaluation metrics.

Prototype results showed that the suggested model reduced response delay, decreased unnecessary MFA prompts, reduced computational overhead, and improved system overall performance while maintaining equivalent security effectiveness.

The finding recommends that adaptive authentication strategies based on risk-aware IAM can highly optimize the security-performance trade-off in cloud-based FinTech systems.

Keywords*—Zero *Trust Architecture, Cloud Computing, Cloud Security, FinTech, Identity and Access Management, Multi-Factor Authentication, Risk-Based Authentication, Authentication Latency.

1. INTRODUCTION

Cloud computing has become a foundational technology for modern FinTech systems by enabling scalable, cost-effective, and highly available financial services with on-demand service. Cloud-based financial platforms support services such as digital payments, mobile banking, lending and real-time transaction processing, making security and availability critical requirements for these systems [2], [10]. However, the demanding adoption of cloud environments has also increased the chances of the attack surface for cyber threats including credential theft, insider attacks, session hijacking, and unauthorized access.

Traditional perimeter-based security models are increasingly insufficient in distributed cloud environments, where users, devices, and services operate across multiple locations and networks. To address these challenges, Zero Trust Architecture (ZTA) has emerged as a modern security paradigm based on the principle of "never trust, always verify", requiring continuous authentication, authorization, and least-privilege access enforcement [3], [13].

Although Zero Trust significantly improves cloud security, its practical implementation introduces performance trade-offs. Continuous verification, repeated token validation, policy enforcement, and mandatory Multi-Factor Authentication (MFA) can increase authentication latency, computational overhead, and resource consumption, particularly in latency-sensitive systems such as FinTech applications [5], [8]. Prior studies have reported measurable overhead in Zero Trust systems, including increased latency and reduced throughput when compared with traditional systems, highlighting the need for optimization strategies in real-time environments [8], [9]. Experimental findings from prior work show Zero Trust-based FinTech frameworks can improve security but at the cost of higher processing time and reduced throughput, motivating further research into lightweight and efficient implementations.

This research suggests a low-latency Zero Trust framework for cloud-based financial platforms using risk-based Identity and Access Management (IAM) and selective MFA. Instead of enforcing MFA for all operations, the model applies additional authentication only to high-risk activities such as money transfers, while allowing lower-risk operations to proceed with lightweight verification. The objective is to reduce authentication overhead and latency while maintaining strong security guarantees in cloud-based financial environments which works on low computational cost. This helps in reducing cost of operations of the application while maintaining zero trust security.

2. LITERATURE REVIEW

Zero Trust Architecture (ZTA) has emerged as a latest security model for securing distributed cloud environments.

It helps in eliminating implicit trust and enforcing continuous verification of users, devices, and services. whereas traditional perimeter-based models, Zero Trust follows the principle of "never trust, always verify", requiring authentication and authorization for every access request regardless of network location [13].

Recent research has shown that Zero Trust significantly strengthens cloud security by reducing insider threats, unauthorized access, and lateral movement attacks. Identity and Access Management (IAM), least-privilege enforcement, and continuous monitoring are considered foundational components of Zero Trust implementations in cloud environments [3], [5].

Identity and Access Management (IAM) plays a critical role in Zero Trust systems by controlling user authentication, authorization, and access privileges. IAM frameworks help ensure that only authenticated and authorized users can access cloud resources based on predefined security policies [3].

Several studies have integrated IAM into Zero Trust cloud frameworks to strengthen access control through token validation, role-based authorization, and policy enforcement mechanisms [5]. However, these mechanisms introduce additional verification steps that may impact system performance.

Multi-Factor Authentication (MFA) is widely adopted in cloud security systems to enhance identity verification by requiring multiple authentication factors. In Zero Trust systems, MFA is often enforced as an essential security layer for protecting sensitive operations [5], [9].

Although MFA improves security, repeated authentication prompts can increase user friction, authentication time, and system overhead. In real-time environments such as FinTech systems, enforcing MFA for all actions may negatively impact user experience and transaction efficiency [8].

Despite strong security benefits, Zero Trust implementations introduce performance overhead due to continuous verification, token validation, session checks, and policy enforcement. Prior experimental studies report measurable increases in authentication time and reduced throughput in Zero Trust-enabled cloud systems when compared with traditional security models

These performance trade-offs become more critical in latency-sensitive domains such as digital payments and cloud-based financial services, where rapid response times are required.

Existing studies primarily focus on improving security through Zero Trust principles, IAM integration, and strong authentication mechanisms. However, limited research addresses optimization of authentication overhead in real-time FinTech environments, where both security and fast response times are critical.

Most existing models apply security mechanisms uniformly across all operations, resulting in unnecessary authentication for low-risk actions. This creates latency and computational inefficiencies in cloud-based financial systems [8], [9].

Therefore, a need exists for lightweight Zero Trust frameworks that incorporate **risk-based IAM** and **selective MFA** to reduce latency while maintaining strong security and helps to use computational power efficiently.

3. Problem Statement and Objective

3.1. Problem Statement

Digital finance models require both strong security and low-latency transaction processing. Zero Trust Architecture has emerged as an effective security model for cloud environments by enforcing continuous authentication, authorization, and least-privilege access control [3], [13]. However, existing Zero Trust implementations often introduce additional latency and computational overhead due to repeated token validation, continuous access checks, and mandatory Multi-Factor Authentication (MFA) for all user operations [5], [8].

In real-time FinTech applications such as digital payments and transaction processing, excessive authentication overhead can negatively affect user experience and system efficiency. Prior studies have demonstrated that Zero Trust improves security but may reduce throughput and increase authentication delay in practical deployments

Therefore, there is a need for an optimized Zero Trust framework that minimizes authentication overhead while preserving strong security in cloud-based FinTech systems.

3.2. Objective of Study

The main objectives of this research are:

1. To analyse the latency and computational overhead introduced by traditional Zero Trust implementations in cloud-based financial models.
2. To design a lightweight Zero Trust framework using risk-based Identity and Access Management (IAM).
3. To implement selective Multi-Factor Authentication (MFA) for high-risk operations such as money transfers while reducing unnecessary authentication for low-risk actions.
4. To evaluate the model in terms of:
   - authentication latency,
   - security effectiveness,
   - computational overhead.
5. To compare the model with standard Zero Trust implementations and analyse performance improvements.

4. Methodology

4.1. Framework Overview

This research suggests a low-latency Zero Trust framework for cloud-based FinTech platforms designed to reduce authentication overhead while maintaining strong security. The framework integrates Zero Trust principles with risk-based Identity and Access Management (IAM) and selective Multi-Factor Authentication (MFA).

Unlike conventional Zero Trust models that apply identical authentication requirements to all operations, this model dynamically adjusts authentication intensity based on the risk level of user actions.

4.2. Risk-Based IAM Model

The system introduces a risk evaluation layer within the IAM system. User requests are classified into different risk categories based on the nature of the requested operation.

Risk classification includes:

- **Low-risk operations:**
  - View balance

- Check transaction history
- Profile access

- **High-risk operations:**
  - Money transfer
  - Beneficiary addition
  - Password change
  - Account modification

For each request, the IAM module evaluates:

- user identity,
- access role,
- requested resource,
- action sensitivity.

Access decisions are then enforced using role-based authorization policies.

4.3. Selective Multi-Factor Authentication Strategy

To minimize authentication overhead, Multi-Factor Authentication is not applied uniformly.
The proposed model enforces MFA only for high-risk actions. Low-risk operations are processed using standard token-based authentication mechanisms.
Authentication logic:

1. User submits login credentials.
2. System validates credentials and generates JWT token.
3. User requests an operation.
4. Risk engine evaluates request sensitivity.
5. If request is high-risk, MFA verification is triggered.
6. Upon successful verification, access is granted.

This selective strategy decreases unnecessary authentication prompts and helps in preserving security for sensitive operations.

4.4. Authentication Workflow

The authentication workflow of the system is as follows:

1. User authentication through username and password.
2. JWT token generation after successful login.
3. Request interception by IAM module.
4. Risk analysis of requested operation.
5. Conditional MFA for high-risk requests.
6. Authorization and resource access.

This workflow ensures continuous verification aligned with Zero Trust principles while optimizing authentication efficiency.

4.5. Authorization Mechanism

Authorization is enforced using **Role-Based Access Control (RBAC)** integrated with IAM.
User roles include:

- User
- Admin

Permissions are assigned based on least-privilege principles.
Examples:

- Users can view account information.
- Only authorized users can perform fund transfers or administrative operations.

This mechanism minimizes unauthorized privilege escalation and strengthens access control in cloud environments.

5. System Design And Architecture

5.1. Architecture Overview

The suggested framework is designed as a layered security architecture for financial cloud platforms. It integrates Zero Trust security principles with risk-based IAM and selective MFA to optimize authentication efficiency while maintaining strong protection for sensitive operations.
The system consists of six primary modules:

1. User Interface Layer
2. Authentication Server
3. Identity and Access Management (IAM) Engine
4. Risk Analysis Module
5. MFA Verification Module
6. Cloud Resource/API Layer

The architecture follows a request validation flow where every access request is authenticated, analyzed, and authorized before resource access is granted.

5.2. System Components

5.2.1. User Interface Layer

The User Interface Layer represents the client interacting with the FinTech system.
Users perform actions such as:

- login,
- balance inquiry,
- transaction requests,
- account management.

All requests are forwarded to the authentication server for verification.

5.2.2. Authentication Server

- The Authentication Server is responsible for:
- validating user credentials,
- generating JWT tokens after successful login,
- managing user sessions.

After authentication, a token is issued and attached to subsequent requests.

5.2.3. Identity and Access Management Engine

The IAM Engine manages:

- identity verification,
- role validation,
- authorization policies.

The system uses Role-Based Access Control (RBAC) to enforce least-privilege access.
Example:

- USER → view balance
- ADMIN → privileged operations

5.2.4. Risk Analysis Module

The Risk Analysis Module evaluates each incoming request and assigns a risk level.
Evaluation factors include:

- requested operation type,
- user role,
- resource sensitivity.

Risk categories:

- **Low Risk:** view-only operations
- **High Risk:** financial transactions and account changes

This module is central to reducing authentication overhead.

5.2.5. MFA Verification Module

The MFA module is activated only for high-risk requests.
Functions:

- OTP generation
- OTP verification
- additional access validation

By applying MFA selectively, the framework reduces unnecessary verification delays for low-risk tasks.

5.2.6. Cloud Resource/API Layer

This layer contains protected resources such as:

- account APIs,
- transaction APIs,
- user data services.

Examples:

- /viewBalance
- /transferMoney

Access is granted only after successful authentication and authorization checks.

5.3. System Workflow

The workflow of the proposed architecture is summarized below:

1. User logs in using credentials.
2. Authentication server validates credentials.
3. JWT token is generated.
4. User submits API request.
5. IAM validates token and role.
6. Risk module classifies request.
7. High-risk request triggers MFA.
8. Access granted after successful verification.

This workflow ensures:

- continuous verification,
- least privilege enforcement,
- optimized authentication overhead.

5.4. Security Features of Proposed Architecture

This design provides the following security mechanisms:

- Zero Trust authentication
- JWT-based session management
- Role-based authorization
- Conditional MFA
- Risk-aware access decisions

These collectively improve security while reducing latency in real-time cloud FinTech systems.

6. Implementation Details

6.1. Development Environment

The proposed framework was implemented as a prototype cloud-based FinTech security system using Java-based technologies. The implementation was designed to simulate a lightweight Zero Trust environment for evaluating processing time and access control overhead.

Table-I: Tools and Technologies

Component	Technology Used
Programming Language	Java 17
Framework	Spring Boot
Security	Spring Security + JWT
Database	MySQL
API Testing	Postman
Build Tool	Maven
IDE	IntelliJ IDEA / Eclipse

6.2. Prototype Modules

The prototype consists of the following modules:

- User Authentication Module
- JWT Token Generation Module
- IAM Authorization Engine
- Risk Classification Engine
- Selective MFA Engine
- Protected API Services

6.3. API Endpoints

The following APIs were implemented for prototype evaluation:

Table II: API Endpoints with their Purpose

API Endpoint	Purpose	Risk Level
/login	User authentication	Low
/viewBalance	Account balance inquiry	Low
/transactionHistory	View transaction records	Low
/transferMoney	Money transfer	High
/changePassword	Password modification	High

6.4. JWT-Based Authentication

After successful credential validation, the authentication server generates a JSON Web Token (JWT) containing:

- user ID
- role
- session validity
- token expiration

JWT is attached to all subsequent requests for authorization. This reduces repeated credential submission and enables token-based verification aligned with Zero Trust principles.

6.5. Risk Classification Logic

Each API request is evaluated using a rule-based risk engine. Risk logic:

- Low-risk operations:
  - balance inquiry
  - transaction history
- High-risk operations:
  - money transfer
  - password change
  - account modification

For high-risk requests:

- MFA is triggered.

For low-risk requests:

- token validation + IAM authorization only.

6.6. Selective MFA Mechanism

The prototype implements OTP-based MFA for sensitive operations.

Selective MFA workflow:

1. User initiates high-risk request.
2. System generates OTP.
3. User submits OTP for verification.
4. Access granted upon successful validation.

This avoids unnecessary MFA prompts for low-risk actions.

6.7. Authorization Policy

Authorization is enforced using Role-Based Access Control (RBAC).

Roles:

- USER
- ADMIN

Example policies:

- USER → view balance, transaction history
- ADMIN → privileged operations and management APIs

This ensures least-privilege access.

6.8. Prototype Objective

The prototype was developed to evaluate:

- authentication latency
- computational overhead
- security effectiveness

under standard Zero Trust and optimized Zero Trust configurations.

7. EXPERIMENTAL SETUP

7.1. *Experimental Environment*

The prototype was evaluated in a controlled local environment to simulate a cloud-based FinTech authentication workflow.

Table III: Configuration Details

Parameter	Configuration
Processor	Intel Core i5 / equivalent
RAM	8 GB
Operating System	Windows 11
Java Version	Java 17
Database	MySQL 8.0
Server	Embedded Tomcat

The experiments were conducted using local REST APIs to simulate cloud resource access and authentication workflows.

7.2. Comparative Models

Two authentication models were evaluated:

7.2.1. *Standard Zero Trust Model (Baseline)*

The baseline system implements conventional Zero Trust mechanisms:

- JWT authentication for all requests
- Role-based authorization
- Mandatory MFA for all sensitive and non-sensitive operations
- Continuous token verification

Characteristics:

- Higher verification frequency
- Uniform security policy

7.2.2. *Suggested Optimized Zero Trust Model*

The proposed model implements:

- JWT authentication
- Risk-based IAM
- Selective MFA only for high-risk requests
- RBAC authorization

Characteristics:

- Reduced authentication checks
- Lower MFA frequency
- Risk-aware access decisions

7.3. Test Scenarios

The following test scenarios were executed:

Table IV: Test Scenarios

Scenario ID	Test Description
T1	Invalid login attempt
T2	Access request without token
T3	Expired token access
T4	Unauthorized role access
T5	Low-risk API request
T6	High-risk API request with MFA

These scenarios validate both security enforcement and performance characteristics.

7.4. Evaluation Metrics

The systems were evaluated using the following metrics:

7.4.1. Authentication Latency

Measures average response time for:

- login request
- low-risk API request
- high-risk API request

Unit:

- milliseconds (ms)

7.4.2. Computational Overhead

Measured through:

- number of authentication checks per request
- MFA invocation frequency

This metric estimates system overhead introduced by security controls.

7.4.3. Security Effectiveness

Measured by successful blocking of:

- invalid credentials
- unauthorized access
- expired sessions
- role violations

7.5. Experimental Procedure

For each test case:

1. Requests were executed 20 times.

2. Average response time was recorded.
3. Security validation outcomes were observed.
4. Results from baseline and proposed systems were compared.

This approach ensured consistency in evaluation.

7.6. Expected Outcome

The experiment is designed to verify that the proposed framework:

- Reduces processing time,
- lowers computational overhead,
- maintains equivalent security performance.

8. Result And Analysis

8.1. Authentication Latency Comparison

Authentication latency was measured for both the baseline Zero Trust model and the proposed optimized framework across three primary operations.

Table V: Authentication Latency Comparison

Operation	Standard Zero Trust (ms)	Proposed Approach (ms)
Login	214	176
View Balance (Low Risk)	158	102
Transaction History (Low Risk)	162	108
Transfer Money (High Risk)	241	223

The approach reduced latency across all operations.
Observations:

- Login latency reduced by approximately 17.8%
- Low-risk API latency reduced by approximately 35%
- High-risk requests showed smaller improvement due to mandatory MFA

This indicates that selective MFA primarily benefits low-risk operations by avoiding unnecessary authentication overhead.

8.2. Computational Overhead Analysis

Authentication overhead was evaluated by measuring the number of security checks and MFA invocations.

Table VI: Computational Overhead Comparison

Metric	Standard Zero Trust	Proposed Approach
Average Auth Checks per Request	4.8	3.1
MFA Invocations per 10 Requests	10	3
Token Validation Frequency	High	Moderate

The proposed model significantly reduced redundant security operations.
Key findings:

- Authentication checks reduced by approximately 35.4%
- MFA prompts reduced by 70%

This demonstrates improved operational efficiency in the optimized framework.

8.3. Security Validation Results

Security effectiveness was evaluated using six test scenarios.

Table VII: Security Validation

Test Scenario	Baseline	Proposed Approach
Invalid Login	Blocked	Blocked
No Token Access	Blocked	Blocked
Expired Token	Blocked	Blocked
Unauthorized Role Access	Blocked	Blocked
Low-Risk Request	Allowed	Allowed
High-Risk Request Without MFA	Blocked	Blocked

Both systems successfully enforced access control policies.
Result:

- No reduction in security effectiveness was observed in the proposed model.

8.4. Throughput Analysis

System throughput was estimated based on average request handling performance.

Table VIII: Throughput Comparison

Metric	Standard Zero Trust	Proposed Approach
Requests/sec	31.4	42.7

The optimized framework demonstrated improved throughput due to reduced authentication overhead.
Throughput improvement:

- approximately 36% increase

These findings align with prior studies reporting Zero Trust performance trade-offs in cloud systems

8.5. Discussion of Findings

The results indicate that the model successfully balances security and performance.
Key observations:

- Selective MFA reduces unnecessary authentication for low-risk requests.
- Risk-based IAM lowers computational overhead.
- Security effectiveness remains consistent with standard Zero Trust.

Although high-risk operations still incur MFA-related latency, the framework significantly improves efficiency for common low-risk user activities.
This makes the proposed model suitable for real-time FinTech systems requiring both strong security and low response times.

9. Discussion

9.1. Performance Improvements

The experimental results demonstrate that the proposed approach improves the operational efficiency of Zero Trust systems in cloud-based FinTech environments. By integrating risk-based IAM and selective MFA, the framework reduced authentication delay across all tested operations.
The most significant performance gains were observed in low-risk operations such as balance inquiry and transaction history access. Since these requests did not require mandatory MFA, authentication overhead was minimized, resulting in lower response times and improved throughput.
Compared to the baseline model, the suggested framework achieved:

- reduced authentication latency,
- fewer security checks per request,
- lower MFA invocation frequency.

These improvements are particularly beneficial in FinTech systems, where rapid response times are critical for user satisfaction and real-time transaction workflows.

9.2. Security Preservation

Although this framework reduces authentication overhead, security effectiveness was maintained.
Security validation results confirmed that the system successfully blocked:

- invalid credentials,
- unauthorized token access,
- expired session tokens,
- role-based access violations.

High-risk operations such as money transfer still required MFA verification, ensuring stronger protection for sensitive actions.
This demonstrates that selective security enforcement can maintain Zero Trust principles without uniformly applying costly authentication mechanisms to all operations.

9.3. Security–Performance Trade-off Optimization

Traditional Zero Trust implementations prioritize security through continuous verification and strict access enforcement. However, these mechanisms often introduce measurable latency and computational overhead, as reported in prior literature
The proposed model addresses this trade-off by:

- applying stronger controls only when required,
- reducing redundant verification for low-risk operations.

This adaptive approach improves efficiency while preserving strong access control.

9.4. Practical Implications for FinTech Systems

Cloud-based FinTech applications require both:

- strong security controls,
- low-latency transaction processing.

The suggested framework is particularly suitable for:

- digital payment platforms,
- mobile banking systems,
- online financial services.

By reducing authentication friction for routine operations while protecting critical actions, the framework improves both system performance and user experience.

9.5. Limitations

This study has several limitations:

- prototype evaluated in a controlled local environment,
- limited user scale,
- rule-based risk engine rather than machine learning-based risk prediction.

Therefore, results may vary in large-scale production cloud deployments.

9.6. Future Scope

Future work may extend this framework through:

- machine learning-based adaptive risk scoring,
- behavioural authentication mechanisms,
- large-scale cloud deployment testing,
- integration with blockchain-based identity management.

These enhancements may further improve security efficiency in next-generation FinTech systems.

10. Conclusion And Future Work

10.1. Conclusion

This research proposed a **low-latency Zero Trust framework for cloud-based FinTech systems** using **risk-based Identity and Access Management (IAM)** and **selective Multi-Factor Authentication (MFA)**.
Standard Zero Trust implementations improve cloud security by enforcing continuous verification, strong authentication, and least-privilege access control. In contrast, existing studies and practical implementations shows that these methods introduce response delay and computational overhead, which is not good for real-time FinTech applications.
To address this issue, the suggested framework introduced:

- risk-aware access evaluation,
- selective MFA for high-risk operations,
- lightweight token-based authentication for low-risk requests.

Experimental evaluation demonstrated that the proposed framework:

- reduced authentication latency,
- lowered computational overhead,
- decreased unnecessary MFA invocations,
- maintained equivalent security effectiveness compared with standard Zero Trust implementations.

The findings suggest that adaptive authentication strategies can highly optimize the security-performance trade-off in cloud-based financial systems.
Therefore, the proposed model provides a practical and efficient approach for implementing Zero Trust security model in latency-sensitive FinTech systems.

10.2. Future Work

Although this approach demonstrated promising results, still several improvements can be explored in future work.
Potential future improvements include:

- machine learning-based risk prediction for dynamic authentication decisions,

- user behaviour analytics for anomaly detection,
- blockchain-based decentralized identity verification,
- large-scale deployment in real cloud environments,
- performance evaluation under high concurrent user loads.

These improvements may further improve both security intelligence and operational efficiency in advanced FinTech security architectures.

11. REFERENCES

[1] J. J. Diaz Rivera, A. Muhammad, and W.-C. Song, "Securing Digital Identity in the Zero Trust Architecture: A Blockchain Approach to Privacy-Focused Multi-Factor Authentication," *IEEE Open Journal of the Communications Society*, vol. 5, May 2024, doi: 10.1109/OJCOMS.2024.3391728.

[2] I. Lee and Y. J. Shin, "Fintech: Ecosystem, business models, investment decisions, and challenges," *Business Horizons*, vol. 61, no. 1, pp. 35–46, 2018.

[3] V. Prajapati, "Role of Identity and Access Management in Zero Trust Architecture for Cloud Security: Challenges and Solutions," *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 5, no. 3, Mar. 2025.

[4] A. M. Mostafa, M. Ezz, M. K. Elbashir, M. Alruily, E. Hamouda, M. Alsarhani, and W. Said, "Strengthening Cloud Security: An Innovative Multi-Factor Multi-Layer Authentication Framework for Cloud User Authentication," *Applied Sciences*, vol. 13, no. 19, p. 10871, 2023, doi: 10.3390/app131910871.

[5] S. Potluri, "A Zero Trust-Based Identity and Access Management Framework for Cross-Cloud Federated Networks," *International Journal of Emerging Research in Engineering and Technology*, vol. 5, no. 2, pp. 28–40, 2024, doi: 10.63282/3050-922X.IJERET-V5I2P104.

[6] S. O. Olabanji, O. O. Olaniyi, C. S. Adigwe, O. J. Okunleye, and T. O. Oladoyinbo, "AI for Identity and Access Management (IAM) in the Cloud: Exploring the Potential of Artificial Intelligence to Improve User Authentication, Authorization, and Access Control within Cloud-Based Systems," *Asian Journal of Research in Computer Science*, vol. 17, no. 3, pp. 38–56, Jan. 2024.

[7] A. Mosayyebzadeh et al., "A Secure Cloud with Minimal Provider Trust," *arXiv preprint arXiv:1907.07627*, Jul. 2019.

[8] S. Rodigari, D. O'Shea, P. McCarthy, M. McCarry, and S. McSweeney, "Performance Analysis of Zero-Trust Multi-Cloud," *arXiv preprint arXiv:2105.02334*, May 2021.

[9] A. Singh, "Blockchain-Enabled Zero Trust Framework for Securing FinTech Ecosystems Against Insider Threats and Cyber Attacks."

[10] H. Kumar, "AI and Machine Learning Integration into Cloud-Based Fintech Platforms," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 8, no. 10, Oct. 2024.

[11] Amazon Web Services, "Multi-Factor Authentication in IAM." [Online]. Available: link

[12] National Cyber Security Centre, "Zero Trust Architecture Design Principles," Jan. 16, 2026. [Online]. Available: link

[13] National Institute of Standards and Technology, "NIST Special Publication 800-207: Zero Trust Architecture," Aug. 2020. [Online]. Available: link

[14] National Institute of Standards and Technology, "NIST SP 800-207A: A Zero Trust Architecture Model for Access Control in Cloud-Native Applications in Multi-Location Environments," Sep. 2023. [Online]. Available: link

[15] National Cyber Security Centre, "Recommended Types of Multi-Factor Authentication," Sep. 26, 2024. [Online]. Available: link

[16] OWASP Foundation, "OWASP API Security Top 10," 2023. [Online]. Available: link

[17] Reserve Bank of India, "Master Direction on Digital Payment Security Controls," Feb. 18, 2021. [Online]. Available: link

[18] European Banking Authority, "Regulatory Technical Standards on Strong Customer Authentication and Secure Communication under PSD2," Apr. 5, 2022. [Online]. Available:link

[19] International Organization for Standardization, "ISO/IEC 27001: Information Security Management Systems — Requirements," 2022. [Online]. Available: link

[20] International Organization for Standardization, "ISO/IEC 27017: Security Techniques for Cloud Services," 2015. [Online]. Available: link

[21] Amazon Web Services, "Security Best Practices in IAM." [Online]. Available: link

[22] Google Cloud, "BeyondCorp Zero Trust Security Model." [Online]. Available: link

[23] Accenture, "How Security Helps Banking Bridge Modernization Gaps," Apr. 6, 2026. [Online]. Available: link

[24] Okta, "Risk-Based Authentication: What You Need to Consider," Sep. 14, 2024. [Online]. Available: link

[25] Verizon, "Data Breach Investigations Report," 2026. [Online]. Available: link