Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Adaptive Zero-Trust Middleware Architecture for Decentralized Cloud Integrations: A Dynamic Policy Enforcement Framework

Srikanth Reddy Jaidi
JNTU HYD (GURU NANAK), India
Corresponding Author: Srikanth Reddy Jaidi, **E-mail**: srikanthreddyjaidi11@gmail.com

| ABSTRACT

Traditional middleware systems employing perimeter-based security models demonstrate inadequate protection capabilities in contemporary cloud-native and hybrid ecosystems. The proliferation of distributed microservices across multiple cloud vendors creates significant challenges for uniform endpoint security, particularly regarding over-permissioned API access and lateral movement vulnerabilities. This work presents a novel Zero-Trust Architecture-enabled middleware framework that dynamically adapts access policies based on real-time contextual factors, including device characteristics, behavioral patterns, and geographic location across multi-cloud integration points. The proposed framework integrates Service Mesh architecture with Policy Decision Points, implementing mutual Transport Layer Security, SPIFFE identifiers, and OAuth 2.1 protocols enhanced by artificial intelligence-driven policy learning mechanisms. The system operates as a pluggable framework compatible with existing API infrastructures while providing comprehensive security coverage for government data hubs, financial sector integrations, and healthcare systems requiring regulatory compliance. Comparative evaluation against conventional API Gateway security patterns reveals substantial improvements in breach risk mitigation within simulated cross-cloud environments. The framework addresses critical security gaps in distributed architectures while maintaining operational efficiency and scalability across diverse enterprise deployment scenarios.

| KEYWORDS

Zero-Trust Architecture, middleware security, multi-cloud integration, dynamic policy enforcement, Service Mesh.

| ARTICLE INFORMATION

ACCEPTED: 15 September 2025 **PUBLISHED:** 17 September 2025 **DOI:** 10.32996/jcsts.2025.4.1.66

1. Introduction and Problem Formulation

1.1 Distributed Microservices and Multi-cloud Infrastructure Challenges

Contemporary enterprise computing has moved decisively away from traditional centralized computing and toward increasingly distributed microservices architectures and multi-cloud deployment modalities. Organizations now deploy services throughout multiple cloud providers to realize improved performance metrics, operational cost savings, and enhanced system resilience, which is achievable through diverse geography and administration of services. These changes have created complex hybrid environments where microservices are distributed across multiple cloud systems, each with its own security policies and procedures [1]. The shift from monolithic applications with fixed security perimeters to distributed systems with many interconnected service components has fundamentally changed the ways that security is conceptualized and applied. Individual microservice components are separate threat vectors, while the container orchestration platforms offer highly dynamic attack surfaces that traditional security tools are unable to sufficiently monitor and defend against.

1.2 Perimeter-based Security Model Deficiencies in Middleware Infrastructure

Contemporary middleware systems rely heavily on perimeter-focused security designs that assume internal network trustworthiness while emphasizing boundary protection between internal and external network zones. These established approaches utilize static security policies based on network location and predefined user roles, creating rigid access

management systems that fail to accommodate the flexible nature of cloud-native applications. Perimeter-based security demonstrates fundamental weaknesses through its binary trust model, where entities receive trust classification solely based on network position without evaluating contextual factors such as device health status, user behavior analytics, geographical coordinates, or time-based access patterns [2]. Traditional API gateway solutions and service proxy technologies implement wide-ranging access permissions that commonly produce excessive privilege scenarios, where services acquire authorization levels surpassing their functional necessities.

1.3 Dynamic Policy Enforcement Deficiencies in API Access Management

The fundamental issue involves significant shortcomings in adaptive, context-sensitive policy enforcement within current API access management systems. Present middleware frameworks lack capabilities for continuous trust assessment using real-time contextual information, causing security policies to remain fixed despite changing risk environments. These constraints manifest across several crucial areas: limited precision in access control determinations, inadequate responsiveness to behavioral irregularities, absence of coordination between security policy implementation and operational runtime context, and insufficient integration of zero-trust principles within distributed computing architectures. Excessive permission assignments enable lateral movement exploits, allowing compromised services to reach system resources outside their authorized operational boundaries.

1.4 Project Goals and Implementation Boundaries

The fundamental aim involves constructing and testing a complete Zero-Trust Architecture-integrated middleware platform that resolves identified security weaknesses through adaptive, context-sensitive policy enforcement systems. The core objective includes creating a modular platform that integrates smoothly with current API infrastructure while providing advanced security functions for multi-cloud deployment environments. Implementation boundaries encompass developing responsive security components that leverage real-time contextual data for precise access control determinations, incorporating machine learning-based policy adaptation features for responding to changing threat landscapes, and confirming platform effectiveness across various enterprise environments, including government data centers, financial system integrations, and healthcare platforms requiring strict regulatory adherence.

1.5 Industry Security Incident Context

Current assessments within the industry plainly underline the important need to address current security vulnerabilities, with cloud environments continuing to have large breach events due to improper access permissions configurations and poor API protections. The high rate of API-centric security incidents and events within cloud infrastructure demonstrates the intrinsic need for adaptable security platforms that quickly adapt to the fluid nature of modern distributed environments while also being performant and compliant with various compliance regulations. Organizations face a significant challenge in keeping consistent security standards when services utilize multiple cloud vendors with different security frameworks, different policy specifications, and this often results in excessive shifts of security from either point of failure, creating risks or overly mitigating policies, keeping the business from operating.

2. Literature Review and Theoretical Framework

2.1 Middleware Security Architecture Development

Middleware security frameworks have experienced substantial changes to accommodate the evolving demands of distributed computing systems. Early middleware implementations utilized centralized security approaches that depended extensively on network boundary protections and fixed authentication protocols. These initial designs operated under the assumption that internal network communications were naturally secure, resulting in security strategies concentrated mainly on external threat prevention rather than internal threat identification and response [3]. The progression toward distributed computing models, especially with the introduction of service-oriented designs and microservices, required fundamental modifications in middleware security implementation approaches. Contemporary middleware frameworks now integrate distributed security components capable of functioning across diverse environments while preserving uniform security standards independent of underlying infrastructure differences.

2.2 Zero-Trust Principles and Cloud Implementation Strategies

Clearly, Zero-Trust frameworks are different than other security paradigms, as they remove assumptions of trust and require continuous affirmations by any entity that wants to access system resources, and they all use the idea that no entity should be trusted by default and must go through verification and authorization procedures regardless of their position in an internal or external network [4]. However, cloud infrastructures provide challenges that require continuous monitoring of user activity, device integrity, network traffic behavior, and application collaboration for access decisions since cloud environments are often fluid, distributed, and collaborative under a shared responsibility model. All clouds are multi-tenancy services, which means that zero-trust deployment requires multiple security control layers when used in cloud contexts, such as user identity verification, device check, isolating the network, and controlling user access at the application layer.

2.3 Service Mesh Security Implementation and Policy Management Components

Service mesh frameworks provide additional security capabilities via sidecar proxies that intercept and manipulate all network traffic between microservices. Service meshes provide security at the different layers of the service-to-service communication layer, giving more fine-grained control over traffic patterns, authentication methods, authorization methods, and encryption methods, without changing any application code. Policy Decision Points in service mesh architectures are centralized points that can evaluate access requests against security rules and contextual information. Policy Decision Points can implement rules dynamically with service mesh frameworks by using current threat intelligence, insights into user behavior, and analyzing environmental factors. Service mesh security implementations also offer transport layer security between workloads, traffic encryption, and identity-based routing to protect the flow of communication between services.

2.4 API Gateway Security Model Evaluation

Across all vendor offerings and open-source alternatives, the security deployments of today's API gateways utilize different approaches to access management, authentication protocols, and policy implementation. For traditional API gateway frameworks, a majority have relied primarily on traditional features to involve rate limiting, resource checks, or simple checks for authentication tokens and authorization—all without advanced contextual assessments or behavioral analyses. The majority of current solutions use static security models that rely on manual reconfiguration or updates to address a shifting security landscape or a changing environment. More modern API gateway designs have begun to use more advanced capabilities to detect threats, analyse anomalies, and connect with external security intelligence services. However, sizeable gaps remain in areas such as dynamic policy updates, context-aware decision making, and integration with zero-trust design principles.

Security Model	Trust Assumption	Policy Type	Context Awareness	Threat Detection	Multi-cloud Support
Perimeter-based	Network location	Static rules	Limited	Reactive	Poor
API Gateway	Token validation	Pre-configured	Minimal	Rule-based	Moderate
Software-defined Perimeter	Identity verification	Dynamic	Moderate	Proactive	Good
Zero-trust Middleware	Continuous verification	Adaptive	Comprehensive	AI-driven	Excellent

Table 1: Comparison of Security Models in Middleware Architectures [2, 3, 4]

2.5 Dynamic Policy Enforcement Research Deficiencies

The existing academic literature reveals significant limitations with any dynamic policy enforcement mechanisms that can make real-time adjustments in the face of new threats or operational contexts. Existing solutions still rely upon traditional static, rule-based mechanisms that require human changes for implementation of the existing policy. Artificial intelligence and machine learning methods for policy adaptation remain at a pilot or academic research level and are still subject to extended training before being applicable for operational use. Academic work is scarce on context-sensitive policy enforcement, as these systems do not leverage the complete context of user behavior analytics, device characteristics, geographical elements, and temporal access patterns into decisions for access control. To date, academic literature is scarce on enriched policy interoperability across multi-cloud and heterogeneous infrastructure platforms.

3. Proposed Zero-Trust Middleware Architecture

3.1 System Design and Core Components

The zero-trust middleware framework establishes a multi-layered security infrastructure that coordinates various protection mechanisms to deliver comprehensive policy enforcement capabilities across distributed cloud platforms. The foundational design incorporates interconnected security modules, including enforcement gateways, evaluation processors, contextual analyzers, and encrypted communication pathways that collectively ensure complete API interaction coverage [5]. Modular component architecture allows independent deployment and customization according to organizational specifications and current infrastructure limitations. Essential system elements encompass identity validation processors, behavioral monitoring engines, location-aware evaluation modules, device status verification components, and policy coordination services that collaborate to maintain continuous security evaluation functions. The framework maintains clear functional boundaries between

policy creation, assessment procedures, and implementation mechanisms to guarantee scalability and operational sustainability across varied deployment environments.

Component	Function	Integration Layer	Security Protocol	Context Input
Policy Enforcement Points	Access control	Service mesh	mTLS, SPIFFE	Device, location
Decision Engines	Policy evaluation	Control plane	OAuth 2.1	Behavior, time
Context Analyzers	Risk assessment	Data plane	Encryption	Geography, user
Identity Validators	Authentication	Application layer	Certificate-based	Device integrity
Policy Orchestrators	Rule coordination	Management plane	API security	All contextual

Table 2: Zero-Trust Middleware Framework Components [5, 6]

3.2 Service Mesh and Zero-Trust Integration

The convergence of service mesh infrastructure with zero-trust security creates a unified protection layer that functions seamlessly across all inter-service communications throughout distributed computing environments. Service mesh components deliver essential infrastructure for zero-trust control implementation, including traffic capture, identity confirmation, and secure connection establishment between distributed services [6]. The framework utilizes service mesh functionalities to implement zero-trust protocols at the network layer while preserving application transparency and performance efficiency. Integration approaches include sidecar proxy setup for policy implementation, service directory integration for identity administration, and control plane coordination for policy propagation and maintenance. The service mesh layer functions as the implementation point for zero-trust protocols while delivering comprehensive monitoring and observation capabilities necessary for ongoing security evaluation and policy enhancement.

3.3 Contextual Policy Adaptation Mechanisms

Contextual policy modification capabilities facilitate immediate security protocol adjustments through comprehensive analysis of device properties, user activity patterns, and geographical positioning elements. The framework continuously gathers and processes contextual data to evaluate threat levels and modify access authorizations automatically without manual oversight or operational interruption. Device-related contextual elements include hardware validation status, software compliance verification, security update levels, and endpoint protection capabilities that collectively determine device reliability measurements. Behavioral monitoring components track user engagement patterns, access timing, resource utilization behaviors, and departures from normal operational baselines to detect potential security threats or account compromises. Geographical processing incorporates location-specific access protocols, regional regulatory requirements, network proximity considerations, and temporal zone factors to maintain appropriate access management based on physical and logical positioning.

3.4 Security Protocol Stack: mTLS, SPIFFE, and OAuth 2.1

Mutual Transport Layer Security protocol deployment ensures encrypted communication pathways and bidirectional verification between all framework components and external service interfaces. SPIFFE identity management integration delivers standardized identity verification and attestation functions across diverse cloud infrastructures, facilitating uniform identity administration independent of underlying platform technologies. OAuth 2.1 implementation creates secure authorization processes that enable precise access management while preserving compatibility with current authentication frameworks and identity service providers. The security protocol infrastructure operates across multiple operational levels to deliver comprehensive protection, including transport encryption, application verification, and session authorization controls. Protocol coordination ensures compatibility across different cloud platforms and service mesh deployments while maintaining uniform security standards and policy implementation capabilities throughout distributed environments.

3.5 Machine Learning Policy Enhancement Framework

The machine learning-based policy enhancement system incorporates algorithmic analysis to automatically recognize security patterns, identify irregularities, and suggest protocol modifications based on historical information and current threat

intelligence. Machine learning algorithms examine access behaviors, security events, and environmental information to create adaptive security protocols that progress with evolving threat conditions and operational needs. The learning infrastructure includes supervised algorithms for recognized threat pattern identification, unsupervised modules for irregularity detection, and reinforcement systems for policy optimization based on security results. Policy enhancement processes operate continuously without disrupting normal system functions to refine security rules, adjust risk evaluation algorithms, and improve access control accuracy. The framework includes transparency capabilities that provide visibility into policy determinations and enable security administrators to understand and validate automated protocol recommendations.

3.6 Modular Framework Integration Specifications

The modular framework architecture facilitates smooth integration with current API infrastructure through standardized connection points and component-based designs that reduce deployment complexity and operational disruption. Framework specifications establish clear integration interfaces, configuration options, and extension capabilities that enable organizations to implement zero-trust features incrementally without complete infrastructure replacement. The modular design supports various deployment approaches, including independent installation, sidecar integration, and gateway-level implementation to accommodate different organizational needs and technical limitations. Integration specifications encompass API compatibility interfaces, configuration management systems, monitoring and logging connection points, and policy synchronization processes that maintain consistent security implementation across hybrid and multi-cloud infrastructures. The framework delivers compatibility features and transition strategies that allow organizations to migrate from current security solutions while preserving operational continuity and protection effectiveness.

4. Implementation and Experimental Design

4.1 Enterprise Testbed Infrastructure Setup

The experimental infrastructure creates a comprehensive cloud-fusion environment that mirrors authentic enterprise operational scenarios across various cloud service providers and hybrid computing configurations. The testbed framework encompasses distributed computing assets, including public cloud resources, private infrastructure installations, and edge computing nodes to replicate genuine enterprise deployment environments [7]. Infrastructure elements include containerized service environments, serverless execution platforms, distributed database systems, and network architectures that reflect standard enterprise multi-cloud implementations. The testing environment enables dynamic resource allocation, automated scaling functions, and resilience capabilities necessary for assessing zero-trust middleware behavior under diverse operational circumstances. Setup specifications include network architecture definitions, security protocol configurations, service mesh implementation patterns, and observation infrastructure that collectively deliver thorough testing capabilities for zero-trust deployment scenarios.

Infrastructure Component	Configuration Type	Cloud Provider	Deployment Model	Security Integration
Containerized Services	Kubernetes clusters	AWS, Azure, GCP	Multi-cloud	Service mesh
API Gateways	Load balancers	Hybrid	Edge deployment	Zero-trust policies
Database Systems	Distributed storage	Private cloud	Replicated	Encrypted channels
Monitoring Systems	Observability tools	SaaS platforms	Centralized	Policy compliance
Network Architecture	SDN implementation	Multi-vendor	Segmented	Identity-based routing

Table 3: Testbed Configuration Parameters [7, 8]

4.2 Multi-cloud Connection Point Establishment

Multi-cloud connection points create secure communication pathways and information exchange protocols between distinct cloud service platforms while preserving uniform security standards and operational procedures. The connection setup encompasses API gateway implementations, service location mechanisms, traffic distribution systems, and inter-cloud networking solutions that facilitate smooth service interactions across platform divisions [8]. Connection configurations include identity federation frameworks, protocol synchronization processes, and security credential administration systems that guarantee consistent security implementation independent of underlying cloud platform variations. The establishment includes observation and logging connection points that deliver visibility into inter-cloud communications, operational metrics, and

security events crucial for assessing zero-trust middleware functionality. Set up specifications to address network delay factors, bandwidth enhancement methods, and backup mechanisms that preserve service accessibility during cloud platform interruptions or scheduled maintenance periods.

4.3 Breach Risk Simulation Framework

Simulation frameworks establish comprehensive threat scenarios and attack methodologies that assess the zero-trust middleware system's capacity to identify, block, and counter various security incidents. The simulation environment includes realistic attack behaviors such as lateral movement efforts, authorization escalation scenarios, information extraction simulations, and internal threat activities that challenge the middleware's protective functions. Framework configurations encompass threat actor characteristics, attack sequence patterns, target selection strategies, and concealment methods that represent current cybersecurity threat environments. The simulation platform enables controlled security incident creation, permitting systematic assessment of middleware reactions to different threat types and severity levels. Evaluation frameworks include detection precision measurements, reaction time assessments, incorrect alert analysis, and system operational impact during security events to deliver comprehensive protection effectiveness evaluations.

4.4 Operational Metrics and Security Assessment Standards

Operational measurement systems create quantitative standards for assessing both functional efficiency and protective effectiveness of the zero-trust middleware deployment across different implementation scenarios. Security assessment metrics encompass threat identification rates, protocol implementation precision, incident reaction times, and regulatory compliance measurements that collectively evaluate the middleware's defensive capabilities. Operational standards include system processing measurements, delay impact evaluation, resource consumption assessments, and expansion capability analysis that determine the middleware's operational feasibility in production environments. The assessment system includes baseline operational measurements from conventional security systems to facilitate comparative evaluation and quantify enhancement levels achieved through zero-trust deployment. Measurement approaches include automated testing protocols, continuous observation systems, and statistical evaluation methods that deliver an objective assessment of middleware functionality under various operational circumstances and security scenarios.

4.5 Regulatory Compliance Verification Procedures

Compliance verification procedures create systematic methods for confirming adherence to regulatory obligations, including healthcare information protection requirements and federal processing standards across multi-cloud implementations. The verification approach includes automated compliance monitoring mechanisms, audit documentation creation systems, and regulatory reporting functions that demonstrate continuous adherence to mandated requirements. Verification processes encompass protocol alignment activities that connect zero-trust middleware configurations with specific regulatory obligations, ensuring comprehensive coverage of required security controls and operational procedures. The approach includes documentation creation systems, evidence gathering mechanisms, and external assessment support capabilities that support regulatory examinations and compliance confirmation processes. Verification systems include continuous observation platforms that monitor compliance status continuously, notifying administrators of potential violations or configuration changes that could affect regulatory adherence across distributed cloud environments.

5. Results and Performance Analysis

5.1 Security Breach Risk Reduction Measurements

The zero-trust middleware platform exhibits substantial enhancements in security breach risk mitigation capabilities compared to conventional perimeter-focused security implementations across diverse threat landscapes and attack methodologies. Comprehensive risk evaluation approaches indicate significant improvements in threat identification functions, incident reaction efficiency, and comprehensive security stance preservation throughout the testing assessment duration [9]. The platform's adaptive policy implementation mechanisms provide enhanced protection against lateral movement exploits, authorization escalation activities, and information extraction scenarios that frequently compromise traditional security frameworks. Risk measurement indicators encompass threat identification precision, incorrect alert minimization rates, incident isolation periods, and security protocol efficiency assessments across varied operational contexts. The assessment reveals consistent risk mitigation trends across different implementation scenarios, demonstrating the platform's reliable security enhancement capabilities independent of underlying infrastructure arrangements or organizational circumstances.

5.2 Operational Benchmarking Versus Static API Gateway Systems

Comparative operational evaluation between the proposed zero-trust middleware and traditional static API gateway solutions indicates significant operational benefits across multiple performance aspects and operational contexts. Benchmarking outcomes show enhanced processing capabilities, minimized delay effects, and optimized resource consumption efficiency when handling API requests through adaptive zero-trust implementation mechanisms. The platform maintains superior operational

characteristics while concurrently delivering advanced security capabilities that conventional API gateways cannot provide without considerable performance costs [10]. Operational measurements include request handling speeds, simultaneous connection management abilities, memory utilization trends, and processor consumption rates under different load scenarios. The benchmarking assessment incorporates stress evaluation scenarios that confirm the platform's capacity to preserve both security effectiveness and operational functionality under high-demand circumstances characteristic of enterprise production environments.

Metric Category	Traditional Gateway	Zero-trust Middleware	Improvement Factor	Measurement Unit
Threat Detection Rate	Baseline	Enhanced	Significant	Percentage
False Positive Rate	Higher	Reduced	Substantial	Per thousand requests
Response Time	Standard	Optimized	Notable	Milliseconds
Policy Adaptation	Manual	Automated	Complete	Configuration changes
Compliance Coverage	Partial	Comprehensive	Full	Regulatory standards

Table 4: Security Performance Metrics Comparison [9, 10]

5.3 Expansion Capability Assessment Across Industry Applications

Expansion capability evaluation across government, financial, and healthcare implementation contexts confirms the zero-trust middleware platform's capacity to support diverse organizational needs and regulatory limitations while preserving uniform security and operational standards. Government sector deployments show the platform's ability to manage large-scale implementations with rigorous security demands and complex multi-department integration scenarios. Financial sector assessments validate the platform's appropriateness for high-volume transaction environments requiring immediate fraud identification, regulatory compliance, and customer information protection across distributed trading and banking infrastructures. Healthcare sector evaluation confirms the platform's effectiveness in securing sensitive patient data while facilitating protected information sharing between healthcare organizations, academic institutions, and regulatory entities. Expansion indicators include simultaneous user handling, transaction processing capabilities, information volume management abilities, and geographical distribution support across different industry-specific implementation configurations.

5.4 Economic Assessment of Implementation Investment

Economic evaluation of zero-trust middleware deployment indicates positive cost-benefit relationships when examining both initial implementation investments and extended operational savings realized through improved security effectiveness and minimized incident management expenses. Implementation investment evaluation includes infrastructure purchase costs, staff education needs, system integration expenditures, and continuous maintenance investments necessary for successful platform deployment. Benefit measurement encompasses decreased security incident expenses, reduced compliance violation costs, enhanced operational efficiency improvements, and strengthened customer confidence metrics that contribute to comprehensive organizational value generation. The evaluation incorporates both direct expense elements, such as equipment and software costs, alongside indirect advantages, including reputation safeguarding, business continuity enhancements, and competitive benefits resulting from improved security capabilities. Economic calculations show positive investment return periods that support the initial implementation expenditures across different organizational sizes and industry environments.

5.5 Regulatory Adherence Verification in Hybrid Cloud Systems

Regulatory adherence verification outcomes validate the zero-trust middleware platform's capability in preserving regulatory compliance across complex hybrid cloud implementations, extending multiple jurisdictions and regulatory structures. Verification processes show consistent compliance preservation for healthcare information protection mandates, financial services regulations, and government security standards, independent of underlying cloud platform diversity or geographical distribution configurations. The platform's automated compliance observation capabilities facilitate continuous regulatory adherence confirmation without manual oversight, decreasing compliance administration burden while enhancing audit preparation and regulatory documentation precision. Compliance verification includes policy implementation consistency, audit documentation

completeness, information protection effectiveness, and regulatory documentation capability across different cloud service platforms and implementation approaches. The verification outcomes demonstrate successful regulatory mandate fulfillment across all evaluated compliance structures, validating the platform's appropriateness for regulated industry implementations requiring strict adherence to multiple intersecting regulatory standards.

6. Conclusion

Adaptive zero-trust middleware architecture provides an effective approach to addressing the security challenges of today's increasingly distributed cloud environment. Through dynamic policy enforcement based on the context of devices, behavior, and location, can assert that adopting zero trust principles will greatly decrease the organization's risk of a breach while maintaining operations. Combining service mesh architecture with zero-trust principles consistently and easily works across government, finance, and healthcare sectors. The economic evaluation shows favorable cost-benefit factors as the organization will see lower incident response costs and increased compliance. The framework is modular, so integrating current APIs into the new architecture can be done seamlessly, allowing the organization flexibility to adopt it at a sustainable and plausible pace to avoid stopping operations. The current design meets the requirements of three regulatory frameworks, demonstrating compliance for regulated industries. Machine learning is utilized throughout the framework, continuously improving security by adapting policies to meet changes in threats. The future work should accommodate a greater AI-facilitated ability to facilitate threat detection and establish interoperability and standards amongst different cloud vendors.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.
Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Abdallah M, et al., (2019) Software-Defined Perimeter (SDP): State of the Art Secure Solution for Modern Networks, IEEE Network Magazine, vol. 33, no. 5, pp. 226-233, Sept.-Oct. 2019. Available: https://www.eng.uwo.ca/oc2/publications/thepublicationpdfs/2019-SDP-IEEE-Network.pdf

[2] Deborah J. B, et al., (2018) Cyber Resiliency Metrics, Measures of Effectiveness, and Scoring, MITRE Corporation, MTR180314, September 2018. Available: https://www.mitre.org/sites/default/files/2021-11/prs-18-2579-cyber-resiliency-metrics-measures-of-effectiveness-and-scoring.pdf

[3] Hamid M F, et al., (2020) Dynamic Multi-objective Scheduling of Microservices in the Cloud, 2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC), Leicester, United Kingdom, December 30, 2020, pp. 1-10. Available: https://ieeexplore.ieee.org/document/9302823

[4] Hammett R. and Ferry M., (2006) A Testbed for the Development, Demonstration and Testing of Information Fusion Systems, 2005 7th International Conference on Information Fusion, IEEE Conference Publication, February 13, 2006. Available: https://ieeexplore.ieee.org/document/1592033

[5] Naeem F S, et al., (2022) Zero Trust Architecture (ZTA): A Comprehensive Survey, IEEE Access, Vol. 10, pp. 6174679, May 12, 2022. Available: https://ieeexplore.ieee.org/document/9773102

[6] Rajesh K, (2022) Quantitative Safety-Security Risk Analysis of Interconnected Cyber-Infrastructures, 2022 IEEE 10th Region 10 Humanitarian Technology Conference (R10-HTC), November 3, 2022. Available: https://ieeexplore.ieee.org/document/9929906

[7] Ramaswamy C, and Zack B, (2023) Zero Trust Architecture Model for Access Control in Cloud-Native Applications in Multi-Location Environments, National Institute of Standards and Technology (NIST), NIST SP 800-207A, September 2023. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207A.pdf

[8] Ravinder R, (2025) Zero Trust Architecture (ZTA) in Cloud-Native Environments: A Technical Deep Dive*, International Research Journal of Modernization in Engineering Technology and Science (IRJMETS),* March 2025. Available: https://www.irjmets.com/uploadedfiles/paper/issue_3_march_2025/70556/final/fin_irjmets1743135566.pdf

[9] Rochak B, et al, (2021) Middleware Architecture – History and Adaptation with IEEE 802.11, in Middleware Architecture, edited by Mehdia Ajana El Khaddar, IntechOpen, May 4, 2021. Available: https://www.intechopen.com/chapters/76136

[10] Saadia D, et al., (2019) Security Risk Assessment of Multi-cloud System Adoption: Review and Open Research Issues,, Springer Nature, February 22, 2019. Available: https://link.springer.com/chapter/10.1007/978-3-030-12048-1_37