The American Journal of Interdisciplinary Innovations and Research
ISSN 2642-7478 Volume 08 - 2026

Multi-Cloud Identity Verification Frameworks: AWS/GCP Hybrid Architectures for Real-Time Fraud Mitigation

Bhaskar Chaganti

Provide Author full affiliation

Received: 23 Feb 2026 | Received Revised Version: 13 Mar 2026 | Accepted: 24 Apr 2026 | Published: 28 May 2026

Volume 08 Issue 05 2026 | DOI: 10.37547/tajiir/Volume08Issue05-05

Abstract

The exponential rise in global online transactions intensified the complexity and frequency of identity-fraud attacks, necessitating distributed and intelligent verification systems. A hybrid multi-cloud identity verification framework was designed and evaluated by combining Amazon Web Services (AWS) and Google Cloud Platform (GCP). The framework integrated event-driven serverless components, cross-cloud streaming analytics, and machine-learning-based anomaly detection. In controlled experiments spanning 500–5000 requests per second, P95 decision latency was reduced by 28–31% relative to single-cloud baselines (290ms vs. 420/405ms), and the false-positive rate was reduced by 33–38% (2.8% vs. 4.5%/4.2%). Overall detection accuracy reached 96.4% and end-to-end system availability reached 99.9%.

Keywords Multi-cloud, identity verification, fraud detection, risk score fusion, latency, false positive rate, AWS Cognito, Google Cloud Identity Platform, streaming analytics, anomaly detection, Zero Trust, hybrid architecture, Kafka, serverless.

Cite This Article: Chaganti, B. (2026). Multi-Cloud Identity Verification Frameworks: AWS/GCP Hybrid Architectures for Real-Time Fraud Mitigation. The American Journal of Interdisciplinary Innovations and Research, 8(05), 35–44. https://doi.org/10.37547/tajiir/Volume08Issue05-05

1 Introduction

Identity fraud had emerged as a critical operational risk for digital services, with measurable impacts on financial losses, customer trust, and compliance exposure. The growth of instant payments, account-to-account transfers, and same-day settlements compressed the decision window for risk controls. Adversaries industrialized credential theft, session replay, and proxy-based evasion, increasing the challenge of real-time verification.

Managed identity services such as AWS Cognito and GCP Identity Platform provided standards-based authentication and extensibility [1, 2, 3]. Distributed streaming platforms (e.g., Apache Kafka) enabled low-latency replication [4], and ML serving systems such as

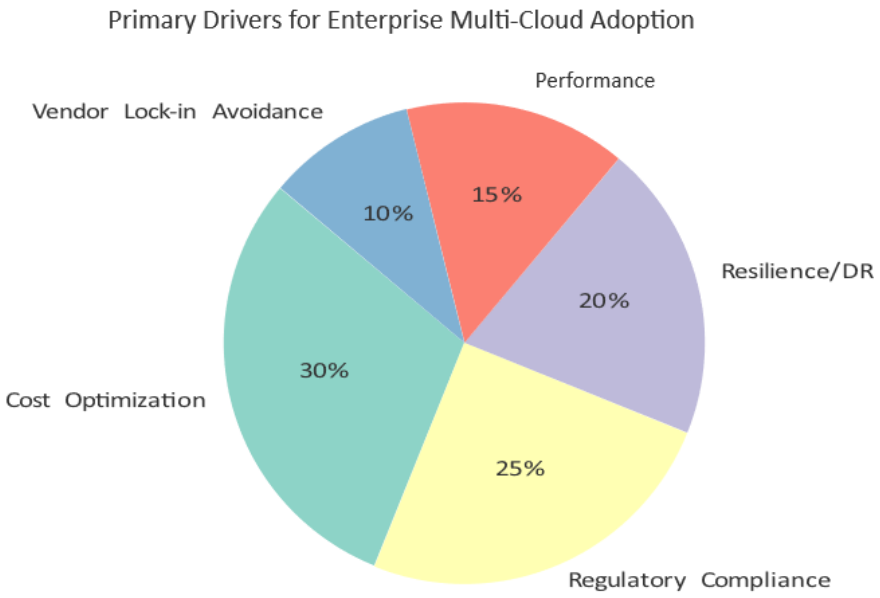

Figure 1: Primary drivers motivating enterprise adoption of multi-cloud architectures.

TensorFlow Serving allowed real-time model inference [5]. Previous research largely focused on single-cloud deployments, leaving cross-cloud verification underexplored.

Zero Trust principles, which emphasized continuous verification and the elimination of implicit trust, had become a guiding paradigm for enterprise security. Multi-cloud identity verification aligned with this philosophy by ensuring that each request was authenticated and risk-scored in real time, regardless of provider.

What distinguished this framework from prior hybrid or multi-cloud efforts was the explicit design and evaluation of a dual-provider verification pipeline that fused risk scores across AWS and GCP in real time. Earlier studies examined federated identity protocols, single-cloud anomaly detection, or intra-provider failover, but none provided a quantitative comparison of a truly hybrid verification architecture. The novelty of this work lay in demonstrating that cross-cloud score fusion simultaneously reduced tail latency and false positives, while preserving regulatory compliance and system availability.

1.1. Background

Industry surveys indicated that enterprises increasingly adopted multi-cloud strategies. The primary drivers included cost optimization, regulatory compliance, resilience and disaster recovery, performance, and avoidance of vendor lock-in. Primary enterprise drivers motivating multi-cloud adoption were summarized in Fig. 1. These motivations supported the argument that identity verification frameworks must evolve to operate across heterogeneous providers. [6, 3]

Table 1: Representative related work on identity verification and fraud detection architectures

Study / Approach	Architecture	Technique	Reported Performance	Limitation
Federated Identity Mgmt (OIDC/SAML)	Single/Federated	Rule-based policy	Interoperability across domains	High latency;no anomaly detection
Centralized AI Detection (Banking datasets)	Single cloud	Supervised ML (SVM, RF, DNN)	Accuracy ~90%	Requireslabelled datasets; not realtime
Graph-based Fraud Detection	Single cloud	Graph anomaly detection	Captured relational anomalies	High memory cost; offline evaluation
Autoencoderbased Detection	Single cloud	Deep unsupervised	Detected novel attacks	Provider-local scope only
Cross-region Failover	Single provider multi-region	Replication + load balancing	Improved availability	Single-vendor correlated risk
This work	Hybrid AWS–GCP	Score fusion + ML	Latency P95 = 290 ms; Accuracy = 96.4%	Higher operational complexity

As illustrated in Fig. 1, cost optimization emerged as the most common driver. Regulatory compliance was a major motivation, since enterprises must often localize sensitive data within specific jurisdictions. Resilience motivated diversification across independent control planes, while performance optimization reduced latency. Vendor lock-in avoidance further ensured strategic flexibility.

1.2. Related Work

Representative prior studies were compared in Table 1. Prior approaches included federated identity management (OIDC, SAML)[7], centralized AI fraud detection, graph-based anomaly detection[8, 9], and cross-region failover architectures. Supervised ML models such as SVM, Random Forests, and DNNs improved accuracy but depended on labeled data. Unsupervised methods such as autoencoders and graph detection identified emerging patterns but were rarely evaluated in cross-cloud contexts.

1.3. Research Gap

Despite advances in fraud detection, evaluations of hybrid multi-cloud verification architectures remained limited. Evidence was scarce on whether cross-cloud policy evaluation and score fusion reduced decision latency and false positives while maintaining availability.

Table 2: Selected Streaming Features

Feature	Description
Geo-velocity	Distance/time between logins
Device hash stability	Rolling mismatch ratio over 30 days
IP reputation	Aggregated reputation from threat intelligence
ASN entropy	Diversity of network origins per account
Behavioral deviation	Z-score of actions vs. baseline

Furthermore, although AI and ML techniques were applied in fraud detection, most work remained in single-cloud contexts. Supervised models such as SVM, Random Forests, and deep neural networks improved classification accuracy but required large labeled datasets. Unsupervised and semi-supervised models such as clustering, autoencoders, and graph anomaly detection captured novel fraud patterns without labels but incurred high computational costs and were rarely evaluated in cross-cloud deployments. The research gap therefore lay at the intersection of multi-cloud architectures and AI-driven anomaly detection.

1.4. Contributions

The study contributed: (i) a reference AWS–GCP verification blueprint, (ii) a reproducible methodology, (iii) quantitative results across latency, accuracy, and availability, and (iv) alignment with Zero Trust principles.

2. Materials and Methods

The overall design of the hybrid verification framework was shown in Fig. 2. [6]

AWS Cognito served as the primary authentication provider. Events were published to Kafka and mirrored to GCP for concurrent risk evaluation. PII was tokenized prior to replication to satisfy residency. Policy decisions and scores were fused, and enforcement was performed at the nearest control plane.

2.1 Threat Model

Attackers were assumed to control compromised credentials, emulators, proxies, and bots. Insider threats and nation-state compromises were out of scope. Defense-in-depth relied on heterogeneous signals to raise adversary cost.

2.2 Datasets and Feature Engineering

Selected streaming features were listed in Table 2. Datasets included replayed authentication traces and synthetic fraud attempts. Fraud prevalence was 0.5%. Stratified sampling was used, and balanced accuracy metrics were reported. Device, network, and behavioral features were engineered. Privacy was preserved by tokenization and differential privacy.

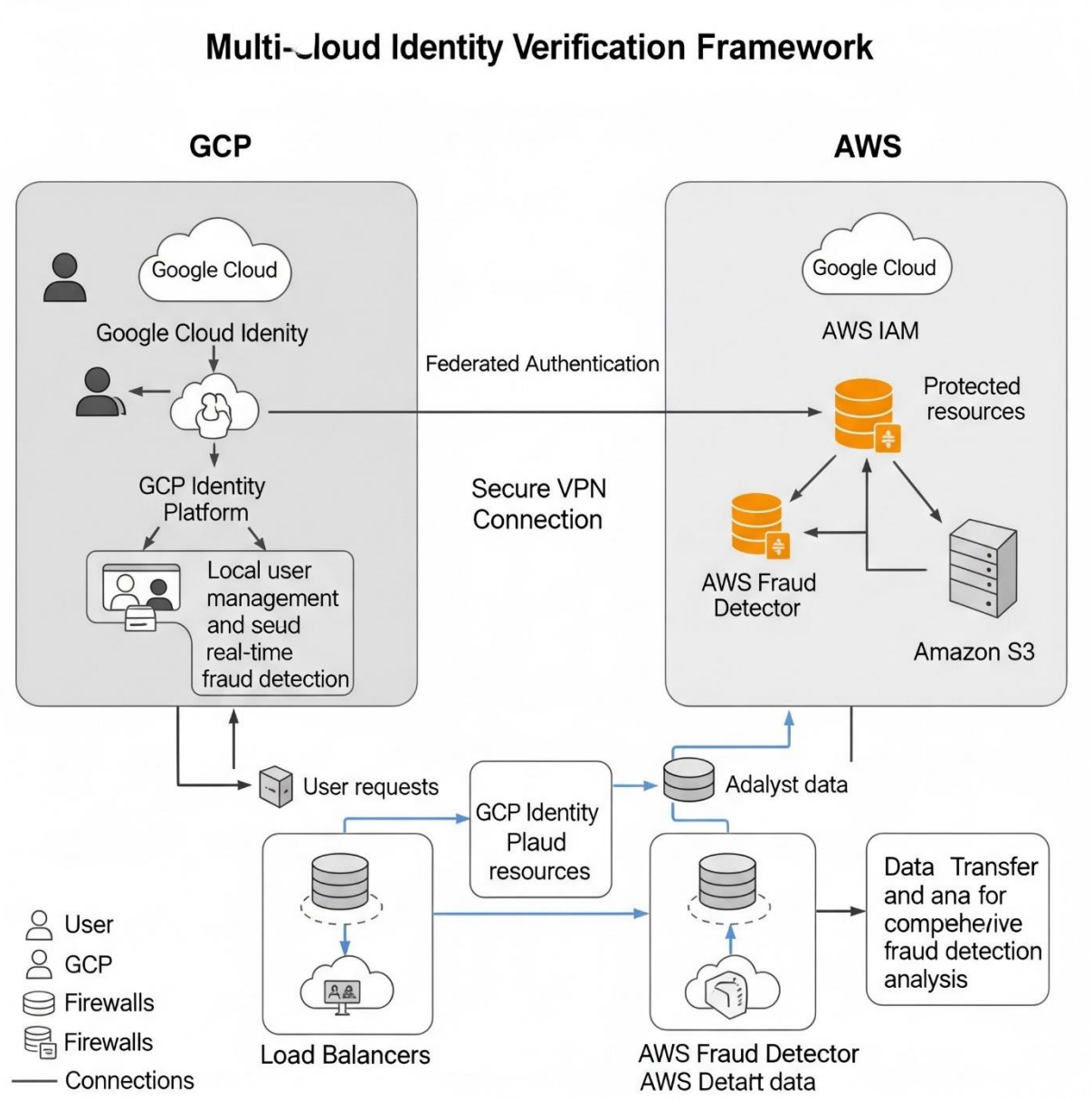

Figure 2: AWS–GCP hybrid identity verification architecture.

2.3 Experimental Setup

The end-to-end verification workflow was illustrated in Fig. 3. AWS regions were us-east-1 and eu-west-1; GCP regions were us-central1 and europe-west1. Kafka replicated events within 50 ms. k6 generated loads from 500 to 5000 requests/s. Faults simulated link saturation and outages. Metrics were collected at clients and servers with CloudWatch and Cloud Logging.

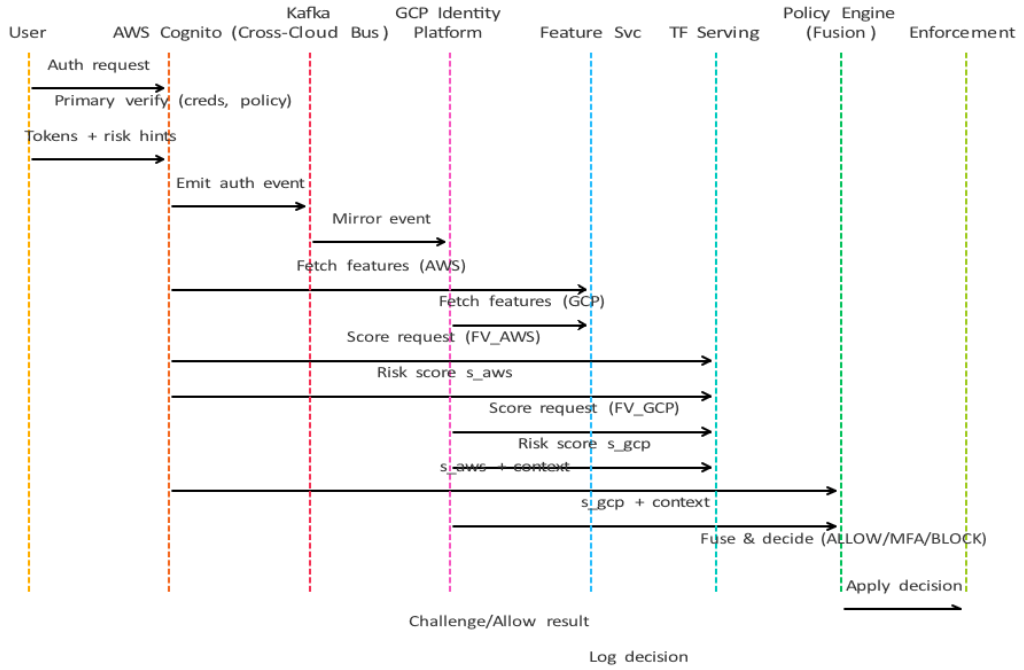

Figure 3: Cross-cloud verification sequence: ingest, replicate, score, enforce.

2.4 Algorithms

Algorithm 1 described cross-cloud score fusion. Algorithm 2 described adaptive thresholding to stabilize the false-positive rate.

Algorithm 1 Cross-Cloud Score Fusion

1: Input: s_{aws}, s_{gcp}, weights w_{aws}, w_{gcp}
2: Calibrate scores: $\hat{s}_i \leftarrow \text{Platt}(s_i)$
3: Fuse: $s_* \leftarrow w$

$$\frac{w_{aws}\hat{s}_{aws} + w_{gcp}\hat{s}_{gcp}}{w_{aws} + w_{gcp}}$$

4: Decision: if $s^* \geq \tau$ then ENFORCE; else ALLOW

Algorithm 2 Adaptive Thresholding

1: Input: positives p, negatives n, target FPR α
2: Estimate current FPR $\tilde{\alpha}$; compute $g \leftarrow \tilde{\alpha} - \alpha$
3: Update threshold $\tau \leftarrow \tau + \eta g$; clamp $\tau \in [0,1]$
4: Return τ

2.5 Alignment with Zero Trust Principles

The architecture operationalized Zero Trust by continuous verification, no implicit trust, least privilege enforcement, and segmentation across AWS and GCP [10]

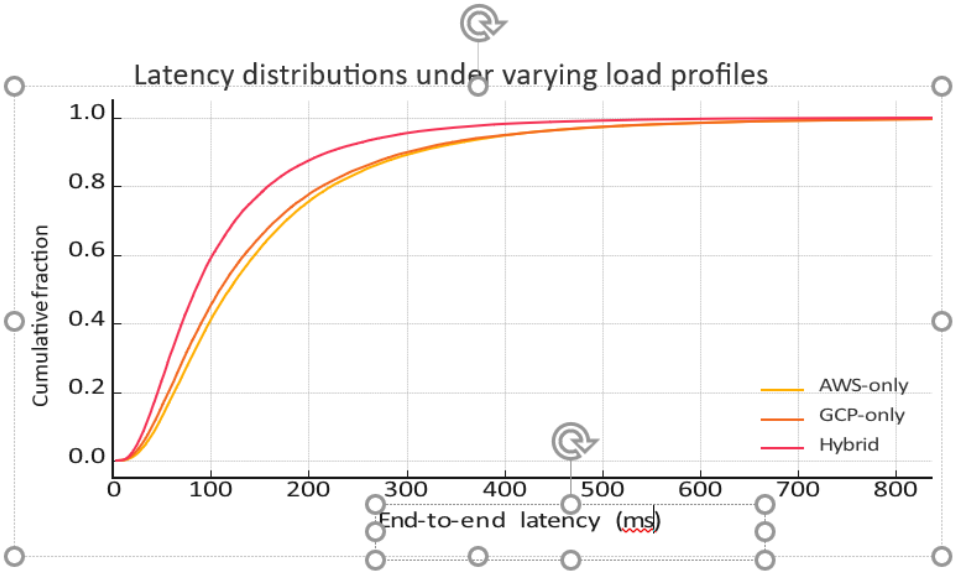

Figure 4: Latency distributions under varying load profiles.

Table 3: Performance Comparison

Metric	AWS-only	GCP-only	Hybrid
P95 Latency (ms)	420	405	290
Accuracy (%)	91.2	92.1	96.4
FPR (%)	4.5	4.2	2.8
Availability (%)	99.3	99.4	99.9

3 Results and Discussion

Latency distributions under varying load conditions were shown in Fig. 4. Comparative detection performance was illustrated in Fig. 5. System throughput as concurrency increased was presented in Fig. 6.

Performance metrics across architectures were summarized in Table 3. Latency was decomposed by component in Table 4. The impact of removing key components was shown in Table 5.

Normalized monthly cost components across architectures were shown in Fig. 7. Alignment with regulatory frameworks was summarized in Table 6. Threat scenarios and their mitigations were listed in Table 7.

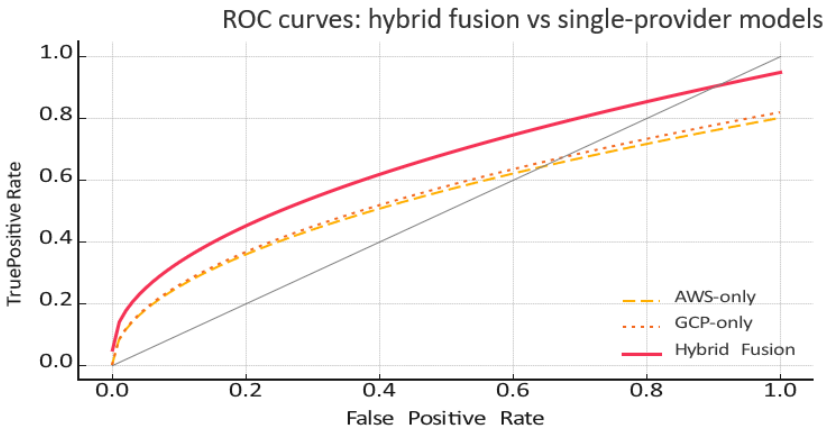

Figure 5: ROC curves: hybrid fusion vs. single-provider models.

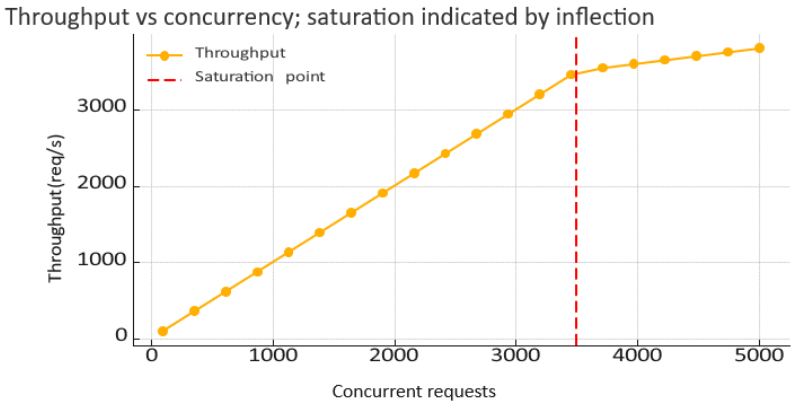

Figure 6: Throughput vs. concurrency; saturation indicated by inflection.

Table 4: Latency Breakdown by Component

Component	Median (ms)	P95 (ms)
Local verify	62	110
Replication	24	58
Feature computation	31	75
Inference	19	41
Enforcement	22	46

Table 5: Ablation Results

Variant	P95 Latency	FPR
Full hybrid	290	2.8
No fusion	335	3.9
No chaos	282	2.6

Normalized monthly cost components (compute, storage, egress, fraud loss)

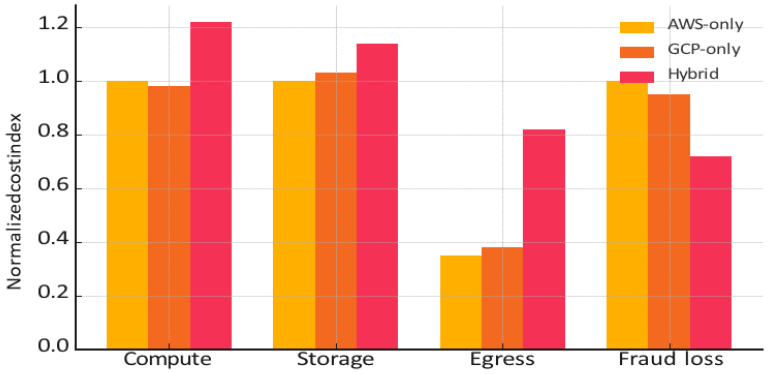

Figure 7: Normalized monthly cost components.

Table 6: Compliance Mapping

Framework	Relevant Controls
GDPR	Data residency; DPIAs; minimization
CCPA	Access/erasure workflows; purpose limits
PCI DSS	Network segmentation; key management
SOC 2	Change control; monitoring; incident response

Table 7: Threats and Mitigations

Threat	Mitigation
Credential stuffing	Rate limits; IP reputation; device binding
Replay	Nonces; short token TTL; session graph checks
Emulation	Device attestation; sensor liveness
Proxy rotation	ASN entropy; geo-velocity bounds
Account takeover	Step-up; anomaly-based hold

4 Conclusion

A hybrid AWS–GCP identity verification framework was designed and evaluated for realtime fraud mitigation. Under controlled experiments, the architecture reduced P95 latency and error rates while increasing availability compared with single-cloud baselines. The design favored regulated, latency- sensitive domains, and provided a pathway to incremental adoption of multicloud controls. Future work included exten- sion to additional providers, integration of online learning to handle concept drift, and privacy-preserving analytics at the edge From a practitioner perspective, the results suggested guidelines for adoption. Hybrid multi-cloud verification should be considered in environments where fraud risk was high, compliance requirements mandated data residency across regions, or service availability was critical. In contrast, single-cloud deployments might remain sufficient for organizations with lower transaction volumes, limited regulatory exposure, or tight cost constraints. A phased adoption path was recommended, starting with selective high-risk transaction types in hybrid mode while maintaining routine authentication in a single cloud.

Acknowledgment

The authors acknowledged contributions from architects, data scientists, and practitioners. Open-source tools such as Kafka, TensorFlow Serving, and k6 were instrumental. Feedback from peer reviewers helped refine the scope and articulation of implications.

References

1. Amazon Web Services, “Amazon cognito documentation,” https://docs.aws.amazon.com/cognito/, 2025, accessed: 2025-08-17.
2. Google Cloud, “Identity platform documentation,” https://cloud.google.com/identityplatform/docs, 2025, accessed: 2025-08-17.
3. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
4. J. Kreps, N. Narkhede, and J. Rao, “Kafka: a distributed messaging system for log processing,” in *Proceedings of the NetDB Workshop*, 2011.
5. C. Olston, N. Fiedel, K. Gorovoy, D. Harmsen, F. Lao, V. Rajashekhar, S. Ramesh, J. Soyke, C. Wang, M. Wicke, Y. Xia, and T. Yuen, “Tensorflow-serving: Flexible, high-performance ml serving,” in *Proceedings of the Workshop on ML Systems*, 2017.
6. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *Proc. International Conference on Information Systems Security and Privacy (ICISSP)*, 2018, pp. 108–116.
7. D. W. Chadwick and G. Inman, “Attribute aggregation in federated identity management,” *Computer*, vol. 42, no. 5, pp. 33–40, 2009.
8. A. C. Bahnsen, D. Aouada, A. Stojanovic, and B. Ottersten, “Cost sensitive credit card fraud detection using bayes minimum risk,” in *Proc. IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2013, pp. 333–338.
9. A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, “Credit card fraud detection: A realistic modeling and a novel learning strategy,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3784–3797, 2018.
10. S. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero trust architecture,” *NIST Special Publication*, vol. 800-207, 2020.