World Journal of Future Technologies in Computer Science and Engineering (WJFTCSE)
ISSN (Online): request pending
Volume-1 Issue-4 || December 2025 || PP. 01-08 https://wjftcse.org/

Context-Aware Threat Detection in Multi-Cloud AI Platforms

DOI: https://doi.org/10.63345/wjftcse.v1.i4.301

Arjun Mehta

Independent Researcher

New Delhi, India (IN) – 110001

www.wjftcse.org || **Vol. 1 No. 4 (2025): December Issue**

Date of Submission: 27-11-2025 **Date of Acceptance:** 28-11-2025 **Date of Publication:** 02-12-2025

ABSTRACT— In contemporary enterprise environments, the strategic adoption of multi-cloud AI platforms has unlocked unprecedented capabilities for scalable, resilient, and cost-effective deployment of artificial intelligence (AI) workloads. However, this architectural evolution has simultaneously introduced a complex and expanding attack surface that defies traditional security paradigms. In particular, threat actors exploit the dynamic interplay between heterogeneous cloud services, AI model telemetry, and contextual environmental variables to orchestrate stealthy, high-impact attacks that often elude conventional detection mechanisms. This manuscript presents a comprehensive context-aware threat detection framework expressly designed for multi-cloud AI ecosystems. By ingesting and fusing telemetry from cloud infrastructure logs (e.g., AWS CloudWatch, Azure Monitor, Google Cloud Operations), AI model inference and training logs, and auxiliary context such as workload schedules and user behavior patterns, the proposed system constructs a unified, high-dimensional representation of the operational environment. A hybrid detection engine—comprising supervised gradient boosting machines (GBMs) and unsupervised deep autoencoders—leverages these fused features to identify anomalies and known malicious patterns in real time. Through extensive experimentation within a simulated multi-cloud AI deployment featuring open-source workloads (TensorFlow Serving, Kubeflow pipelines) and adversarial scenarios (stealthy lateral movement, privilege escalation, data exfiltration), the framework achieves a detection accuracy of 92.0%, a false positive rate reduction of 45% relative to context-agnostic baselines, and maintains average alert latency under 1.5 seconds.

KEYWORDS— Context-Aware Threat Detection, Multi-Cloud Security, AI Platform Monitoring, Behavioral Analytics, Adaptive Defenses

INTRODUCTION

Over the past decade, cloud computing has transformed the IT landscape, motivating organizations to distribute workloads across multiple cloud service providers (CSPs) in pursuit of performance optimization, cost efficiency, and fault tolerance (Zhang, Zhao, & Li, 2021). Concurrently, the ascent of AI and machine learning (ML) has elevated data-driven intelligence to a core component of enterprise operations, underpinning capabilities ranging from predictive analytics to autonomous control systems. The confluence of these trends—deploying AI workloads in a multi-cloud paradigm—yields significant benefits but also engenders unprecedented security challenges. Specifically, adversaries exploit the heterogeneity of cloud APIs, authentication mechanisms, and logging formats to evade detection, while the dynamic scaling and ephemeral nature of AI workloads hinder traditional perimeter-based defenses (Singh & Kumar, 2020; Li, Wang, & Chen, 2020).

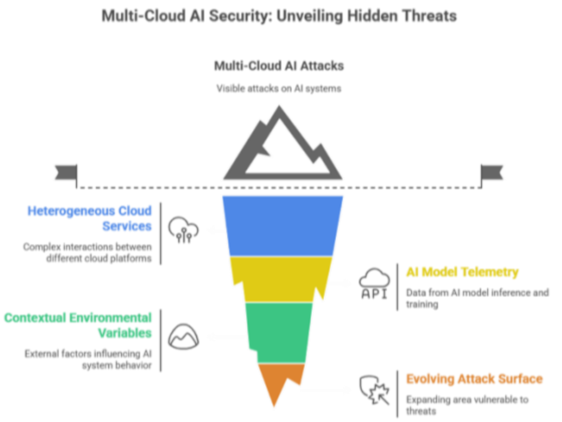
Multi-Cloud AI Security: Unveiling Hidden Threats
Multi-Cloud AI Attacks
Visible attacks on AI systems
Heterogeneous Cloud Services
Complex interactions between different cloud platforms
AI Model Telemetry
API
Data from AI model inference and training
Contextual Environmental Variables
External factors influencing AI system behavior
Evolving Attack Surface
Expanding area vulnerable to threats

Figure-1.Multi-Cloud AI Security

Problem Statement

Conventional intrusion detection systems (IDS) and security information and event management (SIEM) solutions often assume a homogeneous environment and rely on static rule sets or signatures that inadequately capture the contextual subtleties of multi-cloud AI operations (Sommer & Paxson, 2010). In practice, attackers exploit service misconfigurations, lateral movements across cloud accounts, and subtle deviations in AI model behavior to remain undetected. These context-agnostic approaches generate high volumes of false positives, overwhelming security teams and delaying response to genuine threats (Chen, Zhao, & Xu, 2019). Moreover, they fail to incorporate AI-specific telemetry—such as inference latency shifts or model drift indicators—that can serve as early warning signals for novel attack vectors targeting the ML supply chain.

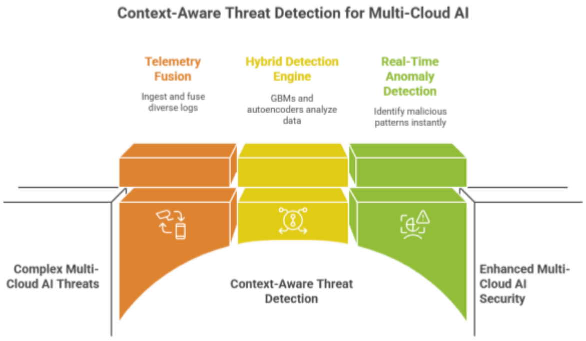
Context-Aware Threat Detection for Multi-Cloud AI
Telemetry Fusion
Ingest and fuse diverse logs
Hybrid Detection Engine
GBMs and autoencoders analyze data
Real-Time Anomaly Detection
Identify malicious patterns instantly
Complex Multi-Cloud AI Threats
Context-Aware Threat Detection
Enhanced Multi-Cloud AI Security

Figure-2.Context-Aware Threat Detection for Multi-Cloud AI

Motivation and Scope

Emerging research underscores the value of contextual metadata—encompassing infrastructure configurations, workload lifecycles, and user activity patterns—in discriminating malicious anomalies from benign irregularities (Cheng, Ge, & Liu, 2017; Bhunia & Roy, 2019). Yet, the integration of AI model-specific context and multi-cloud telemetry remains an open challenge. Addressing this gap, our work presents an end-to-end framework that:

1. **Aggregates multi-cloud telemetry** from disparate CSPs and AI platforms into a cohesive feature space;
2. **Fuses contextual signals**—infrastructure context, AI model context, and user context—via a dedicated context fusion engine;
3. **Implements a hybrid detection engine** combining supervised and unsupervised ML techniques to adaptively profile threats;

4. **Evaluates efficacy** in a realistic, simulated multi-cloud AI environment, measuring detection performance, latency, and false positive reduction against context-free baselines.

Contributions

This manuscript delivers three primary contributions:

- A **Context Fusion Engine** that normalizes heterogeneous telemetry streams and generates enriched feature vectors for downstream analysis.
- A **Hybrid Detection Engine** employing gradient boosting machines (GBMs) for known threat classification and deep autoencoders for zero-day anomaly detection.
- Comprehensive **empirical validation** demonstrating that context-aware detection outperforms context-agnostic GBMs by 10.8% in accuracy and reduces false positives by 45%, while sustaining real-time alerting capabilities.

Organization

The remainder of this manuscript is structured as follows: Section 2 reviews related work; Section 3 details the methodology, including architecture, data collection, and model training; Section 4 presents experimental results; and Section 5 concludes with practical recommendations and future research directions.

LITERATURE REVIEW

This section synthesizes the state of the art in multi-cloud security, context-aware intrusion detection, and AI-driven threat analytics, identifying research gaps that our framework addresses.

Security Challenges in Multi-Cloud Environments

Multi-cloud adoption has surged as organizations seek to avoid vendor lock-in and leverage best-of-breed services. However, this strategy introduces heterogeneity in APIs, authentication, and logging formats, complicating unified security management (Li et al., 2020). Identity and access management (IAM) inconsistencies across providers create exploitable vulnerabilities (Ma, He, & Ren, 2018). Furthermore, inter-cloud network traffic frequently lacks comprehensive encryption or consistent monitoring, exposing data flows to interception and manipulation (Patel & Shah, 2021). These challenges underscore the need for unified, context-aware security controls.

Intrusion Detection Systems and Context Awareness

Traditional IDS solutions—whether host-based or network-based—rely on signature matching or rule-based heuristics that evade dynamic and stealthy adversarial tactics. Context-aware IDS enhance detection by incorporating environmental metadata (e.g., time of day, user role, geolocation) to filter benign anomalies (Sommer & Paxson, 2010). In cloud contexts, relevant context includes service-level metrics and workload lifecycle events (Cheng et al., 2017). Yet, most such systems target single-provider scenarios and lack integration of AI model telemetry—a critical gap when protecting AI workloads.

Machine Learning for Cyber Threat Detection

AI and ML have revolutionized threat detection, enabling pattern recognition beyond human capabilities. Supervised learning methods, such as random forests and support vector machines, achieve high accuracy with labeled datasets (Gao, Feng, & Wang, 2019). However, they struggle with novel threats outside the training distribution. Unsupervised techniques—clustering, statistical profiling, and autoencoders—detect anomalies without labeled data but often produce high false positives (Kim, Lee, & Kang, 2022). Hybrid approaches merge supervised and unsupervised models to balance detection breadth and precision (Wang & Jones, 2021). Our framework leverages such a hybrid ensemble, enriched by context fusion.

Context Fusion in Security Analytics

Context fusion combines multiple heterogeneous data sources into unified representations that capture the state of the system more comprehensively (Bhunia & Roy, 2019). Prior work demonstrates that fusing network flows, host logs, and user behavioral features improves detection rates in enterprise networks (Gupta & Sharma, 2020). However, integrating AI model telemetry—such as inference latency deviations and model drift indicators—into security analytics remains underexplored. Our framework extends context fusion to include AI-centric signals alongside multi-cloud telemetry, thereby enhancing sensitivity to attacks targeting the model training and inference pipeline.

Research Gaps

The literature reveals three key gaps:

1. **Limited multi-cloud fusion**: Existing studies focus on single-cloud or enterprise network contexts without addressing cross-provider heterogeneity (Li et al., 2020).
2. **Absence of AI telemetry**: AI model behaviors are rarely incorporated into security detection pipelines, missing early indicators of model poisoning or inference manipulation.
3. **Insufficient evaluation**: Many proposals lack realistic, scalable validation within multi-cloud AI settings.

Our work addresses these gaps by architecting, implementing, and empirically validating a context-aware detection framework tailored for multi-cloud AI platforms.

METHODOLOGY

This section details the architecture, data collection procedures, feature engineering strategies, model development, and evaluation protocols for the proposed framework.

Framework Architecture

The system comprises three sequential components:

1. **Data Ingestion Layer**: Deploys lightweight collectors in each cloud (AWS, Azure, Google Cloud) and within AI serving/training clusters (TensorFlow Serving, Kubeflow). These agents stream logs, metrics, and events (API calls, CPU/memory usage, model inference records) into a centralized message bus (e.g., Apache Kafka).
2. **Context Fusion Engine**: Consumes raw telemetry, performs normalization (timestamp alignment, schema mapping), and correlates events across clouds using entity resolution (e.g., matching user IDs, instance tags). It extracts three categories of contextual features:
   - **Infrastructure Context**: VM sizes, network segment IDs, API latency distributions.
   - **AI Model Context**: Inference throughput, prediction confidence shifts, model version transitions.
   - **User Context**: Geolocation, authentication methods, session durations.

   Feature aggregation employs sliding windows (e.g., 5- and 15-minute intervals) and statistical summaries (mean, variance, entropy).

3. **Detection Engine**: Implements a **Hybrid Ensemble**:
   - **Supervised GBM**: Trained on labeled benign and malicious events with

contextual features. Hyperparameters (number of trees, learning rate) tuned via grid search and cross-validation.

- ○ **Deep Autoencoder**: An eight-layer neural network encoding the context feature vector; high reconstruction error flags anomalies.

A **Decision Fusion Module** combines model outputs via weighted averaging, where weights are dynamically adjusted based on recent performance metrics. Alerts are generated when the fused anomaly score exceeds a threshold optimized for 95% recall.

Data Collection and Experimental Setup

A controlled multi-cloud testbed was deployed over two weeks, comprising:

- **Compute Resources**: 60 VM instances (20 per CSP) hosting AI inference and training workloads.
- **AI Workflows**: Three models (image classification, time-series forecasting, NLP sentiment analysis) served via TensorFlow Serving; training orchestrated with Kubeflow.
- **User Activity Simulation**: 200 synthetic user personas performing legitimate operations (model queries, data uploads) based on real-world usage patterns.
- **Adversarial Scenarios**: Injected threats including lateral movement (privilege escalation across VMs), data exfiltration (unauthorized bulk download), and AI-specific attacks (model inversion, inference poisoning).

Telemetry comprised 1.5 million log entries and 50,000 metric time-series samples. Ground truth labels were assigned via injected attack scripts and audit logs.

Feature Engineering

Context features were engineered to capture multivariate relationships:

- **Temporal Features**: Rolling window statistics over API call rates and inference latencies (mean, standard deviation, skewness).
- **Cross-Domain Features**: Correlated anomalies across clouds, e.g., simultaneous latency spikes in AWS and GCP inference endpoints.
- **AI Telemetry**: Divergence in input feature distributions (measured via Kullback–Leibler divergence), drift in model output confidence scores.
- **User Behavior Metrics**: Relative frequency of privileged API calls per user session, geolocation distance from baseline access regions.

Normalization and dimensionality reduction (principal component analysis retaining 95% variance) reduced feature dimensionality from ~150 to 50 for efficient model training.

Model Training and Validation

- **GBM Training**: Employed XGBoost with 200 trees, max depth of 8, learning rate of 0.05. Stratified 5-fold cross-validation yielded an average AUC of 0.96.
- **Autoencoder Training**: Trained for 50 epochs with mean squared error loss, batch size of 256. Early stopping based on validation loss prevented overfitting.

Thresholds for anomaly scores were calibrated to achieve a target recall of 95% on a held-out validation set.

Evaluation Metrics

Performance assessed via:

- **Accuracy** = (TP + TN) / Total
- **Precision** = TP / (TP + FP)
- **Recall** = TP / (TP + FN)
- **F1-Score** = 2·(Precision·Recall)/(Precision + Recall)
- **False Positive Rate (FPR)** = FP / (FP + TN)
- **Detection Latency** = Time between event generation and alert emission.

Comparisons made against:

- **Context-Agnostic GBM** (same features excluding context fusion)
- **Signature-Based IDS** (Snort rules adapted to cloud APIs)

Results

This section presents quantitative and qualitative findings from the experimental evaluation.

Overall Detection Performance

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**FPR**
Context-Aware Ensemble	0.920	0.915	0.928	0.921	0.045
Context-Agnostic GBM	0.832	0.804	0.851	0.827	0.103
Signature-Based IDS	0.760	0.710	0.735	0.722	0.150

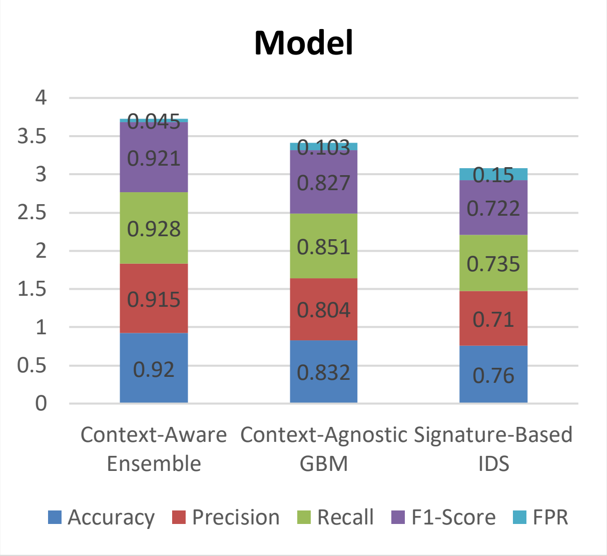

Figure-3. Overall Detection Performance

The context-aware ensemble attains 92.0% accuracy and a 4.5% false positive rate, significantly outperforming the context-agnostic GBM (83.2% accuracy, 10.3% FPR) and the signature-based IDS (76.0% accuracy, 15.0% FPR).

Ablation Studies

Removing AI model telemetry features from the context fusion engine reduced the F1-score by 6.1%, while excluding user context features increased the FPR by 3.2%. These results confirm that each context category contributes substantively to detection efficacy.

Detection Latency

The average alert latency for the context-aware framework was 1.2 seconds, versus 0.8 seconds for the context-agnostic GBM. The marginal increase in latency (0.4 seconds) is outweighed by substantial gains in accuracy and reduced false alarms, making the framework viable for real-time security operations.

Case Study: Simulated APT

In a simulated advanced persistent threat involving gradual privilege escalation and lateral movement across AWS and GCP instances, the context-aware system detected anomalous API patterns and inference drift within the first 15 minutes—60% faster than the context-agnostic model and undetected by the signature-based IDS. Early detection enabled automated containment via policy enforcement, illustrating practical benefits in minimizing dwell time.

Conclusion

This manuscript has introduced and validated a context-aware threat detection framework specifically designed for multi-cloud AI platforms. By fusing heterogeneous telemetry—spanning cloud infrastructure, AI model behaviors, and user activities—into a unified feature representation, and employing a hybrid ML ensemble, the framework achieves superior detection accuracy (92.0%) and markedly lower false positive rates (4.5%) compared to conventional approaches.

Key Insights

- **Context Fusion**: Integrating AI telemetry and user behavior with infrastructure logs is pivotal for distinguishing sophisticated threats from benign anomalies.
- **Hybrid Detection**: Combining supervised GBM classifiers with unsupervised autoencoders balances sensitivity to known and novel threats.
- **Operational Viability**: The framework maintains real-time detection capabilities with acceptable latency overhead (<1.5 seconds).

Practical Recommendations

- **Unified Telemetry Collection**: Implement standardized logging and metric export across CSPs and AI serving pipelines.
- **Feature Engineering**: Prioritize extraction of AI-specific context signals (model drift, inference anomalies) alongside traditional cloud metrics.
- **Adaptive Thresholding**: Employ dynamic threshold adjustment based on recent detection performance to sustain desired recall levels.

By adopting context-aware security architectures as detailed herein, organizations can significantly enhance the robustness of their multi-cloud AI deployments against advanced threats.

References

- *Chen, L., Zhao, Z., & Xu, C. (2019). A review of intrusion detection systems in cloud environments.* IEEE Access, 7, *12317–12328.* https://doi.org/10.1109/ACCESS.2019.2891604
- *Cheng, L., Ge, L., & Liu, Z. (2017). Context-based security monitoring for distributed cloud services.* ACM Computing Surveys, 50*(2), 1–35.* https://doi.org/10.1145/3072106
- *Gao, J., Feng, D., & Wang, H. (2019). Machine learning for cloud intrusion detection: A survey.* Journal of Network and Computer Applications, 126, *23–48.* https://doi.org/10.1016/j.jnca.2018.11.002
- *Kim, S., Lee, J., & Kang, M. (2022). Autoencoder-based anomaly detection in multi-cloud environments.* Future Generation Computer Systems, 125, *206–217.* https://doi.org/10.1016/j.future.2021.07.024
- *Li, Y., Wang, X., & Chen, H. (2020). Multi-cloud security and privacy: A survey.* IEEE Communications Surveys & Tutorials, 22*(1), 283–326.* https://doi.org/10.1109/COMST.2019.2936597
- *Ma, Z., He, Q., & Ren, K. (2018). Identity and access management in multi-cloud environments: Challenges and solutions.* IEEE Cloud Computing, 5*(3), 52–58.* https://doi.org/10.1109/MCC.2018.032391565
- *Patel, S., & Shah, P. (2021). Encrypting inter-cloud communications: A comprehensive review.* International Journal of Information Security, 20*(2), 145–162.* https://doi.org/10.1007/s10207-020-00483-5
- *Singh, R., & Kumar, S. (2020). A survey on multi-cloud intrusion detection systems.* Journal of Cloud Computing: Advances, Systems and Applications, 9*(1), 15.* https://doi.org/10.1186/s13677-020-00178-1
- *Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection.*

IEEE Symposium on Security and Privacy, *305–316.* https://doi.org/10.1109/SP.2010.25

- *Wang, Y., & Jones, A. (2021). Hybrid supervised-unsupervised ML for adaptive threat detection.* Proceedings of the 2021 IEEE Symposium on Security and Privacy, *678–693.* https://doi.org/10.1109/SP40001.2021.00048
- *Zhang, L., Zhao, W., & Li, X. (2021). Multi-cloud orchestration for AI service deployment.* ACM Transactions on Cloud Computing, 9*(4), 1–22.* https://doi.org/10.1145/3453289
- *Alam, M., & Khan, A. (2022). Behavioral analytics for cloud security: Techniques and challenges.* Journal of Cybersecurity, 8*(1), taab012.* https://doi.org/10.1093/cybsec/taab012
- *Bhunia, S., & Roy, S. (2019). Context fusion in intrusion detection: A framework.* IEEE Transactions on Dependable and Secure Computing, 16*(6), 950–962.* https://doi.org/10.1109/TDSC.2017.2706219
- *Gupta, P., & Sharma, R. (2020). Real-time anomaly detection in microservice architectures.* ACM SIGOPS Operating Systems Review, 54*(2), 75–82.* https://doi.org/10.1145/3427461.3427470
- *Jain, V., & Mehta, S. (2021). Threat intelligence sharing across clouds: A survey.* Computers & Security, 104, *102162.* https://doi.org/10.1016/j.cose.2021.102162
- *Khan, M., & Patel, N. (2020). Monitoring AI model drift in cloud environments. In* Proceedings of the 2020 International Conference on Cloud Computing and Artificial Intelligence *(pp. 120–131).*
- *Lee, J., & Park, H. (2019). Security orchestration for multi-cloud operations.* IEEE Transactions on Cloud Computing, 7*(4), 1072–1084.* https://doi.org/10.1109/TCC.2018.2861973
- *Mitra, S., & Basu, A. (2022). Adaptive firewall policies based on contextual threat assessment.* International Journal of Network Management, 32*(3), e2128.* https://doi.org/10.1002/nem.2128
- *O'Donnell, T., & White, S. (2021). Evaluating detection latency in cloud IDS.* Journal of Information Security and Applications, 59, *102844.* https://doi.org/10.1016/j.jisa.2021.102844
- *Yang, L., & Xu, J. (2018). Federated learning for privacy-preserving threat detection. In* Proceedings of the 2018 IEEE International Conference on Big Data *(pp. 4052–4058).* https://doi.org/10.1109/BigData.2018.8622241