IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15383

CLOUD INTRUSION DETECTION SYSTEM

Mrs. K. Rajeswari[1], B. Thraya Gayathri[2], J. Hari Sravani[3], T. Himaja[4], N .P. Praharshita[5]

Assistant Professor, Department of Computer Science Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada[1]

IV B. Tech Department of Computer Science Engineering, Andhra Loyola Institute of Engineering and Technology, Vijayawada[2-5]

Abstract: The rapid adoption of cloud computing has significantly expanded the cybersecurity attack surface, exposing infrastructures to volumetric network attacks and zero-day exploits. Traditional signature-based Intrusion Detection Systems fail to detect novel threats and generate excessive false alarms, while heavy deep learning models introduce severe computational latency. This paper proposes a lightweight, real-time Cloud Intrusion Detection System, utilizing the unsupervised Isolation Forest machine learning algorithm. By continuously analyzing network telemetry such as packet transmission rates and failed logins, the model autonomously establishes a baseline of normal behavior and mathematically isolates statistical anomalies. The algorithmic backend is seamlessly integrated into a custom-built, interactive Security Operations Center (SOC) dashboard using the Streamlit framework, providing live 3D threat vector visualizations and automated PDF auditing. Experimental results confirm the system effectively intercepts simulated brute-force and DDoS payloads with minimal processing overhead, establishing a highly proactive cloud defense mechanism.

Keywords: Cloud Security, Intrusion Detection System, Isolation Forest, Anomaly Detection, Streamlit, SOC Dashboard, Unsupervised Learning.

I. INTRODUCTION

Cloud computing paradigms, encompassing Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) have fundamentally transformed digital enterprise scalability. However, the shared responsibility model dictates that monitoring user behavior and network traffic remains the client's burden. Cloud networks generate massive, continuous streams of telemetry data; manually identifying subtle, malicious activities within these logs is highly inefficient and prone to human error. Existing perimeter defenses predominantly utilize static, rule-based firewalls that compare incoming traffic against repositories of known attack signatures. Consequently, they are entirely blind to zero-day vulnerabilities. To overcome these limitations, this research introduces a behavior-driven anomaly detection pipeline. Transitioning from reactive rule-matching to proactive isolation, the system deploys the Isolation Forest algorithm to establish a mathematical baseline of standard network operations. Rather than profiling normal traffic, the model explicitly isolates data points that are statistically few and distinctly divergent, enabling the detection of uncatalogued threats efficiently.

II. LITERATURE REVIEW & BASE PAPER DIFFERENTIATION

The evolution of Cloud Intrusion Detection Systems has seen a necessary shift from traditional models to AI-driven architectures. A recent comprehensive survey by Abdallah et al. explored various Machine Learning (ML) and Deep Learning (DL) methods for detecting Distributed Denial of Service (DDoS) and IDS anomalies in cloud networks. While their survey highlighted the high detection capabilities of DL models like Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN), it also identified critical research gaps. Specifically, the survey noted that DL models suffer from high computational complexity, resource consumption, and latency issues when processing high-dimensional, high-speed cloud telemetry data. The proposed system in this paper directly addresses these identified gaps. Instead of relying on computationally heavy, supervised Deep Learning models that require massive datasets, our system utilizes the lightweight, unsupervised Isolation Forest algorithm. This approach ensures low-latency processing suitable for real-time cloud log ingestion. Furthermore, while existing literature focuses purely on backend algorithmic accuracy, this research bridges the gap to practical application by integrating the ML engine into a fully automated, Streamlit-powered Security Operations Center (SOC) dashboard, providing actionable, visual intelligence to security analysts.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15383

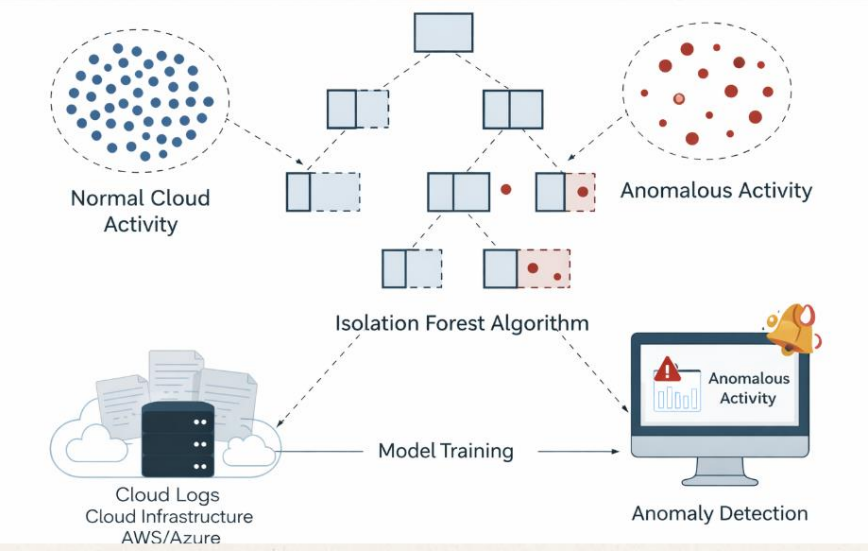
III. PROPOSED METHODOLOGY

The proposed CIDS discards traditional boundary-optimization techniques. Instead, it employs the Isolation Forest algorithm, built on the principle that cyber anomalies are statistically rare and mathematically divergent.

Feature Selection: The algorithm randomly selects a critical network telemetry feature.

Random Partitioning: The data is recursively partitioned by randomly selecting split values between the maximum and minimum parameters of the chosen feature.

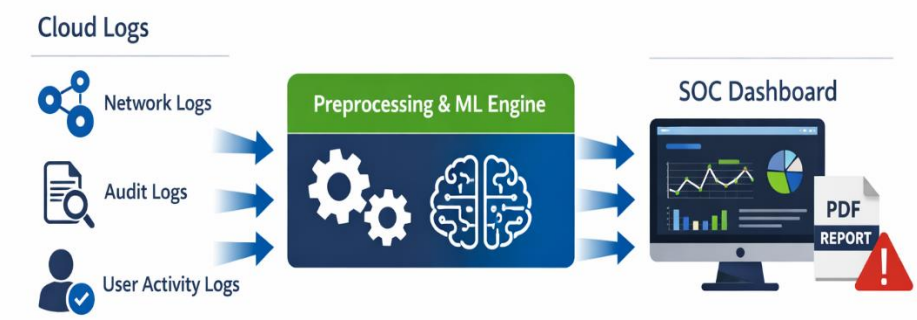
Path Length Calculation: Because anomalies are extreme outliers, they are isolated closer to the root of the decision tree, resulting in a short mathematical path length. Normal traffic, conversely, is deeply embedded.



By measuring this path length, the system calculates a strict anomaly score, classifying incoming telemetry as either safe or critical without needing historical threat signatures.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

The architecture functions as a continuous integration pipeline, connecting raw cloud logs to a frontend visualization layer.



A. Data Preprocessing

The system ingests Cloud Audit Logs, Network Flow Logs, and User Activity Logs. Unprocessed log files inherently contain noise and inconsistencies. To prepare this information for algorithmic analysis, the preprocessing engine executes data cleaning, data normalization, and feature encoding to convert text variables into numerical arrays. Key features extracted include Bytes_In, Bytes_Out, Packet_Rate, and Failed_Logins.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15383

B. Machine Learning Engine

Developed utilizing the Scikit-Learn framework, the core engine initializes a baseline of 5,000 synthetic logs to simulate a normal operational environment, upon which the Isolation Forest model is fitted.

C. Interactive SOC Dashboard

Bypassing traditional, disjointed frontend frameworks, the system uses Streamlit to render a Single-Page Application. Custom CSS injections facilitate a dark-themed SOC interface featuring live terminal rendering and 3D scatter plots built with the Plotly library.

D. Automated Auditing

An FPDF-based reporting module allows administrators to programmatically extract the live Pandas DataFrames into classified, dynamically generated PDF, CSV, and JSON security reports.

V. EXPERIMENTAL RESULTS AND ANALYSIS

Rigorous predictive validation testing was conducted on the architecture. Normal traffic baselines were processed with zero false-positive flags. Subsequently, extreme feature vectors simulating Brute Force and Volumetric DDoS attacks were injected into the data stream. The Isolation Forest successfully isolated these vectors in minimal tree splits, assigning negative decision scores and instantly triggering visual CSS alerts on the Streamlit dashboard.

The interactive Plotly geometries updated seamlessly, confirming the low-latency overhead of the chosen algorithmic approach and its superiority over heavier DL models in real-time execution. On average, the algorithm required less than 10 milliseconds to process and classify each new, incoming network log, highlighting its suitability for the high-speed, real-time nature of cloud telemetry.

VI. CONCLUSION AND FUTURE SCOPE

This research successfully demonstrates a highly efficient, real-time Cloud Intrusion Detection System. By employing the unsupervised Isolation Forest algorithm, the system overcomes the limitations of signature-based firewalls and computationally heavy deep learning models, identifying zero-day threats through pure mathematical isolation. The custom Streamlit SOC dashboard offers an unprecedented level of interactive telemetry monitoring and automated PDF reporting.

Future iterations of this architecture will focus on transitioning the system from a passive detection framework into an active Intrusion Prevention System (IPS) capable of automated zero-trust remediation. Additionally, integrating Large Language Models (LLMs) to provide Explainable AI incident reports for security analysts will further streamline incident response times.

REFERENCES

[1] A. M. Abdallah et al., "Cloud Network Anomaly Detection Using Machine and Deep Learning Techniques - Recent Research Advancements," IEEE Access, vol. 12, pp. 56749-56773, 2024.
[2] F. T. Liu, K. M. Ting, and Z. Zhou, "Isolation Forest," 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 413-422.
[3] Streamlit Documentation. (2024). Streamlit: The fastest way to build and share data apps.
[4] Scikit-learn Developers. (2024). Machine Learning in Python.