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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

A Study of Cloud-Native Intrusion Detection Using VPC Flow Logs and Ensemble Learning

Binny Thomas[1], Hisana Saji[2], Bhavya Shivani H[3], Siva H S[4], Sagara M R[5]

Student, Computer Science, St. Thomas Institute for Science and Technology, Trivandrum, India[1]

Student, Computer Science, St. Thomas Institute for Science and Technology, Trivandrum, India[2]

Student, Computer Science, St. Thomas Institute for Science and Technology, Trivandrum, India [3]

Student, Computer Science, St. Thomas Institute for Science and Technology, Trivandrum, India [4]

Asst. Professor, Computer Science, St. Thomas Institute for Science and Technology, Trivandrum, India [5]

Abstract: Cloud computing has rapidly evolved into the foundational infrastructure for modern digital services, enabling organizations to deploy applications with unprecedented scalability, elasticity, and cost efficiency. However, the dynamic and distributed nature of cloud-native environments introduces complex security challenges that traditional intrusion detection systems are ill-equipped to address. Conventional IDS solutions depend on static network boundaries, deep packet inspection, and predefined attack signatures, all of which are increasingly ineffective in cloud environments dominated by encrypted traffic, ephemeral workloads, and software-defined networking.

This study presents an extensive analysis of cloud-native intrusion detection systems that utilize Virtual Private Cloud (VPC) Flow Logs in conjunction with ensemble learning techniques. VPC Flow Logs provide scalable, lightweight, and privacy-preserving network traffic metadata, making them suitable for large-scale cloud monitoring. Ensemble learning methods combine multiple machine learning classifiers to enhance detection accuracy, reduce false positives, and improve robustness against evolving cyber threats. This paper systematically reviews existing research, explores architectural designs, analyzes detection methodologies, evaluates benefits and limitations, and discusses future research directions. The study demonstrates that ensemble-based intrusion detection using VPC Flow Logs offers a practical and effective solution for securing modern cloud infrastructures.

Keywords: Cloud Security, Intrusion Detection System, VPC Flow Logs, Ensemble Learning, Machine Learning, Cloud-Native Architecture.

I. INTRODUCTION

Cloud computing has fundamentally transformed the way organizations design, deploy, and manage information systems. The shift from traditional on-premise infrastructure to cloud-based platforms has enabled businesses to scale resources dynamically, reduce capital expenditure, and accelerate application development cycles. Cloud-native architectures utilize virtualization, containers, microservices, and orchestration technologies to deliver high availability, fault tolerance, and rapid deployment.

Despite these benefits, cloud environments have become attractive targets for cyber attackers. The concentration of sensitive data, public-facing services, and complex configurations increases the attack surface. Threat actors exploit vulnerabilities such as misconfigured storage services, weak identity and access management policies, exposed APIs, and insecure network configurations. Network-based attacks including port scanning, brute-force authentication, lateral movement, and distributed denial-of-service (DDoS) attacks are increasingly prevalent in cloud infrastructures.

Intrusion Detection Systems (IDS) are a critical component of cybersecurity frameworks, designed to monitor network traffic and detect malicious activities. Traditional IDS solutions were developed for static enterprise networks with well-defined perimeters and predictable traffic patterns. These systems rely on signature-based detection and deep packet inspection, techniques that are increasingly ineffective in cloud-native environments dominated by encrypted traffic and ephemeral workloads.

To address these challenges, cloud-native intrusion detection approaches leverage cloud-provided telemetry data and advanced analytics. Virtual Private Cloud (VPC) Flow Logs capture network traffic metadata at scale without violating

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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

privacy constraints. When combined with machine learning and ensemble learning techniques, these logs enable the development of adaptive, scalable, and intelligent intrusion detection systems. This paper provides an in-depth study of such systems, focusing on their design principles, advantages, limitations, and future potential.

II. CLOUD SECURITY THREAT LANDSCAPE

Cloud security threats differ substantially from those encountered in traditional on-premise networks due to the architectural, operational, and administrative characteristics of cloud computing. Cloud environments operate under a shared responsibility model, in which cloud service providers are responsible for securing the underlying infrastructure, while customers are responsible for securing applications, configurations, identities, and data. Misunderstanding or improper implementation of this model frequently results in security misconfigurations, which remain one of the leading causes of cloud security incidents.

One of the most prevalent threat vectors in cloud environments is misconfiguration. Improperly configured security groups, network access control lists, and storage permissions can expose sensitive services and data to the public Internet. Insecure application programming interfaces (APIs) further increase the attack surface, as cloud services heavily rely on APIs for automation, orchestration, and management. Attackers often exploit weak authentication mechanisms, excessive permissions, and poor key management practices to gain unauthorized access to cloud resources.

Network-level attacks continue to pose a significant threat in cloud infrastructures. Adversaries commonly perform reconnaissance activities, such as port scanning and service enumeration, to identify exposed services and vulnerable endpoints. These reconnaissance attacks are often stealthy and difficult to detect using traditional security mechanisms. Once a vulnerable service is identified, attackers may launch brute-force authentication attacks, exploit software vulnerabilities, or abuse misconfigured services to gain initial access.

After initial compromise, attackers frequently engage in lateral movement within the cloud network. Cloud environments often consist of interconnected virtual machines, containers, and microservices, enabling attackers to move across resources if network segmentation and identity controls are weak. Lateral movement significantly amplifies the impact of an attack, allowing adversaries to access sensitive data, escalate privileges, and compromise critical services. Detecting such behavior is particularly challenging due to the dynamic nature of cloud workloads and the frequent creation and termination of resources.

Insider threats represent another critical challenge in cloud security. These threats may arise from malicious insiders intentionally abusing their access privileges or from accidental insiders who unintentionally expose resources through misconfiguration or poor security practices. The extensive use of role-based access control and automation in cloud environments can magnify the consequences of insider threats if privileges are not carefully managed and monitored.

Distributed Denial-of-Service (DDoS) attacks also remain a major concern, as cloud-hosted applications are often publicly accessible and mission critical. Although cloud providers offer built-in DDoS mitigation mechanisms, application-layer attacks and resource exhaustion attacks can still disrupt services. Additionally, encrypted traffic, which is widely adopted to ensure confidentiality, limits the effectiveness of payload-based inspection techniques traditionally used for threat detection.

The scale, elasticity, and complexity of cloud infrastructures make manual security monitoring impractical. Traffic volumes can change rapidly due to auto-scaling, and workloads may be short-lived, leaving limited time for traditional security tools to detect malicious activity. As a result, automated, intelligent, and scalable intrusion detection systems are essential for effective cloud security monitoring. However, conventional IDS solutions struggle to adapt to these conditions, underscoring the need for cloud-native intrusion detection approaches that leverage flow-based telemetry and advanced machine learning techniques.

III. LIMITATIONS OF TRADITIONAL INTRUSION DETECTION SYSTEMS

Traditional intrusion detection systems (IDS) were primarily designed for conventional enterprise networks characterized by static infrastructures, well-defined network perimeters, and predictable traffic patterns. These systems are broadly classified into signature-based and anomaly-based intrusion detection approaches. While both categories have

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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

demonstrated effectiveness in controlled environments, they exhibit significant limitations when applied to modern cloud-native infrastructures.

Signature-based intrusion detection systems operate by comparing observed network traffic against a database of predefined attack signatures. These signatures represent known patterns associated with previously identified attacks. Signature-based IDS are highly effective in detecting well-known threats with low false-positive rates, making them suitable for environments where attack patterns are stable and frequently updated. However, their effectiveness is fundamentally limited by their reliance on prior knowledge. They are incapable of detecting zero-day attacks, novel attack variants, or sophisticated evasion techniques that do not match existing signatures. Furthermore, maintaining an up-to-date signature database requires continuous updates, which becomes increasingly challenging in large-scale and rapidly evolving cloud environments.

Anomaly-based intrusion detection systems attempt to overcome these limitations by modeling normal network behavior and identifying deviations from this baseline as potential intrusions. These systems are capable of detecting previously unseen attacks, making them attractive for dynamic threat environments. However, anomaly-based IDS face significant challenges in accurately defining normal behavior, particularly in cloud environments where traffic patterns are highly variable. Auto-scaling, workload migration, microservices communication, and fluctuating user demand cause frequent changes in network behavior, making it difficult to establish a stable baseline. As a result, anomaly-based systems often generate high false-positive rates, leading to alert fatigue and reduced trust in detection results.

Another major limitation of traditional IDS is their reliance on deep packet inspection (DPI). DPI examines packet payloads to identify malicious content and protocol violations. While effective in unencrypted networks, this approach introduces significant computational overhead and does not scale well in high-throughput environments. In cloud infrastructures, where encryption is widely adopted to protect data confidentiality, payload inspection becomes largely ineffective. The increasing use of protocols such as HTTPS and TLS renders DPI-based intrusion detection impractical, as encrypted payloads cannot be inspected without decrypting traffic, which raises privacy, compliance, and performance concerns.

Traditional IDS also struggle with the lack of visibility and scalability required in cloud environments. These systems were designed for static network topologies and often require manual configuration and deployment. In contrast, cloud environments are highly dynamic, with virtual machines, containers, and serverless functions being continuously created and terminated. Deploying and maintaining IDS sensors across such environments introduces operational complexity and performance overhead. Additionally, traditional IDS lack seamless integration with cloud-native logging, monitoring, and automation frameworks.

These limitations collectively highlight the inadequacy of conventional intrusion detection approaches for modern cloud infrastructures. The reliance on static signatures, difficulty in handling dynamic traffic patterns, inability to inspect encrypted traffic effectively, and lack of scalability necessitate the adoption of cloud-native intrusion detection systems. Such systems must leverage scalable, metadata-based monitoring mechanisms—such as VPC Flow Logs—and advanced machine learning techniques to provide effective, adaptive, and privacy-preserving intrusion detection in cloud environments.

IV. TAXONOMY OF CLOUD-NATIVE INTRUSION DETECTION SYSTEMS

Cloud-native intrusion detection systems can be broadly classified based on deployment model, detection technique, and data source. Understanding this taxonomy is essential for evaluating the suitability of different IDS approaches in cloud environments.

Based on deployment, cloud IDS can be categorized as host-based, network-based, and hybrid systems. Host-based IDS operate within virtual machines or containers and monitor system-level activities such as file access and process behavior. While they provide detailed visibility, they introduce performance overhead and are difficult to manage at scale. Network-based IDS monitor traffic flows between cloud resources and are better suited for large-scale monitoring. Hybrid systems combine both approaches to improve detection coverage.

Based on detection technique, IDS can be classified into signature-based, anomaly-based, specification-based, and hybrid detection systems. Signature-based systems rely on predefined attack patterns and are effective against known threats.

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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

Anomaly-based systems detect deviations from normal behavior, enabling the identification of unknown attacks but often producing false positives. Specification-based systems define expected behavior and flag violations, while hybrid systems combine multiple techniques to improve accuracy.

Based on data source, cloud IDS utilize packet-level data, flow-level metadata, system logs, or application logs. Packet-level inspection is increasingly impractical due to encryption, making flow-level data such as VPC Flow Logs a preferred data source. This taxonomy highlights why flow-based, ensemble-driven intrusion detection systems are well aligned with cloud-native security requirements.

V. VPC FLOW LOGS AS A DATA SOURCE

Virtual Private Cloud (VPC) Flow Logs are a cloud-native network logging mechanism provided by major cloud service providers to capture metadata about Internet Protocol (IP) traffic flowing within virtual networks. Unlike traditional packet capture systems, VPC Flow Logs record summarized flow-level information rather than raw packet data. The logged attributes typically include source and destination IP addresses, source and destination ports, transport protocol, number of packets and bytes transferred, flow start and end timestamps, and the action taken on the traffic (ACCEPT or REJECT).

One of the primary advantages of VPC Flow Logs is their scalability and low operational overhead. Since the logs are generated at the virtual network interface level, they can monitor traffic across virtual machines, containers, and microservices without requiring additional agents or sensors. This makes VPC Flow Logs particularly suitable for large-scale cloud deployments where installing and managing traditional IDS sensors is impractical. Additionally, VPC Flow Logs integrate seamlessly with cloud logging and analytics services, enabling automated processing and long-term storage.

A critical feature of VPC Flow Logs is their compatibility with encrypted traffic. As cloud communications increasingly rely on encryption protocols such as TLS to ensure confidentiality, payload-based inspection becomes ineffective. VPC Flow Logs bypass this limitation by focusing on metadata rather than content, allowing intrusion detection systems to operate without decrypting traffic. This approach also aligns with privacy and compliance requirements, as sensitive payload data is not collected or stored.

From an intrusion detection perspective, VPC Flow Logs provide valuable insights into network behavior patterns. Attributes such as flow duration, packet rate, byte distribution, and connection frequency can reveal abnormal activities associated with reconnaissance, denial-of-service attacks, and unauthorized access attempts. Although flow logs lack payload visibility, their rich temporal and statistical characteristics make them a powerful data source for cloud-native intrusion detection systems.

VI. FEATURE EXTRACTION AND DATA PREPROCESSING

Effective intrusion detection using VPC Flow Logs depends heavily on feature extraction and data preprocessing, as raw flow logs are not directly suitable for machine learning analysis. Flow log records may contain redundant attributes, missing values, and noisy data that can adversely affect model performance. Consequently, preprocessing is a critical step in building reliable intrusion detection systems.

Data preprocessing typically begins with data cleaning, which involves removing incomplete or corrupted records and handling missing values. Normalization and scaling are applied to numerical features such as packet count and byte count to ensure that features with larger magnitudes do not dominate the learning process. Categorical attributes, including protocol type and traffic action, are encoded using techniques such as one-hot encoding or label encoding.

Feature extraction focuses on deriving meaningful attributes that capture network behavior. Common features include flow duration, packet count, byte count, average packet size, and traffic direction. In addition to basic flow attributes, temporal features play a crucial role in intrusion detection. Features such as the number of connections initiated by a source within a fixed time window, inter-arrival times, and burst rates are particularly effective in identifying reconnaissance activities and flooding attacks.

DOI: 10.17148/IJARCCE.2026.151131

A major challenge in intrusion detection datasets is class imbalance**,** as malicious traffic often represents a small fraction of total network traffic. Imbalanced datasets can bias machine learning models toward normal traffic, reducing detection performance for attacks. Techniques such as oversampling, undersampling, and synthetic data generation (e.g., SMOTE) are commonly employed to address this issue. Proper feature extraction and preprocessing significantly enhance the performance and reliability of ensemble-based intrusion detection systems.

VII. MACHINE LEARNING FOR INTRUSION DETECTION

Machine learning has become a central component of modern intrusion detection systems due to its ability to learn complex patterns from large volumes of network traffic data. In cloud environments, machine learning-based IDS analyze flow-level features derived from VPC Flow Logs to distinguish between normal and malicious behavior.

Supervised learning techniques are widely used when labeled datasets are available. Models such as Logistic Regression, Support Vector Machines, Decision Trees, Random Forests, and Gradient Boosting classifiers have demonstrated strong performance in intrusion detection tasks. These models learn decision boundaries based on labeled examples and can accurately classify known attack types when trained on representative data.

Unsupervised learning approaches focus on anomaly detection by modeling normal traffic behavior and identifying deviations as potential intrusions. Techniques such as clustering and density-based methods are useful for detecting previously unseen attacks. However, unsupervised methods often suffer from high false-positive rates, particularly in cloud environments where traffic patterns are highly variable due to auto-scaling and workload migration.

Despite their advantages, individual machine learning models exhibit inherent limitations. Overfitting, sensitivity to noise, and bias toward majority classes can degrade detection performance. These issues are exacerbated in cloud environments characterized by heterogeneous workloads and rapidly changing traffic patterns. Consequently, relying on a single machine learning model is often insufficient for robust cloud intrusion detection, motivating the adoption of ensemble learning techniques.

VIII. ENSEMBLE LEARNING APPROACHES

Ensemble learning is a machine learning paradigm that combines multiple base models to improve predictive accuracy, robustness, and generalization. By integrating the outputs of diverse classifiers, ensemble methods mitigate the weaknesses of individual models and enhance overall detection performance.

Common ensemble techniques include bagging, boosting, voting, and stacking. Bagging-based methods, such as Random Forests, reduce variance by training multiple models on different subsets of the training data. Boosting-based techniques, including AdaBoost and Gradient Boosting, focus on misclassified instances and iteratively improve model performance. Voting-based ensembles aggregate predictions from heterogeneous classifiers using majority or weighted voting schemes, while stacking employs a meta-classifier to learn optimal combinations of base model outputs.

In the context of cloud-native intrusion detection, ensemble learning offers several advantages. The diversity of base models enables the detection system to handle heterogeneous traffic patterns and evolving attack strategies. Ensemble-based IDS typically achieve higher detection accuracy and lower false-positive rates compared to single-model approaches. These properties make ensemble learning particularly well suited for analyzing VPC Flow Logs in dynamic and large-scale cloud environments.

IX. CLOUD-NATIVE IDS ARCHITECTURE

A cloud-native intrusion detection system (IDS) built using VPC Flow Logs and ensemble learning follows a layered, modular architecture that aligns with cloud-native design principles such as scalability, elasticity, fault tolerance, and automation. Unlike traditional IDS deployments that rely on fixed sensors and centralized inspection points, cloud-native IDS architectures are designed to operate seamlessly within dynamic and distributed cloud infrastructures.

The log collection layer is responsible for continuously gathering VPC Flow Logs from cloud resources, including virtual machines, containers, and managed services. Flow logs can be collected at multiple levels—network interface, subnet,

DOI: 10.17148/IJARCCE.2026.151131

or VPC—allowing flexible visibility depending on organizational requirements. This layer integrates with cloud-native logging services to ensure reliable, low-latency ingestion of high-volume traffic metadata without impacting application performance.

The preprocessing and feature engineering layer performs data cleaning, normalization, encoding, and feature extraction on raw flow logs. This layer transforms raw metadata into structured feature vectors suitable for machine learning analysis. Temporal aggregation and sliding window techniques are often employed to capture time-dependent behavior, which is critical for detecting slow-rate attacks, reconnaissance activities, and distributed attacks.

The machine learning layer consists of multiple base classifiers trained in parallel using processed flow-level features. These classifiers may include decision tree-based models, linear classifiers, and boosting-based learners. Training and inference can be implemented using cloud-native machine learning frameworks, enabling horizontal scaling and efficient resource utilization. The outputs of individual models are then combined using ensemble learning strategies to produce final intrusion detection decisions.

The alerting and response layer integrates detection results with monitoring dashboards, security information and event management (SIEM) systems, and automated incident response workflows. Alerts can trigger predefined actions such as isolating compromised resources, updating firewall rules, or notifying security teams. This integration enables near real-time threat response while reducing manual intervention. Overall, the proposed architecture supports continuous monitoring, adaptive detection, and operational resilience in large-scale cloud environments.

X. DATASETS USED IN CLOUD INTRUSION DETECTION RESEARCH

The performance and reliability of intrusion detection systems depend heavily on the quality, diversity, and realism of datasets used for training and evaluation. Historically, IDS research has relied on benchmark datasets such as KDD Cup 99 and NSL-KDD, which were derived from simulated network environments. Although these datasets facilitated early IDS research, they suffer from outdated attack patterns, redundant records, and unrealistic traffic distributions, limiting their applicability to modern cloud environments.

To address these limitations, more recent datasets such as UNSW-NB15**,** CICIDS2017**,** CICIDS2018**,** and Bot-IoT have been introduced. These datasets incorporate a wider range of contemporary attack types, including brute-force attacks, denial-of-service, botnets, web-based exploits, and infiltration attempts. They provide richer feature sets and more realistic traffic behavior compared to earlier benchmarks, making them suitable for evaluating modern machine learning-based IDS.

Despite these improvements, publicly available datasets are typically generated in controlled laboratory environments and may not fully reflect the complexity, scale, and heterogeneity of real-world cloud traffic. Cloud environments exhibit unique characteristics such as elastic scaling, microservices communication, and multi-tenant workloads, which are difficult to replicate accurately in synthetic datasets.

As a result, cloud-specific intrusion detection research increasingly relies on synthetic datasets derived from VPC Flow Logs. These datasets capture realistic flow-level behavior from operational cloud networks and support large-scale experimentation. However, labeling cloud flow logs remains a major challenge, as ground truth information is rarely available. Researchers often employ a combination of rule-based labeling, attack simulation, and expert knowledge to annotate datasets. Addressing dataset realism and labeling challenges remains a critical research direction for advancing cloud-native intrusion detection.

XI. PERFORMANCE METRICS AND EVALUATION

Evaluating the effectiveness of intrusion detection systems requires the use of appropriate performance metrics that reflect real-world security requirements. Due to the highly imbalanced nature of intrusion detection datasets, accuracy alone is insufficient and may provide misleading results. Instead, a combination of metrics is used to assess detection performance comprehensively.

Commonly used metrics include precision**,** recall**,** and F1-score, which measure the system's ability to correctly identify malicious traffic while minimizing false alarms. The false-positive rate is particularly important in operational

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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

environments, as excessive false alerts can overwhelm security teams and reduce trust in the detection system. The receiver operating characteristic (ROC) curve and the area under the curve (AUC**)** are also widely used to evaluate the trade-off between detection rate and false-positive rate.

In cloud environments, detection latency and computational overhead are critical performance considerations. Intrusion detection systems must operate in near real time to prevent attack propagation and minimize damage. Ensemble-based IDS typically demonstrate improved detection accuracy and robustness compared to single-model approaches, but they may introduce additional computational cost. Therefore, performance evaluation must consider both detection effectiveness and system efficiency.

Empirical studies consistently show that ensemble learning approaches outperform individual classifiers across multiple evaluation metrics, particularly in terms of reducing false positives and improving generalization. These advantages make ensemble-based intrusion detection well suited for deployment in large-scale, dynamic cloud environments.

XII. SECURITY, PRIVACY, AND COMPLIANCE CONSIDERATIONS

Security monitoring in cloud environments must balance effective intrusion detection with privacy protection and regulatory compliance. Traditional intrusion detection techniques based on packet payload inspection can expose sensitive user data and may violate privacy regulations such as the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA).

Flow-based intrusion detection using VPC Flow Logs provides a privacy-preserving alternative by capturing only metadata rather than packet content. This approach significantly reduces the risk of sensitive data exposure while still enabling meaningful security analysis. By avoiding payload inspection, flow-based IDS align well with data minimization principles required by modern privacy regulations.

From a compliance perspective, organizations must ensure that flow logs are stored securely and protected against unauthorized access. Encryption, role-based access control, and secure key management are essential for safeguarding flow log data. Additionally, organizations must comply with cloud provider policies and industry security standards when collecting, storing, and processing network telemetry.

Privacy-aware intrusion detection systems that rely on metadata analysis are increasingly preferred in regulated environments. Ensemble-based cloud-native IDS using VPC Flow Logs offer a strong balance between detection effectiveness, operational scalability, and compliance with privacy and regulatory requirements.

XIII. CHALLENGES AND LIMITATIONS

Despite their demonstrated advantages, ensemble-based cloud-native intrusion detection systems (IDS) face several technical and operational challenges that limit their effectiveness in real-world deployments. One of the primary challenges is the limited availability of high-quality, labeled cloud datasets. Unlike traditional enterprise networks, cloud environments are highly dynamic and multi-tenant, making it difficult to collect representative datasets with accurate ground truth. The lack of standardized, publicly available cloud flow log datasets hampers reproducibility and comparative evaluation of proposed detection approaches.

Another significant limitation arises from the computational complexity associated with ensemble learning techniques. Training and maintaining multiple machine learning models in parallel requires substantial computational resources, particularly when operating at cloud scale. Real-time intrusion detection further increases system demands, as models must process large volumes of flow data with minimal latency. While cloud platforms provide elastic resources, inefficient model design can result in increased operational costs and delayed detection.

A fundamental constraint of flow-based intrusion detection is the lack of contextual information due to metadata-only analysis. VPC Flow Logs do not capture packet payloads, application-level semantics, or user intent, which can limit the ability to distinguish between benign and malicious activities in certain scenarios. As a result, some sophisticated attacks may evade detection or generate false positives, particularly when attack behavior closely resembles legitimate traffic patterns.

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 1, January 2026

DOI: 10.17148/IJARCCE.2026.151131

Additionally, cloud-native IDS must continuously adapt to evolving attack strategies and changing workloads. Static models trained on historical data may quickly become outdated as attackers modify their techniques or as cloud workloads evolve. Continuous model retraining and validation are necessary to maintain detection accuracy, but this introduces challenges related to model drift, retraining frequency, and operational complexity. Addressing these challenges remains an open research problem in cloud-native intrusion detection.

XIV. FUTURE RESEARCH DIRECTIONS

Future research in cloud-native intrusion detection is expected to focus on the development of advanced ensemble techniques that incorporate deep learning models**.** Deep neural networks, including recurrent and graph-based architectures, have the potential to capture complex temporal and structural relationships in network traffic. Combining deep learning models with traditional machine learning classifiers in ensemble frameworks may further improve detection accuracy and robustness against sophisticated attacks.

Another promising research direction involves the real-time stream processing of VPC Flow Logs**.** Most existing studies rely on offline analysis, which limits their applicability to operational environments. Integrating stream processing frameworks with cloud-native IDS can enable near real-time detection and response. Such systems must be designed to handle high-throughput data streams while maintaining low latency and high reliability.

Future work may also explore automated and intelligent response mechanisms that go beyond alert generation. Integrating intrusion detection systems with automated remediation tools, such as dynamic access control updates and resource isolation, can significantly reduce response time and limit attack impact. However, ensuring the correctness and safety of automated responses remains a critical challenge.

Finally, the emergence of multi-cloud and hybrid cloud environments presents new research opportunities. Intrusion detection frameworks capable of operating across multiple cloud providers and on-premise infrastructures can provide unified security visibility and policy enforcement. Standardized telemetry formats and interoperable detection models will be essential for enabling effective intrusion detection in increasingly heterogeneous cloud ecosystems.

XV. CONCLUSION

This study examined the role of cloud-native intrusion detection systems that utilize VPC Flow Logs and ensemble learning techniques to address the evolving security challenges of modern cloud infrastructures. Traditional intrusion detection approaches, which rely heavily on static signatures and deep packet inspection, are increasingly ineffective in cloud environments characterized by encryption, elasticity, and dynamic workloads. By leveraging flow-level telemetry, cloud-native IDS provide scalable and privacy-preserving monitoring capabilities that align with the architectural principles of cloud computing.

The analysis demonstrated that VPC Flow Logs serve as a powerful data source for intrusion detection by capturing rich metadata related to network communication without exposing packet payloads. When combined with effective feature extraction and preprocessing techniques, flow-level data enables the identification of diverse attack behaviors, including reconnaissance, denial-of-service, and unauthorized access attempts. The study further highlighted the importance of ensemble learning approaches in overcoming the limitations of individual machine learning models, particularly in handling noisy data, class imbalance, and heterogeneous traffic patterns.

Ensemble-based intrusion detection systems offer improved detection accuracy, reduced false-positive rates, and enhanced robustness against evolving threats. Their ability to integrate multiple classifiers allows for better generalization across varying cloud workloads and attack scenarios. Moreover, the proposed cloud-native IDS architecture supports scalability, fault tolerance, and real-time detection while maintaining compliance with privacy and regulatory requirements.

In conclusion, the integration of VPC Flow Logs with ensemble learning represents a practical and effective approach for securing cloud environments. While challenges related to dataset availability, computational overhead, and contextual limitations remain, ongoing advancements in machine learning, cloud analytics, and automation are expected to further enhance cloud-native intrusion detection capabilities. This study provides a strong foundation for future research and

DOI: 10.17148/IJARCCE.2026.151131

development in building intelligent, scalable, and privacy-aware security solutions for next-generation cloud infrastructures.

REFERENCES

[1]. K. Scarfone and P. Mell, *Guide to Intrusion Detection and Prevention Systems*, National Institute of Standards and Technology (NIST), 2007.
[2]. R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," IEEE Symposium on Security and Privacy, vol. 10, no. 4, pp. 305–316, 2010.
[3]. T. G. Dietterich, "Ensemble Methods in Machine Learning," Multiple Classifier Systems, Springer, vol. 1, no. 1, pp. 1–15, 2000.
[4]. Amazon Web Services, "VPC Flow Logs Documentation," AWS Whitepaper, 2020.
[5]. M. Roesch, "Snort: Lightweight Intrusion Detection for Networks," Proc. USENIX LISA Conf., vol. 1, no. 1, pp. 229–238, 1999.
[6]. J. Zhang, C. Chen and Y. Xiang, "Flow-Based Intrusion Detection in Cloud Environments," IEEE Access, vol. 7, no. 1, pp. 32124–32134, 2019.
[7]. I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," Proc. Int. Conf. Information Systems Security and Privacy, vol. 1, no. 1, pp. 108–116, 2018.
[8]. M. Ring, S. Wunderlich, D. Grüdl, D. Landes and A. Hotho, "A Survey of Network-Based Intrusion Detection Data Sets," Computers & Security, vol. 86, no. 1, pp. 147–167, 2019.
[9]. S. Mukkamala, G. Janoski and A. Sung, "Intrusion Detection Using Ensemble Learning," IEEE Int. Conf. Fuzzy Systems, vol. 1, no. 1, pp. 1–6, 2005.
[10]. A. Javaid, Q. Niyaz, W. Sun and M. Alam, "A Deep Learning Approach for Network Intrusion Detection System," IEEE Access, vol. 4, no. 1, pp. 213–226, 2016.
[11]. L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
[12]. Y. Freund and R. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119–139, 1997.
[13]. S. Axelsson, "Intrusion Detection Systems: A Survey," Technical Report, Chalmers University of Technology, vol. 1, no. 1, pp. 1–98, 2000.
[14]. N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," Military Communications and Information Systems Conf., vol. 1, no. 1, pp. 1–6, 2015.
[15]. Google Cloud Platform, "VPC Flow Logs Overview," GCP Documentation, 2021.
[16]. Microsoft Azure, "Network Security Group Flow Logs," Azure Documentation, 2021.
[17]. A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.
[18]. S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis and P. Chan, "Cost-Based Modeling for Fraud and Intrusion Detection," ACM Transactions on Information and System Security, vol. 3, no. 4, pp. 1–34, 2006.
[19]. P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández and E. Vázquez, "Anomaly-Based Network Intrusion Detection: Techniques, Systems and Challenges," Computers & Security, vol. 28, no. 1, pp. 18–28, 2009.
[20]. M. Conti, N. Dragoni and V. Lesyk, "A Survey of Man-in-the-Middle Attacks," IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2027–2051, 2016.
[21]. European Union Agency for Cybersecurity (ENISA), "Cloud Security Incident Reporting," ENISA Report, 2019.
[22]. ISO/IEC, "ISO/IEC 27017:2015 – Code of Practice for Information Security Controls Based on ISO/IEC 27002 for Cloud Services," International Organization for Standardization, 2015.
[23]. A. Patcha and J. M. Park, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," Computer Networks, vol. 51, no. 12, pp. 3448–3470, 2007.
[24]. S. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41–50, 2018.
[25]. R. Mitchell and I.-R. Chen, "Behavior Rule Specification-Based Intrusion Detection for Safety Critical Medical Cyber Physical Systems," IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 1, pp. 16–30, 2014.

