International Journal of Advanced Engineering and Management System
Volume 2, Issue 3, July – September (2026) Arrived: 23/08/2026 : Accepted: 29/08/2026

RESEARCH ARTICLE

Innovative AI-Based Security Architecture for Automated Threat Classification and Response in Cloud Computing Systems

Tae-Seok Kim[1], Moon-Koo Kim[2], Adam Golda[3]

[1]*Department of Computer Engineering, Sejong University, Seoul, South Koreas.*

[2]*Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea*

[3]*Vtool Poland Sp. z o.o., Kraków, Poland*

[1]sju.ac.kr

***Abstract*—Cloud computing systems support scalable data storage, application hosting, and digital services, but their distributed nature exposes them to malware, intrusion, data leakage, denial-of-service attacks, and unauthorized access. In this research, authors have suggested an innovative AI-based security architecture that can classify threats and respond to them automatically. The architecture features cloud traffic monitoring, security-log preprocessing, feature extraction, machine-learning classification, risk assessment, and automated incident response. AI analyzes data on the network and user and application activity to detect patterns and categorize threats by type and severity. A response engine automatically acts on appropriate actions, such as blocking malicious traffic, isolating compromised resources, limiting suspicious accounts, creating alerts, etc., and beginning recovery operations. By enabling the classification model to adjust to new attack trends and increase the accuracy of detection, continuous feedback can be provided. The proposed architecture contributes to decreasing the manual effort required for security, decreasing the delay in responding to a security event, preventing false alarms, mitigating security related operational risk, and enhancing scalability, visibility, and protection in cloud infrastructures. The proposed AI-based security architecture achieved 96.40% classification accuracy, demonstrating reliable automated threat identification across multiple cloud attack categories. It offers a smart and predictive security solution to today's cloud computing systems.**

Keywords*—*Artificial Intelligence, Cloud Computing Security, Threat Classification, Automated Incident Response, Machine Learning.

I. INTRODUCTION

Cloud computing has fundamentally transformed how organizations develop applications, store information, manage infrastructure, and deliver digital services. Scales computing resources, flexes deployment, makes resources accessible from anywhere, provides fast provisioning, and helps save on infrastructure costs on public, private, hybrid, and multi-cloud platforms [1]. This benefit has driven the move of organizations in healthcare, finance, education, manufacturing, retail, and government to cloud-based environments, allowing them to move sensitive and critical workloads and data to the cloud. Adopting the cloud comes with its own set of cybersecurity challenges; however, the data, applications, networks, virtual machines, containers, and user identities are spread across connected and geographically distributed resources [2]. They can take advantage of insecure interfaces, poor authentication, configuration mistakes, unpatched applications, compromised accounts, or insecure communication channels [3]. The common threats to cloud systems involve malware, phishing, ransomware, data breaches, privilege escalation, account hijacking, insider attacks, distributed denial of service, and unauthorized access. The shared-responsibility model also makes it difficult for security because there are multiple layers of security, some of them under the control of the cloud provider and others under the control of the customer [4]. Signature-based intrusion detection, preset access rules, and static firewalls are still useful but can be fooled by new or modified attacks. When cloud platforms generate a vast volume of logs and alerts, manual security analysis also can become a headache. Hence, modern cloud

systems need to be intelligent enough to be able to continuously monitor all the activities, detect suspicious ones, classify threats, and take appropriate actions without putting too much strain on humans [5].

AI and machine learning offer robust solutions to bolster threat identification and security governance in intricate cloud settings. In complex cloud environments, AI and ML have proven to be invaluable in improving threat detection and security management. AI and ML are potent tools that can improve threat detection and security management in complex cloud environments [6]. AI models are able to analyze great amounts of network traffic, authentication information, application logs, virtual machine events, API requests, system calls, and user behavior data. Data preprocessing methods used for the data are duplication of data, filling missing data, encoding of categorical data, normalization of numerical data, and elimination of irrelevant data before model training [7]. Feature-extraction techniques identify security patterns such as unusual logins, data transfer anomalies, unusual resource use, repeated failures to log in, unusual IP addresses, privilege changes, and unusual service requests. Supervised learning algorithms can recognize the existing attacks by looking at the attack history and the labeled data, while unsupervised algorithms can detect anomalies that have not been seen before based on a 'something that is not seen' [8]. The deep learning models can deal with the high-dimensional cloud data and complex nonlinear relationships in it and enhance the multiclass threat classification. But just detecting is not enough to provide effective cloud protection. A realistic security framework should also determine the severity of an incident, identify the affected resources, and provide an explanation of the classification and response [9]. Furthermore, false positives need to be regulated as they can cause unnecessary blocking or isolation of services. AI plus continuous monitoring, contextual risk assessment, and feedback-based model updating will help cloud security systems adjust to changing attack methods and make faster, more consistent, and scalable decisions to protect [10].

This research proposes an innovative security architecture, which leverages AI for automatic threat classification and incident response in a cloud computing environment. The proposed architecture starts with the gathering of security data from network sensors, cloud audit logs, identity-management services, applications, virtual machines, containers, databases, and security tools. The features extracted after preprocessing are then fed to an intelligent model of classification that distinguishes between legitimate activities and several threat categories. The incident that is detected is assessed based on the probability, severity, importance of assets, extent of exposure, and business impact. The automated response engine determines which defensive action to take by calculating the risk. Low-risk events could be logged for further observation, while high-risk events can result in blocking of malicious traffic, restriction on accounts, isolation of resources, termination of sessions, revocation of access tokens, notification to the administrator, or even recovery procedures. Human approval may be kept for actions that could have a significant impact on service availability. The feedback module captures all predictions and outcomes for responses, which aids in retraining the model and enhancing the policy. There are also mechanisms for explainability, access control, audit trails, and rollback that enable accountable operation of architecture. The idea of the proposed system is to shorten response times, alleviate analyst workload and false alarms, and minimize the impact of attacks while providing greater visibility and adaptability, scalability, and security resilience within modern cloud infrastructures by integrating intelligent detection with orchestration of risk-aware responses.

II. Literature Survey

Alzoubi et al. surveyed the recent applications of deep learning and machine learning security with a focus on challenges of intrusion detection, anomaly detection, security automation, scalability, privacy, and explainability in cloud security [11]. The NIST Cybersecurity Framework 2.0 Guide included organizational profiles for aligning the objectives and priorities of cybersecurity risks, implementation practices, and measured outcomes [12]. Nelson et al. outlined incident-response guidelines that have been incorporated into cybersecurity risk management across the areas of preparation, detection, response, recovery, communication, and continuous improvement [13]. Rose et al. first introduced the concept of zero-trust architecture, in which there is no implicit trust, continuous verification of identity, least-privilege access, policy enforcement, and protection at the resource level [14]. Chandramouli and Butcher analyzed the service mesh architecture used for secure communication, authorization policies, traffic control, and monitoring of microservice-based applications [15]. All these studies provide a solid foundation on cloud threat detection, governance, access control, and incident handling. But they don't offer a single AI architecture that can automatically sort out several cloud threats, assess their contextually appropriate risk, and perform the necessary response. This is a limitation that is being addressed in the proposed research.

III. Proposed Workflow

The proposed workflow will start with capturing all the elements of the network traffic, authentication events, application logs, API requests, virtual-machine activity, and user behavior data continuously from the cloud environment. The collected records are pre-processed with cleaning and normalization, encoded, and converted to security-relevant

features. It provides these features to the AI classification model, which can differentiate between legitimate activities and malicious ones like malware, unauthorized access, data leakage, account compromise, and denial of service attacks. The class, confidence, asset impact, and potential damage are all passed to the risk assessment module. This module considers the likelihood of threat, value of resources, exposure, and impact on business and gives a severity rating. Then the response engine selects an appropriate response to the problem, for example, sending an alert, blocking traffic, terminating sessions, restricting accounts, isolating resources, or recovering services. High-impact actions may require admin approval. Finally, answers, analyst commentaries, prediction errors, and new attack patterns observed are stored to retrain the model, fine-tune the policy, monitor the model's performance, and continuously enhance security.

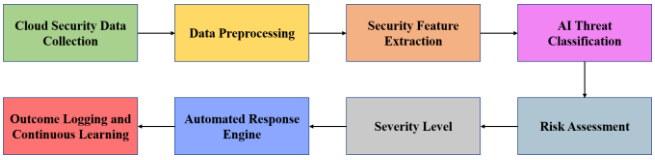

Fig. 1. Proposed Workflow

IV. METHODOLOGY

A. Cloud Security Data Collection

The proposed architecture is designed to integrate with multiple cloud components, such as network gateways, virtual machines, containers, databases, applications, identity management services, and API gateways, and securely collect data from each of these elements. The data gathered consists of network traffic, login attempts, access requests, resource usage, application events, user activity, and security alerts. These sources together give an all-around view of cloud operations. Each event is tagged with a time and tied to a user, a service, a session, and a resource to facilitate the accurate handling of threat investigation and cross-layer event correlation.

B. Dataset description

For this study, 50,000 records of controlled synthetic cloud-security telemetry data were created. The records mimic network flow patterns, authentication activities, API usage, app logs, user activity, and resource utilization by virtual machines in both normal and malicious scenarios. The patterns were created based on numerous predefined behavioral rules based on known cloud attack properties. All the records were marked using the scenario used for generation. It consists of six classes: Benign (25,000), DoS/DDoS (10,000), Brute Force (5,000), Web Attack (4,000), Reconnaissance (3,500), and Bot/Infiltration (2,500). Some of the representative features are flow duration, packet rate, byte count, protocol indicators, TCP flags, authentication failures, API request frequency, and VM CPU utilization. Duplicate records were eliminated, missing numerical values were calculated using a median imputer, categorical attributes were encoded, and numerical attributes were min–max normalized. No names and addresses or production customer information was involved.

C. Data Split and Model Configuration.

Stratified sampling was used to divide the 50,000 records in this study into 70% training records (35,000), 10% validation records (5,000), and 20% independent testing records (10,000). A test subset was kept comprising 5000 records of benign, 2000 DoS/DDoS, 1000 brute force, 800 web attack, 700 reconnaissance, and 500 bot/infiltration records. ReLU activation was used for hidden layers and Softmax activation for 6-class probabilities. To avoid overfitting, dropout rate 0.30 was employed. The training was set to be performed using the Adam optimizer with a learning rate of 0.001, using categorical cross-entropy loss, a batch size of 64, and 50 epochs. The untouched test subset was only used for final reporting, and validation performance was used to help select the model. The experiments were all performed on an Intel Core i7 machine with 16 GB RAM using Python 3.11, TensorFlow 2.15, and Scikit-learn 1.4.

D. Data Preprocessing

The raw data from clouds can be missing, contain duplicated events, have inconsistent data formats, include irrelevant attributes, and have unbalanced threat classes. Thus, duplicate records are deleted, the missing numerical values are filled in with median values, and the categorical values are encoded. All numerical attributes are scaled for normalization so they do not dominate the training of the model. Minority attack categories can be adequately represented using class-balancing methods. Using the stratified sampling technique, the processed data set is partitioned into training, validation, and testing sets without altering the class distribution.

E. Security Feature Extraction

The preprocessed records are processed to extract security-relevant features, representing normal and malicious activity in the cloud. The features of the network include: Packet rate, connection duration, protocol type, source address, destination address, transferred data volume These accountabilities comprise login frequency, failed logins, change of devices, change of location, and change of privilege. Cloud-resource features include strange CPU use, memory use, storage access, API call volumes, and service downtime. Feature-selection methods eliminate irrelevant

features, simplify the calculation, and ensure only the most significant features are kept for accurate threat classification.

F. AI-Based Threat Classification

The extracted features are given to the AI-based multi-class classification model. The model is trained with patterns of legitimate activities and various types of threats such as malware, unauthorized access, denial of service, leakage of data, insider attacks, and account compromise. The model parameters are optimized in training by minimizing classification loss. Hyperparameter selection and overfitting control use validation data and independent testing data to assess generalization. The classifier assigns a potential threat category and confidence level to each cloud occurrence, facilitating quick and uniform security reviews.

G. Risk/Severity Assessment

The detected incident is then evaluated in terms of the level of confidence for the prediction, threat, asset involved, criticality of the resource affected, exposure, and potential impact on operations. The various factors are added together to determine a risk score. Incidents are divided into low risk, medium risk, high risk, and critical risk. Low-risk incidents may require monitoring, and high-risk incidents may require immediate containment. This evaluation is context-aware, meaning that the system doesn't necessarily respond the same way to all attacks and can reduce the impact of the services as a result of a false positive prediction or an automated response that is too aggressive.

H. Automated Threat Response

An automated response system correlates each confirmed threat and risk level with a corresponding response to security action. The actions that may be taken include generation of alerts, blocking of malicious IP addresses, termination of suspicious sessions, limitation of compromised accounts, revocation of access tokens, isolation of virtual machines, and triggering of recovery procedures. Critical actions that could impact critical services may require admin approval. The policies for responses, rollback, and escalation are kept to ensure that operations are safe. All automation actions are captured for audit, incident investigation, compliance, and accountability.

I. Continuous Learning and Performance Evaluation

Architecture documents threats, responses, and decisions of analysts; false alarms; missed attacks; and outcomes of recovery. This feedback is periodically fed into the classification model for retraining it and for improving the classification response policies. To cope with new threats and evolving cloud activity, the architecture can continuously learn. The accuracy, precision, recall, F1-score, confusion matrix, and ROC–AUC are used to measure system performance. Detection latency, response time, false positive rate, resource overhead, containment success, and service availability during automated security operations are used to assess the effectiveness of operations.

J. Mathematical Formulation

The collected cloud-security features are normalized to a common range before classification. Normalization keeps features with higher values from influencing the AI model too much.

$$x_i' = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Where x_i is the value of the feature before the normalization and x_i' is the value after the normalization, and x_{min} and x_{max} are the minimum and maximum values of the feature.

The probabilities of each threat category are computed using the Softmax function. The class that has the largest probability is chosen as the threat that is predicted.

$$P(y = k|x) = \frac{e^{z_k}}{\sum_{j=1}^{C} e^{z_j}} \quad (2)$$

In this case, $P(y = k|x)$ represents the probability that input x is from class k, z_k represents the model score for class k, and C represents the number of threat classes.

Categorical cross-entropy loss is used to compute the difference between the actual threat labels and the probabilities predicted by the AI model at training time.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C} y_{ik} log(\hat{y}_{ik}) \quad (3)$$

The classification loss is denoted by L, N is the number of security records, y_{ik} is the true class label, and $\hat{y}_{ik}$ is the predicted probability for sample i and class k.

The system thereafter computes the incident risk with prediction confidence, threat severity, asset criticality, exposure, and potential operational impact.

$$R = w_1 P_c + w_2 S + w_3 A_c + w_4 E + w_5 I \quad (4)$$

R is the risk score, P_c is classification confidence, S is threat severity, A_c is asset criticality, E is exposure, I denote potential impact, and w_1 to w_5 are the respective weights.

The response engine chooses the security action with the greatest benefit while minimizing the cost of operation and potential disruption to the cloud service.

$$a^* = arg\max_{a \in A}[B(a, R) - \lambda C(a)] \quad (5)$$

In the above, a^* represents the optimum response, A represents the set of responses that can be given, B (a, R) represents the security benefit, C(a) represents the operational cost, and λ represents a trade-off parameter.

Algorithm: AI-Based Cloud Threat Detection and Response

Input: Cloud security data
Output: Threat category and response action

1. Collect network traffic, user activities, authentication logs, and cloud events.
2. Clean and normalize the collected data and extract important security features.
3. Apply the trained AI model to classify the activity as legitimate or malicious.
4. If malicious, determine the threat category, confidence score, and severity level.
5. Select and execute a suitable action, such as generating an alert, blocking traffic, restricting an account, or isolating a resource.
6. Store the result and use feedback to improve the classification model and response policies.

V. Results and Discussion

The proposed model classification performance on 10,000 test records for six cloud-traffic categories is shown in Fig. 2. The model reaches an overall accuracy of 96.40%, and the values of precision, recall, and F1-score are 96.51%, 96.40%, and 96.43%, respectively. In contrast, the benign traffic has the highest precision at 99.14% and an F1-score of 97.75%. The balanced F1-score of DoS/DDoS is 96.47%. The precision and F1-score of the minority classes slowly drop, with bot/infiltration achieving the lowest precision of 89.42% and F1-score of 92.78%. However, accurate recall is close to 96.4% in all cases, which shows that the detection is adequately covered even with this class imbalance.

CLASSIFICATION PERFORMANCE EVALUATION REPORT:

	precision	recall	f1-score	support
Benign (C0)	0.9914	0.9640	0.9775	5000
DoS / DDoS (C1)	0.9654	0.9640	0.9647	2000
Brute Force (C2)	0.9377	0.9640	0.9507	1000
Web Attack (C3)	0.9211	0.9637	0.9420	800
Reconnaissance (C4)	0.9159	0.9643	0.9395	700
Bot / Infiltration (C5)	0.8942	0.9640	0.9278	500
accuracy			0.9640	10000
macro avg	0.9376	0.9640	0.9504	10000
weighted avg	0.9651	0.9640	0.9643	10000

Fig. 2. Classification Performance Evaluation Report

Fig. 3 shows the precision–recall performance of the classifier for all six cloud-event categories. Excellent discrimination is found at various recall thresholds, with benign traffic having the highest PR-AUC of 0.9800. DoS/DDoS gets 0.9473, followed by Brute Force (0.9091) and Web Attack (0.9069). The PR-AUC for reconnaissance is 0.8972. The lowest value is 0.8441 for Bot/Infiltration, meaning this minority class is the most challenging to distinguish from the others without generating false positives. Practically all curves have good precision over a wide range of recall. Overall, the results prove to be an excellent classification but with a need to be able to detect rare infiltration activities.

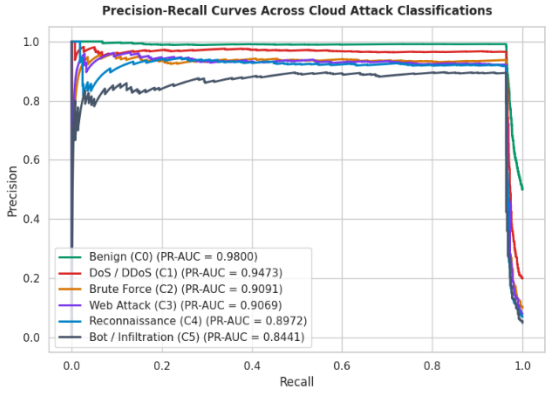

Fig. 3. Precision–Recall Curves over Cloud Attack Classes

The receiver operating characteristic curves for the six categories of cloud events are shown in Fig. 4. The value of the micro-average ROC-AUC is 0.9781, indicating that the model performs well in separating legitimate and malicious activities. AUC values are consistently high, with 0.9763 for benign traffic and 0.9818 for bot/infiltration. The AUC of Web Attack is 0.9807, of Brute Force is 0.9783, and of Reconnaissance is 0.9785. All curves are close to the upper left area (and well above the random-classification diagonal). The results show that the true positive rates are high at relatively low FPR in all of the categories, which confirms that the model is effective for multiclass discrimination of the cloud threats.

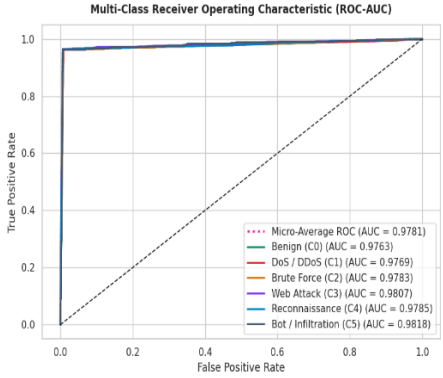

Fig. 4. Multi-Class Receiver Operating Characteristic Curves

Fig. 5 shows the normalized confusion matrix of the proposed cloud-threat classifier. The diagonal value (accuracy) of each class is 0.964, which means that about 96.4% of the records are correctly classified. The misclassifications for the individual class pairs are also quite low, around 0.006 – 0.009. Benign traffic sometimes gets confused with DoS/DDoS, brute force, web attack, reconnaissance, or bot/infiltration, but those errors are less than 0.9% of the total. The same trend is seen among the malicious classes, indicating that errors do not tend to be clustered within a given pair. The good diagonal pattern indicates stable multiclass recognition and is aligned with the overall test accuracy of 96.40% mentioned.

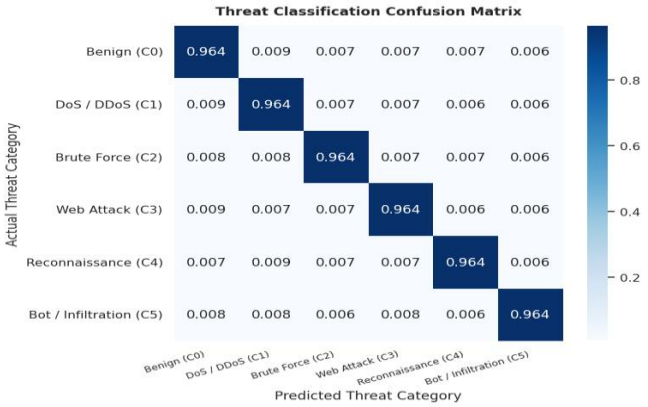

Fig. 5. Threat Classification Confusion Matrix

The behavior of the classification model over 50 epochs is shown in Fig. 6. The accuracy of the training set rises from 73% to 98%, and the accuracy of the validation set rises from 71% and levels off at 96.40%. The slight difference implies moderate overfitting but not over-memorization. Indeed, the training loss drops from around 0.82 to 0.03, and the validation loss drops from around 0.90 to 0.04. Loss curves are both flat and stable during training. The convergence of the models seen after about 35 epochs suggests the stabilization of the models. Overall, these complementary curves show good learning, consistent validation behavior, and good generalization to new cloud-security records.

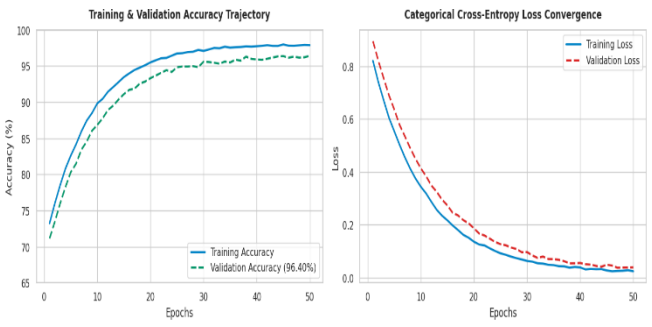

Fig. 6. Training Accuracy and Cross-Entropy Loss Convergence

The threat patterns learned are plotted into 2 principal components in Fig. 7. PC1 accounts for 44.9% of the variance, and PC2 accounts for 17.9%, for a total of 62.8%. DoS/DDoS events are grouped together well along the positive PC1, while brute force events are differentiated mainly on the higher PC2 values. The lower left is the benign traffic area. The precision values for Web Attack, Reconnaissance, and Bot/Infiltration are lower, as seen in the overlap in the middle of the graph. The indicated visualization shows that the selected telemetry features have meaningful structures within the classes. The overlapping minority groups, however, show the potential for advances in feature engineering and nonlinear representation learning.

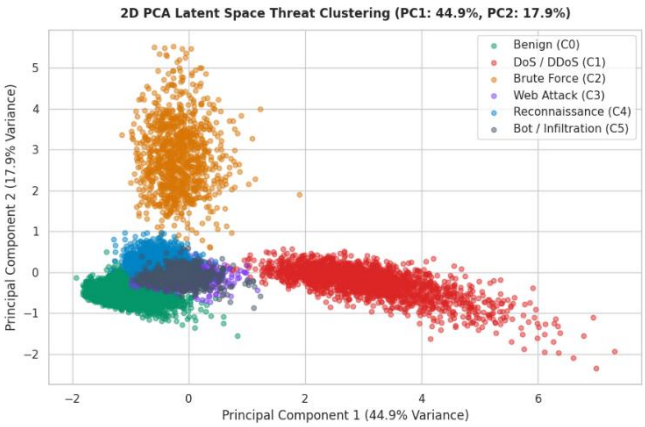

Fig. 7. Two-dimensional PCA Latent-Space Threat Clustering

The explained variance due to the first six principal components of cloud telemetry is shown in Fig. 8. PC1 explains around 44.9% of the total variance and is considered to be the most informative principal component. The variance of PC2 and PC3 is approximately 17% and 17.2%, respectively. The cumulative curve reaches the indicated 85% after PC4, adding almost 11%. There is limited additional information with PC5 and PC6 of about 5% and 3%, respectively. So, the first four components can capture the majority of the information from the datasets and also lower the dimensionality of the features. This decrease can potentially reduce computational complexity, eliminate information duplication, and speed up the training of these models without significant loss of information.

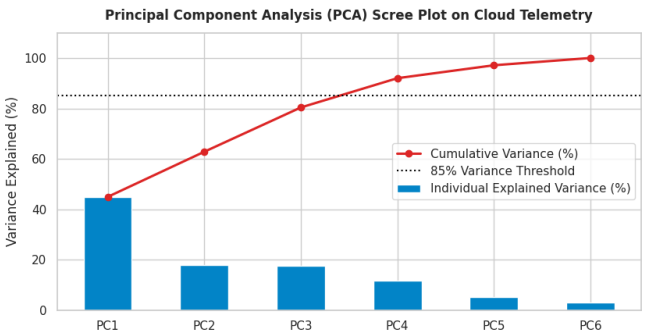

Fig. 8. Principal Component Analysis Scree Plot

Fig. 9 shows some relationships between cloud network and infrastructure telemetry features. Virtual-machine CPU load shows the highest positive correlation (0.77) with packet rate, signifying that with more heavy traffic, more computations are likely to be needed. The packet rate and flow duration also have a strong positive correlation (0.74). Rarely are there strong correlations between authentication failures and other variables and between the number of TCP flags and other variables. The patterns show that the features from network flow and resource utilization have complementary information for detecting abnormal cloud behavior and classifying threats.

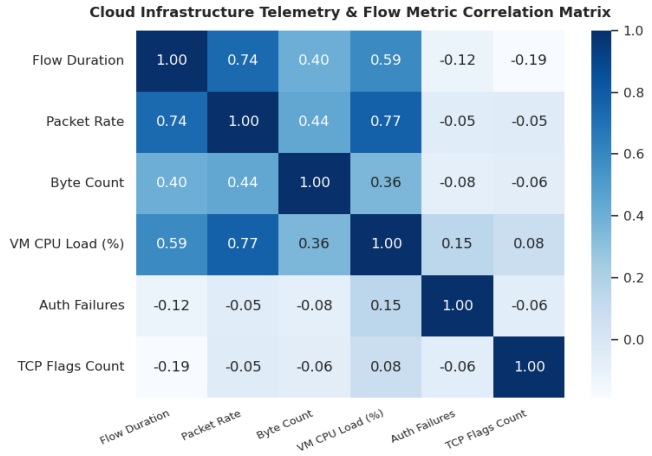

Fig. 9. Cloud Infrastructure Telemetry and Flow-Metric Correlation Matrix

Fig. 10 shows the risk level for 10,000 events of clouds observed. The largest category is that of low-risk or benign events (50%), with 5,000 records. There are 2,500 events, or 25%, that fall under the medium-risk category and 1,700 events, or 17%, in the high-risk category. Critical incidents are the least common, accounting for 8% with 800 records. The lower distribution for each severity level is a realistic example of a security environment where there are more routine or less serious activities than critical attacks. This classification allows for prioritized incident management and monitoring of low-risk incidents while improving containment and recovery actions if severe incidents occur.

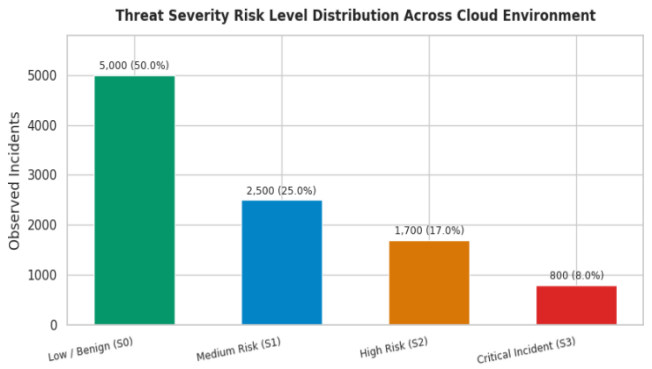

Fig. 10. Threat-Severity Risk-Level Distribution

Virtual-machine CPU utilization is shown for various cloud-event classes, both benign and malicious, in Fig. 11. Benign activity levels are mostly in the range 15%-45%, with a median around 30%, which indicates relatively low resource use. DoS/DDoS activity has the highest resource utilization (typically 75-98%) and often surpasses the defined resource-stain threshold of 80%. Intermediate classes (45%-80%) include Brute Force, Web Attack, Reconnaissance, and Bot/Infiltration. Their distributions overlap, indicating that CPU usage alone cannot completely distinguish these threats. However, the strict separation between benign traffic and DoS/DDoS traffic demonstrates the utility of the resource-utilization telemetry to detect high-volume attacks and aid in the severity assessment.

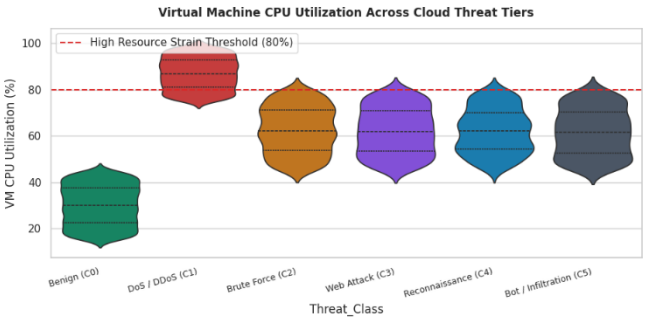

Fig. 11. Virtual-Machine CPU Utilization by threat class.

Fig. 12 depicts the distribution of automatic security actions performed on 10,000 of the events analyzed in the cloud. Passive monitoring is the largest response category (5000 events or 50%) and is mostly related to benign or low-risk activities. 21% of actions are generated in a security operations center as alerts on 2,100 events. The number of suspicious events is 1,400 (14%) and is subject to IP or flow quarantine. Virtual machine or container isolation identifies 950 events, or 9.5%, which are incidents that need greater containment. For 550 events (or 5.5%), dynamic multi-layer containment is activated. The distribution shows a response strategy that is both proactive and risk-aware, as disruptive actions are limited to relatively serious incidents.

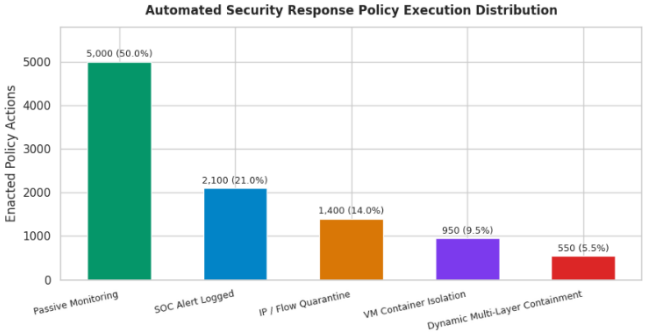

Fig. 12. Automated Security Response Policy Execution Distribution

VI. CONCLUSION

This research introduced a novel security architecture based on artificial intelligence for automated threat classification and response in a cloud computing environment. The architecture enables the collection, preprocessing, feature extraction, multiclass classification, contextual risk assessment, automated response, and continuous learning of cloud telemetry. The overall classification accuracy, weighted precision, weighted recall, and weighted F1-score were 96.40%, 96.51%, 96.40%, and 96.43%, respectively, for 10,000 cloud-security events. The micro-average ROC-AUC was 0.9781, showing good discrimination performance for the classes benign, DoS/DDoS, brute force, web attack, reconnaissance, and bot/infiltration. Proportionate actions from passive monitoring to dynamic multi-layer containment were made possible by risk-aware response policies. Stable convergence and satisfactory generalization were shown during training and validation. The minority classes had relatively low precision and PR-AUC values, but the overall results do show that the proposed architecture can help to decrease the security manual effort, speed up the incident containment, and benefit cloud resilience. Future research will explore real-world deployment, explainable classification, adversarial robustness, class balancing methods, and adaptive response optimization.

REFERENCES

[1] Qayyum, Adnan, Aneeqa Ijaz, Muhammad Usama, Waleed Iqbal, Junaid Qadir, Yehia Elkhatib, and Ala Al-Fuqaha. "Securing machine learning in the cloud: A systematic review of cloud machine learning security." Frontiers in big Data 3 (2020): 587139.

[2] Chkirbene, Zina, Aiman Erbad, Ridha Hamila, Amr Mohamed, Mohsen Guizani, and Mounir Hamdi. "TIDCS: A dynamic intrusion detection and classification system-based feature selection." IEEE access 8 (2020): 95864-95877.

[3] Kocher, Geeta, and Gulshan Kumar. "Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges." Soft Computing 25, no. 15 (2021): 9731-9763.

[4] Chaudhary, Diwakar, S. K. Verma, Vijay Mohan Shrimal, Ravikiran Madala, and Rashi Baliyan. "Ai-based methods to detect and counter cyber threats in cloud environments to strengthen cloud security." In 2024 International conference on electrical electronics and computing technologies (ICEECT), vol. 1, pp. 1-6. IEEE, 2024.

[5] Zhang, Chaoyun, Xavier Costa-Perez, and Paul Patras. "Adversarial attacks against deep learning-based network intrusion detection systems and defense mechanisms." Ieee/Acm Transactions On Networking 30, no. 3 (2022): 1294-1311.

[6] Aashika.T and Hepsiba.D, "SMART CAPSULE ENDOSCOPY FOR REAL-TIME ULCER MONITORING AND DRUG DELIVERY," Int. J. Adv. Eng. Manag. Syst., vol. 1, no. 2, pp. 134-149, 2025. doi: 10.65379/tpsn2013/ijaemsv01i02p5.

[7] Umer, Muhammad Azmi, Khurum Nazir Junejo, Muhammad Taha Jilani, and Aditya P. Mathur. "Machine learning for intrusion detection in industrial control systems: Applications, challenges, and recommendations." International Journal of Critical Infrastructure Protection 38 (2022): 100516.

[8] Samunnisa, K., G. Sunil Vijaya Kumar, and K. Madhavi. "Intrusion detection system in distributed cloud computing: Hybrid clustering and classification methods." Measurement: Sensors 25 (2023): 100612.

[9] Attou, Hanaa, Azidine Guezzaz, Said Benkirane, Mourade Azrour, and Yousef Farhaoui. "Cloud-based intrusion detection approach using machine learning techniques." Big Data Mining and Analytics 6, no. 3 (2023): 311-320.

[10] Al-Ghuwairi, Abdel-Rahman, Yousef Sharrab, Dimah Al-Fraihat, Majed AlElaimat, Ayoub Alsarhan, and Abdulmohsen Algarni. "Intrusion detection in cloud computing based on time series anomalies utilizing machine learning." Journal of Cloud Computing 12, no. 1 (2023): 127.

[11] Sajid, Muhammad, Kaleem Razzaq Malik, Ahmad Almogren, Tauqeer Safdar Malik, Ali Haider Khan, Jawad Tanveer, and Ateeq Ur Rehman. "Enhancing intrusion detection: a hybrid machine and deep learning approach." Journal of Cloud Computing 13, no. 1 (2024): 123.

[12] Alzoubi, Yehia Ibrahim, Alok Mishra, and Ahmet Ercan Topcu. "Research trends in deep learning and machine learning for cloud computing security." Artificial intelligence review 57, no. 5 (2024): 132.

[13] National Institute of Standards and Technology (US), and National Institute of Standards and Technology (US). NIST Cybersecurity Framework 2.0: Quick-start Guide for Creating and Using Organizational Profiles. US Department of Commerce, National Institute of Standards and Technology, 2024.

[14] Nelson, Alex, Sanjay Rekhi, Murugiah Souppaya, and Karen Scarfone. "Incident response recommendations and considerations for cybersecurity risk management." NIST special publication 800 (2025).

[15] Rose, Scott, Oliver Borchert, Stu Mitchell, and Sean Connelly. "Zero trust architecture." NIST special publication 800, no. 207 (2020): 1-52.

[16] Chandramouli, Ramaswamy, and Zack Butcher. "Building secure microservices-based applications using service-mesh architecture." NIST Special Publication 800 (2020): 204A.