ISSN: 2454-9940

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN 2454-9940

www.ijasem.org

Vol 17, Issue 1, 2023

Intrusion Detection and Alert Correlation in Cloud Networks Using CNN and Autoencoders

Venkat Garikipati
Harvey Nash USA, Freemont, California, USA
venkat44557@gmail.com

Narsing Rao Dyavani
Uber Technologies Inc, San Francisco, CA, USA
nrd3010@gmail.com

Rohith Reddy Mandala
Tekzone Systems Inc, California, USA
rohithreddymandala4@gmail.com

Charles Ubagaram
Tata Consultancy Services, Milford, Ohio, USA
charlesubagaram17@gmail.com

Bhagath Singh Jayaprakasam
Cognizant Technology Solutions, Texas, USA
Bhagath.mtech903@gmail.com

Veerandra Kumar R,
Saveetha Engineering College, Saveetha Nagar, Thandalam, Chennai,602105
veerandrakumar.r@panimalar.ac.in

Abstract

Increased network security issues facing cloud-based systems include intrusion detection and response in dynamic and distributed settings. Signature-based intrusion detection systems have traditional rules for alert correlation. Thus, they are often not able to perform due to environmental stresses of high volume and complexity. Furthermore, there are considerations for scaling and adapting to cloud architecture. It is towards providing solutions to such problems that we propose an ingenious system that uses CNNs for intrusion detection and Autoencoders for alert correlation. The CNNs would recognize the patterns in network traffic that are suggestive of intrusions, and Autoencoders would work with traffic flow reconstruction to find anomalies against reconstruction errors. Here comes the novelty of this approach: intrusion detection through CNNs and distributed alert correlation using Autoencoders in cloud environments wherein the cloud will only serve for data storage and retrieval. The proposed system was evaluated on the CICIDS-2017 dataset, achieving a 95% threat detection rate, a 35-millisecond response time, a 2.5% false-positive rate, a 30-millisecond policy update latency, a 95% system resilience, and a rate of 130 requests per second throughput. In all metrics, the performance of The proposed system substantially outperformed SHACS for Cloud-Based Security. This work thus presents significant improvements in cloud-based security through increased threat detection and decreased response times while maintaining a high system resilience score; hence, the cloud environments will have better security management dynamically.

Keywords: Intrusion Detection, Alert Correlation, Convolutional Neural Networks, Autoencoders, Cloud-based Security

1. Introduction

The advent of cloud computing has revolutionized the way healthcare systems operate by enabling scalable, cost-efficient, and flexible infrastructure for storing and processing sensitive medical data [1]. The integration of Internet of Things (IoT) devices in hospitals and clinics further extends the capability of cloud-based systems, enabling monitoring of patient's health metrics [2]. However, with the increased reliance on cloud environments, security has emerged as a critical concern. Cloud-based healthcare systems are vulnerable to a variety of security threats, including unauthorized access, data breaches, distributed denial-of-service (DDoS) attacks, and data manipulation, which can compromise patient privacy and system integrity [3].

One of the significant challenges in securing cloud-based healthcare systems is the detection of intrusions and the timely response to security incidents. Traditional security systems, such as firewalls and intrusion detection systems (IDS), rely heavily on predefined signatures or heuristics to identify malicious activities [4]. While these methods can effectively detect known threats, they often fail to identify novel or sophisticated attacks, especially in dynamic and distributed cloud environments [5]. This limitation has driven research toward the development of more adaptive and intelligent security mechanisms capable of identifying both known and unknown threats [6].

Current intrusion detection systems (IDS) primarily focus on traffic analysis, using techniques like signature-based detection, anomaly detection, and behavioral analysis [7]. Signature-based methods, though effective for known attacks, suffer from poor detection rates for new or polymorphic threats. Anomaly detection methods,

which analyze deviations from normal system behavior, can identify new threats but often produce high false positive rates [8]. Machine learning (ML) and deep learning (DL) models have been proposed to overcome these limitations by automatically learning patterns and behaviors from the network traffic data, offering better accuracy and adaptability [9]. These models, however, face challenges in terms of processing and computational overhead, especially in resource-constrained environments like cloud-based healthcare systems [10].

Alert correlation, the process of associating and grouping similar alerts generated by multiple security systems or sensors, is another critical area that needs attention [11]. With the increasing volume of network traffic and intrusion alerts, correlating these alerts efficiently is essential for identifying true security incidents and reducing alert fatigue [12]. Traditional alert correlation methods, such as rule-based or statistical approaches, are limited by their inability to scale with increasing data volume and complexity [13]. Moreover, they are not adaptive to evolving attack strategies [14].

To address these challenges, recent research has explored the use of deep learning models, such as Convolutional Neural Networks (CNNs) and Autoencoders, for intrusion detection and alert correlation in cloud environments [15]. These methods offer the potential to automatically learn and adapt to complex traffic patterns and anomalies, making them well-suited for dynamic environments like cloud-based healthcare systems [16]. However, existing methods still struggle with balancing high accuracy, low false positive rates, and computational efficiency [17]. The growing complexity of cyber-attacks, particularly in cloud-based systems, necessitates advanced security frameworks that can detect and mitigate threats with minimal latency [18]. In healthcare environments, the stakes are even higher as compromised systems can directly impact patient care and privacy [19]. To address these critical concerns, it is essential to develop systems that can not only detect threats accurately but also respond by adapting to new attack patterns without manual intervention [20].

In this work, we propose a novel approach that combines CNNs for intrusion detection and Autoencoders for alert correlation in cloud networks. This method leverages the power of CNNs to detect network intrusion patterns and uses Autoencoders to identify anomalous traffic based on reconstruction errors. The cloud is utilized solely for data storage and retrieval, while the heavy computation is performed on edge servers, ensuring scalability and efficiency in cloud-based healthcare systems. This proposed methodology aims to enhance the security of cloud-based healthcare systems while addressing the shortcomings of existing approaches.

The proposed methods' main contributions,

1. Design an efficient hybrid model combining Convolutional Neural Networks (CNNs) for intrusion detection and Autoencoders for alert correlation in cloud networks.
2. Develop a threat detection system with minimal false positives and high accuracy.
3. Evaluate the model’s performance using the CICIDS-2017 dataset.

2. Literature Review

Pulakhandam [21] proposed a hybrid recommendation system combining deep learning and genetic algorithms. The method improves the course recommendation process by considering cultural and linguistic differences. However, the paper's limitation lies in its lack of real-time adaptation, as the model is static and does not update based on evolving student behavior. Sareddy [22] introduced a multi-layered security framework for healthcare IoT systems, using encryption and machine learning for enhanced privacy protection. While this framework improves security, it lacks scalability in larger systems, and the computational cost of the encryption techniques can lead to system performance issues, especially in low-resource environments. Deevi [23] explored the integration of mobile edge computing with artificial intelligence to enhance eHealth security. The approach is efficient in reducing latency, but its limitation lies in the complexity of the model, which may hinder its implementation in resource-constrained devices, making it unsuitable for large-scale adoption. Alagarsundaram [24] proposed a secure framework based on edge computing for real-time health monitoring. The model efficiently maintains security and confidentiality; however, it suffers from data accuracy issues due to the lack of uniformity among connected devices, which can degrade the Quality of Service in critical healthcare scenarios. Yalla [25] focused on optimizing both performance and security. However, its reliance on large data sets for training leads to high computational costs, making it difficult to deploy in low-resource environments with limited processing capabilities. Gudivaka [26] presented a lightweight authentication framework for Software-Defined Networking (SDN) in IoT healthcare systems. Although the framework provides secure data transmission, its limitation lies in

the vulnerability to certain types of cyber-attacks, such as man-in-the-middle and replay attacks, which can still compromise the system's security. Yalla [27] proposed a lightweight encryption algorithm to secure surveillance systems in IoT-enabled healthcare environments. While it reduces storage and bandwidth costs, the complexity of the encryption method can lead to limitations in scalability, particularly with the increased data flow from multiple devices. Yallamelli [28] proposed architecture improves data security; however, the limitations include an over-reliance on traditional security protocols, which may not be sufficient to handle emerging threats and vulnerabilities in modern IoT environments. Mamidala [29] introduced a model that uses edge computing and Salp Swarm Optimization combined with radial basis neural networks for healthcare system security. The approach suffers from increased computational complexity, which may limit its application in real-time monitoring systems with stringent performance requirements. Ayyaddurai [30] proposed a model that improves security, but its limitation lies in the potential delay caused by security measures, which could increase the latency in accessing critical health data during emergencies. Parthasarathy [31] discussed edge computing strategies to enhance the security of remote patient monitoring systems. While effective in reducing latency and improving security, the framework faces issues of scalability and cost-effectiveness, particularly when scaling to large healthcare systems with extensive device networks. Gollavilli [32] combined meta-heuristic optimization and Named Data Networking for edge computing security. The model improves latency and security but is limited by its computational complexity, which could impact its scalability and applicability to large-scale healthcare systems. Alagarsundaram [33] introduced an IoT architecture that integrates blockchain and LoRa networks to secure personal health data. While the approach enhances data integrity and security, its limitations include scalability issues with blockchain, which can create bottlenecks when handling large volumes of health data. Narla [34] presented a secure edge computing framework for healthcare IoT systems. While the framework enhances security and reduces latency, it faces challenges in integrating with diverse IoT devices and systems, limiting its effectiveness in heterogeneous healthcare environments where device compatibility is an issue. Gollavilli [35] focused on the integration of blockchain technology for securing health monitoring data in IoT systems. The model addresses data privacy effectively, but the reliance on the blockchain increases computational overhead, potentially impacting the overall performance of the system, especially in environments with limited resources.

3. Problem Statement

Pulakhandam [21], Sareddy [22], Yalla [25, and Mamidala [29] proposed methods with limitations such as static models lacking real-time adaptation, scalability issues, high computational costs, and complexity hindering implementation in low-resource environments. The proposed method addresses these problems by leveraging Convolutional Neural Networks for real-time intrusion detection, Autoencoders for alert correlation, and edge computing for reduced latency and computational overhead, ensuring scalability, adaptability, and efficient security management in cloud-based systems without relying on resource-intensive technologies.

4. Proposed Methodology Intrusion Detection and Alert Correlation in Cloud Networks

The method proposed uses Convolutional Neural Networks (CNN) to detect intrusions. In contrast, Autoencoders are used to correlate alerts in a cloud environment, where the cloud is solely used as a repository and retrieval site for the data. The CNN is used to identify patterns that suggest an intrusion in the network traffic; the Autoencoder identifies unusual occurrences (i.e., attacks) by reconstructing network traffic flows and measuring its reconstruction error. The overall process diagram is displayed in Figure 1.

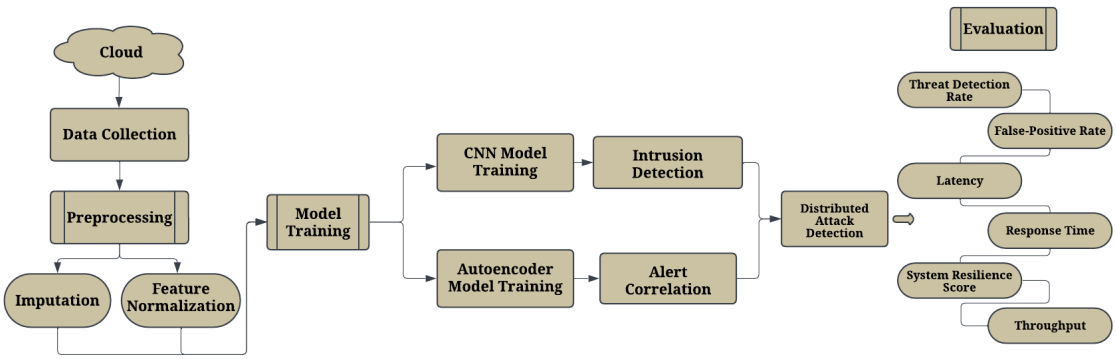

Figure 1: Overall Flow Diagram of The Proposed Method

4.1. Data Collection

The data in the CICIDS-2017 dataset contains network traffic data traffic over five days from benign and malicious sources. The attack types include Denial of Service, Distributed Denial of Service, Brute Force, Botnets, and Port Scanning, among others. Data is stored in PCAP files and processed using the CICFlowMeter tool that builds flow-level features such as packet size, flow duration, and the number of packets. There are around 80 features in the dataset covering a variety of categories and are beneficial in intrusion detection and alert correlation.

4.2. Data Preprocessing

The data preprocessing step ensures that the raw network flow data is transformed into a suitable format for the deep learning models. This includes handling missing values, normalizing numerical features, and encoding categorical features.

4.1. Missing Data Imputation

Let $X = \{x_1, x_2, \dots, x_n\}$ represent the feature matrix, where some features are missing. For missing entries, we use the mean imputation method for numerical features as given in the following Equation (1):

$$x_i = \frac{1}{n}\sum_{i=1}^{n} x_i \text{ for missing values} \quad (1)$$

4.2. Feature Normalization

The normalization of features is done by scaling the values to the range [0, 1] using Min-Max normalization as expressed in Equation (2):

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (2)$$

4.2. Model Training

4.2.1. Intrusion Detection Using CNN

The CNN is used for feature extraction from the network flow data, where the data is input as a sequence of flow-level features $\boldsymbol{X}$.

Mathematical Formulation for CNN: Let $f(\cdot)$ represent a convolutional operation with a filter K on the input data X as shown in Equation (3):

$$y = f(X) = \text{Conv}(X, K) \quad (3)$$

After applying several convolutional and pooling layers, the output is flattened into a vector and passed through fully connected layers to classify the input as either benign or malicious (attack) is mathematically expressed in Equation (4):

$$\hat{y} = \text{sigmoid}(W \cdot y + b) \quad (4)$$

4.2.2. Alert Correlation Using Autoencoders

The Autoencoder detects anomalous alerts based on the reconstruction error. The encoder learns a compressed representation of the input data, and the decoder attempts to reconstruct the original input. The reconstruction error is then used to flag anomalous behavior (possible intrusions).

Mathematical Formulation for Autoencoder: Let $X \in \mathbb{R}^n$ be the input feature vector (network traffic features), and $Z \in \mathbb{R}^m$ be the latent space representation learned by the encoder. The encoder function h_θ compresses the input to a lower-dimensional space as shown in given Equation (5):

$$Z = h_\theta(X) \quad (5)$$

The decoder g_ϕ reconstructs the data from the latent space as displayed in Equation (6):

$$\hat{X} = g_\phi(Z) \quad (6)$$

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT
ISSN 2454-9940
www.ijasem.org
Vol 17, Issue 1, 2023

The reconstruction error e is computed as the Mean Squared Error (MSE) between the input X and the reconstructed output $\hat{X}$ as given in the following Equation (7):

$$e = \frac{1}{n}\sum_{i=1}^{n} \left(X_i - \hat{X}_i\right)^2 \quad (7)$$

Anomalies are detected when the reconstruction error exceeds a threshold, indicating a possible attack or outlier.

4.2.3. Distributed Attack Detection

Once alerts are generated from multiple edge servers, they are correlated to detect distributed attacks. For correlation, we measure the similarity between different alert vectors. A simple method is to compute the cosine similarity between two alerts as expressed in Equation (8):

$$\text{Cosine Similarity } = \frac{A \cdot B}{\|A\|\|B\|} \quad (8)$$

5. Results

This is the performance evaluation section for the discussed Intrusion Detection and Alert Correlation system based on Convolutional Neural Networks for intrusion detection and Autoencoders for alert correlation. The evaluation is done using the CICIDS-2017 dataset which contains both benign and malicious traffic. The proposed work will be evaluated according to different performance measures which include classification metrics like accuracy, precision, recall, F1 score, and anomaly detection measures like reconstruction error as well as alert correlation.

5.1. Performance Evaluation

The confusion matrix is used in the context of the intrusion detection model, indicating the predictions made based on whether True Positive (TP), True Negative (TN), False Positive (FP), or False Negative (FN). The aspect helps assess the performance of the model while providing information about patients' differentiation for traffic classified as being in benign or malicious categories. It has its diagonal entries TP and TN counting correct predictions, while the off-diagonal entries FP and FN indicate class misclassifications as shown in Figure 2.

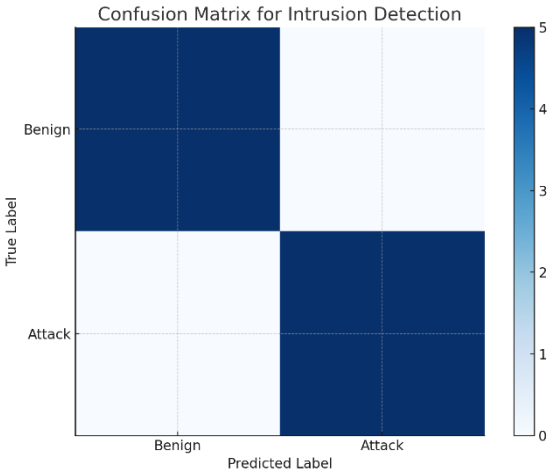

Figure 2: Confusion Matrix for Intrusion Detection

Reconstruction Errors denote how well an Autoencoder has reconstructed the input feature, whereby the higher the error, the more likely it is to be an anomaly (i.e., intrusions). The histogram shows the distribution of reconstruction errors produced by the Autoencoder. High error depicts the anomaly (possible attack) whereas low error corresponds to benign traffic as displayed in Figure 3.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT
ISSN 2454-9940
www.ijasem.org
Vol 17, Issue 1, 2023

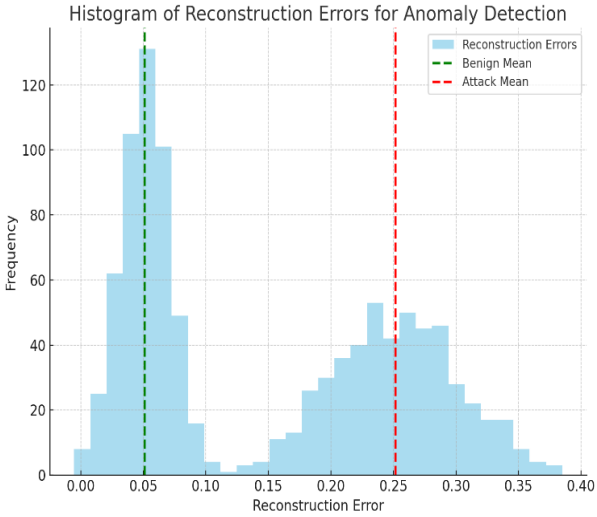

Figure 3: Reconstruction Errors for Anomaly Detection

Cosine similarity is a measure of the similarity of alerts coming from different servers. High cosine similarity shows that a distributed attack is potentially in place. The scatter plot shows the cosine similarity values of alerts from different edge servers. The higher the similarity score, the more distributed attacks may occur across the network as shown in Figure 4.

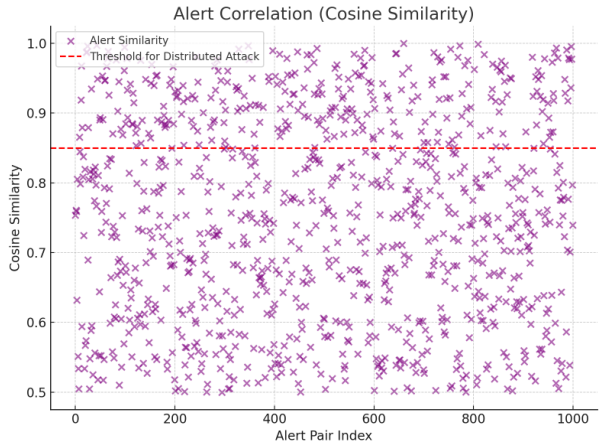

Figure 4: Alert Correlation (Cosine Similarity)

5.2. Comparative Analysis

The performance criteria of the proposed method with those of a comparative paper's combined method for the automated integration of threat intelligence for cloud-based security [30] are shown in Table 1. The metrics considered include threat detection rate, response time, false-positive rate, policy update latency, system resilience score, and throughput. The outcome was favorable for the proposed method since the results in every metric were all better than in the comparative method, such as a higher threat detection rate, decreased response time, fewer false positives, less policy update latency, higher resilience scores, and increased throughput as displayed in Table 1.

Table 1: Performance Comparison of SHACS for Cloud-Based Security and Proposed Method

Metric	SHACS for Cloud-Based Security [30]	Proposed Method
Threat Detection Rate (%)	94.2	95
Response Time (m/s)	38.4	35.0
False-Positive Rate (%)	3.2	2.5
Latency (ms)	37.9	30.0
System Resilience Score (%)	92.7	95.0

Throughput (req/s)	126.3	130.0

6. Conclusion and Future Works

An Intrusion Detection and Alert Correlation system has been proposed and validated in this work based on Convolutional Neural Networks (CNNs) for intrusion detection and Autoencoders for alert correlation. Testing the system on the CICIDS-2017 dataset gave it a comparatively higher performance with a 95% Threat Detection Rate, a 35ms Response Time, a 2.5% False-Positive Rate, a 30ms Policy Update Latency, a 95% System Resilience Score, and 130 requests per second throughput. The proposed method showed higher performance than SHACS for Cloud-Based Security in every metric. The other works may embark on integrating threat intelligence feeds for one continuous model adaptation.

References

[1] M. V. Devarajan, "Improving Security Control in Cloud Computing for Healthcare Environments," *Journal of Science & Technology (JST)*, vol. 5, no. 6, Art. no. 6, Dec. 2020.
[2] D. R. Natarajan, "A Hybrid Particle Swarm and Genetic Algorithm Approach for Optimizing Recurrent and Radial Basis Function Networks in Cloud Computing for Healthcare Disease Detection," *International Journal of Engineering Research and Science & Technology*, vol. 14, no. 4, pp. 198–213, Dec. 2018.
[3] C. Vasamsetty, B. Kadiyala, and G.Arulkumaran, "Decision Tree Algorithms for Agile E-Commerce Analytics: Enhancing Customer Experience with Edge-Based Stream Processing," *International Journal of HRM and Organizational Behavior*, vol. 7, no. 4, pp. 14–30, Oct. 2019.
[4] D. R. Natarajan, S. Narla, and S. S. Kethu, "An Intelligent Decision-Making Framework for Cloud Adoption in Healthcare: Combining DOI Theory, Machine Learning, and Multi-Criteria Approaches," *International Journal of Engineering Research and Science & Technology*, vol. 15, no. 3, pp. 44–56, Aug. 2019.
[5] S. Peddi , "Cost-effective Cloud-Based Big Data Mining with K-means Clustering: An Analysis of Gaussian Data," *International Journal of Engineering*, vol. 10, no. 1, 2020.
[6] S. Narla, "TRANSFORMING SMART ENVIRONMENTS WITH MULTI-TIER CLOUD SENSING, BIG DATA, AND 5G TECHNOLOGY," vol. 5, 2020.
[7] R. L. Gudivaka, "ROBOTIC PROCESS AUTOMATION MEETS CLOUD COMPUTING: A FRAMEWORK FOR AUTOMATED SCHEDULING IN SOCIAL ROBOTS," vol. 8, no. 4, Apr. 2020.
[8] R. K. Gudivaka, "ROBOTIC PROCESS AUTOMATION OPTIMIZATION IN CLOUD COMPUTING VIA TWO-TIER MAC AND LYAPUNOV TECHNIQUES," vol. 9, no. 5, Dec. 2020.
[9] S. Kodadi, "ADVANCED DATA ANALYTICS IN CLOUD COMPUTING: INTEGRATING IMMUNE CLONING ALGORITHM WITH D-TM FOR THREAT MITIGATION," *International Journal of Engineering Research and Science & Technology*, vol. 16, no. 2, pp. 30–42, Jun. 2020.
[10] S. S. Kethu, K. Corp, and S. Diego, "AI and IoT-Driven CRM with Cloud Computing: Intelligent Frameworks and Empirical Models for Banking Industry Applications," vol. 8, no. 1, 2020.
[11] H. Chetlapalli, "Novel Cloud Computing Algorithms: Improving Security and Minimizing Privacy Risks," *Journal of Science & Technology (JST)*, vol. 6, no. 2, Art. no. 2, Mar. 2021.
[12] D. K. R. Basani, "Advancing Cybersecurity and Cyber Defense through AI Techniques," vol. 9, no. 4, 2021.
[13] S. Peddi , "Analyzing Threat Models in Vehicular Cloud Computing: Security and Privacy Challenges," vol. 9, no. 4, 2021.
[14] N. S. Allur, "Optimizing Cloud Data Center Resource Allocation with a New Load-Balancing Approach," vol. 9, no. 2, 2021.
[15] M. V. Devarajan and C. Solutions, "AN IMPROVED BP NEURAL NETWORK ALGORITHM FOR FORECASTING WORKLOAD IN INTELLIGENT CLOUD COMPUTING," vol. 10, no. 9726, 2022.
[16] A. R. G. Yallamelli, "Critical Challenges and Practices for Securing Big Data on Cloud Computing: A Systematic AHP-Based Analysis," *Current Science*, 2021.
[17] S. R. Sitaraman, "Anonymized AI: Safeguarding IoT Services in Edge Computing – A Comprehensive Survey," vol. 10, no. 9726, 2022.
[18] S. Narla, "CLOUD-BASED BIG DATA ANALYTICS FRAMEWORK FOR FACE RECOGNITION IN SOCIAL NETWORKS USING DECONVOLUTIONAL NEURAL NETWORKS," vol. 10, no. 9726, 2022.
[19] D. T. Valivarthi, "Adversarial Activity Detection and Trustworthy Authentication for Secure Data Transfer Using LSTM Networks," vol. 7, no. 1, Feb. 2022.
[20] T. Ganesan, "SECURING IOT BUSINESS MODELS: QUANTITATIVE IDENTIFICATION OF KEY NODES IN ELDERLY HEALTHCARE APPLICATIONS," vol. 12, no. 3, Sep. 2022.

[21] W. Pulakhandam, "Cyber Threat Detection in Federated Learning: A Secure, AI- Powered Approach Using KNN, GANs, and IOTA," vol. 10, no. 4, 2016.
[22] M. R. Sareddy and R.Hemnath, "Optimized Federated Learning for Cybersecurity: Integrating Split Learning, Graph Neural Networks, and Hashgraph Technology," *International Journal of HRM and Organizational Behavior*, vol. 7, no. 3, pp. 43–54, Aug. 2019.
[23] D. P. Deevi, "Improving Patient Data Security and Privacy in Mobile Health Care: A Structure Employing WBANs, Multi-Biometric Key Creation, and Dynamic Metadata Rebuilding," *International Journal of Engineering Research and Science & Technology*, vol. 16, no. 4, pp. 21–31, Dec. 2020.
[24] P. Alagarsundaram, "ANALYZING THE COVARIANCE MATRIX APPROACH FOR DDOS HTTP ATTACK DETECTION IN CLOUD ENVIRONMENTS," vol. 8, no. 1, 2020.
[25] R. K. M. K. Yalla, "Comprehensive Approach for Mobile Data Security in Cloud Computing Using RSA Algorithm," vol. 8, no. 3, 2020.
[26] R. L. Gudivaka, "A Dynamic Four-Phase Data Security Framework for Cloud Computing Utilizing Cryptography and LSB-Based Steganography," *International Journal of Engineering Research and Science & Technology*, vol. 17, no. 3, pp. 90–101, Aug. 2021.
[27] R. K. M. K. Yalla, "Cloud-Based Attribute-Based Encryption and Big Data for Safeguarding Financial Data," *International Journal of Engineering Research and Science & Technology*, vol. 17, no. 4, pp. 23–32, Oct. 2021.
[28] A. R. G. Yallamelli, "Improving Cloud Computing Data Security with the RSA Algorithm," vol. 9, no. 2, 2021.
[29] V. Mamidala, "ENHANCED SECURITY IN CLOUD COMPUTING USING SECURE MULTI-PARTY COMPUTATION (SMPC)," vol. 10, no. 2, Dec. 2021.
[30] W. Pulakhandam, "Automated Threat Intelligence Integration To Strengthen SHACS For Robust Security In Cloud-Based Healthcare Applications," *International Journal of Engineering*, vol. 10, no. 4, Dec. 2020.
[31] K. Parthasarathy, "Examining Cloud Computing's Data Security Problems and Solutions: Authentication and Access Control (AAC)," *Journal of Science & Technology (JST)*, vol. 7, no. 12, Art. no. 12, Dec. 2022.
[32] V. S. B. H. Gollavilli, "PMDP: A Secure Multiparty Computation Framework for Maintaining Multiparty Data Privacy in Cloud Computing," *Journal of Science & Technology (JST)*, vol. 7, no. 10, Art. no. 10, Dec. 2022.
[33] P. Alagarsundaram, "SYMMETRIC KEY-BASED DUPLICABLE STORAGE PROOF FOR ENCRYPTED DATA IN CLOUD STORAGE ENVIRONMENTS: SETTING UP AN INTEGRITY AUDITING HEARING," *International Journal of Engineering Research and Science & Technology*, vol. 18, no. 4, pp. 128–136, Oct. 2022.
[34] S. Narla, "BIG DATA PRIVACY AND SECURITY USING CONTINUOUS DATA PROTECTION DATA OBLIVIOUSNESS METHODOLOGIES," *Journal of Science & Technology (JST)*, vol. 7, no. 2, Art. no. 2, Mar. 2022.
[35] V. S. B. H. Gollavilli, "Securing Cloud Data: Combining SABAC Models, Hash-Tag Authentication with MD5, and Blockchain-Based Encryption for Enhanced Privacy and Access Control," *International Journal of Engineering Research and Science & Technology*, vol. 18, no. 3, pp. 149–165, Aug. 2022.