Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

Machine Learning-Based Intrusion Detection for Cloud Network Traffic using Feature Engineering

Dr. D. Jayalakshmi
Computer Science and Engineering
RMD Engineering College
Kavarapettai, India

Pooja Swaminathan
Computer Science and Engineering
RMD Engineering College
Kavarapettai, India

Nethra Suresh
Computer Science and Engineering
RMD Engineering College
Kavarapettai, India

Abstract - Cloud computing is an indispensable platform for providing computing resources and services via the Internet. Even when the cloud environment is growing, there is still a possibility of cyberattacks.The classical IDSs are based on rules and are not very effective against emerging attacks. To overcome this, this paper proposes an intrusion detection method based on machine learning technology to detect cloud network intrusion. Network behavior is represented by the NSL-KDD dataset and traffic is classified as normal or attack. Data is preprocessed for model training using several techniques such as label encoding and feature scaling. Furthermore, feature engineering methods are used to create meaningful traffic related features that enhance the capacity of the models, including total traffic volume, byte ratio, duration category and source and destination byte percentages. There are two machine learning algorithms implemented and evaluated, namely Random Forest and Logistic Regression. The experimental results indicate that the accuracy of the proposed model based on Random Forest is around 78.1%, which is higher than that of the model based on Logistic Regression (74.8%). The findings suggest that by leveraging feature engineering and machine learning, it is possible to successfully classify normal and malicious traffic and that it can be a useful tool for improving security in cloud computing. The proposed method offers a solid base to build an intelligent Intrusion Detection System (IDS) for the modern cloud network.

Keywords - Cloud Computing, Intrusion Detection System, Machine Learning, Feature Engineering, Random Forest, Logistic Regression, NSL-KDD Dataset, Cloud Network Security.

I. INTRODUCTION

Cloud computing is one of the widely used technologies which offers computing resources and services over Internet. It provides several benefits, including scalability, flexibility, and cost savings on infrastructure, leading organizations to consider cloud-based solutions for sectors like healthcare, education, business, and data analytics. Even with these advantages, there are also several security issues being raised with the rise of cloud usage. Cloud environments are made available via networks, and are utilized by a number of users, meaning they can be subject to various cyberattacks.

In cloud security, the detection of malicious actions in network traffic is one of the biggest issues. Denial-of-service attack, probing attack, unauthorized access, and data-breach attacks are examples of attacks that attackers can launch, and which can impact the availability and reliability of cloud services. The traditional IDSs are mostly trained using a set of rules and signature of attacks. These systems are effective in recognising attacks that have been seen before, but are unable to detect new and emerging attacks. Furthermore, keeping rules consistent and up-to-date is challenging and difficult in cloud environments with the high volume of traffic.

Addressing these challenges, AI and ML solutions have emerged as a hot topic in cybersecurity. Machine learning models can be trained with historical network traffic data, and classify incoming network traffic as normal or malicious. These techniques offer greater flexibility, and can identify attack patterns that are not known and thus cannot be detected with traditional techniques.

The performance of machine learning models largely depends on the quality of the input data. Classification models may be less effective if they include only a portion of the network traffic data and it contains redundant or less informative features. Thus, Feature Engineering is a significant aspect of enhancing model performance. The learning ability of the model can be improved by adding more features related to traffic, which will lead to a more accurate intrusion detection result.

The NSL-KDD dataset is used to model the behavior of cloud network traffic in this work. The data is collected both normal and attack traffic, so supervised machine learning methods can be used. The data is preprocessed using various methods, such as label encoding and feature scaling. Additionally, features are engineered to create extra features like total traffic volume, byte ratio, duration category, and source and destination byte percentages.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

Random Forest and Logistic Regression are two machine learning algorithms that are implemented and evaluated for intrusion detection. Experimental results show that the Random Forest model performs better than Logistic Regression model to classify normal traffic and attack traffic. The results prove that by using machine learning methods in conjunction with feature engineering, the intrusion detection capability can be enhanced and help to improve the security in the cloud.

II. RELATED WORKS

The paper by Goce Stevanoksi et.al,[1] starts from the claim that, anomaly detection in network traffic can improve network security as IDS depends on separating benign from malicious behaviour while it also stresses that detecting it will be hard as the “normal” behaviour is difficult to define and anomalies are best understood as patterns. The primary point is that machine learning is useful for IDS because it can operate with little human intervention to help detect both known and unknown intrusions.

The paper by Mala K et.al,[2] propose that cloud-deployed deep learning IDS can improve network traffic and reduce false positives and it presents the model-development study build around the UNSW-NB15 intrusion dataset. The central point represents that the results in the deep models outperformed the Random Forest baseline ranging from 88% to 95%.

Ze Yang et.al,[3] proposes that a large language model based system for the cloud network traffic monitoring and anomaly detection which was evaluated on CICIDS2017 against AutoEncoder, RandomForest, SVM, LSTM baselines. Its main claim is to add the transform style temporal attention, a supervised anomaly layer. And transfer learning improves detection accuracy and lowers false positives while keeping computation practical for deployment.

According to Robin Thomas et.al,[4] the starting point is that network traffic analysis has become harder because of the traffic volume, speed and heterogeneity which are increased by IoT devices, cloud services, streaming, mobile traffic and 5G infrastructure. They also stated that the traditional tools like SNMP and sampled flow monitoring are now insufficient for fine-grained visibility as they miss short lived bursts, congestion spikes and parts of network that are not instrumented.

The paper proposed by Bhadule Priyanka Umesh et.al,[5] has three different core functions of IDS: Anomaly detection, behavioural analysis, and signature-based detection. Their idea was to learn the baseline of normal cloud activity, flag the deviations, and adapt as the threats evolve. Furthermore they have used several algorithms like, SVM for classification, Random forest for robust ensemble prediction, KNN for proximity-based labeling and Neural Networks for the complex pattern extraction.

Dave Michael [6] proposed a cloud based IDS, defined as the intrusion detection system that operates within or across the cloud infrastructure to monitor resources, detect threats, and alert administrators in real time. This paper emphasize that cloud IDS differs from traditional IDS as it must work across elastic, multi-tenant and abstracted cloud layers such as IaaS, PaaS and SaaS.

According to Tabinda Shehzadi [7], she states that the core problem is that cloud adoption increases flexibility and scalability but also creates the data protection risks tied to distributed infrastructure, shared responsibilities, unauthorized access and compliance obligations. The paper identifies recurring challenge areas: Data Breaches, Misconfigurations and weak access controls, regulator compliance and insider or evolving cyber threats. It also point up that the distributed storage across regions creates sovereignty and privacy compilations under rules such as GDPR and CCPA.

According to Roma Soni and Nitin Uikey [8], the Cloud Computing offers scalable and on-demand resources for SaaS, PaaS, IaaS and public, private, hybrid which is why they became central in IT field. The core point is that, these same cloud features also expands the attack. The paper talks about the need for the cloud security because the organizations has to maintain confidentiality, integrity and availability while also preserving the trust, compliance and business continuity.

Beatrice Baah [9] proposed that the cloud computing offers scalable, on demand services, but its distributed, shared resource architecture creates distinctive security and privacy risks. It frames these risks around the loss of direct control, multi tenancy, public-network exposure, and weak visibility into where data and physical controls actually reside.

Proposed Methodology

The proposed methodology aims to develop an intelligent intrusion detection system capable of identifying malicious activities in cloud network traffic using machine learning techniques. The proposed framework combines data preprocessing, feature engineering, supervised machine learning, and performance evaluation to accurately classify network traffic as either normal or malicious. The overall workflow begins with the NSL-KDD dataset, which is used to simulate cloud network traffic. The collected data undergoes preprocessing to improve its quality and compatibility with machine learning algorithms. After preprocessing, additional traffic-related features are generated through feature

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

engineering to enhance the learning capability of the models. The processed dataset is then used to train two supervised machine learning algorithms: Random Forest and Logistic Regression. Finally, the trained models are evaluated using multiple performance metrics, and the model with the best performance is selected for intrusion detection.

The overall methodology consists of six major stages:

1. Dataset Collection
2. Data Preprocessing
3. Feature Engineering
4. Model Development
5. Performance Evaluation
6. Intrusion Detection

A suitable dataset plays an essential role in the performance of machine learning models. In this research, the NSL-KDD dataset is selected because it is one of the most widely used benchmark datasets for intrusion detection research. The dataset was developed to overcome the redundancy and duplication problems present in the earlier KDD Cup 1999 dataset.

Each record in the NSL-KDD dataset represents one network connection and contains 41 original traffic features along with one class label. These features describe different characteristics of a network connection, including protocol type, service, flag status, connection duration, source bytes, destination bytes, error rates, login attempts, and traffic statistics.

For this study, all attack categories are grouped into a single attack class. Therefore, the classification problem is converted into binary classification:

- Normal Traffic = 0
- Attack Traffic = 1

This binary representation simplifies the classification task while maintaining the objective of identifying malicious activities in cloud network traffic.

Raw datasets usually contain categorical values, different numerical scales, and irrelevant information that cannot be directly processed by machine learning algorithms. Therefore, several preprocessing operations are performed before model training.

The original NSL-KDD dataset contains several attack names, such as DoS, Probe, R2L, and U2R, along with normal traffic records. These labels are converted into binary values where normal traffic is represented by 0 and all attack types are represented by 1. This conversion transforms the problem into binary classification.

Some attributes in the NSL-KDD dataset are categorical, including:

- Protocol Type
- Service
- Flag

Machine learning algorithms cannot process textual values directly. Therefore, Label Encoding is applied to convert each categorical value into a unique numerical value while preserving consistency throughout the dataset.

Numerical attributes have different value ranges. For example, duration values may be very small, whereas source byte values can be much larger. Such differences may negatively influence certain machine learning algorithms. To address this issue, StandardScaler is applied to normalize numerical features by transforming them to a standard distribution with zero mean and unit variance. Although Random Forest is less sensitive to scaling, Logistic Regression benefits significantly from normalized input data.

Feature engineering is one of the most important components of the proposed methodology. Instead of relying only on the original 41 features provided by the NSL-KDD dataset, additional features are generated to better represent cloud network traffic behavior.

The first engineered feature is the total amount of data transferred during a network connection.

Formula:

Total Bytes = Source Bytes + Destination Bytes

This feature represents the overall communication volume between two endpoints. Larger traffic volumes may indicate abnormal behavior such as flooding attacks or heavy data transfers.

The byte ratio measures the relationship between transmitted and received data.

Formula:

Byte Ratio = Source Bytes / (Destination Bytes + 1)

Adding one prevents division by zero. This feature helps identify asymmetric communication patterns that are commonly observed during malicious activities.

Instead of using raw duration values directly, the connection duration is divided into three categories:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

- Short Duration
- Medium Duration
- Long Duration

This categorization enables the models to better capture temporal characteristics of network traffic while reducing the effect of extreme duration values.

The proportion of source bytes relative to the total traffic is calculated using:

Source Byte Percentage = Source Bytes / Total Bytes

This feature indicates the contribution of outgoing traffic within a network connection.

Similarly, the destination byte percentage is calculated as:

Destination Byte Percentage = Destination Bytes / Total Bytes

This feature reflects the contribution of incoming traffic during communication.

The engineered features provide additional traffic-level information that is not directly available in the original dataset, thereby improving the ability of the models to distinguish between normal and malicious network behavior.

After preprocessing and feature engineering, the processed dataset is divided into training and testing datasets. The training dataset is used to construct machine learning models, while the testing dataset is used to evaluate their performance.

Two supervised machine learning algorithms are implemented.

Random Forest is an ensemble learning algorithm that combines multiple decision trees to improve prediction accuracy and reduce overfitting. During training, each decision tree is constructed using randomly selected subsets of data and features. The final prediction is determined through majority voting among all trees.

Random Forest is selected because it performs well on high-dimensional datasets, handles nonlinear relationships effectively, and is robust against noisy data.

Logistic Regression is used as a baseline classification algorithm. It estimates the probability that a network connection belongs to either the normal or attack class using a logistic function. Although it is simpler than Random Forest, it provides an effective benchmark for comparison.

The trained models are evaluated using the testing dataset. Several evaluation metrics are employed to provide a comprehensive assessment of model performance.

Accuracy measures the proportion of correctly classified records.

Precision indicates how many predicted attacks are actually attacks.

Recall measures the percentage of actual attacks correctly detected.

The F1-score represents the harmonic mean of precision and recall, providing a balanced evaluation of classification performance.

The confusion matrix summarizes prediction results by showing:

- True Positives
- True Negatives
- False Positives
- False Negatives

This matrix provides detailed insight into the strengths and weaknesses of the proposed intrusion detection model.

Once the model is trained, incoming cloud network traffic undergoes the same preprocessing and feature engineering steps before being supplied to the trained classifier. The model analyzes each connection and predicts whether it represents normal activity or a cyberattack.

If the prediction indicates normal traffic, the connection is allowed to continue. If malicious activity is detected, the system generates an alert, allowing cloud administrators to take immediate security measures such as blocking the connection, isolating the affected resource, or initiating further investigation.

By combining preprocessing, engineered traffic features, and machine learning algorithms, the proposed methodology provides an efficient and intelligent approach for cloud network intrusion detection. Experimental evaluation demonstrates that the Random Forest model achieves higher classification accuracy than Logistic Regression, confirming the effectiveness of the proposed framework for identifying malicious cloud network traffic.

III. RESULTS

The proposed intrusion detection system was evaluated using the NSL-KDD dataset to classify cloud network traffic as either

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

normal or attack. Two supervised machine learning algorithms, namely Random Forest and Logistic Regression, were implemented and compared. The evaluation was carried out using accuracy, confusion matrix, precision, recall, and F1-score.

The Random Forest classifier achieved an overall accuracy of **78.10%**, whereas the Logistic Regression model achieved an accuracy of **74.78%**. The results indicate that the Random Forest model performs better than Logistic Regression in identifying malicious network traffic because of its ability to capture complex and nonlinear relationships among the network features.

The confusion matrix obtained from the Random Forest classifier is shown in Table 4.1.

Table 4.1 Confusion Matrix of Random Forest

Actual Class	Predicted Normal	Predicted Attack
Normal	9429	282
Attack	4656	8177

The confusion matrix shows that the model correctly classified **9,429 normal traffic records** and **8,177 attack records**. However, **282 normal records** were incorrectly classified as attacks, while **4,656 attack records** were misclassified as normal traffic.

The classification report obtained from the Random Forest model is summarized in Table 4.2.

Table 4.2 Classification Report

Class	Precision	Recall	F1-Score	Support
Normal (0)	0.67	0.97	0.79	9,711
Attack (1)	0.97	0.64	0.77	12,833
Overall Accuracy	-	-	**0.78**	**22,544**

For the **normal traffic class**, the model achieved a precision of **67%**, recall of **97%**, and an F1-score of **79%**. The high recall indicates that most normal traffic records were correctly identified.

For the **attack traffic class**, the model achieved a precision of **97%**, recall of **64%**, and an F1-score of **77%**. The high precision shows that when the model predicts an attack, it is usually correct. However, the lower recall indicates that some attack records were incorrectly classified as normal traffic.

Logistic Regression was used as a baseline model to compare its performance with the Random Forest classifier. The model achieved an overall accuracy of **74.78%**, which is lower than the accuracy obtained by the Random Forest model.

The lower accuracy suggests that Logistic Regression is less effective in capturing the complex relationships present in cloud network traffic. Since Logistic Regression assumes a linear decision boundary, it cannot model nonlinear attack patterns as effectively as Random Forest.

Table 4.3 Performance Comparison

Model	Accuracy (%)
Random Forest	**78.10**
Logistic Regression	**74.78**

The comparison clearly shows that the Random Forest classifier outperformed Logistic Regression by approximately **3.3%** in terms of overall accuracy. The ensemble learning mechanism of Random Forest enables it to learn more complex traffic patterns and improves its classification capability.

The experimental results demonstrate that combining feature engineering with machine learning improves the effectiveness of intrusion detection in cloud network traffic. The engineered features, including total traffic volume, byte ratio, duration category, source byte percentage, and destination byte percentage, provide additional information that helps the models distinguish between normal and malicious traffic.

Although the proposed Random Forest model achieved satisfactory performance with an accuracy of **78.10%**, some attack records were still classified as normal traffic, as reflected by the recall value of **64%** for the attack class. This indicates that there is still room for improvement. Future work may explore advanced ensemble methods, deep learning models, feature selection techniques, or hyperparameter optimization to further enhance detection accuracy and reduce false negatives.

Overall, the results confirm that the proposed feature engineering approach, together with the Random Forest classifier, provides a reliable and effective solution for cloud network intrusion detection.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

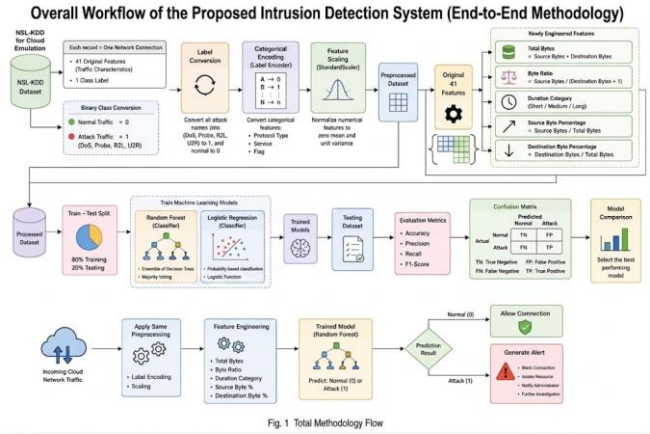

Fig. 1 Total Methodology Flow

IV. CONCLUSION

Cloud computing has become an essential platform for delivering computing resources and services over the Internet. However, the rapid growth of cloud environments has also increased the risk of cyber threats, making intrusion detection an important component of cloud security. This paper presented a machine learning-based intrusion detection approach for cloud network traffic using the NSL-KDD dataset. The proposed framework combined data preprocessing, feature engineering, and supervised machine learning techniques to classify network traffic as either normal or malicious.

Several preprocessing techniques, including label encoding and feature scaling, were applied to prepare the dataset for model training. In addition, feature engineering was performed by generating new traffic-related features such as total traffic volume, byte ratio, duration category, source byte percentage, and destination byte percentage. These engineered features provided additional information about network traffic patterns and improved the learning capability of the machine learning models.

Two classification algorithms, Random Forest and Logistic Regression, were implemented and evaluated. Experimental results showed that the Random Forest classifier achieved an accuracy of **78.10%**, outperforming the Logistic Regression model, which achieved an accuracy of **74.78%**. The confusion matrix and classification metrics further demonstrated that the Random Forest model was more effective in distinguishing between normal and malicious network traffic. The results indicate that feature engineering, combined with ensemble learning techniques, can improve intrusion detection performance in cloud environments.

Although the proposed model achieved satisfactory performance, there is still scope for improvement. The recall value for attack detection indicates that some malicious traffic was classified as normal traffic. Future research can focus on applying advanced machine learning and deep learning techniques, feature selection methods, hyperparameter optimization, and real-time cloud traffic analysis to further improve detection accuracy and reduce false negatives.

In conclusion, the proposed intrusion detection framework demonstrates that integrating feature engineering with machine learning provides an effective and practical solution for enhancing cloud network security. The proposed approach can serve as a strong foundation for developing intelligent intrusion detection systems capable of protecting modern cloud computing environments from evolving cyber threats.

REFERENCES

[1] **Goce Stevanoski et al. [1]** proposed that machine learning-based anomaly detection improves intrusion detection by identifying both known and unknown attacks with minimal human intervention. Their work highlights the importance of learning normal network behavior to detect anomalies effectively.

[2] **Mala K et al. [2]** presented a deep learning-based intrusion detection system for cloud environments using the UNSW-NB15 dataset. Their proposed model achieved higher detection accuracy than the Random Forest baseline while reducing false positives.

[3] **Ze Yang et al. [3]** developed a large language model-based approach for cloud network anomaly detection. Their method improved detection accuracy and reduced false positives by incorporating temporal attention and transfer learning.

[4] **Robin Thomas et al. [4]** discussed the challenges of modern network traffic analysis caused by cloud computing, IoT, and 5G networks. They concluded that traditional monitoring techniques are insufficient for handling dynamic and large-scale network traffic.

[5] **Bhadule Priyanka Umesh et al. [5]** proposed an intrusion detection system combining anomaly detection, behavioural analysis, and signature-based detection. Their work evaluated multiple machine learning algorithms, including SVM, Random Forest, KNN, and Neural Networks.

[6] **Dave Michael [6]** presented a cloud-based intrusion detection system designed to monitor cloud resources and detect security threats in real time. The study emphasized the need for IDS solutions that support IaaS, PaaS, and SaaS environments.

[7] **Tabinda Shehzadi [7]** analyzed major cloud security challenges, including data breaches, misconfigurations, weak access control, and regulatory compliance. The study highlighted the importance of protecting sensitive cloud data in distributed environments.

[8] **Roma Soni and Nitin Uikey [8]** discussed the security challenges introduced by cloud computing despite its scalability and flexibility. The authors emphasized maintaining confidentiality, integrity, and availability to ensure secure cloud services.

[9] **Beatrice Baah [9]** examined the security and privacy issues in cloud computing caused by shared resources and distributed infrastructure. The study focused on challenges such as data protection, multi-tenancy, and limited user control over cloud environments.