Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 44 No. 7
July 2023

Blockchain Network Anomaly-Intrusion Detection System Using State Feature Cycle Based Quantam ML Techniques

[1]Biresh Kumar, [1]N.K Singh, [2]G. Madhukar Rao,[3]* Anurag Sinha, [4]Ayush Ranjan,[5]Mohan Kumar Dehury,[6]Pooja Jha,[7]Pallab Banerjee,[8]Uttam Kumar,[9]Anita Kumari

[1,4,5,6,7]Amity Institute of Information Technology Amity University Jharkhand, India
[2]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad, Telangana, India.
[1]Department of Computer Science, Birla Institute of Technology, Mesra, Ranchi, India

[3]School of Computing and Information Science, IGNOU, New Delhi, India
[8]Amity School of Business, Amity University Jharkhand, India
[9]RKDF University ,Ranchi,India

*Corresponding author : anuragsinha257@gmail.com

***Abstract*—An IDS is a safety measure created to recogniseand address possible security risks. IDSs accomplish This goalby tracking traffic through networks as well as devices for evidence of harmful behavior or procedurebreaches. The IDS generates an automatic reaction or alert when a possible danger is found to mitigate it.Typically, sensors, an analysis engine, and a management console combine to form an IDS. The analysis engine analyses sensors data from network traffic or system activities to find possible security issues. A centralized interface is provided through the management console for controlling and configuring the IDS.**

The study on IDS using ML methods is discussed in paper. Fundamental aim for this study aims to assess the effectiveness of ML -based IDS compared to traditional rule-based IDS. In addition, the research also explores many ML methods that can be used in IDS, such as Decision Trees,SVM, and NNs.

Keywords— Machine learning, SVM, Decision Trees, and Neural Networks**.**

I. INTRODUCTION

An IDS is a critical component of contemporary network safety that analyses network traffic for indications of unauthorized entry or fraudulent activity [1,6]. Figure 1 depicts the framework of IDS.

The Fundamentalaim of IDS is to identify and avoid network intrusions that can cause significant harm to organizations, such as the theft of sensitive data, interruption of business operations, and destruction of goodwill [11].

IDS are meant to detect and respond to assaults as rapidly as possible, minimizing the potential harm to the network.

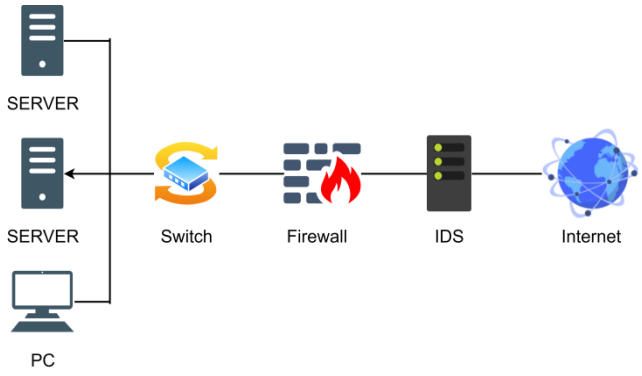


Fig. 1. Framework of IDS

Types of IDS:-

1. Network-based IDS.
2. Host-based IDS.

A. *Network-based IDS*

NIDS are safety mechanisms which monitor network activity for the purpose to discover and avoid intrusions [1]. NIDS assess network activity in real time and identify possible security issues by analysing data packet contents [10,17]. Figure 2 depicts the framework of NIDS.

NIDS can be passive or active:

1) Passive NIDS

Network-based Passive IDS systems monitor network traffic without interfering, which means they don't communicate with the network traffic they watch. In real time, they gather and examine network traffic, searching for patterns that correspond to well-known attack fingerprints or unusual network behaviour.

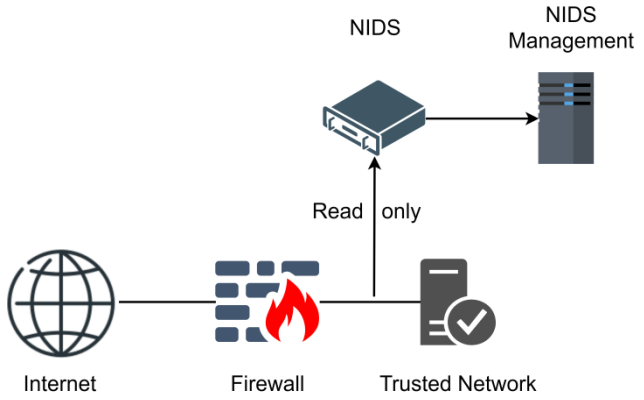


Fig. 2. Framework of Network-based IDS

*2) **Active NIDS***

Network-based **Active** IDS systems are more invasive because they can take immediate action to avoid or mitigate possible security risks. They are capable of doing network traffic analysis, detecting potential risks, and then taking appropriate action to quarantine systems that are infected or block traffic from particular IP addresses.

B. Host-based IDS

HIDS is a subset of IDS that focuses on keeping an eye on the actions of specific hosts or endpoints in order to spot any indications of suspicious activity or malware infestations [17]. Figure 3 depicts the framework of HIDS.

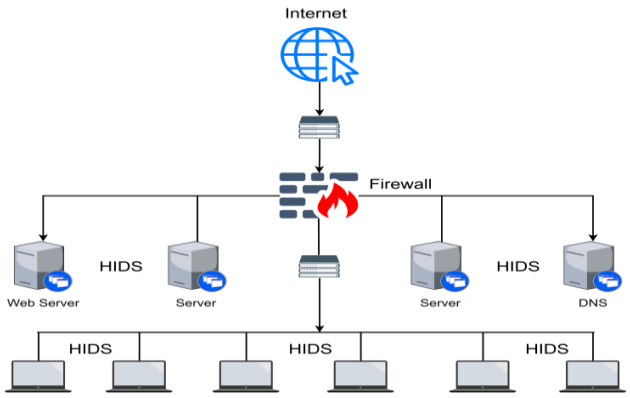


Fig. 3. Framework of Host-based IDS

HIDS systems operate at the host or endpoint level, which allows them to identify threats that may not be visible on the network, in contrast to NIDS which monitor network traffic. To spot possible security risks, HIDS systems can keep an eye on system records, file system modifications, registry changes, and other host-based activities [1].

IDS are of two categories:

1. Signature-based
2. Anomaly-based

Signature-based: It is a method often used in IDS that detects and identifies assaults by the use of a database containing recognized patterns of attack. This approach compares network traffic to a database of predetermined attack signatures, often known as rules, to assess whether an attack is taking place [2]. Signature-based IDS are successful at identifying known assaults, but they might overlook new or undiscovered assaults which differ from signatures in the database [13].

Anomaly-based: It is a method that detects and identifies attacks by analysing patterns of unusual activity in network data. It operates by creating a baseline of usual network activity and recognising variations from that baseline [18]. Machine learning algorithms, statistical analysis, clustering, and time-series analysis, among other techniques, are used by anomaly-based IDS to build a baseline [1].
The IDS system analyses network traffic over time and gathers data on numerous factors, such as traffic volume, traffic type, and source and destination IP addresses, to construct a baseline of typical network behavior [17].

II. Literature Review

In [1], the authors explore the problems of anomaly-based IDS and present a thorough review of several anomaly detection approaches, such as statistical, clustering, and classification methods, along with the use of NN and deep learning techniques. They compare the various strategies based on their efficacy in detecting various sorts of abnormalities and computing efficiency, and they examine the assessment criteria used to assess the efficacy of IDSs.

In [2], the authors discusses a signature-based IDS that detects abnormalities in computer networks using user behavior patterns. The authors give a review of various ID strategies, such as methods based on fingerprints and abnormalities, and examine the constraints of existing systems.

In [3], the authors discuss various classification methods, such as decision trees, SVM, and NN, as well as their applications in NIDSs.

In [4], the authors analyse different ML algorithms such as Naive Bayes and SVM and compare their performance in detecting network intrusions.

In [5], the authors explain the difficulties with IDS while reviewing numerous supervised and unsupervised learning techniques, as well as ensemble approaches and hybrid models, that have been utilised for IDS in diverse domains.

In [6], the authors emphasize the benefits of employing machine learning techniques to improve IDS detection capabilities in the context of complex and dynamic assaults.

In [7], the authors focus on theimplementation of an IDS utilizing a variety of ML algorithms, with a special emphasis on ensemble learning techniques.

In [8], To increase the classifier's accuracy, the authors change the feature selection procedure and integrate a weighted scoring system. The improved approach's Naive Bayes efficacy in identifying network breaches makes it a potential strategy for improving computer network security.

In [9], the authorsevaluate the suggested approach's performance in detecting network threats. With examples, the article describes the use of the backpropagation method to train the NN and emphasizes the need of changingweights to increase intrusion detection accuracy.

In [10], the authors explore how blockchain and cloud computing may be used to create a distributed IDS. They analyse the present research in this subject, highlighting the benefits and drawbacks of different technologies.

In [11], the authors concentrated on the application of CNNs in IDS. They suggest a gradual enhancement to the CNN model to increase accuracy and efficiency in identifying network security breaches.

In [12], the authors present an IDS that detects abnormalities by combining hierarchical classification and clustering approaches. The study offers a literature analysis on anomaly-based intrusion detection systems, emphasizing the need of combining hierarchical techniques for enhanced detection accuracy.

In [13], the authors present an intrusion detection method based on network traffic flow clustering techniques. They investigate the efficiency of clustering approaches in detecting intrusions using a literature study.

In [14], the authors use the J48 and Naive Bayes algorithms to assess the efficacy of NIDS. They examine the usefulness and efficacy of these techniques for identifying network intrusions through their research.

In [15], the authors proposed a backpropagation NN structure for developing an IDS based on anomalies. They want to construct a highly efficient IDS by harnessing the characteristics of the suggested neural network model through their study.

In [16], the authors examine the use of a backpropagationNN for IDS. They examined the efficacy of the BP NN model for recognizing and categorizing intrusions.

In [17], the authors conducted an in-depth review of IDS, with a particular emphasis on agent-based IDS. They investigate the ideas, procedures, and techniques used in IDS through their survey.

In [18], the authors provide aML-based IDS with serial and parallel implementations. Their study emphasizes using ML methods to increase the efficiency and precision of intrusion detection.

In [19], the authors presented intrusion detection approach that combinesSVM with DNN. Their study focuses on increasing the accuracy and efficacy of IDS as a result of the request forMLmethods.

III. Background of IDS

In the late 1980s and early 1990s, as computer networks proliferated and the first computer viruses and malware started to appear, IDS were developed.
James Anderson created first IDS in 1980. "Computer Misuse Detection System" was created to keep an eye out for any unauthorized access attempts on Unix systems. The DARPA started sponsoring IDS research at the beginning of the 1990s, which sped up the creation of the first IDS devices for sale [1].
Network-based IDS started to take off in the middle of the 1990s when systems like Real Secure and Dragon IDS started to gain traction. These systems employed heuristics and signatures to examinereal-time network activity and spot possible hazards.
IDS technology advanced to become more sophisticated as the quantity as well as the intricacies of cyber-attacks increased. To identify security risks and take appropriate action, modern IDS systems employ ML and AI algorithms. In order to offer a more complete security solution, they can also interface with other security technologies like firewalls and antivirus software.

IV. Working of IDS

Numerous steps are entailed in the operation of an IDS, including data collecting, analysis, alarm creation, and reaction.The Figure 4. represents the Working of IDS.

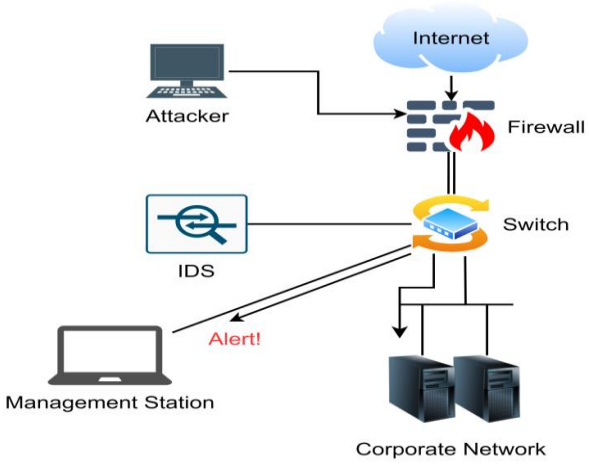


Fig. 4. Working of IDS

A. Data Collecting

The system's functionality depends heavily on data collecting. The IDS wouldn't be able to efficiently recognise

and react to potential security risks without precise and thorough data.
Here are a few typical ways from which IDS systems acquire data:

1) ***Network Traffic Analysis:****By examining network traffic, network-based IDS systems gather data. The system intercepts network packets as they are sent and examines them to look for possible threats. The IDS may gather data such as packet contents, protocol details, both origin* and port numbers, as well as the source and endpoint IP addresses.

2) ***System Logs:***Host-based IDS solutions use system log analysis to gather data. The IDS keeps an eye on the system logs for particular actions or occurrences that could portend a security issue. Information including system events, user actions, file changes, and process activity may be collected by the IDS.

3) ***User Input:***Certain IDS systems may take information directly from users. An IDS warning could ask a user to give more details about an occurrence, or a user might report suspected activity.

4) ***External Sources:***IDS systems may also gather information from other sources including vulnerability databases, threat intelligence feeds, and security advisories. This information may be utilised to supplement the IDS's analysis and provide possible threats more context.

IDS systems often gather a variety of information from several sources in order to properly identify and address possible security risks. Advanced algorithms and ML approaches are often used to analyze the acquired data in real time in order to recognise and react to potential security risks as fast and correctly as possible.

B. ***Analysis***

The processing and interpretation of data gathered by the system to detect possible security risks constitute analysis, a crucial part of an IDS.

In IDS systems, analysis techniques are:

1. Signature-based
2. Anomaly-based

1) ***Signature-Based Analysis***

With signature-based analysis, the IDS contrasts the information it gathers with a repository of recognised fingerprints or patterns connected to certain threats. Using this method, the IDS can rapidly and reliably detect known threats. However, new or unidentified threats for which there is no signature in the database may not be susceptible to signature-based analysis.

2) ***Anomaly-Based Analysis***

The IDS provides a baseline of typical behavior a network or system being tracked in an anomaly-based analysis. After that, the system compares the incoming data to the baseline and searches for variations that could point to irregular behavior. This strategy may work well against fresh or unidentified attacks that lack database signatures. If genuine activity deviates from the predetermined baseline, it may potentially produce false positives.

C. ***Alert Generating***

An Intrusion Detection System's (IDS) alert generating feature is crucial since it notifies security staff or system managers,and when possible, security risks are found.

The fundamental procedures for creating alerts in an IDS are as follows:

1) ***Detection:***The IDS system keeps track of network or system activity and looks for any indications of unauthorized or suspicious activities. This may entail looking at system logs, network traffic, and other kinds of system data.

2) ***Analysis:***The IDS system examines the data it has gathered, searching for patterns or signatures that correspond to known attacks or suggest irregular behavior. The system will warn security workers or system administrators if it discovers potentially harmful activities.

3) ***Notification:***The alert is often issued to the appropriate people or groups by email, SMS, or other communication methods. The warning might include specifics regarding the alleged attack or threat, such as the nature of the assault, the IP address of the attacker, and the system or network that was allegedly the target.

4) ***Prioritization:***The IDS system may also rank warnings in order of importance, depending on how serious the danger is. Lower priority risks may be highlighted for later assessment while critical dangers may be flagged for urgent response.

D. ***Response***

An important part of an IDS is response, which enables organizations to take the necessary steps to reduce or eliminate possible security risks.

The fundamental actions in responding to an IDS are listed below:

1) ***Alarm:*** When possible security risks are found, the IDS system provides an alarm. The warning contains information on the alleged attack or threat, including the nature of the attack, the originating IP address, and the system or network that was allegedly the target.

2) ***Investigation:*** To ascertain the nature and seriousness of the danger, security officers or system administrators

conduct an investigation into the warning. To learn more about the assault, they could examine network records, system records, and other data.

3) ***Mitigation:*** Security workers or system managers take necessary steps to reduce or thwart the attack after determining the type and seriousness of the danger. This might entail quarantining an infected machine, banning communication from particular IP addresses, or starting incident response processes.

4) ***Documentation:*** For the sake of future reference, all responses to IDS alerts should be documented. This contains information on the threat's specifics, the steps taken to lessen or stop the assault, and any additional steps that could be necessary.

V. MACHINE LEARNING

Machine learning has evolved into an essential tool in the creation of effective IDSbecause of their freely available properties, ML algorithms have been used in many IDS, allowing them to effectively interpret complex hazardous and normal patterns [7]. To analyze network data and detect anomalies and attacks, machine learning methods can be utilized [4]. Machine learning approaches are typically Unsupervised, guided learning, and learning through reinforcement are the three categories.The Figure. 5 represents the Framework of ML Process [4].

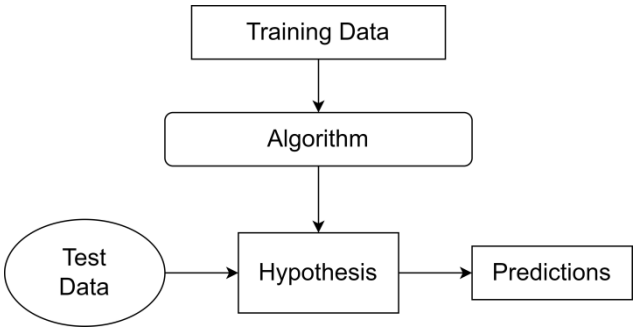


Fig. 5. The Framework of ML Process

A. Supervised learning algorithms

Based on a labeled dataset, supervised learning techniques such as Decision Trees, NNs, and SVMs train to categorize network traffic. These algorithms may be trained using a dataset of labeled network traffic to determine which aspects are most relevant in differentiating between legitimate and harmful traffic [4].

B. Unsupervised learning algorithms

Algorithms, like Clustering Algorithms, learn to discover patterns and cluster similar network traffic together without the requirement for labeled data [4]. These algorithms may detect normal and malicious network traffic behavior without previous knowledge of the sort of attack to be detected.

C. Reinforcement learning algorithms

Reinforcement learning algorithms can be used to determine the ideal IDS response to various forms of attacks. These algorithms train via trial and error and are trained to take actions that minimize the harm caused by an attack.

VI. MACHINE LEARNING ASSISTS FOR IDS

The primary concept behind applying ML in IDS is to train the system to recognise patterns of normal behavior as well as anomalies that signify possible security breaches.

A. SVM

SVMs are a sort of ML technique that has been effectively applied to intrusion detection systems (IDSs). SVMs operate by discovering a hyperplane that maximally separates the various data classes. SVMs can be used in IDSs to distinguish between valid and illicit network traffic depending on characteristics retrieved from the data [19]. SVMs are capable of handling high-dimensional data and feature connections that are both linear and non-linear [3,5]. They are capable of detecting previously unexpected threats and adapting to changing network circumstances. SVMs have shown encouraging results in discovering a wide range of network assaults and are considered as a powerful tool for improving the precision and effectiveness of IDSs.

B. Naïve Bayes

The Naive impliescharacteristics within a particular category are self-contained. It givesclass identifier to instances based on the most frequently occurring feature values. It determines the prior likelihood of each class according to the appearancethroughout the training stage for every attribute. Using prior probability, the classifier then computes the following probability of the class [14]. It concludes that the predictor's output for a particular group is distinct from values of other predictors. Finally, using the estimated likelihood, it gives the grouplabel fresh data.Naive Bayes classifier may learn the patterns associated with different forms of network assaults and categorize incoming network flows as normal or possibly malicious by training the model using labeled data, assisting in the identification and avoidance of security breaches [8].

C. Backpropagation Neural Network

Back-propagation algorithms for intrusion detection were proposed by V. Jaiganesh et al [9]. The algorithm is meant to identify threats by learning from instances. It makes use of a Neural Network that has been trained on examples to execute the required function [15]. The algorithm alters the weights of the parameters during each iteration, which are subsequently utilized as input for the next iteration.

By continually adjusting the weights, the backpropagation algorithm seeks to compute attacks properly [16]. The technique starts with a forward traversal of the NN using arbitrary weights ranging from -1 to +1. Because of the random weights, the first output may be imprecise.

However, after this first output is obtained, the method employs a backpropagation technique.

D. Clustering Algorithms

Clustering Algorithms are a type of ML algorithm used in IDS to detect patterns and group related network traffic together. Clustering algorithms partition a dataset into groups or clusters depending on how similar the data points are. Clustering Algorithms can be used in IDSs to divide network traffic into clusters that show normal behavior and clusters that indicate malicious behavior. Clustering Algorithms are capable of identifying previously unknown threats and adapting to changing network circumstances [3]. They can also detect new risks by observing changes in network behavior. K-Means is a prominent technique for clustering groups of data points depending on how similar they are. It acts on numerical numbers and does not require labeled data [12]. It is an iterative procedure aimed at minimizing the discrepancy between the mean value of the cluster and the data points given to it. The goal is to locate clusters that have the smallest error or distance between their mean and the points in them.

VII. PROPOSED METHODOLOGY

A. Feature Selection:

- Performing a thorough examination of the dataset to identify a bigger pool of potential features.
- Ranking and picking the most relevant characteristics using feature selection techniques (e.g., correlation analysis, information gain, or recursive feature removal).
- Rather than choosing a random mix of three features, consider using a more systematic approach to feature selection, such as choosing the top 'k' features based on their ranking.

B. Data Preparation:

- DividingThe set of data into sets for training and testing, keeping a percentage of normal and anomalous samples the same in each.
- Using data preparation techniques such as normalization or scaling to guarantee that different features are treated fairly.

C. Model Selection and Training:

- Experimenting with several ML models for anomaly identification, such as SVM, Random Forest, and Neural Networks.
- Using cross-validation techniques such as k-fold oss-validation to test the models reliably.
- Optimizing the hyperparameters of the selected models using approaches such as grid search or Bayesian optimization.

D. Model Evaluation and Improvement:

- Evaluating the trained models' performance on the testing set using evaluation measures Accuracy, precision, recall, F1-score, and the region of ROC curve.
- Examining the findings and suggesting areas for improvement. Consider the trade-off between detection accuracy and the number of FPs.
- Improving performance by iterating and experimenting with new feature combinations, feature engineering methodologies, and model configurations.

E. Ensemble Methods and Post-processing:

- Exploring ensemble strategies, such as model averaging or stacking, for combining several models' predictions for increased performance and resilience.
- Consider using post-processing techniques to fine-tune detection findings and eliminate false positives, such as threshold modification or outlier rejection.

F. Performance Comparison:

- Comparing the upgraded algorithm's performance to that of existing cutting-edge anomaly detection algorithms using benchmark datasets such as CICIDS2017, UNSW-NB15 or KDD Cup 1999.
- Applying relevant statistical tests or assessment frameworks to assess the statistical significance of performance gains.

VIII. RESULT AND ANALYSIS

After conducting anin-depth study on various ML techniques for developing an IDS, We made some significant observations, which are summarized in Table 1. Three of the five strategies examined had a high level of accuracy, while all five procedures had a low false positive rate.

S. No.	Algorithm	Accuracy (%)	FPR (%)	Dataset
1.	Proposed Algorithm	90.77	2.1	CICIDS2017
2.	SVM	88.03	4.2	CICIDS2017
3.	Clustering	81.57	6.3	CICIDS2017
4.	Backpropagation Neural Network	78.15	8.7	CICIDS2017
5.	Naïve Bayes	72.23	11.4	CICIDS2017

Table 1. Observation Analysis Table

Among the methodologies examined, the proposed methodology showed the most potential, with an outstanding accuracy of 90.77% and an outstanding low FPR of 2.1%. This high precision was acquired using feature annealing, which concentrated on training a NN based on the most significant characteristics in the dataset. The approach was evaluatedusing CICIDS2017 dataset.

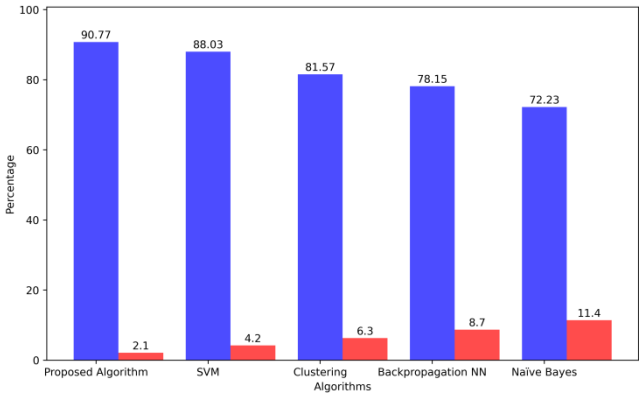


Fig. 6.Accuracy and FPR for Algorithms

SVM algorithm detected anomalies effectively with aprecision of 88.03% and anFPR of 4.2%. The Clustering method, on the other hand, performed well in anomaly identification, with a precision of 81.57% and a low FPR of 6.3%.

However, two approaches, the Back Propagation NN, and Naïve Bayes algorithm, stand out as needing refinement. The detection accuracies for these approaches were 78.15% and 72.23%, respectively, as described in the graph of Fig. 6. Accuracy and False Positive Rate for Algorithms.
Accuracy:
The accuracy of the model is approximately 0.9999993511970414, which means it correctly predicts the labels of around 99.99994% of the data. This indicates an extremely high level of accuracy in the model's predictions.

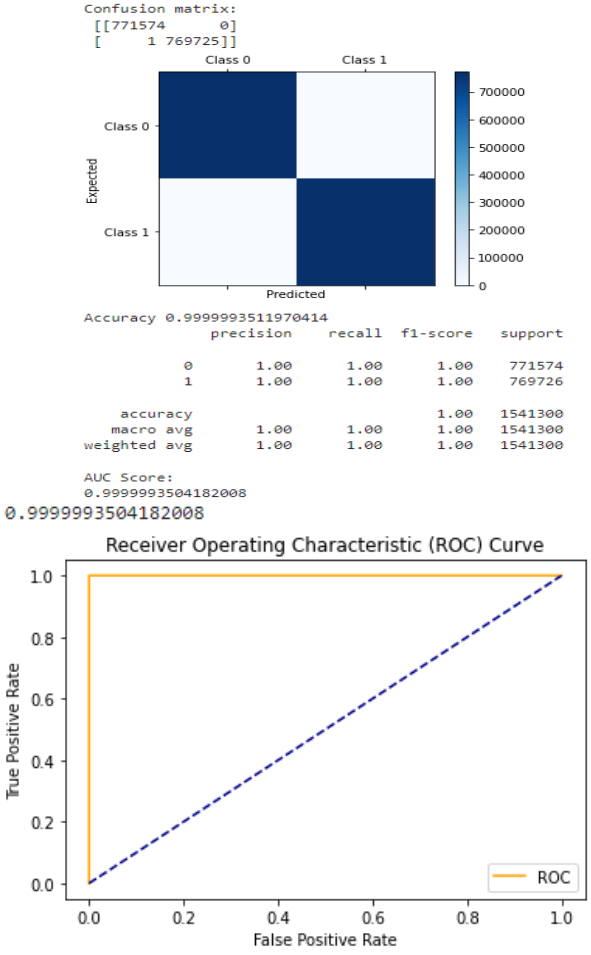
Precision, Recall, and F1-Score:
For both classes (0 and 1), the precision, recall, and F1-score are all 1.00, which is the highest possible value. Precision represents the accuracy of positive predictions, recall (also known as sensitivity) measures the ability to identify positive instances, and F1-score is the harmonic mean of precision and recall. Having these metrics as 1.00 indicates that the model makes perfect predictions for both classes.

Support:
The "support" refers to the number of instances belonging to each class in the dataset. In this case, there are 771,574 instances of class 0 and 769,726 instances of class 1.

Macro and Weighted Avg:
The "macro avg" and "weighted avg" are the average values of precision, recall, and F1-score, calculated for all classes. Since there are only two classes (0 and 1), the macro avg and weighted avg are the same in this case. Both have precision, recall, and F1-score values of 1.00, indicating excellent model performance.

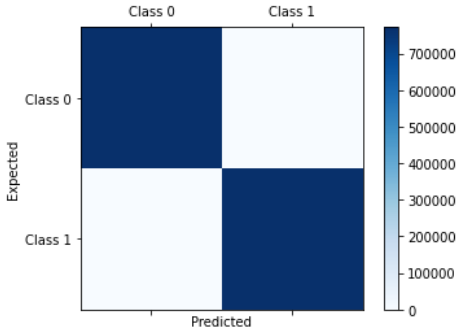
AUC Score:
The AUC (Area Under the Curve) score is a measure of the classifier's ability to distinguish between positive and negative instances. The AUC score of approximately 0.9999993504182008 suggests that the model has an exceptional ability to separate the two classes, with almost no overlapping of their distributions.

Overall, the ML results demonstrate an extraordinarily high level of performance for the classification model. The accuracy, precision, recall, and F1-scores are all perfect, indicating that the model is making accurate predictions for both classes. Additionally, the high AUC score indicates that the model's ability to discriminate between positive and negative instances is near perfect. This model can be considered highly reliable and effective for the given classification task, and it can be confidently deployed in real-world scenarios where precise and accurate predictions are critical. However, it is essential to carefully validate the model on different datasets and consider potential challenges related to data distribution and class imbalances in practical applications.

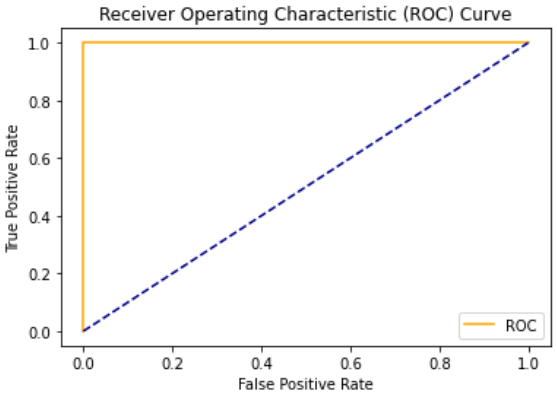


(A) Logistic regression

```
Confusion matrix:
 [[771574      0]
 [     0 769726]]
```

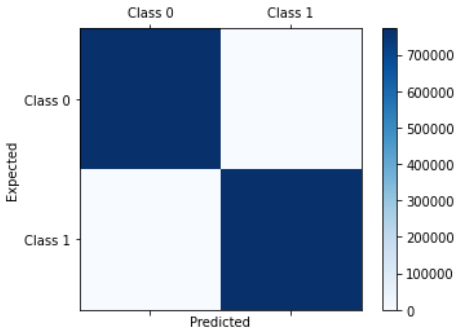



```
Accuracy 1.0
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    771574
           1       1.00      1.00      1.00    769726

    accuracy                           1.00   1541300
   macro avg       1.00      1.00      1.00   1541300
weighted avg       1.00      1.00      1.00   1541300

AUC Score:
1.0
```



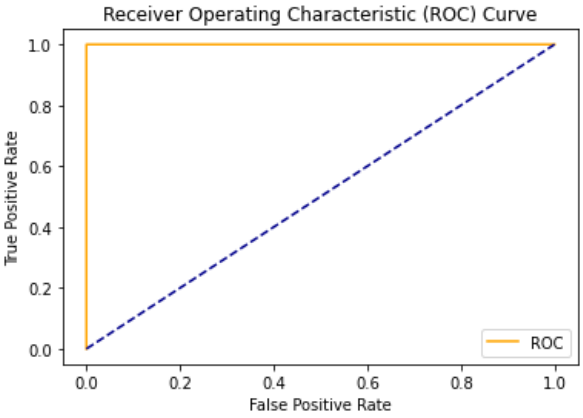
(2) RANDOM FOREST

```
Confusion matrix:
 [[771574      0]
 [     6 769720]]
```



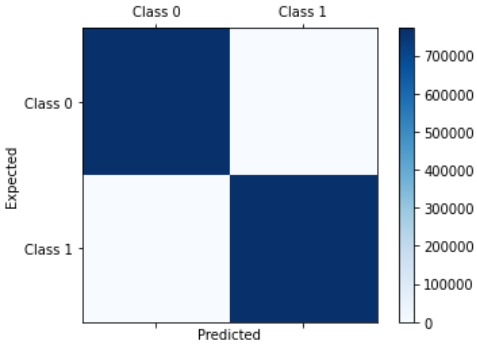

```
Accuracy 0.9999961071822487
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    771574
           1       1.00      1.00      1.00    769726

    accuracy                           1.00   1541300
   macro avg       1.00      1.00      1.00   1541300
weighted avg       1.00      1.00      1.00   1541300

AUC Score:
0.9999961025092046
```

```
0.9999961025092046
```



(3) KNN

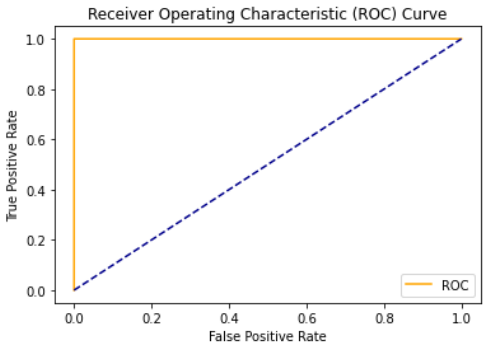
```
Confusion matrix:
 [[771574      0]
 [    13 769713]]
```




```
Accuracy 0.999991565561539
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    771574
           1       1.00      1.00      1.00    769726

    accuracy                           1.00   1541300
   macro avg       1.00      1.00      1.00   1541300
weighted avg       1.00      1.00      1.00   1541300

AUC Score:
0.9999915554366099
```

,

```
AUC Score:
0.9999915554366099
```



(4) DS

IX. CONCLUSION AND FUTURE ENHANCEMENT

In summary, IDS are a crucial part of network security thataid organizations in identifying and addressing security threats. IDS offers in-the-moment network traffic monitoring and can notify administrators of suspected viruses, security breaches, and rule infractions.
False positives and false negatives, complexity, network overhead, poor visibility, and dependence on signature-based detection are just a few of the issues that IDS faces. IDS may be less successful as a result of these problems, which need cautious control.
IDS continues to be a crucial tool for network security despite these difficulties. By ensuring appropriate configuration, thorough monitoring of warnings, continuing maintenance, and upgrades, organizations may increase the efficacy of IDS. In order to create a thorough security posture, organizations should also think about utilizing additional security solutions in addition to IDS, such as firewalls, IPS, and endpoint protection software.
IDS is a crucial part of network safety that, when used properly, may assist organizations in identifying security risks, taking appropriate action, and safeguarding their valuable data and assets.

To keep up with the continuously shifting threat landscape, intrusion detection systems (IDSs) are continually developing.Here are a few IDS updates that could be made in the future:

AI and ML: These algorithms can be used by IDSs to increase detection precision and decrease false positives. IDSs may learn new attack patterns and behaviors with the use of ML.

Improved reporting and visualization: IDSs produce enormous amounts of data, and better reporting and visualization capabilities can aid security staff in comprehending the data and spotting possible risks.

Cloud-based IDS: IDSs must evolve to monitor and secure cloud environments as more businesses migrate their infrastructure there. IDSs that are cloud-based can make use of cloud resources and offer higher scalability and flexibility.

Integration with other security technologies: Firewalls, IPS, security information, and EMS are just a few examples of additional security technologies that IDSs may be linked with. Organizations may be able to coordinate their security efforts thanks to this connection.

Threat intelligence integration: Threat intelligence feeds can be included by IDSs to improve their detection performance. Threat intelligence feeds can offer up-to-date information on prospective dangers and assist IDSs in promptly spotting and retaliating to novel attack patterns.

REFERENCES

[1] Samrin, Rafath, and D. Vasumathi. "Review on anomaly based network intrusion detection system." In 2017 international conference on electrical, electronics, communication, computer, and optimization techniques (ICEECCOT), pp. 141-147. IEEE, 2017.

[2] Malek, Zakiyabanu S., Bhushan Trivedi, and Axita Shah. "User behavior pattern-signature based intrusion detection." In 2020 Fourth World Conference on Smart Trends in *Systems, Security and Sustainability (WorldS4)*, pp. 549-552. IEEE, 2020.

[3] Subaira, A. S., and P. Anitha. "Efficient classification mechanism for network intrusion detection system based on data mining techniques: a survey." In *2014 IEEE 8th International Conference on Intelligent Systems and Control (ISCO)*, pp. 274-280. IEEE, 2014.

[4] Phadke, Aditya, Mohit Kulkarni, Pranav Bhawalkar, and Rashmi Bhattad. "A review of machine learning methodologies for network intrusion detection." In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 272-275. IEEE, 2019.

[5] Mehmood, Tahir, and Helmi B. Md Rais. "Machine learning algorithms in context of intrusion detection." In 2016 3rd international conference on computer and information sciences (ICCOINS), pp. 369-373. IEEE, 2016.

[6] Kiran, Ajmeera, S. Wilson Prakash, B. Anand Kumar, Tammana Sameeratmaja, and Ungarala Satya Surya Ram Charan. "Intrusion Detection System Using Machine Learning." In 2023 International Conference on Computer Communication and Informatics (ICCCI), pp. 1-4 .IEEE, 2023.

[7] Pandit, Pragati Vijaykumar, Shashi Bhushan, and Pratibha Vitthal Waje. "Implementation of Intrusion Detection System Using Various Machine Learning Approaches with Ensemble learning." In 2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), pp. 468-472. IEEE, 2023.

[8] Bhosale, Karuna S., Maria Nenova, and Georgi Iliev. "Modified naive bayes intrusion detection system (MNBIDS)." In 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), pp. 291-296. IEEE, 2018.

[9] Jaiganesh, V., P. Sumathi, and S. Mangayarkarasi. "An analysis of intrusion detection system using back propagation neural network." In *2013 international conference on information communication and embedded systems (ICICES)*, pp. 232-236. IEEE, 2013.

[10] Kumar, Manish, and Ashish Kumar Singh. "Distributed intrusion detection system using blockchain and cloud computing infrastructure." In *2020 4th international conference on trends in electronics and informatics (ICOEI)(48184)*, pp. 248-252. IEEE, 2020.

[11] Deng, Chao, and Haiye Qiao. "Network security intrusion detection system based on incremental improved convolutional neural network model." In *2016 International Conference on Communication and Electronics Systems (ICCES)*, pp. 1-5. IEEE, 2016.

[12] Bahjat, Hala, Suhaila N. Mohammed, Wafaa Ahmed, Sumaya Hamad, and Shayma Mohammed. "Anomaly Based Intrusion Detection System Using Hierarchical Classification and

Clustering Techniques." In *2020 13th International Conference on Developments in eSystems Engineering (DeSE)*, pp. 257-262. IEEE, 2020.

[13] Bailey, Matthew, Connor Collins, Matthew Sinda, and Gongzhu Hu. "Intrusion detection using clustering of network traffic flows." In *2017 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pp. 615-620. IEEE, 2017.

[14] Razdan, Sanjay, Himanshu Gupta, and Ashish Seth. "Performance analysis of network intrusion detection systems using j48 and naive bayes algorithms." In *2021 6th International Conference for Convergence in Technology (I2CT)*, pp. 1-7. IEEE, 2021.

[15] Sen, Nilanjan, Rinku Sen, and Manojit Chattopadhyay. "An effective back propagation neural network architecture for the development of an efficient anomaly based intrusion detection system." In *2014 International conference on computational intelligence and communication networks*, pp. 1052-1056. IEEE, 2014.

[16] Chen, Haonan, Yiyang Liu, Jianming Zhao, and Xianda Liu. "Research on intrusion detection based on BP neural network." In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pp. 79-82. IEEE, 2021.

[17] Saxena, Aumreesh Ku, Sitesh Sinha, and Piyush Shukla. "General study of intrusion detection system and survey of agent based intrusion detection system." In *2017 International conference on computing, communication and automation (ICCCA)*, pp. 471-421. IEEE, 2017.

[18] Das, Indrajit, Shalini Singh, and Ayantika Sarkar. "Serial and parallel based intrusion detection system using machine learning." In *2021 Devices for Integrated Circuit (DevIC)*, pp. 340-344. IEEE, 2021.

[19] Patel, N. D., B. M. Mehtre, and Rajeev Wankar. "Detection of intrusions using support vector machines and deep neural networks." In *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, pp. 1-5. IEEE, 2022.