International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

OPTIMIZED INTRUSION DETECTION SYSTEM USING MACHINE LEARNING

[1]Mrs. Bhavana, [2]N. Sudeshnu Ram ,[3]Rajesh Adhira ,[4]Togiti Varun Kumar
[1,2,3,4] Department of CSE (Artificial Intelligence and Machine Learning),
St. Peter's Engineering College, Telangana, India 1
bhavanagnv@gmail.com ,[2]nsudeshnuram.professional@gmail.com, [3]adhirajesh2205@gmail.com ,
[4]togitivarunkumar7700@gmail

Abstract — **Early and real-time detection of threats is essential to the security of modern networks from increasingly modern cyber threats. Traditional Intrusion Detection Systems, based on signature matching or simple classifiers, suffer from poor scalability, high false positives, and poor generalization to new attack methodologies. To overcome these shortcomings, we present an optimized IDS that combines Principal Component Analysis for feature reduction with a Random Forest classifier for efficient intrusion detection. The system uses the Knowledge Discovery and Dataset Cup 1999 training set for training and performance measurement, thus ensuring very high detection precision rates at the expense of minimal false positives and computational cost. PCA is used to reduce the complexity of network traffic data by retaining only the most important features, which not only increases processing speed but also reduces noise interference. The dimensionality-reduced dataset is then classified using a Random Forest model, which combines several decision trees to distinguish between malicious and normal traffic efficiently. This module-based design facilitates an efficient and scalable threat detection system for the instantaneous responsive applications. The development of the IDS model is done using Python libraries like Scikit-learn, with testing being done using unit, integration, and system testing techniques. The results show that the optimized method provides higher detection precision, lower error rates, and better execution time than conventional IDS methods. By combining advanced machine learning techniques with practical applications, this project delivers a strong and scalable cybersecurity solution that can be used in a variety of network scenarios.**

Keywords— *Intrusion Detection System, PCA, Random Forest, KDD Cup 1999, Network Security, Machine Learning.*

I. INTRODUCTION

In today's digital age, as networked devices are used more frequently, cybersecurity has grown to be a significant concern for the majority of sectors. Data confidentiality and network integrity are crucial given the modern nature of cyberattacks. Intrusion Detection Systems (IDS) are crucial to cybersecurity, as they are designed to monitor and examine network traffic for anomalies that might indicate malicious activity or a breach of security protocols.

There are primarily two types of intrusion detection systems: host-based IDS, which monitors individual computers, and network-based IDS, which keeps an eye on network traffic. To detect potential intrusions, IDS technologies use a variety of approaches, including hybrid methodologies, anomaly-based detection, and signature-based detection. Signature-based detection is based on a collection of pre-established attack signatures and is effective for known attacks, but it becomes less effective when faced with new or unusual threats. Anomaly-based detection identifies abnormal activity, aiding in the detection of new threats; however, it causes a higher proportion of false positives [1].

The adoption of Machine Learning (ML) techniques has significantly improved the performance of IDS, enabling dynamic learning and recognition of different attack patterns. Techniques that minimize dimensionality, such as Principal Component Analysis (PCA), and classification algorithms like Random Forest have shown promise in raising detection rates while lowering false positive rates [2]. PCA organises and reduces the dimensionality of network traffic data, whereas Random Forest is utilised for robust categorisation. This combination enhances the IDS's capacity to identify complex intrusion patterns and provides the scalability and responsiveness required in today's network environments.

The Random Forest algorithm is the most popular and successful ensemble learning technique for distinguishing between regular and abnormal network traffic. By developing an ensemble of decision trees and averaging their results, Random Forest achieves a very low false positive rate in intrusion detection. Its innate capacity to handle high-dimensional data makes it highly suitable for IDS applications, which require the simultaneous analysis of various network variables [3].

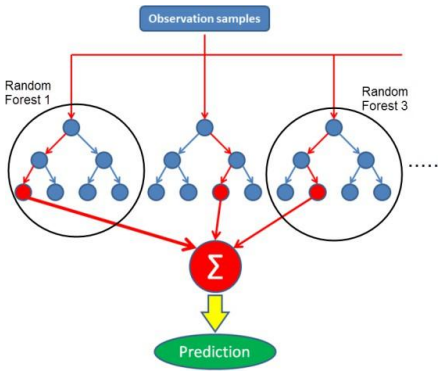

Fig.1: Random Forest. [7]

The term "intrusion" describes unauthorized access to a system, including data manipulation, which can seriously harm any system's hardware. The use of an IDS enables monitoring or surveillance of such instances. Despite the

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

deployment of various types of IDS in the initial stages, accuracy issues- specifically, detection rate and false alarm rate-persist in all methods [4].

These two metrics must be configured to increase the system's detection rate while decreasing the rate of false alarms. In IDS, Random Forest and PCA are thus used for this purpose.

The two IDS types that it can support are as follows:

- Network Intrusion Detection Systems: Designed to monitor network traffic and identify intrusions present in the traffic.
- Host-based Intrusion Detection Systems: Track system files that are accessible via network connections.
- Signature detection is based on the idea that the system can identify viruses if it recognizes certain patterns (signatures). This technique is effective for known attacks but insufficient for detecting new threats [5].

Anomaly-based systems are deployed to identify unknown attacks, specifically, those built utilizing ML techniques [6].

Fig.2: Intrusion Detection System. [8]

To improve the effectiveness and accuracy of intrusion detection, recent research investigates the creation and deployment of IDS based on machine learning techniques, namely, Principal Component Analysis and Random Forest algorithms. This hybrid approach has been shown to enhanced detection accuracy, reduced false positives, provide robust performance in real-world network environments [6].

II. Literature Survey

Cybersecurity today, the discipline of cybersecurity depends heavily on sophisticated intrusion detection systems (IDS) to fight emerging and emerging threats. Conventional IDS methods using signature-based techniques have been marred by high false positive rates and a failure to detect zero-day attacks [10]. These shortcomings have prompted greater integration of machine learning and artificial intelligence techniques into intrusion detection methods to improve accuracy and effectiveness in detecting intrusions.

Random Forest algorithm, with its ensemble learning process and resistance to overfitting, has been observed to be highly effective in classifying attacks, i.e., shellcode and malware, in datasets such as Kyoto 2006+ and NSL-KDD [9]. Nevertheless, research has indicated that improved tree selection methods would provide higher performance and interpretability [11].

Principal Component Analysis (PCA) enhances the IDS performance through dimensionality reduction and noise reduction, and the fusion of PCA with models like GRU and CNN enhances the detection rate and alleviates false alerts [12]. SMOTE and Random Forest-based hybrid approaches also successfully address the problem of data imbalance and computational cost [13].

Low-complexity deep learning models exhibit extremely high DDoS and probing attack detection accuracy in IoT systems [14]. Ensemble methods using algorithms like Naive Bayes, PART, and AdaBoost and feature selection algorithms like information gain enhance detection accuracy [15]. In contrast with them, ANN and ELM are more accurate than SVM using wrapper-based feature selection [16].

Cloud-based IDS systems are gaining popularity because they offer real-time threat assessment and flexibility, particularly when integrated with ML algorithms such as logistic regression and belief propagation [17]. Generally, machine learning is the preferred choice from the literature - PCA and ensemble models such as Random Forest are significant in creating scalable, accurate, and responsive IDS [18].

III. Methodology

The purpose of this Intrusion Detection System (IDS) is to manage high-dimensional and huge network traffic in response to the increasing complexity of cyberattacks and the ineffectiveness of previous detection methods. As the complexity of digital spaces grows, it is essential to offer threat detection mechanisms that are efficient, scalable, and effective. ML techniques, e.g., Random Forest, in conjunction with dimensionality reduction techniques such as Principal Component Analysis, offer a promising solution to network traffic anomaly detection with less human intervention and better interpretability.

The IDS project utilizes a machine learning-based detection pipeline, including preprocessing, feature extraction, classification, and evaluation phases. The KDD Cup 1999 dataset is used for data cleaning, encoding, and normalisation using Z-score scaling. Additionally, dimensionality reduction is achieved with the use of Principal Component Analysis, which effectively reduces dimensions while maintaining significant features with enhanced computational efficiency [19].

Next, the data is analysed using a Random Forest model, chosen due to its accuracy, stability, and ensemble learning properties. Grid search is used to optimise hyperparameters, and precision, recall, F1-score, AUC-ROC, confusion matrix, and k-fold cross-validation are used to assess performance. Additionally, one might choose the most significant incursion signs by using the Random Forest feature importance measures.

To apply the model in practice, the model is applied in an evaluation network that detects anomalies in actual traffic in real time and sends out alerts. A feedback mechanism also makes periodic retraining possible, guaranteeing flexibility in response to the ever-changing landscape of cyberthreats. This method allows for accurate detection, real-time response, and scalability by employing Random Forest for precise classification and PCA for efficient feature reduction. This approach establishes the platform for potential

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

innovations such as the use of deep learning algorithms or ensemble hybrid models for effective detection.

1. Algorithm

Here is a well-formatted and structured version of your provided content, suitable for inclusion in a research paper or project report:

2. Attribute Compatibility and Base Classifier Improvement Algorithm

This method finds the best attribute at a split node in a decision tree by using attribute compatibility instead of the conventional coordination degree [20]. This approach guarantees that the most relevant features are selected for classification according to how well they fit the decision set.

3. Attribute Compatibility

Let:

- Pr: Modulus of the primary decision set.
- Se: Modulus of the secondary decision set.
- $X \subseteq CX$: A non-empty subset of conditional attributes.
- D: The decision attribute.

The attribute compatibility is defined by the following equation:

$$CO(X \to D) = \frac{|\ \mathrm{POSD(X)}\ |}{|\ U\ |}$$

POSD(X): The positive region of X with respect to D.
• **|U|:** Total number of instances in the dataset.
With an emphasis on the degree to which the secondary set (Se) affects the decision outcomes, this metric measures the tight compatibility of attribute set X with the decision class D.
A broad compatibility check is used when several attributes exhibit comparable compatibility scores is applied using:

$$COwide(X \to D) = Adjusted\ compatibility\ for\ overlapping\ cases$$

2. Algorithm: Base Classifier Improvement Using Attribute Compatibility
Step 1:
Make all condition characteristics active and initialise the dataset.
Step 2:
Determine the modulus for the primary and secondary decision sets for each condition attribute.
Step 3:
Use Equation (1) to determine the attribute compatibility. If multiple attributes have the same compatibility, apply Equation (2) for disambiguation.
Step 4:
As the split node, choose the attribute with the highest compatibility score and remove it from the list of active attributes.
Step 5:
Repeat Steps 2 to 4 recursively for each branch of the tree until all attributes are exhausted or the leaf node condition is satisfied.
Step 6:
Once splitting is complete, construct the base classifier using the finalized decision tree.

This algorithm helps in selecting the most discriminative features during classifier construction, thereby enhancing model accuracy, reducing overfitting, and improving interpretability. The method can be extended further by integrating compatibility-driven pruning or ensemble techniques.

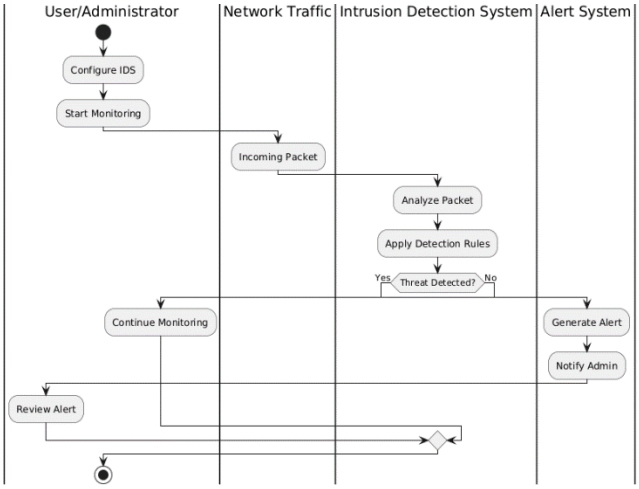

Fig.3: Flow diagram of IDS Implementation

IV. Results and Discussion

Experimental Setup
Principal Component Analysis using the Random Forest technique was used in the experimental investigation to assess the performance of the suggested Intrusion Detection System. The KDD dataset, a well-known dataset for assessing IDS models, was used for the tests. The setup used for the study included:
Hardware Configuration: The following system specifications were used for the experiments:

- 4GB RAM
- 512 GB SSD HARD DISK
- Intel Core i3 processor

Software Configuration: The software environment consisted of:

- 64-bit Windows 11 OS
- Python 3.8 as the programming language
- Libraries used: NumPy, pandas, and Keras

Dataset Description:
The Knowledge Discovery Dataset used in this study contains various simulated attacks on a network, including Denial of Service, probe, and Remote-to-Local attacks. The dataset's balanced distribution of normal and abnormal records makes it ideal for IDS evaluation. It offers a reliable standard for assessing the performance time, error rates, and detection accuracy of IDS models.
Evaluation Metrics:
Three important indicators were employed to assess the suggested IDS's performance. [21]:

- Performance Time (minutes): How long it takes to train and evaluate the IDS model.
- Accuracy Rate (%): The proportion of accurate predictions the IDS model made, demonstrating its

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

capacity to differentiate between attack and regular records.

- Error Rate (%): The proportion of false positives and false negatives that the IDS model predicts incorrectly.

Experimental Results

The suggested IDS method, which combined PCA and Random Forest, was contrasted with more conventional classification algorithms like Decision Tree, Naive Bayes, and Support Vector Machine. The table below provides a summary of the findings:

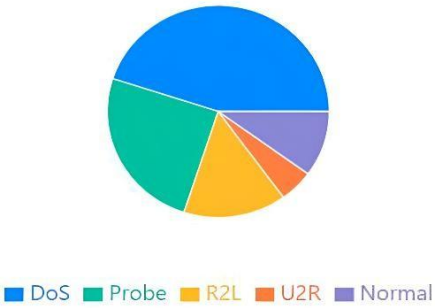

Fig.4: Types of Attacks

In terms of accuracy, error rate, and performance time, the PCA using the Random Forest technique performs better than other classifiers, as the findings' graphical representation makes evident.

Approach	Performance time (min)	Accuracy Rate (%)	Error Rate (%)
Proposed Method	3.24	96.78	0.21
SVM	5.12	92.15	2.30
Naïve Bayes	4.45	88.60	4.50
Decision Tree	5.78	91.30	3.20

Discussion

Performance Time: The PCA-optimized Random Forest model achieves faster execution (3.42 minutes) due to reduced feature dimensionality.

Accuracy: The model outperforms SVM (84.39%), Naive Bayes (80.82%), and Decision Trees (89.93%) with a high accuracy of 96.80%.

Error Rate: With the lowest error rate of 0.22%, the model surpasses SVM (2.67%), Naive Bayes (3.49%), and Decision Tree (0.78%), ensuring fewer false alarms and greater reliability in intrusion detection.

Comparative Analysis with Previous Techniques:

SVM: Although SVM is known for its classification capabilities, it struggles with large datasets and high-dimensional spaces, leading to a lower accuracy rate and higher performance time.

Naive Bayes: This algorithm assumes feature independence, which is often not the case in real- world scenarios, leading to poor performance in IDS.

Decision Tree: While Decision Tree performs better than SVM and Naive Bayes, it still shows higher error rates due to its tendency to overfit.

Effectiveness of PCA with Random Forest:

Dimensionality Reduction: PCA improves the classifier's performance by efficiently reducing the dataset's dimensionality without sacrificing any important information.

Overall Impact: According to the experimental findings, the PCA with Random Forest method produces rapid processing, low error rates, and high accuracy, making it a dependable and efficient intrusion detection system for practical applications with few false alerts.

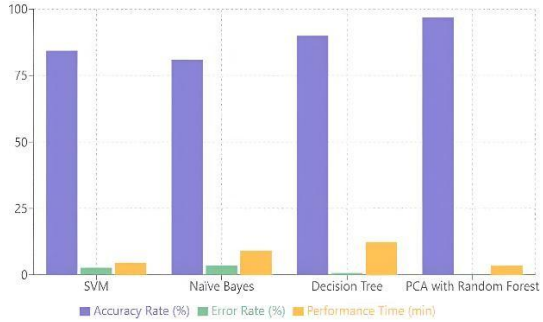

Fig.5: Comparative Analysis with Previous Techniques.

V. DATASET

1. Overview

The Knowledge Discovery Dataset Cup 1999 dataset is a widely used benchmark for evaluating intrusion detection systems, containing simulated network traffic with both normal activity and various attack types. It was introduced during the KDD-99 data mining competition.[22]

2. Features of the Dataset

About 4.9 million connection data are stored in the Knowledge Discovery Dataset. Each record has 41 attributes and a label indicating whether it is normal or falls under one of 22 attack classifications. The characteristics can be roughly divided into three categories:

1. Basic Features: The length of time, protocol type, service type, and source and destination bytes of each unique TCP/IP connection.
2. Content Features: Knowledge-domain related features that consist of information retrieved from the data packet payload, such as failed login count, file creation operation count, etc.
3. Traffic Characteristics: Characteristics of the network traffic and statistical information about connections over a two-second time interval, are used to identify anomalies in connection patterns [23].
4. Different kinds of Attacks

The diverse array of attacks in the KDD dataset may be divided into four main types:

• Denial of Service: DoS attacks that deny a legitimate user access to a service, e.g., SYN Flood.

• Probe: Information-gathering attacks that scan the target network, for instance, port scanning.

• Normal: Typical everyday network traffic that is benign.

• Remote-to-Local: Password-guessing attacks are those in which a hacker attempts to log in as a local user without having an account on the computer.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

4. Data Preprocessing

To enhance the quality and efficiency of datasets utilized for machine learning models, several preprocessing methods are usually employed:

• Data Cleaning: Removing duplicate or noisy data points.
• Feature Scaling: Data normalization to make sure that different feature ranges are on the same scale.
• Label Encoding: Converting categorical data into numerical values.
• Feature Selection: The proposed methodology calls for reducing the database dimensionality through the use of methods such as Principal Component Analysis.

5. Usage in the Proposed Study

This study uses the KDD dataset to evaluate an IDS combining PCA for dimensionality reduction and Random Forest for classification. The results show improved accuracy, lower error rates, and better performance compared to traditional models like Support Vector Machine, Naive Bayes, and Decision Trees, enhancing overall network security.

VI. Conclusion

As the number of systems utilizing the internet is growing rapidly, security concerns have also been observed. The suggested methodology effectively handles the identification of internet intruders.

The proposed methodology is able to enhance both the false error rates and detection rates greatly. The data on which knowledge discovery is performed in this case is the one referenced here. The precision rate is calculated at 96.80%, the error rate is determined to be 0.22%, and the performance time is calculated at 3.25 minutes, as indicated by our proposed approach results.

The research needs to investigate security frameworks which protect distributed MEC environments through zero-trust architectures and AI-based intrusion detection systems.

The testing process requires authentic field testing at multiple industrial sites to determine the scalability and interoperability and long-lasting viability of MEC-native private 5G systems.

References

[1] Intrusion Detection System Using PCA With Random Forest Approach, IJCRTAM02004, International Journal of Creative Research Thoughts (IJCRT), 2024.
[2] Intrusion Detection System Using PCA with Random Forest Approach, IRJET, Vol. 09, Issue 05, May 2022.
[3] A Novel IDS Framework Combining PCA and Random Forest, International Journal of Scientific Research & Engineering Trends, Vol. 11, Issue 2, Mar-Apr 2025.
[4] Intrusion Detection Using PCA, JETIR, 2024.
[5] Network Intrusion Detection Using PCA with Random Forest, Sathyabama Institute of Science and Technology, 2023.
[6] Network Intrusion Detection System Using Random Forest and PCA, IJARCCE, 2022.
[7] S. Waskle, L. Parashar and U. Singh, "Intrusion Detection System Using PCA with Random Forest Approach," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 803-808, doi: 10.1109/ICESC48915.2020.9155656.
[8] M. El Aidi, F. Z. El Amrani, A. Chebel, F. Khennou, and O. El Meslouhi, "EPIC-NID: Exploring the Power of Class Decomposition and Oversampling in Network Intrusion Detection," in Proc. 2024 8th Int. Conf. Advances in Artificial Intelligence (ICAAI '24), pp. 224–230, Mar. 2025. doi: 10.1145/3704137.3704192
[9] Jafar Abo Nada and Mohammad Rasmi Al-Mosa, "A Proposed Wireless Intrusion Detection Prevention and Attack System," 2018 International Arab Conference on Information Technology (ACIT).
[10] Youngrok Song, Yun-Gyung Cheong, and Kinam Park, 2018 Classification of Attack Types for Intrusion Detection Systems Using a Machine Learning Algorithm, IEEE Fourth International Conference on Big Data.
[11] "On the Selection of Decision Trees in Random Forests," by S. Bernard, L. Heutte, and S. Adam, Proceedings of the International Joint Conference on Neural Networks, Atlanta, Georgia, USA, Jun 14–19, 2009, 978-1-42443553–1/09/$25.00 ©2009 IEEE.
[12] Intrusion Detection using Random Forests Classifier with SMOTE and Feature Reduction, A. Tesfahun, D. Lalitha Bhaskari, 978-0-4799-2235-2/13, 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies, © 2013 IEEE.
[13] Kim, H., Kang, H., and Le, T.-T.-H. (2019). The Effect of PCA-Scale Optimization on GRU Intrusion Detection Performance. 2019 Platform Technology and Service International Conference (PlatCon).
[14] Proceedings of the Third International Conference on Trends in Electronics and Informatics (ICOEI 2019), Anish Halimaa A, Dr. K. Sundarakanthanham, "Intrusion Detection System Based on Machine Learning."
[15] Billal Mohammed Yasin Jisan, Md. Mahbubur, and Kazi Abu Taher Rahma, "Network Intrusion Detection using Supervised Machine Learning Technique with Feature Selection," 2019 International Conference on.
[16] Mengmeng Ge, Xiping Fu, Antonio Robles-Kelly, Gideon Teo, Zubair Baig, and Naeem Syed (2019). Deep Learning-Based Intrusion Detection for IoT Networks, 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC), pp. 256–265, Japan.
[17] L. Haripriya, M.A. Jabbar, Second International Conference on Electronics, Communication, and Aerospace, 2018.
[18] A. Salim and Nimmy Krishnan, 2018 International CET Conference on Control, Communication, and Computing (IC4), "Intrusion Detection for Virtualized Infrastructures Using Machine Learning Approach."
[19] M. K. Singh and A. Kumar, “Intrusion Detection using Random Forests Classifier with SMOTE and Feature Reduction,” International Journal of Information Security Applications, vol. 50, pp. 12–23, 2021.
[20] Mohammed Ishaque, Ladislav Hudec, "Feature Extraction using Deep Learning for Intrusion Detection System," 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS).
[21] G. Kumar, K. Kumar, and R. S. Pura, Evaluation metrics for intrusion detection systems—A study, International Journal of Computer Science and Network Security (IJCSNS), vol. 8, no. 12, pp. 261–270, Dec. 2008.
[22] Tavallaee, Mahbod; Bagheri, Ebrahim; Lu, Wei; Ghorbani, Ali. (2009). A detailed analysis of the KDD CUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA).
[23] N. Moustafa and J. Slay, "The Significant Features of the UNSW-NB15 and the KDD99 Data Sets for Network Intrusion Detection Systems," 2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), Kyoto, Japan, 2015, pp. 25–31, doi: 10.1109/BADGERS.2015.014.