

# DETECTION OF CYBER ATTACKS IN NETWORK USING MACHINE LEARNING

Muneshwar Tejashwini<sup>1</sup>, Muneshwar Pavan Kalyan<sup>2</sup>, K. Balakrishna Maruthiram<sup>3</sup>

<sup>1</sup>Post Graduate Student, M. Tech(CNIS), Department of Information Technology, Jawaharlal Nehru Technological University, Hyderabad, India

<sup>3</sup>Assistant Professor of CSE, Department of Information Technology, Jawaharlal Nehru Technological University, Hyderabad, India

**Abstract—** Cybersecurity is one of the major challenges in today's digital world due to the increasing number of network-based attacks. Traditional security systems such as firewalls and antivirus software are no longer sufficient to detect sophisticated cyber-attacks. This paper presents a machine learning-based approach to detect different types of cyber attacks using the Network Security Laboratory - Knowledge Discovery in Databases(NSL-KDD) dataset. Various classification algorithms including Random Forest(RD), Decision Tree(DL), Support Vector Machine (SVM), and K-Nearest Neighbour (KNN) are applied and evaluated. The proposed method enhances the detection accuracy of cyber attacks and classifies them into categories like Denial of Service(DoS), Probing Attack(Probe), Remote to Local(R2L), and User to Root(U2R). The results indicate that machine learning techniques are effective in improving intrusion detection systems (IDS) with better accuracy and efficiency.

**Keywords:** *Cybersecurity, Intrusion Detection System (IDS), Machine Learning(ML), NSL-KDD, Network Security, Cyber Attacks, DoS, Probe, R2L, U2R, Cyber Attack Detection, Supervised Learning, Classification, Random Forest, Decision Tree, Logistic Regression, Support Vector Machine.*

## I. INTRODUCTION

In today's hyper-connected digital world, cybersecurity is one of the most critical challenges faced by organizations and individuals alike. Every year, cyber threats become more advanced and harder to detect, posing significant risks to data integrity, confidentiality, and availability. Intrusion Detection Systems (IDS) have traditionally played a crucial role in identifying and responding to malicious network activity. However, conventional IDS approaches such as signature-based and rule-based systems often fail to detect new or previously unknown attacks (zero-day attacks).

Machine learning offers a promising solution by enabling systems to learn from data and identify patterns that signify malicious behaviour. By analysing historical network traffic and attack data, machine learning models can be trained to recognize anomalies and classify them into specific types of attacks. This paper explores the use of machine learning algorithms in building an effective IDS using the NSL-KDD dataset, which is a refined version of the popular KDD Cup 1999 dataset. The aim is to evaluate the effectiveness of various ML algorithms in detecting and classifying cyber attacks and to propose an optimal model for real-world deployment.

## II. LITERATURE SURVEY

Cyber attacks have become a major cybersecurity concern, prompting extensive research into detection methods. Various approaches, including Rule-Based methods, Machine Learning and Deep Learning, have been explored to enhance cyber attack detection. This survey categorizes and discusses key contributions in these areas.

### *A. Signature-Based vs. Anomaly-Based Detection*

Traditionally, Intrusion Detection Systems (IDS) were designed using signature-based methods, which detect attacks by matching network activity against known patterns. Although accurate for known threats, these systems fail to identify zero-day attacks and variations of existing exploits.

Anomaly-based detection systems, by contrast, learn patterns of normal behaviour and flag deviations as potential intrusions. Denning (1987) was among the first to propose a model for anomaly detection based on statistical profiling. This concept laid the foundation for modern machine learning-based IDS, which benefit from generalization, adaptability, and the ability to detect previously unseen attack vectors.

However, anomaly-based systems tend to produce high false positive rates if the boundary between normal and abnormal behaviour is not well defined, highlighting the need for precise models and balanced datasets.

### *B. Machine Learning-Based Approaches*

Numerous studies have applied supervised machine learning algorithms to the task of intrusion detection using labeled datasets such as KDD Cup 1999, NSL-KDD, UNSW-NB15, and CICIDS2017.

#### *1. Decision Tree and Random Forest*

Decision Trees (DT) are widely used due to their interpretability and low computational cost. In one study, Revathi and Malathi (2013) used J48 (a variant of C4.5) on the NSL-KDD dataset and reported over 90% accuracy for binary classification of normal vs. attack traffic. However, Decision Trees are prone to overfitting when trained on high-dimensional datasets.

Random Forests (RF), an ensemble of decision trees, overcome this limitation by reducing variance and improving robustness. Meera Gandhi and Ramesh Babu (2016) applied RF to detect DoS and Probe attacks and achieved superior accuracy compared to single classifiers. RF models also allow computation of feature importance, aiding in feature selection and model interpretability.

#### *2. Support Vector Machine (SVM)*

SVM is a powerful binary classifier that constructs a hyperplane to separate different classes in high-dimensional space. Mukkamala et al. (2005) applied SVMs to intrusion detection and observed better generalization for minority classes like U2R and R2L. However, the high training complexity of SVM makes it less suitable for large-scale or real-time systems.

#### *3. K-Nearest Neighbors (KNN)*

KNN is a distance-based classifier that assigns labels based on the majority class among its k-nearest neighbors. Despite its simplicity, KNN has been applied successfully in network intrusion detection tasks, as demonstrated by Dhanabal and Shantharajah (2015), who used it on the NSL-KDD dataset. However, KNN's performance is sensitive to the choice of distance metric and the value of k, and it becomes computationally expensive for large datasets.

### *C. Deep Learning-Based Approaches*

Deep learning (DL) offers superior performance in tasks involving complex and high-dimensional data by automatically learning hierarchical feature representations. Unlike traditional ML models, deep neural networks eliminate the need for manual feature engineering.

### 1. *Artificial Neural Networks (ANN)*

ANNs have been applied to classify traffic as either normal or malicious with considerable success. However, shallow ANNs suffer from limited capacity and often struggle with imbalanced data.

### 2. *Convolutional Neural Networks (CNN)*

CNNs, typically used in image recognition, have been adapted to extract spatial patterns in traffic flow or feature maps derived from network packets. Yin et al. (2017) proposed a CNN-based IDS model that achieved high detection accuracy and generalizability on the NSL-KDD dataset. CNNs are effective in processing structured inputs such as packet-level or flow-level features, although they may require transformation of tabular data into image-like formats.

### 3. *Recurrent Neural Networks (RNN)*

RNNs and their variants, such as Long Short-Term Memory (LSTM) networks, are capable of modeling temporal dependencies in sequential data, making them well-suited for analyzing network flows over time. Shone et al. (2018) applied a deep autoencoder with LSTM layers to detect advanced persistent threats and achieved improved detection rates with fewer false positives.

Although DL models outperform traditional methods in many cases, they are computationally expensive and require large volumes of labeled training data. Furthermore, they may act as black boxes, lacking transparency and interpretability for security analysts.

### D. *Ensemble and Hybrid Models*

To balance accuracy and robustness, hybrid and ensemble models combine multiple classifiers or learning paradigms. Arafat et al. (2020) developed a hybrid IDS combining Random Forest and Naive Bayes, achieving improved precision and recall for minority attack classes like R2L and U2R. Similarly, Kim et al. (2018) used a combination of SVM and KNN, with a feature selection stage based on Information Gain.

Ensemble models such as boosting and bagging have also shown promise. AdaBoost and Gradient Boosting Machines (GBM) have been employed to combine weak learners into strong classifiers, achieving higher accuracy and better generalization.

Hybrid deep learning models, combining CNNs for spatial analysis and LSTMs for temporal learning, have emerged as powerful tools in modern intrusion detection. However, they face challenges including increased model complexity, tuning difficulties, and the need for real-time adaptability.

### E. *Feature Selection and Data Preprocessing*

The quality of input features significantly influences the performance of ML-based IDS. Various studies have employed dimensionality reduction techniques like Principal Component Analysis (PCA), Information Gain, and Recursive Feature Elimination (RFE) to improve model efficiency and reduce overfitting.

Tavallaee et al. (2009) introduced the NSL-KDD dataset as an improved version of KDD'99 to mitigate data redundancy and imbalance. Their work emphasized the importance of well-structured and representative datasets in training reliable detection models.

Preprocessing steps such as encoding categorical variables, normalizing numerical features, and handling class imbalance (using SMOTE or undersampling) are critical for optimizing ML algorithms.

#### *F. Challenges and Limitations*

While machine learning techniques have significantly advanced the field of intrusion detection, several challenges persist:

- High false positive rates in anomaly-based models.
- Poor performance on minority classes (e.g., U2R and R2L).
- Lack of real-time adaptability in complex models.
- Computational overhead in training and inference.
- Vulnerability to adversarial attacks and data poisoning.

Efforts to address these challenges have led to innovations in adversarial training, online learning, explainable AI (XAI), and federated learning. However, further research is required to integrate these solutions into scalable and deployable IDS platforms.

#### *Conclusion of Literature Review*

The literature highlights the evolution of cyber attacks detection systems from rule-based to intelligent, data-driven approaches. While traditional ML models such as Random Forest and SVM offer simplicity and interpretability, deep learning methods provide higher detection accuracy and the ability to process complex network behaviour. Hybrid and ensemble models show promise in combining the best of both worlds, albeit with added complexity. Building on this foundation, the present research investigates the comparative performance of individual ML models and their hybrid combinations for cyber attack detection, using the NSL-KDD dataset as a testbed.

### **III. PROPOSED WORK**

The proposed work aims to design and implement an intelligent and accurate intrusion detection system (IDS) for detecting cyber attacks in network traffic using machine learning techniques. The system focuses on distinguishing between normal and malicious traffic and further classifying the types of attacks based on patterns learned from the NSL-KDD dataset.

#### *A. Dataset Description*

The dataset used in this project is a NSL-KDD dataset is an enhanced and refined version of the KDD Cup 1999 dataset, widely used for evaluating intrusion detection systems (IDS). It addresses several issues in the original KDD dataset, like redundancy and imbalance, which could lead to biased machine learning results. Dataset containing 1,48, 517 entries, 41 input features and 1 target label(attack class) in which 125,973 rows used for training and 22,544 rows used for testing.

- 41 input features (both continuous and categorical).
- Class labels: normal and attack (further categorized into DoS, Probe, R2L, and U2R).
- Examples of attacks:
  - DoS: neptune, smurf, back
  - Probe: portsweep, nmap, satan
  - R2L: guess\_passwd, ftp\_write

- U2R: rootkit, buffer\_overflow

The dataset is divided into KDDTrain+ and KDDTest+ for training and testing, respectively, ensuring a proper evaluation of model performance.

Key features in the dataset include:

The features can be grouped into the following categories:

1. **Basic Features** (9 attributes): These are derived from packet headers and represent basic connection information.
2. **Content Features** (13 attributes): These analyze the content of the data in the connection.
3. **Time-Based Traffic Features** (9 attributes): These analyze traffic behavior over 2-second windows.
4. **Host-Based Traffic Features** (10 attributes): These look at behavior based on the same host over a time window.
5. **Target Label** (Class Attribute): Either **normal** or a **type of attack**.

duration	Protocol_type	service	src_bytes	dst_bytes	flag	logged_in	label	Attack type
0	icmp	telnet	181	540	sf	1	Normal	Normal
0	tcp	http	6	900	sf	0	Satan	Probe
0	tcp	http	239	486	sf	1	Neptune	DoS
0	tcp	smtp	190	0	sf	0	Guess_password	R2L
0	tcp	domain	0	0	sf	0	Rootkit	U2R

Table-1:Dataset Description

## B. Architecture

The General Architecture of this project is as mentioned below,.

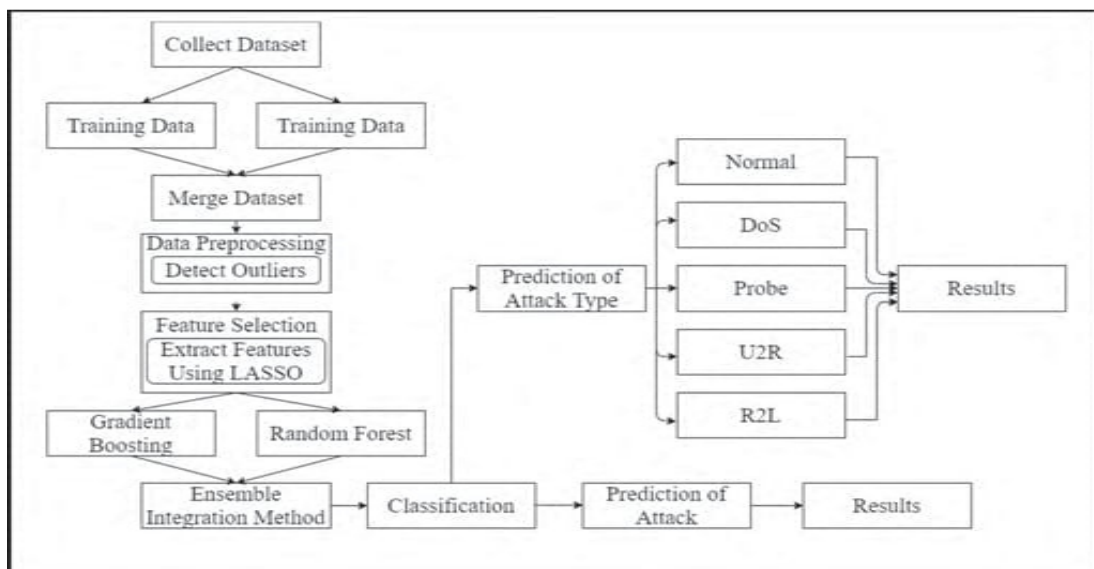


Figure-1: General Architecture

### C. Methodology

The entire project methodology is as follows:

The proposed system involves several key stages:

#### 1 Data Preprocessing

The NSL-KDD dataset contains 41 features along with a class label indicating whether a record is normal or an attack. Preprocessing includes:

- Converting categorical attributes such as protocol\_type, service, and flag into numerical format using label encoding.
- Normalizing numerical features to bring them to a similar scale.
- Handling class imbalance by applying sampling techniques if necessary.

#### 2 Feature Selection

To reduce complexity and enhance model performance, feature selection techniques such as correlation analysis and feature importance ranking (using Random Forest) are employed. Only the most relevant features are retained for model training.

#### 3 Model Building

Several machine learning classifiers are trained and evaluated, including:

- Decision Tree (DT): A tree-based model that splits data based on feature thresholds.
- Random Forest (RF): An ensemble of decision trees providing better generalization and robustness.
- Support Vector Machine (SVM): A powerful classifier suitable for binary and multi-class problems.
- K-Nearest Neighbors (KNN): A simple, instance-based algorithm that classifies based on proximity to training samples.

#### 4 Evaluation Metrics

The models are assessed using metrics such as:

- Accuracy: Proportion of correct predictions.
- Precision: Proportion of true positives among predicted positives.
- Recall: Proportion of true positives among actual positives.
- F1-Score: Harmonic mean of precision and recall.

## IV. EXPERIMENTAL ANALYSIS AND RESULTS

### A. Performance Metrics

After training the models on the preprocessed NSL-KDD dataset, the following results were obtained:

#### Execution Results:

Model	Accuracy(%)	Precision(%)	Recall(%)	F1 Score(%)
Random Forest	99.72	99.84	99.56	99.70
Decision Tree	99.51	99.49	99.46	99.47
SVM	99.03	99.39	98.54	98.96
KNN	99.42	99.46	99.30	99.38

Table-2: Results

Random Forest outperformed the other models due to its ensemble nature and ability to reduce overfitting. It achieved the highest accuracy and F1-score across all attack categories.

### B. Output Screens

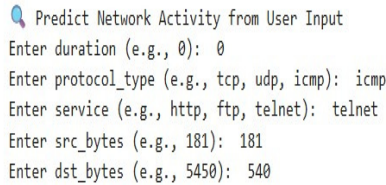

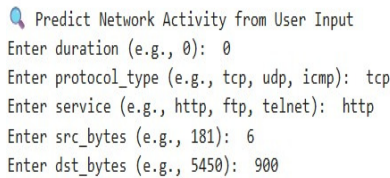

 <p>Predict Network Activity from User Input  Enter duration (e.g., 0): 0  Enter protocol_type (e.g., tcp, udp, icmp): icmp  Enter service (e.g., http, ftp, telnet): telnet  Enter src_bytes (e.g., 181): 181  Enter dst_bytes (e.g., 5450): 540</p> <p> Predicted Network Activity: normal</p>	 <p>Predict Network Activity from User Input  Enter duration (e.g., 0): 0  Enter protocol_type (e.g., tcp, udp, icmp): tcp  Enter service (e.g., http, ftp, telnet): http  Enter src_bytes (e.g., 181): 6  Enter dst_bytes (e.g., 5450): 900</p> <p> Predicted Network Activity: satan</p>
--	---

Figure-2: Network Activity as Normal

Figure-3: Network Activity as Probe(satan)

## V. CONCLUSIONS

This study highlights the importance of machine learning in enhancing network intrusion detection systems. By applying classification algorithms to the NSL-KDD dataset, we demonstrated that Random Forest provides superior performance in detecting various cyber attacks. Although Decision Trees and SVMs also show promising results, they are relatively less accurate for complex attack types such as U2R and R2L.

In future work, deep learning techniques such as Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) can be explored to capture temporal patterns in network traffic. Additionally, applying real-time data streaming and integrating with Security Information and Event Management (SIEM) tools could make these systems more adaptive and practical for enterprise use.

## REFERENCES

- [1] K. B. Maruthiram and G. Vijayakrishna, "Tackling Cyber Hatred with Machine Learning and Fuzzy Logic," *International Journal of Innovative Research in Technology*, vol. 11, no. 6, p. 2034, 2024.
- [2] K. B. M. Bushra Fatima, "Detection and Classification of Malicious Software Using Machine Learning and Deep Learning," *International Journal of Innovative Research in Technology*, vol. 11, no. 2, pp. 1812–1816, 2024.

- [3] K. S. B. Katroth Balakrishna Maruthiram, "Effect of MANETS With and Without Malicious Node," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 5, 2014.
- [4] K. B. M. Kokkala Rachana, "Machine Learning Safeguards: Network Attack Detection," *Journal of Emerging Technologies and Innovative Research*, vol. 11, no. 8, p. e832, 2024.
- [5] M. R., K. B. Maruthiram, and G. Venkatarami Reddy, "Secure and Efficient OutSourced Clustering Using K-Mean with Fully Homomorphic Encryption by Ciphertext Packing Technique," *International Journal of Innovative Research in Technology*, vol. 11, no. 2, pp. 637–644, 2024.
- [6] M. K. A., K. B. Maruthiram, and G. Venkatarami Reddy, "Predicting Students Results Based on Study Hours Using Machine Learning," *International Journal of Innovative Research in Technology*, vol. 11, no. 2, p. 1006, 2024.
- [7] K. B. Maruthiram, "A Framework for Early-Stage Detection of Autism Spectrum Disorders Utilizing Machine Learning," *International Journal of Research and Analytical Reviews*, vol. 11, no. 5, p. 881, 2024.
- [8] K. B. M. Ryan Husain, "Multi-Sensor based Physical Activity Recognition and Classification Using Machine Learning Techniques," *International Journal of Creative Research Thoughts*, vol. 12, no. 7, pp. h809–h814, 2024.
- [9] K. B. M. Rathod Sai Vamshi Krishna, "Advance Genome Disorder Prediction Model Empowered with Machine Learning," *International Journal of Creative Research Thoughts*, vol. 12, no. 7, pp. h797–h802, 2024.
- [10] K. B. M. Pillalamarri Veena, "Unveiling Chronic Stress: A Social Media Perspective Using Machine Learning," *International Journal of Innovative Research in Technology*, vol. 11, no. 3, pp. 733–739, 2024.
- [11] M. K., K. B. Maruthiram, and G. Venkatarami Reddy, "Real VS AI Generated Image Detection and Classification," *International Journal of Innovative Research in Technology*, vol. 11, no. 2, p. 1076, 2024.
- [12] K. B. Maruthiram and R. Muralikrishna, "Augmented Attention: Enhancing Morph Detection in Face Recognition," *International Journal of Innovative Science and Research Technology*, vol. 9, no. 8, 2024.
- [13] V. K. Katroth Balakrishna Maruthiram, "Optimizing Human Face Detection with Multi-Intensity Image Fusion in Deep Learning," *International Journal of All Research Education and Scientific Methods*, 2024.
- [14] K. B. Maruthiram, "Robust Encryption and Access Control Mechanisms for Ensuring Confidentiality in Cloud-Based Data Storage," *IN Patent*, no. 10, 2024.
- [15] K. B. Maruthiram, J. M., "Advanced Secure Campus Network System Design and Implementation Using Cisco Packet Tracer," *International Journal of Creative Research Thoughts*, vol. 12, no. 7, 2024.
- [16] Tavallaei, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. *IEEE Symposium on Computational Intelligence for Security and Defense Applications*.



- [17] Dhanabal, L., & Shantharajah, S. P. (2015). A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*.
- [18] Revathi, S., & Malathi, A. (2013). A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. *International Journal of Engineering Research & Technology*.
- [19] Mukkamala, S., Sung, A. H., & Abraham, A. (2005). Intrusion detection using ensemble of soft computing paradigms. *Network and Computer Applications*.
- [20] Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*.
- [21] Rastogi, S. et al. (2022). An Analysis using SVM, KNN, Logistic Regression, RF, NB, Decision Tree. *Journal of Computing Research and Innovation*
- [22] Wu, P., Guo, H., & Moustafa, N. (2020). Pelican: Deep residual network for IDS tested on NSL-KDD.
- [23] Rawat, S., Srinivasan, A., & Vinayakumar, R. (2019). Unsupervised feature learning and DNN vs. classical models on NSL-KDD.
- [24] Kale, R., Lu, Z., Fok, K. W., & Thing, V. L. L. (2022). Hybrid anomaly detection using K-means + GANomaly + CNN on NSL-KDD.
- [25] Silivery, A. K. & Kovvur, R. M. R. (2023). Multi-attack classification with LSTM-RNN on NSL-KDD.
- [26] Khan, Z. A. et al. (2021). Systematic study of ML and DL approaches on NSL-KDD. *Transactions on Emerging Telecommunications Technologies*.
- [27] Raj, S., Jain, M., & Chouksey, P. (2024). Categorical Boosting (CatBoost) for NSL-KDD. *Indian Journal of Cryptography and Network Security*.
- [28] Rawat, S. & Vinayakumar R. (2019). Hybrid DL-FFNN with K-Means & IG feature reduction on NSL-KDD. *Journal of Big Data*.
- [29] Gogoi et al. (2013), Panwar et al. (2014), Ambusaidi et al. (2016), Yin et al. (2017), Xu et al. (2018), etc.—comprehensive comparative study of classical ML on NSL-KDD.
- [30] Gonçalves et al. (2024). XGBoost-LSTM hybrid achieves ~94.4% on NSL-KDD. *Journal of Cloud Computing*.
- [31] Mdpi 2023 comparative survey: Random Forest, KNN, SVM, CNN-LSTM, GRU etc. on NSL-KDD and UNSW-NB15.
- [32] An SFS-based feature selection + ML applied to NSL-KDD for IoT intrusion detection.
- [33] Wu et al. (2020) – “*Network Attacks Detection Methods Based on Deep Learning Techniques: A Survey*” (Security and Communication Networks, 2020). A broad survey of DL methods (autoencoders, CNNs, RNNs, GANs) for network attack detection, detailing architectures and benchmark results.