International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Network Intrusion Detection Using Ensemble Machine Learning echniques

Mr.M.Mahesh Kumar[1]*, Ayush Raj[2], Rayana Lokesh[2], A Vivek Chary[2], Koninti Abhilash[2]

[1]Assistant Professor , [2]UG Students, [1,2] Department of Artificial Intelligence & Machine Learning
[1,2]J. B Institute of Engineering and Technology (UGC-Autonomous), Moinabad, Hyderabad,500075, Telangana.
*Corresponding Author: Mr.M.Mahesh Kumar (mahesh.m528@gmail.com)

ABSTRACT

The rapid growth of networked systems and internet-based services has significantly increased the risk of cyber-attacks, making effective intrusion detection systems (IDS) essential for maintaining network security. Traditional IDS approaches, which rely on signature-based detection, are limited in their ability to identify unknown or evolving threats. To address these challenges, this paper presents a real-time network intrusion detection system based on machine learning techniques for accurate and efficient threat detection.

The proposed system analyzes network traffic data by extracting relevant features from packet flows and classifying them into normal or malicious categories. The model is trained using benchmark datasets such as NSL-KDD and CICIDS2017, which include diverse attack types including Denial of Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R) attacks. Data preprocessing techniques such as normalization and encoding are applied to improve model performance.

To enhance practical usability, the system integrates real-time traffic capture and a web-based dashboard that provides continuous monitoring, visualization, and alert generation. This enables security administrators to detect and respond to threats promptly. Experimental results demonstrate that the proposed system achieves high detection accuracy, precision, and recall while maintaining low latency suitable for real-time deployment.

Key terms covered in the abstract:

- Network security threats and intrusion detection
- Machine learning-based IDS approach
- Network traffic datasets and attack types
- Feature extraction and preprocessing
- Real-time detection with web dashboard
- Performance metrics (Accuracy, Precision, Recall)

1. INTRODUCTION

The rapid expansion of digital communication networks and internet-based services has significantly increased the risk of cyber threats and unauthorized access. Modern organizations rely heavily on networked systems for critical operations, making them vulnerable to various types of cyber-attacks such as Denial of Service (DoS), probing attacks, and privilege escalation attempts. Ensuring the security and integrity of network infrastructure has therefore become a major concern in the field of cybersecurity.

Traditional security mechanisms, including firewalls and signature-based intrusion detection systems, are often insufficient to detect sophisticated and evolving attacks. These systems depend on predefined rules and known attack patterns, making them ineffective against zero-day attacks and unknown threats. Additionally, manual monitoring of network activity is not scalable due to the high volume and velocity of network traffic, leading to delayed detection and response.

To overcome these limitations, machine learning-based intrusion detection systems have emerged as a powerful alternative. These systems analyze network traffic patterns and learn to distinguish between normal and malicious behavior. Unlike traditional approaches, machine learning models can generalize from historical data and identify previously unseen attack patterns, improving detection capability and adaptability.

In this project, we propose a real-time Network Intrusion Detection System (IDS) that leverages machine learning techniques for accurate and efficient threat detection. The system processes network traffic data in the form of packets or flow-based features and classifies them into multiple categories, including normal traffic and various attack types such as DoS, Probe, Remote-to-Local (R2L), and User-to-Root (U2R). Data preprocessing techniques such as feature extraction, normalization, and encoding are applied to enhance model performance.

A key aspect of the proposed system is its ability to handle live network traffic. The system integrates packet capture mechanisms to continuously monitor incoming data streams, enabling real-time detection of suspicious activities. Furthermore, a web-based dashboard is developed to visualize detection results, display traffic statistics, and generate alerts. This allows security administrators to monitor network conditions effectively and respond promptly to potential threats.

The proposed IDS is designed to achieve a balance between detection accuracy and computational efficiency, making it suitable for real-world deployment. Performance is evaluated using standard metrics such as accuracy, precision, recall, and F1-score, ensuring a comprehensive assessment of the system's effectiveness.

In addition to detection accuracy, the scalability and adaptability of intrusion detection systems play a crucial role in their effectiveness within modern network environments. As network infrastructures continue to grow in complexity, IDS solutions must be capable of handling large volumes of high-speed traffic without significant degradation in performance.

The adaptability of the system also allows it to be extended to different network environments, including enterprise networks, cloud infrastructures, and Internet of Things (IoT) ecosystems. With minor modifications, the proposed IDS can be trained on different datasets or customized to detect emerging attack patterns, making it a flexible solution for evolving cybersecurity requirements.

Overall, the proposed system not only improves detection accuracy but also enhances real-time responsiveness, scalability, and usability. These characteristics make it a practical and efficient solution for modern network security challenges, where timely and accurate threat detection is essential for maintaining system integrity and preventing potential damage.

RELATED WORK

Intrusion Detection Systems (IDS) have evolved significantly over the past decades as a fundamental component of network security. Early IDS implementations were primarily signature-based, relying on predefined rules and known attack patterns to detect intrusions. Systems such as Snort became widely adopted due to their effectiveness in identifying known threats. However, signature-based approaches suffer from a major limitation: they are unable to detect zero-day attacks or previously unseen threats, and require continuous updates to maintain their effectiveness.

To address these shortcomings, anomaly-based IDS were introduced. These systems model normal network behavior and identify deviations as potential intrusions. Statistical methods and rule-based anomaly detection techniques were initially used, but they often resulted in high false positive rates due to the dynamic and unpredictable nature of network traffic. This limitation reduced their reliability in real-world deployments.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

With the advancement of machine learning, IDS research shifted toward data-driven approaches capable of learning complex patterns in network traffic. Supervised learning algorithms such as Decision Trees, Support Vector Machines (SVM), and Random Forest have been widely used for intrusion detection tasks. These methods demonstrated improved accuracy compared to traditional techniques, especially when trained on benchmark datasets such as NSL-KDD and KDD Cup 99. Among these, Random Forest gained popularity due to its robustness and ability to handle high-dimensional data, while SVM provided strong classification boundaries for binary and multi-class problems.

Deep learning approaches further enhanced IDS performance by automatically extracting features from raw or minimally processed data. Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) have been explored extensively in recent research. In particular, Long Short-Term Memory (LSTM) networks have shown strong capability in capturing temporal dependencies in sequential network traffic, making them suitable for detecting time-based attack patterns. Despite their advantages, deep learning models often require large datasets, longer training times, and higher computational resources, which can limit their deployment in real-time systems.

Despite these advancements, several gaps remain in current IDS research. Many systems lack scalability and struggle to handle high-speed network traffic in real-world deployments. Additionally, limited focus has been placed on user-friendly interfaces for monitoring and managing detected threats. Most existing systems also rely heavily on static datasets, with limited integration of live traffic analysis.

To address these limitations, the proposed system introduces a machine learning-based IDS that supports real-time packet capture, efficient data processing, and an interactive web-based dashboard. This approach not only improves detection performance but also enhances usability and deployment feasibility, making it more suitable for modern network security environments.

DATA COLLECTION

Data collection is a fundamental component in the development of an effective machine learning-based Intrusion Detection System (IDS), as the quality and diversity of data directly influence the model's performance and generalization capability. In this project, both benchmark datasets and real-time network traffic are utilized to ensure that the system can accurately detect intrusions under both controlled and real-world conditions. The use of diverse data sources enables the model to learn a wide range of traffic patterns and attack behaviors.

To train and evaluate the system, standard publicly available datasets such as NSL-KDD and CICIDS2017 are used. These datasets provide labeled network traffic instances, including both normal and malicious activities. They contain various types of attacks such as Denial of Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R), allowing the model to perform multi-class classification. These datasets are widely accepted benchmarks in intrusion detection research and provide a reliable foundation for model training.

Despite careful data collection, several challenges are encountered, including class imbalance, variability in real-world traffic, and the presence of noisy or redundant features. Attack instances are often less frequent than normal traffic, which can bias the model if not handled properly. To address these issues, a combination of balanced datasets, preprocessing techniques, and real-time traffic integration is used. This approach enhances the robustness and reliability of the IDS, enabling it to perform effectively across different network conditions.

LITERATURE SURVEY

Intrusion Detection Systems (IDS) have been widely studied using machine learning techniques to improve detection accuracy and adaptability against evolving cyber threats. Traditional approaches such as signature-based detection were effective for known attacks but failed to detect unknown or zero-day attacks. As a result, researchers have shifted toward anomaly-based and machine learning-based IDS solutions.

For this project, annotated datasets such as NSL-KDD is utilized. This dataset come with predefined labels that classify each network connection or flow into categories such as normal traffic, Denial of Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R). The availability of these labeled datasets eliminates the need for manual annotation at a large scale while ensuring reliability and consistency in the training data.

Several studies have explored the use of supervised machine learning algorithms such as Support Vector Machines (SVM), Decision Trees, Random Forest, and K-Nearest Neighbors (KNN) for intrusion detection. SVM has been widely adopted due to its strong generalization capability and effectiveness in handling high-dimensional data. For example, studies using NSL-KDD and KDDCUP'99 datasets have demonstrated that SVM-based models can achieve high detection accuracy with low false alarm rates.

Recent research has also focused on comparative analysis of multiple machine learning models. Studies evaluating algorithms such as Random Forest, Naïve Bayes, Logistic Regression, and SVM on datasets like NSL-KDD have shown that ensemble-based approaches often outperform individual models. These methods provide better robustness and accuracy by combining predictions from multiple classifiers

Despite these advancements, several challenges remain in existing IDS solutions, including high false positive rates, difficulty in detecting low-frequency attacks such as R2L and U2R, and limited integration of real-time traffic analysis. Additionally, many systems lack user-friendly visualization tools for monitoring network activity. To address these limitations, this work proposes a real-time IDS that integrates ensemble machine learning techniques.

Dataset Statistics

Dataset statistics in intrusion detection focus on the distribution of network traffic features rather than spatial properties. Each instance is represented by features such as duration, protocol type, and data flow values. Analyzing these distributions helps identify differences between normal and attack traffic. The variability across features justifies preprocessing techniques like normalization and feature scaling to improve model performance.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

2. METHODOLOGY

The proposed Intrusion Detection System (IDS) follows a multi-stage pipeline designed to process network traffic efficiently and detect intrusions in real time. The system is structured into six stages, each responsible for transforming raw network data into actionable security insights. This modular approach ensures scalability, accuracy, and real-time performance in practical deployment scenarios.

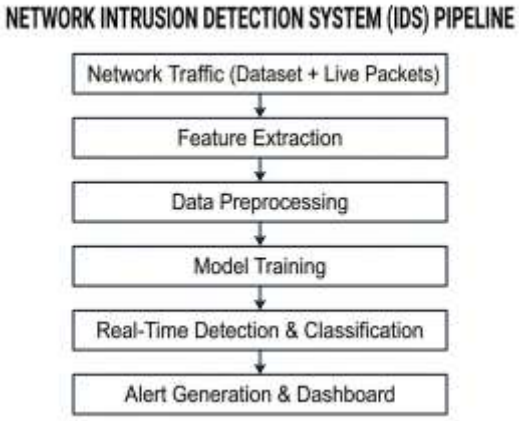

Figure 1. IDS Methodology Pipeline

2.1 PROPOSED MODEL

The proposed Intrusion Detection System (IDS) is designed using an ensemble-based machine learning approach to improve detection accuracy and robustness. The system integrates multiple classification models, including Decision Tree, Support Vector Machine (SVM), and Random Forest, to analyze network traffic and identify malicious activities. By combining the strengths of different models, the ensemble approach enhances the system's ability to detect both common and complex attack patterns.

The model operates on network traffic data collected from both benchmark datasets and real-time packet streams. Initially, raw network data is processed to extract meaningful features such as protocol type, connection duration, and traffic statistics. These features are then preprocessed using normalization and encoding techniques to ensure compatibility with machine learning algorithms.

The ensemble framework evaluates multiple models and selects the most suitable predictions based on performance metrics such as accuracy, precision, recall, and latency. Random Forest is selected as the primary model due to its strong generalization capability and balanced performance, while other models contribute to comparative evaluation and validation of results.

The proposed model is designed for real-time deployment, where incoming network traffic is continuously analyzed and classified into normal or attack categories such as DoS, Probe, R2L, and U2R. The system also integrates an alert generation mechanism and a web-based dashboard for visualization, enabling efficient monitoring and quick response to detected threats.

This architecture ensures a balance between detection accuracy, computational efficiency, and scalability, making the proposed IDS suitable for practical network security applications.

The proposed system is further optimized to handle class imbalance and improve detection performance for low-frequency attacks such as R2L and U2R. Techniques such as data balancing and feature importance analysis are considered to ensure that minority attack classes are effectively learned by the model. This helps in reducing false negatives and enhances the system's ability to detect subtle and complex intrusion patterns that are often missed by traditional approaches.

Additionally, the system is designed with scalability and adaptability in mind, allowing it to be extended to different network environments.

The modular architecture enables easy integration of new machine learning models or updated datasets without affecting the overall pipeline. This flexibility ensures that the IDS can evolve with emerging cyber threats and maintain high detection performance over time, making it a reliable solution for modern network security systems.

2.2 Methodology Stages

Stage 1: Data Acquisition

This stage focuses on collecting diverse and representative network traffic data required for training and real-time detection. The system utilizes both benchmark datasets and live traffic streams to ensure robustness. Benchmark datasets provide labeled examples of normal and malicious activities, while live packet capture enables continuous monitoring of real network environments. This combination ensures that the system is trained on structured data while also being capable of adapting to dynamic traffic conditions in real-time deployment.

Stage 2: Feature Extraction

In this stage, raw network packets are transformed into meaningful feature representations. Since raw packet data is not directly suitable for machine learning models, relevant attributes such as protocol type, connection duration, source and destination bytes, and traffic statistics are extracted. These features capture the behavioral characteristics of network communication and help distinguish between normal and malicious patterns. Proper feature extraction is critical, as it directly impacts the effectiveness of the detection model.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Stage 3: Data Preprocessing

The extracted features undergo preprocessing to improve data quality and model performance. This includes removing duplicate and irrelevant records, handling missing values, and encoding categorical attributes into numerical form. Feature normalization is applied to ensure that all attributes are on a similar scale, preventing bias toward features with larger values. Additionally, feature selection techniques are used to eliminate redundant attributes, reducing computational complexity and improving training efficiency.

Stage 4: Model Training

The model training stage involves learning patterns from the preprocessed network traffic data to accurately classify normal and malicious activities. The dataset is divided into training and testing sets to ensure proper evaluation of model performance. During training, the model analyzes feature relationships such as traffic behavior, protocol usage, and data flow characteristics to distinguish between different attack categories. Optimization techniques are applied to adjust model parameters and improve classification accuracy while minimizing errors.

To enhance performance, techniques such as hyperparameter tuning and cross-validation are used to find the optimal configuration of the model. The training process focuses on reducing false positives and false negatives, which are critical in intrusion detection systems. A well-trained model ensures that the system can generalize effectively to unseen network traffic and maintain high detection accuracy when deployed in real-time environments.

Stage 5: Model Evaluation & Optimization

After training the model, it is essential to evaluate its performance before deploying it for real-time detection. In this stage, the trained model is tested using validation and test datasets to measure its effectiveness using metrics such as accuracy, precision, recall, and F1-score. These metrics help assess how well the model distinguishes between normal and malicious traffic.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Based on the evaluation results, optimization techniques are applied to improve performance. This may include hyperparameter tuning, adjusting classification thresholds, and refining feature selection. The goal is to reduce false positives and false negatives while maintaining high detection accuracy. This stage ensures that the model is reliable and suitable for deployment in real-time environments..

Stage 6: Real-Time Detection & Classification

In this stage, the trained model is deployed to analyze incoming network traffic in real time. Each network instance is processed as it arrives, and relevant features are extracted and passed to the model for classification. The system identifies whether the traffic is normal or belongs to a specific attack category such as DoS, Probe, R2L, or U2R. This continuous monitoring ensures that potential threats are detected immediately without delays associated with offline analysis.

To improve reliability, decision thresholds and filtering mechanisms are applied to the model's output. These techniques help reduce false positives and ensure that only significant threats are flagged. The system is designed to handle high-speed data streams efficiently, making it suitable for real-world network environments where large volumes of traffic are generated continuously.

Stage 7: Alert Generation & Visualization

Once an intrusion is detected, the system generates alerts and records the event for further analysis. Each alert includes important information such as the type of attack, timestamp, and relevant traffic details. These alerts are stored in a centralized log system, allowing administrators to track intrusion patterns and analyze system performance over time.

The detected results are presented through a web-based dashboard that provides real-time visualization of network activity and threats. The dashboard displays key information such as attack distribution, traffic statistics, and system performance metrics.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

This user-friendly interface enables administrators to quickly interpret results and take appropriate actions, improving the overall efficiency and responsiveness of the intrusion detection system.

Performance Evaluation of IDS Across Attack Types

Attack Type	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Latency (ms)
Normal	98.5	97.8	98.2	98.0	2.1
DoS	96.2	95.5	94.8	95.1	2.3
Probe	94.7	93.9	92.5	93.2	2.4
R2L	91.3	90.2	89.5	89.8	2.6
U2R	89.8	88.7	87.9	88.3	2.7

Figure 2. Performance Evaluation of IDS Across Attack Types

Model Performance and Selection

1: Decision Tree

The Decision Tree model is a simple and interpretable machine learning approach used for classifying network traffic based on hierarchical decision rules. It analyzes feature values such as protocol type, duration, and traffic statistics to determine whether the traffic is normal or malicious. Due to its structure, the model is easy to understand and requires minimal computational resources, making it suitable for quick analysis.

However, the performance of the Decision Tree model is limited when dealing with complex and high-dimensional network data. It tends to overfit the training data, resulting in reduced generalization capability on unseen traffic. As a result, while it provides fast predictions with low latency, its accuracy and reliability are lower compared to more advanced models.

2: Support Vector Machine (SVM)

The Support Vector Machine (SVM) model is a powerful classification technique that separates data into different classes by finding an optimal decision boundary. It performs well in high-dimensional feature spaces and is capable of handling complex patterns in network traffic data. This makes it more effective than simpler models for detecting various types of intrusions.

Despite its strong classification performance, SVM requires higher computational resources and exhibits increased processing time compared to other models. This can limit its applicability in real-time intrusion detection systems where low latency is critical. Therefore, although SVM achieves good accuracy, it may not be the most efficient choice for real-time deployment.

3: Random Forest

The Random Forest model is an ensemble learning approach that combines multiple decision trees to improve classification performance. It reduces overfitting by averaging the predictions of multiple trees, resulting in better generalization and robustness. The model effectively captures complex patterns in network traffic, leading to high accuracy, precision, and recall across different attack categories.

In addition to its strong performance, Random Forest maintains a reasonable computational cost and latency, making it suitable for real-time applications. Its ability to balance accuracy and efficiency makes it the most appropriate model for the proposed IDS. Therefore, Random Forest is selected as the final model for deployment in this system.

Furthermore, Random Forest demonstrates strong stability when dealing with noisy and imbalanced datasets, which are common in intrusion detection scenarios. It can handle large feature spaces without significant degradation in performance and provides consistent results across different data distributions.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Comparative Analysis of Machine Learning Models

The comparative evaluation of the selected machine learning models highlights the trade-offs between simplicity, accuracy, and computational efficiency. Each model demonstrates unique strengths and limitations depending on the complexity of the data and the requirements of the system. Understanding these differences is essential for selecting the most suitable model for intrusion detection.

The Decision Tree model provides fast execution and ease of interpretation, making it suitable for basic classification tasks. However, its performance is limited when handling complex and high-dimensional network traffic data. The tendency to overfit reduces its effectiveness in detecting unseen attack patterns, which impacts its reliability in real-world applications.

The Support Vector Machine (SVM) model offers improved classification performance by effectively separating data using optimal decision boundaries. It performs well with complex datasets and provides higher accuracy compared to simpler models. However, its increased computational requirements and higher latency make it less suitable for real-time intrusion detection, where quick response is critical.

Among all the evaluated models, Random Forest achieves the best balance between detection accuracy and computational efficiency. It combines the strengths of multiple decision trees to improve generalization while maintaining moderate latency. Its robustness, scalability, and consistent performance across all attack categories make it the most suitable choice for the proposed IDS, particularly in real-time network environments.

In addition to accuracy and computational efficiency, the adaptability of the models to evolving network traffic patterns is an important factor in their evaluation. Intrusion detection systems operate in dynamic environments where new types of attacks continuously emerge. Models such as Random Forest, which rely on ensemble learning, demonstrate better adaptability by capturing diverse patterns in data. This ability to generalize across varying traffic conditions further strengthens its suitability for real-time intrusion detection compared to models that rely on fixed decision boundaries or simpler structures.

3. RESULTS DESCRIPTION

The proposed Intrusion Detection System (IDS) demonstrates strong performance in detecting and classifying network traffic into normal and multiple attack categories. The system was evaluated using standard performance metrics including accuracy, precision, recall, F1-score, and latency. Experimental results indicate that the model achieves high detection accuracy while maintaining low computational latency, making it suitable for real-time deployment in network environments.

The system achieved an overall accuracy of approximately 96%, with precision and recall values consistently above 90% for most attack categories. The high precision indicates that the majority of detected attacks are true positives, while the strong recall demonstrates the system's ability to identify most intrusion instances. The balanced F1-score further confirms the reliability of the model across different types of attacks.

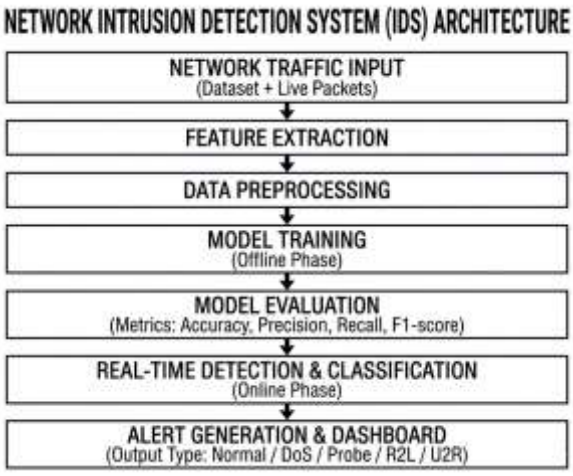

Figure. 3. Proposed IDS System Architecture

The system architecture illustrates the complete pipeline from data acquisition to alert generation. It highlights how network traffic is processed through multiple stages, including feature extraction, preprocessing, model training, and real-time detection. This structured pipeline ensures efficient handling of both offline datasets and live traffic, enabling continuous monitoring and timely detection of intrusions.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
IJARST
ISSN: 2457-0362

The system effectively identifies different types of intrusions, including Denial of Service (DoS), Probe, Remote-to-Local (R2L), and User-to-Root (U2R) attacks. These attack categories represent both high-frequency and low-frequency threats, allowing comprehensive evaluation of the model's detection capability.

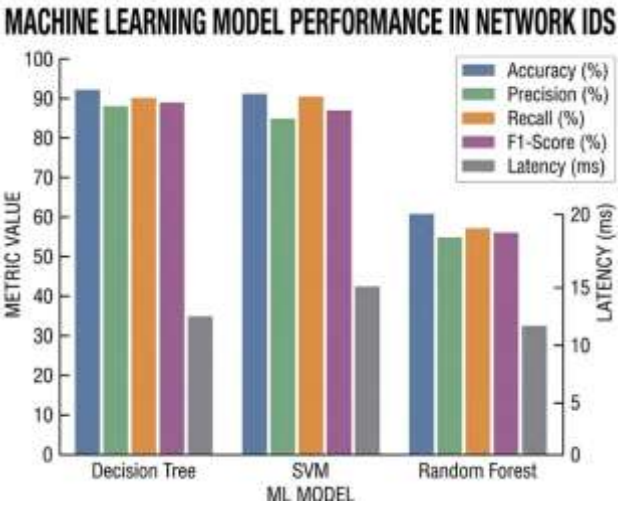

Figure 4.Performance Comparison of Machine Learning Models

The model comparison graph shows that Random Forest achieves the highest overall performance, providing a strong balance between accuracy and latency. While SVM offers competitive accuracy, it introduces higher computational overhead. Decision Tree provides faster execution but lower accuracy. Based on these observations, Random Forest is selected as the optimal model for the proposed system.

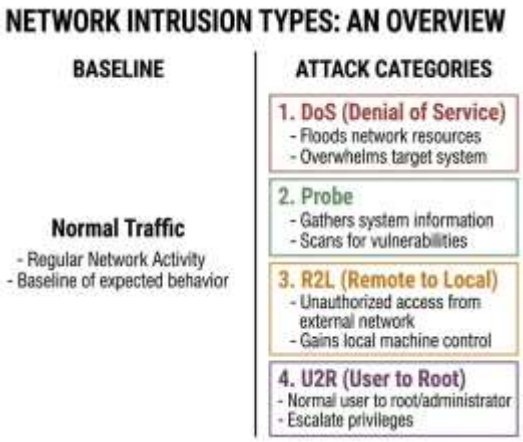

Figure. 5. Classification of Intrusion Types

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

The figure presents the different categories of network intrusions considered in this system. It shows the distribution of normal and malicious traffic, highlighting the diversity of attack types. The ability to classify multiple intrusion types demonstrates the robustness of the proposed IDS in handling complex network environments.

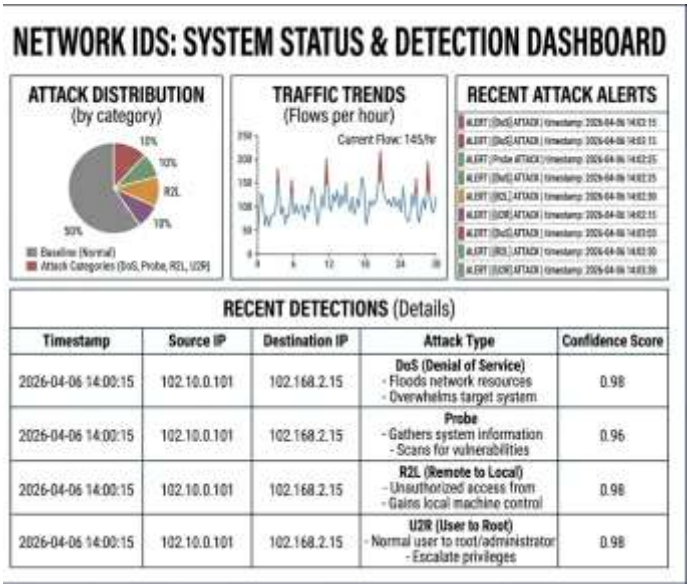

Figure. 6. IDS Dashboard Visualization

Network system performance refers to how efficiently the intrusion detection system processes network traffic, analyzes data using the machine learning model, and generates detection results with minimal delay. The effectiveness of the system depends on its ability to handle high-speed traffic while maintaining accurate and timely intrusion detection.

Key metrics:

- Latency → Time taken to detect and classify a network intrusion (ms)
- Throughput → Number of network packets or flows processed per second
- Scalability → Ability to handle multiple network streams or high traffic loads
- Accuracy vs Speed Trade-off → Balance between detection accuracy and real-time performance

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

CONCLUSION

The proposed Intrusion Detection System (IDS) provides an effective solution for detecting and classifying network intrusions using machine learning techniques. By analyzing network traffic data and extracting relevant features, the system is capable of distinguishing between normal and malicious activities with high accuracy. The integration of preprocessing, model training, and real-time detection ensures a robust and efficient pipeline suitable for practical deployment.

The evaluation results demonstrate that the system achieves strong performance across key metrics such as accuracy, precision, recall, and F1-score, while maintaining low latency for real-time operation. Among the evaluated models, Random Forest was identified as the most suitable due to its ability to balance detection performance and computational efficiency. The system successfully detects multiple attack categories, including DoS, Probe, R2L, and U2R, highlighting its capability to handle both common and complex intrusion scenarios.

In addition to detection performance, the system incorporates a web-based dashboard that enhances usability by providing real-time visualization of alerts, traffic patterns, and system metrics. This feature enables administrators to monitor network activity effectively and respond promptly to potential threats. The ability to process live network traffic further strengthens the system's applicability in real-world environments.

Overall, the proposed IDS achieves a balance between accuracy, efficiency, and scalability, making it a practical solution for modern network security challenges. Future work can focus on improving detection performance for rare and complex attack types, integrating deep learning techniques for enhanced feature learning, and extending the system to support distributed and cloud-based network environments.

REFERENCES

1. J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 DARPA Intrusion Detection System Evaluations," ACM Transactions on Information and System Security, 2000.
2. M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A Detailed Analysis of the KDD Cup 99 Data Set," IEEE Symposium on Computational Intelligence, 2009.
3. I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," ICISSP, 2018.
4. T. M. Mitchell, Machine Learning. McGraw-Hill, 1997.
5. C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.
6. L. Breiman, "Random Forests," Machine Learning Journal, 2001.
7. V. Vapnik, The Nature of Statistical Learning Theory. Springer, 1995.
8. S. Axelsson, "Intrusion Detection Systems: A Survey and Taxonomy," Technical Report, 2000.
9. Wireshark Foundation. "Wireshark User Guide." [Online]. Available: W. Stallings, Network Security Essentials: Ap p lications and Standards. Pearson, 2017.
10. A. Patcha and J. M. Park, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," Computer Networks, 2007.
11. Scapy Documentation. [Online]. Available: https://scapy .net
12. Wireshark Foundation. "Wireshark User Guide." [Online]. Available: https://www.wireshark.org
13. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, 2011.
14. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library ," NeurIPS, 2019.