IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Intrusion Detection System Using Machine Learning

[1]Aishwarya C Kodag,[2]Dayanand G Savakar,[3]Padma Yadahalli

[1]Department of Computer Science, Bagalkot University Jamakhandi

[2]Professor,Department of Computer Science,Bagalkot University Jamakhandi

[3]Department of Computer Science,Bagalkot University Jamakhandi.

Abstract: In today's digital era, networks are highly important but face growing threats such as malware, phishing, and denial-of-service attacks. Traditional intrusion detection systems, which rely on signatures, fail to detect zero-day and evolving attacks. To address this, the research proposes an intelligent IDS that uses machine learning models like SVM, Random Forest, and Logistic Regression to classify traffic as normal or malicious. The system includes real-time monitoring, automated classification, database logging, and an interactive web dashboard. Its performance is validated using metrics such as accuracy, precision, recall, and F1-score, showing strong robustness. Overall, the proposed IDS achieves higher accuracy and scalability, making it suitable for enterprises, institutions, and critical infrastructures.

Keywords: Intrusion Detection System, Machine Learning, Cybersecurity, Random Forest, Support Vector Machine, Logistic Regression

I. INTRODUCTION

Networked systems are essential across government,healthcare, business, and infrastructure but face rising threats like ransomware, malware, phishing, and DoS attacks. Traditional signature-based IDS can detect only known attacks and fail against zero-day or polymorphic malware, making them reactive and limited. To overcome this, the proposed research introduces an intelligent IDS using machine learning models SVM, Random Forest, and Logistic Regression to classify normal and malicious traffic. The system provides real-time monitoring, secure login, and an interactive dashboard for managing detection tasks. By learning from data instead of static rules, it can detect both known and unknown threats more effectively. This framework builds on past studies and offers a scalable, adaptive, and proactive defense for modern cybersecurity needs.

Flask web application that leverages machine learning models to classify network traffic as normal or malicious, addressing the limitations of traditional signature-based systems against modern threats. The system's architecture (Figure 1) facilitates user interaction through a secure login (SQLite for authentication) and an interactive Dashboard that manages the core machine learning workflow, which involves Train, Test, and Predict functions utilizing the scikit-learn library for real-time traffic analysis from CSV data. The system supports multi-class classification for attacks like DoS, R2L, and Malware. Performance evaluation showed that the

Random Forest model was the most effective, achieving the highest overall accuracy at 99.20% (Figure 2), thanks to its ability to handle complex network data and minimize misclassifications (Figure 3). In comparison, both SVM and Logistic Regression achieved an accuracy of approximately 91% (Figure 2), with their respective confusion matrices (Figures 4 and 5) indicating they struggled more with complex or imbalanced patterns compared to Random Forest. Overall, the IDS provides real-time monitoring, secure data logging, and an adaptive, proactive defense highly suitable for safeguarding critical network environments

II. REVIEW OF LITERATURE

[1]**Scarfone & Mell (2007)** authored the Guide to Intrusion Detection and Prevention Systems (IDPS), which provides comprehensive standards and best practices for IDPS deployment. It remains a critical reference for understanding system architecture and operational requirements.

[2] **Tavallaee et al. (2009)** conducted a Detailed Analysis of the KDD CUP 99 Dataset, one of the most widely used benchmarks in network intrusion detection research. Their work revealed significant limitations and biases in the dataset, influencing the development of improved datasets in later studies.

[3]**Moustafa & Slay (2015)** introduced the UNSW-NB15 dataset. This comprehensive dataset was designed to represent a realistic and modern mix of benign and malicious traffic, advancing the state of research in Network Intrusion Detection Systems.

[4] **Sharafaldin, Lashkari, & Ghorbani (2018)** further refined dataset generation with their work Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. Their effort produced a more current and sophisticated dataset, better suited to evaluating IDS performance against contemporary threats.

[5]**Biswas (2018)** explored methods for improving Intrusion Detection Systems (IDS), with a particular focus on Feature Selection Techniques. This study emphasizes the role of data preprocessing and feature engineering in enhancing the accuracy and efficiency of detection models.

III. PROPOSED SYSTEM

The proposed system introduces an intelligent Intrusion Detection System (IDS) that leverages machine learning algorithms instead of relying solely on static rule-based approaches. By training on benchmark datasets such as NSL-KDD, UNSW-NB15, and CICIDS, the IDS is capable of identifying both known and previously unseen attacks, thereby addressing the limitations of conventional signature-based systems.The system supports multi-class traffic classification, categorizing network activity into Normal, DoS, Probe, R2L, U2R, Malware, and Phishing. This fine-grained classification enables administrators to understand not only the presence of an intrusion but also its type, thus enhancing situational awareness and response strategies.

To ensure ease of use, the IDS is developed as a Flask-based web application with a secure login mechanism and interactive modules for Training, Testing, Prediction, and Traffic History. Users can seamlessly manage datasets, initiate model training, visualize results, and track network activity through an intuitive dashboard.Another key feature is the flexibility of model selection and evaluation. Users can train and compare multiple algorithms such as Logistic Regression, Random Forest, and Support Vector Machines (SVM), and assess them with accuracy, precision, recall, F1-score, and confusion matrix metrics. This comparative approach empowers administrators to select the most effective model for their specific network environment.

IV.METHODOLOGY

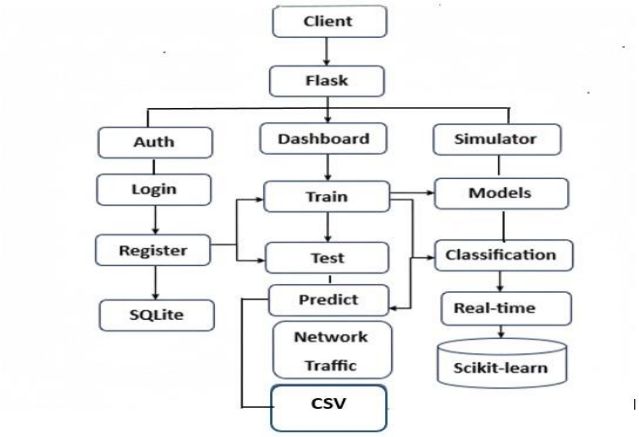

Figure 1: Block Diagram

The architecture of the intelligent Intrusion Detection System (IDS), illustrated in Figure 1, is built around a Flask-based web application that orchestrates all core processes, beginning with client interaction. The system follows a modular design, integrating authentication, user management, model operations, and network monitoring into a unified flow. Authentication manages user login and registration, securely handling credentials and storing activity history in an SQLite database. Once authenticated, users access the Dashboard, which serves as the primary interface for controlling the machine learning pipeline. This pipeline supports uploading network traffic CSV files, performing training and testing routines to optimize the models, and generating prediction outputs that classify traffic as either malicious or normal, with all results logged for reference. Alongside the Dashboard, the Simulator module enables direct interaction with the models to support classification tasks and real-time intrusion detection. At the core of these operations, machine learning algorithmsRandom Forest, Support Vector Machine, and Logistic Regression—are implemented using the scikit-learn library. Together, this structure ensures secure user management, real-time network analysis, and an interactive environment for efficiently managing intrusion detection tasks.

V. RESULTS AND DISCUSSION

The results and discussion, illustrated in Figure 2, focus on the performance evaluation of the implemented classifiers, namely Support Vector Machine (SVM), Logistic Regression, and Random Forest, which were applied to analyze and compare detection accuracy and misclassification trends.

- **Random Forest:**The Random Forest (RF) model is the undisputed best performer, demonstrating superior reliability across the entire dataset. Its success is rooted in its high overall Accuracy (0.99) and excellent Weighted Average F1-Score (0.99), which reflects strong performance across all samples. Crucially, RF exhibited exceptional robustness on class imbalance, evidenced by the highest Macro Average F1-Score (0.97). This capability translated into nearly perfect detection of the critical minority classes, successfully classifying instances in Label 2 (0.94 F1-Score) and Label 4 (0.92 F1-Score), where other models failed entirely.
- **SVM:**.The performance of the Logistic Regression (LR) model is characterized by a deceptive overall Accuracy of 0.91, which is artificially inflated by its correct classification of the large majority class (Label 0). This high score masks its true weakness: a catastrophic failure on minority classes. LR recorded a 0.00 F1-Score for the smallest class (Label 2), indicating it could not detect any of those instances. Consequently, its low Macro Avg F1-Score of 0.74 demonstrates poor generalization and high unreliability when dealing with the entire range of attack types.
- **Logistic Regression:**The Support Vector Machine (SVM) model presented an overall Accuracy of 0.91, matching Logistic Regression, but this score is highly misleading due to severe issues with class imbalance. SVM exhibited complete failure to detect Label 2, resulting in a 0.00 F1-Score. This catastrophic inability to classify the most rare instances led to the lowest Macro Avg F1-Score of 0.66

among all models, confirming its weakest generalization performance and making it the least reliable choice for this multi-class classification task.

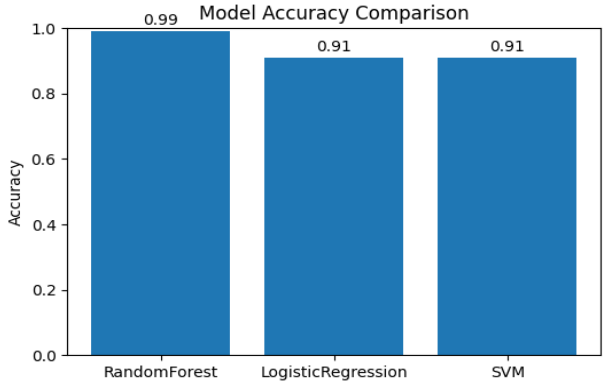

Figure 2: Model Accuracy Comparison

VI. Figures and Tables

Table:1Classification Report

Label	Random Forest				Logistic regression				SVM			
	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support
0	0.99	1.00	0.99	3645	0.93	0.94	0.93	3645	0.92	0.95	0.94	3645
1	0.98	0.99	0.98	393	0.83	0.65	0.73	393	0.83	0.65	0.73	393
2	1.00	0.89	0.94	19	0.00	0.00	0.00	19	0.00	0.00	0.00	19
3	0.99	1.00	0.99	609	0.87	0.89	0.88	609	0.94	0.85	0.89	609
4	1.00	0.86	0.92	7	0.86	0.86	0.86	7	1.00	0.14	0.25	7
5	0.99	0.99	0.99	251	0.80	0.99	0.89	251	0.80	0.97	0.88	251
6	1.00	0.97	0.98	436	0.94	0.89	0.91	436	0.95	0.88	0.91	436
Accuracy			0.99	5360			0.91	5360			0.91	5360
Macro Avg	0.99	0.96	0.97	5360	0.75	0.75	0.74	5360	0.78	0.63	0.66	5360
Weighted Avg	0.99	0.99	0.99	5360	0.91	0.91	0.91	5360	0.91	0.91	0.91	5360

In the above table is a Classification Report comparing the performance of Random Forest, Logistic Regression, and SVM on a multi-class dataset with 7 labels and 5360 samples, highlighting class imbalance. Random Forest is the superior model, achieving 0.99 Accuracy and a high Macro Avg F1-Score of 0.97, demonstrating robust performance even on minority classes (e.g., Label 2). In contrast, Logistic Regression and SVM both fail completely to identify the smallest class (Label 2), resulting in a much lower Macro Avg F1-Score (0.74 and 0.66) despite having decent overall accuracy (0.91) due to the dominant Label 0.

The classification performance of the IDS was analyzed using confusion matrices for Random Forest, Logistic Regression, and
Support Vector Machine (SVM)

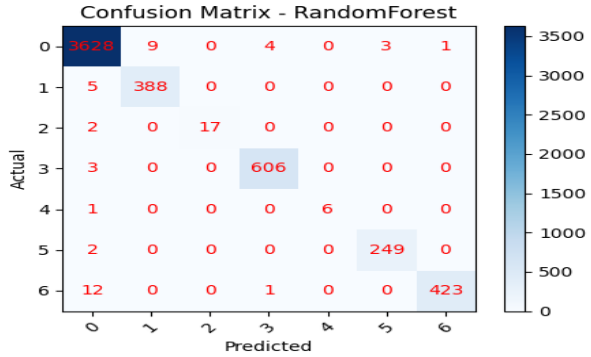

Figure 3: Confusion Matrix RandomForest

As shown in Figure3 the RandomForest results show that the diagonal values, such as 3628, 388, and 606, represent the correctly classified instances for each attack class. Since the majority of values lie on the diagonal and the off-diagonal values are very close to zero, it indicates that the model makes very few misclassifications. This demonstrates that the RandomForest classifier achieves high accuracy and strong performance in distinguishing between multiple attack classes, making it reliable for intrusion detection and network security analysis.

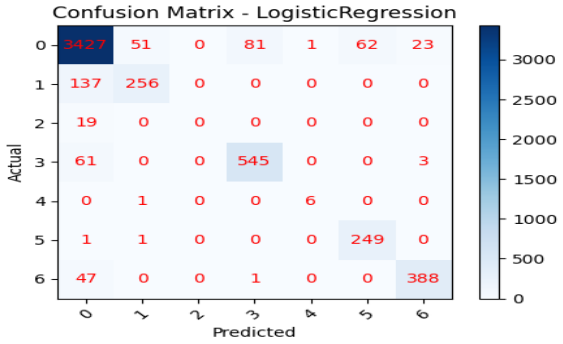

Figure 4: Confusion Matrix Logistic Regression

In the above Figure 4 Logistic Regression shows correct predictions (e.g., 3427, 545, 249, 388), but higher misclassifications compared to RandomForest. For instance, many class 1 samples are wrongly classified as class 0. This indicates that Logistic Regression struggles with capturing complex, non-linear patterns in network traffic, leading to reduced accuracy.

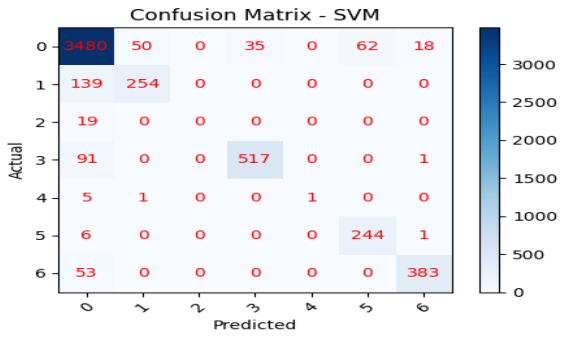

Figure 5: Confusion Matrix SVM

As shown in Figure5 SVM performs better than Logistic Regression but is less accurate than RandomForest. While classes like 3 and 6 are well classified, class 1 shows several misclassifications. This suggests that SVM can capture patterns effectively but may struggle with larger or imbalanced datasets, limiting its overall performance.

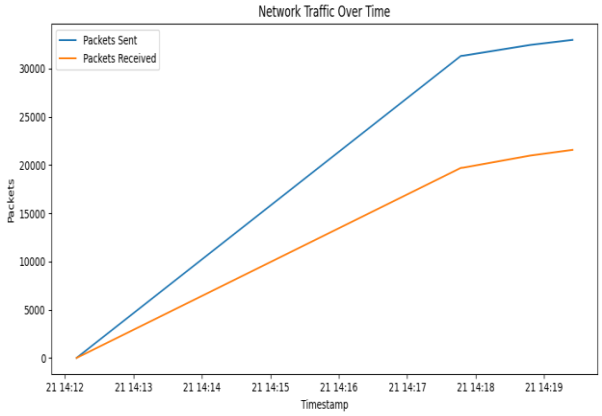

Figure 6: Network Traffic Over Time

Figure 6 illustrates the network traffic variation over timegraph shows that packets sent are consistently higher than packets received, indicating normal client-server behavior but also a possible sign of packet loss or delay. Both values increase steadily over time, though the growth rate slows later, suggesting congestion or network limitations. It helps analyze traffic flow, reliability, and overall network performance.

CONCLUSION

The Intrusion Detection System (IDS) with Machine Learning is a fast and intelligent solution for monitoring network activity and detecting potential security threats. It provides real-time traffic data analysis, effective intrusion detection, and a full history of user actions and network occurrences. The solution lowers the need for manual monitoring, improves network security, and increases administrator responsiveness by sending timely notifications. Its modular design, secure data processing with SQLite, and integration with machine learning models all contribute to maintainability, scalability, and future improvements. Overall, the solution provides a dependable and powerful way to safeguarding network settings against unwanted access and other cyber threats.

REFERENCES

[1] Scarfone, K., & Mell, P. (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS)*. NIST Special Publication 800- 94.

[2] Tavallaee, 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*.

[3] Moustafa, N., & Slay, J. (2015). "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," *Military Communications and Information Systems Conference (MilCIS)*.

[4] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *International Conference on Information Systems Security and Privacy (ICISSP)*.

[5] Biswas, S. K. (2018). *Improving Intrusion Detection Systems through Feature Selection Techniques*. [Journal/Conference].