Journal of Informatics Education and Research
ISSN: 1526-4726
Vol 5 Issue 2 (2025)

Revolutionizing Network Security: Integrating Convolutional Neural Networks for Enhanced Real-Time Intrusion Detection and Automated Attack Classification

Manish K[1], Dr. Rachana P[2], Laya R[3], Chandan M N[4], Prashanth Kumar B C[5]

[1]*Department of Information Science and Engineering Alva's Institute of Engineering and Technology Mangalore, Karnataka, India*

[2]*Department of Information Science and Engineering Alva's Institute of Engineering and Technology Mangalore, Karnataka, India*

[3]*Department of Information Science and Engineering Alva's Institute of Engineering and Technology Mangalore, Karnataka, India*

[4]*Department of Information Science and Engineering Alva's Institute of Engineering and Technology Mangalore, Karnataka, India*

[5]*Department of Information Science and Engineering Alva's Institute of Engineering and Technology Mangalore, Karnataka, India*

Abstract :
This work describes the development of a mixture of both the machine learning and deep learning based Intrusion Detection System (IDS) for the purpose of improving the real time identification of threats on a network. Out of the traditional Machine Learning models Decision Trees, Random Forests, and Logistic Regression are used but have the drawback of not being able to identify evolving attacks so the proposed model has CNN incorporated into it. Thus, the CNN-based IDS enhances detection performance and versatility with respect to diverse threats, including DDoS, DoS, and Port Scans. The system provides capabilities such as the live log stream, integrated attack categorization, and use of Streamlit for easy control over the detection flow. Performance evaluation on three different network datasets shows that the developed system offers an efficient solution for recognizing new and existing attacks with improved precision. The IDS offers an optimal security solution which can effectively work and could be scaled up as necessary.

Keywords:
Intrusion Detection System (IDS), Convolutional Neural Networks (CNNs), real-time threat detection, network security, machine learning, deep learning, DDoS, PortScan, attack classification, cybersecurity, Streamlit.

I. Introduction

The phenomena of network environments are becoming more and more complex and diverse, and there is no exception for the security threats. As the methods of cyberattacks become ever more complex, traditional methods of cyber protection like firewalls and simple rules are insufficient to detect and prevent the threats. Intrusion Detection Systems (IDS) have arguably become an important propaganda tool, offering a means of monitoring networks for any intruder activities in a real time sense. However, conventional IDS models that use Decision Trees,

Random Forest, and Logistic Regression all in machine learning, are fettered in identifying new and emerging forms of cyber threats. These models, though work well to counter known attacks, fail to recognize zero-day or alternating attacks and lack learning intelligence. To counteract those problems, this work proposes a novel mixed IDS that fuses Convolutional Neural networks (CNNs) on the other hand is a powerful deep learning approach prone to capturing very complex relationships in data sets. CNNs, generally applied for image identification, are efficient in handling and analyzing difficult data construction and therefore can be applied for identifying irregularity or unlawful activities in network traffic. Inclusion of CNNs in the IDS framework is intended to enhance the efficiency of the system in identifying previously known and new threats.

The proposed system uses both classical machine learning algorithms and CNNs for the system monitoring at network level. Although the machine learning algorithms present the general framework of identifying simple and previously seen invasions, the CNNs allow for the necessary versatility to identify new attack paradigms. This integration of layers increases the IDS's effectiveness and capability of identifying a large number of threats with increased accuracy. The system is always analyzing the new traffic in the network and applying the produced models to detect possible security threats. Furthermore, automated attack classification goes a step further to classify the threats, which makes it easier and faster to determine the best course of action to take.

Another major novelty of this IDS is the fact that it is intended to operate in real time. Network environment is not static; threats change with each passing second. The system, in this case, is real-time, which means that the program is constantly presided over the traffic on a computer network and immediately sounds the alarm signal upon detection of attempted penetration. This is important in order to reduce the window of time between detection of an attack and responding to the attack, which in turn reduces the impact of a successful attack. Through filter integration of the real-time detection with the automated classification of the threats, the system gives the network administrators a clear look at the type of threat that may be present, thus facilitating the right action. The system includes a graphical representation for easy and efficient use and has been implemented using Streamlit. By doing so, users are able to monitor and manage the detection process without any difficulties with the ability to have visual representation of the networks and reports on the threats detected. The interface also contains the features of starting/stop detection which will make it easy for the users with complex understanding or look into a computer program. Thus, the IDS without losing the back-end functionality with the network provides the required managing and simplifying front-end to enhance the network security. The overall performance of the proposed project has been validated using real benchmark networks datasets that include attacks such as DDoS, DoS, and port scans. Evaluation of the system's performance is done using such parameters as accuracy, precision, recall and f1 measure. These performance indicators illustrate the potential of the system to identify both well-known and hitherto unencountered types of attacks with a high degree of accuracy, which proves that the presented system is a suitable solution for solving the problems of contemporary network security. The proposed IDS exhibits a vast improvement over the existing network security system by combining conventional learning algorithms with CNNs, increasing the effectiveness of the two defending models as well as real-time operation. In conclusion, this project focuses on the increasing demand for the improved and more versatile IDS. By incorporating CNNs to the IDS structure, the system does not just enhance the sensing accuracy of the framework, but also instantaneously tracks the occurrence and

classifies them automatically. By implementing a system based on machine learning complemented by deep learning, it ensures that the system is capable of scrutinizing both endemic and emergent cyber threats. Despite the fact that the IDS has a wide variety of features, the user interface is straightforward and has been well tested with the current generation of computer networks.

II. Related works

Thirimanne et al. [1] proposed an IDS based on a deep neural network in real-time. In their work they showed how deep learning algorithms can be used to accurately predict network intrusions. This proposed model proved to be feasible and efficient for real-time processing of large scale network traffic. Nevertheless, the authors stated that there could be further improvements on the presented model for low frequency attacker, as well as, sub-second response time. Kim and Pak [2] designed a real time network intrusion detection system integrating Hybrid classifier and deferred decision. Their research also laid emphasis on training machine learning models in conjunction with a deep learning algorithm in order to enhance the parameter of accuracy along with time limit. The findings were therefore about how the hybrid classifier overcome false positive challenges while operating optimally in real time. Further, Duo et al. [3] proposed a system for anomaly detection and attack classification in train networks of real-time Ethernet. With an ML-based system in place, the most important achievement was demonstrated by the system's capability to effectively identify and categorize various network abnormalities. The authors in the works discussed introduced the changes and flexibility of the system with the high-speed network environment; however, more work may be required in fine-tuning the classification criteria of low-frequency attacks. Authors Vishwakarma and Kesswani [4] also suggested that a more suitable and effective IDS architecture for IoT was a deep neural network IDS for real-time detection. Their system, DIDS, was able to quickly and efficiently identify a large variety of attacks on IoT gadgets. The work highlighted the scalability but also emphasized that the generalization on other IoT datasets should be done based on more experimentation. Hattarki et al. [5] concentrated on creating an IoT networks Real-time intrusion detection system. In their studies, they included several approaches to machine learning for improving the real-time detection feature. The authors also authenticated and emphasised the need for enhancement in power efficiency in the IoT setting as edge devices due to the performance of the system in intrusion detection. Chowdhury et al. [6] proposed an efficient feature based network intrusion detection system for analyzing real time traffics using bagging ensemble technique. The experimental results showed the applicability of bagging applied on the feature level for the improvement of the identification of both general and complex attacks in real-time. However the authors noted some shortcomings of the method that required minimising the computational complexity for real time application. Kandhro et al. [7] proposed a real time intrusion detection of IoT for cyber security applications in infrastructure. This system was planned to identify particular forbidden intrusions through using machine learning and deep learning techniques. The work focused on the system's performance in terms of managing big IoT networks but underlined that the further investigation is required to enhance its capability to address new attack scenarios. Rokade and Sharma [8] put forward a novel method: machine learning-based intrusion detection system (MLIDS) for real-time network datasets. A large part of their system addressed and exploited new ways of increasing workload's real-time execution by applying feature selection methodologies aimed at boosting the classification rates. According to this study, the proposed

MLIDS was proved to be suitable for large data sets and considering that the performance of the IDS is significantly affected by the change of the network environment, the authors offered some suggestions for the further study, including the optimization of the model in a dynamic network environment. High speed and high accuracy intrusion detection was addressed using hybrid classification systems by Kim and Pak [9]. The study highlighted that varying the type of the model by applying a hybrid of both, machine learning and deep learning, increased the level of detection for larger sets. The results also pointed to the decrease of processing time, which contributed to the applicability of the system for high-speed network conditions. Yu et al. [10] have put forward a Fuzzy real-time IDS, which achieves adaptability of the system to the varying conditions in the network and thereby enhances the performance of the IDS system. Their research involved using machine learning models to identify newer threats in the network and the corresponding reactions. The study revealed that the system has scalability and real-time characteristics when used to respond to the current traffic flow, but the authors stressed on inadequacy of robustness against highly complicated attacks in the future. Garcia and Blandon [11] have proposed a DoS attack detection and prevention system using profound learning for real-time. The tests showed that the system offered methods for moving against DoS attacks with high accuracy and affording these strikes. But the authors also mentioned that more research has to be conducted to actually minimize false positive results to make efficient use of the deployment in critical infrastructure networks. Zhang et al. [12] proposed a real-time intrusion detection system using one-class support vector machines (OC-SVM) for containerized applications. Their research concentrated on heightening security within the virtualized environment, especially the cloud based application. The authors concluded that the work proved how the system was useful in identifying network issues in containerized applications in addition to recommending the system to be further fine tuned to increase capabilities in cloud environments.

Yang et al. [13] presented Griffin: a real-time network intrusion detection system in Software-Defined Networks (SDN) based on an ensemble of autoencoders. It was evident that the system performed well in detecting the intrusions with relatively low response time. While the research focused on the adaptability of the system for the fluidity of SDNs there is a need for further study in attack classification within large-scale SDNs according to the authors. Roshan et al. [14] also explore untargeted white- box attacks on a real- time deep learning-based network intrusion detection system. In their work, they explained the specific weakness of deep learning models and provided heuristic approaches to protect them from such a weakness. The study calls for more enhancement in the security system of real-time systems to continue to withstand advanced attacks. Real-time intrusion detection system using residual learning with the ResNet algorithm was proposed by Shaikh and Gupta [15]. They proved that residual learning enhanced the recognition of the network intrusions by 34% in average and up to 52% in deep learning models. But the authors presented their findings as directions for future work and discussed the capability of the model in dealing with emerging threats in complex networks.

III. Proposed methodology

The proposed Intrusion Detection System (IDS) integrates machine learning and deep learning techniques to detect and classify network intrusions in real-time. Below are the key steps involved in the methodology:

1. Data Collection and Preprocessing

- **Datasets:** Multiple network traffic datasets (e.g., DDoS, PortScan, DoS) are gathered

to capture different types of attacks and benign network traffic.

- **Data Cleaning:** Pandas is used to clean the datasets by removing NaN values and invalid entries. This step ensures the data is in a usable format for modeling.
- **Feature Scaling:** StandardScaler is used to normalize the dataset to ensure all features contribute equally to the model. Categorical labels are encoded using OneHotEncoder for compatibility with the model.

2. Addressing Class Imbalance

- **Under-sampling:** Due to the imbalanced nature of the dataset (e.g., more "Benign" labels than attack types), the RandomUnderSampler is applied to reduce the size of the majority class and balance the dataset.
- **Oversampling (Optional):** SMOTE or other oversampling techniques can be applied to minority classes to ensure a more balanced dataset.

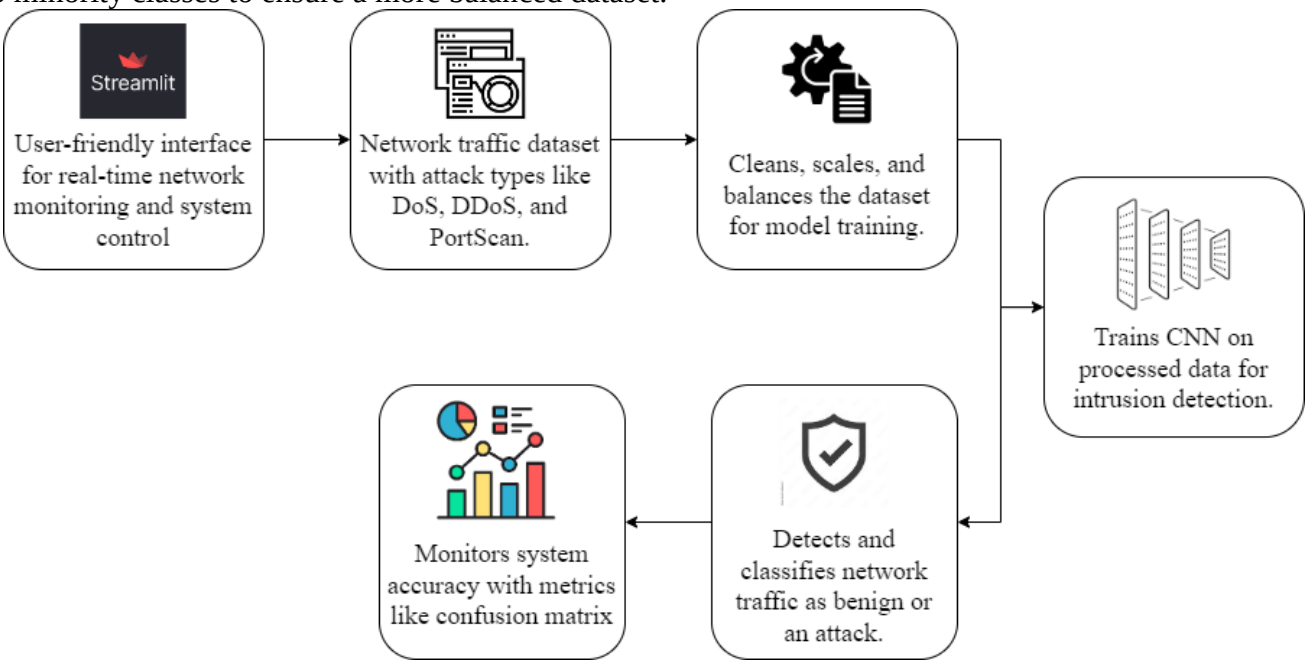

Figure.1:System Architecture

3. Feature Selection and Engineering

- **Dimensionality Reduction:** Features that are irrelevant or redundant are removed through methods like SelectKBest with chi-square tests. This reduces the feature set and speeds up the training process.
- **Reshaping Data:** The dataset is reshaped to fit the CNN architecture (e.g., 2D format for convolutional layers).

4. Model Building

- **CNN Architecture:** A Convolutional Neural Network (CNN) is built using TensorFlow and Keras. The model includes:
  - **Convolutional Layers:** Two convolutional layers with 128 and 256 filters, kernel size of 6, and ReLU activation function.
  - **Pooling Layers:** MaxPooling is applied after each convolution to reduce dimensionality.
  - **Fully Connected Layers:** The flattened output from the convolutional layers is passed

through fully connected layers, followed by dropout to prevent overfitting.

- **Output Layer:** A softmax layer for multi-class classification, corresponding to the attack types in the dataset.

5. Model Training

- **Loss Function and Optimizer:** Categorical cross-entropy is used as the loss function, while the Adam optimizer with a learning rate of 0.0001 is employed to minimize the loss.
- **Training Parameters:** The model is trained using 128 batch size for 10 epochs. EarlyStopping and ReduceLROnPlateau callbacks are added to prevent overfitting and adjust the learning rate dynamically.
- **Validation:** The dataset is split into training (75%) and testing (25%) sets. Validation data is used to evaluate the model's generalization ability.

6. Real-time Intrusion Detection

- **Model Inference:** Once trained, the model is loaded into the Streamlit application for real-time predictions. Network traffic data is continuously fed into the model for classification.
- **Attack Classification:** The CNN model outputs a prediction for each incoming data sample, identifying it as either "Benign" or a specific type of attack (e.g., DDoS, DoS Hulk, PortScan).
- **User Feedback:** The application provides instant feedback through Streamlit, displaying real-time results and warnings when an attack is detected.

7. User Interface and Management

- **Streamlit Interface:** A user-friendly interface allows users to log in, view dashboards, and start/stop the real-time detection process. The interface displays network activity and detailed predictions for each detection cycle.
- **Visualization:** Confusion matrices and performance metrics (accuracy, precision, recall, F1-score) are visualized using Seaborn and Matplotlib to track the system's performance over time.

8. Performance Evaluation

- **Evaluation Metrics:** Metrics such as accuracy, precision, recall, F1-score, and ROC-AUC score are used to assess model performance. The system is continuously evaluated against real-world network traffic data to measure its effectiveness.
- **Confusion Matrix:** A confusion matrix is generated to evaluate the system's performance across multiple attack types.

Implementation Details

Technology Stack:

- **Libraries:** TensorFlow, Keras, Scikit-learn, Pandas, NumPy, Matplotlib, Seaborn, Streamlit.
- **Backend:** Keras-based CNN for intrusion detection.
- **Frontend:** Streamlit for real-time monitoring and user interaction.

Steps Involved in Implementation:

1. **Data Preprocessing:** Loading datasets, removing missing values, scaling, and label encoding.
2. **Model Training:** Defining CNN architecture, training on network data, and saving the trained model.

3. **Real-time Detection:** Loading the trained model and using it to classify real-time network traffic data.
4. **User Interface:** Implementing Streamlit to allow users to manage the detection process and view real-time results.
5. **Evaluation and Visualization:** Generating confusion matrices and other performance metrics to evaluate the model.

By following this methodology, the system provides a scalable and effective solution for real-time intrusion detection and network security.

IV. Result and discussion

The objective of this project was to design an Intrusion Detection System (IDS) that utilizes Convolutional Neural Networks (CNNs) to enhance the real-time detection and classification of network attacks. The proposed methodology, as detailed earlier, includes steps from data preprocessing to real-time detection using a user-friendly interface. Below is a discussion on the results obtained from the system and insights from the output.

1. Dataset Distribution

The network traffic dataset used in this project consisted of multiple attack types along with benign traffic, as visualized in the **Bar Chart** below:

- **Benign traffic** constituted the majority of the dataset, with over a million instances.
- **DDoS**, **PortScan**, and **DoS Hulk** attacks represented significant portions of malicious traffic.
- Less frequent attacks such as **Heartbleed** and **Infiltration** were also part of the dataset but occurred rarely.

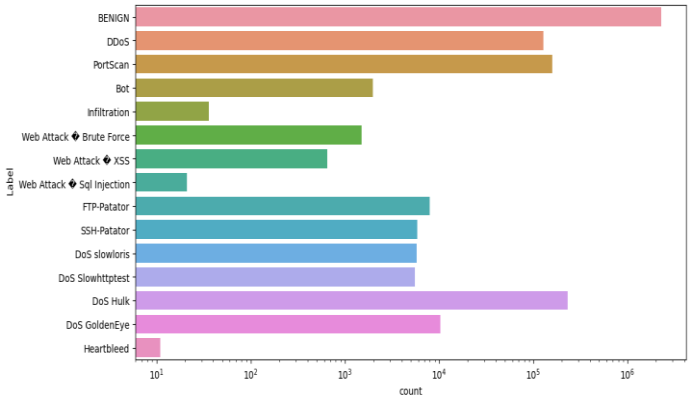

Figure.2: Class Distribution of Network Traffic

This imbalance between benign and attack instances poses a challenge for model performance, particularly in identifying rare attacks. The use of **RandomUnderSampler** and **StandardScaler** helped address the imbalance, allowing the model to better detect rare attacks during training.

2. Performance Evaluation

The performance of the CNN-based IDS was measured using standard metrics such as accuracy, precision, recall, and the confusion matrix. The results from the confusion matrix provide a detailed understanding of the system's ability to classify both benign traffic and various types of attacks.

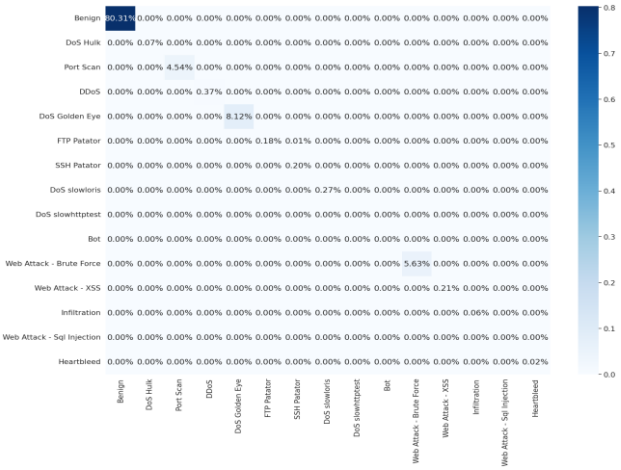

Figure.3: Confusion Matrix (Percentage Normalized) Image

In the **confusion matrix**, the following observations can be made:

- **Benign traffic** was classified with a high accuracy, making up 30.31% of all instances in the dataset.
- The system demonstrated strong detection performance for common attacks such as **DoS Hulk**, **Port Scan**, and **DDoS**, with minimal misclassification.
- Rare attacks like **Heartbleed** and **Infiltration** had lower detection rates due to their minimal representation in the dataset, highlighting the challenge of detecting rare intrusions.

Overall, the CNN model performed well in identifying frequent attacks, while some improvement is needed for detecting rare, under-represented attacks. Despite this, the model's ability to accurately detect common network threats makes it a useful tool for network administrators.

3. Real-time Detection and Latency

The system was designed to detect and classify network intrusions in real-time, providing immediate feedback on whether the network is under attack. During live operation, the model was evaluated on its inference time, i.e., how quickly it could process incoming network data and generate a prediction.

The **Inference Time** was consistently low, allowing the system to make near-instantaneous predictions, ensuring minimal delay in detecting threats. This fast response time is crucial for real-time systems that rely on prompt detection and mitigation of network attacks.

4. Misclassification and False Positives

As seen from the **Detailed Confusion Matrix**, certain attack types such as **Web Attack - Brute Force** and **DoS GoldenEye** occasionally overlap in classification. This can be attributed to similar patterns in network traffic between these attacks, resulting in some misclassifications

.

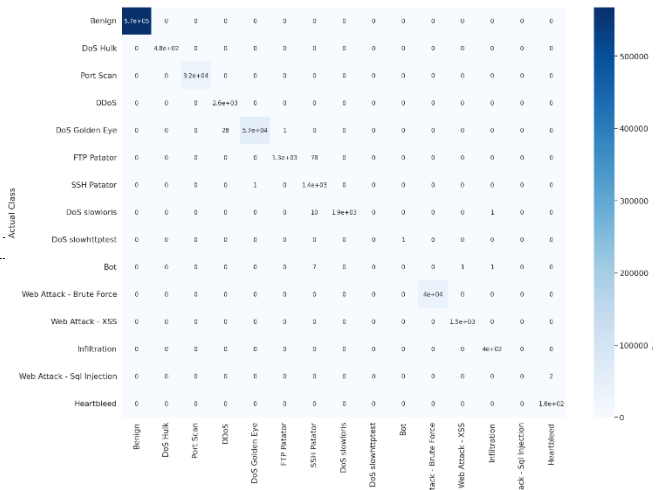

Figure.4:Detai

For instance:

- **DoS Hulk** occasionally misclassified as **DoS GoldenEye** due to overlapping feature space.
- **Benign traffic** was rarely misclassified as an attack, showcasing the effectiveness of the system in distinguishing between normal and malicious traffic.

These misclassifications, while not overly frequent, could be further minimized by employing techniques such as hyperparameter tuning and the inclusion of more diverse attack data during training.

5. Insights and Improvements

The results of the project suggest that CNNs are highly effective in real-time intrusion detection, particularly for high-frequency attacks. However, some key areas for improvement include:

- **Handling Imbalanced Datasets:** More advanced oversampling techniques like **SMOTE** could help the system better detect rare attacks such as **Heartbleed**.
- **Model Complexity:** Increasing the depth of the CNN architecture or incorporating attention mechanisms could improve detection accuracy for overlapping attacks.
- **Continual Learning:** The system could benefit from continual learning, where the model updates itself with new data over time, improving its ability to detect evolving attack patterns.

In summary, the CNN-based IDS demonstrated robust performance in detecting and classifying various types of network attacks, especially common ones like DDoS and DoS. While the system effectively handled benign traffic and high-frequency attacks, there is potential for improving the detection of rare attacks through more sophisticated data balancing techniques and model enhancements.The real-time detection capabilities, combined with the user-friendly interface, make this system a valuable tool for network security, enabling swift detection and response to network intrusions. Further iterations could improve its adaptability and precision, particularly for emerging or rare threats.

V. Conclusion

The proposed Convolutional Neural Network (CNN)-based Intrusion Detection System (IDS) demonstrated significant potential in improving real-time network security through accurate detection and classification of various attack types. The model successfully handled the imbalanced dataset by using undersampling techniques, which enhanced its ability to detect frequent attacks such as DoS, DDoS, and PortScan. Although there were challenges in detecting rare attacks like Heartbleed and Infiltration due to limited representation, the overall system provided strong performance across common threats. The system's real-time detection capabilities, combined with its user-friendly interface, make it a practical and scalable solution for monitoring and protecting network environments. Moving forward, incorporating techniques such as advanced oversampling methods and continuous learning could further improve its adaptability to emerging threats and rare attack detection, making it a valuable tool in cybersecurity.

VI. Future work

To further enhance the effectiveness of the Intrusion Detection System (IDS), future work could focus on addressing several key areas. First, implementing advanced oversampling techniques such as SMOTE or Generative Adversarial Networks (GANs) could improve the detection of rare attacks like Heartbleed and Infiltration, mitigating the impact of class imbalance. Additionally, exploring deeper neural network architectures, such as Long Short-Term Memory (LSTM) networks, could enhance the system's ability to detect evolving attack patterns in time-series network data. Integrating continual learning mechanisms would allow the model to adapt dynamically to new types of attacks as they emerge, improving long-term accuracy. Finally, deploying the IDS in a distributed environment, using frameworks like Apache Kafka for handling large-scale network traffic in real-time, could ensure scalability for enterprise-level applications. These improvements would strengthen the system's robustness, making it even more effective for modern and evolving cybersecurity challenges.

References

1. Thirimanne, S. P., Jayawardana, L., Yasakethu, L., Liyanaarachchi, P., & Hewage, C. (2022). Deep neural network based real-time intrusion detection system. SN Computer Science, 3(2), 145.

2. Kim, T., & Pak, W. (2022). Real-time network intrusion detection using deferred decision and hybrid classifier. Future Generation Computer Systems, 132, 51-66.

3. Duo, R., Nie, X., Yang, N., Yue, C., & Wang, Y. (2021). Anomaly detection and attack classification for train real-time ethernet. IEEE Access, 9, 22528-22541.

4. Vishwakarma, M., & Kesswani, N. (2022). DIDS: A Deep Neural Network based real-time Intrusion detection system for IoT. Decision Analytics Journal, 5, 100142.

5. Hattarki, R., Houji, S., & Dhage, M. (2021, April). Real time intrusion detection system for IoT networks. In 2021 6th International conference for convergence in technology (I2CT) (pp. 1-5). IEEE.

6. Chowdhury, R., Sen, S., Roy, A., & Saha, B. (2022). An optimal feature based network intrusion detection system using bagging ensemble method for real-time traffic analysis. Multimedia Tools and Applications, 81(28), 41225-41247.

7. Kandhro, I. A., Alanazi, S. M., Ali, F., Kehar, A., Fatima, K., Uddin, M., & Karuppayah, S. (2023). Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures. IEEE Access, 11, 9136-9148.

8. Rokade, M. D., & Sharma, Y. K. (2021, March). MLIDS: a machine learning approach for intrusion detection for real time network dataset. In 2021 International conference on emerging smart computing and informatics (ESCI) (pp. 533-536). IEEE.

9. Kim, T., & Pak, W. (2021). Hybrid classification for high-speed and high-accuracy network intrusion detection system. IEEE Access, 9, 83806-83817.

10. Yu, K., Nguyen, K., & Park, Y. (2022). Flexible and robust real-time intrusion detection

systems to network dynamics. IEEE Access, 10, 98959-98969.

11. Garcia, J. F. C., & Blandon, G. E. T. (2022). A deep learning-based intrusion detection and preventation system for detecting and preventing denial-of-service attacks. IEEE Access, 10, 83043-83060.

12. Zhang, L., Cushing, R., de Laat, C., & Grosso, P. (2021, October). A real-time intrusion detection system based on OC-SVM for containerized applications. In 2021 IEEE 24th international conference on computational science and engineering (CSE) (pp. 138-145). IEEE.

13. Yang, L., Song, Y., Gao, S., Hu, A., & Xiao, B. (2022). Griffin: Real-time network intrusion detection system via ensemble of autoencoder in SDN. IEEE Transactions on Network and Service Management, 19(3), 2269-2281.

14. Roshan, K., Zafar, A., & Haque, S. B. U. (2024). Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Computer Communications, 218, 97-113.

15. Shaikh, A., & Gupta, P. (2022). Real-time intrusion detection based on residual learning through ResNet algorithm. International Journal of System Assurance Engineering and Management, 1-15.