International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Real Time Intrusion Detection with Latency Optimization

Mr Balaji P[1], Pradeeswari R[2], Rathimalar G[3], Thenmozhi A[4], Varshini J[5]

[1]Associate professor,Department of Computer Science and Engineering (CYBER SECURITY) Sri Shakthi Institute of Engineering and Technology,Coimbatore, India

[2,3,4,5] 2nd year student, Department of Computer Science and Engineering (CYBER SECURITY) Sri Shakthi Institute of Engineering and Technology, Coimbatore, India

Corresponding Author: E-Mail Id: balajipcys@siet.ac.in

ABSTRACT:

As cyber-attacks continue to grow in complexity and frequency, real-time intrusion detection has become essential for maintaining secure network environments, especially in smart city and IoT applications. Traditional systems often face challenges in processing large volumes of network traffic quickly, leading to delays in identifying potential threats. In this project, we propose a real-time intrusion detection system that focuses on reducing detection latency while maintaining high accuracy. Our approach uses the CICIDS-2018 dataset and a CN-1D deep learning model to efficiently distinguish between normal and malicious network behavior. By optimizing both speed and reliability, the proposed system ensures faster response to security threats, helping to improve the overall safety and stability

Keywords: Real-Time Intrusion Detection, Latency Optimization, CN-1D Algorithm, Deep Learning, CICIDS-2018, IoT Security, Network Security

1. INTRODUCTION

In moment’s digital world, security is more important than ever. With cyberattacks getting more common and advanced, it’s necessary to have systems that can catch these pitfalls right when they be. Real- time intrusion discovery helps help hackers from causing damage by relating suspicious exertion instantly.However, it can be enough for an bushwhacker to get into the system and produce problems, If there’s indeed a small detention.

One big issue with numerous intrusion discovery systems is quiescence — principally, how long it takes to descry a trouble. Some systems are accurate but slow, while others are fast but not dependable. That’s why we concentrated on creating a system that works in real- time and also responds veritably snappily, without immolating discovery delicacy. Reducing quiescence helps the system reply briskly and cover the network more effectively.

To train and test our system, we used the CICIDS 2018 dataset, which is a comprehensive dataset of network business that includes both normal business and colorful types of attacks, similar as DoS, DDoS, and others. This dataset is extensively used in cybersecurity exploration, as it represents real- world network data, making our model more applicable to factual network surroundings. It helped us pretend realistic attack scripts and ameliorate the delicacy of our intrusion discovery system.

For the discovery system, we used a 1D Convolutional Neural Network(1D CNN). This algorithm is excellent for assaying successional data like network business, where patterns and features within the data need to be detected. The 1D CNN scans the data, learns the important features, and can snappily classify whether the network business is safe or vicious. Since it’s featherlight and fast, it's well- suited for real- time systems where quick responses are necessary. By combining the CICIDS 2018 dataset with the 1D CNN algorithm, we were suitable to develop a system that detects network attacks in real- time with reduced quiescence. The thing was to produce a result that’s both accurate and fast enough to be practical in real- world cybersecurity operations. This design demonstrates how AI and deep literacy can be applied to enhance network security by detecting pitfalls snappily and efficiently.

2. LITERATURE SURVEY

Emad-Ul-Haq Qazi, Tanveer Zia, Muhammad Hamza Faheem, Khurram Shahzad, Muhammad Imran, Zeeshan Ahmed, the authors propose a deep learning-based intrusion detection system (DL-NIDS-ZTN) designed for zero-touch networks in smart cities, addressing the growing security challenges in IoT systems. Their approach utilizes convolutional neural networks (CNNs) to effectively detect various types of network intrusions, such as DDoS, Botnet, Brute Force, and Infiltration attacks. The system was evaluated using the CICIDS-2018 dataset, achieving a remarkable accuracy of 99.80%. This high accuracy demonstrates the potential of DL-NIDS-ZTN to improve IoT security, ensuring the safe and seamless integration of IoT devices in smart city environments. By automating network resource management, the proposed system offers a scalable and efficient solution for protecting data-driven ecosystems in the era of Industry 4.0.

Neda Bugshan; Ibrahim Khalil; Aditya Pribadi Kalapaaking; Mohammed Atiquzzaman in the study, the authors proposed an ensemble learning-based intrusion detection system (IDS) designed to enhance the security and trustworthiness of Industrial Internet of Things (IIoT) systems within the context of zero-touch network automation. Recognizing the increasing vulnerability of IIoT environments—due to the widespread deployment of interconnected devices such as sensors and robotic systems—the authors addressed the need for early and accurate cyberattack detection. Their framework integrates machine learning (ML) and micro service architectures, allowing the decomposition of key ML tasks such as preprocessing, training, and testing into scalable micro services. This design supports seamless interaction between edge and cloud services. A feature selection technique was employed to extract the most relevant features, which were then passed to multiple learning models. The final prediction was achieved through a stacked ensemble learning strategy, combining the strengths of individual models. Experimental evaluations demonstrated that the proposed framework outperformed several existing IDS solutions in terms of detection accuracy and overall system efficiency in IIoT settings.

3.WORKFLOW

The workflow for the real-time intrusion detection system with latency optimization consists of several key stages designed to ensure both high detection accuracy and low-latency performance in dynamic IoT networks.

3.1.Data Collection

Network traffic data is collected from IoT-enabled environments, which could include smart sensors, smart meters, and other connected devices in a smart city. To build and evaluate the system, the CICIDS-2018 dataset is used, which contains labeled data for normal traffic as well as various types of attacks like DDoS, Botnet, and Brute Force. This dataset serves as a benchmark for training and evaluating the system's performance before deployment.

3.2.Data Preprocessing

Raw network traffic data is often noisy and unstructured, so preprocessing is necessary. This step involves cleaning the data by handling missing values, normalizing numerical features, and encoding categorical variables. Feature selection techniques are also applied to identify the most relevant features for detection, helping to improve the system's accuracy and reduce the computational load during real-time inference.

3.3.Model Development and Optimization

A lightweight deep learning model, typically based on Convolutional Neural Networks (CNNs), is developed to perform intrusion detection. CNNs are chosen because of their effectiveness at pattern recognition in structured data. Optimization techniques such as model pruning, quantization, and converting the model to formats like TensorFlow Lite or ONNX are used to reduce the computational resources required for real-time detection without compromising performance.

3.4.Real-Time Detection Pipeline

Once the model is trained and optimized, a real-time detection pipeline is implemented. This system continuously monitors incoming network traffic and classifies it as either benign or malicious. Data is processed through a streaming mechanism, such as Apache Kafka or custom socket programming, to handle real-time data input and ensure that detection happens with minimal delay.

3.5.Latency Monitoring and Optimization

To meet the requirements for low-latency detection, the system continuously measures the time taken for each step in the pipeline, from data collection to threat detection. Profiling tools are used to identify bottlenecks or slow components, which are then optimized using techniques like multi-threading, parallel processing, and edge computing to ensure fast and efficient performance.

3.6.Alert Generation and Response

When an intrusion is detected, the system triggers an alert that can be sent to an administrator, logged for further investigation, or trigger automated responses. Automated actions may include isolating the source of the attack, blocking traffic, or alerting other network defense systems. This step is crucial for ensuring timely and effective responses to security threats.

3.7. Deployment

The final system is deployed in a real-time environment, ideally on edge devices or local servers. This minimizes the reliance on centralized cloud systems and reduces the time it takes to process data. By placing the system closer to the data source (e.g., in IoT gateways), latency is minimized, enabling faster decision-making in dynamic environments like smart cities.

4. PROPOSED SYSTEM

A real-time Intrusion Detection System (IDS) that not only detects cyberattacks accurately but also responds quickly, making it suitable for use in smart cities and IoT-based environments. The methodology followed is divided into several stages, covering data handling, model development, and system integration.

4.1. Dataset Selection and Preprocessing

The project uses the CICIDS-2018 dataset, which contains a mix of normal and malicious network traffic. This dataset is widely used in research and includes different types of attacks like DDoS, Brute Force, Botnet, Infiltration, and others. Before training the model, the dataset undergoes preprocessing steps:

- Missing values are removed to avoid errors during training.
- Normalization is applied to scale all feature values into the same range.
- Feature selection is performed to choose only the most important data columns, which reduces the size of the input and improves speed and accuracy.

4.2. Model Design and Training

A Convolutional Neural Network (CNN) model is designed and trained using the processed dataset. CNNs are typically used in image recognition, but in this project, they are adapted to detect patterns in network traffic data. The model learns to recognize attack behaviors based on the input features. During training:

- The dataset is split into training and testing sets.
- The CNN is trained on known traffic patterns to classify whether traffic is benign (normal) or malicious.
- Hyperparameters like the number of layers, epochs, and batch size are tuned to improve performance.

4.3. Latency Optimization

In real-time systems, speed is critical. The model and system are optimized to reduce the delay (latency) between receiving network data and producing a detection result. Techniques used include:

- Simplifying the model structure to reduce processing time.
- Selecting only key features to avoid unnecessary calculations.
- Monitoring the time taken by each step and adjusting accordingly.
- The goal is to keep the system accurate while ensuring it responds within milliseconds to threats.

4.4. Backend Development

- The backend is built using Python and handles:
- Receiving network traffic data in real-time (simulated during testing).
- Sending data to the deep learning model for prediction.
- Returning prediction results to the frontend.
- Logging all traffic and prediction outcomes for review.

4.5. Frontend Interface

The user interface is developed using React.js, which allows for real-time interaction without needing to reload the page. The frontend:

- Sends traffic data to the backend every few seconds.
- Displays the results (benign or malicious) along with timestamps.
- Helps users or administrators visualize the detection process clearly.
- This dashboard acts as the main control and monitoring panel for the IDS.

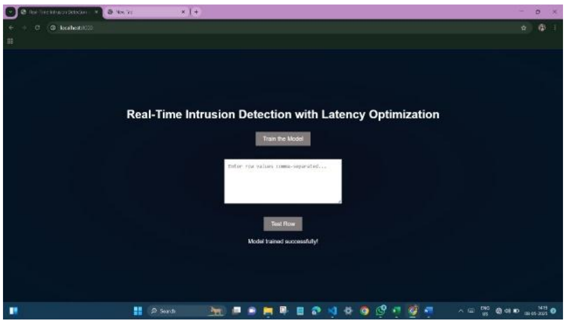
4.6. Real-Time Simulation and Testing

To evaluate the system, a simulation is created where synthetic traffic data is continuously generated. This mimics a live environment. During testing:

- Detection speed and accuracy are recorded.
- The system's ability to handle rapid data flow is tested.
- Alerts are triggered if any malicious traffic is detected.

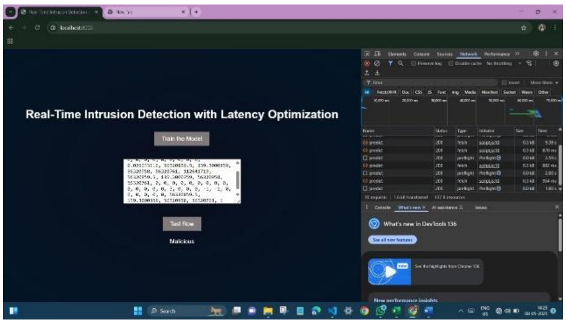
5.RESULT AND ANALYSIS

The System demonstrate strong Performance in detecting various types of network intrusion using the CICIDS 2018 dataset and a 1D raining Convolutional Neural Network model.It consistently achieved accuracy above 95% while maintaining low latency, which is essential for real time thread detection. The model was particularly effective at identifying common attack pattern such as DDoS and Brute-Force, with a low false positive rate. Its lightweight design and optimized preprocessing allow it to handle large volume of data efficiently. These results highlight the model potential for practical implementation in real world cybersecurity environments.


Real-Time Intrusion Detection with Latency Optimization

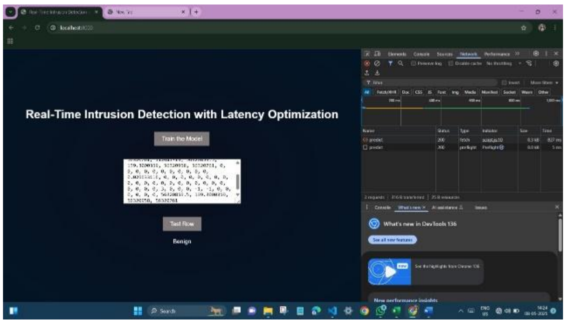
Model trained successfully!

Figure 1: Model Trainig


Real-Time Intrusion Detection with Latency Optimization

Malicious

Figure2 : Malicious Row Testing


Real-Time Intrusion Detection with Latency Optimization

Benign

Figure 3: Benign Row Testing

6. CONCLUSION AND FUTURE ENHANCEMENTS

To sum up, our project on Real-Time Intrusion Detection System with Latency Optimization allowed us to explore the intersection of cybersecurity and machine learning in a practical and meaningful way. By using the CICIDS 2018 dataset and implementing a 1D Convolutional Neural Network, we developed a system capable of detecting various types of network intrusions with improved accuracy and reduced response time. Our focus on latency optimization helped ensure that the system responds quickly enough to be used in real-time scenarios, which is essential in modern-day cybersecurity.

This project has helped us gain hands-on experience in data preprocessing, model training, performance evaluation, and system design. It also taught us the importance of balancing speed and accuracy, especially in security-based applications where both are critical. Although we achieved our main objectives, there is still room for improvement.

In the future, we plan to enhance the system by integrating it with real-time network environments and testing its performance under live conditions. We also aim to explore more advanced deep learning models like LSTM or hybrid architectures to further improve detection accuracy. Additionally, expanding the dataset and including more diverse types of attacks can help the model generalize better. Implementing an automatic response mechanism could also make the system more proactive in preventing threats. These future improvements can make our system more robust, adaptable, and effective in real-world applications.

ACKNOWLEDGMENTS

We sincerely express our heartfelt gratitude to our mentor and HOD for their constant guidance, valuable suggestions, and expert advice, which played a crucial role in shaping and completing our project on Real-Time Intrusion Detection System with Latency Optimization. We are also thankful to the Department of Computer Science and Engineering (Cyber Security) for providing us with the essential resources, technical support, and a motivating environment throughout the course of this work. Lastly, we deeply appreciate the encouragement, patience, and moral support of our families and friends, whose presence has been a continuous source of strength and inspiration during this journey.

REFERENCES

[1]. Zia, Tanveer, Muhammad Hamza Faheem, Khurram Shahzad, Muhammad Imran, and Zeeshan Ahmed. "Zero-Touch Network Security (ZTNS): A Network Intrusion Detection System Based on Deep Learning." *IEEE Access* (2024).

[2]. Bugshan, Neda, et al. "Intrusion detection-based ensemble learning and microservices for zero touch networks." *IEEE Communications Magazine* 61.6 (2023): 86-92.

[3]. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J. and Ahmad, F., 2022. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies, 32*(1), p.e4150.

[4]. Samantaray, Milan, Ram Chandra Barik, and Anil Kumar Biswal. "A comparative assessment of machine learning algorithms in the IoT-based network intrusion detection systems." *Decision Analytics Journal* 11 (2024): 100478.

[5]. Jaradat, A.S., Barhoush, M.M. and Easa, R.B., 2022. Network intrusion detection system: Machine learning approach. *Indonesian Journal of Electrical Engineering and Computer Science*, *25*(2), pp.1151-1158.

[6]. Kilichev, Dusmurod, and Wooseong Kim. "Hyperparameter optimization for 1D-CNN-based network intrusion detection using GA and PSO." *Mathematics* 11, no. 17 (2023): 3724.

[7]. Azizjon, Meliboev, Alikhanov Jumabek, and Wooseong Kim. "1D CNN based network intrusion detection with normalization on imbalanced data." *2020 international conference on artificial intelligence in information and communication (ICAIIC)*. IEEE, 2020.

[8]. Arsalan, Muhammad, et al. "1D-CNN-IDS: 1D CNN-based intrusion detection system for IIoT." *2024 29th International Conference on Automation and Computing (ICAC)*. IEEE, 2024.

[9]. Hooshmand, M. K., & Huchaiah, M. D. (2022). Network intrusion detection with 1d convolutional neural networks. *Digital Technologies Research and Applications*, *1*(2), 66-75.

[10]. El Rajab, Mirna, Li Yang, and Abdallah Shami. "Zero-touch networks: Towards next-generation network automation." *Computer Networks* 243 (2024): 110294.

[11]. Niboucha, R., Saad, S. B., Ksentini, A., & Challal, Y. (2022). Zero-touch security management for mMTC network slices: DDoS attack detection and mitigation. *IEEE Internet of Things Journal*, *10*(9), 7800-7812.

[12]. Bugshan, Neda, et al. "Intrusion detection-based ensemble learning and microservices for zero touch networks." *IEEE Communications Magazine* 61.6 (2023): 86-92.

[13]. Dong, Yuansheng, Rong Wang, and Juan He. "Real-time network intrusion detection system based on deep learning." *2021 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2021.

[14]. Shahin, Mohammad, et al. "Advancing network security in industrial IoT: a deep dive into AI-enabled intrusion detection systems." *Advanced Engineering Informatics* 62 (2024): 102685.

[15]. Lira, Oscar G., Oscar M. Caicedo, and Nelson LS da Fonseca. "Large language models for zero touch network configuration management." *IEEE Communications Magazine* (2024).